TRẦN THẾ SAN - TS. NGUYỄN NGỌC PHƯƠNG
KHOA CƠ KHÍ CHẾ TẠO MÁY
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM

SỔ TAY LẬP TRÌNH

CNC

THỰC HÀNH - LẬP TRÌNH GIA CÔNG TRÊN MÁY CNC

http://www.me-cad.com/

- MÁY TIỆN CNC
- MÁY PHAY CNC
- TRUNG TÂM GIA CÔNG CNC

Ấn Bản MỚI

NHÀ XUẤT BẢN ĐÀ NẴNG

Nhiều tài liệu hay có tại : http://www.me-cad.com/

SỔ TAY LẬP TRÌNH CNC

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

TRẦN THẾ SAN — TS. NGUYỄN NGỌC PHƯƠNG
KHOA CƠ KHÍ CHẾ TẠO MÁY
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP HỒ CHÍ MINH

SỔ TAY LẬP TRÌNH CNC

THỰC HÀNH LẬP TRÌNH GIA CÔNG TRÊN MÁY CNC

http://www.me-cad.com/

NHÀ XUẤT BẢN ĐÀ NẴNG

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

SỔ TAY LẬP TRÌNH CNC

TRẦN THẾ SAN - NGUYỄN NGỌC PHƯƠNG

Chịu trách nhiệm xuất bản
Giám Đốc: **VÕ VĂN ĐÁNG**
Tổng biên tập: **NGUYỄN ĐỨC HÙNG**

Biên tập: **TRẦM MY**
Trình bày: **DUY TRẦN**
Vẽ bìa: **DUY TRẦN**

http://www.me-cad.com/

Liên kết xuất bản

C.TY VĂN HÓA **TRÍ DÂN** - HS. **NGUYỄN TRÃI**
96/7 Duy Tân - P.15 - Q. Phú Nhuận - Tp. HCM
ĐT: 8383669 - 9901846 - Fax: 9971765

In 1000 cuốn khổ (19 X 27 Cm) tại xưởng in trung tâm hội chợ triển lãm Việt Nam. Theo TNKH Số 09/1386/XB-QLXB. Cục xuất bản cấp ngày 19-8-2005. Số 99 / QĐ ĐaN Nhà xuất bản cấp ngày 17-02-2006 .In xong nộp lưu chiểu tháng 06 năm 2006

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lời nói đầu

CNC là bước phát triển kế tiếp của máy công cụ trong thời đại công nghệ thông tin. Nói chung, các nguyên lý và phương pháp gia công cắt gọt trên máy công cụ cổ điển (tiện, phay, bào, khoan, khoét, doa, gia công ren,...) và trên máy CNC hầu như không thay đổi, chủ yếu dựa trên chuyển động của phôi (chi tiết) và chuyển động của dụng cụ cắt theo hệ quy chiếu được chọn trước. Khác biệt cơ bản giữa công nghệ gia công cổ điển và công nghệ CNC là hệ thống điều khiển. Công nghệ cổ điển thường áp dụng điều khiển bằng cơ cấu cam, các relay, và một số mạch điều khiển đơn giản, còn công nghệ CNC áp dụng điều khiển bằng chương trình máy tính.

Để điều khiển hiệu quả trên máy CNC cần phải viết chương trình gia công. Bạn có thể viết chương trình gia công bằng tay sử dụng giấy và bút, hoặc viết trên máy tính sử dụng phần mềm CAD/CAM. Về nguyên tắc, bạn phải nắm vững các nguyên lý viết chương trình, sử dụng thành thạo các kỹ thuật và công cụ lập trình bằng tay trước khi viết chương trình trên máy tính. Cuốn sách này hướng dẫn chi tiết các phương pháp lập trình gia công chi tiết trên máy công cụ CNC (phay, tiện, doa, cắt ren,...), trên cơ sở giả thiết bạn đọc đã có đủ kiến thức về vẽ kỹ thuật, nguyên lý và chi tiết máy, vật liệu cơ khí, công nghệ gia công cắt gọt, máy công cụ,...

Gia công cắt gọt trên máy CNC là công nghệ hiện đại và liên tục phát triển, nhưng không quá khó như suy nghĩ của một số người. Công nghệ này chỉ đòi hỏi kiến thức cơ bản về chế tạo máy và tính sáng tạo của người sử dụng.

Cuốn sách này là tài liệu hướng dẫn cơ bản và hệ thống, bao quát hầu như mọi vấn đề về lập trình bằng tay, bao gồm hệ thống điều khiển, quy hoạch lập trình, cấu trúc chương trình, các hàm và các lệnh (mã G, mã M,...), các chu kỳ gia công, các chế độ bù, chương trình con,... và các ví dụ lập trình cụ thể. Bạn đọc là kỹ sư cơ khí, nhà quản lý, sinh viên,... sẽ tìm thấy ở đây nhiều kiến thức bổ ích và thú vị. Đây cũng là tài liệu tham khảo cho mọi người quan tâm đến lĩnh vực gia công hiện đại trên hệ thống CNC.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

1 ĐIỀU KHIỂN SỐ

Công nghệ ***điều khiển số*** xuất hiện vào giữa thế kỷ 20, khoảng năm 1952, nhưng cho đến đầu những năm 1960 vẫn chưa được áp dụng trong sản xuất hàng loạt. Sự bùng nổ thực tế ở dạng CNC bắt đầu từ năm 1972, và thập kỷ kế tiếp với sự xuất hiện của máy tính.

Trong lĩnh vực sản xuất, đặc biệt là gia công kim loại, công nghệ điều khiển số đã góp phần dẫn đến cuộc cách mạng. Ngay cả trước khi máy tính trở thành thiết bị không thể thiếu trong mọi công ty và từng gia đình, máy công cụ được trang bị hệ thống điều khiển số đã có vị trí đặc biệt trong xưởng cơ khí. Sự phát triển của vi điện tử và máy tính, gồm cả tác động của chúng đối với điều khiển số, đã đem đến các thay đổi cơ bản trong sản xuất nói chung và trong ngành cơ khí nói riêng.

ĐỊNH NGHĨA ĐIỀU KHIỂN SỐ

Trong nhiều ấn phẩm, các năm qua đã đưa ra hàng loạt định nghĩa về điều khiển số. Hầu hết các định nghĩa đó đều có cùng ý tưởng, cùng khái niệm cơ bản, chỉ khác nhau về từ ngữ.

Đa số các định nghĩa có thể được tổng kết như sau:

Điều khiển số có thể được định nghĩa là sự vận hành máy công cụ bằng cách dùng các lệnh mã hóa đặc biệt cho hệ thống điều khiển máy.

Các lệnh là sự phối hợp các chữ cái, chữ số và ký hiệu được chọn, ví dụ, dấu thập phân, dấu phần trăm, dấu ngoặc. Tất cả các lệnh đều được viết với thứ tự logic theo dạng cho trước. Tập hợp *tất cả* các lệnh cần thiết để gia công chi tiết được gọi là *chương trình NC*, *chương trình CNC*, hoặc *chương trình gia công*. Chương trình có thể được lưu để sử dụng trong tương lai hoặc tái sử dụng để đạt được các kết quả gia công đồng nhất vào thời điểm bất kỳ.

CÔNG NGHỆ NC VÀ CNC

Về thuật ngữ, các viết tắt *NC* và *CNC* có sự khác biệt về ý nghĩa. *NC* là viết tắt của công nghệ *điều khiển số* (Numerical Control), còn *CNC* là viết tắt của công nghệ *điều khiển số máy tính hóa* (Conputerized Numerical Control), là sự phát triển cao hơn của NC. Tuy nhiên, trong thực tiễn, *CNC* được dùng rộng rãi hơn. Để làm rõ công dụng của từng thuật ngữ, ở đây sẽ nêu ra các khác biệt chính giữa hệ thống NC và CNC.

Cả hai hệ thống đều thực hiện các tác vụ như nhau, xử lý các dữ liệu với mục đích gia công chi tiết. Trong cả hai trường hợp, thiết kế bên trong của hệ thống điều khiển chứa các lệnh logic để xử lý dữ liệu. Đó là sự giống nhau giữa hai hệ thống.

Hệ thống NC (khác với hệ thống CNC) sử dụng các hàm logic cố định, được xây dựng sẵn và được nối mạch bên trong bộ điều khiển. Nhà lập trình hoặc người vận hành không thể thay đổi các lệnh này. Do sự nối mạch cố định của logic điều khiển, hệ thống điều khiển NC đồng nghĩa với thuật ngữ *"mạch cố định"*. Hệ thống có thể diễn dịch chương trình chi tiết, nhưng không cho phép thay đổi chương trình, sử dụng các tính năng điều khiển. Mọi thay đổi đều phải được thực hiện bên ngoài hệ thống điều khiển, thường là trong môi trường văn phòng. Ngoài ra hệ thống NC đòi hỏi bắt buộc sử dụng băng đục lỗ để nhập thông tin chương trình.

Hệ thống CNC hiện đại, khác với hệ thống NC cũ, sử dụng bộ vi xử lý bên trong (ví dụ máy tính). Máy tính này có các thanh ghi bộ nhớ lưu các chương trình con có khả năng thực hiện các hàm logic. Điều đó có nghĩa là nhà lập trình hoặc người vận hành máy có thể thay đổi chương trình ngay trên bộ điều khiển (lắp trong máy), với các kết quả tức thời. Tính linh hoạt này là ưu thế lớn nhất của hệ thống CNC và có lẽ là yếu tố quyết định, góp phần vào sự ứng dụng rộng rãi công nghệ này trong sản xuất hiện đại. Các chương trình CNC và các hàm logic được lưu trên các vi mạch máy tính đặc biệt, dưới dạng các *lệnh phần mềm*, thay vì được nối kết cứng, sử dụng chẳng hạn các dây, điều khiển cái hàm logic đó. Khác với hệ thống NC, hệ thống CNC đồng nghĩa với thuật ngữ *"mạch linh hoạt"*.

Khi nói về chủ đề nào đó liên quan với công nghệ điều khiển số, nói chung có thể sử dụng thuật ngữ NC hoặc CNC. Bạn cần nhớ NC có thể có nghĩa là CNC trong nói chuyện hàng ngày, nhưng CNC không có ý nghĩa là NC. Mọi hệ thống điều khiển ngày nay đều là thiết kế CNC.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

GIA CÔNG CNC VÀ GIA CÔNG CỔ ĐIỂN

Điều gì làm cho gia công CNC vượt trội so với các phương pháp cổ điển? Đâu là các ưu điểm chính? Nếu so sánh giữa CNC và gia công cổ điển, bạn sẽ thấy sự tiếp cận trong khi gia công chi tiết gồm các bước:

1. Nhận và nghiên cứu bản vẽ.
2. Chọn phương pháp gia công thích hợp nhất
3. Quyết định phương pháp kẹp chặt (định vị chi tiết gia công)
4. Chọn dụng cụ cắt.
5. Thiết lập chế độ cắt và lượng ăn dao.
6. Gia công chi tiết.

Sự tiếp cận cơ bản này là như nhau đối với cả hai kiểu gia công. Sự khác biệt chính là phương pháp nhập dữ liệu. Tốc độ ăn dao 10 inch/phút (10 in/min) là như nhau, trong CNC và gia công cổ điển, nhưng phương pháp áp dụng khác nhau. Điều đó cũng có thể nói về chất làm nguội, kích hoạt bằng cách xoay nút, nhấn công tắc, hoặc lập trình mã đặc biệt. Tất cả đều làm cho chất làm nguội phun ra từ vòi phun. Trong cả hai dạng gia công, người dùng đều phải có kiến thức về chi tiết gia công. Cuối cùng, gia công kim loại, đặc biệt là cắt gọt, chủ yếu là kỹ năng, và ở mức độ lớn còn là nghệ thuật và nghề nghiệp của nhiều người. Đó là sự ứng dụng *Điều khiển số máy tính hóa*. Tương tự mọi ngành nghề khác, việc nắm vững CNC đến từng chi tiết là cần thiết để đi đến thành công. Không chỉ biết kiến thức kỹ thuật, bạn cần có kinh nghiệm và trực giác, đôi khi được gọi là *"giác quan thứ sáu"* bổ sung cho kỹ năng.

Trong gia công cổ điển, người vận hành điều chỉnh máy và dịch chuyển từng dụng cụ cắt, sử dụng một hoặc cả hai tay, để gia công chi tiết theo yêu cầu. Thiết kế của máy công cụ cổ điển có nhiều tính năng hỗ trợ sự gia công chi tiết, các cần gạt, tay quay, bánh răng, đĩa chia độ, ... Người vận hành thực hiện các động tác như nhau cho từng chi tiết cùng loại. Tuy nhiên các động tác *"như nhau"*, thực tế được hiểu là *"tương tự nhau"* thay vì *"đồng nhất"*. Con người không có khả năng lặp lại chính xác từng động tác, đây là công việc của máy móc. Con người không thể làm việc với cùng mức năng suất ở mọi thời điểm mà không nghỉ ngơi. Mọi người đều có các thời điểm "tốt" và thời điểm "xấu". Kết quả của các thời điểm này, khi được áp dụng để gia công chi tiết, rất khó dự đoán, do đó sẽ có các khác biệt trong loạt chi tiết cùng loại. Các chi tiết đó không hoàn toàn như nhau. Duy trì dung sai kích thước và độ bóng bề mặt là các vấn đề cơ bản trong gia công cổ điển. Thợ cơ khí có thể có các phương pháp khác nhau tùy theo kinh nghiệm và cảm nhận riêng của họ. Kết hợp các yếu tố này và nhiều yếu tố khác dẫn đến sự bất định khá lớn giữa các chi tiết gia công.

Gia công với sự điều khiển số sẽ cho phép loại bỏ hầu hết các yếu tố dẫn đến sự bất định. Phương pháp này không đòi hỏi sự tham gia trực tiếp của con người như trong gia công cổ điển. Sự gia công điều khiển số không cần các đĩa chia độ, cần gạt, tay quay, ... như trong gia công cổ điển. Khi chương trình chi tiết đã được chứng minh, có thể sử dụng với số lần không hạn chế, luôn luôn dẫn đến kết quả đồng nhất. Điều đó không có nghĩa là không còn các yếu tố giới hạn. Dụng cụ cắt mòn dần, vật liệu phôi có thể không đồng nhất, định vị có thể thay đổi ... Các yếu tố này cần được xem xét và xử lý mỗi khi có yêu cầu.

Sự xuất hiện của công nghệ điều khiển số không có nghĩa là sẽ kết thúc các máy công cụ thao tác bằng tay. Đôi khi phương pháp gia công cổ điển có ưu thế hơn so với điều khiển số. Ví dụ, sản xuất đơn chiếc hoặc số lượng ít có thể hiệu quả hơn trên máy công cụ cổ điển so với máy CNC. Một số nguyên công cũng có thể được thực hiện trên máy cổ điển hoặc bán tự động hiệu quả hơn gia công điều khiển số. Máy công cụ CNC không thay thế cho máy cổ điển, chỉ bổ sung cho chúng.

Trong nhiều trường hợp, quyết định có gia công trên máy CNC hay không, hoàn toàn dựa trên số lượng chi tiết cần gia công. Tuy số lượng chi tiết gia công theo loạt luôn luôn là chuẩn quan trọng nhưng không phải là yếu tố duy nhất. Sự xem xét còn bao gồm độ phức tạp của chi tiết, dung sai, độ bóng bề mặt... Đôi khi, gia công CNC một chi tiết phức tạp sẽ có lợi, nhưng gia công CNC 50 chi tiết tương đối đơn giản lại không hiệu quả bằng gia công cổ điển.

Bạn cần nhớ, điều khiển số không phải là gia công chi tiết. Điều khiển số chỉ là quy trình hoặc phương pháp cho phép sử dụng máy công cụ theo phương thức chính xác, năng suất, và ổn định.

ƯU THẾ CỦA ĐIỀU KHIỂN SỐ

Các ưu thế chính của điều khiển số là gì?

Điều quan trọng cần biết là lĩnh vực gia công nào có lợi hơn từ CNC và lĩnh vực nào gia công cổ điển hiệu quả hơn. Bạn không nên nghĩ rằng máy phay CNC công suất 2 hp sẽ gia công nhiều sản phẩm hơn so với máy phay thông dụng có công suất đến 15 hp. Bạn cũng

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

đừng hy vọng các cải thiện rõ rệt về tốc độ cắt và lượng ăn dao so với máy móc cổ điển. Nếu các điều kiện gia công và dao cắt như nhau, thời gian cắt gọt trong cả hai trường hợp sẽ gần như bằng nhau.

Người sử dụng CNC có thể nhận được các cải tiến sau:

- Giảm thời gian xác lập máy.
- Giảm thời gian chuẩn bị.
- Độ chính xác và tính lặp lại.
- Gia công biên dạng các hình phức tạp.
- Đơn giản hóa dụng cụ và định vị chi tiết.
- Thời gian cắt gọt ổn định.
- Tăng năng suất chung.

Mỗi lĩnh vực nêu trên đều chỉ cung cấp khả năng cải thiện. Người dùng sẽ có các mức cải thiện cụ thể, tùy theo loại sản phẩm được chế tạo và loại máy CNC được sử dụng, phương pháp chuẩn bị, độ phức tạp của đồ gá, chất lượng dụng cụ cắt, triết lý quản lý và thiết kế kỹ thuật, trình độ của đội ngũ, thái độ đối với công việc.

Giảm thời gian xác lập máy

Trong nhiều trường hợp, thời gian xác lập máy CNC có thể giảm, đôi khi rất đáng kể. Điều quan trọng cần biết là sự xác lập máy chủ yếu được thực hiện bằng tay, phụ thuộc phần lớn vào tay nghề của người vận hành CNC, kiểu định vị, và thực tiễn tại xưởng cơ khí. Thời gian xác lập máy không thực sự sản xuất, nhưng cần thiết, là một phần trong chi phí tổng thể. Để duy trì thời gian xác lập máy ở mức tối thiểu, cần có các khảo sát và nghiên cứu của quản đốc xưởng, nhà lập trình và công nhân vận hành.

Do thiết kế của máy CNC, thời gian xác lập máy không phải vấn đề chính. Sự định vị theo module, chuẩn hóa dụng cụ cắt, đồ gá chuyên dùng, thay dao tự động, và các tính năng tiên tiến khác, đều làm cho thời gian xác lập máy trở nên hiệu quả hơn so với máy công cụ cổ điển. Với kiến thức về sản xuất hiện đại, năng suất lao động có thể tăng rõ rệt.

Số lượng chi tiết gia công trong một lần xác lập máy cũng rất quan trọng, chủ yếu để giảm chi phí và thời gian xác lập máy. Nếu nhiều chi tiết được gia công trong một xác lập máy, chi phí xác lập / chi tiết sẽ hầu như không đáng kể. Sự giảm tương tự cũng có thể đạt được bằng cách chia nhóm các nguyên công khác nhau theo từng xác lập máy. Thậm chí nếu thời gian xác lập dài hơn, vẫn có thể được xét đến khi so với thời gian cần thiết để xác lập các máy công cụ cổ điển.

Giảm thời gian chuẩn bị

Khi chương trình gia công chi tiết được viết và được chứng minh, bạn có thể sử dụng lại trong tương lai. Dù thời gian chuẩn bị cho lần gia công đầu tiên thường dài hơn, nhưng hoàn toàn không đáng kể trong lần gia công thứ hai. Kể cả nếu sự thay đổi kỹ thuật của chi tiết đòi hỏi chỉnh sửa chương trình cũng chỉ cần thời gian ngắn, do đó giảm thời gian chuẩn bị.

Thời gian chuẩn bị dài, cần thiết để thiết kế và chế tạo đồ gá cho máy cổ điển, thường giảm rõ rệt bằng cách chuẩn bị chương trình (phần mềm) gia công chi tiết và sử dụng đồ gá đơn giản.

Độ chính xác và tính lặp lại

Tính lặp lại và độ chính xác cao của các máy CNC hiện đại là lợi ích cơ bản đối với người dùng. Dù chương trình được lưu trên đĩa, trong bộ nhớ máy tính hay trên băng từ (phương pháp ban đầu), chương trình đó vẫn không thay đổi. Chương trình bất kỳ đều có thể được chỉnh sửa theo ý muốn, nhưng một khi đã được chứng minh, thường không cần thay đổi. Chương trình có thể tái sử dụng với với số lần tùy ý mà không bị mất dữ liệu. Trong thực tế, chương trình cho phép các yếu tố có thể thay đổi, chẳng hạn sự mòn dao, nhiệt độ vận hành, luôn luôn có thể lưu một cách an toàn, nói chung chỉ cần sự can thiệp rất ít của người lập trình hoặc vận hành CNC. Độ chính xác cao của máy CNC và tính lặp lại của chúng cho phép chế tạo các chi tiết chất lượng cao rất ổn định theo thời gian.

Biên dạng các hình dạng phức tạp

Máy tiện CNC và trung tâm gia công có khả năng tạo biên dạng các hình phức tạp. Nhiều người dùng CNC mua máy chỉ để có khả năng gia công các chi tiết phức tạp. Các ví dụ là sự áp dụng CNC trong công nghiệp hàng không và xe hơi. Việc sử dụng lập trình máy tính hầu như là bắt buộc đối với thế hệ dụng cụ cắt chuyển động trong ba chiều không gian.

Các hình dạng phức tạp, chẳng hạn khuôn mẫu, có thể được gia công mà không cần tăng chi phí để chế tạo dưỡng mẫu hoặc đồ gá phức tạp. Các chi tiết có độ bóng như gương có thể đạt được chỉ bằng động tác nhấn nút công tắc. Sự lưu các chương trình đơn giản hơn nhiều so

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

với việc bảo quản các dưỡng mẫu, mô hình bằng gỗ, và các đồ gá mài dao cắt.

Đơn giản hóa dụng cụ và định vị chi tiết

Bạn có thể loại bỏ các dụng cụ cắt phi tiêu chuẩn và các dao cắt "tự chế", dùng cho máy cổ điển, bằng cách sử dụng các dụng cụ cắt tiêu chuẩn được thiết kế đặc biệt cho các ứng dụng điều khiển số. Các dụng cụ cắt nhiều bước, chẳng hạn mũi khoan dẫn hướng, mũi khoan bậc, dụng cụ cắt kết hợp, dao doa biên dạng, ..., đều được thay bằng các dụng cụ cắt tiêu chuẩn. Các dụng cụ này thường rẻ hơn và dễ thay thế hơn so với các dụng cụ cắt đặc biệt và phi tiêu chuẩn. Các biện pháp cắt giảm chi phí buộc các nhà cung cấp dụng cụ cắt phải giảm số sản phẩm dự trữ, tăng thời gian cung ứng cho khách hàng. Dụng cụ cắt tiêu chuẩn thường được cung cấp nhanh hơn loại phi tiêu chuẩn.

Đồ gá và định vị chi tiết trên máy CNC chỉ có một mục đích là giữ cho chi tiết cứng vững và cùng vị trí cho tất cả các chi tiết trong một loạt gia công. Đồ gá được thiết kế để gia công chi tiết trên CNC thường rất đơn giản, không cần các lỗ dẫn hướng, lỗ hoặc thanh định vị.

Thời gian cắt gọt và năng suất

Thời gian cắt trên máy CNC, thường được gọi là thời gian chu kỳ, luôn luôn ổn định. Khác với gia công thông thường, trong đó tay nghề, kinh nghiệm, và sự mệt mỏi của công nhân thường ảnh hưởng đến chất lượng gia công, sự gia công CNC do máy tính điều khiển. Công việc bằng tay rất ít, có lẽ chỉ gồm xác lập máy, cấp phôi và lấy chi tiết. Đối với các loạt gia công lớn (gồm nhiều chi tiết cùng loại), chi phí về thời gian không sản xuất được chia đều cho nhiều chi tiết, do đó chi phí này hầu như không đáng kể. Lợi ích cơ bản của thời gian cắt ổn định chủ yếu xuất phát từ các nguyên công lặp lại nhiều lần, trong đó lịch trình sản xuất và sự phân chia các chi tiết cho từng máy công cụ có thể được thực hiện rất chính xác.

Lý do chính để các công ty đặt mua máy CNC là tính kinh tế, đây là sự đầu tư nghiêm túc, và tính cạnh tranh luôn luôn thường trực trong đầu các nhà quản lý. Công nghệ điều khiển số là phương tiện tuyệt vời để đạt được sự cải thiện rõ rệt về năng suất lao động và tăng chất lượng chung của các chi tiết gia công. Tương tự các phương tiện khác, CNC cần được sử dụng một cách hợp lý với các kiến thức khoa học. Khi công nghệ CNC được sử dụng ngày càng rộng rãi, CNC không còn là ưu thế riêng của vài công ty lớn. Công ty hướng về phía trước là công ty biết cách sử dụng công nghệ này một cách hiệu quả và ứng dụng trong sự cạnh tranh của nền kinh tế toàn cầu.

Để đạt mục đích tăng rõ rệt năng suất lao động, điều quan trọng là người dùng phải hiểu các nguyên lý cơ bản của công nghệ CNC. Các nguyên lý này bao quát nhiều lĩnh vực, chẳng hạn mạch điện tử, sơ đồ bậc thang phức tạp, logic máy tính, điều khiển tự động, thiết kế máy, nguyên lý cắt gọt, ... Trong cuốn sách này, sự nhấn mạnh sẽ tập trung vào các chủ đề liên quan trực tiếp với sự lập trình CNC, và kiến thức về các máy công cụ CNC thông dụng nhất, *Trung tâm gia công* và *máy tiện* (đôi khi còn được gọi là *Trung tâm Tiện*). Sự khảo sát chất lượng chi tiết là rất quan trọng đối với nhà lập trình và người vận hành máy, mục đích này được phản ánh trong nhiều ví dụ cụ thể qua các chương trong cuốn sách này.

CÁC KIỂU MÁY CÔNG CỤ CNC

Hiện có nhiều kiểu máy CNC, số lượng chủng loại máy CNC tăng nhanh cùng với sự phát triển của công nghệ tự động hóa, rất khó liệt kê tất cả các ứng dụng của chúng. Các nhóm máy CNC bao gồm:

- Máy phay và trung tâm gia công.
- Máy tiện và trung tâm tiện.
- Máy khoan.
- Máy doa và gia công biên dạng.
- Máy gia công tia lửa điện.
- Máy dập cắt.
- Máy cắt bằng ngọn lửa.
- Máy hành trình.
- Máy cắt biên dạng bằng laser và tia nước.
- Máy mài trụ.
- Máy hàn.
- Máy uốn,...

Trung tâm gia công CNC và máy tiện được dùng nhiều trong công nghiệp. Hai nhóm máy này có thị phần gần như ngang nhau. Một số ngành có nhu cầu cao hơn về một nhóm máy, tùy theo yêu cầu của họ. Bạn cần nhớ, có nhiều kiểu máy tiện và trung tâm gia công. Tuy nhiên, các bước lập trình cho máy đứng là tương tự máy ngang hoặc máy phay CNC đơn giản. Ngay cả giữa các nhóm máy khác nhau, nhiều ứng dụng tổng quát và qui trình lập trình nói chung là như nhau. Ví dụ, biên dạng được phay với dao phay mặt đầu có nhiều điểm chung với máy cắt biên dạng bằng tia lửa điện.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Máy phay và trung tâm gia công

Số trục tiêu chuẩn trên máy phay là 3, gồm các trục X, Y và Z. Chi tiết trên máy phay hầu như luôn luôn tĩnh tại, được lắp trên bàn máy di động. Dao cắt quay, có thể dịch chuyển lên và xuống (hoặc vào và ra), nhưng không dịch chuyển hoàn toàn theo quỹ đạo cắt.

Máy phay CNC, thường là các máy nhỏ và đơn giản, không có bộ đổi dao hoặc các tính năng tự động hóa, công suất của chúng nói chung tương đối thấp. Trong công nghiệp, chúng được sử dụng trong phân xưởng chế tạo dụng cụ, cho các mục đích bảo trì, hoặc sản xuất hàng loạt nhỏ. Chúng thường được thiết kế để gia công biên dạng, khác với máy khoan.

Trung tâm gia công CNC được sử dụng phổ biến hơn, hiệu quả hơn so với máy khoan và máy phay, do có tính linh hoạt cao. Ưu thế chính của trung tâm gia công CNC đối với người dùng là khả năng chia nhóm các nguyên công khác nhau và cùng một quy trình xác lập máy. Ví dụ, khoan, doa, doa biên dạng, cắt ren, gia công bề mặt và phay biên dạng, đều được tích hợp vào một chương trình CNC. Ngoài ra, tính linh hoạt được tăng cường bằng sự thay dao tự động, sử dụng bộ thay dao để giảm thời gian chạy không, phân độ cho mặt bên của chi tiết, sử dụng chuyển động quay của các trục phụ và nhiều tính năng khác. Trung tâm gia công CNC có thể được trang bị với phần mềm đặc biệt điều khiển tốc độ cắt và lượng ăn dao, tuổi thọ dao cắt, đo tự động trong khi gia công và điều chỉnh độ lệch nếu có, và các tính năng khác cho phép tiết kiệm thời gian và tăng năng suất gia công.

Trung tâm gia công CNC có hai thiết kế cơ bản, *đứng* và *ngang*. Khác biệt chính giữa hai kiểu này là bản chất của nguyên công có thể thực hiện trên máy một cách hiệu quả. Đối với trung tâm gia công CNC *kiểu đứng*, kiểu gia công thích hợp nhất là các chi tiết phẳng, lắp với đồ gá trên bàn máy, hoặc kẹp trong mâm cặp. Trung tâm gia công CNC *kiểu ngang* thường dùng để gia công nhiều bề mặt trên chi tiết. Chẳng hạn vỏ hộp bơm và các hình dạng kiểu khối lập phương. Gia công nhiều bề mặt trên các chi tiết nhỏ cũng có thể được thực hiện trên trung tâm gia công CNC kiểu đứng được trang bị bàn máy quay.

Các bước lập trình là như nhau cho cả hai kiểu thiết kế, nhưng kiểu ngang thường cần có thêm trục phụ (thường là trục B). Đây là trục định vị đơn giản (trục chia độ) cho bàn máy hoặc là trục quay đồng thời gia công biên dạng.

Trong sách này sẽ tập trung vào các ứng dụng trung tâm gia công CNC kiểu đứng, với một phần đặc biệt về xác lập và gia công trên trung tâm kiểu ngang. Các phương pháp lập trình này cũng có thể áp dụng cho máy phay CNC nhỏ, máy khoan và cắt ren, nhưng nhà lập trình cần xem xét giới hạn của các máy đó.

Máy tiện và trung tâm tiện

Máy tiện CNC thường là máy công cụ có hai trục, trục đứng X và trục ngang Z. Tính năng chính của máy tiện khác biệt với máy phay là chi tiết quay xung quanh đường tâm máy. Ngoài ra dụng cụ cắt thường tĩnh tại, lắp trong bàn xe dao di trượt. Dụng cụ cắt chạy theo biên dạng của quỹ đạo dụng cụ cắt được lập trình. Đối với máy tiện CNC có đồ gá phay, được gọi là *dụng cụ cắt sống*, dao phay có động cơ riêng, có thể quay trong khi trục chính tĩnh tại.

Thiết kế máy tiện hiện đại có thể là ngang hoặc đứng. Kiểu ngang phổ biến hơn kiểu đứng, nhưng cả hai thiết kế đều có mục đích riêng trong sản xuất. Mỗi kiểu đều có nhiều dạng thiết kế. Ví dụ máy tiện CNC kiểu ngang có thể được thiết kế với băng máy ngang hoặc băng máy nghiêng, kiểu thanh, kiểu có mâm cặp, kiểu vạn năng. Bổ sung cho các phối hợp này là nhiều loại phụ tùng giúp cho máy tiện CNC trở thành máy công cụ rất linh hoạt. Nói chung, các thiết bị phụ, chẳng hạn ụ động, gối tựa cố định hoặc di động, đồ gá kẹp chi tiết, đồ gá trục phay, ... đều là những thành phần phổ biến trên máy tiện CNC, làm cho máy trở nên rất linh hoạt và đa năng, thường được gọi là *trung tâm tiện CNC*. Các ví dụ lập trình sử dụng thuật ngữ *máy tiện CNC*, nhưng vẫn chấp nhận mọi tính năng hiện đại của trung tâm tiện CNC.

NHÂN LỰC SỬ DỤNG CNC

Máy tính và máy công cụ đều không có trí tuệ. Chúng không thể suy nghĩ, không thể đánh giá tình huống theo lý trí. Chỉ có con người với kiến thức và kỹ năng xác định mới có thể thực hiện điều đó. Trong lĩnh vực điều khiển số, các kỹ năng này thường thuộc về hai nhóm người, nhóm *lập trình* và nhóm *gia công*. Số lượng và nhiệm vụ của họ tùy thuộc vào công ty, loại sản phẩm và quy mô sản xuất. Tuy nhiên, các vị trí này rất khác nhau, dù nhiều công ty kết hợp cả hai nhóm chức năng cho một người, được gọi là *người lập trình/vận hành CNC*.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhà lập trình CNC

Nhà lập trình CNC là nhân vật có trách nhiệm lớn nhất trong xưởng máy CNC. Người này thường chịu trách nhiệm về sự thành công của công nghệ điều khiển số trong nhà máy, phụ trách các vấn đề liên quan đến sự vận hành CNC. Dù nhiệm vụ có thể thay đổi, nhưng nhà lập trình CNC còn chịu trách nhiệm về các tác vụ liên quan đến sự sử dụng có hiệu quả các máy CNC. Đây là nhân vật chịu trách nhiệm về năng suất và chất lượng vận hành CNC.

Nhiều người lập trình CNC là các nhà chế tạo máy giàu kinh nghiệm thực tiễn về vận hành là cơ sở cho khả năng "gia công" chi tiết trong môi trường văn phòng. Nhà lập trình CNC phải có khả năng hình dung mọi chuyển động của dụng cụ cắt và nhận biết mọi yếu tố giới hạn có thể có. Nhà lập trình phải có khả năng thu thập, phân tích, xử lý và tích hợp một cách logic mọi dữ liệu đã thu thập thành chương trình gia công. Nói một cách đơn giản, nhà lập trình CNC phải có khả năng quyết định phương pháp sản xuất tối ưu trong các điều kiện cụ thể.

Ngoài kỹ năng gia công và công nghệ, nhà lập trình CNC còn phải hiểu các nguyên lý toán học, đặc biệt là áp dụng các phương trình, giải các góc và cung tròn, các hàm lượng giác. Ngay cả với lập trình được máy tính hóa, các phương pháp lập trình thủ công là cơ sở để hiểu rõ phần xuất từ máy tính và điều khiển phần xuất đó.

Phẩm chất quan trọng của nhà lập trình CNC chuyên nghiệp là khả năng lắng nghe đồng nghiệp, các kỹ sư, người vận hành CNC, nhà quản lý. Khả năng đó là điều kiện tiên quyết để bạn trở nên linh hoạt hơn, từ đó có thể nâng cao chất lượng lập trình CNC của mình.

NGƯỜI VẬN HÀNH MÁY CNC

Người vận hành máy công cụ CNC giữ cương vị bổ sung cho nhà lập trình CNC. Nhà lập trình và người vận hành có thể là một người, điều này thường được áp dụng trong các xưởng cơ khí nhỏ. Tuy hầu hết các nhiệm vụ của thợ vận hành máy công cụ cổ điển đã được chuyển giao cho nhà lập trình, nhưng người vận hành CNC vẫn có các trách nhiệm đặc thù. Trong các trường hợp phổ biến, người vận hành chịu trách nhiệm về dụng cụ cắt và xác lập máy, thay đổi các chi tiết, thường kèm theo sự giám sát trong khi gia công. Nhiều công ty tiến hành kiểm tra chất lượng ngay tại máy, người vận hành máy công cụ, dù bằng tay hay máy tính hóa, đều phải chịu trách nhiệm về chất lượng công việc được thực hiện trên máy đó. Một trong những trách nhiệm rất quan trọng của người vận hành máy CNC là báo cáo các phát hiện về từng chương trình cho nhà lập trình. Ngay cả với kiến thức, kỹ năng, sự tập trung và kinh nghiệm, chương trình đã viết, vẫn có thể được cải tiến. Người vận hành CNC, là người trực tiếp gia công trên máy, có thể nhận biết những điều cần cải tiến trong các điều kiện cụ thể.

AN TOÀN VỚI MÁY CNC

Trên tường của nhiều công ty thường có khẩu hiệu an toàn với một câu đơn giản:

Quy tắc an toàn thứ nhất là tuân theo mọi quy tắc an toàn

An toàn ở đây không định hướng theo lập trình hay gia công. Lý do là an toàn hoàn toàn *độc lập*. An toàn có vị trí đặc biệt là hướng dẫn hành vi của từng người bên trong và cả bên ngoài xưởng cơ khí. Thoạt nhìn, an toàn liên quan chặt chẽ với sự gia công, vận hành máy, và có lẽ cả sự xác lập máy. Điều đó là đúng nhưng rõ ràng chưa đủ để bảo đảm an toàn.

An toàn là yếu tố quan trọng nhất trong lập trình, xác lập máy, gia công, dụng cụ cắt, gá lắp chi tiết, giám sát, vận chuyển, và thao tác của chính bạn trong công việc hàng ngày ở xưởng cơ khí. Sự chú trọng về an toàn không bao giờ thừa.

Thoạt nhìn, dường như trong lĩnh vực CNC, an toàn chỉ là vấn đề thứ yếu. Ở đây, hầu hết đều tự động. Chương trình gia công chạy lặp lại nhiều lần, sử dụng cùng loại dụng cụ cắt, sự xác lập máy đơn giản, ... Toàn bộ điều này có thể dẫn đến sự chủ quan, thậm chí coi thường các quy định về an toàn. Quan điểm đó có thể có các hậu quả nghiêm trọng.

An toàn là chủ đề lớn, có vài điểm rất quan trọng liên quan đến hoạt động CNC. Từng nhà chế tạo máy đều cần biết các nguy hiểm của thiết bị điện và cơ khí. Bước thứ nhất hướng đến làm việc an toàn là khu vực làm việc phải sạch sẽ, phoi vụn, vệt dầu mỡ, và các loại rác không được để đọng lại trên sàn xưởng. Sự chú ý đến an toàn cá nhân là rất quan trọng. Y phục quá rộng, đồ trang sức, khăn quàng, tóc dài không được bảo vệ, sử dụng găng tay không chuẩn, và các tiểu tiết khác, đều nguy hiểm trong môi trường gia công cơ khí. Sự bảo vệ mắt, tai, tay và chân là rất cần thiết.

Trong khi máy đang hoạt động, các thiết bị

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

bảo hộ phải ở đúng vị trí, mọi bộ phận chuyển động đều được che chắn. Cần đặc biệt chú ý xung quanh các trục quay và các bộ đổi dao cắt tự động. Các bộ phận khác có thể gây nguy hiểm bao gồm hệ thống cấp phôi và lấy sản phẩm, bộ chuyển tải phôi cắt, khu vực điện áp cao, tời nâng, ... sự ngắt nối kết các khóa liên động, hoặc các tính năng an toàn khác luôn luôn nguy hiểm, và bất hợp pháp, nếu không được phép của người có trách nhiệm.

Trong lập trình, sự quan sát các quy định an toàn cũng rất quan trọng. Sự chuyển động của dụng cụ cắt có thể được lập trình theo nhiều cách. Tốc độ và lượng ăn dao có tính thực tiễn, không chỉ "chính xác" theo toán học. Chiều sâu cắt, các đặc tính dao cắt, tất cả đều có ảnh hưởng đối với tính an toàn chung.

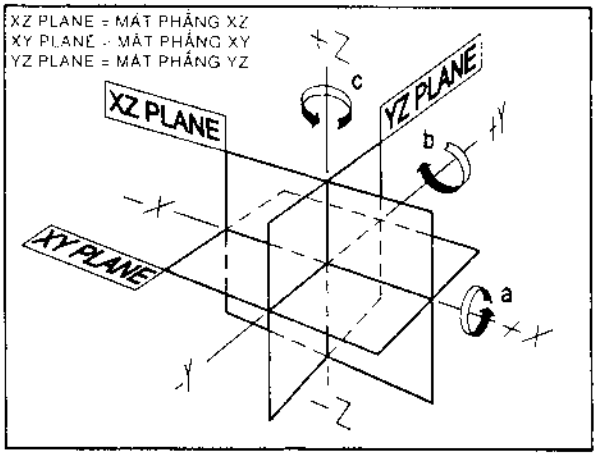
HÌNH HỌC MÁY

Hệ tọa độ tay phải

Trên *trục số*, *góc phần tư* và các *trục*, điểm *gốc tọa độ* chia từng trục thành hai phần. Điểm zero – *gốc tọa độ* – phân chia giữa phần dương và phần âm của trục tọa độ. Trong hệ tọa độ tay phải, trục *dương* bắt đầu từ điểm gốc và hướng về *bên phải* đối với trục X, *hướng lên* đối với trục Y, và *hướng đến điểm chiếu vuông góc* đối với trục Z. Các chiều ngược lại là âm.

Nếu đặt hệ tọa độ này lên bàn tay phải, chúng sẽ tương ứng chiều từ gốc đến đầu ngón cái hoặc ngón trỏ và ngón giữa. Ngón cái hướng theo chiều X, ngón trỏ chiều Y và ngón giữa là chiều Z.

Hầu hết các máy CNC đều được lập trình với phương pháp tọa độ *tuyệt đối*, dựa trên gốc tọa độ X0Y0Z0. Phương pháp tọa độ tuyệt đối



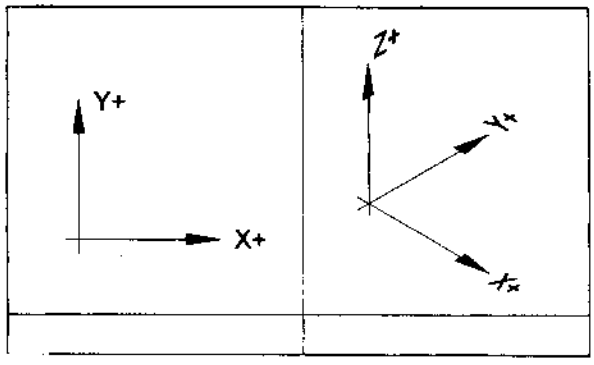
Hình 1.1. Hướng tiêu chuẩn của các mặt phẳng và các trục máy CNC

tuân thủ chặt chẽ các nguyên tắc của hệ tọa độ vuông góc (Hệ tọa độ Descartes).

Hình học máy là quan hệ của các khoảng cách giữa *điểm cố định trên máy* và *điểm lựa chọn trên chi tiết*. Hình học máy CNC thường sử dụng hệ tọa độ tay phải chiều trục dương và âm được xác định bằng quy ước chiều. Nguyên tắc cơ bản đối với trục Z là đây luôn luôn là trục, mà theo trục đó có thể gia công lỗ đơn giản với dao một lưỡi cắt, chẳng hạn khoan, chuốt, EDM hoặc chùm laser. Hình 1.1 minh họa sự định hướng điều khiển của máy công cụ kiểu XYZ.

Định hướng trục – phay

Máy 3 trục cơ bản sử dụng ba trục điều khiển chuyển động, đó là *trục X*, *trục Y* và *trục Z*. Trục X song song với chiều (kích thước) dài nhất của bàn máy, trục Y song song với chiều (kích thước) ngắn nhất của bàn máy và trục Z là chuyển động của trục chính. Trên trung tâm gia công đứng, trục X là chiều *dọc* bàn máy, trục Y là chiều *ngang* và trục Z là chiều *trục chính*. Đối với trung tâm gia công ngang, thuật ngữ hơi thay đổi so thiết kế của các máy này. Trục X là chiều dọc bàn máy, trục Y là chiều cột, và trục Z là chiều trục chính. Máy ngang có thể được coi là máy đứng khi quay 90^0 trong không gian. Tính năng bổ sung của trung tâm gia công ngang là trục phân độ B. Các trục máy CNC đứng được minh họa trên Hình 1.2.



Hình 1.2. Các trục máy của trung tâm gia công CNC đứng.

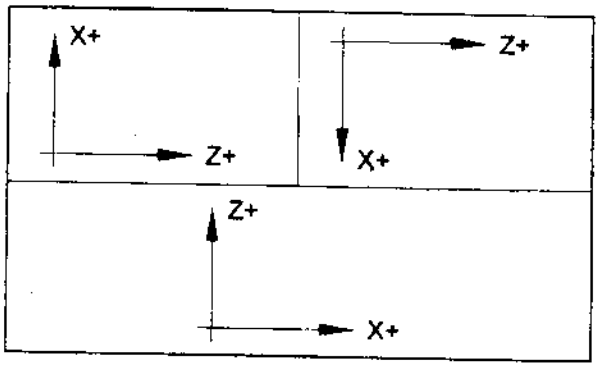
Định hướng trục – tiện

Hầu hết các máy tiện CNC đều có hai trục X và Y và có thể có thêm các trục phục. Trục thứ ba, trục C, được thiết kế cho các nguyên công phay và là tùy chọn trên máy tiện CNC.

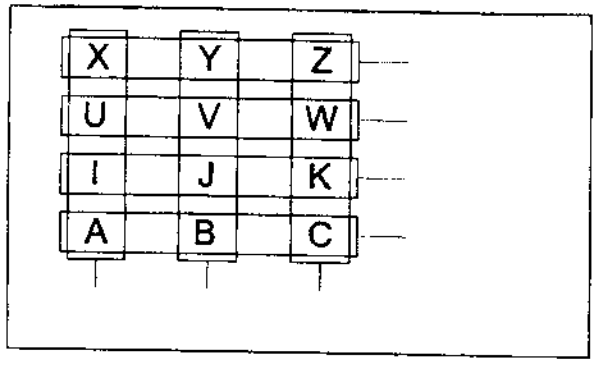
Điều phổ biến đối với các máy tiện CNC trong công nghiệp là sự định hướng kép của các trục XY. Máy tiện được chia thành máy tiện

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 1.3. Các trục máy của máy tiện (trung tâm tiện) CNC



Hình 1.4. Quan hệ giữa các trục máy chính và phụ

trước và sau. Máy tiện trước tương tự máy tiện cổ điển. Mọi kiểu máy tiện có băng máy nghiêng đều là máy tiện sau. Ký hiệu các trục có thể không tuân theo các quy tắc toán học.

Các trục phụ

Kiểu máy CNC bất kỳ đều có thể được thiết kế với một hoặc vài trục bổ sung, thường được coi là các trục phụ với ký hiệu U, V, và W. Các trục này thường song song với các trục chính X, Y, và Z tương ứng. Đối với các ứng dụng quay hoặc phân độ, các trục phụ được ký hiệu là A, B, C khi quay quanh trục X, Y và Z tương ứng. Chiều dương của trục quay (hoặc phân độ) là chiều tiến của ren phải theo chiều dương của trục X, Y hoặc Z. Quan hệ giữa các trục chính và trục phụ được nêu trên Hình 1.4.

Các vector tâm cung (tròn) không phải là các trục thực sự, nhưng có quan hệ với các trục chính XYZ. Điều này sẽ được trình bày chi tiết trong Chương 28.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

2 PHAY CNC

Nhiều kiểu máy CNC được sử dụng trong công nghiệp, đa số là *trung tâm gia công CNC* và *máy tiện CNC*. Tiếp sau chúng là các máy gia công tia lửa điện EDM, máy tạo hình và các máy có thiết kế đặc biệt. Tuy trọng tâm của cuốn sách này là hai kiểu máy chiếm phần lớn thị trường, nhưng nhiều ý tưởng chung cũng được áp dụng cho các trang thiết bị CNC khác.

MÁY CNC - PHAY

Nội dung về máy phay CNC rất nhiều, có thể trình bày trong hàng chục cuốn sách. Mọi máy công cụ từ máy phay đơn giản đến máy chép hình 5 trục đều thuộc loại này. Chúng khác nhau về kích cỡ, tính năng, tính thích hợp đối với các nguyên công xác định,... nhưng đều có một điểm chung — *các trục chính của chúng là trục X và trục Y*, vì lý do đó chúng được gọi là máy XY.

Trong nhóm máy XY còn có máy gia công tia lửa điện (EDM), máy cắt bằng laser và tia nước, máy cắt bằng ngọn lửa, máy hành trình... Tuy chúng không được phân loại theo máy phay, nhưng được nêu ra ở đây do phần lớn các kỹ thuật lập trình áp dụng cho máy phay đều đồng nhất với các kiểu máy đó. Ví dụ rõ nhất là *gia công biên dạng*, quy trình phổ biến đối với nhiều máy CNC.

Máy phay có thể được định nghĩa như sau:

Máy phay là máy công cụ có khả năng chuyển động cắt đồng thời, sử dụng đầu phay làm dụng cụ cắt chính, dọc theo ít nhất hai trục trong cùng thời điểm.

Định nghĩa này không bao gồm máy ép khoan, do thiết kế của chúng chỉ xét đến vị trí mà không tính đến biên dạng. Định nghĩa cũng không gồm máy EDM và các loại máy cắt khác, do chúng có khả năng gia công biên dạng nhưng không sử dụng đầu phay. Người dùng những loại máy đó vẫn có nhiều điều bổ ích từ các nội dung được trình bày trong sách này. Các nguyên tắc chung ở đây đều có thể áp dụng cho hầu hết các máy công cụ CNC. Ví dụ, máy EMD sử dụng đường kính dây (điện cực) cắt rất nhỏ. Máy cắt laser sử dụng chùm laser làm dụng cụ cắt, cũng có đường kính cho trước, quyết định chiều rộng rãnh cắt. Nội dung sẽ tập trung vào các máy công cụ cắt gọi kim loại sử dụng nhiều kiểu đầu phay khác nhau làm dụng cụ chính để gia công biên dạng. Do đầu phay có thể sử dụng theo nhiều cách, trước hết bạn hãy khảo sát các kiểu máy phay thông dụng.

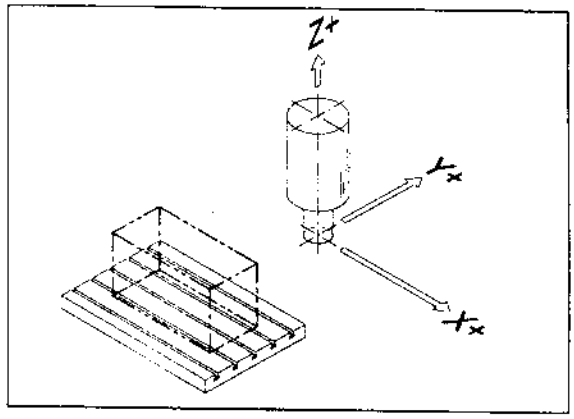
Các kiểu máy phay

Máy phay có thể được chia thành ba nhóm chính

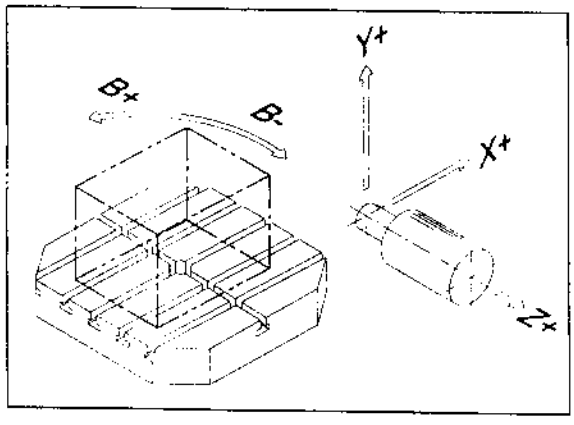
- Theo số trục – hai, ba, hoặc lớn hơn
- Theo hướng trục – đứng hoặc ngang
- Có hoặc không có bộ thay dao

Máy phay với chuyển động trục chính *lên/xuống*, được gọi là máy phay *đứng*. Máy phay có chuyển động trục chính *vào/ra*, là máy phay *ngang* (Hình 2.1 và 2.2)

Các định nghĩa đơn giản này chưa phản ánh đúng thực tế đang diễn ra trong chế tạo máy công cụ hiện nay. Công nghiệp máy công cụ liên tục phát triển. Các máy mới và mạnh



Hình 2.1. Sơ đồ trung tâm gia công CNC kiểu



Hình 2.2. Sơ đồ trung tâm gia công CNC kiểu ngang

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

hơn được thiết kế và chế tạo trên toàn thế giới ngày càng nhiều tính năng hơn.

Hầu hết các máy hiện đại được thiết kế để gia công phay đều có khả năng thực hiện nhiều nguyên công, không chỉ có phương pháp phay truyền thống. Các máy này còn có nhiều tính năng tích hợp, chẳng hạn khoan, chuốt lỗ, cắt ren, gia công biên dạng, ... chúng có thể được trang bị với hộp chứa nhiều dao, bộ thay dao hoàn toàn tự động (ATC), bộ thay chi tiết (APC), và bộ điều khiển máy tính hóa (CNC), ... Một số kiểu máy còn có các tính năng bổ sung, chẳng hạn giao diện robot, cấp và lấy phôi tự động, hệ thống đo và cảm biến, các tính năng gia công tốc độ cao và nhiều đặc tính khác của công nghệ hiện đại. Câu hỏi là các máy công cụ tích hợp nhiều tính năng tiên tiến có thể được coi là máy phay CNC đơn giản? Chắc chắn là không. Các máy phay có ít nhất là vài tính năng tiên tiến, trở thành nhóm máy công cụ mới – *Trung tâm gia công CNC*. Thuật ngữ này chỉ liên quan đến CNC, hoàn toàn không có *trung tâm gia công bằng tay*.

Trục máy

Máy phay và trung tâm gia công có ít nhất ba trục X, Y, và Z. Máy sẽ trở nên linh hoạt hơn nếu có thêm trục thứ tư, thường là trục quay hoặc phân độ (ký hiệu là trục A đối với máy đứng và trục B đối với máy ngang). Có thể đạt được tính linh hoạt cao hơn với máy năm trục hoặc nhiều trục hơn. Máy đơn giản với năm trục có thể doa lỗ, gồm ba trục chính, một trục quay (thường là trục B) và trục song song với trục Z (ký hiệu là trục W). Tuy nhiên máy phay biên dạng 5 trục linh hoạt và phức tạp là kiểu được dùng trong công nghiệp máy bay, cần có sự chuyển động cắt đồng thời nhiều trục để gia công các hình dạng phức tạp, các mặt cong lồi và lõm.

Trước đây các thuật ngữ máy *hai và bán trục* hoặc máy *ba và bán trục* được dùng để biểu thị kiểu máy với chuyển động cắt đồng thời tất cả các trục còn nhiều hạn chế. Ví dụ máy đứng 4 trục, có ba trục chính là X, Y, Z và bàn phân độ, được gọi là trục A. Bàn phân độ dùng để định vị, nhưng không thể quay đồng thời với chuyển động của các trục chính. Kiểu máy này được gọi là máy *ba và bán trục*. Ngược lại, máy tương tự nhưng phức tạp hơn được trang bị bàn máy quay toàn phần, được thiết kế là máy bốn trục. Bàn quay có thể chuyển động đồng thời với chuyển động cắt của các trục chính. Đây là ví dụ về máy công cụ "*bốn trục*" thực sự.

Mỗi trung tâm gia công đều có các đặc tính kỹ thuật do nhà chế tạo máy công cụ cung cấp. Nhà chế tạo liệt kê nhiều đặc tính kỹ thuật, cho phép dễ dàng so sánh máy này với máy khác.

Trong hệ thống phay, có ba kiểu máy công cụ phổ biến:

- Trung tâm gia công CNC kiểu đứng–VMC
- Trung tâm gia công CNC kiểu ngang– HMC
- Máy phay doa CNC kiểu ngang

Các phương pháp lập trình hầu như không khác nhau giữa các kiểu máy, trừ các tùy chọn bổ sung và phụ tùng đặc biệt. Các khác biệt chính bao gồm hướng của các trục máy, trục phụ để chia độ hoặc quay, kiểu nguyên công thích hợp trên từng loại máy, kiểu trung tâm gia công thông dụng nhất – *Trung tâm gia công đứng (VMC)* có thể đại diện cho các máy khác trong nhóm này.

Trung tâm gia công đứng

Trung tâm gia công đứng được sử dụng chủ yếu cho kiểu gia công phẳng, chẳng hạn các tấm, hầu hết quá trình gia công được thực hiện chỉ trên một mặt của chi tiết trong một xác lập máy.

Trung tâm gia công CNC kiểu đứng (VMC) còn có thể được sử dụng với trục thứ tư, thường là đầu quay lắp ngang hoặc đứng, tùy theo yêu cầu và kiểu máy. Trục thứ tư này có thể được dùng để phân độ hoặc quay toàn phần. Nếu kết hợp với ụ động, trục thứ tư trong cấu hình đứng có thể được dùng để gia công các chi tiết, dài cần đỡ ở cả hai đầu.

Hầu hết các thợ vận hành trung tâm gia công đứng đều làm việc với cấu hình ba trục và bàn máy không có trục thứ tư.

Dưới góc độ lập trình, bạn cần chú ý ít nhất hai vấn đề:

- **MỘT** – Lập trình luôn luôn xác định vị trí *từ điểm quy chiếu theo trục chính*, thay vì theo chiều thợ vận hành. Điều này có nghĩa là điểm chiếu là chiếu thẳng xuống, theo góc 90^0 hướng đến bàn máy để triển khai chuyển động của dụng cụ cắt. Nhà lập trình luôn luôn nhìn mặt trên của chi tiết gia công.
- **HAI** – Các dấu mốc định vị trên máy biểu thị chuyển động theo chiều dương và âm của các trục máy. Đối với lập trình, cần bỏ qua dấu mốc này, chúng chỉ biểu thị chiều gia công không biểu thị chiều lập trình. Các chiều lập trình hoàn toàn ngược với các dấu mốc trên máy công cụ.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trung tâm gia công ngang

Trung tâm gia công ngang còn được phân loại là các máy vạn năng nhiều dụng cụ cắt, được dùng cho các chi tiết hình khối, hầu hết các quy trình gia công được thực hiện trên nhiều bề mặt trong một xác lập máy.

Nói chung, có nhiều ứng dụng trong lĩnh vực này, thường là các chi tiết lớn, chẳng hạn vỏ hộp bơm, vỏ hộp số, block động cơ đốt trong... Trung tâm gia công ngang luôn luôn có bàn phân độ đặc biệt, thường có bộ thay dao và các tính năng khác.

Do tính linh hoạt và phức tạp của chúng, các trung tâm gia công ngang có giá cao hơn nhiều so với trung tâm gia công đứng.

Dưới góc độ lập trình, có nhiều khác biệt đặc thù, chủ yếu liên quan tới bộ thay dao tự động, bàn phân độ, và trong một số trường hợp, các phụ tùng bổ sung tính năng. Tất cả các khác biệt này đều tương đối nhỏ. Viết chương trình cho trung tâm gia công ngang không khác với viết chương trình cho trung tâm gia công đứng.

Máy phay doa ngang

Phay doa ngang cũng là máy CNC. Máy này khá giống trung tâm gia công ngang CNC, nhưng có các khác biệt riêng. Nói chung, máy phay doa ngang thiếu một số tính năng chung, chẳng hạn bộ thay dao tự động. Công dụng chính của máy là thực hiện các nguyên công

Bảng 2.1 Trung tâm gia công đứng và ngang – các đặc tính kỹ thuật

Đặc tính	Trung tâm gia công đứng	Trung tâm gia công ngang
Số trục	3 trục (XYZ)	4 trục (XYZB)
Kích thước bàn máy	780 x 400 mm 31 x 16 inch	500 x 500 mm 20 x 20 inch
Số dụng cụ cắt	20	36
Hành trình cực đại trục X	575 mm 22.5 inch	725 mm 28.5 inch
Hành trình cực đại trục Y	380 mm 15 inch	560 mm 22 inch
Hành trình cực đại trục Z	470mm 18.5 inch	560 mm 22 inch
Góc phân độ bàn máy	N/A	0.001 độ
Tốc độ trục chính	60-8000 v/ph	40 – 4000 v/ph
Công suất trục chính	AC 7.5/5.5 kw AC 10/7 HP	AC 11/8 kW AC 15/11 HP
Khoảng cách mũi trục chính – bàn máy – trục Z	150-625 mm 6-24.6 inch	150 – 710 mm 6 - 28 inch
Khoảng cách mũi trục chính – bàn máy – trục Y	430 mm 17 inch	30 – 560 mm 1.2 – 22 inch
Độ côn trục chính	No. 40	No. 50
Kích cỡ cán dao	BT 40	CAT50
Khoảng tốc độ cắt	2-1000 mm/phút 0.100 – 393 in/ phút	1 – 10000 mm/ph 0.04 – 393 in/ph
Tốc độ chạy ngang nhanh	30000 mm/ph (XY)–24000 mm/ph (Z) 1181 in/ph (XY) – 945 in/ph (Z)	30000 mm/ph (XY) – 24000 mm/ph (Z) 1181 in/ph (XY) – 945 in/ph (Z)
Chọn dụng cụ cắt	Bộ nhớ ngẫu nhiên (RAM)	Bộ nhớ ngẫu nhiên (RAM)
Đường kính dao cắt max	80 mm (rộng 150/ ổ dao rỗng) 3.15 in (rộng 5.8/ ổ dao rỗng)	105 mm 4.1 inch
Chiều dài dao cắt max	300 mm 11.8 inch	350 mm 13.75 inch
Trọng lượng dao cắt max	6 kg 13 lbs	20 kg 44 lbs

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

doa, đặc biệt là doa lỗ sâu. Vì lý do đó, đoạn nối đến trục chính được thực hiện bằng ống lồng với thiết bị đặc biệt. Tính năng thứ hai là trục song song với trục Z, được gọi là trục W. Dù có trục thứ năm (X, Y, Z, B, W) nhưng máy phay doa ngang vẫn chưa phải là máy năm trục thực sự. Trục Z (ống lồng) và trục W (bàn máy) làm việc theo hai chiều ngược nhau, để chúng có thể được dùng cho các chi tiết lớn và các *không gian khó với tới*. Điều đó còn có nghĩa là trong khi khoan, bàn máy tiến về phía ống lồng. Ống này là một phần lắp với trục chính. Ống lồng trong trục chính ở nơi dụng cụ cắt quay, còn chuyển động vào/ra do bàn máy thực hiện. Bạn hãy suy nghĩ phương pháp thay thế trên máy phay doa ngang, nếu ống lồng quá dài sẽ giảm độ bền và tính cứng vững. Giải pháp là chia chuyển động trục X thành hai phần – ống lồng dọc theo trục Z chỉ chuyển động phần hành trình hướng đến bàn máy và chỉnh bàn máy, trục W mới, sẽ chuyển động phần hành trình hướng đến trục chính. Cả hai sẽ gặp nhau ở phần của chi tiết có thể được gia công sử dụng toàn bộ nguồn dao cắt.

Máy phay – doa ngang có thể được gọi là máy CNC 3½ trục, nhưng chắc chắn không phải là máy CNC 5 trục, dù có thể đếm số trục là năm. Các bước lập trình trên máy phay doa ngang tương tự trung tâm gia công đứng và ngang.

Các đặc tính kỹ thuật thông dụng

Bảng 2.1 liệt kê các đặc tính kỹ thuật phổ biến của *Trung tâm gia công đứng CNC và trung tâm gia công ngang CNC*. Các đặc tính kỹ thuật này được xếp trong hai cột chỉ để thuận tiện, không nhằm mục đích so sánh. Đây là hai kiểu máy khác nhau với các đặc tính riêng, không thể và không nên so sánh. Để so sánh các máy công cụ cùng loại, nhà chế tạo cung cấp các bảng đặc tính kỹ thuật là cơ sở để so sánh. Các bảng này liệt kê dữ liệu đã kiểm định, chủ yếu là dữ liệu kỹ thuật, biểu thị các tính năng chính của từng máy. Sự so sánh chỉ nên thực hiện giữa hai máy cùng loại chẳng hạn giữa hai trung tâm gia công đứng hoặc hai trung tâm ngang, và sự lựa chọn dựa trên nhu cầu gia công thực tế của xưởng cơ khí. Ở đây sẽ tập trung chú ý vào các đặc tính kỹ thuật cần thiết đối với nhà lập trình và thợ vận hành CNC.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

3 TIỆN CNC

MÁY CNC - TIỆN

Máy tiện là máy công cụ phổ biến trong mọi xưởng cơ khí. Máy tiện được dùng để gia công mặt trục hoặc côn, chẳng hạn trục, vòng chặn, bánh xe, lỗ, ren,... Nguyên công tiện thông dụng nhất là cắt bỏ vật liệu từ phôi trục, sử dụng dao tiện để cắt mặt ngoài. Máy tiện còn được dùng để gia công lỗ, cắt rãnh, cắt ren,... nếu dùng dụng cụ cắt thích hợp. Máy tiện quay (revolve) thường có khả năng gia công yếu hơn máy tiện thông dụng, mỗi lần chỉ lắp một hoặc hai dao cắt, nhưng năng lực gia công lớn hơn.

Gia công tiện phổ biến được điều khiển bằng hệ thống CNC sử dụng các máy trong công nghiệp được gọi là *trung tâm tiện CNC* hoặc đơn giản là máy tiện CNC.

Thuật ngữ *"trung tâm tiện CNC"* ít được sử dụng nhưng có ý nghĩa chính xác hơn, được hiểu là máy tiện máy tính hóa (máy tiện CNC) có thể được dùng cho nhiều nguyên công chỉ với một xác lập máy. Ví dụ, ngoài các nguyên công tiện tiêu chuẩn, như tiện mặt ngoài, tiện lỗ, máy tiện CNC còn có thể được dùng để khoan, cắt rãnh, cắt ren, chà gai nhám, đánh bóng. Máy có thể được dùng trong các chế độ khác nhau, chẳng hạn gia công với mâm cặp, ống kẹp, cấp phôi thanh, hoặc giữa các tâm, ngoài ra còn có nhiều tổ hợp khác. Máy tiện CNC được thiết kế để lắp nhiều dụng cụ cắt trong giá xoay, chúng có thể có đồ gá phay, mâm cặp phân độ, trục phụ, ụ động, gối tựa và nhiều tính năng khác, có thể không có trong máy tiện truyền thống. Máy tiện với hơn bốn trục bắt đầu trở nên phổ biến. Với các tiến bộ liên tục trong công nghệ máy công cụ, ngày càng nhiều máy tiện CNC trên thị trường được thiết kế để thực hiện nhiều nguyên công trong một chế độ xác lập máy, đa số các nguyên công đó trước đây chỉ thực hiện trên máy phay hoặc trung tâm gia công.

Các kiểu máy tiện CNC

Về cơ bản, máy tiện CNC có thể được phân loại theo *kiểu thiết kế* và theo *số lượng trục*. Hai kiểu cơ bản là máy tiện CNC *đứng* và máy tiện CNC *ngang*. Trong đó, kiểu ngang thông dụng hơn trong sản xuất và trong xưởng cơ khí. Máy tiện CNC đứng (đôi khi bị gọi sai là máy phay doa đứng) ít phổ biến hơn nhưng không thể thiếu đối với chi tiết gia công đường kính lớn. Đối với nhà lập trình CNC, hầu như không có các khác biệt trong phương pháp lập trình giữa hai kiểu máy đó.

Số lượng trục

Sự phân biệt cơ bản giữa các máy tiện là số lượng các trục có thể lập trình. Máy tiện đứng CNC có hai trục trong hầu hết các thiết kế hiện nay. Máy tiện ngang CNC thường được thiết kế với hai trục lập trình, ngoài ra còn có 3 trục, 4 trục, 6 trục, để tăng tính linh hoạt khi gia công các chi tiết phức tạp.

Mấy tiện ngang CNC có thể được phân loại theo kiểu thiết kế kỹ thuật:

- ❑ Máy tiện trước .... kiểu máy tiện thường
- ❑ Máy tiện sau .... kiểu máy tiện với băng máy nghiêng đặc thù.

Kiểu băng máy nghiêng rất thông dụng trong gia công chung, do thiết kế này cho phép phoi tiện rơi cách xa người vận hành CNC và nếu có sự cố, chi tiết gia công sẽ rơi xuống khu vực an toàn, hướng đến băng tải phoi.

Giữa hai loại băng máy phẳng và băng máy nghiêng, máy tiện trước và sau, thiết kế máy tiện ngang và đứng, còn có các kiểu máy tiện khác. Nhóm này phân loại máy tiện CNC theo *số lượng trục*, có lẽ là phương pháp phân loại đơn giản nhất và phổ biến nhất.

KÝ HIỆU TRỤC

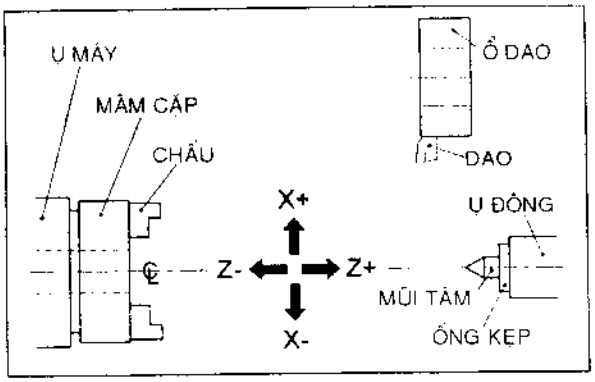
Máy tiện CNC được thiết kế với hai trục tiêu chuẩn là trục X và trục Z. Hai trục này vuông góc với nhau và biểu thị chuyển động tiện hai trục. Trục X còn biểu thị *hành trình ngang* của dụng cụ cắt, trục Z biểu thị *chuyển động dọc*. Tất cả các dụng cụ cắt đều được lắp trên ổ dao, có thể bên ngoài hoặc bên trong. Do thiết kế này, ổ dao có tất cả các dao sẽ dịch chuyển dọc theo các trục X và Y, nghĩa là mọi dụng cụ cắt đều trong vùng làm việc.

Tuân theo các tiêu chuẩn được thiết lập cho máy phay và trung tâm gia công, trục máy ngang là chuyển động *lên/xuống* theo trục X, và chuyển động *trái/phải* theo trục Z khi quan sát từ vị trí người vận hành. Điều này được minh họa trên các Hình 3.1, 3.2 và 3.3.

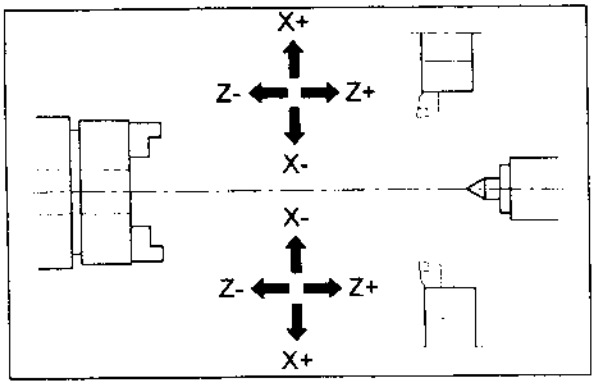
http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

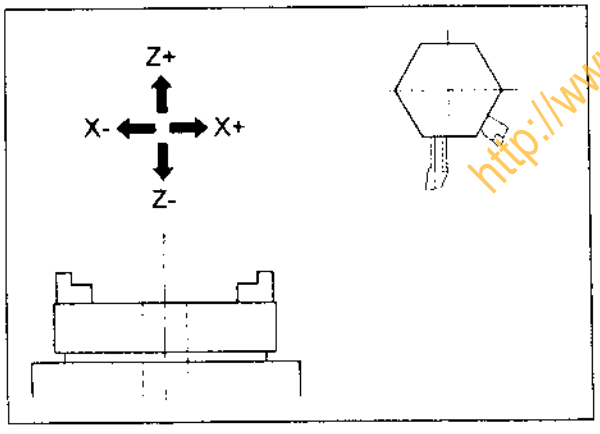
Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 3.1 Cấu hình máy tiện CNC 2 trục băng máy nghiêng – kiểu sau



Hình 3.2 Cấu hình máy tiện CNC với hai ổ dao



Hình 3.3 Sơ đồ máy tiện đứng CNC

Điều này là đúng cho cả hai kiểu máy tiện trước và sau, và cho các máy tiện có hơn hai trục. Mặt mâm cặp được định hướng thẳng đứng (vuông góc) với đường tâm trục chính của máy tiện ngang. Máy tiện đứng, do thiết kế, quay 90^0, mặt mâm cặp định hướng ngang với đường tâm trục chính đứng.

Ngoài các trục X và Y, máy tiện nhiều trục có các mô tả riêng cho từng trục phụ, ví dụ, trục C thường là trục thứ ba, được dùng cho các nguyên công phay, sử dụng các dụng cụ sống. Chương kế tiếp sẽ trình bày chi tiết về hệ tọa độ và hình học máy công cụ.

Máy tiện hai trục

Đây là kiểu máy tiện CNC phổ biến nhất. Bộ phận kẹp chi tiết, thường là mâm cặp, được lắp ở bên trái máy (phía trái của người vận hành). Kiểu sau, với bàn máy nghiêng, là thiết kế thông dụng nhất. Đối với một số dạng gia công đặc biệt, chẳng hạn trong công nghệ dầu khí (tiện các đầu ống), băng máy phẳng có thể thích hợp hơn. Dụng cụ cắt (dao tiện) được lắp trong ổ dao phân độ có thiết kế đặc biệt, có thể lắp 4, 6, 8, 10, 20,... dao cắt. Một số máy tiện loại này có hai ổ dao.

Máy tiện ba trục

Máy tiện ba trục về cơ bản là máy hai trục có thêm một trục phụ. Trục này thường được ký hiệu là C trong chế độ tuyệt đối (trục H trong chế độ số gia) và có thể được lập trình toàn phần. Nói chung, trục thứ ba được dùng cho các nguyên công phay – ngang, cắt rãnh, khoan lỗ tròn, các mặt lục giác, các mặt bên, rãnh xoắn... Trục này có thể thay cho một số nguyên công đơn giản trên máy phay, giảm thời gian xác lập máy. Nhiều kiểu máy có một số giới hạn, ví dụ nguyên công phay hoặc khoan chỉ có thể thực hiện ở các vị trí chìa ra từ đường tâm dao cắt đến đường tâm trục chính (trong phạm vi mặt phẳng gia công), một số máy khác có các điều chỉnh lệch tâm.

Trục thứ ba có động cơ điện riêng nhưng định mức công suất tương đối thấp so với đa số các trung tâm gia công. Hạn chế kế tiếp có thể là số gia nhỏ nhất của trục thứ ba, đặc biệt là các máy ba trục kiểu cũ. Số gia nhỏ nhất khoảng 1^0 chắc chắn là hữu dụng hơn so với số gia 2^0 hoặc 3^0. Các máy đời mới có số gia 0.1^0, 0.01^0 và 0.001^0. Nói chung, máy tiện ba trục có số gia góc nhỏ, cho phép chuyển động quay đồng thời. Các máy có giá trị số gia nhỏ thường được thiết kế chỉ có cữ chặn trục chính định hướng.

Dưới góc độ lập trình CNC, không đòi hỏi nhiều kiến thức bổ sung khi lập trình gia công trên máy tiện. Các nguyên lý chung về máy phay và nhiều tính năng lập trình cũng được áp dụng cho các máy tiện, kể cả các chu kỳ cố định và các thủ thuật đi tắt.

Máy tiện bốn trục

Theo thiết kế, máy tiện CNC bốn trục là hoàn toàn khác với máy tiện ba trục. Về nguyên tắc, lập trình cho máy tiện bốn trục thực chất là lập trình *đồng thời cho hai máy tiện hai trục*. Điều này có vẻ hơi lạ, nếu bạn chưa hiểu nguyên lý máy tiện bốn trục.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ở đây thực sự có hai điều khiển (và hai tập hợp XZ), mỗi cặp trục XZ có một điều khiển riêng. Chỉ một chương trình có thể được dùng để gia công *đường kính ngoài* (OD) và chương trình thứ hai gia công *mặt trong* (ID – đường kính trong). Do máy tiện bốn trục có thể gia công với từng cặp trục *một cách độc lập*, OD và ID có thể được gia công cùng một lúc, đồng thời thực hiện cả hai nguyên công. Các yếu tố chính để lập trình thành công cho máy tiện bốn trục là sự phối hợp các dụng cụ và các nguyên công, thời chuẩn chuyển động của các dụng cụ cắt và kiến thức cơ bản về công nghệ.

Vì nhiều lý do, cả hai cặp trục không thể làm việc trong toàn bộ thời gian. Do hạn chế đó, cần có các tính năng lập trình đặc biệt, chẳng hạn các mã chờ đồng bộ hóa (thường là *hàm phụ M*), khả năng ước lượng thời gian mỗi dao cắt hoàn tất từng nguyên công, ... Ở đây cần có sự dung hòa, do chỉ *một* tốc độ trục chính có thể được dùng cho cả hai dao cắt hoạt động, và lượng ăn dao là độc lập cho cả hai cặp trục. Điều này có nghĩa là một số nguyên công không thể thực hiện đồng thời.

Không phải mọi nguyên công tiện đều có thể thực hiện trên máy tiện bốn trục. Nhiều trường hợp gia công trên máy tiện bốn trục sẽ có hiệu quả thấp hơn và chi phí cao hơn so với máy tiện hai trục.

Máy tiện sáu trục

Máy tiện CNC sáu trục là máy tiện được thiết kế đặc biệt với hai ổ dao và tập hợp ba trục/ổ dao. Thiết kế này phối hợp nhiều trạm dụng cụ cắt, đa số được truyền động bằng động cơ, và các khả năng gia công ngược. Sự lập trình loại máy tiện này tương tự lập trình đồng thời hai máy tiện ba trục. Hệ thống điều khiển cung cấp sự đồng bộ hóa một cách tự động tùy theo yêu cầu.

Máy tiện CNC sáu trục cỡ nhỏ đến trung bình được dùng nhiều trong các xưởng gia công ren và các chi tiết nhỏ với số lượng lớn.

TÍNH NĂNG VÀ ĐẶC TÍNH KỸ THUẬT

Các nhà chế tạo thường cung cấp cho khách hàng bảng *tính năng* và *đặc tính kỹ thuật* với nhiều dữ liệu quan trọng đã được kiểm tra.

Đặc tính kỹ thuật máy tiện CNC

Máy tiện ngang CNC thông dụng với hai trục và thiết kế băng máy nghiêng, có thể có các đặc tính kỹ thuật như sau.

Bảng 3.1. *Đặc tính kỹ thuật máy tiện CNC*

Đặc tính	Dữ liệu
Số lượng trục	Hai (X, Z) hoặc ba (X, Y, Z)
Độ quay cực đại băng máy	560 mm 22.05 inch
Đường kính tiện max	350 mm 13.76 inch
Chiều dài tiện max	550 mm 21.65 inch
Lỗ trục chính	85 mm 3.34 inch
Dung lượng thanh	71 mm 2.79 inch
Số dao cắt	12
Cỡ dao vuông	25 mm (1 inch)
Cỡ dao tròn	∅ 40mm ∅ 1.57 inch
Thời gian phân độ	0.1 giây
Hành trình trục theo X	222 mm 8.75 inch
Hành trình trục theo Z	635 mm 25 inch
Tốc độ dịch chuyển ngang nhanh trên trục X	16000 mm/phút 629 in/phút
Tốc độ dịch chuyển ngang nhanh trên trục Z	24000 mm/ph 944 in/ph
Lượng ăn dao	0.01 – 500 mm/ph 0.0001–19.68 in/ph
Cỡ mâm cặp	254 mm 10 inch
Động cơ trục chính	AC 15/11 kW AC 20/14.7 HP
Tốc độ trục chính	35-3500 v/ph
Số gia tối thiểu (min)	0.001 mm 0.0001 inch
Đầu có động cơ	
Số dụng cụ cắt quay	12
Tốc độ quay dao cắt	30 – 3600 v/ph
Động cơ phay	AC 3.7/2.2 kW AC 5/2.95 HP
Kích cỡ ống lót	1 – 16 mm 0.04 – 0.63 inch
Cỡ ta rô (cắt ren)	Hệ mét M3 – M16 # 5 – 5/8 inch

Điều quan trọng là hiểu các tính năng và đặc tính kỹ thuật của các máy công cụ CNC trong xưởng cơ khí. Nhiều tính năng liên quan đến hệ thống điều khiển, số khác liên quan đến máy công cụ. Trong lập trình CNC, nhiều quyết định quan trọng dựa trên một hoặc vài tính

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

năng đó, ví dụ số trạm dao cắt khả dụng, tốc độ cực đại của trục chính,...

Các tính năng điều khiển

Bạn cần biết các tính năng điều khiển đặc trưng của máy tiện và các khác biệt của chúng so với điều khiển trên máy phay. Các tính năng điều khiển được trình bày chi tiết trong Chương 5.

Các tính năng điều khiển chính bao gồm:

- Trục X biểu thị đường kính, thay vì bán kính.
- Tốc độ bề mặt không đổi (CSS) là tính năng điều khiển tiêu chuẩn (G96 là lệnh cho CSS và G97 là lệnh cho vòng/phút)
- Chế độ lập trình tuyệt đối là X, Y, hoặc C.
- Chế độ lập trình số gia là U, W, hoặc H.
- Cắt các dạng ren có thể được thực hiện, tùy theo hệ thống điều khiển.
- Thời gian tạm dừng có thể dùng địa chỉ P, U, hoặc X (G04)
- Lựa chọn dao cắt sử dụng ký hiệu 4-chữ số
- Lựa chọn lượng ăn dao (thông thường) theo mm/vòng (mm/rev) hoặc in/vòng (in/rev).
- Lựa chọn lượng ăn dao (đặc biệt) theo m/min hoặc in/min.
- Tốc độ dịch chuyển ngang nhanh khác nhau giữa trục X và trục Z.
- Các chu kỳ lặp lại nhiều lần về tiện, vạt mặt, doa, lặp lại biên dạng, cắt rãnh, và cắt ren.
- Sự vượt quá lượng ăn dao là 0 – 200% với số gia 10% (trên một số máy là 0-150%).
- Trục X có thể đối xứng gương.
- U động có thể được lập trình.
- Vạt góc hoặc bo tròn góc tự động R và I/K trong chế độ G01.
- Lượng ăn dao khi cắt ren cho phép độ chính xác đến sáu chữ số thập phân (theo đơn vị inch)
- Số gia nhỏ nhất theo trục X là 0.001 mm (0.0001 inch) trên đường kính, mỗi phía là một nửa giá trị này.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

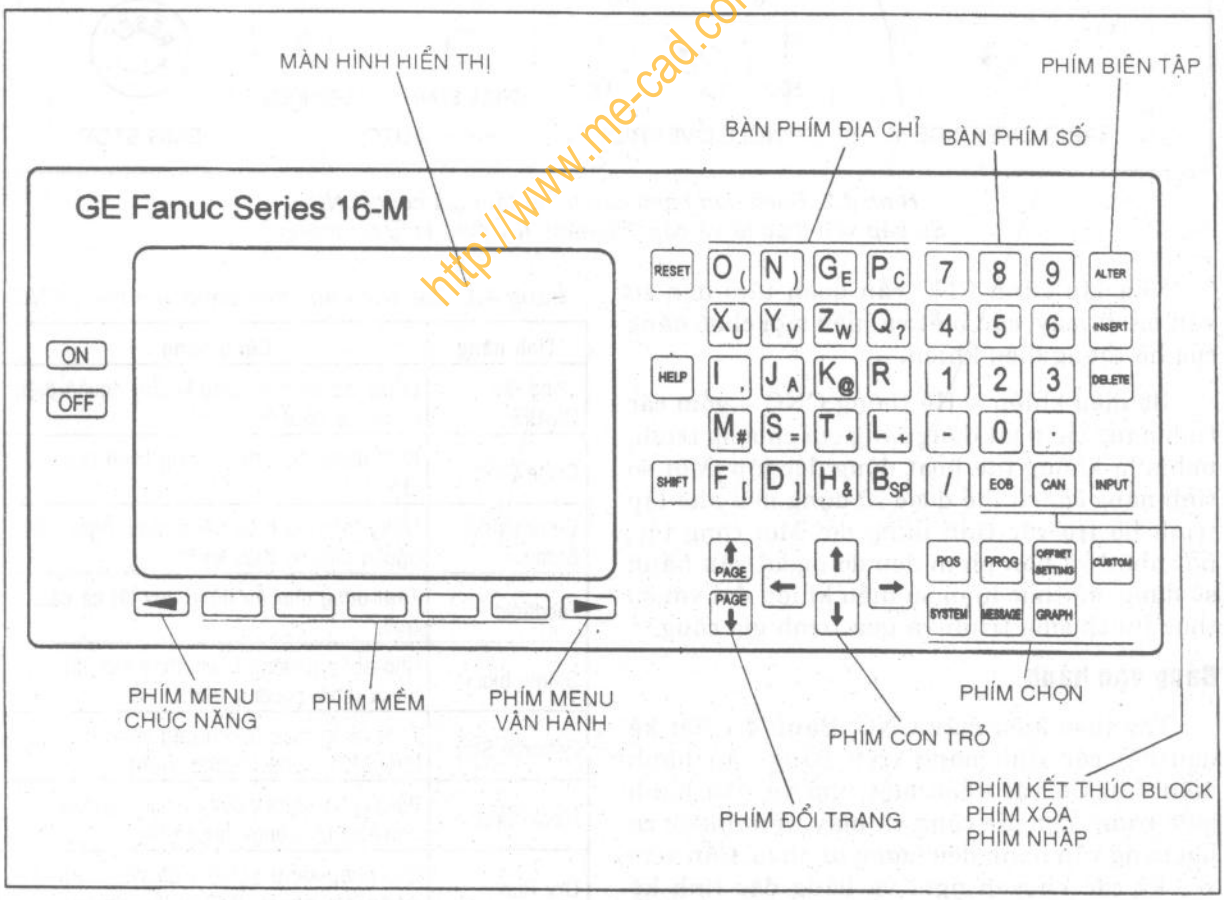
Nhiều tài liệu hay có tại : http://www.me-cad.com/

4 HỆ THỐNG ĐIỀU KHIỂN

Máy công cụ được trang bị hệ thống điều khiển máy tính hóa được gọi là máy CNC. Phần máy công cụ được coi là *thân* (hệ thống thực hiện) của hệ thống máy CNC, phần điều khiển là *bộ não*. Không có cần gạt, nút, và tay quay trên máy CNC, chúng chỉ có trên máy tiện và máy phay cổ điển. Tất cả các chức năng về tốc độ máy, lượng ăn dao, chuyển động trục, và hàng trăm tác vụ khác đều được lập trình do nhà lập trình CNC thực hiện và được máy tính điều khiển, đây là phần chính của máy CNC. Lập trình cho máy công cụ CNC có nghĩa là lập trình cho hệ thống điều khiển. Máy công cụ thực thi mọi tác vụ, nhưng bộ điều khiển xác định các định dạng, cấu trúc và cú pháp của chương trình. Để hiểu đầy đủ quy trình lập trình CNC, điều quan trọng là không chỉ hiểu rõ tính phức tạp của công nghệ chẳng hạn phương pháp gia công chi tiết, lựa chọn dụng cụ cắt, tốc độ và lượng ăn dao, kẹp cắt và định vị, và nhiều tính năng khác, mà còn phải nắm vững về máy tính, bộ CNC, các công việc thực tế mà không nhất thiết phải là chuyên gia về điện tử hoặc khoa học máy tính. Hình 4.1 minh họa bảng điều khiển Fanuc thực tế.

KHÁI QUÁT

Bộ điều khiển có hai nhóm thành phần cơ bản – thứ nhất là *bảng vận hành*, gồm các công tắc xoay, bộ chuyển đổi (hai trạng thái) và các nút nhấn; thứ hai là *màn hình hiển thị* với bàn phím. Nhà lập trình, người thường không làm việc trực tiếp với máy CNC, sẽ ít khi có dịp sử dụng bảng vận hành hoặc màn hình hiển thị. Chúng chỉ khả dụng trên máy, phục vụ cho người vận hành CNC, được dùng để xác lập máy và điều khiển các hoạt động của máy.

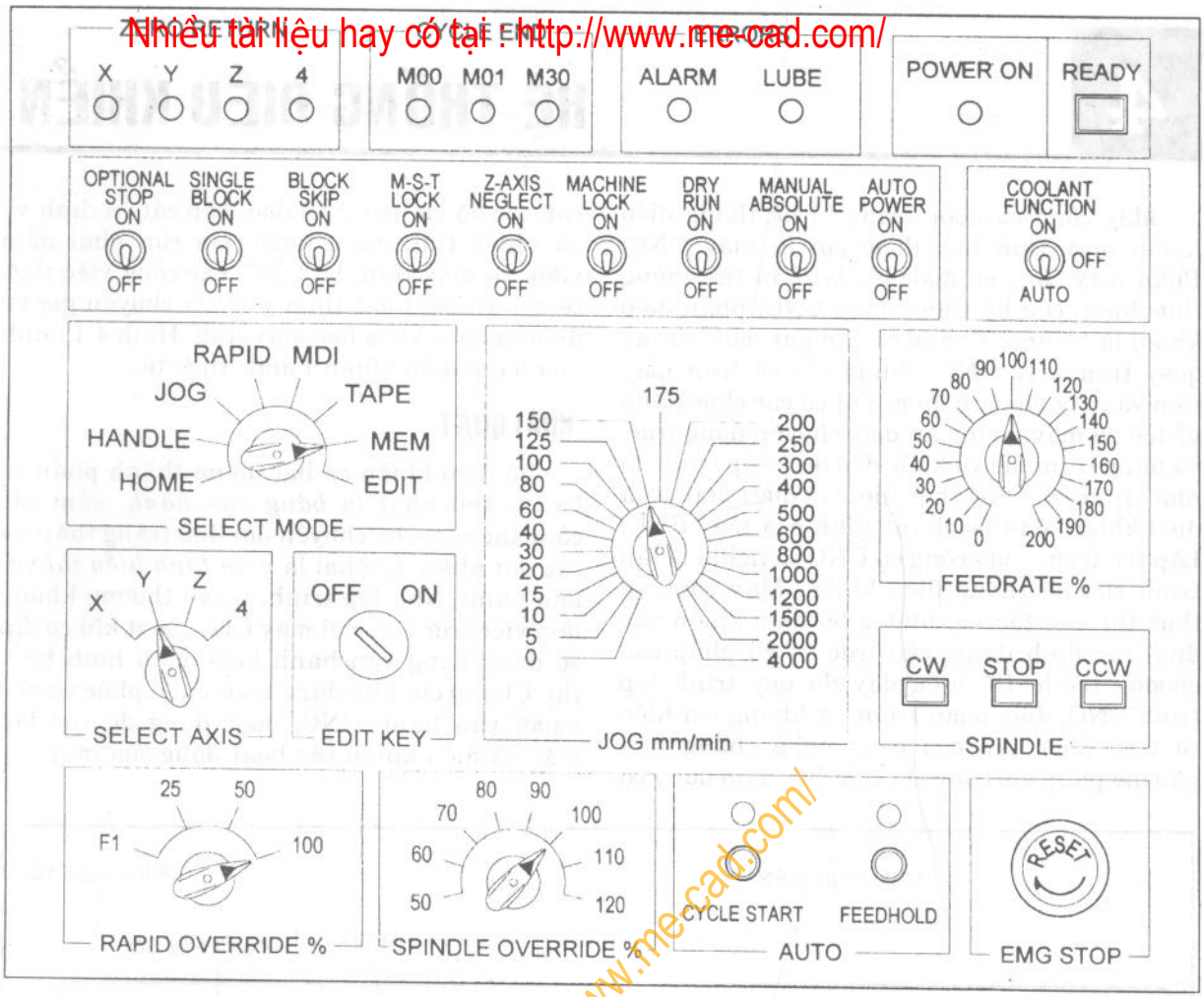


Hình 4.1. Bảng điều khiển Fanuc – sự sắp xếp thực tế và các tính năng có thể hơi khác nhau tùy theo model máy (Fanuc 16M)

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 4.2. Bảng vận hành của trung tâm gia công CNC-
Sự sắp xếp thực tế và các tính năng tùy theo kiểu hệ thống.

http://www.me-cad.com/

Nhà lập trình CNC cần quan tâm đến sự vận hành máy, cần biết và hiểu mọi chức năng của hệ thống điều khiển.

Bộ điều khiển – Hệ thống CNC – gồm các tính năng chỉ hoạt động *cùng với* chương trình, nghĩa là không thể hoạt động độc lập. Một số tính năng *chỉ* có thể được sử dụng nếu nhà lập trình hỗ trợ các tính năng đó. Mọi công tắc, nút nhấn và bàn phím đều do người vận hành sử dụng, để thực hiện sự điều khiển đối với sự thực thi chương trình và quá trình gia công.

Bảng vận hành

Tùy theo kiểu máy CNC, Bảng 4.1 liệt kê hầu hết các tính năng trên bảng vận hành hiện đại. Có vài khác biệt nhỏ về vận hành giữa trung tâm gia công và máy tiện nhưng cả hai bảng vận hành đều tương tự nhau. Cần xem xét kỹ các khuyến nghị và bảng đặc tính kỹ thuật do nhà chế tạo cung cấp, nhiều máy được dùng trong xưởng cơ khí thường có thêm các tính năng đặc biệt.

Bảng 4.1. *Các tính năng trên bảng vận hành CNC*

Tính năng	Công dụng
Công tắc ON/OFF	Công tắc điện và điều khiển nguồn điện chính và bộ điều khiển.
Cycle Start	Khởi động thực thi chương trình hoặc lệnh MDI.
Emergency Stop	Dừng toàn bộ hoạt động máy, ngắt điện nguồn đến bộ điều khiển.
Feedhold	Tạm dừng chuyển động của tất cả các trục.
Single block	Cho phép chương trình chạy mỗi lần một block (khối chương trình).
Optional Stop	Tạm dừng thực thi chương trình (cần có lệnh M01 trong chương trình).
Block Skip	Bỏ qua các block đứng trước với dấu nghiêng (/) trong chương trình.
Dry Run	Cho phép kiểm nghiệm chương trình với sự ăn dao nhanh (không lắp chi tiết).
Spindle Override	Vượt quá tốc độ trục chính được lập trình, trong khoảng 50 – 120%.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tính năng	Công dụng
Feedrate Override	Vượt quá lượng ăn dao được lập trình, trong khoảng 0-200%.
Chuck Clamp	Trạng thái kẹp mâm cặp hiện hành (kẹp ngoài / kẹp trong).
Table Clamp	Trạng thái kẹp chặt trên bàn máy.
Công tắc Coolant	Điều khiển chất làm nguội ON/OFF/AUTO.
Gear Selection	Trạng thái chọn khoảng bánh răng (tốc độ) làm việc hiện hành.
Spindle Rotation	Chiều quay trục chính (thuận hoặc ngược chiều kim đồng hồ).
Spindle Orientation	Định hướng bằng tay cho trục chính.
Tool Change	Công tắc cho phép thay dao bằng tay
Reference Position	Các công tắc và đèn liên quan đến xác lập máy từ chuẩn quy chiếu.
Handle	Bộ tạo xung bằng tay (MPG) dùng cho các công tắc Số gia Chọn và Điều khiển trục.
Công tắc Tailstock	Công tắc ụ động và/hoặc ống kẹp để định vị ụ động bằng tay
Công tắc Indexing Table	Phân độ bàn máy bằng tay trong khi xác lập máy.
MDI Model	Chế độ nhập dữ liệu bằng tay (MDI).
AUTO Model	Cho phép vận hành tự động.
Chế độ MEMORY	Cho phép thực thi chương trình từ bộ nhớ của CNC.
Chế độ TAPE	Cho phép thực thi chương trình từ thiết bị bên ngoài (máy tính, băng đột lỗ).
Chế độ EDIT	Cho phép thực hiện các thay đổi đối với chương trình lưu trong bộ nhớ CNC.
Chế độ MANUAL	Cho phép thao tác bằng tay trong khi xác lập máy.
Chế độ JOG	Chọn chế độ JOG để xác lập máy.
Chế độ RAPID	Chọn chế độ RAPID để xác lập máy.
Memory Access	Phím (công tắc) cho phép biên tập chương trình
Đèn Error	Đèn đỏ báo hiệu có lỗi.

Tuy một số tính năng có thể chưa được liệt kê nhưng hầu như mọi tính năng trong Bảng 4.1 đều có quan hệ với chương trình CNC. Nhiều hệ điều khiển có các tính năng đặc trưng riêng. Người vận hành CNC phải biết các tính năng đó. Chương trình cung cấp cho máy phải *linh hoạt*, không được *cứng nhắc*, và "thân thiện với người dùng".

Màn hình hiển thị và bàn phím

Màn hình hiển thị là "cửa sổ" của máy tính. Chương trình hoạt động bất kỳ đều có thể được hiển thị để kiểm tra trạng thái điều khiển, vị trí dao hiện hành, các điều chỉnh, các tham số, kể cả đồ thị về quỹ đạo dao cắt. Trên mọi thiết bị CNC, có thể chọn màn hình màu hoặc đơn sắc để có hiển thị mong muốn vào thời điểm bất kỳ, sử dụng các phím nhập. Ngoài ra còn có thể lựa chọn ngôn ngữ để hiển thị (tùy loại CNC).

Bàn phím được dùng để *nhập* các lệnh điều khiển. Các chương trình hiện hữu có thể được xóa hoặc chỉnh sửa, có thể bổ sung các chương trình mới. Sử dụng nhập bàn phím, không chỉ có thể điều khiển chuyển động của các trục máy, mà còn cho phép điều khiển tốc độ trục chính và lượng ăn dao. Sự thay đổi các tham số bên trong và đánh giá các chẩn đoán là tương đối phức tạp, chỉ dành cho các nhà chuyên môn. Bàn phím và màn hình được dùng để xác lập chuẩn chương trình và nối với các thiết bị ngoại vi, chẳng hạn với máy tính khác. Ở đây còn có nhiều tùy chọn khác. Bàn phím cho phép sử dụng *ký tự*, *chữ số*, và *ký hiệu* trong mục nhập dữ liệu. Không phải mọi bàn phím đều cho phép sử dụng *tất cả* các chữ cái hoặc *tất cả* các ký hiệu khả dụng. Một số phím trên bảng điều khiển có sự mô tả về *thao tác*, thay vì ký tự, chữ số, hoặc ký hiệu, ví dụ phím *Read*, *Punch* hoặc *Offset*.

Nút xoay

Đối với các mục đích xác lập máy, mọi máy CNC đều có nút xoay để dịch chuyển một trục chọn trước ít nhất bằng số gia nhỏ nhất của hệ thống điều khiển. Tên chính thức do Fanuc đặt cho nút xoay của họ là *Manual Pulse Generator* (MPG, bộ tạo xung bằng tay). Liên quan với nút xoay này là công tắc *Axis Select* (thường ở trên bảng vận hành và trên nút xoay) và khoảng số gia (số gia nhỏ nhất X1, X10 và X100). Chữ X trên nút xoay sẽ dịch chuyển trục được chọn theo X lần số gia tối thiểu theo đơn vị đo đang dùng. Hình 4.3 và Bảng 4.2 minh họa nút xoay Fanuc.

Bảng 4.2. *Số gia dịch chuyển trục*

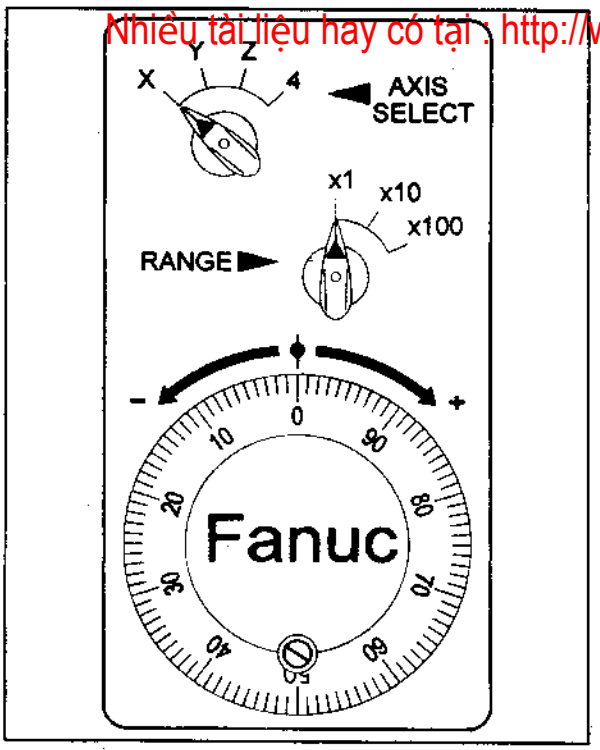
Số nhân	Một đơn vị chuyển dịch	
	Hệ mét	Hệ Anh
X1	0.001 mm	.0001 inch
X10	0.010 mm	.0010 inch
X100	0.100 mm	.0100 inch

TÍNH NĂNG HỆ THỐNG

Bộ CNC thực chất là máy tính với công dụng đặc biệt. "Công dụng đặc biệt" trong trường hợp này là *máy tính có khả năng điều khiển các hoạt động của máy công cụ*, ví dụ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 4.3. *Nút xoay Manual Pulse Generator (MPG)*

máy tiện, trung tâm gia công. Điều đó có nghĩa là máy tính được công ty chuyên ngành máy công cụ thiết kế. Khác với nhiều kiểu máy tính phổ thông, bộ CNC được thiết kế riêng cho từng nhóm khách hàng, người chế tạo máy công cụ, không phải là người sử dụng thực sự. Nhà chế tạo chuyên biệt các yêu cầu đối với hệ thống điều khiển, phản ánh tính đặc thù của các máy công cụ do họ chế tạo. Sự điều khiển cơ bản không thay đổi, nhưng có thể thêm (hoặc giảm bớt) một số tính năng chuyên biệt cho máy. Nhà chế tạo máy sẽ bổ sung thêm các tính năng mới cho hệ thống đó, liên quan chặt chẽ với thiết kế và các năng lực của kiểu máy cụ thể.

Ví dụ, bộ CNC dùng cho hai máy là hoàn toàn như nhau trừ một ngoại lệ. Một máy có bộ thay dao bằng tay, máy kia có bộ thay dao tự động. Để hỗ trợ bộ thay dao tự động, bộ CNC phải có thêm các tính năng đặc biệt, không cần thiết đối với máy CNC không có bộ thay dao. Hệ thống CNC càng phức tạp, giá tiền càng cao. Người dùng có thể không cần có đủ mọi tính năng, do đó chỉ nên chọn các tính năng cần thiết phù hợp nhu cầu gia công và khả năng tài chính của họ.

Các xác lập tham số

Thông tin thiết lập sự nối kết bên trong giữa điều khiển CNC và máy công cụ được lưu dưới dạng dữ liệu đặc biệt trong các bộ ghi nội, được gọi là *tham số hệ thống*. Thông tin được trình bày ở phần này chỉ có tính tham khảo. Nhà lập trình với kinh nghiệm hạn chế có thể không cần biết sâu về tham số hệ thống. Các xác lập tại nơi chế tạo máy đủ cho hầu hết các công việc gia công thực tế.

Khi hiển thị màn hình tham số, bạn có thể thấy các tham số với dữ liệu ghi theo hàng ngang. Mỗi hàng số biểu thị một *byte*, mỗi chữ số trong byte được gọi là *bit*. Thuật ngữ bit là viết tắt của *Binary digIT* (chữ số nhị phân) và là đơn vị nhỏ nhất của mục nhập tham số. Sự đánh số các bit bắt đầu với chữ số 0 và đọc từ *phải* sang *trái*.

Số	#7	#6	#5	#4	#3	#2	#1	#0
XXX	0	1	1	0	1	0	0	1

Các tham số hệ thống điều khiển Fanuc thuộc một trong ba nhóm, được chuyên biệt trong khoảng cho phép:

- Mã nhị phân.
- Mục nhập đơn vị.
- Giá trị xác lập.

Các nhóm này sử dụng các giá trị nhập khác nhau. *Nhập nhị phân* chỉ có thể nhập số 0 hoặc 1 theo định dạng dữ liệu bit, 0 đến 127 cho kiểu byte. *Nhập đơn vị* có phạm vi rộng hơn, đơn vị có thể là mm, inch, mm/min, in/min, độ, millisecond,... *Giá trị* cũng có thể được chuyên biệt trong khoảng cho trước, ví dụ, số trong khoảng 0-99, 0-99999 hoặc +127 đến −127, ...

Ví dụ về *nhập nhị phân* là sự lựa chọn giữa *hai tùy chọn*. Ví dụ, tính năng *dry run*, chỉ có thể được xác lập là *effective* hoặc *ineffective*. Để lựa chọn, số bit tùy ý của tham số được xác lập là 0 để dry run là effective và là 1 để dry run là ineffective.

Nhập đơn vị, ví dụ, được dùng để xác lập hệ thống số gia, đơn vị kích thước. Máy tính nói chung không phân biệt giữa hệ mét và inch, chỉ coi đó là các số. Chỉ người dùng *và* sự xác lập tham số, mới giúp cho bộ điều khiển nhận biết 0.001mm hay .0001 inch là số gia nhỏ nhất. Xác lập tham số lưu tốc độ cắt cực đại trên từng trục, tốc độ trục chính cực đại,... Các giá trị đó không được phép xác lập cao hơn giá trị máy có thể hỗ trợ. Trục phân độ với số gia tối thiểu là 1^0, sẽ *không* trở thành trục quay với số gia 0.001^0. Do tham số này được xác lập với giá trị thấp hơn, ngay cả nếu điều đó là có thể. ***Xác lập đó là sai và có thể dẫn đến sự cố nghiêm trọng.***

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Để hiểu rõ hơn về các tham số hệ thống là các cần tối ưu. Thực tế, chúng được chọn trên cơ sở sử dụng phổ biến. Nhiều xác lập bảo toàn giá trị của chúng, vì các lý do an toàn.

CNC, dưới đây sẽ liệu kê sự phân loại tham số của hệ thống điều khiển Fanuc (đa số chỉ có ý nghĩa đối với chuyên gia về điều chỉnh và sửa chữa hệ thống điều khiển):

Các tham số về xác lập
Các tham số về dữ liệu điều khiển trục
Hệ tọa độ
Lượng ăn dao
Điều khiển tăng tốc / giảm tốc
Servo
DI/DO
MDI, EDIT và CRT
Chương trình
Xuất trục nối tiếp
Hiển thị đồ thị
Giao diện I/O
Giới hạn hành trình
Bù sai số bước
Bù độ nghiêng
Bù độ thẳng.
Điều khiển trục chính.
Điều chỉnh dao cắt
Chu kỳ hoàn tất
Lập tỉ lệ và quay tọa độ
Vượt qua góc tự động
Nội suy xoắn
Định vị một chiều
Lệnh Macro của người dùng.
Tái khởi động chương trình.
Nhập tín hiệu bỏ qua tốc độ cao
Bù dụng cụ cắt tự động.
Quản lý tuổi bền dụng cụ cắt.
Điều khiển trục ổ dao
Điều khiển biên dạng chính xác cao
Bảo trì, sửa chữa .... *và các tham số khác*

Bạn cần bảo quản cẩn thận danh sách các xác lập tham số gốc do nhà chế tạo cung cấp.

Cẩn thận trọng khi thay đổi các tham số hệ thống điều khiển!

Nhiều tham số được cập nhật định kỳ trong quá trình xử lý chương trình. Người vận hành CNC thường không chú ý đến hoạt động này. Thật ra không cần phải giám sát hoạt động đó. Nguyên tắc quan sát an toàn nhất là sau khi các tham số đã được người có trách nhiệm xác lập, mọi thay đổi *tạm thời* cần thiết đối với công việc cho trước đều phải được thực hiện thông qua chương trình CNC. Nếu cần các thay đổi *lâu dài*, người có trách nhiệm được chỉ định thực hiện điều đó.

Các mặc định hệ thống

Nhiều xác lập tham số trong bộ điều khiển vào thời điểm đặt mua máy đã được nhà chế tạo nhập sẵn dưới dạng các *lựa chọn bắt buộc*, các *lựa chọn thích hợp nhất* và *lựa chọn thông dụng nhất*. Điều đó không có nghĩa là chúng sẽ

Tập hợp các giá trị tham số được thiết lập vào thời điểm lắp đặt máy được gọi là xác lập *mặc định*. Khi có điện nguồn đến bộ điều khiển, không có giá trị nào được chuyển đến các tham số, do chưa có chương trình được sử dụng. Tuy nhiên, một số xác lập trở nên hoạt động một cách tự động mà *không có* chương trình bên ngoài. Ví dụ, sự điều chỉnh bán kính dao cắt sẽ được xóa một cách tự động khi khởi động hệ thống điều khiển, chế độ chu kỳ cố định và điều chỉnh chiều dài dao cắt cũng được xóa. Bộ điều khiển *"giả định"* một số điều kiện được ưa thích hơn các điều kiện khác. Nhiều người vận hành có lẽ đồng ý với hầu hết các xác lập ban đầu này, tuy không phải là tất cả. Một số xác lập có thể chuyên biệt hóa bằng cách thay đổi các xác lập tham số. Các xác lập đó sẽ là lâu dài và tạo ra *"mặc định" mới*.

Luôn luôn ghi lại mọi thay đổi đối với các tham số!

Khi máy CNC được cấp điện, phần mềm bên trong xác lập một số tham số hiện hữu theo điều kiện mặc định của chúng, do các kỹ sư thiết kế, không phải mọi tham số hệ thống, chỉ một số tham số có thể có điều kiện giả định, được coi là *giá trị mặc định*.

Ví dụ, chuyển động của dụng cụ cắt có ba chế độ cơ bản – chuyển động *nhanh*, chuyển động *tuyến tính*, và chuyển động *tròn*. Xác lập chuyển động mặc định do một tham số điều khiển. Chỉ một xác lập có thể hoạt động khi khởi động. Xác lập nào? Câu trả lời *tùy thuộc vào xác lập tham số*. Nhiều tham số có thể được xác lập trước theo trạng thái mong muốn. Trong ví dụ này, chỉ có chế độ *nhanh* hoặc *tuyến tính* là có thể xác lập theo mặc định.

Hầu hết các điều khiển đều xác lập chuyển động *tuyến tính* là mặc định (Lệnh G01) khi khởi động, hoàn toàn vì *lý do an toàn*. Khi các trục máy được chuyển động bằng tay, xác lập tham số này không còn tác dụng. Nếu giá trị lệnh trục được nhập bằng tay, thông qua chương trình hoặc từ bảng điều khiển, kết quả sẽ là chuyển động của dụng cụ cắt. Nếu lệnh chuyển động không được chuyên biệt, hệ thống sẽ sử dụng chế độ lệnh đã xác lập sẵn là mặc định trong các tham số. Do chế độ mặc định là chuyển động tuyến tính G01, kết quả là *điều kiện sai*, do hệ thống *chưa có tốc độ cắt*. Ở đây chưa có xác lập lượng ăn dao cần thiết đối với lệnh G01, chuyển động nhanh sẽ được thực hiện do chế độ này không cần lập trình lượng ăn dao.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Bảng tài liệu xử lý có thể được lưu trữ trong bộ nhớ điều khiển trên máy CNC trong xưởng cơ khí. Trừ khi có lý do chính đáng để thay đổi, mặc định đối với các điều khiển tương tự đều như nhau.

Dung lượng nhớ

Các chương trình CNC có thể được lưu trong bộ nhớ điều khiển, kích cỡ chương trình chỉ bị giới hạn theo dung lượng nhớ của bộ điều khiển. Dung lượng này được đo theo nhiều cách, có nguồn gốc tương đương với *chiều dài băng từ* tính theo mét, hoặc feet, gần đây được tính theo *số byte* hoặc số *trang màn hình*. Dung lượng nhớ tối thiểu của bộ điều khiển máy tiện CNC là 20 m (66 ft) băng từ. Đây là phương pháp cũ còn lại cho đến ngày nay. Trên các hệ thống phay CNC, các yêu cầu nhớ dựa trên các chuẩn nói chung lớn hơn dung lượng nhớ tối thiểu là 80 m (263 ft). Dung lượng thực tế có thể bổ sung thêm dung lượng nhớ cho hệ thống điều khiển. Dung lượng nhớ tối thiểu của bộ điều khiển khác nhau giữa các máy, bạn cần xem kỹ các đặc tính kỹ thuật do nhà chế tạo cung cấp.

Các phương pháp hiện đại đo dung lượng nhớ thường sử dụng *bytes* làm đơn vị thay cho chiều dài băng từ đã quá lạc hậu. Byte là đơn vị nhớ nhỏ nhất và gần tương đương với một ký tự trong chương trình.

Dung lượng nhớ của hệ thống điều khiển phải đủ lớn để lưu chương trình CNC dài nhất trên cơ sở lâu dài. Điều này yêu cầu sự quy hoạch trước khi đặt mua máy CNC. Ví dụ, trong gia công khuôn mẫu (gia công nhựa) ba chiều hoặc gia công tốc độ cao, chi phí cho dung lượng nhớ bổ sung có thể rất cao. Tuy chi phí là vấn đề cần xem xét, nhưng có thể còn có các phương pháp khác tin cậy và rẻ tiền hơn.

Phương pháp thứ hai là chạy chương trình CNC từ máy tính cá nhân. Phần mềm truyền thông và dây cáp được dùng để nối kết máy tính với hệ thống CNC. Phiên bản đơn giản nhất là *truyền* chương trình CNC từ máy tính này đến máy khác. Khả năng hiện nay là dùng phần mềm và các dây cáp có thể thực sự chạy máy CNC từ máy tính cá nhân, không cần tải trước chương trình đó vào CNC. Phương pháp này thường được gọi là "*truyền chương trình*" hoặc "*nhập bit*". Khi vận hành từ máy tính cá nhân, chương trình CNC có thể dài tùy theo dung lượng nhớ trong ổ cứng máy tính.

Hầu hết các chương trình CNC đều được đưa vào bộ nhớ nội của hệ thống điều khiển. Nhiều bộ điều khiển sử dụng số lượng ký tự khả dụng hoặc chiều dài tương đương băng từ. Dưới đây là một số công thức có thể được dùng để tính toán dung lượng nhớ gần đúng.

➲ **Công thức 1:**

Để tìm chiều dài tính theo *mét*, khi biết dung lượng nhớ tính theo số lượng *ký tự*, bạn hãy dùng công thức sau:

$$S_m = N_c \times .00254$$

S_m = Dung lượng lưu trữ tính theo mét.
N_c = Dung lượng nhớ (số lượng ký tự).

➲ **Công thức 2:**

Để tìm chiều dài tính theo *feet*, khi biết dung lượng nhớ tính theo số lượng *ký tự*, bạn hãy dùng công thức sau:

$$S_f = \frac{N_c}{120}$$

S_f = Dung lượng lưu trữ tính theo feet.
N_c = Dung lượng nhớ (số lượng ký tự).

➲ **Công thức 3:**

Tìm số lượng ký tự trong chương trình đã cho, nếu biết dung lượng nhớ tính theo mét.

$$C = \frac{m}{.00254}$$

C: số lượng ký tự khả dụng
m: Dung lượng nhớ (mét)

➲ **Công thức 4:**

Tìm số lượng ký tự, nếu biết dung lượng nhớ tính theo feet

$$C = f \times 120$$

C: Số lượng ký tự khả dụng.
f: Dung lượng nhớ (feet)

Các bộ điều khiển Fanuc hiện đại biểu thị dung lượng nhớ khả dụng theo số lượng *trang hiển thị còn trống*. Kiểu dữ liệu này rất khó chuyển đổi sang các đơn vị khác.

Trong trường hợp dung lượng nhớ khả dụng quá nhỏ để lưu chương trình lớn, một số kỹ thuật được dùng để giảm thiểu vấn đề, ví dụ các phương pháp giảm chiều dài chương trình (Chương 49).

DỪNG CHƯƠNG TRÌNH BẰNG TAY

Nếu cần dừng chương trình trong khi đang xử lý, hệ thống điều khiển đưa ra vài phương pháp để thực hiện điều đó, sử dụng bảng vận hành của máy. Các tính năng phổ biến của kiểu dừng chương trình này là công tắc hai chiều hoặc nút nhấn, đối với thao tác *single block*, *feedhold*, và *emergency stop*.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Thao tác single block

Mục đích chung của chương trình là điều khiển máy công cụ một cách tự động và theo chuỗi thứ tự trong chế độ liên tục. Chương trình là chuỗi các lệnh được định dạng, được viết theo các dòng mã riêng rẽ, được gọi là *block* (khối mã). Block và các khái niệm khối mã sẽ được trình bày trong các chương kế tiếp. Tất cả các lệnh chương trình trong một khối được xử lý theo *single instruction*. Hệ thống điều khiển nhận các khối theo thứ tự, từ trên xuống dưới và theo thứ tự của chúng trong chương trình. Nói chung máy CNC chạy theo chế độ liên tục, các block được xử lý một cách tự động, lần lượt block này đến block kế tiếp. Công tắc *Single Block* trên bảng vận hành được dùng để tạm dừng sự thực thi chương trình. Trong chế độ Single Block, chỉ thực thi một block chương trình mỗi khi nhấn phím *Cycle Start*. Trên bảng vận hành, chế độ Single Block có thể được sử dụng riêng rẽ hoặc kết hợp với các xác lập khác, giúp cho chương trình chạy nhanh và chính xác hơn.

Feedhold

Feedhold là nút nhấn đặc biệt ở trên bảng vận hành, gần nút *Cycle Start*. Khi nhấn nút này trong chế độ chuyển động trục nhanh, tuyến tính, hoặc tròn, sự chuyển động sẽ dừng ngay. Điều này tác động đồng thời cho tất cả các trục hoạt động. Tính năng này rất thuận tiện khi xác lập máy hoặc khi gia công chi tiết thứ nhất. Một số kiểu chuyển động ngăn chặn chức năng này của feedhold hoặc xóa bỏ chúng. Ví dụ, chế độ cắt ren làm cho công tắc này không hoạt động.

Sự kích hoạt feedhold trên máy không làm thay đổi các giá trị chương trình, chỉ tác động đến sự chuyển động. Công tắc feedhold sẽ được chiếu sáng (với đèn đỏ), khi chế độ này còn tác dụng. Nhà lập trình CNC có thể vượt qua feedhold từ *bên trong* chương trình, nếu có các lý do đặc biệt.

Emergency Stop

Mọi máy CNC đều có nút nhấn hình nấm, màu đỏ, ở nơi dễ tiếp cận trên máy công cụ. Nút này có ghi rõ *Emergency Stop* hoặc *E-Stop*. Khi nhấn nút này, *mọi hoạt động của máy đều dừng lại một cách tức thời*. Điện nguồn cung cấp bị ngắt và máy sẽ phải khởi động lại. Công tắc Emergency Stop (dừng khẩn cấp) là tính năng an toàn bắt buộc trên mọi máy công cụ CNC.

Nhấn nút E-Stop không phải lúc nào cũng là cách *tốt nhất*, thậm chí *duy nhất* để dừng sự vận hành máy. Các bộ điều khiển hiện đại còn cung cấp các tính năng khác, được thiết kế để tránh va đập giữa dụng cụ cắt và chi tiết hoặc đồ gá. Như phần trên đã đề cập, nút feedhold chỉ là một tùy chọn, cùng với các tính năng khác. Nếu buộc phải sử dụng E-Stop thì đó phải là *giải pháp cuối cùng* khi các tính năng khác đòi hỏi thời gian dài hơn mức có thể chấp nhận.

Đối với một số hoạt động máy, tác dụng của *Emergency Stop* không phải lúc nào cũng rõ rệt. Ví dụ, trục chính đòi hỏi thời gian giảm dần tốc độ trước khi dừng hẳn.

NHẬP DỮ LIỆU BẰNG TAY – MDI

Máy CNC không phải lúc nào cũng vận hành bằng chương trình. Trong khi xác lập chi tiết, người vận hành CNC thực hiện nhiều thao tác đòi hỏi sự chuyển động của máy, quay trục chính, thay dao cắt,... Không có thiết bị cơ học trên máy CNC. Nút xoay (*Manual Pulse Generator – MPG bộ tạo xung bằng tay*) là thiết bị điện tử, không phải là cơ học. Để vận hành máy CNC mà không cần các thiết bị cơ học cổ điển, hệ thống điều khiển cung cấp tính năng *Manual Data Input-MDI* (nhập dữ liệu bằng tay).

Manual Data Input cho phép nhập dữ liệu chương trình mỗi lần *một khối lệnh chương trình*. Nếu quá nhiều khối lệnh được nhập một cách lặp lại, chẳng hạn chương trình dài, quá trình này sẽ rất không hiệu quả. Trong khi xác lập hoặc đối với các mục đích tương tự, từ MDI mỗi lần có thể nhập một hoặc vài khối lệnh.

Để tiếp cận chế độ MDI, cần chọn phím MDI trên bảng vận hành. Điều này sẽ mở màn hình hiển thị với trạng thái hiện hành của hệ thống. Không phải tất cả, nhưng đa số các mã lập trình đều được phép trong chế độ MDI. Định dạng của chúng đồng nhất với định dạng của chương trình CNC ở dạng viết. Đây là một lĩnh vực, nơi người vận hành CNC hoạt động như nhà lập trình. Điều quan trọng là người vận hành CNC phải được đào tạo ít nhất ở mức cơ sở lập trình CNC, có khả năng xử lý các khối lệnh xác lập đối với *Manual Data Input*.

VƯỢT QUA DỮ LIỆU CHƯƠNG TRÌNH

Mọi thiết bị CNC đều được thiết kế với các công tắc đặc biệt có cùng một tính năng – chúng cho phép người vận hành *vượt qua* tốc độ đã được lập trình của trục chính hoặc của chuyển động trục. Ví dụ, lượng ăn dao 15in/min trong chương trình gây ra sự va đập dao cắt. Người vận hành biết, bằng cách *tăng* lượng ăn

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

dao hoặc *giảm* tốc độ trục chính và giải quyết được sự va đập đó. Có thể thay đổi lượng ăn dao hoặc tốc độ trục chính bằng cách biên tập chương trình, nhưng phương pháp này đôi khi không hiệu quả. Sự "thực nghiệm" có thể là cần thiết trong khi thực sự cắt gọt để tìm giá trị xác lập tối ưu. Các công tắc xoay, cho phép vượt qua dữ liệu chương trình, có lẽ là phương tiện thích hợp nhất, do có thể sử dụng để thử nghiệm trong khi gia công. Có bốn công tắc loại này trên hầu hết các bảng điều khiển.

- **Rapid feedrate override** (rapid traverse) (điều chỉnh chuyển động nhanh của máy công cụ).
- **Spindle speed override** (điều chỉnh tốc độ trục chính v/ph đã lập trình).
- **Feedrate override** (cutting feedrate) (điều chỉnh tốc độ cắt đã lập trình).
- **Dry run mode** (đưa chuyển động cắt sang tốc độ biến thiên).

Các công tắc override có thể được dùng riêng rẽ hoặc chung với nhau. Chúng khả dụng trên bộ điều khiển, giúp cho công việc trở nên dễ dàng hơn đối với cả nhà lập trình và người vận hành. Người vận hành không cần "thí nghiệm" với các tốc độ và lượng ăn dao bằng cách liên tục biên tập chương trình và nhà lập trình có thể tự do hơn khi xác lập các giá trị hợp lý cho tốc độ trục chính và lượng ăn dao. Sự hiện hữu của các công tắc override không có nghĩa là tùy ý lập trình các giá trị cắt gọt bất kỳ. Override thực chất là các công cụ tinh chỉnh, chương trình luôn luôn phải phản ánh các điều kiện gia công chi tiết. Các công tắc override không thực hiện các thay đổi chương trình, chỉ trao cơ hội cho người vận hành CNC để sau này có thể chỉnh sửa chương trình nhằm phản ánh các điều kiện cắt gọt tối ưu. Được sử dụng một cách hợp lý, các công tắc override có thể tiết kiệm thời gian lập trình và thời gian xác lập trên máy CNC.

Rapid Motion Override

Các chuyển động nhanh được chọn trong chương trình CNC bằng lệnh được chuẩn bị sẵn không cần lượng ăn dao riêng biệt. Nếu máy được thiết kế để chuyển động với 500in/min (12700mm/min) trong chế độ nhanh, tốc độ này không xuất hiện trong chương trình. Thay vào đó, bạn gọi chế độ chuyển động nhanh bằng cách lập trình với lệnh G00. Trong khi thực thi chương trình, mọi chuyển động trong chế độ G00 sẽ theo tốc độ cố định của nhà chế tạo. Chương trình này sẽ chạy nhanh hơn trên máy có định mức chuyển động nhanh cao so với trên máy có định mức chuyển động nhanh thấp.

Trong khi xác lập, chuyển động nhanh có thể đòi hỏi sự điều khiển nào đó để minh chứng chương trình, khi các tốc độ chuyển động nhanh cao là không thích hợp. Sau khi chương trình đã được chứng minh, tốc độ nhanh có thể được áp dụng với giá trị cực đại. Các máy CNC được trang bị công tắc *rapid override* cho phép các xác lập chuyển động nhanh tạm thời. Ở trên bảng điều khiển, công tắc này có thể được xác lập theo một trong bốn giá trị. Ba giá trị tính theo *phần trăm* tốc độ cực đại, là 100%, 50% và 25%, chuyển đổi giữa chúng, tốc độ chuyển động nhanh sẽ thay đổi. Ví dụ, nếu tốc độ nhanh cực đại là 500 in/min (12700 mm/ph), các tốc giảm sẽ là 250 in/min (6350 mm/min) với xác lập 50% và 125 in/min (3175 mm/min) với giá trị xác lập 25%.

Vị trí thứ tư của công tắc này không ghi giá trị phần trăm, mà thường là F1 hoặc ký hiệu nhỏ. Trong xác lập này, tốc độ chuyển động nhanh thậm chí có thể chậm hơn xác lập 25%. Hệ thống điều khiển cho phép lựa chọn *chuyên biệt* theo giá trị tùy ý giữa 0 và 100%. Xác lập mặc định thường là 10% tốc độ chuyển động nhanh cực đại. Xác lập F1 không được cao hơn 25% và chỉ có thể thực hiện thông qua sự xác lập tham số hệ thống. Bạn cần bảo đảm mọi người làm việc trên máy CNC đó chú ý đặc biệt đến các thay đổi, nếu có.

Spindle Speed Override

Logic được dùng cho ứng dụng rapid rate override cũng có thể được sử dụng cho spindle speed override. Thay đổi cần thiết có thể được thiết lập trong khi cắt gọt thực sự bằng cách dùng công tắc *spindle speed override* ở trên bảng điều khiển. Ví dụ, nếu tốc độ trục chính được lập trình 1000 v/ph là quá cao hoặc quá thấp, bạn có thể tạm thời thay đổi bằng công tắc này. Trong khi cắt gọt thực tế, người vận hành CNC có thể thử nghiệm với công tắc spindle speed override để tìm tốc độ tối ưu theo các điều kiện cắt gọt cụ thể. Phương pháp này nhanh hơn nhiều so với "thử nghiệm" bằng cách thay đổi các giá trị chương trình.

Công tắc spindle speed override có thể là liên tục trên một số bộ điều khiển hoặc lựa chọn các số gia 10% thường trong khoảng 50 – 120% tốc độ trục chính đã lập trình. Tốc độ được lập trình là 1000v/ph có thể được thiết lập lại theo các giá trị 500, 600, 700, 800, 900, 1000, 1100, 1200 v/ph. Khoảng giá trị này cho phép người vận hành CNC có tính linh hoạt cao khi tối ưu hóa sự quay trục chính theo các điều kiện cắt gọt cụ thể. Tuy nhiên, ở đây có thể có một nhược điểm. Sự thay đổi tốc độ trục chính tối ưu có thể chỉ áp dụng cho một trong

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại: http://www.me-cad.com/

nhiều dụng cụ cắt được dùng trong chương trình đó. Phương pháp nên dùng là tìm tốc độ tối ưu cho từng dao cắt, ghi lại, sau đó thay đổi chương trình một cách tương ứng, sao cho *tất cả các dụng cụ cắt* đều có thể được sử dụng ở xác lập spindle override 100% trong sản xuất.

So sánh các số gia trên công tắc spindle override với các số gia trên công tắc rapid motion override và feedrate override sẽ cho thấy khoảng giới hạn lớn hơn. Lý do về khoảng tốc độ trục chính 50 – 120% là tính an toàn.

Để thay đổi xác lập override đã chọn theo 100% tốc độ trong chương trình, bạn có thể tính toán tốc độ trục chính. Nếu tốc độ trục chính được lập thành là 1200 v/ph cho dụng cụ nào đó luôn luôn xác lập theo 80%, bạn nên chỉnh sửa giá trị trong chương trình là 960 v/ph, sau đó sử dụng giá trị 100%. Công thức rất đơn giản:

$$S^n = S_p \times p \times 0.01$$

S_n = Tốc độ trục chính mới.
S_p = Tốc độ trục chính lập trình.
p = Số phần trăm spindle override

Override đối với tốc độ trục chính đã lập trình trên máy CNC chỉ nhằm một mục đích, thiết lập sự quay trục chính để có các điều kiện cắt gọt tốt nhất.

Feedrate Override

Công tắc override được dùng nhiều nhất là công tắc thay đổi tốc độ cắt đã lập trình. Đối với các điều khiển phay, tốc độ cắt được lập trình theo *in/min* hoặc *m/min*. Đối với các điều khiển tiện, tốc độ cắt được lập trình theo *in/vòng* hoặc *mm/vòng*. Tốc độ cắt trên máy tiện tính theo đơn vị phút (min) chỉ dùng trong các trường hợp khi trục chính không quay mà cần điều khiển lượng ăn dao.

Sự tính toán tốc độ cắt mới, dựa trên xác lập feedrate override, tương tự cách tính tốc độ trục chính:

$$F_n = F_p \times p \times 0.01$$

F_n = Tốc độ cắt mới.
F_p = Tốc độ cắt lập trình
p = Số phần trăm feedrate override.

Tốc độ cắt có thể thay đổi (override) trong khoảng rộng, thường là 0-200%, ít nhất cũng trong khoảng 0–150%. Khi công tắc feedrate override được xác lập theo 0%, máy CNC sẽ dừng chuyển động cắt. Một số máy CNC không có xác lập 0%, tối thiểu là 10%. Tốc độ cắt cực đại 150% hoặc 200% sẽ cắt 1.5 hoặc 2.0 lần nhanh hơn so với giá trị đã lập trình.

Có thể có các tình huống, khi sử dụng feedrate override sẽ làm hư hại chi tiết gia công, dụng cụ cắt, hoặc cả hai, chẳng hạn các chu kỳ cắt ren bằng tarô hoặc bằng dao một lưỡi cắt. Các nguyên công này đòi hỏi sự quay trục chính đồng bộ với lượng ăn dao. Trong các trường hợp đó, feedrate override sẽ trở nên *không hiệu quả*. Feedrate override sẽ hiệu quả, nếu các lệnh chuyển động tiêu chuẩn G00 và G01 được dùng để lập trình các chuyển động cắt ren. Lệnh cắt ren với dao một lưỡi cắt G32, các chu kỳ cố định cắt ren bằng tarô G74 và G84, chu kỳ tiện ren G92 và G76 sẽ xóa feedrate override được xây dựng trong phần mềm. Tất cả các lệnh này và các lệnh liên quan sẽ được trình bày chi tiết trong các chương kế tiếp.

Sự vận hành Dry Run

Dry run là dạng override đặc biệt, được kích hoạt từ bảng điều khiển bằng công tắc *Dry Run*, chỉ tác dụng trực tiếp đối với lượng ăn dao và cho phép tốc độ cắt cao hơn nhiều so với sự gia công thực tế. Trong thực tiễn, điều này có nghĩa là chương trình có thể được thực thi nhanh hơn nhiều so với sử dụng feedrate override theo xác lập cực đại. Sự gia công thực không xảy ra khi công tắc dry run ở chế độ hoạt động.

Mục đích và lợi ích của dry run là gì? Mục đích là kiểm tra tính toàn vẹn của chương trình *trước khi* người vận hành gia công chi tiết thứ nhất. Lợi ích chủ yếu là tiết kiệm thời gian trong khi kiểm chứng chương trình mà không gia công thực sự. Trong quá trình dry run, chi tiết thường *không* được lắp vào máy. Nếu lắp chi tiết vào máy và sử dụng chế độ dry run, điều rất quan trọng là phải có các khoảng hở đủ lớn. Nói chung, điều đó có nghĩa là *dịch chuyển dao cắt ra xa chi tiết*. Chương trình sau đó được thực thi "khô", không thực sự cắt gọt, không sử dụng chất làm nguội. Do lượng ăn dao lớn trong dry run, không thể gia công chi tiết một cách an toàn. Trong quá trình dry run (chạy khô), chương trình có thể được kiểm tra về mọi lỗi sai khả dĩ, trừ các lỗi liên quan với sự tiếp xúc của dao cắt với vật liệu.

Dry run là sự hỗ trợ xác lập rất hiệu quả để minh chứng tính toàn vẹn chung của chương trình CNC. Sau khi chương trình được chứng minh trong khi chạy khô (dry run), người vận hành CNC có thể tập trung vào các phần của chương trình bảo đảm sự gia công thực sự. Dry run có thể được sử dụng kết hợp với các tính năng khác trên bảng điều khiển.

Xóa chế độ dry run trước khi gia công

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Z Axis Neglect

Công cụ tiếp theo rất hữu dụng để kiểm tra các chương trình trên trung tâm gia công CNC (không phải máy tiện) là công tắc hai trạng thái trên bảng vận hành có tên là *Z Axis Neglect* hoặc *Z Axis Ignore*. Khi kích hoạt công tắc này, chuyển động bất kỳ được dùng để gia công biên dạng của chi tiết, sẽ vô nghĩa khi tạm thời dừng chuyển động của một trong các trục này. Bằng cách tạm thời bỏ qua trục Z, người vận hành CNC có thể tập trung vào sự minh chứng độ chính xác của biên dạng chi tiết mà không cần quan tâm đến chiều sâu. Phương pháp kiểm tra chương trình này phải được thực hiện mà không lắp chi tiết gia công (và không có chất làm nguội). *Ở đây cần rất cẩn thận!* Điều quan trọng là *kích hoạt* hoặc *ngắt công tắc* này vào *đúng thời điểm*. Nếu chuyển động trục Z bị bỏ qua trước khi nhấn phím Cycle Start, mọi lệnh trục Z kế tiếp đều bị bỏ qua. Nếu chuyển động trục Z được kích hoạt hoặc ngắt trong khi thực thi chương trình, vị trí trục Z có thể sẽ mất chính xác.

Công tắc Z axis neglect có thể được sử dụng trong cả hai chế độ tự động và bằng tay. Bạn chỉ cần bảo đảm chuyển động dọc theo trục Z trở lại trạng thái *kích hoạt*, sau khi hoàn tất sự kiểm tra chương trình. Một số máy CNC đòi hỏi cài đặt lại các xác lập vị trí trục Z.

Manual Absolute Setting

Nếu có tính năng này trên bộ điều khiển (một số bộ điều khiển sử dụng xác lập này một các tự động), sẽ cho phép người vận hành CNC bắt đầu lại chương trình ở giữa quá trình xử lý. Manual Absolute Setting (xác lập tuyệt đối bằng tay) có thể tiết kiệm thời gian, đặc biệt khi xử lý các chương trình dài. Công tắc xác lập *Manual Absolute* không phải là tùy chọn phổ biến. Để mở rộng, về chức năng xác lập này có quan hệ với xác lập *Sequence Return* (xác lập quay lại chuỗi thứ tự). Bạn hãy xem kỹ tài liệu của máy công cụ CNC trước khi sử dụng các tính năng này.

Sequence Return

Sequence Return là chức năng được điều khiển bằng công tắc hoặc phím trên bảng điều khiển. Mục đích là cho phép người vận hành CNC khởi động chương trình từ giữa chương trình bị tạm dừng. Một số chức năng đã lập trình vẫn được lưu giữ (thường là tốc độ và lượng ăn dao), số khác cần nhập bằng phím *Manual Date Input*. Sự vận hành của chức năng này có quan hệ chặt chẽ với thiết kế máy công cụ được ghi trong sổ tay hướng dẫn vận hành máy. Chức năng Sequence Return rất hữu ích khi dao cắt bị gãy trong quá trình xử lý các chương trình dài, cho phép tiết kiệm thời gian sản xuất nếu được sử dụng hợp lý.

Khóa các hàm phụ

Có ba chức năng khả dụng để vận hành máy CNC là một phần trong nhóm *"các chức năng phụ"*, bao gồm:

Khóa các hàm Miscellaneous	Khóa các hàm M
Khóa các hàm Spindle	Khóa các hàm S
Khóa các hàm Tool	Khóa các hàm T

Các hàm phụ sẽ được trình bầy ở cuối chương này, nói chung liên quan đến các khía cạnh *công nghệ* của lập trình CNC. Chúng điều khiển các chức năng của máy, chẳng hạn sự quay trục chính, hướng trục chính, lựa chọn chất làm nguội, thay dao, phân độ bàn máy... ở mức độ thấp hơn, chúng còn điều khiển các chức năng chương trình, chẳng hạn dừng chương trình bắt buộc hoặc tùy ý, dòng chương trình con, đóng chương trình...

Khi khóa các hàm phụ, mọi hàm M liên quan đến máy, mọi hàm trục chính S, và mọi hàm dụng cụ cắt T đều tạm dừng. Một số nhà chế tạo máy công cụ ưa thích thuật ngữ *MST Lock* hơn thuật ngữ *Anxiliary Functions Lock*. MST là viết tắt của *Miscellaneous*, *Spindle* và *Tool* là các hàm chương trình sẽ bị khóa.

Ứng dụng sự khóa các hàm này chỉ giới hạn trong phạm vi xác lập máy và minh chứng chương trình, không dùng trong quá trình gia công trên máy.

Machine Lock

Chức năng *Machine Lock* (khóa máy) là tính năng điều khiển để minh chứng chương trình. Phần trên đã trình bày chức năng *Z Axis Neglect* và khóa các hàm phụ. Chức năng *Z Axis Neglect* chỉ dừng chuyển động của trục X và *Auxiliary Functions Lock* (còn gọi là *MST lock*) khóa các hàm M, S, và T. Chức năng *Machine Lock* cũng ở trên bảng điều khiển. Khi kích hoạt chức năng này, chuyển động của *tất cả* các trục đều bị khóa. Điều này có vẻ hơi lạ khi kiểm tra chương trình bằng cách khóa tất cả các chế độ dao cắt, nhưng có lý do để sử dụng chức năng này. Machine lock cung cấp cơ hội cho người vận hành khả năng kiểm tra chương trình hầu như hoàn toàn không xảy ra va chạm.

Khi kích hoạt Machine Lock, *chỉ chuyển động trục bị khóa*. Mọi hàm chương trình khác

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

đều được thực thi bình thường, kể cả các hàm trục chính và thay dao cụ. Chức năng này có thể được sử dụng độc lập hoặc kết hợp với các chức năng khác để phát hiện lỗi chương trình, nếu có. Có lẽ các lỗi phổ biến nhất là lỗi cú pháp và các chức năng điều chỉnh dao cắt.

Ứng dụng thực tế

Nhiều tính năng điều khiển trình bày trong chương này được sử dụng *chung* với nhau. Ví dụ *Dry Run* được dùng chung với *Z Axis Neglect* hoặc *Auxiliary Functions Lock*. Nếu biết các tính năng khả dụng, người vận hành CNC có thể lựa chọn phù hợp với yêu cầu vào thời điểm cụ thể. Có nhiều lĩnh vực với tầm quan trọng ngang nhau, người vận hành CNC cần tập trung khi xác lập quy trình gia công được chạy chương trình mới. Nhiều tính năng của bộ điều khiển được thiết kế để giúp cho công việc của người vận hành trở nên dễ dàng hơn. Chúng cho phép tập trung vào một hoặc hai vấn đề thay vì sự phức tạp của toàn bộ chương trình. Các tính năng này đã được trình bày tương đối chi tiết, phần tiếp theo sẽ đề cập các ứng dụng thực tế.

Khi khởi động để chạy chương trình mới, người vận hành CNC cần tuân theo các quy tắc xác định. Ví dụ, chi tiết thứ nhất cần được kiểm nghiệm với xác lập chuyển động nhanh 25% hoặc 50% tốc độ nhanh khả dụng. Đây là xác lập tương đối chậm, cho phép người vận hành giám sát tính toàn vẹn của sự xử lý chương trình và các chi tiết cụ thể. Các chi tiết này có thể gồm khả năng khoảng hở không đủ giữa dao cắt và vật liệu, xem xét quỹ đạo chạy dao, v.v...

Người vận hành CNC có vài tác vụ cần thực hiện đồng thời. Một số tác vụ bao gồm giám sát tốc độ trục chính, lượng ăn dao, chuyển động của dao cắt, thay dao, làm nguội v.v... Phương pháp tiếp cận hợp lý và cẩn thận cho phép bạn tin tưởng vào tính toàn vẹn của chương trình CNC. Có thể phải đến chi tiết thứ hai hoặc thứ ba người vận hành mới nhận thức về sự tối ưu hóa các giá trị cắt gọt, chẳng hạn tốc độ trục chính và lượng ăn dao. Sự tối ưu hóa này phản ánh lượng ăn dao và tốc độ lý tưởng đối với chi tiết gia công.

Sau khi người vận hành tìm được các giá trị cần thay đổi trong chương trình, chương trình đó phải được chỉnh sửa để phản ánh các thay đổi đó. Không chỉ đối với công việc đang thực hiện mà còng dùng cho công việc được lặp lại trong tương lai. Về nguyên tắc, mục đích của nhà lập trình và người vận hành CNC là thực hiện công việc với hiệu suất 100%. Hiệu suất máy hoàn toàn có thể đạt được khi kết hợp nỗ lực của người vận hành và nhà lập trình. Nhà lập trình luôn luôn nỗ lực để đạt hiệu suất 100% tại bàn làm việc, sau đó sẽ tiếp tục cải tiến chương trình khi được áp dụng trong các điều kiện thực tế.

CÁC TÙY CHỌN HỆ THỐNG

Các tính năng tùy chọn hệ thống CNC tương tự các tùy chọn trên xe hơi, máy móc hoặc các trang thiết bị khác. Tùy theo nhà chế tạo và chiến lược kinh doanh của họ, một tính năng trên CNC có thể là tiêu chuẩn hoặc tùy chọn.

Dưới đây là một số tính năng điều khiển có thể là tùy chọn hoặc tiêu chuẩn trên hệ thống cụ thể. Điều quan trọng cần chú ý là:

Nội dung tiếp theo trình bày hầu hết các tính năng điều khiển, bất kể chúng là tiêu chuẩn hay tùy chọn trên hệ thống CNC. Trong điều kiện cụ thể, người dùng sẽ biết các tính năng nào là tùy chọn được lắp trên hệ thống điều khiển của họ.

Hiển thị đồ họa

Biểu diễn đồ thị quỹ đạo chạy dao trên màn hình là một trong các tùy chọn quan trọng nhất. Không nên nhầm lẫn tùy chọn này với kiểu lập trình thông thường bất kỳ, cũng sử dụng giao diện đồ thị quỹ đạo chạy dao. Khi không có sự lập trình được hỗ trợ bằng máy tính (CAM), hiển thị đồ họa trên bảng điều khiển là rất cần thiết. Mọi nhà lập trình và người vận hành đều muốn quan sát chuyển động của dụng cụ cắt trên màn hình, dù đơn sắc hay màu, *trước khi* thực sự gia công trên máy.

Tùy chọn đồ họa thường hiển thị các trục máy và hai con trỏ để phóng to hoặc thu nhỏ. Khi kiểm tra quỹ đạo chạy dao, các dụng cụ cắt được phân biệt bằng màu sắc hoặc độ sáng. Các chuyển động nhanh được biểu thị bằng đường nét đứt, chuyển động cắt bằng đường nét liền. Nếu áp dụng chức năng đồ họa trong khi gia công, có thể quan sát các chuyển động dao cắt trên màn hình.

Sự lập tỉ lệ tăng hoặc giảm trên màn hiển thị cho phép đánh giá chuyển động dao theo tổng thể hoặc từng phần riêng rẽ. Nhiều bộ điều khiển còn có sự mô phỏng quỹ đạo chạy dao thực tế, hình dạng chi tiết và dao cắt được xác lập trước, sau đó quan sát trên màn hình.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Đo trong khi gia công

Trong các quy trình gia công tự động, sự kiểm tra định kỳ và sự điều chỉnh dung sai kích thước là điều bắt buộc. Khi dụng cụ cắt bị mòn dần, hoặc do các nguyên nhân khác, kích thước có thể lệch khỏi khoảng dung sai cho phép. Sử dụng thiết bị dò (cảm biến) và chương trình thích hợp, tùy chọn *In-process Gauging* (đo trong khi gia công) cung cấp giải pháp chuẩn cho vấn đề này. Chương trình gia công CNC đối với tùy chọn *In-process Gauging* có các tính năng định dạng đặc thù, được viết theo tham số, sử dụng tùy chọn khác của hệ điều khiển – *Custom Macros* (đôi khi còn được gọi là *User Macros*) cung cấp sự lập trình kiểu biến thiên.

Nếu xưởng máy CNC sử dụng tùy chọn *In–process Gauging*, nói chung các tùy chọn điều khiển khác cũng được cài đặt và khả dụng cho nhà lập trình CNC. Các tùy chọn phổ biến bao gồm phần mềm dò tìm (cảm biến), quản lý tuổi bền dao cắt, macros, ... Công nghệ này hơi vượt quá sự lập trình CNC tiêu chuẩn, dù có quan hệ chặt chẽ và sử dụng thường xuyên. Nếu bạn sử dụng công nghệ điều khiển số, bạn nên quan tâm đến các tùy chọn đó.

Giới hạn hành trình được lưu giữ

Sự xác định diện tích trên máy tiện CNC hoặc hình khối trên trung tâm gia công CNC cho phép làm việc an toàn, có thể được lưu theo tham số của hệ thống điều khiển được gọi là *stored stroke limits* (giới hạn hành trình lưu giữ). Các giới hạn hành trình này được thiết kế để tránh sự va đập giữa dụng cụ cắt và đồ gá, máy công cụ hoặc chi tiết. Diện tích (2D) hoặc hình khối (3D) có thể được xác định theo *enable* (cho phép) nhập dao cắt hoặc *disable* (không cho phép) nhập dao cắt. Điều này có thể được xác lập bằng tay trên máy hoặc nếu khả dụng, bằng mục nhập chương trình. Một số bộ điều khiển chỉ cho phép xác định một diện tích hoặc thể tích, số khác có thể cho phép nhiều hơn.

Khi tùy chọn này hoạt động và máy CNC phát hiện sự chuyển động trong chương trình xảy ra bên trong vùng bị cấm, sẽ dẫn đến điều kiện lỗi và sự gia công sẽ dừng lại. Các ứng dụng có thể gồm vùng có ụ động, đồ gá, mâm cặp, bàn xoay và cả chi tiết có hình dạng đặc biệt.

Nhập kích thước vẽ

Tùy chọn, có vẻ ít được nhắc tới, là phương pháp lập trình bằng cách nhập các kích thước từ bản vẽ kỹ thuật. Khả năng nhập các tọa độ đã biết, bán kính, vạt góc và các góc đã biết trực tiếp từ bản vẽ, là rất hấp dẫn đối với nhà lập trình. Tùy chọn này, nếu được dùng, phải được cài đặt trên tất cả các máy CNC trong xưởng, để sử dụng các tính năng lập trình một cách hiệu quả.

Chu kỳ gia công

Cả điều khiển phay và tiện đều cung cấp nhiều chu kỳ gia công. Các chu kỳ gia công phay thường được gọi là *chu kỳ cố định* hoặc *chu kỳ hoàn tất*. Chúng đơn giản hóa các gia công điểm - điểm đơn giản, chẳng hạn khoan, chuốt, doa, cắt ren. Một số hệ thống CNC còn cung cấp các chu kỳ phay mặt, phay hốc, lỗ định hình, v.v..

Các máy tiện CNC cũng có nhiều chu kỳ gia công khả dụng để cắt gọt vật liệu bằng cách cắt thô tự động, cắt tinh biên dạng, tiện côn, cắt rãnh, và tiện ren. Bộ điều khiển Fanuc gọi các chu kỳ này là *Multiple Repetitive Cycles* (chu kỳ lặp lại nhiều lần)

Tất cả các chu kỳ này đều được thiết kế để dễ lập trình hơn và thay đổi trên máy nhanh hơn. Chúng được xây dựng *bên trong* bộ điều khiển và *không thể thay đổi*. Nhà lập trình cung cấp các giá trị cắt gọt trong khi chuẩn bị chương trình bằng cách dùng lệnh gọi chu kỳ tương ứng. Toàn bộ sự xử lý được thực hiện tự động do hệ thống CNC điều khiển. Chúng luôn luôn là các dự án lập trình đặc biệt, không thể sử dụng chu kỳ bất kỳ và cần được lập trình bằng tay hoặc sử dụng máy tính riêng.

Tạo hoạt hình dụng cụ cắt

Nhiều đồ thị quỹ đạo dụng cụ cắt được biểu thị bằng các đường hoặc các cung đơn giản. Vị trí dao cắt hiện hành thường là điểm cuối của cung hoặc đường trên màn hình. Tuy phương pháp hiển thị này đối với chuyển động của dụng cụ cắt có tính đồ thị rất hữu ích, nhưng vẫn có hai nhược điểm. Hình dạng dao cắt và vật liệu được cắt gọt không thể quan sát trên màn hình và sự mô phỏng quỹ đạo dao cắt chỉ có tác dụng hạn chế. Nhiều bộ điều khiển hiện đại có tính năng đồ họa được gọi là *Cutting Tool Animation* (tạo hoạt hình dụng cụ cắt). Nếu khả dụng trên bộ điều khiển, tính năng này sẽ hiển thị phôi (chi tiết), đồ gá, và hình dạng dao cắt. Khi thực thi chương trình, người vận hành CNC có sự hỗ trợ hình ảnh rất chính xác để minh chứng chương trình. Mỗi phần tử đồ họa đều được hiển thị bằng màu riêng. Kích cỡ phôi, đồ gá, và hình dạng dao cắt có thể được xác lập trước theo tỉ lệ chính xác và nhiều dạng dao cắt có thể được lưu để tái sử dụng.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tùy thiểu một được coi là có khả năng tương thích CAD/CAM được xây dựng trong hệ thống điều khiển độc lập.

Nối kết với thiết bị bên ngoài

Máy tính CNC có thể được nối với thiết bị bên ngoài, thường là máy tính khác. Mỗi bộ CNC đều có một hoặc nhiều đầu nối, được thiết kế đặc biệt để giao tiếp với các thiết bị ngoại vi. Cái phổ biến nhất là RS – 232 (tiêu chuẩn EIA), được thiết kế để truyền thông giữa hai máy tính. Sự xác lập nối kết với thiết bị bên ngoài là ứng dụng được chuyên biệt hóa. Người vận hành CNC sử dụng nối kết đó để truyền các chương trình và các xác lập giữa hai máy tính, thường nhằm mục đích báo quản hoặc lưu trữ.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

5 QUY HOẠCH CHƯƠNG TRÌNH

Sự triển khai chương trình CNC bất kỳ đều bắt đầu với quá trình lập kế hoạch (quy hoạch) rất cẩn thận. Quá trình đó bắt đầu từ *bản vẽ* kỹ thuật của chi tiết cần gia công. Trước khi gia công chi tiết đó, nhiều bước cần được khảo sát và đánh giá một cách cẩn thận. Càng nhiều nỗ lực trong giai đoạn quy hoạch chương trình, chương trình kết quả sẽ càng tốt.

CÁC BƯỚC QUY HOẠCH CHƯƠNG TRÌNH

Các bước cần thiết trong quy hoạch chương trình được quyết định theo bản chất của công việc. Ở đây không có công thức chung cho mọi công việc, tuy nhiên bạn có thể xem xét các bước cơ bản dưới đây:

- ❑ Thông tin ban đầu/Các tính năng máy công cụ
- ❑ Độ phức tạp của chi tiết
- ❑ Lập trình bằng tay / Lập trình máy tính hóa
- ❑ Quy trình lập trình chung
- ❑ Bản vẽ chi tiết / số liệu kỹ thuật
- ❑ Các phương pháp gia công/đặc tính kỹ thuật của vật liệu
- ❑ Thứ tự gia công (các bước gia công, thứ tự nguyên công)
- ❑ Lựa chọn dụng cụ cắt
- ❑ Xác lập chi tiết
- ❑ Các quyết định công nghệ
- ❑ Bản vẽ phác chi tiết và các tính toán
- ❑ Các vấn đề chất lượng trong lập trình CNC

Các bước nêu trên chỉ có tính gợi ý, chúng hoàn toàn linh hoạt và có thể thay đổi tùy theo chi tiết và các điều kiện gia công.

LẬP TRÌNH BẰNG TAY

Lập trình bằng tay (không dùng máy tính) đã từng là phương pháp thông dụng để chuẩn bị chương trình chi tiết từ nhiều năm trước. Các bộ điều khiển CNC mới giúp cho sự lập trình bằng tay trở nên dễ dàng hơn trước bằng cách sử dụng chu kỳ gia công lặp lại hoặc cố định, lập trình kiểu biến thiên, mô phỏng chuyển động chạy dao, mục nhập toán học tiêu chuẩn, và nhiều tính năng tiết kiệm thời gian khác. Trong lập trình bằng tay, mọi tính năng đều được thực hiện bằng tay, với sự hỗ trợ của máy tính bỏ túi, hoàn toàn không lập trình trên máy vi tính. Dữ liệu lập trình có thể được truyền qua máy CNC, qua dây cáp, sử dụng máy tính để bàn hoặc xách tay (laptop). Quy trình này nhanh hơn và tin cậy hơn so với các phương pháp khác. các chương trình ngắn cũng có thể được nhập bằng tay, từ bàn phím, trực tiếp trên máy CNC. Băng đục lỗ là phương tiện nhập chương trình phổ biến trong quá khứ, hầu như không còn sử dụng trong các xưởng cơ khí hiện nay.

Nhược điểm

Lập trình bằng tay có một số nhược điểm. Có lẽ phổ biến nhất là thời gian cần thiết để viết chương trình CNC với đầy đủ chức năng. Các tính toán bằng tay kiểm chứng, và các hoạt động liên quan trong lập trình bằng tay đều tốn nhiều thời gian. Các nhược điểm khác bao gồm số lượng lỗi khá lớn, không kiểm chứng quỹ đạo chạy dao, khi thay đổi chương trình, ...

Ưu điểm

Lập trình chi tiết bằng tay có vài ưu điểm đặc biệt. Lập trình bằng tay có cường độ rất cao đòi hỏi toàn bộ tâm trí và sức lực của nhà lập trình CNC và sự sáng tạo hầu như không hạn chế trong quá trình triển khai cấu trúc chương trình. Lập trình bằng tay tuy còn vài nhược điểm nhưng sẽ luyện cho bạn tính kỷ luật chặt chẽ, tính tổ chức trong khi viết chương trình. Quá trình này buộc nhà lập trình phải hiểu rõ các kỹ thuật lập trình đến từng chi tiết nhỏ. Nhiều kỹ năng hữu dụng trong lập trình bằng tay được áp dụng trực tiếp vào lập trình CAD/CAM. Nhà lập trình phải biết những điều xảy ra ở mọi thời điểm và lý do điều đó xảy ra. Điều rất quant rọng là *hiểu sâu sắc* từng chi tiết trong khi phát triển chương trình.

CAD/CAM VÀ CNC

Nhu cầu về cải thiện hiệu suất và độ chính xác trong lập trình CNC là lý do chính để phát triển nhiều phương pháp sử dụng máy tính để chuẩn bị các chương trình gia công chi tiết. Lập trình CNC với máy tính đã xuất hiện từ nhiều năm trước. Ban đầu, ở dạng lập trình dựa trên ngôn ngữ, chẳng hạn APT™ hoặc Compact II™. Từ cuối thập kỷ 1970, CAD/ CAM đã có vai trò quan trọng do bổ sung đặc tính hiển thị cho quá trình lập trình. *CAD/CAM* là viết tắt của *Computer Aided Design* và *Computer Aided*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

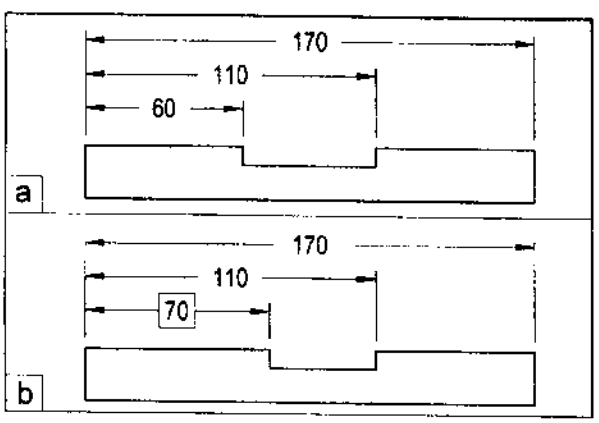
Manufacturing. CAD bao quát lĩnh vực thiết kế và vẽ kỹ thuật. CAM bao quát lĩnh vực sản xuất máy tính hóa, trong khi lập trình CNC chỉ là một phần nhỏ trong CAM. Toàn bộ đối tượng của CAD/CAM rộng hơn nhiều so với thiết kế, vẽ, và lập trình. Đây là một phần của công nghệ hiện đại, được gọi là *CIM–Computer Intergrated Manufacturing* (sản xuất tích hợp máy tính).

Trong lĩnh vực điều khiển số, máy tính có vai trò chính trong thời gian dài. Các điều khiển máy trở nên tinh vi hơn, tích hợp các kỹ thuật mới nhất về xử lý dữ liệu, lưu giữ, đồ họa quỹ đạo chạy dao, chu kỳ gia công... Hiện nay các chương trình có thể được chuẩn bị với sự sử dụng máy tính và giao diện đồ họa. Chi phí hầu như không còn là vấn đề lớn, các xưởng cơ khí nhỏ hoàn toàn có đủ khả năng trang bị hệ thống lập trình. Các hệ thống này còn được ưa chuộng rộng rãi do có tính linh hoạt cao. Hệ thống lập trình máy tính hóa không chỉ dành cho lập trình, tất cả các nhiệm vụ liên quan, thường do nhà lập trình thực hiện, đều có thể được thực thi trên máy tính đó. Ví dụ, quản lý dụng cụ cắt dự trữ, cơ sở dữ liệu dùng cho các chương trình gia công, thông tin kỹ thuật về vật liệu, dụng cụ cắt, máy công cụ, ... Máy tính này còn được dùng để tải các chương trình CNC.

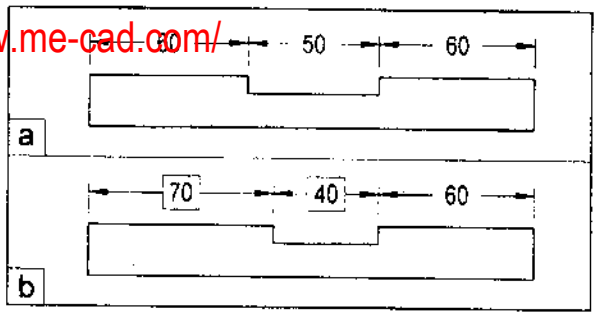
Sự tích hợp

Cốt lõi trong CIM là sự *tích hợp*. Điều đó có nghĩa là mọi phần tử của quy trình sản xuất đều được đặt vào mối quan hệ chung và xem xét chúng như một tổng thể nhằm tăng hiệu suất, giảm chi phí và tiết kiệm thời gian. Ý tưởng chính để tích hợp thành công là tránh sự sao chép. Một trong các nguyên tắc quan trọng nhất về sử dụng phần mềm máy tính CAD/CAM là:

KHÔNG THỰC HIỆN HAI LẦN MỘT VIỆC BẤT KỲ



Hình 5.1. *Chương trình sử dụng kích thước tuyệt đối, chỉ cần một thay đổi trong chương trình.*

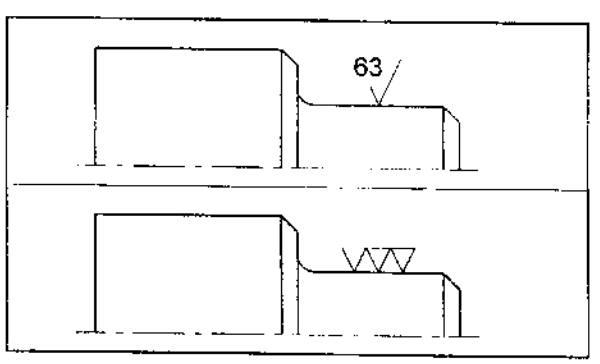


Hình 5.2. *Chương trình sử dụng kích thước theo số gia (tương đối) cần hai (hoặc nhiều) thay đổi trong chương trình.*

Khi thực hiện bản vẽ trong phần mềm CAD (ví dụ *AutoCAD*), sau đó lại thực hiện trong phần mềm CAM (chẳng hạn *Mastercam*), đó là sự lặp lại, một việc hai lần. Sự lặp lại dẫn đến lỗi sai. Để tránh lặp lại, hầu hết các hệ thống CAD đều có phương pháp truyền thiết kế sang hệ thống CAM đã chọn để sử dụng cho lập trình CNC. Các phương pháp truyền này được thực hiện thông qua các tập tin DXF hoặc IGES đặc biệt. *DXF* là viết tắt của *Data Exchange Files* (tập tin trao đổi dữ liệu) hoặc *Drawing Exchange Files*, còn *IGES* là từ viết tắt của *Initial Graphic Exchange Specification* (đặc tính kỹ thuật trao đổi đồ họa ban đầu). Sau khi dạng hình học được truyền từ hệ thống CAD sang CAM, (dạng đặc biệt của bộ định dạng), phần mềm máy tính sẽ chuẩn bị chương trình gia công, cho phép tải trực tiếp vào máy CNC.

CÁC BƯỚC LẬP TRÌNH ĐIỂN HÌNH

Quy hoạch chương trình CNC không khác với các quy hoạch khác, có thể thực hiện tại nhà, ở nơi làm việc, trong văn phòng, ... cần tiếp cận theo cách thức *logic và có phương pháp*. Các quyết định khác liên quan với phương pháp để đạt đến các mục tiêu đã chọn theo cách thức an toàn và hiệu quả. Phương pháp tiệm tiến không chỉ phân tích các vấn đề



Hình 5.3. *Ký hiệu độ bóng bề mặt trên bản vẽ: hệ Anh (trên) và hệ mét (dưới)*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

riêng rẽ khi chúng nhất triết mà còn phải giải quyết chúng *trước khi* chuyển sang bước kế tiếp.

Các mục tiếp theo tạo thành chuỗi thứ tự logic các nhiệm vụ cần thực hiện trong lập trình CNC. Các mục này chỉ được xếp theo *thứ tự có tính gợi ý*, cần tiếp tục xem xét và đánh giá. Thứ tự đó có thể thay đổi để phản ánh các điều kiện hoặc thói quen làm việc. Một số mục có thể được bỏ qua hoặc hơi dư.

1. Nghiên cứu thông tin ban đầu (bản vẽ và các phương pháp)
2. Đánh giá phôi và vật liệu
3. Các đặc tính kỹ thuật của máy công cụ
4. Các chức năng của hệ thống điều khiển
5. Chuỗi thứ tự các nguyên công
6. Lựa chọn và sắp xếp dụng cụ cắt
7. Xác lập chi tiết
8. Dữ liệu công nghệ (tốc độ, lượng ăn dao,...)
9. Xác định quỹ đạo chạy dao
10. Bản phác thảo chi tiết và các tính toán
11. Viết chương trình và chuẩn bị để truyền cho CNC
12. Kiểm tra và gỡ rối chương trình
13. Ghi và lập tài liệu cho chương trình

Chỉ có một mục đích trong quy hoạch chương trình CNC và đó là hoàn tất mọi khối lệnh ở dạng chương trình cho phép gia công CNC hiệu quả, an toàn và không có lỗi. Các bước lập trình nêu trên có thể thay đổi, chẳng hạn dụng cụ cắt được chọn *trước* hoặc *sau* khi xác định sự xác lập chi tiết? Các phương pháp lập trình chi tiết bằng tay có thể được dùng một cách hiệu quả? Có cần các bản vẽ chi tiết? Bạn đừng ngại chỉnh sửa kể cả các bước lập trình được coi là lý tưởng, dù chỉ tạm thời cho công việc cụ thể, hoặc lâu dài để phản ánh phong cách lập trình CNC. *Bạn cần nhớ, không có các bước lập trình lý tưởng.*

BẢN VẼ CHI TIẾT

Bản vẽ chi tiết là tài liệu quan trọng nhất trong lập trình CNC, nêu rõ hình dạng, kích thước, dung sai, độ bóng bề mặt và nhiều yêu cầu khác đối với chi tiết hoàn tất. Chi tiết phức tạp có thể có nhiều bản vẽ, với các hình chiếu, các tiết diện (mặt cắt), ... Nhà lập trình trước hết cần đánh giá toàn bộ dữ liệu trên bản vẽ, lựa chọn các số liệu thích ứng để triển khai chương trình cụ thể. Cả nhà lập trình và người thiết kế đều phải hiểu các phương pháp của nhau và tìm cơ sở chung để toàn bộ quá trình thiết kế và chế tạo gắn kết với nhau và hiệu quả hơn.

Kích thước

Kích thước trên bản vẽ chi tiết được ghi theo hệ đơn vị *Anh* hoặc *mét*. Các kích thước riêng rẽ có thể được quy chiếu từ mốc chuẩn xác định, hoặc có thể là liên tiếp, được đo từ kích thước trước đó. Nói chung, cả hai kiểu kích thước đều được sử dụng trên cùng một bản vẽ. Khi viết chương trình có thể sẽ thuận tiện hơn nếu chuyển mọi kích thước liên tiếp – theo *số gia* – sang kích thước chuẩn - *tuyệt đối*. Hầu hết các chương trình CNC đều hưởng lợi từ các bản vẽ sử dụng kích thước chuẩn hoặc tuyệt đối. Tương tự, khi triển khai chương trình con để diễn dịch quỹ đạo chạy dao, phương pháp lập trình theo số gia có lẽ là lựa chọn đúng – và lựa chọn này tùy thuộc vào ứng dụng cụ thể. Phương pháp lập trình phổ biến nhất đối với các máy CNC sử dụng phương pháp khi kích thước *tuyệt đối* (Hình 5.1), chủ yếu do khả năng biên tập dễ dàng trong hệ thống CNC.

Với hệ thống ghi kích thước tuyệt đối, nhiều thay đổi trong chương trình có thể được thực hiện chỉ bằng một chỉnh sửa. Phương pháp số gia đòi hỏi ít nhất hai chỉnh sửa. Các khác biệt giữa hai hệ thống kích thước có thể được so sánh trên Hình 5.1 sử dụng phương pháp ghi kích thước tuyệt đối và Hình 5.2 sử dụng cách ghi kích thước theo số gia. Thuật ngữ số gia được dùng nhiều trong CNC, trong vẽ kỹ thuật thuật ngữ tương đương là kích thước tương đối. Cả hai hình này đều nêu rõ: a) bản vẽ trước khi chỉnh sửa và b) bản vẽ sau khi chỉnh sửa.

Dung sai

Để bảo đảm gia công chính xác, hầu hết các kích thước chi tiết đều có khoảng chuyên biệt về độ sai lệch được chấp nhận so với kích cỡ danh định, trong phạm vi hệ quy chiếu. Ví dụ, dung sai hệ Anh +.001/-.000 inch sẽ khác với dung sai hệ mét +0.1/-0.0mm. Các kích thước kiểu này thường là kích thước chính và phải được duy trì trong gia công CNC. Có thể khẳng định người vận hành CNC chịu trách nhiệm chính về bảo đảm kích thước chi tiết trong khoảng dung sai cho phép (với điều kiện chương trình là chuẩn xác), nhưng nhà lập trình CNC có thể giúp cho nhiệm vụ của người vận hành trở nên dễ dàng hơn. Bạn hãy xét ví dụ dưới đây về tiện CNC:

➲ Kích thước lỗ trên bản vẽ ∅75 +0.00/-0.05, kích thước thực trong chương trình là bao nhiêu?

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ở đây có vài lựa chọn kích thước phù hợp. Giới hạn có thể lập trình là X75.0 và phía thấp của khoảng dung sai là X74.95. Kích thước giữa (trung bình) X74.975 là một lựa chọn. Các lựa chọn đều phải chính xác về toán học. Nhà lập trình CNC có tính sáng tạo không chỉ tìm các điểm toán học mà còn tìm các điểm kỹ thuật. Lưỡi cắt của dao bị mòn dần khi gia công nhiều chi tiết. Điều đó có nghĩa là người vận hành máy cần *tinh chỉnh* kích cỡ gia công bằng cách sử dụng *sự điều chỉnh mòn dao*, khả dụng trên hầu hết các hệ thống CNC. Sự can thiệp bằng tay trong quá trình gia công là chấp nhận được, nhưng nếu lạm dụng điều đó sẽ làm giảm năng suất và tăng chi phí chung.

Độ bóng bề mặt

Độ chính xác của chi tiết đòi hỏi chất lượng độ bóng bề mặt xác định. Bản vẽ kỹ thuật nêu rõ độ bóng yêu cầu cho các bề mặt khác nhau của chi tiết. Các bản vẽ theo hệ Anh biểu thị độ bóng theo *micro inch*, với *1 micro inch = 0.000001"* (một phần triệu inch). Bản vẽ hệ mét sử dụng đơn vị *micron*, với *1 micron = 0.001 mm* (một phần nghìn mm), ký hiệu micron là μ. Một số bản vẽ sử dụng các ký hiệu được nêu trên Hình 5.3.

Các yếu tố quan trọng tác động đến chất lượng bề mặt gồm tốc độ trục chính, lượng ăn dao, bán kính dụng cụ cắt và chiều sâu cắt. Nói chung, bán kính dao cắt càng lớn, lượng ăn dao càng chậm, độ bóng bề mặt càng cao. Thời gian chu kỳ sẽ dài hơn nhưng có thể bù lại bằng cách loại bớt các nguyên công kế tiếp, chẳng hạn mài tinh, mài rà, ...

Chỉnh sửa bản vẽ

Phần quan trọng kế tiếp của bản vẽ, thường bị các nhà lập trình CNC bỏ qua, biểu thị các thay đổi kỹ thuật (được gọi là *chỉnh sửa*) được thực hiện trên bản vẽ. Sử dụng các số hoặc các ký tự quy chiếu, người thiết kế sẽ nhận biết các thay đổi đó, thường với cả hai giá trị, cũ và mới, ví dụ:

```
REV.3 / DIMENSION 5.75 WAS 5.65
```

Chỉ các thay đổi mới nhất là quan trọng đối với sự triển khai chương trình. Bạn cần bảo đảm chương trình không chỉ phản ánh thiết kế kỹ thuật hiện hành, mà còn có các ký hiệu rõ ràng để phân biệt với các phiên bản cũ của chương trình đó. Nhiều nhà lập trình giữ bản sao của bản vẽ chi tiết tương ứng chương trình trong các tập tin do đó có thể tránh được sự hiểu nhầm sau này.

Một số công ty có đội ngũ các kỹ sư công nghệ hoặc lập kế hoạch gia công chịu trách nhiệm xác định quy trình công nghệ gia công. Những người này đưa ra chuỗi các hướng dẫn gia công, vạch chi tiết về hành trình của từng phôi qua các bước công nghệ. Họ phân chia nguyên công cho từng máy, triển khai thứ tự gia công và các phương pháp định vị, chọn dụng cụ cắt, ... Các hướng dẫn của họ được ghi lên bảng công nghệ kèm theo chi tiết gia công qua tất cả các bước chế tạo. Nếu có bảng công nghệ, bản sao của bảng đó là một phần trong tài liệu kỹ thuật. Một trong các mục đích của bảng công nghệ là cung cấp thông tin tối đa cho nhà lập trình để rút ngắn vòng quay giữa các chương trình. Ưu điểm chính của bảng công nghệ trong lập trình là bao quát mọi nguyên công cần thiết, cả CNC và cổ điển, do đó cung cấp khái quát về toàn bộ quy trình chế tạo. Bảng công nghệ chất lượng cao thường đưa ra quy trình công nghệ tương hợp gần gũi với các phương pháp lập trình gia công chi tiết.

Vì nhiều lý do, một số xưởng CNC thường không sử dụng bảng công nghệ hoặc tài liệu tương tự, do đó nhà lập trình CNC đồng thời phải là kỹ sư công nghệ. Môi trường đó cung cấp tính linh hoạt khá cao nhưng đòi hỏi kiến thức, kỹ năng và tinh thần trách nhiệm cao.

ĐẶC TÍNH KỸ THUẬT CỦA VẬT LIỆU

Nghiên cứu quan trọng trong quy hoạch lập trình là đánh giá *vật liệu phôi*. Phôi còn thô và chưa được gia công cắt gọt (tấm, thanh, vật đúc, vật rèn, ...) Một số phôi có thể được gia công sơ bộ, chuyển đến từ máy khác hoặc nguyên công kohác, có thể là nguyên khối hoặc có lỗ sẵn, với lượng dư cần gia công bằng CNC. Kích cỡ và hình dạng phôi xác định phương pháp gá lắp và định vị. Loại vật liệu (thép, gang, hợp kim đồng, hợp kim nhôm, ...) tác động đến sự lựa chọn dụng cụ cắt và điều kiện gia công cắt gọt.

Chương trình không thể lập quy hoạch nếu không biết chủng loại, kích cỡ, hình dạng và điều kiện của vật liệu.

Định mức tính gia công

Một vấn đề quan trọng trong các đặc tính kỹ thuật vật liệu là *tính gia công*. Các bảng và đồ thị về lượng ăn dao và tốc độ được đề nghị cho các vật liệu thông dụng thường được các công ty chế tạo dụng cụ cắt cung cấp các đồ thị này rất hữu dụng trong lập trình, đặc biệt khi chưa biết rõ rệt vật liệu gia công. Các giá trị

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

được [illegible] tốt, và sau này có thể được tối ưu hóa, khi biết rõ hơn các tính chất của vật liệu.

Định mức tính gia công theo đơn vị Anh là *feet / minute* (ft/min). Nói chung, có thể sử dụng các thuật ngữ *surface feet per minute* (feet bề mặt / min), *constant surface speed* (tốc độ bề mặt không đổi, CSS hoặc CS), *peripheral speed* (tốc độ chu vi) hoặc *surface speed* (tốc độ bề mặt). Trong hệ mét, định mức tính gia công được tính theo đơn vị m/min. trong cả hai trường hợp, tốc độ trục chính (v/ph) đối với đường kính dao cho trước (trên máy phay) hoặc đường kính chi tiết đã cho (đối với máy tiện) được tính toán, sử dụng các công thức chung. Đối với hệ Anh, tốc độ trục chính có thể được tính theo vòng/phút (v/ph).

$$r/min = \frac{12 \times ft/min}{\pi \times D}$$

Đối với hệ mét, công thức như sau:

$$r/min = \frac{1000 \times m/min}{\pi \times D}$$

r/min	= Vòng/phút (tốc độ trục chính S)
12	= Hệ số đổi foot sang inch.
1000	= Hệ số đổi m sang mm.
ft/min	= Tốc độ chu vi feet/phút.
m/min	= Tốc độ chu vi m/phút
π(pi)	= Hằng số.
D	= Đường kính dao (phay) hoặc đường kính phôi (tiện)

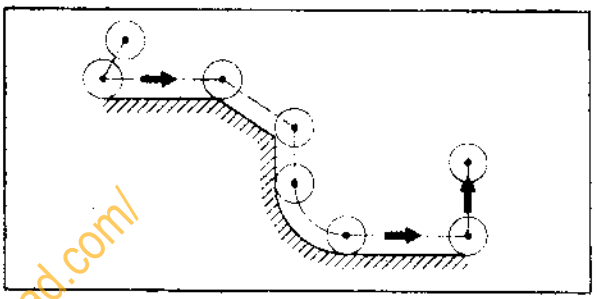
CHUỖI THỨ TỰ GIA CÔNG

Chuỗi thứ tự gia công xác định thứ tự các bước gia công. Kỹ năng và kinh nghiệm sẽ giúp quy hoạch lập trình hiệu quả, nhưng phương pháp tiếp cận cũng rất quan trọng. Chuỗi thức tự gia công phải có tính logic – ví dụ khoan phải được lập trình trước khi cắt ren, gia công thô trước gia công tinh,... Trong thứ tự logic đó, các chuyên biệt tiếp theo về thứ tự chuyển động dụng cụ cắt riêng rẽ cần được xác định cho từng dao cắt.

Phương pháp cơ bản để xác định chuỗi thứ tự gia công là đánh giá các nguyên công liên quan. Nói chung, chương trình cần được lập kế hoạch theo cách thức dụng cụ cắt, sau khi đã chọn, cần thực hiện sự cắt gọt ở mức cao nhất có thể trước khi thay dao khác. Trên hầu hết các máy CNC, thời gian cần thiết để định vị dao cắt thường ngắn hơn thời gian thay dao. Khảo sát thứ hai là khả năng lập trình các bước gia công thô trước, tiếp theo là gia công bán tinh, cuối cùng là gia công tinh. Điều đó có thể làm tăng thêm số lần thay dao, nhưng cho phép giảm số lần dịch chuyển phôi trong đồ gá hoặc định vị trong khi gia công. Yếu tố quan trọng là định vị hiện hành của dao khi hoàn tất một nguyên công. Ví dụ, khi khoan các lỗ theo thứ tự 1 –2 – 3 – 4, dao cắt kế tiếp (chẳng hạn doa, chuốt, cắt ren, ...) nên được lập trình theo thứ tự 4 – 3 – 2 – 1 để giảm các chuyển động dao (Hình 5.4).

T01 = Khoan định vị	T02 = Khoan	T03 = Tarô
Lỗ 1	Lỗ 4	Lỗ 1
Lỗ 2	Lỗ 3	Lỗ 2
Lỗ 3	Lỗ 2	Lỗ 3
Lỗ 4	Lỗ 1	Lỗ 4

Hình 5.4. *Thứ tự gia công (lấy dấu tâm, khoan, cắt ren)*



Hình 5.5. *Quỹ đạo chạy dao theo biên dạng (tiện hoặc phay)*

Thứ tự gia công này có thể thay đổi sau khi chọn các dụng cụ cắt và phương pháp gá lắp. Thứ tự ngược có thể không thích hợp trong các chương trình con.

Quy hoạch lập trình không phải là sự thực thi độc lập các bước riêng rẽ, ngược lại chúng liên hệ chặt chẽ với nhau theo thứ tự logic để đạt đến mục đích xác định.

LỰA CHỌN DỤNG CỤ CẮT

Lựa chọn giá dao và dụng cụ cắt là bước quan trọng trong quy hoạch lập trình CNC. Bộ dụng cụ cắt đầy đủ bao gồm dụng cụ cắt, giá dao và bộ kẹp hoặc định vị dao cắt (vít kẹp, đồ gá dao, mâm cặp, bàn phân độ, kẹp rút, kẹp ống, ...) Dụng cụ cắt đòi hỏi chú ý đặc biệt, do chúng rất đa dạng về chủng loại và tác động trực tiếp đến sự gia công.

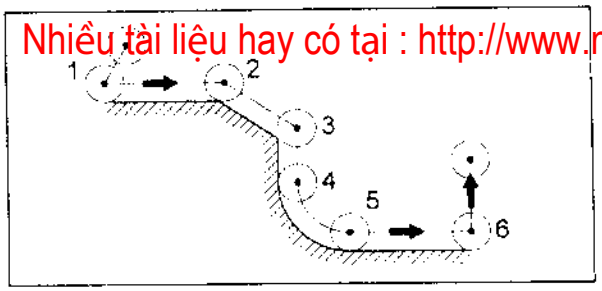
Dụng cụ cắt thường là lựa chọn quan trọng nhất. Khi lựa chọn phải dựa vào hai nhóm tiêu chuẩn cơ bản

- ❑ Hiệu quả trong sử dụng
- ❑ An toàn trong gia công

Sự sắp xếp các dụng cụ cắt theo thứ tự sử

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 5.6. Quỹ đạo chạy dao theo biên dạng với các điểm thay đổi đường biên.

dụng là vấn đề quan trọng trong quy hoạch chương trình CNC. Trên máy tiện CNC, mỗi dụng cụ cắt được gắn trên giá xoay xác định, bảo đảm *sự phân bố dao cắt* cân bằng giữa dụng cụ cắt ngắn và dài (chẳng hạn dao tiện trục và dao doa dài). Điều này là quan trọng để tránh khả năng các dao va chạm nhau khi thay dao. Vấn đề tiếp theo là thứ tự sử dụng từng dao cắt, đặc biệt trên các máy không có sự phân độ dao hai chiều. Hầu hết các trung tâm gia công đều sử dụng sự chọn dao theo kiểu ngẫu nhiên, trong đó thứ tự dao là không quan trọng, chỉ xét đường kính và trọng lượng dao.

Mọi chỉ số điều chỉnh dao và các mục nhập chương trình khác cần được ghi vào tài liệu dưới dạng *bảng dụng cụ cắt*. Tài liệu này được dùng để hướng dẫn người vận hành trong khi xác lập máy, nội dung cơ bản liên quan đến sự lựa chọn dao. Ví dụ, tài liệu có thể gồm mô tả dụng cụ cắt, đường kính và chiều dài, số lưỡi cắt, tốc độ và lượng ăn dao,...

GÁ LẮP CHI TIẾT

Quyết định kế tiếp trong quy hoạch chương trình liên quan đến sự gá lắp và xác lập chi tiết – *phương pháp gá lắp phôi, giá đỡ* dụng cụ cắt, đồ gá và định vị, số *nguyên công* cần thiết để hoàn tất quy trình công nghệ, vị trí để chọn *sự bắt đầu chương trình* (chuẩn gia công), ... Sự xác lập này là rất cần thiết và phải được thực hiện một cách hiệu quả.

Một số kiểu máy được thiết kế để giảm thời gian xác lập. Các trung tâm gia công nhiều trục hoặc máy tiện có thể gia công đồng thời hai hoặc nhiều chi tiết. Các tính năng đặc biệt chẳng hạn cấp phôi thanh trên máy tiện, tự động đổi dao, gá lắp kép trên bàn máy, ... đều nhằm giảm thời gian gia công và tăng năng suất cắt gọt.

Bảng xác lập

Tại giai đoạn quy hoạch lập trình này, sau khi quyết định sự xác lập, cần xây dựng *bảng xác lập*. Bảng này có thể là phác thảo đơn giản, được thiết kế chủ yếu để sử dụng tại máy, nêu rõ sự định hướng phôi khi lắp vào đồ gá, chỉ số điều chỉnh dao do liên quan với các yêu cầu đặc thù được thiết lập trong khi quy hoạch các bước trong chương trình (chẳng hạn vị trí kẹp chặt, kích thước kẹp rút,...). Bảng xác lập và bảng dụng cụ cắt có thể kết hợp thành một bảng thông tin chung. Hầu hết các nhà lập trình đều có bảng riêng tùy theo kinh nghiệm của họ.

QUYẾT ĐỊNH CÔNG NGHỆ

Bước kế tiếp trong quy hoạch chương trình CNC là chọn tốc độ trục chính, lượng ăn dao, chiều sâu cắt, chất làm nguội, ... Tất cả các yếu tố đã xét đều có ảnh hưởng đến sự gia công. Ví dụ, khoảng tốc độ trục chính khả dụng là cố định đối với máy CNC bất kỳ, nhưng kích cỡ dao cắt và loại vật liệu sẽ ảnh hưởng đến tốc độ và lượng ăn dao, công suất của máy công cụ sẽ giúp xác định lượng kim loại được cắt gọt một cách an toàn, ... Các yếu tố khác, ảnh hưởng đến thiết kế chương trình, bao gồm độ cứng vững gá lắp, chiều dài dao cắt, vật liệu dao cắt, ... Bạn không nên bỏ qua sự lựa chọn dung dịch làm nguội và chất bôi trơn, chúng là các yếu tố quan trọng ảnh hưởng đến chất lượng gia công.

Quỹ đạo dao cắt

Cốt lõi của lập trình CNC bất kỳ là xác định *quỹ đạo dao cắt*. Quá trình này chủ yếu là xác định các chuyển động chạy dao trong quan hệ với chi tiết.

Trong lập trình CNC, cần đặc biệt chú ý đến dao cắt đang chuyển động xung quanh chi tiết gia công. Nguyên tắc này áp dụng cho mọi máy công cụ CNC.

Yếu tố cốt lõi để hiểu nguyên tắc này là hình dung sự chuyển động của *dụng cụ cắt*, *không phải* chuyển động của máy. Sự khác biệt rõ nhất giữa lập trình trung tâm gia công và máy tiện là sự quay của dao cắt và sự quay của chi tiết gia công. Trong cả hai trường hợp, nhà lập trình đều phải suy nghĩ về *chuyển động của dao cắt xung quanh chi tiết gia công* (Hình 5.5).

Quỹ đạo chạy dao đối với mọi dụng cụ gia công biên dạng (profile) luôn luôn xem xét bán kính dao, bằng cách lập trình quỹ đạo khoảng cách tương đương theo tâm bán kính hoặc bằng cách sử dụng sự điều chỉnh bán kính dao cắt. Máy CNC phay hoặc tiện đều cung cấp các tính năng tiêu chuẩn về chuyển động nhanh, nội suy tuyến tính và nội suy vòng. Để tạo ra các quỹ đạo phức tạp, chẳng hạn chuyển động phay

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

xoắn, có thể sử dụng tùy chọn đặc biệt trong bộ điều khiển. Có hai nhóm quỹ đạo chạy dao cơ bản:

- ☐ Điểm – điểm *còn gọi là* *(theo)* vị trí
- ☐ Liên tục *còn gọi là* *(theo)* biên dạng

Chạy dao theo vị trí được dùng cho các gia công vị trí điểm, chẳng hạn khoan, chuốt, tarô ren và các nguyên công tương tự, *chạy dao theo biên dạng* tạo ra biên dạng. Trong cả hai trường hợp, dữ liệu lập trình quy chiếu theo vị trí của dao cắt khi hoàn tất một chuyển động xác định. Vị trí này được gọi là vị trí đích của dao cắt. (Hình 5.6).

Các vị trí bắt đầu và kết thúc của biên dạng đều được ghi rõ và là vị trí cho từng thay đổi biên dạng. Vị trí đích được gọi là *điểm thay đổi biên dạng*, cần được tính toán. Thứ tự các vị trí đích trong chương trình là rất quan trọng. Điều đó có nghĩa là *vị trí 1* của dao cắt là vị trí *đích* bắt đầu ở điểm *Start*, *vị trí 2* là *đích* bắt đầu ở điểm 1, *vị trí 3* là *đích* từ vị trí 2, ... cho đến khi đạt đến vị trí *End*. Nếu biên dạng là nguyên công phay, các đích được tính theo trục X và trục Y. Trong gia công tiện, chúng sẽ là các tọa độ X và Z.

Hầu hết các gia công biên dạng đều đòi hỏi ít nhất hai chuyển động dao cắt, ví dụ gia công thô và gia công tinh. Một phần trong lập trình là phân biệt lượng gia công thô. Khi lập trình các chuyển động không cắt gọt cũng cần chú ý như với chuyển động cắt gọt. Trọng tâm là giảm đến mức thấp nhất các chuyển động nhanh (không cắt gọt) của dao cắt và bảo đảm các khoảng hở an toàn.

Định mức công suất máy

Máy dụng cụ được phân loại theo công suất, lượng cắt lớn đòi hỏi công suất cao và ngược lại. Chiều sâu hoặc chiều rộng cắt quá lớn có thể làm gãy dao hoặc đứng máy. Các trường hợp đó là không thể chấp nhận và cần phải ngăn chặn. Đặc tính kỹ thuật máy CNC cho biết công suất động cơ ở trục máy, được tính theo kW (kilowatt) hoặc Horsepower (mã lực). Các công thức được dùng để tính công suất, tính lượng cắt gọt, hệ số mòn dao, chế độ cắt, ... công thức chuyển đổi giữa kW và HP (dựa trên đơn vị 1 HP = 550 foot – pounds per second – ft.lbs/s)

1 kW = 1.344 HP
1 HP = 0.746 kW

Chất làm nguội và bôi trơn

Khi dao cắt tiếp xúc với vật liệu (phôi) trong thời gian đủ dài, sẽ phát sinh lượng nhiệt khá lớn. Lưỡi cắt bị quá nhiệt, mòn dần và có thể bị mẻ hoặc gãy. Để tránh các khả năng này cần sử dụng chất làm nguội thích hợp.

Dầu hòa tan trong nước là chất làm nguội phổ biến nhất, chất làm nguội được pha trộn hợp lý sẽ giải nhiệt từ lưỡi cắt đồng thời có tác dụng như chất bôi trơn. Công dụng chính của bôi trơn là làm giảm ma sát và giúp sự cắt gọt trở nên dễ dàng hơn. Dòng chất làm nguội phải hướng đến lưỡi cắt, với ống mềm hoặc thông qua lỗ dẫn chất làm nguội trong dụng cụ cắt.

Không được dùng nước thuần túy làm chất làm nguội, do có thể gây hư hại nghiêm trọng cho máy công cụ.

Dụng cụ cắt bằng hợp kim gốm (cermet) thường được lập trình khô, *không dùng chất làm nguội*. Một số loại gang đúc không đòi hỏi dòng chất lỏng làm nguội, nhưng có thể dùng dầu ở dạng phun sương hoặc thổi không khí. Các chức năng làm nguội thay đổi tùy theo máy, do đó cần xem kỹ sổ tay hướng dẫn sử dụng máy.

Dòng chất làm nguội có thể được dùng để làm nguội chi tiết do đó đạt được dung sai tốt hơn. Ngoài ra chất làm nguội còn giúp đẩy phôi ra xa khu vực cắt, đặc biệt khi gia công lỗ sâu hoặc các hốc lõm.

Chất làm nguội liên quan đến lập trình CNC là xác định *thời điểm* cung cấp chất làm nguội trong chương trình. Lệnh M08 trong chương trình chỉ kích hoạt động cơ bơm chất làm nguội, bạn cần bảo đảm chất làm nguội thực sự đi đến lưỡi cắt *trước khi* tiếp xúc với chi tiết gia công. Sự lập trình sớm cho chất làm nguội thường tốt hơn là trễ.

PHÁC THẢO CHI TIẾT VÀ TÍNH TOÁN

Các chương trình được chuẩn bị bằng tay đòi hỏi một số tính toán. Phần tính toán này có thể gây khó khăn cho nhà lập trình, nhưng là bước cần thiết. Nhiều biên dạng phức tạp đòi hỏi thêm các tính toán, nhưng thường không quá khó. Hầu như mọi tính toán trong lập trình CNC đều có thể giải quyết bằng số học, đại số, và lượng giác. Toán học cao cấp – hình học giải tích, lượng giác cầu, phương trình vi phân, ma trận,... chủ yếu chỉ dùng khi lập trình khuôn mẫu phức tạp (gia công chất dẻo) khuôn đúc áp lực, khuôn dập nóng, ... trong các trường hợp đó, cần sử dụng hệ thống lập trình CAD/CAM.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Kiến thức lượng giác sẽ giúp bạn thực hiện các phép toán trong nhiều chương trình CNC. Khi làm việc với các biên dạng phức tạp, nói chung lời giải thực sự không quá khó, chủ yếu là khả năng và phương pháp đi đến lời giải đó.

Các tính toán sẽ dễ dàng hơn nếu được biểu diễn bằng hình ảnh. Các tính toán này thường cần bản vẽ phác chi tiết. Bản vẽ phác có thể được thực hiện bằng tay theo tỉ lệ gần đúng. Lập tỉ lệ cho bản phác thảo có một ưu điểm lớn, bạn có thể *thấy rõ các quan hệ*, so sánh các kích thước, quan hệ giữa các phần tử riêng rẽ, hình dạng của các phần tử nhỏ nhất... Tuy nhiên, bạn không nên lạm dụng bản phác thảo.

Không dùng bản phác thảo để dự đoán các kích thước chưa biết.

Lập tỉ lệ bản phác thảo là công việc thiếu tính chuyên nghiệp, thường là dấu hiệu của sự làm biếng và thiếu kiên nhẫn.

Các phương pháp nhận biết

Phác thảo tính toán có thể được thực hiện trực tiếp trên bản vẽ hoặc trên giấy. Mỗi phác thảo đều liên hệ với các tính toán. Sử dụng mã màu hoặc đánh số các điểm làm phương pháp nhận biết sẽ giúp bạn nhận thức bản vẽ rõ ràng hơn. Thay vì ghi tọa độ từng điểm thay đổi biên dạng, bạn hãy dùng các số quy chiếu điểm và lập bảng tọa độ riêng sử dụng các số quy chiếu (Hình 5.7).

Vị trí	Trục X	Trục Y	Trục Z

Hình 5.7. Ví dụ về bảng tọa độ (chưa ghi các giá trị)

Bảng này có thể được dùng cả cho phay và tiện, bằng cách điền vào cột tương ứng. Mục đích là triển khai phong cách lập trình ổn định từ chương trình này đến chương trình khác. Bạn hãy điền tất cả các giá trị, kể cả các giá trị không thay đổi. Bảng này sẽ là tài liệu quy chiếu rất tốt khi lập trình.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

6 CẤU TRÚC CHƯƠNG TRÌNH GIA CÔNG

Chương trình CNC gồm một chuỗi ký tự các khối lệnh liên quan đến sự gia công chi tiết. Mỗi khối lệnh được chuyên biệt trong định dạng hệ thống CNC có thể chấp nhận. Mỗi khối lệnh còn phải tương hợp với các đặc tính kỹ thuật của máy công cụ. Phương pháp nhập chương trình này có thể được định nghĩa là *sự sắp xếp gia công và các khối lệnh liên quan*, được viết theo định dạng của hệ thống CNC và hướng đến máy CNC cụ thể.

Các điều khiển có định dạng khác nhau, nhưng hầu hết đều tương tự nhau. Các khác biệt nhỏ tồn tại giữa các máy tùy theo nhà chế tạo, kể cả loại được trang bị cùng loại hệ thống điều khiển. Điều này là phổ biến, bạn có thể xem xét các yêu cầu chuyên biệt do các nhà chế tạo máy đặt ra đối với nhà chế tạo bộ điều khiển để đáp ứng các tính năng thiết kế máy đặc thù của họ. Các khác biệt này thường là nhỏ nhưng vẫn quan trọng đối với lập trình.

CÁC THUẬT NGỮ LẬP TRÌNH CƠ BẢN

Lĩnh vực CNC có các thuật ngữ và biệt ngữ riêng, có các viết tắt và mệnh đề chỉ những người trong nghề mới hiểu. Lập trình CNC chỉ là một phần nhỏ của gia công máy tính hóa và có nhiều mệnh đề riêng. Hầu hết các thuật ngữ đó đều liên quan với *cấu trúc* của chương trình.

Có *bốn thuật ngữ cơ bản* được dùng trong lập trình CNC, chúng xuất hiện trong các tài liệu, bài giảng, sách, báo chuyên môn... Các thuật ngữ này là cơ bản để hiểu thuật ngữ CNC chung:

Ký tự → Từ ngữ → Khối → Chương trình

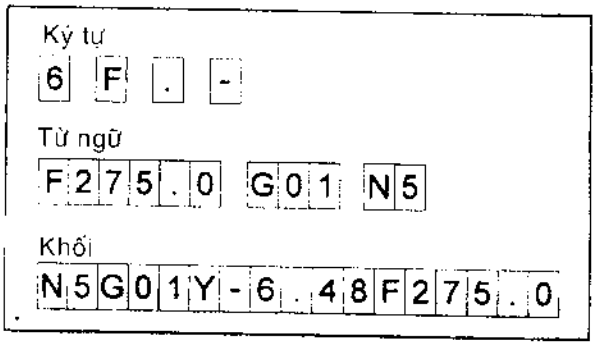
Các thuật ngữ này rất phổ biến và quan trọng trong lập trình CNC, ý nghĩa của chúng được giải thích như sau:

Ký tự

Ký tự là đơn vị nhỏ nhất của chương trình CNC, có thể có một trong ba dạng:

- ❑ Chữ số
- ❑ Chữ cái
- ❑ Ký hiệu

Các ký tự kết hợp với nhau thành các từ ngữ có nghĩa. Sự kết hợp các chữ số, chữ cái, và ký hiệu được gọi là mục nhập chương trình *số-chữ*.



Hình 6.1. Ví dụ về định dạng lập trình địa chỉ từ ngữ

Chữ số

Có mười chữ số, 0 đến 9, khả dụng trong chương trình để tạo ra các số. Các chữ số được dùng theo hai chế độ – một là các giá trị *nguyên* (số không có dấu thập phân) và thứ hai là các *số thực* (số có dấu thập phân). Số có thể có giá trị *dương* hoặc *âm*. Trên một số bộ điều khiển, số thực có thể được sử dụng có hoặc không có dấu thập phân. Các số được dùng trong hai chế độ này chỉ có thể nhập trong một khoảng do hệ thống điều khiển xác định.

Chữ cái

Về lý thuyết cả 26 chữ cái trong vần chữ cái đều có thể được sử dụng để lập trình. Hầu hết các hệ thống điều khiển chỉ chấp nhận một số chữ cái, loại bỏ các chữ còn lại. Ví dụ, bộ điều khiển trên máy tiện CNC loại bỏ chữ Y do trục Y chỉ dành cho các nguyên công phay (máy phay và trung tâm gia công). Các chữ hoa thường được dùng trong lập trình CNC, nhưng một số bộ điều khiển chấp nhận các chữ thường với cùng ý nghĩa như chữ hoa.

Nếu nghi ngờ, bạn hãy chỉ dùng các chữ hoa!

Ký hiệu

Một số ký hiệu được dùng trong lập trình, bổ sung cho chữ số và chữ cái. Các ký hiệu phổ biến nhất là *dấu chấm thập phân, dấu trừ, dấu phần trăm, dấu ngoặc*,... tùy theo các tùy chọn điều khiển.

Từ ngữ

Từ ngữ chương trình là sự kết hợp các ký tự số-chữ, tạo ra lệnh đơn cho hệ thống điều khiển. Nói chung mỗi từ đều bắt đầu với chữ hoa, tiếp theo là số biểu thị mã chương trình hoặc giá trị

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

thực. Các từ ngữ phổ biến thường hiển thị là *khối, các trục, tương ấn đạo, các lệnh chuẩn bị, các chức năng chung*, và nhiều định nghĩa khác.

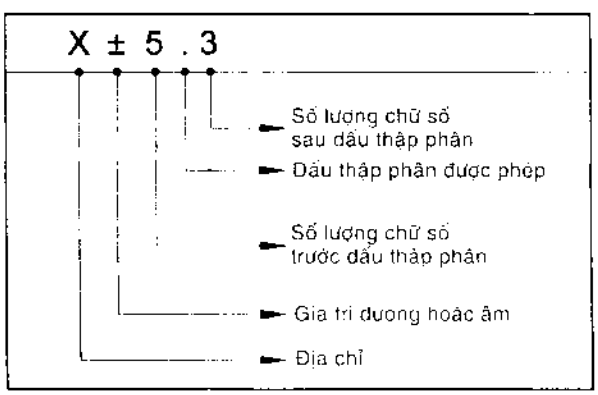
Khối

Từ ngữ được dùng làm lệnh *đơn* cho hệ thống CNC, khối được dùng cho nhóm *nhiều* lệnh (khối lệnh). Chương trình được nhập vào hệ thống điều khiển gồm các dòng lệnh riêng rẽ, sắp xếp theo thứ tự logic. Mỗi dòng được gọi là *khối chuỗi thứ tự* hoặc đơn giản là *khối* (block) – gồm một hoặc nhiều từ ngữ (word) và mỗi từ ngữ gồm *hai hoặc nhiều ký tự*.

Trong hệ thống điều khiển, từng khối phải được tách riêng rẽ. Để tách các khối trong chế độ MDI *(Manual Data Input, nhập dữ liệu bằng tay)* ở bộ điều khiển, mỗi khối kết thúc bằng mã (ký hiệu) *End Of Block* (kết thúc khối). Mã này được viết tắt là EOB trên bảng điều khiển. Khi chuẩn bị chương trình trên máy tính, phím Enter trên bàn phím sẽ kết thúc khối. Khi viết chương trình trên giấy, mỗi khối chương trình chỉ được chiếm một dòng trên giấy. Mỗi khối chương trình chứa dãy các lệnh đơn được thực thi cùng với nhau.

Chương trình

Cấu trúc chương trình thay đổi tùy theo bộ điều khiển nhưng sự tiếp cận logic không thay đổi giữa các bộ điều khiển. Chương trình CNC thường bắt đầu với *số chương trình* hoặc ký hiệu tương tự, tiếp theo là các khối lệnh theo thứ tự logic. Chương trình kết thúc với *stop code* hoặc ký hiệu chấm dứt chương trình, chẳng hạn *dấu phần trăm - %*. Tài liệu bên trong và các thông tin báo cho người vận hành có thể được đặt ở các vị trí đặc biệt trong chương trình. Định dạng chương trình liên tục phát triển theo thời gian, hiện nay đang sử dụng nhiều loại định dạng.



Hình 6.2. *Giải thích định dạng địa chỉ từ ngữ – định dạng trục X theo hệ mét*

ĐỊNH DẠNG CHƯƠNG TRÌNH

Từ những ngày đầu của điều khiển số, đã sử dụng đến ba định dạng, được liệt kê theo thứ tự xuất hiện:

- Tab Sequential Format
  chỉ dùng cho NC không có điểm thập phân
- Fixed Format
  chỉ dùng cho NC không có điểm thập phân
- Word Address Format
  dùng cho NC và CNC, có điểm thập phân

Chỉ những hệ thống điều khiển *thời kỳ đầu* mới sử dụng *tab sequential format* hoặc *fixed format*. Cả hai đều đã lạc hậu và không còn được sử dụng từ đầu thập kỷ 1970. Chúng được thay bằng *Word Address Format* (định dạng địa chỉ từ ngữ) thuận tiện và hiệu quả hơn.

WORD ADDRESS FORMAT

Định dạng *word address format* (địa chỉ từ ngữ) dựa trên sự kết hợp một chữ cái với một hoặc nhiều chữ số (Hình 6.1).

Trong một số ứng dụng, sự kết hợp này có thể được bổ sung bằng ký hiệu, chẳng hạn dấu trừ hoặc dấu chấm thập phân. Mỗi chữ cái, chữ số hoặc ký hiệu biểu thị một ký tự trong chương trình và trong bộ nhớ của hệ điều khiển. Sự sắp xếp đặc thù số – chữ tạo ra một từ ngữ, trong đó chữ cái là địa chỉ, tiếp sau là dữ liệu số có hoặc không có ký hiệu. *Địa chỉ* từ ngữ quy chiếu thanh ghi chuyên biệt trong bộ nhớ của hệ điều khiển. Một số từ ngữ thông dụng bao gồm:

```
G01  M30  D25  X5.75 N105   H01  Y0 S2500
Z-5.14  F12.0 T0505 T05    /M01 B180.0
```

Địa chỉ chữ cái trong block (khối) xác định ý nghĩa của từ ngữ và luôn luôn được viết trước. Ví dụ X5.75 là đúng, 5.75X là sai. Không được phép có khoảng trắng (ký tự trắng) trong từ ngữ, chỉ có thể đặt khoảng trắng trước từ ngữ, ý nghĩa trước chữ cái.

Dữ liệu biểu thị sự gắn số cho từ ngữ. Giá trị này thay đổi khá rộng và tùy thuộc vào địa chỉ đứng trước. Dữ liệu có thể là số chuỗi thứ tự N, số đăng ký bù D hoặc H, từ ngữ tọa độ X, Y hoặc Z, hàm tốc độ cắt F, hàm trục chính S, dụng cụ cắt T,...

Một *từ ngữ* bất kỳ là chuỗi các ký tự (ít nhất hai ký tự) định nghĩa lệnh đơn cho bộ điều khiển và máy. Các ví dụ nêu trên về từ ngữ có ý nghĩa như sau trong lập trình CNC:

G01	*Lệnh chuẩn bị*
M30	*Hàm phụ*
D25	*Chọn số bù - phay*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

X5.75	*Từ ngữ tọa độ – giá trị dương*
N105	*Số chuỗi thứ tự (số block)*
H01	*Số bù chiều dài dụng cụ cắt*
Y0	*Từ ngữ tọa độ – giá trị zero*
S2500	*Hàm tốc độ trục chính*
Z-5.14	*Từ ngữ tọa độ – giá trị âm*
F12.0	*Hàm tốc độ cắt*
T0505	*Hàm dụng cụ cắt – tiện*
T05	*Hàm dụng cụ cắt – phay*
/M01	*Hàm phụ với dấu (/) ký hiệu bỏ qua khối (block)*
B180.0	*Hàm bàn phân độ*

Các từ ngữ riêng rẽ là các lệnh được xếp nhóm với nhau để tạo thành các chuỗi thứ tự mã lập trình. Mỗi chuỗi thứ tự sẽ xử lý dãy các lệnh đồng thời tạo thành một đơn vị được gọi là *sequence block* (khối chuỗi thứ tự) hoặc đơn giản là *block*. Dãy các block sắp xếp theo thứ tự logic được yêu cầu để gia công *hoàn tất* chi tiết hoặc *hoàn tất* nguyên công là chương trình (gia công) chi tiết, còn được gọi là *chương trình CNC*.

Khối lệnh kế tiếp biểu thị chuyển động dao nhanh đến vị trí tuyệt đối X13.0Y4.6 với chất làm nguội được kích hoạt:

```
N25 G90 G00 X13.0 Y4.6 M08
```

Trong đó:

N25	Chuỗi thứ tự hoặc số khối (block)
G90	Chế độ tuyệt đối
G00	Chế độ chuyển động nhanh
X13.0Y4.6	Vị trí tọa độ
M08	Hàm bôi trơn ON

Bộ điều khiển coi một block là đơn vị hoàn chỉnh, *không xử lý nửa chừng*. Hầu hết các bộ điều khiển đều cho phép thứ tự từ ngữ ngẫu nhiên trong block, nếu chỉ số block được chuyên biệt trước.

GIẢI THÍCH ĐỊNH DẠNG

Từng từ ngữ chỉ có thể viết theo cách thức chuyên biệt. Số lượng chữ số được phép trong từ ngữ, tùy thuộc địa chỉ và số lượng tối đa các chữ số thập phân, do nhà chế tạo bộ điều khiển quy định. Không phải mọi chữ cái đều được sử dụng. Chỉ các chữ cái với ý nghĩa được gán trước mới có thể lập trình, ngoại trừ các ghi chú. Các ký hiệu chỉ có thể được dùng trong một số từ ngữ, và vị trí của chúng trong từ ngữ là cố định. Một số ký hiệu chỉ được dùng trong các macros riêng biệt. Các giới hạn điều khiển là rất quan trọng. Các ký hiệu bổ sung cho *chữ số* và *chữ cái*, cung cấp cho chúng ý nghĩa rộng hơn. Các ký hiệu lập trình phổ biến gồm dấu trừ, dấu chấm thập phân, dấu phần trăm và vài ký hiệu khác. Tất cả các ký hiệu được liệt kê trong Bảng 6.3.

Dạng rút gọn

Các nhà chế tạo bộ điều khiển thường chuyên biệt định dạng nhập ở dạng viết tắt hoặc rút gọn (Hình 6.2).

Sự mô tả định dạng đầy đủ từng ý nghĩa là rất dài và không cần thiết. Bạn hãy xem xét sự mô tả đầy đủ (không viết tắt) đối với địa chỉ X, từ ngữ tọa độ được dùng trong hệ mét (Hình 6.2)

> Địa chỉ X chấp nhận dữ liệu dương hoặc âm với tối đa năm chữ số trước dấu thập phân và ba chữ số sau dấu thập phân, dấu chấm thập phân là được phép.

Nếu không có dấu thập phân trong định dạng có nghĩa là không được sử dụng dấu này, nếu không có dấu cộng (+) nghĩa là giá trị địa chỉ không thể là số âm, *không có dấu được ngầm hiểu là giá trị dương*. Các ví dụ về định dạng với ý nghĩa kèm theo được nêu dưới đây.

G2	Tối đa hai chữ số, không có dấu hoặc chấm thập phân
N5	Tối đa năm chữ số, không có dấu hoặc chấm thập phân
F5	Tối đa năm chữ số, không có dấu hoặc chấm thập phân
F3.2	Tối đa năm chữ số, tối đa ba chữ số trước dấu thập phân, tối đa hai chữ số sau dấu thập phân, được phép có dấu thập phân, không sử dụng dấu.

Bạn cần cẩn thận khi đánh giá dạng rút gọn. Hiện chưa có các tiêu chuẩn công nghiệp và không phải mọi nhà chế tạo bộ điều khiển đều sử dụng các phương pháp như nhau. Do đó ý nghĩa của các dạng rút gọn có thể rất khác nhau. Danh sách các địa chỉ, ghi định dạng và giải thích được nêu trong các Bảng 6.1 và 6.2 dựa trên hệ điều khiển Fanuc.

Định dạng hệ thống phay

Ý nghĩa của địa chỉ thay đổi với nhiều địa chỉ, tùy theo các đơn vị nhập. Bảng 6.1 trình bày định dạng theo hệ Anh (định dạng hệ mét được ghi trong dấu ngoặc, nếu có thể áp dụng). Bảng này ghi định dạng của hệ thống phay, cột thứ nhất là địa chỉ, thứ hai là ví dụ về ghi định dạng và cột thứ ba giải thích về công dụng.

Bảng 6.1. *Định dạng hệ thống phay*

Địa chỉ	Định dạng	Giải thích
A	A+5.3	Trục quay hoặc phân độ – đơn vị là độ – dùng cho trục X
B	B+5.3	Trục quay hoặc phân độ – đơn vị là độ – dùng cho trục Y

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Địa chỉ	Định dạng	Giải thích
D	D2	Chỉ số bù bán kính dao cắt (đôi khi sử dụng địa chỉ H)
F	F5.3	Hàm lượng ăn dao – có thể thay đổi
G	G2	Các lệnh chuẩn bị
H	H3	Chỉ số bù (vị trí dao và/hoặc bù chiều dài dao)
I	I+4.4 (I+5.3)	Chỉnh sửa tâm cung đối với trục X. Lượng dịch chuyển trong các chu kỳ cố định (X). Chọn các vector góc cho trục X (kiểu điều khiển cũ).
J	J+4.4 (J+5.3)	Chỉnh sửa tâm cung đối với trục Y. Lượng dịch chuyển trong các chu kỳ cố định (Y). Chọn vector góc cho trục Y (kiểu điều khiển cũ)
K	K+4.4 (K+5.3)	Chỉnh sửa tâm cung đối với trục Z
L	L4	Đếm sự lặp lại chu kỳ cố định Đếm sự lặp lại chương trình con
M	M2	Hàm phụ
N	N5	Số block hoặc số chuỗi thứ tự
O	O4	Số chương trình (EIA) hoặc (:4 đối với ISO)
P	P4	Gọi số chương trình con Gọi số macro riêng
	P3	Số bù chi tiết – sử dụng với G10
	P53	Thời gian tạm dừng tính theo mili giây (ms)
	P5	Số block trong chương trình khi dùng với M99
Q	Q4.4 (Q5.3)	Chiều sâu khoét trong các chu kỳ cố định G73 và G83
	Q+4.4 (Q+5.3)	Lượng xê dịch trong các chu kỳ cố định G76 và G87
R	R+4.4 (R+5.3)	Điểm rút dao trong chu kỳ cố định. Gán bán kính cung
S	S5	Tốc độ trục chính v/ph
T	T4	Hàm dụng cụ cắt
X	X+4.4 (X+5.3)	Gán giá trị tọa độ trục X
	X5.3	Hàm tạm dừng với G04
Y	Y+4.4 (Y+5.3)	Gán giá trị tọa độ trục Y
Z	Z+4.4 (Z+5.3)	Gán giá trị tọa độ trục Z

Định dạng hệ thống tiện

Định dạng hệ thống tiện, tương tự hệ thống phay (Bảng 6.1). Số lượng các định nghĩa là như nhau và được gộp lại chỉ để tăng tính thuận tiện. Giải thích theo định dạng hệ Anh, hệ mét được ghi trong dấu ngoặc, nếu áp dụng được cho địa chỉ đó.

Bảng 6.2. *Định dạng hệ thống tiện*

Địa chỉ	Định dạng	Giải thích
A	A3	Góc ren đối với G76 Góc đối với nhập bản vẽ trực tiếp
C	C+4.4 (C+5.3)	Vạt góc đối với nhập bản vẽ trực tiếp
D	D4	Số lượng các khoảng chia trong G73
	D44 (D53)	Chiều sâu cắt trong G71 và G72 Lượng giảm trong G74 và G75 Chiều sâu ren thứ nhất trong G76
E	E2.6	Tốc độ cắt chính xác khi tiện ren
F	F2.6	Hàm tốc độ cắt, có thể thay đổi
G	G2	Các lệnh chuẩn bị
I	I+4.4 (I+5.3)	Chỉnh sửa tâm cung đối với trục Chiều cao côn theo X dùng cho các chu kỳ Lượng giảm trục X trong G73 Chiều vạt góc Lượng chuyển động theo trục X trong G74
K	K+4.4 (I+5.3)	Chỉnh sửa tâm cung đối với trục Chiều cao côn theo Z dùng cho các chu kỳ Lượng giảm trục Z trong G73 Chiều vạt góc Lượng chuyển động theo trục Z trong G75 Chiều sâu ren trong G76
L	L4	Đếm sự lặp lại chương trình con
M	M2	Hàm phụ
N	N5	Số block hoặc số chuỗi thứ tự
O	O4	Số chương trình (EIA) hoặc (: 4 đối với ISO)
P	P4	Gọi số chương trình con Gọi số macro riêng Số dịch chuyển với G10
	P53	Thời gian tạm dừng tính theo mili giây (ms)
Q	Q5	Số kết thúc khối trong G71 và G72
R	R+4.4 (R+5.3)	Gán bán kính cung Bán kính cong đối với các góc
S	S5	Tốc độ trục chính v/ph hoặc ft/min
T	T4	Hàm dụng cụ cắt
U	U+4.4 (U+5.3)	Giá trị số gia theo trục X Khoảng hở phôi theo trục X
	U5.3	Hàm tạm dừng với G04
W	W+4.4 (W+5.3)	Giá trị số gia theo trục Z Khoảng hở phôi theo trục Z
X	X+4.4 (X+5.3)	Giá trị tuyệt đối theo trục X
	X5.3	Hàm tạm dừng với G04
Z	Z+4.4 (Z+5.3)	Giá trị tuyệt đối theo trục Z
	P5	Số block trong chương trình con Số khởi đầu block trong G71 và G72

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Địa chỉ nhiều từ ngữ

Đặc điểm dễ nhận thấy trong cả hai bảng nêu trên là một số địa chỉ có các ý nghĩa khác nhau. Đây là tính năng cần thiết của định dạng địa chỉ từ ngữ. Thực tế chỉ có 26 chữ cái trong tiếng Anh, nhưng vẫn nhiều hơn số lượng các lệnh và các hàm. Khi bổ sung thêm các tính năng điều khiển mới, có thể cần thêm các biến thể. Một số địa chỉ có ý nghĩa đã được thiết lập (ví dụ, X, Y, Z là các từ ngữ tọa độ), tăng thêm ý nghĩa cho chúng và sẽ gây nhầm lẫn. Nhiều chữ cái, được sử dụng tương đối ít, do đó có thể gán cho chúng nhiều ý nghĩa (ví dụ, các chữ I, J, K, P). Ngoài ra, ý nghĩa của các địa chỉ có thể khác nhau giữa hệ thống tiện và hệ thống phay.

Hệ thống điều khiển có các phương tiện chấp nhận một từ cụ thể với ý nghĩa được xác định chính xác trong chương trình. Trong hầu hết các trường hợp, lệnh chuẩn bị G sẽ xác định ý nghĩa vào các thời điểm sẽ là hàm M hoặc xác lập của các tham số hệ thống.

KÝ HIỆU TRONG LẬP TRÌNH

Ngoài các ký hiệu cơ bản, Fanuc còn có thể chấp nhận các ký hiệu khác trong nhiều ứng dụng. Bảng 6.3 liệt kê các ký hiệu thông dụng trong các bộ điều khiển Fanuc.

Bảng 6.3. *Các ký hiệu trong bộ điều khiển Fanuc*

Ký hiệu	Tên gọi	Giải thích
.	Dấu thập phân	Phần thập phân của một số
+	Dấu cộng	Giá trị dương hoặc dấu *cộng* trong các macro của Fanuc
–	Dấu trừ	Giá trị âm hoặc dấu *trừ* trong các macro của Fanuc
*	Dấu nhân	Dấu nhân trong các macro của Fanuc
/	Dấu nghiêng (phải)	Ký hiệu hàm bỏ qua block, dấu *chia* trong các macro của Fanuc
( )	Dấu ngoặc	Chú thích trong chương trình
%	Dấu phần trăm	Mã dừng (kết thúc tập tin chương trình)
:	Dấu hai chấm	Gán số chương trình
,	Dấu phẩy	Chỉ sử dụng trong các chú thích
[ ]	Dấu ngoặc vuông	Giải thích trong các marco của Fanuc
;	Dấu chấm phẩy	Ký hiệu kết thúc khối (End of block) không lập trình (chỉ trên màn hình hiển thị)
#	Dấu thăng	Định nghĩa biến hoặc gọi trong các marco của Fanuc
=	Dấu bằng	Sự bằng nhau trong các marco của Fanuc

Bảng 6.3 liệt kê cả ký hiệu đặc biệt và ký hiệu tiêu chuẩn. Các ký hiệu đặc biệt chỉ được dùng với các tính năng tùy chọn, chúng không được phép sử dụng trong lập trình tiêu chuẩn, do có thể gây ra lỗi. Các ký hiệu tiêu chuẩn là loại trên bàn phím máy tính. Các tổ hợp *Ctrl*, *Shift* và *Alt* đều không được phép.

Dấu cộng và dấu trừ

Ký hiệu phổ biến nhất trong lập trình CNC là dấu đại số: cộng hoặc trừ. Dữ liệu bất kỳ trong lệnh chuyển động có thể là dương hoặc âm. Để thuận tiện, hầu như mọi điều khiển đều cho phép bỏ qua dấu cộng đối với mọi giá trị dương. Tính năng này đôi khi được gọi là nghiêng về dương trong hệ thống điều khiển. Nghiêng về dương là thuật ngữ biểu thị giá trị được coi là dương nếu không có dấu được lập trình trong từ ngữ:

`X+125.0` *đồng nhất với* `X125.0`

Dấu trừ phải được lập trình. Nếu bỏ sót dấu trừ, số đó sẽ trở thành dương, dẫn đến kết quả sai (ví dụ vị trí dụng cụ cắt):

`X-125.0` *giá trị âm*
`X125.0` *giá trị dương (dấu + được bỏ qua)*
`X+125.0` *giá trị dương*

Các ký hiệu bổ sung cho chữ cái và chữ số, và là phần tích hợp của cấu trúc chương trình.

TIÊU ĐỀ CHƯƠNG TRÌNH

Các chú thích và ghi chú có thể được đưa vào chương trình, nhưng phải đặt trong dấu ngoặc. Dạng tài liệu bên trong này rất hữu ích cho cả nhà lập trình và người vận hành CNC. Chuỗi các ghi chú ở phần đầu chương trình được gọi là *tiêu đề chương trình*, ghi rõ các tính năng của chương trình. Ví dụ dưới đây nêu rõ nội dung của tiêu đề chương trình (gồm tên tập tin, ngày tháng chỉnh sửa, tên nhà lập trình, máy, bộ điều khiển, đơn vị đo, chỉ số gia công, các nguyên công, vật liệu, kích cỡ phôi, chuẩn chương trình, ...)

```
( ____________________________________ )
(FILE NAME . . . . . 01234.NC)
(LAST VERSION DATE . . . . . 07-DEC-01)
(LAST VERSION TIME . . . . . 19:43)
(PROGRAMMER . . . . . PETER SMID)
(MACHINE . . . . . OKK - VMC)
(CONTROL . . . . . FANUC 15M)
(UNITS . . . . . INCHES)
(JOB NUMBER . . . . . 4321)
(OPERATION . . . . . DRILL-BORE-TAP)
(STOCK MATERIAL . . . . . H.R.S. PLATE)
(MATERIAL SIZE . . . . . 8 X 6 X 2)
(PROGRAM ZERO . . . . . XO - LEFT EDGE)
(                          YO - BOLT EDGE)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
(                                   Z0  TOP FACE )
(STATUS . . . . . . . . . . . . . . . NOT VERIFIED)
(_____ _____ _____ _____ _____ ____)
```

Từng dụng cụ cắt đều được xác định bên trong chương trình

```
(*** T03 - 1/4-20 PLUG TAP***)
```

Các chú thích và ghi chú dành cho người vận hành có thể được bổ sung vào chương trình theo yêu cầu.

CẤU TRÚC CHƯƠNG TRÌNH

Mặc dầu có lẽ hơi sớm khi đưa ra chương trình hoàn chỉnh nhưng bạn sẽ hình dung vấn đề rõ hơn nếu xem xét cấu trúc chương trình điển hình. Sự triển khai cấu trúc chương trình cơ bản là cốt lõi của vấn đề, cấu trúc đó sẽ được sử dụng thường xuyên. Mỗi chương trình dưới đây đều có giải thích kèm theo.

Ghi chú: Các block (khối) chương trình chỉ sử dụng số block mẫu. Các block trong dấu ngoặc là không cần thiết đối với các chu kỳ cố định. Chẳng hạn, khối N88 là vị trí hiện hành của các trục X và Y. Nếu chưa biết vị trí tuyệt đối, bạn hãy đổi khối này sang phiên bản số gia.

```
N88 G91 G28 X0 Y0
```

Nếu có dụng cụ cắt cần lặp lại, bạn phải bảo đảm không gộp block thay dao cho dụng cụ hiện hành. Nhiều hệ thống CNC sẽ đưa ra cảnh báo nếu lệnh thay dao không thể tìm được dao đó trong ổ dao. Trong chương trình mẫu nêu trên, các block dao cắt lặp lại là N5, N38, và N67.

Ví dụ cấu trúc chương trình nêu trên dùng cho máy với chế độ chọn dao ngẫu nhiên và hệ thống điều khiển thông dụng, có thể có vài thay đổi nhỏ. Bạn hãy nghiên cứu chuỗi kết cấu của chương trình thay vì nội dung cụ thể, hãy chú ý tính lặp lại của các khối đối với từng dụng cụ cắt và sự bổ sung dòng trắng (khối còn trống) giữa các dụng cụ cắt riêng rẽ để định hướng trong chương trình dễ dàng hơn.

```
00701  (ID MAX 15 CHARS)
(SAMPLE PROGRAM STRUCTURE)
(PETER SMID - 07-DEC-01)

N1 G20
N2 G17 G40 G80 G49
N3 T01
N4 M06
N5 G90 G54 G00 X.. Y.. S.. M03 T02
N6 G43 Z2.0 H01 M08
(N7 G01 Z-.. F..)
 (----CUTTING MOTIONS WITH TOOL T0 ----)
...
N33 G00 G80 Z2.0 M09
N34 G28 Z2.0 M05
N35 M01

N36 T02
N37 M06
N38 G90 G54 G00 X.. Y.. S.. M03 T03
N39 G43 Z2.0 H02 M08
 (N40 G01 Z-.. F..)
 (----CUTTING MOTIONS WITH TOOL T02----)
...
N62 G00 G80 Z2.0 M09
N63 G28 Z2.0 M05
N64 M01

N65 T03
N66 M06
N67 G90 G54 G00 X.. Y.. S.. M03 T01
N68 G43 Z2.0 H03 M08
 ( N69 G01 Z-.. F..)
 (----CUTTING MOTIONS WITH TOOL T03----)
...
N86 G00 G80 Z2.0 M09
N87 G28 Z2.0 M05
N88 G28 X.. Y..
N89 M30
%
```

```
(PROGRAM NUMBER AND ID)
(BRIEF PROGRAM DESCRIPTION)
(PROGRAMMER AND DATE OF LAST REVISION)
(BLANK LINE)
(UNITS SETTING IN A SEPARATE BLOCK)
(INITIAL SETTINGS AND CANCELLATIONS)
(TOOL T01 INTO WAITING POSITION)
(T01 INTO SPINDLE)
(T01 RESTART BLOCK - T02 INTO WAITING POSITION)
(TOOL LG OFFSET - CLEAR ABOVE WORK - COOLANT ON)
(FEED TO Z DEPTH IF NOT A CYCLE)


(CLEAR ABOVE PART - COOLANT OFF)
(HOME IN Z ONLY-SPIDLE OFF)
(OPTIONAL STOP)
(-- BLANK LINE --)
(TOOL T02 INTO WAITING POSITION - CHECK ONLY)
(T02 INTO SPINDLE)
(T02 RESTART BLOCK - T03 INTO WAITING POSITION)
(TOOL LG OFFSET - CLEAR ABOVE WORK - COOLANT ON)
(FEED TO Z DEPTH IF NOT A CYCLE)


(CLEAR ABOVE PART - COOLANT OFF)
(HOME IN Z ONLY - SPINDLE OFF)
(OPTIONAL STOP)
(-- BLANK LINE --)
(TOOL T03 INTO WAITING POSITION - CHECK ONLY)
(T03 INTO SPINDLE)
(T03 RESTART BLOCK - T01 INTO WAITING POSITION)
(TOOL LG OFFSET - CLEAR ABOVE WORK - COOLANT ON)
(FEED TO Z DEPTH IF NOT A CYCLE)


(CLEAR ABOVE PART - COOLANT OFF)
(HOME IN Z ONLY - SPINDLE OFF)
(HOME IN XY ONLY)
(END OF PROGRAM)
(STOP CODE - END OF FILE TRANSFER)
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

7 CÁC LỆNH CHUẨN BỊ

Địa chỉ chương trình G xác định *lệnh chuẩn bị*, thường được gọi là *mã G*. Địa chỉ này có một và chỉ một nội dung, đó là *xác lập trước* hoặc *chuẩn bị* cho hệ điều khiển về điều kiện mong muốn xác định, chế độ xác định hoặc trạng thái vận hành. Ví dụ, địa chỉ G00 chuẩn bị chế độ chuyển động nhanh cho máy công cụ, địa chỉ G81 xác lập trước cho chu kỳ phay, ... Thuật ngữ *lệnh chuẩn bị* nêu rõ ý nghĩa của nhóm lệnh này, mã G chuẩn bị cho hệ điều khiển chấp nhận các lập trình *tiếp sau* mã G theo cách thức chuyên biệt.

Ý NGHĨA VÀ CÔNG DỤNG

Ví dụ về một block sẽ minh họa công dụng của các lệnh chuẩn bị trong mục nhập chương trình:

```
N7 X13.0 Y10.0
```

Chỉ xem qua block này bạn cũng có thể thấy các tọa độ X13.Y10.0 liên quan với *vị trí cuối* của dụng cụ cắt khi khối N7 được thực thi (do bộ điều khiển xử lý). Block không nêu rõ các tọa độ được tính theo chế độ tuyệt đối hay chế độ số gia, cũng không cho biết các giá trị tính theo hệ mét hay hệ Anh, cũng không nên chuyển động đến vị trí đích đã chuyển biệt là chuyển động nhanh hay chuyển động tuyến tính. Nếu quan sát block này bạn không thể thiết lập ý nghĩa của nội dung block, cũng không biết đây là hệ điều khiển nào. Thông tin được cung cấp trong block là *không đầy đủ*, do đó chưa thể sử dụng, cần phải bổ sung thêm các lệnh cho block.

Ví dụ, để làm cho block N7 gán cho dụng cụ cắt trong chế độ chuyển động nhanh sử dụng các kích thước tuyệt đối, *tất cả* các lệnh đó phải được chuyên biệt *trước* hoặc *trong* block.

➲ **Ví dụ A**

```
N7 G90 G00 X13.0 Y10.0
```

➲ **Ví dụ B**

```
N3 G90
N4 ...
N5 ...
N6 ...
N7 G00 X13.0 Y10.0
```

➲ **Ví dụ C**

```
N3 G90 G00
N4 ...
N5 ...
N6 ...
N7 X13.0 Y10.0
```

➲ **Ví dụ D**

```
N2 G90
N3 G00
N4 ...
N5 ...
N6 ...
N7 X13.0 Y 10.0
```

Cả bốn ví dụ này đều có cùng kết quả gia công với điều kiện không thay đổi chế độ mã G bất kỳ giữa các block N4 và N6 trong các ví dụ B, C và D

Một mã G trong nhóm chế độ thay cho mã G khác của cùng nhóm đó

Các mã G chế độ và không chế độ sẽ được đề cập một cách ngắn gọn. Mỗi hệ điều khiển đều có danh sách riêng về các mã G khả dụng. Nhiều mã G là chung và có thể hiện diện trong hầu như mọi hệ điều khiển, số khác là đặc thù riêng của hệ điều khiển cụ thể, kể cả máy công cụ. Do bản chất của các ứng dụng gia công, danh sách mã G là khác nhau giữa hệ thống phay và hệ thống tiện. Điều này cũng đúng cho các kiểu máy khác nhau. Mỗi nhóm mã G cần được duy trì riêng rẽ.

Cần đọc kỹ tài liệu về máy để biết các mã G khả dụng.

ỨNG DỤNG TRÊN HỆ THỐNG PHAY

Bảng 7.1 trình bày danh sách chi tiết các lệnh chuẩn bị thông dụng nhất khi lập trình các máy phay CNC và trung tâm gia công CNC. Các mã G được liệt kê có thể không dùng được đối với hệ thống điều khiển hoặc máy cụ thể, do đó bạn cần xem kỹ sổ tay hướng dẫn sử dụng máy và bộ điều khiển để biết các mã G khả dụng. Một số mã G được liệt kê là tùy chọn đặc biệt phải khả dụng trên máy và trong hệ thống điều khiển.

Bảng 7.1. Mã G trên hệ thống phay

Mã G	Công dụng
G00	Định vị nhanh
G01	Nội suy tuyến tính
G02	Nội suy vòng thuận chiều
G03	Nội suy vòng ngược chiều
G04	Tạm dừng (khi là block riêng rẽ)
G09	Kiểm tra sự dừng chính xác – chỉ một block

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Mã G	Công dụng
G10	Nhập dữ liệu lập trình (Data Setting)
G11	Xóa chế độ Data Setting
G15	Xóa lệnh tọa độ cực
G16	Lệnh tọa độ cực
G17	Gán mặt phẳng XY
G18	Gán mặt phẳng ZX
G19	Gán mặt phẳng YZ
G20	Nhập đơn vị Anh
G21	Nhập đơn vị mét
G22	Kiểm tra hành trình đã lưu ON
G23	Kiểm tra hành trình đã lưu OFF
G25	Tìm dao động tốc độ trục chính ON
G26	Tìm dao động tốc độ trục chính OFF
G27	Kiểm tra vị trí zero trên máy
G28	Trả về zero trên máy (điểm quy chiếu 1)
G29	Trả về từ zero trên máy
G30	Trả về zero trên máy (điểm quy chiếu 2)
G31	Hàm Skip (bỏ qua)
G40	Xóa bù bán kính dao cắt
G41	Bù bán kính dao cắt – trái
G42	Bù bán kính dao cắt – phải
G43	Bù chiều dài dao cắt – dương
G44	Bù chiều dài dao cắt – âm
G45	Bù vị trí – tăng một
G46	Bù vị trí – giảm một
G47	Bù vị trí – tăng đôi
G48	Bù vị trí – giảm đôi
G49	Xóa bù chiều dài dao cắt.
G50	Xóa hàm lập tỷ lệ
G51	Hàm lập tỷ lệ
G52	Xác lập hệ tọa độ cục bộ
G53	Hệ tọa độ máy
G54	Bù tọa độ chi tiết 1
G55	Bù tọa độ chi tiết 2
G56	Bù tọa độ chi tiết 3
G57	Bù tọa độ chi tiết 4
G58	Bù tọa độ chi tiết 5
G59	Bù tọa độ chi tiết 6
G60	Định vị một chiều
G61	Chế độ dừng chính xác
G62	Chế độ override góc tự động
G63	Chế độ tarô ren
G64	Chế độ cắt
G65	Gọi marco riêng
G66	Gọi chế độ marco riêng
G67	Xóa gọi chế độ marco riêng
G68	Quay hệ tọa độ
G69	Xóa quay hệ tọa độ
G73	Chu kỳ khoan tốc độ cao (lỗ sâu)
G74	Chu kỳ cắt ren trái
G76	Chu kỳ doa tinh
G80	Xóa chu kỳ cố định
G81	Chu kỳ khoan
G82	Chu kỳ khoan – điểm (lấy dấu tâm)
G83	Chu kỳ khoan bậc (chu kỳ khoan lỗ sâu)
G84	Chu kỳ cắt ren phải
G85	Chu kỳ doa
G86	Chu kỳ doa
G87	Chu kỳ doa ngược
G88	Chu kỳ doa
G89	Chu kỳ doa
G90	Chế độ kích thước tuyệt đối
G91	Chế độ kích thước số gia (tương đối)
G92	Bộ ghi vị trí dao cắt.
G98	Trở lại mức ban đầu trong chu kỳ cố định
G99	Trở lại mức R trong chu kỳ cố định

Nếu có khác biệt giữa các mã G được liệt kê trong Bảng 7.1 và trong sổ tay hướng dẫn của hệ thống điều khiển cụ thể, cần sử dụng các mã G do nhà chế tạo bộ điều khiển quy định.

ỨNG DỤNG TRÊN HỆ THỐNG TIỆN

Các điều khiển máy tiện của công ty Fanuc sử dụng ba kiểu nhóm mã G-A,B. và C. Kiểu A là phổ biến nhất. Trong sách này mọi ví dụ và giải thích đều thuộc nhóm kiểu A, kể cả Bảng 7.2. Mỗi lần chỉ được sử dụng một kiểu. Các kiểu A và B có thể được xác lập bằng tham số hệ thống điều khiển, còn kiểu C là tùy chọn. Nói chung, hầu hết các mã G là như nhau, chỉ có vài khác biệt trong kiểu A và kiểu B. Chi tiết về các nhóm mã G được trình bày ở cuối chương này.

Bảng 7.2. Mã G trên hệ thống tiện

Mã G	Công dụng
G00	Định vị nhanh
G01	Nội suy tuyến tính
G02	Nội suy vòng thuận chiều
G03	Nội suy vòng ngược chiều
G04	Tạm dừng (khi là block riêng rẽ)
G09	Kiểm tra sự dừng chính xác – chỉ một block
G10	Nhập dữ liệu lập trình (Data Setting)
G11	Xóa chế độ Data Setting
G20	Nhập đơn vị Anh
G21	Nhập đơn vị mét
G22	Kiểm tra hành trình đã lưu ON
G23	Kiểm tra hành trình đã lưu OFF
G25	Tìm sự thay đổi tốc độ trục chính ON
G26	Tìm sự thay đổi tốc độ trục chính OFF

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Mã G	Công dụng
G27	Kiểm tra vị trí zero trên máy
G28	Trả về zero trên máy (điểm quy chiếu 1)
G29	Trả về từ zero trên máy
G30	Trả về zero trên máy (điểm quy chiếu 2)
G31	Hàm Skip (bỏ qua)
G32	Cắt ren – tiến liên tục
G35	Cắt ren tròn thuận chiều
G36	Cắt ren tròn ngược chiều
G40	Xóa bù bán kính mũi dao
G41	Bù bán kính mũi dao trái
G42	Bù bán kính mũi dao phải
G50	Bộ ghi vị trí dao/xác lập trước v/ph cực đại
G52	Xác lập hệ tọa độ cục bộ
G53	Xác lập hệ tọa độ máy
G54	Bù tọa độ chi tiết 1
G55	Bù tọa độ chi tiết 2
G56	Bù tọa độ chi tiết 3
G57	Bù tọa độ chi tiết 4
G58	Bù tọa độ chi tiết 5
G59	Bù tọa độ chi tiết 6
G61	Chế độ dừng chính xác
G62	Chế độ override góc tự động
G64	Chế độ cắt
G65	Gọi marco riêng
G66	Gọi chế độ marco riêng
G67	Xóa gọi chế độ marco riêng
G68	Hình ảnh (đối xứng) gương đối với các ổ dao kép
G69	Xóa ảnh (đối xứng) gương đối với các ổ dao kép
G70	Chu kỳ gia công tinh biên dạng
G71	Chu kỳ gia công thô biên dạng – chiều trục Z
G72	Chu kỳ gia công thô biên dạng – chiều trục X
G73	Chu kỳ lặp lại theo mẫu
G74	Chu kỳ khoan
G75	Chu kỳ cắt rãnh
G76	Chu kỳ cắt ren
G90	Chu kỳ cắt A *(kiểu nhóm A)*
G90	Lệnh tuyệt đối *(kiểu nhóm B)*
G91	Lệnh số gia *(kiểu nhóm B)*
G92	Chu kỳ cắt ren *(kiểu nhóm A)*
G92	Bộ ghi vị trí dao *(kiểu nhóm B)*
G94	Chu kỳ cắt B *(kiểu nhóm A)*
G94	Lượng ăn dao theo phút *(kiểu nhóm B)*
G95	Lượng ăn dao theo vòng quay *(kiểu nhóm B)*
G96	Chế độ tốc độ bề mặt không đổi *(CSS-constant – surface speed)*
G97	Nhập v/ph trực tiếp *(xóa chế độ CSS)*
G98	Lượng ăn dao theo phút *(kiểu nhóm A)*
G99	Lượng ăn dao theo vòng quay *(kiểu nhóm A)*

Hầu hết các lệnh chuẩn bị đều được đề cập trong từng ứng dụng riêng rẽ, ví dụ G01 được giải thích trong *Nội suy tuyến tính*, G02 và G03 trong *Nội suy vòng*. Trong phần này, mã G chỉ được trình bày khái quát, bất kể kiểu máy hoặc bộ điều khiển.

MÃ G TRONG BLOCK CHƯƠNG TRÌNH

Khác với các hàm chung, được gọi là *hàm M* và được trình bày trong Chương 8, vài lệnh chuẩn bị có thể được dùng trong một khối, với điều kiện chúng không trái logic với nhau.

```
N25 G90 G00 G54 X6.75 Y10.5
```

Phương pháp viết chương trình thứ hai sử dụng vài block ngắn hơn so với một block:

```
N25 G90
N26 G00
N27 G54
N28 X6.75 Y 10.5
```

Cả hai phương pháp đều như nhau trong khi xử lý liên tục. Tuy nhiên, phương pháp thứ hai, khi được thực thi trong chế độ *block đơn*, mỗi block đều đòi hỏi nhấn phím Cycle Start để khởi động. Phương pháp rút gọn có tính thực tiễn cao hơn, không chỉ do chiều dài còn do liên kết logic giữa các lệnh trong block.

Một số nguyên tắc áp dụng và các khảo sát tổng quát đối với mã G được sử dụng với dữ liệu *khác* trong block, quan trọng nhất trong các nguyên tắc đó là *tính chế độ*.

Tính chế độ của các lệnh G

Ở phần trước, ví dụ C được dùng để minh họa vị trí chung của mã G trong block chương trình.

➲ **Ví dụ C - gốc:**

```
N3 G90 G00
N4 ...
N5 ...
N6 ...
N7 X13.0 Y10.0
```

Nếu thay đổi cấu trúc này và sử dụng dữ liệu thực, năm block có thể như sau:

➲ **Ví dụ: C - chỉnh sửa (khi lập trình):**

```
N3 G90 G00 X5.0 Y3.0
N4 X0
N5 Y20.0
N6 X15.0 Y22.0
N7 X13.0 Y10.0
```

Bạn hãy chú ý lệnh chuyển động nhanh G00 – lệnh này xuất hiện mấy lần trong chương trình? Đúng một lần – trong block N3. Lệnh chế độ tuyệt đối – G90, cũng chỉ xuất hiện một lần. Cả lệnh G00 và G90 đều không lặp lại là do chúng vẫn tiếp tục hoạt động kể từ thời

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

điểm chúng xuất hiện trong chương trình. Thuật ngữ *chế độ* được dùng để biểu thị đặc tính đó.

Đối với lệnh có tính chế độ, lệnh đó vẫn giữ chế độ xác định cho đến khi bị xóa bằng chế độ khác.

Do hầu hết các mã G đều có tính chế độ, không cần lặp lại lệnh đó trong từng block. Với ví dụ C nêu trên, bộ điều khiển sẽ thực hiện sự diễn dịch dưới đây, trong khi thực thi chương trình.

➲ **Ví dụ C - chỉnh sửa (khi thực thi):**

```
N3 G90 G00 X50.0 Y30.0
N4 G90 G00 X0
N5 G90 G00 Y200.0
N6 G90 G00 X150.0 Y220.0
N7 G90 G00 X130.0 Y100.0
```

Chương trình này không ứng dụng thực tiễn khi chuyển từ vị trí này đến vị trí khác với tốc độ nhanh, chỉ nhằm minh họa tính chế độ của các lệnh chuẩn bị. Mục đích của các giá trị chế độ là tránh sự sao chép không cần thiết đối với các chế độ lập trình. Các mã G được dùng thường xuyên, do đó việc viết chúng trong chương trình có thể rất nhàm chán. Thật may mắn, đa số các mã G có thể chỉ áp dụng một lần, với điều kiện chúng có tính chế độ. Trong bảng đặc tính kỹ thuật của hệ điều khiển, các lệnh chuẩn bị được chia thành hai nhóm, chế độ và không chế độ.

Va chạm các lệnh trong block

Công dụng của các lệnh chuẩn bị là chọn giữa hai hoặc nhiều chế độ vận hành. Nếu chọn G00 chuyển động nhanh, đây là lệnh chuyên biệt liên quan đến chuyển động chạy dao. Không thể có chuyển động nhanh và chuyển động cắt hoạt động *cùng một lúc*, không thể có lệnh G00 và G01 đồng thời hoạt động. Kết hợp này gây ra sự *va chạm* trong block. Nếu các mã G va chạm được dùng trong cùng block, mã G *cuối* sẽ được sử dụng.

```
N74 G01 G00 X3.5 Y6.125 F20.0
```

Trong ví dụ này, hai lệnh G01 và G00 va chạm nhau. Do G00 là lệnh *đứng sau* trong block, lệnh đó sẽ hoạt động. Lệnh lượng ăn dao được bỏ qua trong block dưới đây.

```
N74 G00 G01 X3.5 Y6.125 F20.0
```

Ví dụ này ngược với ví dụ trên. Ở đây G00 đứng trước do đó G01 được ưu tiên và chuyển động này sẽ xảy ra khi chuyển động cắt có lượng ăn dao được chuyên biệt theo giá trị 20.0 in/min.

Thứ tự từ ngữ trong block

Các mã G được lập trình ở phần đầu block, sau chỉ số block, nhưng trước các dữ liệu khác.

```
N40 G91 G01 Z-0.625 F8.5
```

Đây là thứ tự truyền thống, dựa trên ý tưởng nếu công dụng của mã G là *chuẩn bị* hoặc *xác lập trước* điều kiện xác định cho hệ điều khiển thì các lệnh chuẩn bị phải được đặt trước. Hỗ trợ luận điểm này là yêu cầu chỉ các mã không va chạm mới được phép trong block đơn. Nói một cách chính xác, khối nêu trên có thể sắp xếp lại như sau:

```
N40 G91 Z-0.625 F8.5 G01
```

Có lẽ hơi bất thường, nhưng hoàn toàn chính xác. Nếu sắp xếp lại mã G trong block như dưới đây sẽ hoàn toàn khác:

```
N40 Z-0.625 F8.5 G01 G91
```

Bạn hãy cảnh giác với các tình huống tương tự ví dụ nêu trên. Điều xảy ra trong trường hợp này là chuyển động cắt G01, lượng ăn dao F và chiều sâu Z sẽ kết hợp với nhau và được thực thi *sử dụng chế độ kích thước hiện hành*. Nếu chế độ hiện hành là tuyệt đối, chuyển động trục Z sẽ được thực thi theo giá trị tuyệt đối, mà không theo giá trị số gia (tương đối). Lý do về ngoại lệ này là Fanuc cho phép phối hợp các giá trị kích thước trong cùng một block. Điều này là rất hữu ích, nếu được dùng một cách cẩn thận. Ứng dụng *đúng* của tính năng này được minh họa trong ví dụ dưới đây:

```
(G20)
N45 G90 G00 G54 X1.0 Y1.0 S1500 M03      (G90)
N46 G43 Z0.1 H02
N47 G01 Z-0.25 F5.0
N48 X2.5 G91 Y1.5          (G90 MIXED WITH G91)
N49 ...
...
```

Các block N45 đến N47 đều trong chế độ tuyệt đối. Trước khi N48 được thực thi, vị trí tuyệt đối của các trục X và Y là 1.0, 1.0. Từ vị trí khởi đầu này, vị trí đích là vị trí tuyệt đối X2.5 kết hợp với chuyển động số gia 1.5 inch dọc theo trục Y. Kết quả vị trí tuyệt đối sẽ là X2.5 Y2.5, tạo ra chuyển động 45^0. Trong trường hợp này, G91 vẫn có hiệu lực đối với tất cả các block kế tiếp, cho đến khi G90 được lập trình. Hầu như chắc chắn, block N48 sẽ được viết theo chế độ tuyệt đối.

```
...
N48 X2.5 Y2.5
...
```

Nói chung, không có lý do để chuyển đổi giữa hai chế độ này, do có thể đưa đến các kết quả không mong muốn. Trong một số trường

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

hợp, kỹ thuật này có thể có ích, chẳng hạn trong các chương trình con.

CHIA NHÓM CÁC LỆNH

Ví dụ về sự va chạm các mã G trong một block đưa đến giải pháp chia nhóm các lệnh. Bạn dễ dàng nhận thấy các lệnh chuyển động, chẳng hạn G00, G01, G02 và G03, không thể cùng tồn tại trong một block. Sự phân biệt này có lẽ không thật rõ đối với các lệnh chuẩn bị khác. Ví dụ, lệnh bù chiều dài dao cắt G43 có thể được lập trình trong cùng một block với lệnh bù bán kính dao cắt G41 hoặc G42? Câu trả lời là được, nhưng bạn hãy xem xét lý do.

Hệ điều khiển Fanuc chấp nhận các lệnh chuẩn bị bằng cách xếp chúng vào các *nhóm* riêng. Mỗi nhóm, được gọi là nhóm mã G, có số hai chữ số tùy ý do Fanuc quy định. Nguyên tắc về sự cùng tồn tại của các mã G trong một block rất đơn giản. Nếu hai hoặc nhiều mã G từ cùng một nhóm được xếp vào một block, chúng sẽ va chạm với nhau.

Chỉ số nhóm

Các nhóm mã G được đánh số từ 00 đến 25. Khoảng này thay đổi giữa các bộ điều khiển khác nhau, tùy theo đặc tính của chúng. Số này có thể cao hơn trong các bộ điều khiển đời mới hoặc khi cần nhiều mã G hơn. Một trong các nhóm này, nhóm đặc trưng nhất và có lẽ quan trọng nhất – là *nhóm 00*.

Mọi lệnh chuẩn bị trong nhóm 00 đều là loại *không chế độ*. Chúng chỉ hoạt động trong block được lập trình. Nếu các mã G phi chế độ có tác dụng trong vài block liên tiếp, chúng phải được lập trình trong *từng* block đó. Trong hầu hết các lệnh phi chế độ, nói chung sự biểu thị này ít được sử dụng.

Ví dụ, sự tạm dừng (lệnh dwell) là khoảng dừng được lập trình với đơn vị đo là miligiây (ms). Lệnh này là cần thiết chỉ trong khoảng thời gian xác định. Sẽ là không logic khi lập trình dwell trong hai hoặc nhiều block liên tiếp. Ba block dưới đây đều có lệnh dwell.

```
N56 G04 P2000
N57 G04 P3000
N58 G04 P1000
```

Chương trình sẽ hiệu quả hơn và gọn hơn nếu nhập giá trị dwell tổng vào một block:

```
N56 G04 P6000
```

Bảng 7.3 liệt kê các nhóm mã G của hệ điều khiển Fanuc. Các ứng dụng phay và tiện được phân biệt bằng chữ M (milling – phay) và T (turning - tiện) trong cột *Kiểu* của bảng này.

Bảng 7.3. Nhóm mã G

Nhóm	Giải thích	Mã G	Kiểu
00	Mã G phi chế độ	G04 G09 G10 G11 G27 G28 G29 G30 G31 G37 G45 G45 G47 G48 G52 G53 G65 G51 G60 G92 G50 G70 G71 G72 G73 G74 G75 G76	M/T M/T M/T M/T M/T M T T T
01	Lệnh chuyển động Chu kỳ cắt	G00 G01 G02 G03 G32 G35 G36 G90 G92 G94	M/T T T
02	Chọn mặt phẳng	G17 G18 G19	M
03	Chế độ lập kích thước	G90 G91 (U và W dùng cho máy tiện)	M T
04	Hành trình đã lưu	G22 G23	M/T
05	Lượng ăn dao	G93 G94 G95	T
06	Nhập đơn vị	G20 G21	M/T
07	Bù bán kính dao	G40 G41 G42	M/T
08	Bù chiều dài dao	G43 G44 G49	M
09	Chu kỳ	G73 G74 G76 G80 G81 G82 G83 G84 G85 G86 G87 G88 G89	M M M M
10	Chế độ trả về	G98 G99	M
11	Xóa lập tỷ lệ, hình ảnh gương	G50 G68 G69	M T
12	Hệ tọa độ	G54 G55 G56 G57 G58 G59	M/T M/T
13	Chế độ cắt	G61 G62 G64 G63	M/T M
14	Chế độ macro	G66 G67	M/T
16	Quay tọa độ	G68 G69	M
17	CSS	G96 G97	T
18	Nhập (tọa độ) cực	G15 G16	M
24	Dao động tốc độ trục chính	G25 G26	M/T

Quan hệ trong nhóm nêu rõ ý nghĩa của từng nhóm. Ngoại lệ có thể là nhóm 01 các *lệnh chuyển động* và nhóm 09 *chu kỳ*. Quan hệ giữa hai nhóm này là nếu mã G từ nhóm 01 được chuyên biệt trong chu kỳ cố định bất kỳ thuộc nhóm 09, chu kỳ đó lập tức bị xóa, nhưng *không xảy ra điều ngược lại*. Nói cách khác, lệnh chuyển động hoạt động *không* bị xóa trong chu kỳ cố định.

Nhóm 01 không bị các mã G thuộc *nhóm 9* tác động. Tóm lại:

Mã G bất kỳ từ nhóm cho trước sẽ thay thế mã G khác một cách tự động trong cùng nhóm đó.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

CÁC KIỂU MÃ G

Hệ thống điều khiển Fanuc cung cấp sự lựa chọn linh hoạt các lệnh chuẩn bị. Đây là sự khác biệt giữa Fanuc và các bộ điều khiển khác. Các bộ điều khiển Fanuc được sử dụng rộng rãi trên toàn thế giới. Do đó cấu hình điều khiển tiêu chuẩn phải đáp ứng các tiêu chuẩn của từng quốc gia có sử dụng Fanuc, chẳng hạn sự lựa chọn đơn vị kích thước. Ở Châu Âu, Nhật và nhiều nước khác hệ mét là tiêu chuẩn. Ở Bắc Mỹ, hệ thống kích thước sử dụng đơn vị Anh. Do hai thị trường này chiếm hầu hết thương mại quốc tế, nhà chế tạo bộ điều khiển phải tham gia vào hai khu vực đó. Hầu như mọi nhà chế tạo bộ điều khiển đều đưa ra sự lựa chọn hệ thống kích thước, nhưng Fanuc và các bộ điều khiển tương tự đã đưa ra sự lựa chọn các mã lập trình trước khi họ đạt đến thị trường toàn cầu.

Phương pháp các bộ điều khiển Fanuc sử dụng là rất đơn giản, chỉ cần xác lập tham số. Bằng cách chọn tham số hệ thống thích hợp, có thể chọn một trong hai hoặc ba kiểu mã G, kiểu thích hợp nhất đối với người dùng. Mặc dầu đa số các mã G là như nhau đối với từng kiểu, nhưng minh họa rõ nhất là các mã G được dùng để chọn đơn vị đo theo hệ mét và hệ Anh. Nhiều bộ điều khiển của Mỹ ở thời kỳ đầu sử dụng G70 cho đơn vị Anh và G71 cho hệ mét. Hệ thống Fanuc cũ hơn lâu đã sử dụng G20 cho hệ Anh và G21 cho hệ mét.

Xác lập một tham số có thể chọn kiểu mã G có tính thực tiễn cao nhất. Thực tiễn này, nếu được thực hiện, thì chỉ được thực hiện một lần và chỉ khi lắp đặt hệ thống điều khiển, trước khi chương trình bất kỳ được viết cho hệ đó. Sự thay đổi kiểu mã G một cách ngẫu nhiên có thể đưa đến các hậu quả nghiêm trọng. Bạn cần nhớ, thay đổi ý nghĩa của một mã sẽ tác động đến ý nghĩa của mã khác. Sử dụng các đơn vị, ví dụ trên máy tiện, nếu G70 là lệnh nhập kích thước theo đơn vị Anh, bạn không thể sử dụng lệnh này để lập trình chu kỳ gia công thô. Fanuc cung cấp mã khác. Bạn chỉ nên dùng kiểu mã G tiêu chuẩn. Mọi mã G trong sách này đều dùng nhóm mặc định là các mã *kiểu A*, và là nhóm phổ biến nhất.

Mã G và dấu thập phân

Nhiều bộ điều khiển Fanuc hiện nay có mã G với dấu chấm thập phân, chẳng hạn G72.1 (sao chép quay) hoặc G72.2 (sao chép song song). Vài lệnh chuẩn bị trong nhóm này liên quan với máy cụ thể, hoặc ít sử dụng, do đó bạn cần đọc kỹ sổ tay hướng dẫn sử dụng máy CNC đó.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

8 CÁC HÀM CHUNG

Địa chỉ *M* trong chương trình CNC biểu thị hàm chung, đôi khi còn được gọi là *hàm máy*. Không phải mọi hàm chung đều liên quan đến sự vận hành của máy CNC, chỉ vài hàm liên hệ với sự xử lý chương trình. Trong sách này sẽ dùng thuật ngữ các *hàm chung* (miscellaneous function)

GIẢI THÍCH VÀ CÔNG DỤNG

Bên trong cấu trúc của chương trình CNC, nhà lập trình thường cần có vài phương tiện kích hoạt các tính năng xác định về vận hành máy hoặc điều khiển dòng chương trình. Nếu không có các phương tiện đó, chương trình là không đầy đủ và không thể thực thi. Trước hết bạn hãy xem xét các hàm chung liên quan đến sự vận hành máy, các *hàm máy* thực sự.

Các hàm liên quan với máy

Nhiều chế độ vận hành trên máy CNC cần được điều khiển bằng chương trình, để đảm bảo gia công hoàn toàn tự động. Các tính năng này thường sử dụng địa chỉ M và gồm các thao tác.

- Quay trục chính
  CW (thuận chiều) hoặc CCW (ngược chiều)
- Thay đổi khoảng bánh răng (số)
  Thấp / Trung bình / Cao
- Thay dao tự động — ATC
- Thay pallet tự động — APC
- Vận hành chất làm nguội — ON hoặc OFF
- Chuyển dịch ụ động — IN hoặc OUT

Các thao tác này thay đổi tùy theo máy, do thiết kế khác nhau giữa các nhà chế tạo. Thiết kế máy, theo quan điểm kỹ thuật, dựa trên ứng dụng máy cơ bản. Máy phay CNC sẽ đòi hỏi các chức năng liên quan đến máy hơi khác so với trung tâm gia công hoặc máy tiện CNC. Máy cắt EDM (tia lửa điện) được điều khiển bằng kỹ thuật số sẽ có nhiều tính năng đặc thù, đặc trưng cho kiểu gia công và không có trên các máy khác.

Ngay cả hai máy được thiết kế cho cùng loại công việc, chẳng hạn hai trung tâm gia công đứng, cũng có các tính năng khác nhau, nếu chúng có hệ thống CNC khác nhau hoặc các tùy chọn khác nhau rõ rệt. Các model máy khác nhau của cùng một nhà chế tạo vẫn có các tính năng đặc thù riêng.

Mọi máy công cụ được thiết kế để cắt gọt kim loại đều có các tính năng chung. Ví dụ, sự quay trục chính có thể có ba – *và chỉ có ba* – lựa chọn khả dĩ trong chương trình:

- Quay trục chính bình thường
- Quay trục chính đảo chiều
- Dừng trục chính.

Ngoài các tính năng này, còn có tính năng được gọi là sự định hướng trục chính cũng là tính năng liên hệ với máy. Ví dụ thứ hai là chất làm nguội. Chất làm nguội chỉ có thể điều khiển theo ON (hoạt động) hoặc OFF (dừng hoạt động)

Các thao tác đó là chung cho hầu hết các máy CNC. Tất cả đều được lập trình với hàm M, tiếp theo là không quá hai chữ số, dù một số bộ điều khiển cho phép sử dụng hàm M với ba chữ số, ví dụ Fanuc 16/18

Fanuc còn sử dụng các hàm M ba chữ số trong một số ứng dụng đặc biệt, ví dụ, để đồng bộ hóa hai ổ dao trên máy tiện 4 trục. Tất cả các tính năng này và nhiều tính năng khác liên quan với máy đều thuộc nhóm có tên chung là *hàm M* hoặc *mã M*.

Các hàm liên quan với chương trình

Ngoài các hàm máy, một số hàm M được dùng để điều khiển sự thực thi chương trình CNC. Ví dụ hàm M được dùng để tạm dừng sự thực thi chương trình trong khi thay đổi sự gá lắp, chẳng hạn đổi chiều chi tiết gia công. Ví dụ thứ hai là tình huống một chương trình gọi một hoặc nhiều chương trình con. Trong các trường hợp đó, mỗi chương trình đều có hàm gọi chương trình, số lần lặp lại,... Các hàm M xử lý các yêu cầu đó.

Từ các ví dụ nêu trên, sự sử dụng hàm M được chia thành hai nhóm chính, dựa trên ứng dụng cụ thể:

- Điều khiển các chức năng máy
- Điều khiển sự thực thi chương trình

Ở đây chỉ trình bày các hàm M phổ biến nhất, được hầu hết các bộ điều khiển sử dụng. Thật không may, có nhiều hàm khác nhau giữa các máy và hệ điều khiển. Các hàm đó được gọi là *hàm chuyên biệt theo máy*. Vì lý do đó, bạn

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

cần xem kỹ tài liệu về máy CNC và hệ điều khiển cụ thể.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

CÁC ỨNG DỤNG PHỔ BIẾN

Trước khi nghiên cứu các hàm M, bạn hãy chú ý kiểu hoạt động do các hàm đó thực hiện bất kể hoạt động liên quan đến máy hay chương trình. Ngoài ra, bạn cần chú ý các chế độ chuyển đổi hai trạng thái, chẳng hạn ON và OFF, IN và OUT (vào/ra), Forward (tiến) và Backward (lui), ... Bạn cần xem trước tài liệu về máy để bảo đảm tính thống nhất, mọi hàm M ở đây được liệt kê trong Bảng 8.1 và 8.2.

Bảng 8.1. Các ứng dụng phay

Mã M	Công dụng
M00	Dừng chương trình bắt buộc
M01	Dừng chương trình tùy chọn
M02	Kết thúc chương trình (với cài đặt lại, không quay lại từ đầu)
M03	Quay trục chính bình thường
M04	Quay trục chính đảo chiều
M05	Dừng trục chính
M06	Thay dao tự động (ATC)
M07	Phun sương chất làm nguội ON
M08	Chất làm nguội ON (động cơ bơm chất làm nguội ON)
M09	Chất làm nguội OFF (động cơ bơm chất làm nguội OFF)
M19	Định hướng trục chính
M30	Kết thúc chương trình (cài đặt lại và quay lại từ đầu)
M48	Xóa override lượng ăn dao OFF *(khử kích hoạt)*
M49	Xóa override lượng ăn dao ON *(kích hoạt)*
M60	Thay pallet tự động (APC)
M78	Kẹp trục B *(phi tiêu chuẩn)*
M79	Mở kẹp trục B *(phi tiêu chuẩn)*
M98	Gọi chương trình con
M99	Kết thúc chương trình con

Bảng 8.2. Các ứng dụng tiện

Mã M	Công dụng
M00	Dừng chương trình bắt buộc
M01	Dừng chương trình tùy chọn
M02	Kết thúc chương trình (với cài đặt lại, không quay lại từ đầu)
M03	Quay trục chính bình thường
M04	Quay trục chính đảo chiều
M05	Dừng trục chính
M07	Phun sương chất làm nguội ON
M08	Chất làm nguội ON (động cơ bơm chất làm nguội ON)
M09	Chất làm nguội OFF (động cơ bơm chất làm nguội OFF)
M10	Mở mâm cặp
M11	Đóng mâm cặp
M12	U động IN
M13	U động OUT
M17	Phản độ ổ dao thuận
M18	Phản độ ổ dao đảo
M19	Định hướng trục chính (tùy chọn)
M41	U động tiến
M22	U động lui
M23	Ren ON
M24	Ren OFF
M30	Kết thúc chương trình (cài đặt lại và quay lại từ đầu)
M41	Chọn số (bánh răng) thấp
M42	Chọn số (bánh răng) trung bình 1
M43	Chọn số (bánh răng) trung bình 2
M44	Chọn số (bánh răng) cao
M48	Xóa override lượng ăn dao OFF *(khử kích hoạt)*
M49	Xóa override lượng ăn dao ON *(kích hoạt)*
M98	Gọi chương trình con
M99	Kết thúc chương trình con

Các hàm MDI đặc biệt

Một số hàm M không được phép sử dụng trong chương trình CNC. Nhóm này được dùng hoàn toàn trong chế độ *Manual Data Input* (MDI, nhập dữ liệu bằng tay). Ví dụ, sự thay dao từng bước trên trung tâm gia công, chỉ dùng để bảo dưỡng máy, không được đưa vào chương trình.

Các nhóm ứng dụng

Hai nhóm hàm M chính, đã đề cập ở phần trên, có thể tiếp tục chia thành nhiều nhóm nhỏ dựa trên ứng dụng chuyên biệt của các hàm M trong từng nhóm. Danh sách các nhóm được nêu trong Bảng 8.3

Bảng 8.3. Các nhóm hàm M

Nhóm	Hàm M
Chương trình	M00 M01 M02 M30
Trục chính	M03 M04 M05 M19
Thay dao	M06
Chất làm nguội	M07 M08 M09
Phụ tùng	M10 M11 M12 M13 M17 M18 M21 M22 M78 M79
Gia công ren	M23 M24
Khoảng hộp số	M41 M42 M43 M44
Override lượng ăn dao	M48 M49
Chương trình con	M98 M99
Pallet	M60

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Bảng này không nêu ra tất cả các nhóm, do có sự khác biệt giữa các máy. Mặt khác, Bảng 8.3 nêu rõ các *kiểu* ứng dụng với các hàm M được dùng trong lập trình CNC hàng ngày.

Các hàm M được đề cập trong chương này sẽ được sử dụng trong nhiều chương kế tiếp, một số sẽ xuất hiện thường xuyên, phản ánh công dụng của chúng trong lập trình. Các hàm không ứng với hệ thống điều khiển cụ thể sẽ không được sử dụng hoặc không cần thiết. Tuy nhiên, khái niệm về ứng dụng của chúng luôn luôn tương tự nhau trong hầu hết các hệ điều khiển và các máy CNC

Trong chương này, chỉ đề cập chi tiết các hàm tổng quát. Các hàm M còn lại sẽ được trình bày trong các phần về từng ứng dụng riêng rẽ. Ở giai đoạn này, chỉ tập trung vào công dụng và hành vi của các hàm M phổ biến nhất.

HÀM M TRONG BLOCK

Nếu hàm M được lập trình trong block sẽ không có dữ liệu đi kèm, chỉ có hàm này được thực thi, ví dụ:

```
N45 M01
```

là sự dừng tùy chọn. Block này đúng, hàm M có thể là mục nhập duy nhất trong block. Khác với các lệnh chuẩn bị (mã G) chỉ *một* hàm M là được phép trong block, trừ khi bộ điều khiển cho phép nhiều hàm M trong một block, lỗi chương trình sẽ xuất hiện (các bộ điều khiển hiện đại)

Phương pháp thực tiễn lập trình một số hàm M xác định là trong block có *chuyển động dao cắt*. Ví dụ, tiện với chất làm nguội được kích hoạt và *cùng lúc đó* có thể phải dịch chuyển dao cắt đến vị trí xác định trên chi tiết gia công. Khi không có sự va chạm giữa các lệnh, block này có thể có dạng như sau:

```
N56 G00 X12.9854 Y9.474 M08
```

Trong ví dụ này, block N56, *thời gian* chính xác hàm M08 sẽ được kích hoạt là không quá quan trọng. Trong các trường hợp khác, thời gian này có thể rất quan trọng. Một số hàm M *phải* có hiệu lực trước hoặc sau khi xảy ra một hoạt động nào đó. Ví dụ, bạn hãy quan sát sự kết hợp – chuyển động trục Z được áp dụng cùng với hàm M00 *dừng chương trình* trong cùng một block.

```
N319 G01 Z-12.8456 F20.0 M00
```

Đây là tình huống rất khó xác định, cần tìm hai câu trả lời. Một là *chính xác* điều gì sẽ xảy ra, thứ hai là *thời điểm chính xác* điều đó sẽ xảy ra, khi hàm M00 được kích hoạt. Có ba khả năng và ba câu hỏi.

1. Sự dừng chương trình sẽ xảy ra tức thì, khi chuyển động cuối kích hoạt – ở đầu block?
2. Sự dừng chương trình sẽ xảy ra khi dao cắt trên hành trình - trong khi chuyển động.
3. Sự dừng chương trình sẽ xảy ra khi hoàn tất lệnh chuyển động ở cuối block?

Một trong ba khả năng này sẽ xảy ra – nhưng là khả năng nào? Ngay cả nếu mục đích thực tiễn của các ví dụ nêu trên có thể chưa rõ ràng ở giai đoạn này, nhưng vẫn hữu ích khi biết cách thức hệ thống điều khiển diễn dịch các block có chứa chuyển động dao cắt và hàm M.

Mỗi hàm M đều được thiết kế một cách logic và được thiết kế để có *ý nghĩa chung*.

Sự khởi động thực sự của hàm M được chia thành hai nhóm, không phải ba nhóm:

- ❑ Hàm M kích hoạt ở đầu block (đồng thời với chuyển động dao)
- ❑ Hàm M kích hoạt ở cuối block (sau khi hoàn tất chuyển động dao)

Không có hàm M nào được kích hoạt *trong* quá trình thực thi block. Sự khởi động logic của hàm M08 (chất làm nguội ON) trong block N56 nêu trên là gì? Câu trả lời chính xác là chất làm nguội sẽ được kích hoạt vào *cùng thời điểm* khi *bắt đầu* chuyển động chạy dao. Câu trả lời chính xác cho block N319 là hàm dừng chương trình M00 sẽ được kích hoạt *sau* khi hoàn tất chuyển động chạy dao. Điều này thực sự có ý nghĩa. Nhưng còn các hàm khác, chúng sẽ ứng xử trong block như thế nào? Bạn hãy tiếp tục xem xét.

Khởi động các hàm M

Bạn hãy xem lại Bảng 8.1 và 8.2 về các hàm M đặc trưng. Bạn hãy bổ sung chuyển động chạy dao và thử xác định cách thức hàm đó ứng xử, dựa theo các ví dụ đã nêu ở phần trên. Một chút suy nghĩ logic sẽ cung cấp cho bạn cơ hội đi đến kết luận đúng. Bạn hãy so sánh hai nhóm dưới đây.

Các hàm M kích hoạt ở ĐẦU BLOCK	
M03	Sự quay trục chính bình thường
M04	Sự quay trục chính đảo chiều
M06	Thay dao tự động (ATC)
M07	Phun sương chất làm nguội ON
M08	Chất làm nguội ON (động cơ bơm chất làm nguội ON)

Các hàm M kích hoạt ở CUỐI BLOCK	
M00	Dừng chương trình bắt buộc
M01	Dừng chương trình tùy chọn

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Các hàm M kích hoạt ở CUỐI BLOCK	
M02	Kết thúc chương trình (cài đặt lại, không quay lại từ đầu)
M05	Dừng trục chính
M09	Chất làm nguội OFF (động cơ bơm chất làm nguội OFF)
M30	Kết thúc chương trình (cài đặt lại và quay lại từ đầu)
M60	Thay pallet tự động (APC)

Nếu chưa biết chắc về cách thức hàm M tương tác với chuyển động dao, lựa chọn an toàn nhất là lập trình hàm M theo *block riêng*. Theo cách đó, hàm M sẽ luôn luôn được xử lý *trước* hoặc *sau* block chương trình liên quan. Trong hầu hết các ứng dụng đây sẽ là giải pháp an toàn.

Thời hạn của hàm M

Khi biết *thời điểm* hàm M có hiệu lực, sẽ là logic khi đặt câu hỏi khoảng thời gian có hiệu lực là *bao nhiêu*. Một số hàm M có hiệu lực chỉ trong block hàm đó xuất hiện. Các hàm khác sẽ tiếp tục hoạt động cho đến khi được xóa bằng hàm M khác. Điều này là tương tự với tính chế độ của các lệnh chuẩn bị G, tuy nhiên, thuật ngữ chế độ thường không được sử dụng với hàm M. Để xét thời hạn của hàm M, bạn hãy xem ví dụ về các hàm M00 và M01. Hai hàm này chỉ có hiệu lực trong *một* block. Hàm chất làm nguội ON (M08), sẽ hoạt động cho đến khi hàm *xóa* hoặc hàm *thay thế* được lập trình. Bạn cần nhớ, hàm bất kỳ trong số các hàm sau đây sẽ xóa chế độ chất làm nguội ON – M00, M01, M02, M09 và M30. Bạn hãy so sánh hai nhóm hàm dưới đây:

Hàm M hoàn tất trong MỘT BLOCK	
M00	Dừng chương trình bắt buộc
M01	Dừng chương trình tùy chọn
M02	Kết thúc chương trình (cài đặt lại, không quay lại từ đầu)
M06	Thay dao tự động (ATC)
M30	Kết thúc chương trình (cài đặt lại và quay lại từ đầu)
M60	Thay pallet tự động (APC)

Hàm M hoạt động CHO ĐẾN KHI BỊ XÓA hoặc bị THAY THẾ	
M03	Sự quay trục chính bình thường
M04	Sự quay trục chính đảo chiều
M05	Dừng trục chính
M07	Phun sương chất làm nguội ON
M08	Chất làm nguội ON (động cơ bơm chất làm nguội ON)
M09	Chất làm nguội OFF (động cơ bơm chất làm nguội OFF)

Sự phân loại này là hoàn toàn logic và có ý nghĩa. Bạn không cần nhớ từng hàm M và hoạt động chính xác của chúng. Cách tốt nhất là xem sổ tay hướng dẫn đi kèm với máy CNC và quan sát chương trình chạy trên máy đó.

CÁC HÀM CHƯƠNG TRÌNH

Các hàm M điều khiển sự xử lý chương trình có thể được dùng để tạm dừng sự xử lý (ở giữa chương trình) hoặc dừng hẳn (ở cuối chương trình). Vài hàm khả dụng cho mục đích này.

Dừng chương trình

Hàm M00 được định nghĩa là hàm dừng chương trình *bắt buộc* hoặc *vô điều kiện*. Thời điểm bất kỳ hệ điều khiển gặp hàm này trong khi xử lý chương trình, mọi sự vận hành tự động của máy công cụ sẽ dừng lại:

- Chuyển động của tất cả các trục
- Sự quay của trục chính
- Chức năng làm nguội
- Sự thực thi chương trình tiếp theo

Bộ điều khiển sẽ không cài đặt lại khi hàm M00 được xử lý. Mọi dữ liệu chương trình đang có hiệu lực đều được duy trì (lượng ăn dao, xác lập tọa độ, tốc độ trục chính, ...). Sự xử lý chương trình chỉ có thể được tái lập bằng cách kích hoạt phím *Cycle Start*. Hàm M00 xóa sự quay trục chính và chức năng chất làm nguội, cần lập trình lại trong các block kết tiếp.

Hàm M00 có thể được lập trình với block riêng rẽ hoặc trong block có các lệnh khác, thường là chuyển động trục. Nếu hàm M00 được lập trình cùng với lệnh chuyển động, sự chuyển động sẽ hoàn tất trước, sau đó sự dừng chương trình sẽ có hiệu lực:

➲ **M00 được lập trình sau lệnh chuyển động:**

```
N38 G00 X13.5682
N39 M00
```

➲ **M00 được lập trình với lệnh chuyển động:**

```
N39 G00 X13.5682 M00
```

Trong cả hai trường hợp, lệnh chuyển động sẽ được hoàn thành trước, *trước khi* thực thi sự dừng chương trình. Sự khác biệt giữa hai ví dụ nêu trên chỉ là chế độ xử lý trong một block (ví dụ, khi cắt thử) hoặc hai block, không có sự khác biệt thực tiễn trong chế độ xử lý tự động (công tắc Single Block ở vị trí OFF).

Công dụng thực tiễn

Hàm dừng chương trình được sử dụng trong chương trình làm cho công việc của người vận hành CNC trở nên dễ dàng hơn. Hàm này rất

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

hữu ích đối với nhiều ứng dụng. Ứng dụng phổ biến là kiểm tra chi tiết gia công trên máy. Trong khi dừng, có thể kiểm tra các kích thước chi tiết và dụng cụ cắt. Các phoi tích tụ trong lỗ doa hoặc khoan có thể được loại bỏ, ví dụ, trước khi khởi đầu nguyên công kế tiếp, chẳng hạn cắt ren lỗ cụt. Hàm dừng chương trình còn cần thiết để thay đổi sự gá lắp hiện hành ở giữa chương trình, ví dụ, để đảo chiều chi tiết. Sự thay dao bằng tay cũng đòi hỏi phải có hàm M00 trong chương trình.

Hàm dừng chương trình M00 chỉ được dùng để can thiệp bằng tay trong khi xử lý chương trình.

Bộ điều khiển còn cung cấp hàm M01, dừng chương trình *tùy chọn*. Nguyên tắc chính về sử dụng M00 là sự cần thiết phải *can thiệp bằng tay* đối với *từng* chi tiết gia công. Thay dao bằng tay trong chương trình phải dùng hàm M00, do từng chi tiết đều đòi hỏi điều đó. Bạn có thể không cần dùng M00 để kiểm tra kích thước chi tiết, nếu điều đó là không thường xuyên. M01 là lựa chọn tốt hơn. Dù sự khác biệt giữa hai hàm này là tương đối nhỏ, nhưng khác biệt thực tế về thời gian chu kỳ có thể rất rõ rệt khi số lượng chi tiết gia công đủ lớn.

Khi sử dụng hàm M, cần thông báo cho người vận hành về *nguyên nhân và mục đích* sử dụng để tránh nhầm lẫn. Điều này có thể trở nên khả dụng cho người vận hành theo hai cách.

- Trong bảng hướng dẫn, bạn ghi rõ block chứa hàm M00 và thao tác bằng tay cần thực hiện.

```
BLOCK N39.....LOẠI BỎ PHOI
```

- Trong chương trình, có phần chú thích với thông tin cần thiết. Phần chú thích phải được đặt trong ngoặc đơn, ví dụ:

```
[A] N39 M00 (LOẠI BỎ PHOI)
[B] N30 X13.5682 M00 (LOẠI BỎ PHOI)
[C] N38 X13.5682 M00 (LOẠI BỎ PHOI)
```

Ví dụ bất kỳ trong ba ví dụ nêu trên sẽ cung cấp thông tin cần thiết cho người vận hành CNC. Trong hai phương pháp nêu trên, phần chú thích trong chương trình được ưa thích hơn, do các hướng dẫn hoặc chú thích có thể đọc trực tiếp trên màn hình của bảng hướng dẫn.

Dừng chương trình tùy chọn

Hàm M01 là hàm dừng chương trình *tùy chọn* hoặc *có điều kiện*. Hàm này tương tự hàm M00 nhưng có một khác biệt. Khác với hàm M00, khi gặp hàm M01 trong chương trình, sự xử lý chương trình sẽ không dừng lại, trừ khi người vận hành can thiệp thông qua bảng điều khiển. Nút nhất hoặc công tắc chuyển đổi *Optional Stop* trên bảng này có thể được xác lập ON hoặc OFF. Khi hàm M01 trong chương trình được thực thi, xác lập của công tắc này sẽ xác định chương trình tạm dừng hay tiếp tục thực thi:

Xác lập công tắc Optional Stop	Kết quả của M01
ON	Sự xử lý *sẽ* dừng lại
OFF	Sự xử lý *sẽ không* dừng lại

Nếu không có hàm M01 trong chương trình, xác lập của công tắc *Optional Stop* sẽ không có tác dụng. Nói chung công tắc này thường ở vị trí OFF trong sản xuất hàng loạt.

Khi hoạt động, hàm M01 ứng xử hoàn toàn tương tự như M00. Chuyển động của tất cả các trục, sự quay trục chính, chức năng làm nguội, và sự thực thi chương trình đều tạm dừng. Lượng ăn dao, các xác lập tọa độ, xác lập tốc độ trục chính, ... vẫn được giữ lại. Sự xử lý chương trình chỉ có thể tái kích hoạt bằng phím *Cycle Start*. Mọi nguyên tắc lập trình đối với hàm M00 đều có thể áp dụng cho M01.

Ý tưởng tốt là lập trình hàm M01 ở cuối từng dụng cụ cắt, tiếp theo là dòng trắng không có dữ liệu. Nếu sự xử lý chương trình có thể tiếp tục mà không dừng lại, công tắc *Optional Stop* sẽ được xác lập theo OFF và không bị mất thời gian sản xuất. Nếu cần dừng chương trình tạm thời ở cuối dụng cụ cắt, công tắc này sẽ được xác lập theo ON và sự xử lý dừng lại ở cuối dao cắt đó. Tổn thất thời gian thường được đánh giá theo hoàn cảnh cụ thể, ví dụ, thay dao mới, kiểm tra kích thước hoặc độ bóng bề mặt chi tiết,...

Kết thúc chương trình

Mỗi chương trình đều phải có hàm đặc biệt xác định sự kết thúc chương trình hiện hành. Đối với mục đích này có hai hàm M khả dụng – M02 và M30. Hai hàm này tương tự nhau, nhưng có mục đích khác nhau. Hàm M02 sẽ kết thúc chương trình, nhưng *không quay lại* block thứ nhất ở đầu chương trình. Hàm M30 kết thúc chương trình, đồng thời *quay trở lại* đầu chương trình.

Khi bộ điều khiển đọc hàm kết thúc chương trình M02 hoặc M30, bộ này sẽ xóa mọi chuyển động trục, sự quay trục chính, chức năng làm nguội và thường cài đặt lại hệ thống theo các điều kiện mặc định. *Trên một số bộ điều khiển sự cài đặt lại có thể là không tự động, nhà lập trình cần chú ý điều đó.*

Nếu chương trình kết thúc với hàm M02, bộ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

điều khiển vẫn là phần kết thúc đó sẵn sàng cho Cycle Start (khởi động chu kỳ) kế tiếp. Trên thiết bị CNC hiện đại, hoàn toàn không cần hàm M02, trừ tính tương thích ngược (với các hệ điều khiển cũ). Hàm này được sử dụng để bổ sung cho hàm M30 đối với các máy (chủ yếu là máy tiện NC) có đầu đọc băng từ mà không có cuộn băng, chỉ dùng băng từ vòng ngắn. Bộ kéo băng từ được đặt chung với đầu dẫn băng, tạo thành vòng kín. Khi chương trình hoàn tất, sự khởi đầu băng từ là ở cuối băng của chu kỳ đã qua, do đó không cần trả băng. Các băng dài không sử dụng vòng kín do đó phải có cuộn băng và M30.

M02 và M30

Trên hầu hết các bộ điều khiển hiện tại, tham số hệ thống có thể được xác lập với M02 có cùng ý nghĩa như M30. Xác lập này cho phép trở lại từ đầu chương trình (quay hoặc trả lại băng từ), rất hữu ích trong các tình huống sử dụng chương trình cũ trên máy có bộ điều khiển hiện đại không cần thực hiện các thay đổi (do tính tương thích ngược).

Tóm lại, nếu sự kết thúc chương trình được thực hiện bằng hàm M30, sự trả lại băng từ *sẽ* được thực hiện, nếu dùng hàm M02 *sẽ không* trả lại băng từ.

Khi viết chương trình, bạn cần bảo đảm block cuối cùng trong chương trình chỉ chứa M30 để kết thúc (block chuỗi thứ tự được phép khởi động block đó).

```
N65 ...
N66 G91 G28 X0 Y0
N67 M30                    (kết thúc chương trình)
%
```

Trên một số bộ điều khiển, hàm M30 có thể được sử dụng cùng với chuyển động của các trục. *Điều này là không nên*!

```
N65 ...
N66 G91 G28 X0 Y0 M30      (Kết thúc ch. trình)
%
```

Dấu phần trăm

Dấu phần trăm (%) sau M30 là mã dừng đặc biệt. Ký hiệu này kết thúc sự tải chương trình từ thiết bị ngoại vi. Dấu này còn được gọi là *dấu kết thúc tập tin*.

Kết thúc chương trình con

Hàm M cuối cùng để kết thúc chương trình là M99. Công dụng chính của hàm này là trong các chương trình con. Nói chung, hàm M99 sẽ kết thúc chương trình con và trở về sự xử lý của chương trình cũ. Nếu M99 được dùng trong chương trình tiêu chuẩn, sẽ tạo ra chương trình *không có phần cuối*. Trường hợp này được gọi là *vòng (lặp) vô tận*. Do đó M99 chỉ được dùng trong các chương trình con.

CÁC HÀM MÁY

Các hàm M liên quan với sự vận hành máy được xếp vào một nhóm riêng. Phần này sẽ trình bày một số hàm M quan trọng trong nhóm hàm liên quan đến máy công cụ.

Các hàm điều khiển chất làm nguội

Hầu hết các nguyên công cắt gọt kim loại đều đòi hỏi làm nguội dụng cụ cắt bằng chất làm nguội thích hợp. Để điều khiển dòng chất làm nguội trong chương trình, có ba hàm M được sử dụng cho mục đích này:

M07	Phun sương ON
M08	Phun dòng ON
M09	Phun sương hoặc phun dòng OFF

Sương là hỗn hợp dầu cắt gọt pha trộn với không khí nén. Tùy theo nhà chế tạo máy công cụ, hàm này có thể là tiêu chuẩn hoặc phi tiêu chuẩn trên máy CNC cụ thể. Một số nhà chế tạo thay hỗn hợp dầu và không khí bằng không khí, hoặc chỉ phun dầu, ... Trong các trường hợp đó, trên máy công cụ thường có thêm thiết bị chuyên dùng. Nếu có tùy chọn này trên máy công cụ, hàm kích hoạt sự phun sương là M07.

Hàm tương tự M07 là M08, *phun dòng* chất lỏng làm nguội. Đây là ứng dụng làm nguội phổ biến trong lập trình CNC và là tiêu chuẩn đối với hầu hết các máy CNC. Chất làm nguội, thường là hỗn hợp thích hợp của dầu hòa tan và nước, được pha sẵn và chứa trong thùng chứa trên máy công cụ. Sự làm nguội lưỡi cắt của dụng cụ cắt là rất quan trọng vì ba lý do:

- Giải nhiệt
- Loại bỏ phoi
- Bôi trơn

Lý do chính để sử dụng dung dịch làm nguội hướng đến lưỡi cắt là giải nhiệt phát sinh trong khi cắt gọt. Lý do thứ hai là loại bỏ phoi ra khỏi khu vực cắt, sử dụng áp lực của dung dịch này. Cuối cùng, dung dịch làm nguội tác động như chất bôi trơn để giảm ma sát giữa dao cắt và phôi. Sự bôi trơn làm tăng tuổi bền dụng cụ cắt và cải thiện độ bóng bề mặt.

Nói chung, thường không cần dùng đến dung dịch làm nguội khi dao cắt bắt đầu tiến đến bề mặt phôi hoặc khi dao trở về vị trí thay dao. Để dừng chức năng làm nguội, bạn hãy dùng hàm M09 – *Coolant off*. M09 sẽ dừng sự

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

cung cấp sương dầu hoặc dung dịch chất làm nguội. Trong thực tế, hàm M09 sẽ tắt động cơ bơm chất làm nguội.

Các hàm điều khiển chất làm nguội có thể được lập trình trong các block riêng rẽ hoặc cùng với chuyển động trục. Có các khác biệt nhỏ nhưng rất quan trọng trong thứ tự và thời điểm xử lý chương trình. Các ví dụ dưới đây sẽ giải thích rõ những khác biệt đó:

➲ **Ví dụ A - Phun sương ON, nếu khả dụng**

```
N 110 M07
```

➲ **Ví dụ B - chất làm nguội ON:**

```
N340 M08
```

➲ **Ví dụ C - Chất làm nguội OFF:**

```
N500 M09
```

➲ **Ví dụ D - chuyển động trục và chất làm nguội ON:**

```
N230 G00 X11.5 Y10.0 M08
```

➲ **Ví dụ E - chuyển động trục và chất làm nguội OFF**

```
N400 G00 Z1.0 M09
```

Các ví dụ này nêu rõ sự khác biệt trong xử lý chương trình. Nguyên tắc chung về lập trình chất làm nguội bao gồm:

- Chất làm nguội ON hoặc OFF trong block *riêng rẽ* sẽ hoạt động trong block hàm đó được lập trình (các ví dụ A, B và C).
- Chất làm nguội ON, khi lập trình với *chuyển động của các trục*, sẽ trở nên hoạt động đồng thời với sự chuyển động của các trục (ví dụ D).
- Chất làm nguội OFF, khi lập trình với *chuyển động của các trục*, sẽ hiệu lực chỉ khi hoàn tất sự chuyển động của các trục (ví dụ E).

Công dụng chính của hàm M08 là *kích hoạt động cơ bơm chất làm nguội*, không bảo đảm lưỡi cắt nhận được chất làm nguội một cách tức thời. Trên các máy lớn với ống dẫn chất làm nguội dài hoặc máy với áp suất bơm chất làm nguội thấp, chất làm nguội có thể bị trễ do phải vượt qua khoảng cách từ bơm đến dao cắt.

Chất làm nguội cần được lập trình với hai vấn đề quan trọng:

- Chất làm nguội không được phép phun ra ngoài khu vực làm việc (xung quanh máy)
- Không được phép xảy ra tình huống chất làm nguội phun đến lưỡi cắt nóng.

Vấn đề thứ nhất tương đối nhỏ. Nếu hàm điều khiển chất làm nguội bị sai vị trí trong chương trình, kết quả có thể chỉ là sự bất tiện. Khu vực bị ướt xung quanh máy sẽ dẫn đến các điều kiện không an toàn và cần được giải quyết một cách nhanh chóng. Tình huống nghiêm trọng hơn xảy ra khi chất làm nguội đột ngột phun vào lưỡi cắt trong khi cắt gọt. Sự thay đổi nhiệt độ ở lưỡi cắt có thể làm cho dao bị gãy và chi tiết gia công bị hư. Lưỡi cắt bằng hợp kim cứng bị ảnh hưởng của nhiệt độ lớn hơn so với thép gió (thép dụng cụ cắt gọt tốc độ cao). Bạn có thể ngăn chặn các vấn đề này trong khi lập trình, bằng cách sử dụng hàm M08 trong *các block đứng trước* block cắt. Ống dẫn dài hoặc áp suất chất làm nguội không đủ có thể làm chậm sự khởi động phun chất làm nguội.

Các hàm trục chính

Chương 11 - Điều khiển trục chính, trình bày chi tiết về điều khiển trục chính của máy CNC trong chương trình CNC. Các hàm khả dụng để điều khiển trục chính là hàm *quay* và *định hướng* trục.

Hầu hết các trục chính đều có thể quay theo cả hai chiều, *thuận chiều đồng hồ* (CW- thuận) và *ngược chiều đồng hồ* (CCW-ngược). Chiều quay luôn luôn liên quan với điểm chiếu tiêu chuẩn. Điểm chiếu được thiết lập từ phía trục chính theo *chiều dọc theo đường tâm trục chính hướng đến mặt trục*. Quay CW theo điểm chiếu này được lập trình với hàm M03. chiều CCW là M04, nếu trục chính có thể quay theo cả hai chiều.

Các kiểu máy khoan và phay thường sử dụng quy ước này. Quy ước đó cũng được áp dụng cho các máy tiện CNC. Trên máy phay CNC hoặc trung tâm gia công, sẽ thực tế hơn khi quan sát hướng đến chi tiết từ phía trục chính thay vì từ phía bàn máy. Trên máy tiện (kiểu ngang với băng máy nghiêng), sẽ thực tế hơn khi dùng điểm chiếu *từ ụ động hướng tới trục chính*, do đây là khoảng cách ngắn nhất đối với người vận hành CNC đứng trước máy tiện. Tuy nhiên, các chiều trục chính M03 và M04 được thiết lập theo cùng cách thức như với trung tâm gia công. Sự phức tạp còn do các dao cắt trái được dùng trên máy tiện thường xuyên hơn so với máy phay. Bạn nên nghiên cứu kỹ sổ tay hướng dẫn của máy CNC cụ thể và xem chi tiết trong Chương 11.

Hàm để dừng trục chính trong chương trình là M05. Hàm này sẽ dừng trục chính *bất kể chiều quay*. Trên nhiều máy, hàm M05 còn phải được lập trình *trước khi* đảo chiều quay trục chính:

```
M03                                    (SPINDLE CW)
...
<... Maching at the current location ...>
...
M05                                    (SPINDLE STOP)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
<... Usually a tool change ...>
M04                          (SPINDLE CCW)
...
<... Maching at the current location ...>
...
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Hàm M05 còn có thể được yêu cầu khi thay đổi khoảng tốc độ (bánh răng) trên máy tiện CNC. Sự dừng trục chính, lập trình trong block có chứa chuyển động trục, sẽ xảy ra *sau khi* hoàn tất chuyển động đó.

Hàm điều khiển trục chính cuối cùng là M19, được gọi là *định hướng trục chính*. Một số nhà chế tạo bộ điều khiển gọi đây là hàm khóa chốt trục chính. Bất kể tên gọi, hàm M19 sẽ làm cho trục chính dừng lại ở vị trí định hướng. Hàm này được sử dụng chủ yếu trong khi xác lập máy, rất ít khi trong chương trình. Trục chính cần được định hướng trong hai tình huống cơ bản:

- ❑ Thay dụng cụ cắt tự động (ATC).
- ❑ Dịch chuyển dụng cụ cắt trong nguyên công doa (chỉ các chu kỳ doa G76 và G87).

Khi sử dụng hàm M06 *(Automatic Tool Change – ATC, thay dao tự động)* trong chương trình, đối với hầu hết các trung tâm gia công, bạn không cần lập trình định hướng trục chính. Định hướng này đã được lập trình trong chuỗi thứ tự thay dao tự động và bảo đảm định vị chính xác mọi giá đỡ dao. Một số nhà lập trình thích lập trình M19 với sự trả về zero của máy đối với vị trí thay dao, để tiết kiệm một hoặc hai giây thời gian chu kỳ.

Sự định hướng trục chính là cần thiết đối với một số nguyên công doa trên máy phay. Để ra khỏi lỗ, dao doa không chạm vào vách trục đã gia công, cần phải dừng trục chính, đầu lưỡi cắt phải được định hướng, sau đó lấy dao ra khỏi lỗ. Phương pháp tương tự cũng được áp dụng cho các nguyên công doa ngược. Tuy nhiên, các nguyên công cắt gọt đặc biệt này sử dụng chu kỳ cố định trong chương trình, với định hướng trục được lập trình (Chương 24).

Nói chung, hàm M19 ít khi được dùng trong chương trình, thường chỉ có tác dụng hỗ trợ lập trình và gá lắp chi tiết khi sử dụng MDI (nhập dữ liệu bằng tay).

Lựa chọn khoảng tốc độ

Hầu như mọi lựa chọn khoảng tốc độ có thể lập trình đều áp dụng cho máy tiện CNC. Trên các trung tâm gia công, khoảng tốc độ trục chính có thể thay đổi hoàn toàn tự động. Hầu hết các máy tiện CNC đều có không dưới hai khoảng tốc độ, các máy công suất cao có đến bốn khoảng tốc độ. Nguyên tắc lập trình cơ bản là lựa chọn khoảng bánh răng dựa trên thao tác gia công.

Ví dụ, hầu hết các chế độ gia công thô đều đòi hỏi *công suất* của trục chính hơn là *tốc độ* trục chính. Trong trường hợp đó, chọn khoảng tốc độ thấp. Đối với gia công tinh, thường dùng khoảng tốc độ cao hoặc trung bình để bảo đảm độ bóng cần thiết.

Sự phân phối các hàm M hoàn toàn phụ thuộc vào số khoảng tốc độ khả dụng trên máy tiện. Số các khoảng là 1, 2, 3 hoặc 4. Bảng 8.4 liệt kê sự phân phối các hàm M, nhưng bạn nên xem kỹ các lệnh thực tế trong sổ tay hướng dẫn sử dụng máy công cụ.

Bảng 8.4. *Hàm M và khoảng tốc độ*

Khoảng	Hàm M	Khoảng bánh răng (hộp số)
1	N/A	Không lập trình
2	M41 M42	Khoảng thấp Khoảng cao
3	M41 M42 M43	Khoảng thấp Khoảng trung bình Khoảng cao
4	M41 M42 M43 M44	Khoảng thấp Khoảng trung bình 1 Khoảng trung bình 2 Khoảng cao

Nguyên tắc chung là khoảng cách bánh răng càng cao, tốc độ trục chính càng cao và công suất càng thấp, hoặc ngược lại. Nói chung, không cần dừng trục chính để đổi số (tốc độ) nhưng bạn cần xem kỹ sổ tay hướng dẫn. Nếu có nghi ngờ, bạn nên dừng trục chính trước, đổi khoảng tốc độ, sau đó khởi động lại trục chính.

Phụ tùng máy

Hầu hết các hàm M đều được dùng cho sự vận hành các phụ tùng kèm theo máy công cụ. Trong nhóm này, các ứng dụng phổ biến đã được trình bày chi tiết, đặc biệt là điều khiển chất làm nguội và thay đổi tốc độ. Các hàm M còn lại sẽ được giải thích chi tiết trong những ứng dụng có liên quan. Các hàm M liên hệ với máy được nêu trong Bảng 8.5.

Bảng 8.5. *Các hàm M thông dụng*

Hàm M	Chức năng	Kiểu
M06	Thay dao tự động (ATC)	M
M60	Thay pallet tự động (APC)	M
M23 M24	Điều khiển cắt ren ON/OFF	T
M98 M99	Gọi chương trình con/kết thúc chương trình con	M/T

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

9 Block chuỗi thứ tự

Mỗi dòng trong chương trình CNC được gọi là *block*. Block được định nghĩa là nhóm lệnh đơn được hệ thống CNC xử lý.

Block chuỗi thứ tự, block chương trình, hoặc đơn giản là block, thường là một dòng viết tay trong bản thảo chương trình, hoặc gõ trên máy tính trong bộ soạn thảo văn bản và kết thúc bằng phím *Enter*. Dòng này có thể gồm một hoặc vài từ ngữ chương trình, xác định nhóm lệnh đơn cho máy CNC. Nhóm lệnh chương trình có thể là tổ hợp gồm các lệnh chuẩn bị, từ ngữ tọa độ, các hàm và lệnh dụng cụ cắt, hàm điều khiển chất làm nguội, các lệnh tốc độ và lượng ăn dao, vị trí, ... Nội dung của một block sẽ được xử lý như một đơn vị trước khi bộ điều khiển xử lý block kế tiếp. Khi toàn bộ chương trình CNC được xử lý, hệ thống điều khiển sẽ đánh giá các nhóm lệnh riêng rẽ (block) *như một bước vận hành máy hoàn chỉnh*. Từng chương trình đều có chuỗi các block cần thiết để hoàn tất một quá trình gia công. Chiều dài toàn bộ chương trình phụ thuộc vào tổng số block và kích cỡ của chúng.

CẤU TRÚC BLOCK

Trong block, số lượng từ ngữ được phép là tùy theo nhu cầu. Một số bộ điều khiển giới hạn số lượng ký tự trong một block. Giá trị tối đa chỉ có tính lý thuyết đối với Fanuc và các bộ điều khiển tương tự, nhưng hầu như không có ảnh hưởng trong thực tiễn. Hạn chế duy nhất là trong một block không được phép có *hai hoặc nhiều từ ngữ trùng lặp* (hàm hoặc lệnh) trừ các mã G. Ví dụ, chỉ một hàm M (có thể có ngoại lệ) hoặc chỉ một từ ngữ tọa độ đối với trục X là được phép trong một block. Thứ tự các từ ngữ trong block có thể hơi tự do, nghĩa là các từ ngữ có thể theo thứ tự bất kỳ với điều kiện block chuỗi thứ tự (địa chỉ N) được viết địa chỉ *trước*. Tuy thứ tự các từ ngữ riêng và trong block được phép theo thứ tự bất kỳ, nhưng tiêu chuẩn thực tế là đặt các từ ngữ trong block theo *thứ tự logic*, để chương trình CNC dễ đọc và dễ hiểu hơn.

Cấu trúc block chương trình phụ thuộc vào hệ thống điều khiển và kiểu máy CNC. Block có thể chứa các nhóm lệnh sau, sắp xếp theo thứ tự chung. Mỗi block không cần chuyên biệt mọi dữ liệu chương trình, chỉ chọn những dữ liệu được yêu cầu.

- ❑ Số block — N
- ❑ Các lệnh chuẩn bị — G
- ❑ Các hàm chung — M
- ❑ Các lệnh chuyển động trục — X Y Z A B C U V W...
- ❑ Từ ngữ liên quan đến các trục — J K R Q ...
- ❑ Tốc độ, lượng ăn dao, dụng cụ cắt S F T

Nội dung của block chương trình khác nhau tùy theo loại máy công cụ, nhưng về logic, chúng luôn luôn tuân theo các nguyên tắc chung, bất kể hệ thống CNC hoặc máy công cụ.

Xây dựng cấu trúc block

Mỗi block của chương trình CNC đều được xây dựng với cùng các suy nghĩ và sự cẩn thận như mọi kết cấu quan trọng khác, chẳng hạn nhà cửa, máy móc, xe hơi,... Điều này bắt đầu từ sự quy hoạch tỉ mỉ. Các quyết định cần được thực hiện là chọn những điều được đưa vào block chương trình, tương tự các cấu trúc khác. Ngoài ra, bạn cần quyết định thứ tự các lệnh – *nhóm lệnh* – được thiết lập trong block và các khảo sát khác.

Vài ví dụ kế tiếp sẽ so sánh giữa các cấu trúc block điển hình dùng trên máy phay và dùng trên máy tiện. Mỗi ví dụ sẽ trình bày một block.

Cấu trúc block trên máy phay

Trong các nguyên công phay, cấu trúc của block chương trình điển hình sẽ phản ánh các đặc tính thực tế của trung tâm gia công CNC.

➲ Các ví dụ block phay

```
N11 G43 Z2.0 S780 M03 H 01          (ví dụ 1)
N98 G01 X2.15 Y.575 F13.0           (ví dụ 2)
```

Ví dụ phay thứ nhất, block N11 là minh họa block bù chiều dài dao cắt, được sử dụng cùng với tốc độ trục chính và chiều quay trục chính.

Ví dụ thứ hai, block N98, là nhóm lệnh lập trình đối với chuyển động cắt tuyến tính đơn giản, sử dụng phương pháp nội suy tuyến tính và lượng ăn dao thích hợp.

➲ Các ví dụ block tiện

```
N67 G00 G42 X2.5 T0202 M08          (ví dụ 1)
N23 G02 X7.5 Z-2.8 R0.5 F0.012      (ví dụ 2)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại: http://www.me-cad.com/

Trong các ví dụ tiện, block N67 minh hoạ chuyển động nhanh đến vị trí GZ và vài lệnh khác, khởi động bù bán kính mũi dao G42, kích hoạt bù dụng cụ cắt (70202) và hàm M08 chất làm nguội ON. Block 23 có lệnh nội suy vòng (tròn) và lượng ăn dao.

NHẬN BIẾT CHƯƠNG TRÌNH

Chương trình CNC có thể được nhận biết theo chỉ số, và trên một số bộ điều khiển, chương trình được nhận biết theo tên. Sự nhận biết bằng chỉ số chương trình là cần thiết để lưu nhiều chương trình trong bộ nhớ CNC. Tên chương trình, nếu có, được dùng để mô tả tóm tắt về chương trình, có thể đọc trên màn hiển thị của bộ điều khiển.

Chỉ số chương trình

Block thứ nhất được dùng trong chương trình CNC bất kỳ thường là chỉ số chương trình, nếu được hệ thống điều khiển yêu cầu từ chương trình đó. Chỉ số chương trình có thể sử dụng một trong hai địa chỉ – chữ cái tiêu chuẩn O của định dạng EIA và dấu hai chấm [:] trong định dạng ASCII (ISO). Trong bộ nhớ, hệ thống điều khiển luôn luôn biểu thị chỉ số chương trình với chữ O. Block chứa chỉ số chương trình *không* nhất thiết phải có trong mọi chương trình CNC.

Nếu chương trình sử dụng các chỉ số, chúng phải được chuyên biệt, trong khoảng cho phép. Các chương trình điển hình của bộ điều khiển Fanuc phải trong khoảng 1-9999, chỉ số 0 (00 hoặc 000) là không được phép. Một số bộ điều khiển cho phép chỉ số có 5 chữ số. Dấu thập phân hoặc dấu âm cũng không được phép. Việc bớt hoặc thêm các số 0 là được phép, ví dụ 01, 001, 0001, 00001 đều hợp lệ, chỉ số chương trình là *một*.

Tên chương trình

Trên các hệ điều khiển Fanuc hiện đại, tên chương trình được *gộp chung* với chỉ số, nhưng *không thay thế* chỉ số chương trình. Tên chương trình (mô tả ngắn về chương trình) có thể dài đến 16 ký tự (kể cả khoảng trắng và ký hiệu). Tên chương trình phải trên *cùng một dòng* (trong cùng block) với chương trình.

```
O1001 (DWG. A-124D IT.2)
```

Tính năng này có ưu thế là khi thư mục nhớ hiển thị trên màn hình, tên chương trình xuất hiện kế bên chỉ số, giúp cho thư mục dễ hiểu và hữu dụng hơn. Sự mô tả chương trình có thể đọc được trên màn hình, làm cho sự nhận biết từng chương trình trở nên dễ dàng hơn.

Nếu tên chương trình dài hơn 16 ký tự, sẽ không có lỗi, nhưng chỉ hiển thị 16 ký tự đầu. Bạn nên tránh các tên chương trình có ý nghĩa không rõ ràng khi hiển thị trên màn hình. Bạn hãy xét hai tên chương trình dưới đây:

```
O1005 (LOWER SUPPORT ARM - OP 1)
O1006 (LOWER SUPPORT ARM - OP 2)
```

Do màn hình của bộ điều khiển chỉ có thể hiển thị *16* ký tự đầu của tên chương trình, các tên ví dụ nêu trên sẽ tối nghĩa khi hiển thị

```
O1005 (LOWER SUPPORT AR)
O1006 (LOWER SUPPORT AR)
```

Để giải quyết vấn đề này, bạn có thể dùng cách viết tắt trong phạm vi 16 ký tự kết hợp các dữ liệu cần thiết.

```
O1005 (LWR SUPP ARM - OP1)
O1006 (LWR SUPP ARM - OP2)
```

Nếu cần sự mô tả chi tiết hơn, bạn có thể xếp tên chương trình trên hai hoặc ba dòng:

```
O1005 (LWR SUPP ARM - OP1)
(OPERATION 1 - ROUGHING)
```

Nội dung trong một hoặc vài block tiếp sau chỉ số chương trình sẽ *không* xuất hiện trong danh sách thư mục trên màn hình, nhưng vẫn hữu ích đối với người vận hành CNC. Chúng sẽ hiển thị trong khi thực thi chương trình và trên bản in.

Bạn nên giữ các tên chương trình ngắn và súc tích. Công dụng của chúng là hỗ trợ người vận hành CNC tìm kiếm các chương trình lưu trong bộ nhớ của hệ thống điều khiển. Dữ liệu thích hợp trong tên chương trình là số bản vẽ, số chi tiết, tên chi tiết rút gọn, nguyên công,... Dữ liệu không thích hợp gồm tên máy, kiểu bộ điều khiển, tên nhà lập trình, tên công ty hoặc khách hàng,...

Trên nhiều bộ điều khiển, khi tải chương trình vào bộ nhớ, người vận hành CNC phải chuyên biệt chỉ số chương trình trên bảng điều khiển, bất kể chỉ số đó trong chương trình. Đây có thể đơn giản chỉ là số khả dụng trong hệ thống, hoặc là số có ý nghĩa đặc thù, biểu thị nhóm riêng (ví dụ, mọi chương trình bắt đầu với O10xx đều thuộc nhóm riêng cho một khách hàng hoặc đối tác). Các chương trình con phải được lưu với chỉ số do nhà lập trình chuyên biệt. Việc sử dụng các chỉ số chương trình còn giúp theo dõi chương trình được triển khai cho từng máy hoặc chi tiết gia công cụ thể.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Các block chuỗi thứ tự riêng rẽ trong chương trình CNC có thể có số quy chiếu để dễ định hướng trong chương trình. Địa chỉ chương trình với số block là chữ N, tiếp sau là một đến năm chữ số – từ chuỗi thứ tự 1 đến 9999 hoặc 99999, tùy theo hệ điều khiển, khoảng số block từ N1 đến N9999 trên các bộ điều khiển kiểu cũ hoặc đến N99999 trên các bộ điều khiển hiện đại. Một số bộ điều khiển cũ chỉ chấp nhận số block có tối đa 3 chữ số, N1 đến N999

Địa chỉ N luôn luôn là *từ ngữ thứ nhất* trong block. Để dễ định hướng trong chương trình có các chương trình con, không nên lặp lại các chỉ số giữa hai kiểu chương trình. Ví dụ, chương trình chính bắt đầu với N1 và chương trình con cũng bắt đầu với N1 có thể dẫn đến nhầm lẫn, nhưng không sai về kỹ thuật. Bạn hãy xem Chương 38 về ghi số các block trong chương trình con.

Lệnh số chuỗi thứ tự

Trong Bảng 9.1, cột thứ nhất biểu thị các số chuỗi thứ tự theo cách thức sử dụng bình thường, cột thứ hai là số chuỗi thứ tự theo định dạng được chấp nhận trong hệ thống điều khiển khi áp dụng cho chương trình CNC.

Bảng 9.1. *Số chuỗi thứ tự*

Số gia	Số block thứ nhất
1	N1
2	N2
5	N5
10	N10
50	N50
100	N100
99999	N99999

Sự sử dụng các chuỗi thứ tự (số block) trong chương trình CNC có nhiều ưu điểm và ít nhất là một nhược điểm.

Về ưu điểm, số block làm cho sự tìm kiếm chương trình trở nên đơn giản hơn khi biên tập hoặc lặp lại dụng cụ cắt trên máy. Chúng còn làm cho chương trình dễ đọc hơn trên màn hình CNC khi xử lý hoặc trên bản in. Điều này có nghĩa là cả *nhà lập trình* và *người vận hành* đều hưởng lợi.

Về nhược điểm, số block làm *giảm* dung lượng nhớ khả dụng trên máy CNC. Điều này có nghĩa là số lượng chương trình lưu trong bộ nhớ sẽ giảm và chương trình dài có thể không lưu trọn vẹn trong bộ nhớ.

Định dạng block thứ tự

Định dạng nhập chương trình đối với số block, sử dụng địa chỉ N, là N5 trên các bộ điều khiển hiện đại, N4 hoặc N3 trên các bộ điều khiển kiểu cũ. Số block N0 là không được phép, dấu trừ, dấu thập phân, số thập phân đều không được sử dụng. Số gia tối thiểu khi đánh số block luôn luôn là số nguyên nhỏ nhất được phép là một (N1, N2, N3, N4, ...). Số gia lớn hơn cũng được phép và sự lựa chọn tùy theo kinh nghiệm lập trình cá nhân hoặc theo tiêu chuẩn do công ty thiết lập. Các số gia thường dùng để đánh số block khác với số một được nêu trên Bảng 9.2.

Bảng 9.2. *Số gia khi đánh số block*

Số gia	Ví dụ trong chương trình
2	N2, N4, N6, N8, ...
5	N5, N10, N15, N20, ...
10	N10, N20, N30, N40, ...
100	N100, N200, N300, N400, ...

Một số nhà lập trình thường bắt đầu với mặc định N100, lập trình theo các số gia 100, 10, hoặc nhỏ hơn. Điều này tuy không sai, nhưng chương trình trở nên dài một cách không cần thiết và có thể khó quản lý hơn.

Trong tất cả các trường hợp số gia khác một, mục đích của chương trình là như nhau, nhằm cho phép các block bổ sung được chèn vào giữa các block hiện có, nếu cần thiết. Yêu cầu này có thể phát sinh trong khi kiểm chứng hoặc tối ưu hóa chương trình trên máy CNC, cần bổ sung cho chương trình hiện hữu. Tuy các block mới sẽ *không* theo thứ tự số gia như nhau, nhưng ít nhất cũng theo các số tăng dần. Ví dụ, vạt mặt trên máy tiện với một lần cắt (ví dụ A) được người vận hành chỉnh sửa thành hai lần cắt (ví dụ B).

➲ Ví dụ A – vạt mặt một lần cắt

```
N40 G00 G41 X3.5 Z0 T0303 M08
N50 G01 X-0.07 F0.01
N60 G00 W0.1 M09
N70 G40 X3.5
```

➲ Ví dụ B – vạt mặt hai lần cắt

```
N40 G00 G41 X3.5 Z0.05 T0303 M08
N50 G01 X-0.07 F0.01
N60 G00 W0.1
N61 X3.5
N62 Z0
N63 G01 X-0.07
N64 G00 W0.1 M09
N70 G40 X3.5
```

Bạn hãy chú ý sự thay đổi trong block N40 và các block bổ sung từ N61 đến N64. Trong

Nhiều tài liệu hay có tại : http://www.me-cad.com/

sách này sẽ lập trình block theo số gia là 1, các block có liên tiếp hay số đếm mặc định cần thiết khi bổ sung các block mới. Nhiều bộ điều khiển cho phép bỏ qua chỉ số block.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

➲ Ví dụ A – vạt mặt một lần cắt

```
N40 G00 G41 X3.5 Z0 T0303 M08
N41 G01 X-0.07 F0.01
N42 G00 W0.1
N43 G40 X3.5
```

➲ Ví dụ B – vạt mặt hai lần cắt

```
N40 G00 G41 X3.5 Z0.05 T0303 M08
N41 G01 X-0.07 F0.01
N42 G00 W0.1
X3.5
Z0
G01 X-0.07
G00 W0.1
N43 G40 X3.5
```

Chú ý, khi bỏ qua chỉ số block mới, chương trình sẽ được rút gọn, các bạn dễ dàng nhận biết block mới trên màn hình hoặc trên bản in.

Các chữ số 0 đứng trước có thể bỏ qua trong số block, ví dụ N00008 có thể viết thành N8. Việc bỏ qua các chữ số 0 đứng trước cho phép giảm chiều dài chương trình. Các chữ số 0 đứng sau không được phép bỏ qua, ví dụ N08 có thể viết là N8 nhưng N80 phải giữ nguyên.

Sử dụng số block trong chương trình là tùy chọn, như đã nêu trong các ví dụ. Chương trình chứa chỉ số block sẽ dễ đọc hơn. Đối với người vận hành CNC, tìm và chỉnh sửa các hàm trong chương trình cũng dễ dàng hơn. Bạn cần lưu ý, một số ứng dụng lập trình phụ thuộc vào chỉ số block. Ví dụ, các chu kỳ tiện lặp lại nhiều lần G70, G71, G72, G73. Trong trường hợp này, ít nhất các block quan trọng phải được đánh số (Chương 34).

Đánh số theo số gia

Các block trong chương trình có thể có thứ tự bất kỳ – tăng, giảm, hoặc phối hợp – chúng có thể được lặp lại hoặc có thể bỏ qua. Một số thực hành lập trình được ưa chuộng do có tính logic và dễ hiểu. Bạn không nên phối hợp thứ tự tăng và giảm, cũng không nên dùng các chỉ số lặp lại. Nếu chương trình có các số block lặp lại và cần tìm chỉ số block trên máy, hệ điều khiển sẽ tìm sự xảy ra thứ nhất của số block đó, có thể không phải là block cần tìm. Sự tìm kiếm tiếp theo sẽ lặp lại từ chuỗi đã tìm được trước đó. Lý do về sự rộng rãi chung trong đánh số block là cung cấp tính linh hoạt cho người vận hành CNC *sau khi* chương trình hoàn tất và được tải vào bộ điều khiển.

Số thứ tự block không ảnh hưởng đến thứ tự xử lý chương trình, bất kể số gia được sử dụng. Ngay cả khi các block được đánh số theo chiều giảm hoặc phối hợp tăng và giảm, chương trình gia công vẫn luôn luôn xử lý theo chuỗi thứ tự, dựa vào *nội dung* của block, không dựa vào chỉ số. Số gia 5 hoặc 10 được dùng nhiều trong thực tế do cho phép chèn thêm 4 hoặc 9 block giữa hai block hiện hữu bất kỳ, điều này là quá đủ cho hầu hết các chỉnh sửa chương trình.

Đối với các nhà lập trình CNC, sử dụng hệ thống lập trình trên cơ sở máy tính chỉ có vài từ ngữ liên quan đến lập trình của các số chuỗi thứ tự. Tuy hệ thống lập trình máy tính cho phép số khởi đầu của block và số gia có thể là tổ hợp bất kỳ, nhưng chủ yếu vẫn dùng *số bắt đầu và các số gia là 1* (N1, N2, N3,...). Mục đích của lập trình dựa trên máy tính là duy trì cơ sở dữ liệu chính xác của dạng hình học chi tiết gia công và quỹ đạo dao cắt. Nếu chương trình được chỉnh sửa bằng tay, cơ sở dữ liệu của chi tiết trên máy tính sẽ mất chính xác. Sự thay đổi chương trình CNC bất kỳ phải luôn luôn được phản ánh trong *mã nguồn* của chương trình và kết quả của thay đổi đó.

Chương trình dài và số block

Chương trình dài luôn luôn khó tải vào bộ nhớ CNC có dung lượng hạn chế. Trong trường hợp đó, chiều dài chương trình có thể được rút ngắn bằng cách bỏ qua các số block, hoặc chỉ lập trình chúng trong các *block quan trọng*. Block quan trọng là loại được đánh số nhằm tìm kiếm chương trình, sự lặp lại dụng cụ cắt, hoặc các quy trình khác tùy thuộc vào các số chương trình, chẳng hạn chu kỳ gia công hoặc thay dụng cụ cắt. Trong các trường hợp đó, bạn chọn số gia 2 hoặc 5 tùy theo sự thuận tiện của người vận hành CNC. Ngay cả sự sử dụng hạn chế các số block cũng làm tăng chiều dài chương trình, nhưng điều đó là cần thiết.

Nếu bỏ qua *tất cả* các số block trong chương trình, sự tìm kiếm trên bộ điều khiển sẽ trở nên khó khăn hơn. Người vận hành CNC không có lựa chọn nào khác ngoài việc tìm sự xảy ra kế tiếp của địa chỉ cụ thể bên trong block, chẳng hạn X, Y, Z... thay vì tìm số block. Phương pháp này có thể kéo dài thời gian tìm kiếm một cách không cần thiết.

KÝ TỰ KẾT THÚC BLOCK

Do các đặc tính kỹ thuật của hệ thống điều khiển, các block thứ tự phải được tách biệt bằng ký tự đặc biệt, được gọi là ký tự *end – of – block* (kết thúc block) được viết tắt là EOB. Trên hầu hết các hệ thống máy tính, ký tự

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

EOB được tạo ra bằng phím *Enter* trên bàn phím. Khi chương trình được nhập vào bộ điều khiển bằng MDI (Manual Data Input – nhập dữ liệu bằng tay), phím ký tự EOB trên bảng điều khiển sẽ kết thúc block. Ký hiệu kết thúc trên bộ điều khiển Fanuc là dấu chấm phẩy [;].

Ký hiệu dấu chấm phẩy trên màn hình chỉ là *biểu thị đồ họa* của ký tự kết thúc block, không được nhập vào chương trình CNC. Không có trường hợp nào ký hiệu này được gộp vào chương trình. Một số hệ điều khiển cũ dùng dấu sao [*] làm ký hiệu biểu thị kết thúc block thay vì dấu chấm phẩy [;]. Nhiều hệ điều khiển sử dụng các ký hiệu khác biểu thị kết thúc block, chẳng hạn dấu tiền tệ [$]. Trong mọi trường hợp, bạn đều cần nhớ ký hiệu chỉ là sự *biểu thị* ký tự kết thúc block, không phải là ký tự thực.

STARTUP BLOCK HOẶC SAFE BLOCK

Startup block (đôi khi được gọi là *safe block* hoặc *status block*) là block chuỗi thứ tự đặc biệt, chứa một hoặc vài từ ngữ chế độ (thường là các lệnh chuẩn bị của nhiều nhóm G) xác lập trước cho hệ thống điều khiển theo mặc định hoặc trạng thái ban đầu được yêu cầu. Block này được đặt ở đầu từng chương trình hoặc ở đầu từng dụng cụ cắt và là *block thứ nhất* khi lặp lại chương trình (hoặc dụng cụ cắt bên trong chương trình), block thay dao hoặc phân độ dao cắt. Đây là block cần tìm nếu chương trình hoặc dụng cụ cắt được lặp lại trong khi vận hành máy. Loại block này hơi khác giữa hệ thống tiện và hệ thống phay, do các yêu cầu đặc thù của từng hệ thống điều khiển.

Trong Chương 4 đã trình bày vấn đề trạng thái của hệ điều khiển khi được cấp điện, sẽ xác lập các điều kiện mặc định của hệ thống. Nhà lập trình CNC không nên tính đến các mặc định này, do người vận hành máy có thể dễ dàng thay đổi chúng mà nhà lập trình không biết. Nếu có sự thay đổi, các xác lập được lập trình sẽ *không* tương ứng với các mặc định do nhà chế tạo máy công cụ đề nghị hoặc các kỹ sư thiết kế hệ thống điều khiển.

Nhà lập trình CNC chuyên nghiệp phải luôn luôn áp dụng phương pháp lập trình *an toàn*, không nên dựa vào những điều chưa được biết chắc chắn. Nhà lập trình sẽ thử xác lập trước mọi điều kiện được yêu cầu dưới sự điều khiển của chương trình. Phương pháp này không chỉ an toàn hơn, mà còn làm cho chương trình dễ sử dụng hơn trong khi gá lắp, kiểm chứng quỹ đạo dao cắt và sự lặp lại dụng cụ cắt, điều chỉnh kích thước,... Điều này còn rất hữu ích đối với người vận hành máy CNC, đặc biệt là những người còn ít kinh nghiệm. Trong mọi ứng dụng nêu trên, startup block hoàn toàn không làm tăng thời gian chu kỳ, chỉ giúp cho sự chuyển đổi chương trình giữa các máy công cụ trở nên dễ dàng hơn, do không cần tính đến xác lập mặc định của tổ hợp bộ điều khiển máy công cụ.

Tên *safe block* – cách gọi khác của startup block, không làm cho block trở nên *an toàn* – block đó phải được *làm cho* an toàn. Bất kể tên gọi, block này phải chứa các xác lập điều khiển đối với chương trình hoặc dụng cụ cắt để khởi động chương trình trong trạng thái "sạch". Các mục nhập phổ biến xác lập trạng thái ban đầu là hệ thống lập kích thước (hệ Anh/hệ mét, tuyệt đối/tương đối), xóa chu kỳ hoạt động bất kỳ, xóa chế độ bù bán kính dao hoạt động, chọn mặt phẳng phay, chọn lượng ăn dao mặc định trên máy tiện, ... Các ví dụ được giới thiệu sẽ minh họa một số startup block cho cả điều khiển tiện và phay.

Ở đầu chương trình phay, startup block có thể được lập trình với nội dung như sau:

```
N1 G00 G17 G20 G40 G54 G64 G80 G90 G98
```

N1 là số chuỗi thứ tự đầu tiên, G00 chọn chế độ nhanh, G17 chọn mặt phẳng phay XY, G20 chọn đơn vị Anh, G40 xóa sự bù bán kính dao cắt hoạt động, G64 xác lập chế độ cắt liên tục, G80 xóa chu kỳ cố định hoạt động bất kỳ, G90 chọn chế độ tuyệt đối và G98 sẽ trở về mức ban đầu trong chu kỳ cố định. Các điều kiện này chỉ được áp dụng khi startup block được xử lý là *block chính đầu tiên* trong chương trình CNC, các thay đổi chương trình sau đó, nếu có, sẽ có hiệu lực chỉ với block trong đó thay đổi được áp dụng. Ví dụ, nếu G01 có hiệu lực theo mặc định sự sử dụng kế tiếp các lệnh G00, G02, hoặc G03 sẽ xóa lệnh G01.

Ở đầu chương trình tiện CNC, startup block có thể chứa các mã G dưới đây.

```
N1 G20 G00 G40 G99
```

N1 là số block thứ nhất, G20 chọn đơn vị Anh, G00 chọn chế độ chuyển động nhanh G40 xóa sự bù bán kính mũi dao hoạt động bất kỳ, và G99 chọn chế độ lượng ăn dao tính theo số vòng quay. Sự quy chiếu theo hệ thống tuyệt đối hay số gia (tương đối) là không cần thiết, do các bộ điều khiển tiện sử dụng các địa chỉ X và Y để lập kích thước *tuyệt đối*, các địa chỉ U và W để lập kích thước theo *số gia*. Đối với các bộ điều khiển tiện không hỗ trợ địa chỉ U và W, mã G91 tiêu chuẩn được dùng cho các giá trị số gia theo trục X và Y. Cũng như trong ví dụ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

phay, tiện) tại thời điểm được lập trình trong khối block có thể bị vượt qua (override) bằng sự thay đổi kế tiếp các lệnh G.

Một số hệ thống điều khiển không cho phép các mã G xác định trên cùng một dòng. Ví dụ, G20 hoặc G21 có thể không được lập trình với các mã G khác. Nếu bạn chưa chắc chắn, hãy đặt các mã G vào các block riêng. Thay vì:

```
N1 G20 G17 G40 G49 G80
```

hai hoặc nhiều block có thể được sử dụng một cách an toàn:

```
N1 G20
N2 G17 G40 G49 G80
```

CÁC CHÚ THÍCH CHƯƠNG TRÌNH

Nhiều loại chú thích và thông tin trong chương trình có thể được gộp vào thân chương trình trong các block riêng rẽ, hoặc như một phần trong block hiện hữu, chủ yếu khi thông báo là ngắn. Trong các trường hợp này, chú thích hoặc thông báo phải được đặt trong dấu ngoặc (đối với định dạng ASCII/ISO):

➲ **Ví dụ A**

```
N330 M00 (REVERSE PART)
```

➲ **Ví dụ B**

```
N330 M00 (REVERSE PART / CHECK TOOL)
```

➲ **Ví dụ C**

```
N330 M00 (REVERSE PART / CHECK TOOL)
```

Công dụng của thông báo hoặc chú thích trong chương trình là thông báo cho người vận hành máy về thao tác đặc biệt phải được thực hiện *mỗi khi* chương trình đạt đến giai đoạn xử lý khi thông báo xuất hiện. Chú thích rất hữu ích để hiểu chương trình và có thể được dùng để lập tài liệu cho chương trình.

Các thông báo và chú thích thường liên quan đến thông tin về thay đổi gá lắp, loại bỏ phoi ra khỏi lỗ, kiểm tra kích thước, kiểm tra điều kiện dao cắt ... Block chứa thông báo hoặc chú thích chỉ được gộp vào nếu thao tác được yêu cầu là *chưa rõ ràng* từ chính chương trình – không cần mô tả thao tác trong từng block. Các chú thích và thông báo phải ngắn và tập trung, do chúng chiếm không gian nhớ trong bộ nhớ CNC.

Theo quan điểm thực tiễn, chuỗi các block thông báo và chú thích có thể được cung cấp ở đầu chương trình, để liệt kê mọi thông tin quan trọng về bản vẽ và dụng cụ cắt cần thiết cho gia công (Chương 6); Ví dụ:

```
O1001 (SHAFT - DWG B451)
(SHAFT TOOLING - OP 1 - 3 JAW CHUCK)

(T01 - ROUGH TOOL - 1/32R - 80 DEG)
(T02 - FINISH TOOL - 1/32R - 55 DEG)
(T03 - OD GROOVING TOOL - 0.125 WIDE)
(T04 - OD THREADING TOOL - 60 DEG)

N1 G20 G99
N2 ...
```

Nếu không gian nhớ khả dụng của bộ điều khiển CNC bị hạn chế, sự sử dụng các block chú thích theo cách này có thể không thực tế. Sẽ tốt hơn nếu thông báo và chú thích được liệt kê trong bảng dụng cụ hoặc bảng xác lập, với mọi chi tiết cần thiết.

SỰ VA CHẠM TỪ NGỮ TRONG BLOCK

Nhóm lệnh trong block chương trình phải có tính logic và hợp lý. Ví dụ, block thứ nhất của chương trình có các từ ngữ:

```
N1 G20 G21 G17
```

Nội dung block này về logic là không thể chấp nhận, yêu cầu bộ điều khiển:

"Xác lập kích thước theo đơn vị Anh, đồng thời xác lập kích thước theo hệ mét, và xác lập mặt phẳng XY"

Rõ ràng là phi thực tế – điều gì thực sự xảy ra và bộ điều khiển diễn dịch block này như thế nào? Mặt phẳng XY là đúng, nhưng làm sao lựa chọn kích thước? Cả hai lựa chọn cùng một lúc là không thể, block này chứa các từ ngữ *va chạm* nhau, các đơn vị kích thước hoàn toàn không có thông báo lỗi. Điều gì sẽ xảy ra? Bộ điều khiển sẽ đánh giá block chuỗi thứ tự và kiểm tra các từ ngữ bất kỳ trong cùng một nhóm. Sự phân phối các nhóm lệnh đã đề cập trong phần lệnh chuẩn bị – mã G (Chương 7).

Nếu hệ thống máy tính tìm được hai hoặc nhiều từ ngữ thuộc cùng một nhóm, sẽ không đưa ra thông báo lỗi, nhưng sẽ tự động kích hoạt từ ngữ *cuối* của nhóm đó. Trong ví dụ về sự lựa chọn đơn vị kích thước bị va chạm nêu trên, đó sẽ là lệnh chuẩn bị G21 – chọn kích thước theo hệ mét – lệnh này được kích hoạt. Đây có thể là sự lựa chọn không được yêu cầu. Thay vì chờ đợi sự may mắn hiếm hoi, bạn hãy bảo đảm không có các từ ngữ va chạm trong block chương trình bất kỳ.

Trong ví dụ về lựa chọn hệ mét và hệ Anh, đã sử dụng lệnh chuẩn bị G. Điều gì sẽ xảy ra nếu, ví dụ, sử dụng địa chỉ X? Bạn hãy xét ví dụ dưới đây:

```
N120 G01 X11.774 X10.994 Y7.056 F15.0
```

Có hai địa chỉ X trong một block. Hệ điều khiển *sẽ không* chấp nhận giá trị X thứ hai, nhưng sẽ đưa ra thông báo lỗi. Tại sao? Do có sự khác biệt lớn giữa các nguyên tắc lập trình mã G và từ ngữ hệ tọa độ. Bộ điều khiển Fanuc

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

cho phép đặt số lượng mã G tùy ý trong một block nếu chúng không va chạm với nhau, nhưng không cho phép lập trình nhiều hơn một từ ngữ tọa độ có cùng địa chỉ đối với từng block chuỗi thứ tự. Một số nguyên tắc khác cũng có thể được áp dụng. Ví dụ, các từ ngữ trong block có thể được lập trình theo thứ tự bất kỳ với điều kiện địa chỉ N ở vị trí thứ nhất. Ví dụ, block sau đây là hợp lệ (nhưng theo thứ tự hơi bất thường).

```
N340 Z-0.75 Y11.56 F10.0 X6.845 G01
```

Để lập trình chuẩn xác, bạn cần viết các mục nhập cho từng block chuỗi thứ tự theo thứ tự logic. Số block phải là từ ngữ thứ nhất và thường tiếp theo là mã G, các trục chính theo thứ tự vần chữ cái (X..., Y..., Z...), các trục bổ sung hoặc các chỉnh sửa (I..., J..., K...), các hàm M và các từ ngữ, tốc độ cắt thường ở vị trí cuối cùng. Bạn chỉ nên chọn các từ ngữ thực sự cần thiết cho từng block:

```
N340 G01 X6.845 Y11.56 Z-0.75 F10.0
```

Ngoài ra còn có hai khả năng có thể đòi hỏi sự chú ý đặc biệt khi lập trình. Ví dụ, block dưới đây sẽ được diễn dịch như thế nào?

```
N150 G01 G90 X5.5 G91 Y7.7 F12.0
```

Dường như có va chạm giữa chế độ tuyệt đối và chế độ số gia (tương đối). Hầu hết các bộ điều khiển Fanuc đều sẽ xử lý block này theo đúng cách thức được viết. Vị trí đích của trục X sẽ đạt tới theo các giá trị tuyệt đối, nhưng trục Y sẽ là khoảng cách theo số gia, được đo từ vị trí hiện hành của dao cắt. Đây có thể không phải là cách tiếp cận phổ biến nhưng có ưu điểm trong một số trường hợp. Bạn cần nhớ – block tiếp sau block N150 sẽ theo chế độ *số gia*, do G91 được chuyên biệt *sau* lệnh G90.

Sự ứng dụng lập trình kế tiếp cần chú ý là block được lập trình trong chế độ nội suy đường tròn. Phần trình bày về chủ đề này (Chương 28) sẽ chuyên biệt cung hoặc đường tròn có thể được lập trình với bộ chỉnh sửa cung I, J, và K (tùy theo hệ điều khiển phay hay tiện được sử dụng). Chế độ này còn chuyên biệt mục nhập bán kính trực tiếp, sử dụng địa chỉ R. Cả hai ví dụ dưới đây đều đúng, kết quả sẽ là cung 90^0 với bán kính 1.5 inch:

➲ Với các bộ chỉnh sửa cung I và J

```
N21 G01 X15.35 Y11.348
N22 G02 X16.85 Y12.848 I1.5 J0
N23 G01 ...
```

➲ Với địa chỉ bán kính trực tiếp R

```
N21 G01 X15.35 Y11.348
N22 G02 X16.85 Y12.848 R1.5
N23 G01 ...
```

Bạn hãy khảo sát cách thức hệ thống điều khiển sẽ xử lý block N22 nếu chứa *cả hai*, các bộ chỉnh sửa I J và nhập bán kính R.

```
N22 G02 X16.85 Y12.848 J0 R1.5
```

hoặc

```
N22 G02 X16.85 Y12.848 R1.5 I1.5 J0
```

Câu trả lời có thể hơi lạ, trong *cả hai trường hợp*, bộ điều khiển sẽ bỏ qua các giá trị I và J, *chỉ xử lý giá trị bán kính R*. Trong trường hợp đặc biệt này, thứ tự định nghĩa địa chỉ là không quan trọng. Địa chỉ R có độ ưu tiên điều khiển cao hơn các địa chỉ I và J nếu được lập trình chung trong một block. Các ví dụ này đều giả thiết hệ điều khiển hỗ trợ sự nhập bán kính R.

CÁC GIÁ TRỊ LẬP TRÌNH CHẾ ĐỘ

Nhiều từ ngữ chương trình có tính chế độ, thuật ngữ *chế độ* xuất phát từ thuật ngữ "*mode*" có nghĩa là lệnh chuyên biệt vẫn trong chế độ đó sau khi đã được sử dụng trong chương trình. Lệnh này chỉ có thể xóa bằng lệnh chế độ khác trong cùng một nhóm. Không có tính năng này, chương trình sử dụng sự nội suy tuyến tính trong chế độ tuyệt đối với lượng ăn dao 18.0 in/min phải chứa lệnh tuyệt đối G90, lệnh chuyển động tuyến tính G01 và lượng ăn dao F18.0 trong từng block. Với các giá trị chế độ, sự xuất chương trình sẽ ngắn hơn nhiều. Hầu như mọi bộ điều khiển đều chấp nhận lệnh chế độ. Hai ví dụ dưới đây sẽ minh họa các khác biệt.

➲ Ví dụ A – không có các giá trị chế độ

```
N12 G90 G01 X1.5 Y3.4 F18.0
N13 G90 G01 X5.0 Y3.4 F18.0
N14 G90 G01 X5.0 Y6.5 F18.0
N15 G90 G01 X1.5 Y6.5 F18.0
N16 G90 G01 X1.5 Y3.4 F18.0
N17 G90 G00 X1.5 Y3.4 Z1.0
```

➲ Ví dụ B – có các giá trị chế độ

```
N12 G90 G01 X1.5 Y3.4 F18.0
N13 X5.0
N14 Y6.5
N15 X1.5
N16 Y3.4
N17 G00 Z1.0
```

Cả hai chế độ đều tạo ra các kết quả như nhau. Bạn hãy so sánh từng block của ví dụ A với block tương ứng trong ví dụ B. Bạn hãy quan sát, các lệnh chế độ là *không* cần lặp lại trong chương trình CNC. Quả vậy, trong lập trình hàng ngày, nhiều lệnh lập trình được sử dụng đều có tính chế độ. Ngoại lệ là các lệnh chương trình có chức năng khởi đầu và kết thúc trong block (ví dụ, trả về zero máy, thay dao, phân độ bàn máy...). Các hàm M có cách ứng xử

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tương tự. Ví dụ, nếu chương trình có thể về zero máy trong hai block liên tiếp (thường vì lý do an toàn), có thể tương tự như sau:

```
N83 G28 Z1.0 M09
N84 G28 X5.375 Y4.0 M05
```

Không thể loại bỏ lệnh G28 ra khỏi block N84, do lệnh G28 *không* có tính chế độ và cần phải thay thế.

TÍNH ƯU TIÊN THỰC THI

Các trường hợp đặc biệt đã được đề cập ở phần trên, trong đó *thứ tự* các lệnh trong block xác định tính ưu tiên thực thi các lệnh. Để kết thúc chủ đề về block, bạn hãy xét tình huống:

Hai block không liên hệ với nhau được dùng trong các ví dụ:

```
N410 G00 X22.0 Y34.6 S850 M03
```

và

```
N560 G00 Z5.0 M05
```

Trong block N410, chuyển động nhanh được lập trình cùng với hai lệnh trục chính. Điều gì *thực sự xảy ra* khi thực thi chương trình. Điều quan trọng cần biết là thời điểm trục chính được kích hoạt trong quan hệ với chuyển động dao cắt. Trong Fanuc và nhiều bộ điều khiển khác, hàm trục chính sẽ có hiệu lực *đồng thời* với chuyển động dao cắt.

Trong block N560, chuyển động dao cắt trên trục Z được lập trình (Z5.0), cùng thời điểm với hàm dừng trục chính (M05). Ở đây kết quả sẽ khác nhau, trục chính chỉ dừng khi chương trình *hoàn tất 100%* (Chương 8).

Tình huống tương tự cũng xảy ra với nhiều hàm M và nhà lập trình phải xác định chính xác cách thức máy và bộ điều khiển xử lý sự chuyển động kết hợp với địa chỉ hàm M trong một block. Các kết quả phổ biến được tóm tắt trong phần dưới đây.

Các hàm thực thi *đồng thời* với chuyển động dụng cụ cắt:

```
M03 M04 M07 M08
```

Các hàm sẽ được thực thi *sau khi* hoàn tất chuyển động dụng cụ cắt:

```
M00 M01 M05 M09 M98
```

Ở đây bạn cần cẩn thận, nếu có nghi ngờ, hãy lập trình một cách an toàn. Một số hàm M đòi hỏi điều kiện *bổ sung*, lệnh hoặc hàm khác hoạt động. Ví dụ M03 và M04 sẽ chỉ làm việc nếu hàm trục chính S có hiệu lực (trục chính đang quay). Các hàm M khác cần được lập trình trong các block riêng rẽ, chủ yếu vì lý do logic hoặc an toàn.

```
M02 M06 M10 M11 M19 M30 M60 M99
```

Các hàm biểu thị kết thúc chương trình hoặc chương trình con (M02, M30, M99) phải độc lập, không kết hợp với các hàm khác trong block trừ các trường hợp đặc biệt. Các hàm liên quan đến hoạt động cơ học của máy công cụ (M06, M10, M11, M19, M60) *cần* lập trình mà không có chuyển động, chủ yếu để bảo đảm an toàn. Trường hợp M19 (định hướng trục chính), sự quay của trục phải dừng trước, nếu không dừng quay máy có thể bị hư. Những ví dụ nêu trên chưa liệt kê tất cả các hàm M, nhưng có lẽ đủ để hiểu về cách thức chúng hoạt động khi được lập trình chung với chuyển động. Chương này đề cập đến các hàm M và trình bày thời hạn hiệu lực của các hàm điển hình trong block chương trình.

Để an toàn, bạn cần lập trình các hàm có thể gây va chạm này trong block chuỗi thứ tự không chứa chuyển động dụng cụ cắt. Đối với các hàm cơ học, cần xây dựng cấu trúc chương trình sao cho có thể cung cấp các điều kiện làm việc an toàn, các hàm này được định hướng chủ yếu đối với sự xác lập máy.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

10 NHẬP KÍCH THƯỚC

Các địa chỉ trong chương trình CNC liên quan đến vị trí dụng cụ cắt vào thời điểm xác định được gọi là *từ ngữ tọa độ*. Từ ngữ tọa độ luôn luôn lấy giá trị kích thước sử dụng đơn vị được chọn hiện hành, Anh hoặc mét. Các từ ngữ tọa độ gồm X, Y, Z, I, J, K, R, ... Chúng là cơ sở của mọi kích thước trong chương trình CNC. Hàng chục, hàng trăm, thậm chí hàng ngàn giá trị cần được tính toán để chương trình thực thi theo đúng yêu cầu – gia công chính xác chi tiết hoàn chỉnh.

Các kích thước trong chương trình gồm hai thuộc tính:

- *Đơn vị* kích thước – Anh hoặc mét
- *Quy chiếu* kích thước – tuyệt đối hoặc số gia (tương đối)

Đơn vị kích thước trong chương trình có thể là một trong hai hệ - *mét* hoặc *Anh*. Quy chiếu kích thước có thể theo *tuyệt đối* hoặc *số gia* (tương đối).

Các giá trị phân số, ví dụ 1/8, là không được phép trong chương trình CNC. Với hệ mét, *milimét* và *mét* được dùng làm đơn vị, hệ Anh dùng đơn vị là inch và feet. Bất kể hệ đơn vị được chọn, số lượng chữ số thập phân đều có thể được điều khiển, việc loại bỏ các số 0 (zero) đứng trước hoặc sau có thể được xác lập và dấu (chấm) thập phân có thể được lập trình hoặc bỏ qua, đều có thể áp dụng cho hệ thống CNC.

ĐƠN VỊ HỆ ANH VÀ HỆ MÉT

Kích thước bản vẽ có thể được dùng trong chương trình theo đơn vị *hệ Anh* hoặc *hệ mét*. Ở đây sẽ đưa ra các ví dụ sử dụng cả hệ Anh, phổ biến ở Hoa Kỳ và một vài quốc gia, và hệ mét, được dùng ở Châu Âu, Nhật và hầu hết các nước khác. Với thị trường toàn cầu, điều quan trọng là hiểu rõ cả hai hệ này. Sự sử dụng hệ mét đang tăng nhanh ở những nước vẫn còn dùng hệ Anh, chủ yếu là Hoa Kỳ.

Máy móc được trang bị bộ điều khiển Fanuc có thể lập trình theo cả hệ mét hoặc hệ Anh. Lựa chọn hệ thống CNC ban đầu (điều kiện *mặc định*) được điều khiển bằng xác lập tham số của hệ điều khiển, nhưng có thể vượt qua bằng lệnh chuẩn bị được viết trong chương trình gia công. Điều kiện mặc định thường do nhà chế tạo máy công cụ hoặc nhà phân phối xác lập, dựa trên các quyết định kỹ thuật của nhà chế tạo và nhu cầu của khách hàng.

Trong khi triển khai chương trình, cần khảo sát kỹ tác động của các điều kiện mặc định trên hệ thống điều khiển đối với sự thực thi chương trình. Điều kiện mặc định có hiệu lực vào thời điểm máy công cụ CNC *được cấp điện nguồn* (công tắc chính ON). Một khi có lệnh đưa ra trong chế độ MDI hoặc trong chương trình, giá trị mặc định có thể bị ghi đè và giữ giá trị mới kể từ thời điểm đó. Sự lựa chọn đơn vị kích thước trong chương trình CNC sẽ thay đổi giá trị *mặc định* (xác lập điều khiển nội). Nói cách khác, nếu đã chọn hệ Anh, hệ điều khiển sẽ duy trì đơn vị Anh cho đến khi nhập lệnh chọn hệ mét. Điều này có thể được thực hiện thông qua chế độ MDI, block chương trình, hoặc tham số hệ thống. Điều này áp dụng cho các các tình huống khi *tắt* công tắc chính (OFF) và sau đó *đóng* công tác chính (ON) cấp điện cho máy công cụ.

Để chọn nhập kích thước, bất kể các điều kiện mặc định, bạn cần có lệnh chuẩn bị G ở đầu chương trình CNC.

G20	Chọn đơn vị Anh (inch và feet)
G21	Chọn đơn vị mét (milimét và mét)

Nếu không chuyên biệt lệnh chuẩn bị trong chương trình, hệ thống điều khiển sẽ mặc định theo trạng thái của xác lập tham số hiện hành. Cả hai lựa chọn lệnh chuẩn bị này đều có tính chế độ, nghĩa là mã G được chọn sẽ hoạt động cho đến khi mã G đối tác được lập trình – hệ mét (G21) hoạt động cho đến khi được thay bằng hệ Anh (G20) và ngược lại.

Thực tế này có thể gợi ra khả năng tự do chuyển đổi giữa hai hệ đơn vị ở vị trí bất kỳ trong chương trình, hầu như ngẫu nhiên và vô hại. *Điều này là không đúng*. Mọi bộ điều khiển kể cả Fanuc đều dựa trên hệ mét, một phần do ảnh hưởng của Nhật, nhưng chủ yếu do hệ mét chính xác hơn. "Sự chuyển đổi" bất kỳ bằng cách dùng lệnh G20 hoặc G21 sẽ không tạo ra sự quy đổi thực giữa hai hệ đơn vị, chỉ dịch chuyển dấu thập phân mà không thay đổi các chữ số. Ví dụ, chọn G20 hoặc G21 sẽ đổi một đơn vị đo sang đơn vị khác trên *một số – không phải tất cả* – màn hình.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Hai ví dụ sau đây sẽ minh họa kết quả sai khi đổi G21 sang G20 và G20 sang G21 trong cùng một chương trình. Khi đọc giải thích cho từng block – bạn sẽ nhận thấy vài điều bất ngờ.

➲ Ví dụ 1 – từ hệ mét sang hệ Anh

G21	*Chọn đơn vị ban đầu (mét)*
G00 X60.0	*Giá trị X được chấp nhận là 60mm*
G20	*Giá trị trên sẽ đổi thành 6.0 inch (chuyển đổi thực là 60mm = 2.3622047 inch)*

➲ Ví dụ 2 - từ hệ Anh sang hệ mét

G20	*Chọn đơn vị ban đầu (Anh)*
G00 X6.0	*Giá trị X được chấp nhận là 6.0 inch*
G21	*Giá trị trên sẽ đổi thành 60mm (chuyển đổi thực là 6.0 inch = 152.4mm)*

Hai ví dụ này minh họa vấn đề có thể xảy ra khi chuyển đổi giữa hai hệ đơn vị kích thước trong một chương trình. Vì thế bạn chỉ nên dùng một đơn vị kích thước trong chương trình gia công. Nếu chương trình gọi chương trình con, bạn cũng phải bảo đảm nguyên tắc đó.

Không được dùng chung cả đơn vị Anh và đơn vị mét trong một chương trình

Thực tế, bạn không nên sử dụng chung hai hệ đơn vị, dù có thể dự đoán kết quả từ hệ điều khiển. Sự lựa chọn đơn vị kích thước sẽ có khác biệt lớn về cách thức hoạt động của một số hàm điều khiển. Các hàm sau đây sẽ bị tác động do thay đổi hệ đơn vị đo:

- Từ ngữ kích thước (các trục X, Y, Z, các bộ chỉnh sửa I, J, K,...)
- Tốc độ bề mặt không đổi (CSS – đối với tiện CNC)
- Hàm lượng ăn dao (địa chỉ F)
- Các giá trị bù (bù H và D khi phay, các giá trị dao cắt xác lập sẵn khi tiện)
- Hiển thị vị trí (số các chữ số thập phân)
- Bộ tạo xung bằng tay – HANDLE (giá trị độ chia)
- Một số tham số hệ thống điều khiển.

Lựa chọn đơn vị kích thước ban đầu còn có thể được thực hiện bằng cách xác lập tham số hệ thống. Trạng thái điều khiển khi công tắc chính (ON) là hoàn toàn như khi tắt công tắc (OFF) lần cuối. Nếu cả G20 và G21 đều không lập trình, bộ điều khiển sẽ chấp nhận đơn vị kích thước được chọn bằng xác lập tham số. Nếu có G20 hoặc G21 trong chương trình, *lệnh chương trình luôn luôn chiếm ưu tiên* trước xác lập tham số điều khiển bất kỳ. Nhà lập trình đưa ra quyết định, hệ điều khiển chỉ diễn dịch chúng, nhưng không có nghĩa là lúc nào cũng đúng.

Hãy cẩn trọng khi xác lập đơn vị trong block riêng, *trước* chuyển động trục, chọn lượng bù, hoặc xác lập tọa độ bất kỳ (G92, G50 và G54 đến G59). Vi phạm nguyên tắc này có thể dẫn đến các kết quả sai, đặc biệt khi thường xuyên thay đổi đơn vị đo cho các công việc gia công khác nhau.

Các giá trị đơn vị đo

Trong hệ Anh và hệ mét có nhiều đơn vị đo. Trong lập trình CNC chỉ sử dụng một số đơn vị. Các đơn vị hệ mét thường dùng là *milimét* hoặc *mét*, tùy theo ứng dụng. Đơn vị Anh là *inch* và *foot*, tùy theo ứng dụng. Viết tắt các đơn vị được nêu trên Bảng 10.1

Bảng 10.1. *Viết tắt các đơn vị đo*

Milimét	mm
Mét	m
Inch	in
Foot	ft

Nhiều thuật ngữ lập trình sử dụng các viết tắt trên Bảng 10.1. Bảng 10.2 nêu ra các đơn vị giữa hai hệ thống đo (các viết tắt đơn vị cũ được ghi trong dấu ngoặc).

Bảng 10.2. *Các đơn vị đo thông dụng*

Hệ mét	Hệ Anh
m/min (MPM)	ft/min (FPM hoặc SFPM)
mm/min	in/min (IPM hoặc ipm)
mm/vòng	in/rev (IPR hoặc ipr)
mm/răng	in/tooth (IPT hoặc ipt)
kW	HP

CÁC CHẾ ĐỘ TUYỆT ĐỐI VÀ SỐ GIA

Kích thước theo các đơn vị nhập (mét hoặc Anh) phải có *điểm quy chiếu chuyên biệt*. Ví dụ, nếu X35.0 xuất hiện trong chương trình và đơn vị được chọn là milimét, lệnh này không cho biết điểm gốc (quy chiếu) của kích thước 35 mm. Hệ điều khiển cần thêm thông tin để lập trình các giá trị kích thước một cách chính xác.

Có hai kiểu quy chiếu trong lập trình:

- Quy chiếu theo điểm chuẩn trên chi tiết
  ... còn được gọi là điểm gốc để nhập TUYỆT ĐỐI
- Quy chiếu theo điểm trước đó trên chi tiết
  ... được gọi là vị trí dao cuối cùng để nhập SỐ GIA

Trong ví dụ này, kích thước X35.0 (và mọi kích thước khác) có thể đo từ điểm cố định được chọn trên chi tiết, được gọi là *gốc*, *zero chương trình*, hoặc *điểm quy chiếu chương trình*, tất cả đều có cùng ý nghĩa. Giá trị X35.0 còn có thể đo

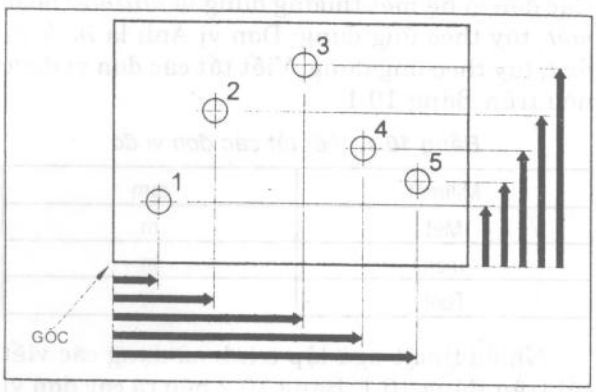
http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

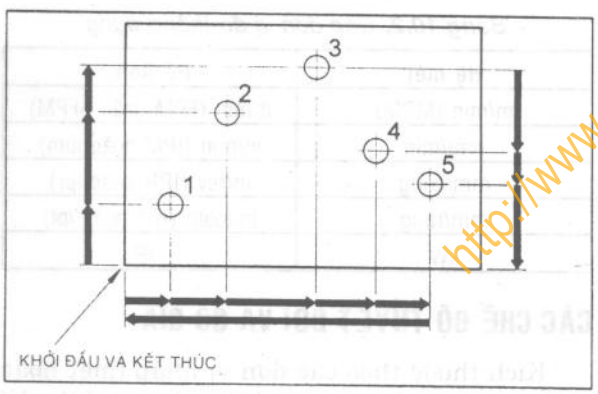
Nhiều tài liệu hay có tại : http://www.me-cad.com/

từ vị trí [illegible] cùng. Vị trí này trở thành vị trí *hiện hành* đối với chuyển động dao kế tiếp. Hệ điều khiển không thể phân biệt giữa hai khả năng này nếu chỉ có lệnh X35.0, cần bổ sung thêm lệnh vào chương trình.

Mọi kích thước trong chương trình CNC được đo từ *điểm chung* (gốc) đều là kích thước *tuyệt đối* (Hình 10.1) và mọi kích thước trong chương trình được đo từ *điểm hiện hành* (điểm cuối) đều là kích thước *số gia* (tương đối) (Hình 10.2)



Hình 10.1. *Kích thước tuyệt đối – đo từ gốc chi tiết, lệnh G90 trong chương trình.*



Hình 10.2. *Kích thước số gia – đo từ vị trí dao hiện hành, lệnh G91 trong chương trình.*

Kích thước tuyệt đối trong chương trình biểu thị các vị trí đích của dụng cụ cắt từ điểm gốc.

Kích thước số gia trong chương trình biểu thị chiều và khoảng cách thực của chuyển động dụng cụ cắt từ vị trí hiện hành.

Do địa chỉ X trong ví dụ nêu trên được viết là X35.0, được lập trình như nhau đối với cả hai hệ quy chiếu, cần có phương tiện bổ sung cho nhà lập trình. Không có phương tiện đó, hệ điều khiển sẽ sử dụng xác lập mặc định của tham số hệ thống. Không phải lúc nào cũng đúng với ý đồ của nhà lập trình. Sự lựa chọn chế độ kích thước được điều khiển bằng hai lệnh chế độ G.

Các lệnh chuẩn bị G90 và G91

Có hai lệnh chuẩn bị dùng để nhập các giá trị kích thước G90 và G91 để phân biệt giữa hai chế độ đơn vị.

G90	Chế độ lập kích thước tuyệt đối
G91	Chế độ lập kích thước theo số gia (tương đối)

Cả hai lệnh này đều có tính chế độ, do đó chúng sẽ xóa lẫn nhau. Hệ điều khiển sử dụng xác lập mặc định ban đầu khi đóng công tắc chính (ON) trên máy, thường là *chế độ số gia*. Xác lập này có thể thay đổi bằng tham số hệ thống xác lập trước trên máy tính khi cấp điện cho máy. Đối với các chương trình CNC riêng rẽ, xác lập hệ thống này có thể điều khiển bằng cách gộp lệnh chuẩn bị thích hợp vào chương trình, sử dụng một trong hai lệnh G90 hoặc G91.

Khi lập trình bạn cần gộp xác lập được yêu cầu trong chương trình CNC, không nên tính đến xác lập mặc định bất kỳ trong hệ điều khiển. Bạn có thể ngạc nhiên khi biết xác lập mặc định phổ biến trong hệ thống điều khiển là *chế độ số gia*, thay vì chế độ tuyệt đối. Nói chung, lập trình tuyệt đối có nhiều ưu điểm hơn lập trình số gia và được ưa chuộng hơn. Ngoài ra, nếu dùng lập trình số gia một cách thường xuyên, chương trình vẫn bắt đầu theo chế độ tuyệt đối. Vậy, *tại sao* chế độ số gia là mặc định? Lý do là, cũng như trong nhiều trường hợp mặc định khác, bảo đảm an toàn gia công. Bạn hãy theo dõi lập luận dưới đây:

Xét sự khởi động của chương trình mới được tải vào bộ điều khiển máy. Bộ điều khiển được kích hoạt, chi tiết được gá lắp an toàn, dụng cụ cắt ở vị trí chuẩn bị, xác lập các giá trị bù và chương trình sẵn sàng khởi động. Các chương trình loại này thường được viết với chế độ kích thước tuyệt đối. Mọi thứ có vẻ đều suôn sẻ, nhưng lệnh tuyệt đối G90 bị thiếu trong chương trình. *Điều gì sẽ xảy ra trên máy CNC?* Bạn hãy tự trả lời và suy nghĩ một cách logic.

Khi xử lý lệnh chuyển động dao cắt thứ nhất, các khả năng bao gồm giá trị đích của dao có thể là dương hoặc giá trị âm nhỏ. Do chế độ nhập kích thước bị thiếu trong chương trình, hệ điều khiển *"giả thiết"* chế độ này là số gia, đây cũng là giá trị mặc định của tham số hệ thống. Chuyển động dao cắt, thường chỉ theo trục X và trục Y, có thể vượt quá hành trình, nếu giá trị đích là dương, hoặc với lượng nhỏ, khi giá trị đích là âm. Trong các trường hợp này, hầu như không xảy ra sự cố đối với máy hoặc chi tiết gia công. Tất nhiên ở đây không

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

có sự bảo điều đặc biệt đối với các chương trình với ý thức bảo đảm tính an toàn gia công.

G91 là chế độ mặc định tiêu chuẩn để nhập kích thước.

Nhập dữ liệu tuyệt đối G90

Trong chế độ lập trình tuyệt đối, mọi kích thước đều được đo từ *điểm gốc* (chuẩn gia công). Điểm gốc là *điểm quy chiếu lập trình*, còn được gọi là *zero chương trình*. Chuyển động thực của máy là hiệu số giữa vị trí tuyệt đối hiện hành của dao và vị trí tuyệt đối trước đó. Các dấu đại số cộng (+) hoặc trừ (-) quy chiếu theo góc phần tư của hệ tọa độ, *không* theo chiều chuyển động. Dấu dương thường không được ghi cho địa chỉ bất kỳ. Mọi giá trị zero, chẳng hạn X0, Y0, Z0, đều quy chiếu theo vị trí dao tại điểm quy chiếu chương trình, không theo chuyển động dao. Bạn phải biết giá trị zero của trục bất kỳ, mỗi khi cần thiết.

Lệnh chuẩn bị G90 đã chọn cho chế độ tuyệt đối sẽ duy trì tính chế độ cho đến khi lệnh số gia G91 được lập trình. Trong chế độ tuyệt đối sự chuyển động trên trục bất kỳ đều không bị bỏ qua trong chương trình.

Ưu điểm chính của lập trình tuyệt đối là nhà lập trình hoặc người vận hành CNC dễ dàng chỉnh sửa. Sự thay đổi một kích thước không ảnh hưởng đến các kích thước còn lại trong chương trình.

Đối với các máy tiện CNC có bộ điều khiển Fanuc, sự biểu thị phổ biến của chế độ tuyệt đối là sự gán trục theo X và Z, mà *không* dùng lệnh G90. Một số máy tiện sử dụng lệnh G90, khi không có bộ điều khiển Fanuc.

Nhập dữ liệu số gia G91

Trong chế độ lập trình theo số gia, còn gọi là chế độ *tương đối*, mọi kích thước lập trình đều được đo theo các khoảng cách xuất phát với chiều đã chọn (tương đương với "*khoảng cách đến*" trên hệ thống điều khiển). Chuyển động thực của máy là lượng chuyên biệt theo từng trục, với chiều dương hoặc âm.

Các dấu cộng (+) hoặc (-) chuyên biệt *chiều* chuyển động dao, *không* dùng góc phần tư của hệ tọa độ. Không cần dùng dấu cộng cho giá trị dương, nhưng giá trị âm phải có dấu trừ. Mọi giá trị nhập zero, chẳng hạn X0, Y0, hoặc Z0 có nghĩa là *không có chuyển động dao dọc theo trục đó*, và không cần ghi. Nếu giá trị trục zero được lập trình trong chế độ số gia, giá trị đó sẽ bị bỏ qua. Lệnh chuẩn bị của chế độ số gia là G91 và giữ nguyên hiệu lực cho đến khi lệnh kỳ đều không được bỏ qua trong block chương trình.

Ưu điểm chính của chế độ số gia là tính chuyển đổi qua lại giữa các phần trong chương trình. Chương trình theo số gia có thể được gọi đến các vị trí khác nhau của chi tiết gia công, thậm chí trong các chương trình khác. Chế độ này được sử dụng chủ yếu khi viết các chương trình con hoặc lặp lại khoảng cách bằng nhau.

Đối với máy tiện CNC sử dụng bộ điều khiển Fanuc, sự biểu thị chế độ số gia thường là gán trục U và W, không dùng lệnh G91. Máy tiện CNC không dùng bộ điều khiển Fanuc, có thể sử dụng lệnh G91.

Phối hợp trong block

Trên nhiều bộ điều khiển Fanuc, chế độ tuyệt đối và số gia có thể kết hợp trong một block chương trình để đáp ứng các mục đích lập trình đặc biệt. Điều này dường như hơi bất thường, nhưng thực tế có các lợi ích rõ rệt trong ứng dụng cao cấp này. Nói chung, chương trình thường chỉ có một chế độ kích thước – hoặc chế độ tuyệt đối hoặc chế độ số gia. Trên nhiều bộ điều khiển, đối với sự chuyển đổi bất kỳ giữa hai chế độ, lệnh chuyển động phải được lập trình trong block riêng. Các bộ điều khiển đó, ví dụ, không cho phép lập trình chuyển động số gia dọc theo một trục và chuyển động tuyệt đối dọc theo trục khác trong cùng một block.

Hầu hết các bộ điều khiển Fanuc đều cho phép lập trình cả hai chế độ *trong một block*. Toàn bộ điều cần thực hiện là chuyên biệt lệnh chuẩn bị G90 hoặc G91 *trước* địa chỉ kích thước quan trọng.

Đối với máy tiện, không sử dụng các lệnh G90 và G91, sự chuyển đổi được thực hiện giữa các cặp trục X và U, Z và W. Các trục X và Z chứa giá trị tuyệt đối, U và W là các giá trị số gia. Cả hai kiểu đều có thể viết trong một block mà không gây ra vấn đề. Dưới đây là các ví dụ điển hình của cả hai ứng dụng.

➲ **Ví dụ phay:**

```
N68 G01 G90 X12.5037 G91 Y4.5177 F18.5
```

Ví dụ phay đưa ra chuyển động nơi dao cắt đạt đến vị trí tuyệt đối X12.5037 inch *đồng thời* chuyển động dọc thec trục Y với khoảng cách 4.5177 inch theo chiều dương. Bạn hãy chú ý vị trí các lệnh G90 và G91 trong block đây là điều rất quan trọng, nhưng có thể không được phép trên một số bộ điều khiển.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

➲ Ví dụ tiện:

```
N60 G01 X13.56 W-2.5 F0.013
```

Block này biểu thị quỹ đạo của dao cắt, trong đó dao cắt đi đến đường kính 13.56 inch và *đồng thời* chuyển động 2.5 inch, theo chiều âm trên trục Z, biểu thị bằng địa chỉ gán số gia W. Lệnh G90 hoặc G91 nói chung không được sử dụng, do *Nhóm A* của các mã G là loại phổ biến nhất nhưng không hỗ trợ sự gán mã G để chọn chế độ kích thước.

Thời điểm bất kỳ chuyển đổi giữa chế độ tuyệt đối và chế độ số gia trong chương trình, nhà lập trình phải cẩn thận, không duy trì chế độ *"cũ"* lâu hơn so với yêu cầu. Sự chuyển đổi giữa hai chế độ thường chỉ có tính tạm thời, nhằm mục đích đặc biệt, có thể tác động đến một hoặc vài block. Bạn cần bảo đảm khôi phục lại xác lập gốc cho chương trình. Bạn cần nhớ, cả hai lệnh này đều có *tính chế độ* – chúng duy trì hiệu lực cho đến khi bị xóa bằng chế độ còn lại.

LẬP TRÌNH ĐƯỜNG KÍNH

Mọi kích thước dọc theo trục X trên máy tiện CNC đều có thể được lập trình theo giá trị đường kính. Phương pháp này đơn giản hóa sự lập trình tiện và giúp cho chương trình dễ đọc hơn. Nói chung, mặc định của hầu hết các bộ điều khiển Fanuc đều là *lập trình đường kính*. Tham số hệ thống điều khiển này có thể thay đổi để diễn dịch trục X theo giá trị nhập bán kính.

```
G00 X4.0
```
Kích thước đường kính, khi xác lập bằng tham số.

```
G00 X2.0
```
Kích thước bán kính, khi xác lập bằng tham số.

Cả hai giá trị đều đúng, với xác lập tham số thích hợp. Lập trình đường kính là dễ hiểu hơn đối với cả nhà lập trình và người vận hành, do bản vẽ sử dụng đường kính cho các chi tiết hình trụ và thường đo đường kính ngay trên máy. Bạn cần đặc biệt chú ý – nếu sử dụng lập trình đường kính, mọi giá trị bù sự mòn dao đối với trục X phải được xử lý theo *đường kính chi tiết* gia công, thay vì bán kính.

Trong chế độ số gia, chuyển động cố ý theo trục X sẽ được lập trình trên trục U, chuyên biệt khoảng cách và chiều hành trình theo *đường kính*.

Ví dụ, hai phần của các chương trình theo hệ mét dưới đây là đồng nhất với nhau, bạn hãy chú ý chúng đều *bắt đầu* với chế độ tuyệt đối và chỉ có các đường kính xuất hiện khác nhau.

➲ Ví dụ 1 – Đường kính tuyệt đối

```
G00 G42 X85.0 Z2.0 T0404 M08 (ABSOLUTE START)
G01 Z-24.0 F0.3
X95.0
Z-40.0
X112.0
Z-112.0
X116.0
G00 ..
```

➲ Ví dụ 2 – đường kính số gia

```
G00 G42 X85.0 Z2.0 T0404 M08 (ABSOLUTE START)
G01 Z-24.0 F0.3
U10.0                          (X95.0)
Z-40.0
U17.0                          (X112.0)
Z-120.0
U4.0                           (X116.0)
G00 ..
```

SỐ GIA CHUYỂN ĐỘNG TỐI THIỂU

Số gia cực tiểu (số gia tối thiểu) là *lượng chuyển động trục nhỏ nhất*, bộ điều khiển có khả năng hỗ trợ. Số gia tối thiểu là đại lượng nhỏ nhất có thể lập trình trong chế độ nhập kích thước được chọn. Tùy theo lựa chọn nhập kích thước, số gia chuyển động trục tối thiểu sẽ tính theo *milimet* trong hệ mét hoặc *inch* theo hệ Anh.

Hệ đơn vị	Số gia tối thiểu
Mét	0.001 mm
Anh	.0001 inch

Trong định nghĩa số gia tối thiểu, các số gia phổ biến nhất là 0.001 mm đối với hệ mét và hệ Anh là 0.0001 inch. Đối với máy tiện CNC số gia tối thiểu đối với trục X vẫn là 0.001 mm hoặc 0.0001 inch nhưng được đo trên *đường kính* – nghĩa là 0.00005 inch trên mỗi phía (bán kính). Sự tinh chỉnh về độ chính xác gia công trên hệ mét linh hoạt hơn và chính xác hơn so với hệ Anh.

Số gia nhỏ nhất	Chuyển đổi tương đương
0.001 mm	.00003947 inch
.0001 inch	0.00254 mm

Đối với chi tiết đòi hỏi độ chính xác cao, hệ mét được ưa thích hơn trong lập trình gia công. Hệ mét *chính xác hơn 154%* so với hệ Anh, còn hệ Anh có *độ chính xác thấp hơn 60.63%* so với hệ mét.

ĐỊNH DẠNG NHẬP KÍCH THƯỚC

Năm 1959 được coi là năm đầu tiên áp dụng điều khiển số trong thực tế. Từ thời kỳ đó, nhiều thay đổi đã diễn ra, ảnh hưởng đến sự định dạng lập trình nhập kích thước.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ngày nay, dữ liệu kích thước có thể lập trình theo một trong bốn phương pháp:

- Định dạng địa chỉ toàn phần
- Bỏ các số zero đứng trước.
- Bỏ các số zero đứng sau.
- Dấu chấm thập phân.

Để hiểu các khác biệt định dạng, cần xem xét lại quá khứ. Các hệ điều khiển cũ (chủ yếu là NC so với các hệ CNC hiện tại) không có khả năng chấp nhận mức nhập kích thước cao nhất – định dạng dấu thập phân – còn các bộ điều khiển mới nhất đều chấp nhận *mọi* định dạng chương trình cũ kể cả khi định dạng thập phân trở nên phổ biến. Lý do là tính tương thích với các chương trình đã có (chương trình cũ). Do phương pháp lập trình dấu thập phân là mới nhất trong số 4 phương pháp, hệ điều khiển cho phép lập trình dấu thập phân cũng có thể chấp nhận các chương trình đã viết từ nhiều năm trước (với giả thiết cả bộ điều khiển và máy công cụ cũng tương thích), nhưng *điều ngược lại là không đúng*.

Đây cũng là vấn đề rất quan trọng, do hiểu biết về phương pháp bộ điều khiển diễn dịch *số không có dấu thập phân* là điều cơ bản đối với mọi lệnh chuyển động dao và lượng ăn dao.

Định dạng địa chỉ toàn phần

Định dạng toàn phần của địa chỉ kích thước được quy định bằng chuẩn +44 trong hệ Anh và +53 trong hệ mét. Điều đó có nghĩa là cả 8 chữ số khả dụng đều được viết cho các từ ngữ trục X, Y, Z, I, J, K,... Ví dụ, kích thước theo hệ Anh .625 khi áp dụng cho trục X, sẽ được viết như sau:

```
X00006250
```

Kích thước hệ mét 0.42 mm, áp dụng cho trục X được viết là:

```
X00000420
```

Lập trình định dạng toàn phần cho các bộ điều khiển *thời kỳ đầu*, nhưng cho tới nay vẫn đúng. Trục được lập trình thường được ghi *không* cần sự gán trục, được xác định bằng *vị trí* của kích thước trong block. Đối với lập trình CNC hiện đại, định dạng toàn phần đã lạc hậu và được nêu ra ở đây chỉ để tham khảo và so sánh. Định dạng này vẫn hoạt động tốt trong các chương trình hiện đại, nhưng không được dùng làm tiêu chuẩn

Bãi bỏ các số zero

Phương pháp bãi bỏ các số zero là sự cải tiến lớn so với định dạng địa chỉ đầy đủ. Đây là sự thích ứng của định dạng mới, giảm số lượng các số zero khi nhập kích thước. Nhiều bộ điều khiển hiện đại vẫn hỗ trợ phương pháp này nhưng chỉ vì lý do tương thích với các chương trình cũ và đã được kiểm chứng.

Sự bãi bỏ zero có nghĩa là không cần viết các số zero đứng *trước* hoặc đứng *sau* của số nhập kích thước cực đại trong chương trình CNC. Kết quả là giảm rõ rệt chiều dài chương trình. Xác lập mặc định đã được nhà chế tạo bộ điều khiển xác định, nhưng chế độ mặc định này có thể được xác lập lại bằng tham số hệ thống. *Bạn không nên thực hiện các thay đổi bất kỳ nếu không có lý do chính đáng.*

Do sự bãi bỏ các số zero đứng *trước* và bãi bỏ các số zero đứng *sau* là loại trừ lẫn nhau. Vậy loại nào cần có thể lập trình cho các địa chỉ mà không có dấu thập phân? Do điều này lệ thuộc vào xác lập tham số của hệ thống điều khiển hoặc sự gán trạng thái do nhà chế tạo bộ điều khiển thực hiện, bạn cần biết trạng thái điều khiển thực. Trạng thái này xác định có thể bãi bỏ các số zero nào, có thể lập các số zero ở *đầu* hoặc các số zero ở *cuối* giá trị kích thước mà không có dấu thập phân. Trong trường hợp đặc biệt, khi hệ thống CNC được trang bị với tính năng bãi bỏ số zero chỉ theo một chế độ, sẽ không thể lập trình với dấu thập phân. Để minh họa kết quả bãi bỏ zero, bạn hãy xem các ví dụ đã nêu.

Nếu kích thước Anh .625 inch được lập trình theo định dạng *bỏ các số zero đứng trước* và áp dụng cho trục X, giá trị này xuất hiện trong chương trình như sau:

```
X6250
```

Kích thước .625 inch với *sự bãi bỏ các số zero đứng sau*, sẽ xuất hiện như sau:

```
X0000625
```

Kích thước hệ mét 0.42 mm, cũng áp dụng cho trục X, được viết khi bãi bỏ các số zero đứng trước, sẽ như sau:

```
X420
```

Kích thước này được viết khi bãi bỏ các số zero đứng sau:

```
X0000042
```

Tuy các ví dụ nêu trên chỉ minh họa một ứng dụng nhỏ, nhưng cũng có thể nhận thấy sự bãi bỏ các số zero đứng trước có tính thực tiễn hơn sự bãi bỏ các số zero đứng sau. Nhiều hệ điều khiển cũ được xác lập để chấp nhận *mặc định là sự bãi bỏ các số zero đứng trước* do tính thực tiễn của định dạng này. Đây là lý do

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

chính, bạn nên nghiên cứu cẩn thận đối ngày nay vấn đề này có vẻ không đáng kể. Nhưng mặt khác, nếu quên một dấu thập phân trong chương trình, kiến thức này trở nên rất hữu ích.

Ưu điểm của bãi bỏ các số zero đứng trước

Giá trị nhập kích thước nhỏ nhất và lớn nhất, hệ điều khiển có thể chấp nhận, gồm 8 chữ số, không có dấu thập phân, trong khoảng 0000001 đến 99999999:

- Minimum 0000.0001 inch hoặc 00000.001 mm
- Maximum 9999.9999 inch hoặc 99999.999 mm

Ở đây không cần viết dấu thập phân. Nếu chương trình sử dụng định dạng bỏ các số zero sự so sánh các giá trị nhập sẽ rất hữu ích.

Bảng 10.3. *So sánh giá trị nhập – inch*

Dấu thập phân	Bãi bỏ các số zero đứng trước	Bãi bỏ các số zero đứng sau
X0.0001	X1	X00000001
X0.001	X10	X0000001
X0.01	X100	X000001
X0.1	X1000	X00001
X1.0	X10000	X0001
X10.0	X100000	X001
X100.0	X1000000	X01
X1000.0	X10000000	X1

Sự bãi bỏ số zero đứng trước được ưa chuộng hơn, do có lợi hơn với các phần thập phân nhỏ so với phần số nguyên lớn.

Bảng 10.4. *So sánh giá trị nhập – mm*

Dấu thập phân	Bãi bỏ các số zero đứng trước	Bãi bỏ các số zero đứng sau
X0.001	X1	X00000001
X0.01	X10	X0000001
X0.1	X100	X000001
X1.0	X1000	X00001
X10.0	X10000	X0001
X100.0	X100000	X001
X1000.0	X1000000	X01
X10000.0	X10000000	X1

Ngay cả khi các chương trình chỉ sử dụng định dạng dấu thập phân, hiểu biết về tác dụng bãi bỏ các số zero vẫn quan trọng. Ví dụ, điều gì sẽ xảy ra nếu nhà lập trình quên lập trình dấu thập phân? Đây sẽ là lỗi nghiêm trọng, và *thường xảy ra*, nhưng có thể tránh được dễ dàng bằng kiến thức và sự cẩn thận.

Để hoàn tất phần định dạng bãi bỏ các số zero, bạn hãy xét giá trị nhập chương trình, sử dụng chữ cái trục nhưng *không phải là từ ngữ tọa độ*. Lệnh dwell sẽ được dùng để giải thích vấn đề này. Chương 23 trình bày mọi chi tiết về lập trình dwell. Phần này chỉ sử dụng định dạng cơ bản và thời gian dwell một giây làm đơn vị. Định dạng dwell có thể được lập trình với trục X, tiếp sau tối đa là 8 chữ số, luôn luôn dương. Nếu hệ điều khiển cho phép dấu thập phân, ở đây cũng không có sự nhầm lẫn. Nếu cần bãi bỏ các số zero đứng trước hoặc đứng sau, giá trị nhập được lập trình là rất quan trọng.

Ví dụ, chương trình đòi hỏi dwell kéo dài 0.5 giây. Trong các định dạng dưới đây, block chứa *½ giây* dwell sẽ là:

- Định dạng đầy đủ X0000050
- Bỏ các zero đứng trước X500
- Bỏ các zero đứng sau X000005
- Dấu thập phân X0.5 hoặc X.5

Chú ý, logic định dạng dwell và các từ ngữ tọa độ là như nhau. Định dạng được lập trình luôn luôn quan hệ chặt chẽ với sự ghi địa chỉ. Trong một số chu kỳ cố định, dwell được biểu thị bằng địa chỉ P, hoàn toàn không dùng dấu thập phân và phải được lập trình với chế độ bỏ zero phía trước. Nửa giây sẽ bằng P500

Lập trình dấu thập phân

Mọi lập trình hiện đại đều sử dụng dấu thập phân khi nhập kích thước. Lập trình dấu thập phân, đặc biệt đối với dữ liệu chương trình đòi hỏi phần thập phân, giúp cho chương trình CNC dễ triển khai hơn và sau này dễ đọc hơn.

Từ mọi địa chỉ lập trình khả dụng, không phải tất cả địa chỉ đều có thể lập trình với dấu thập phân. Những địa chỉ có thể lập trình với dấu thập phân là loại chuyên biệt dữ liệu theo inch, milimét, hoặc giây (trừ một số ngoại lệ).

Dưới đây là hai nhóm địa chỉ với dấu thập phân được phép trong các chương trình điều khiển phay và điều khiển tiện:

- **Các chương trình điều khiển phay**

  X, Y, Z, I, J, K, A, B, C, Q, R

- **Các chương trình điều khiển tiện**

  X, Z, U, W, I, K, R, C, E, F

Hệ điều khiển hỗ trợ tùy chọn lập trình dấu thập phân, còn có thể chấp nhận các giá trị kích thước *không có* dấu thập phân, để tương thích với các chương trình cũ. Trong trường hợp đó, điều quan trọng là hiểu các nguyên tắc định dạng lập trình sử dụng các số zero đứng trước và đứng sau. Nếu chúng được sử dụng một

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

cách chính xác, sẽ không xảy ra vấn đề khi áp dụng các định dạng kích thước khác nhau cho hệ điều khiển bất kỳ, cũ hoặc mới. Nếu có thể, bạn hãy viết chương trình với dấu thập phân là phương pháp tiêu chuẩn.

Tính tương thích này cho phép những người dùng kỳ cựu có thể tải các chương trình cũ của họ (thường với định dạng băng từ) vào các bộ điều khiển CNC mới, nhưng *không có* phương pháp khác, thường với vài chỉnh sửa nhỏ hoặc hoàn toàn không chỉnh sửa.

Một số bộ CNC hiện đại không có khả năng tiếp nhận băng từ do không có đầu đọc băng. Để chuyển đổi các băng từ, chứa các chương trình còn tốt, có hai tùy chọn, thứ nhất là lắp bộ đọc băng từ vào bộ điều khiển, nếu có thể, và thứ hai là lưu nội dung băng từ vào bộ nhớ của máy tính. Phương pháp này có chi phí rất thấp và có các tùy chọn lưu giữ tốt hơn so với băng từ. Với phần mềm thích hợp và đầu đọc băng từ xách tay, bạn dễ dàng thực hiện điều này.

Dữ liệu kích thước theo hệ mét có số gia nhỏ nhất là 0.001mm, hệ Anh là .0001 inch (định dạng bỏ zero đứng trước là mặc định):

`Y12.56` *là* `Y125600` *đối với hệ Anh*

`Y12.56` *là* `Y12560` *đối với hệ mét*

Các giá trị lập trình không hoặc có dấu thập phân có thể phối hợp với nhau trong một block.

```
N230 X4.0 Y-10
```

Điều này có thể rất có lợi khi cần tiết kiệm dung lượng nhớ. Ví dụ X4.0 đòi hỏi ít ký tự hơn so với X40000, mặt khác Y-10 ngắn hơn so với Y-0.001 (cả hai ví dụ đều tính theo đơn vị Anh). Nếu mọi chữ số *trước* hoặc *sau* dấu thập phân đều là zero, bạn không cần viết các số zero đó:

```
X0.5    =  X.5
Y40.0   =  Y40.
Z-0.1   =  Z-.1
F12.0   =  F12.
R0.125  =  R.125
```

Giá trị zero bất kỳ đều *phải được viết* – ví dụ: X0 không được viết là X. Trong sách này, mọi ví dụ chương trình đều sử dụng định dạng dấu thập phân, mỗi khi có thể. Nhiều nhà lập trình thích lập trình với số zero như cột bên trái của ví dụ nêu trên. Họ bổ sung thêm vài ký tự vào bộ nhớ hệ thống, nhưng chương trình dễ đọc hơn.

So sánh giá trị nhập

Bạn có thể dễ dàng nhận thấy sự khác biệt trong định dạng nhập giữa chế độ lập kích thước theo hệ Anh và hệ mét. Dưới đây là các ví dụ.

➲ **Hệ Anh – nhập .625 inch**

Định dạng đầy đủ	X00006250
Không có các zero đứng trước	X6250
Không có các zero đứng sau	X0000625
Dấu thập phân	X0.625 hoặc X.625

➲ **Hệ mét nhập 0.42 mm**

Định dạng đầy đủ	X00000420
Không có các zero đứng trước	X420
Không có các zero đứng sau	X0000042
Dấu thập phân	X0.42 hoặc X.42

NHẬP KIỂU CALCULATOR

Trong một số ngành công nghiệp, chẳng hạn gia công gỗ, đa số kích thước (đặc biệt là hệ mét) không đòi hỏi phần thập phân, chỉ dùng số nguyên. Trong các trường hợp đó, sau dấu thập phân phải có chữ số zero. Fanuc cung cấp giải pháp cho vấn đề này bằng tính năng *calculator input*. Sử dụng tính năng này cũng có thể rút ngắn chương trình.

Nhập kiểu calculator đòi hỏi xác lập tham số hệ thống. Khi tham số này được xác lập, không cần ghi dấu thập phân và số zero đứng sau. Ví dụ, X25 được diễn dịch là X25.0 thay vì X0.0025

Nếu giá trị nhập yêu cầu dấu thập phân, bạn có thể viết như bình thường, điều đó có nghĩa là giá trị với dấu thập phân sẽ được diễn dịch một cách chính xác và các số không có dấu thập phân sẽ được xử lý như các đơn vị chính (inch hoặc mm). Dưới đây là vài ví dụ

Nhập tiêu chuẩn	Calculator Input
X345.0	X345
X1.0	X1
Y0.67	Y0.67
Z7.48	Z7.48

Nói chung, hệ thống điều khiển được xác lập theo chế độ bỏ các chữ số zero đứng trước và các giá trị không phải là thập phân được diễn dịch theo số các đơn vị nhỏ nhất. Ví dụ, Z1000 trong chế độ G21 sẽ tương đương với Z1.0 (mm).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

11 ĐIỀU KHIỂN TRỤC CHÍNH

Cả hai kiểu máy CNC, trung tâm gia công và tiện, đều sử dụng sự quay trục chính khi cắt gọt vật liệu kim loại. Sự quay này có thể là của dụng cụ cắt (phay) hoặc chi tiết gia công (tiện). Trong cả hai trường hợp, hoạt động của trục chính và tốc độ ăn dao của dụng cụ cắt cần được điều khiển chặt chẽ bằng chương trình CNC. Mấy CNC đòi hỏi các lệnh liên quan đến sự lựa chọn *tốc độ* thích hợp của trục chính và *tốc độ ăn dao* của nguyên công cụ thể.

Có nhiều phương pháp điều khiển trục chính và tốc độ cắt, chủ yếu phụ thuộc vào kiểu máy CNC và ứng dụng gia công. Chương này sẽ trình bày *sự điều khiển trục chính* và các ứng dụng lập trình.

HÀM TRỤC CHÍNH

Lệnh chương trình liên quan đến tốc độ trục chính được điều khiển trong hệ thống CNC bằng địa chỉ S. Định dạng lập trình của địa chỉ S thường trong khoảng 1 đến 9999 và không có dấu thập phân:

`S1` *đến* `S9999`

Đối với nhiều máy CNC tốc độ cao, tốc độ trục chính có thể đến 5 chữ số, trong khoảng địa chỉ S:

`S1` *đến* `S99999`

Khoảng tốc độ trục chính cực đại khả dụng trong bộ điều khiển phải luôn luôn *lớn hơn* khoảng tốc độ trục chính cao nhất của máy CNC. Hầu như mọi bộ điều khiển đều hỗ trợ khoảng tốc độ trục chính lớn hơn so với khoảng máy cho phép. Trong lập trình tốc độ trục chính, giới hạn này luôn luôn do máy, *không phải* do bộ điều khiển, đưa ra.

Nhập tốc độ trục chính

Địa chỉ S liên quan đến hàm trục chính của máy và phải được gán trị số trong mọi chương trình CNC. Có ba kiểu gán giá trị số (nhập cho hàm trục chính):

- ❑ Số mã tốc độ trục chính ...các bộ điều khiển cũ, đã lạc hậu
- ❑ Tốc độ trục chính trực tiếp ...r/min (vòng/phút)
- ❑ Tốc độ trục chính theo chu vi ... ft/min hoặc m/min

Trên các máy tiện CNC, cả ba kiểu này đều có thể tồn tại, tùy theo hệ điều khiển. Đối với hệ thống phay CNC, tốc độ trục chính theo chu vi là không áp dụng được, chỉ dùng tốc độ trục chính trực tiếp và số mã tốc độ trục chính. Sự lựa chọn tốc độ trục chính bằng số mã chuyên biệt là phương pháp đã lạc hậu, ít dùng trên các bộ điều khiển hiện đại.

Địa chỉ tốc độ trục chính S chưa đủ để lập trình, cần có thêm vài thuộc tính bổ sung cho địa chỉ này. Các thuộc tính đó điều khiển môi trường hàm trục chính. Ví dụ, nếu tốc độ trục chính được chọn là S400 trong chương trình, lệnh lập trình này là chưa hoàn chỉnh. Giá trị tốc độ trục chính, có thể là 400 m/min, hoặc 400 in/min (tùy theo ứng dụng gia công) còn thiếu *chiều quay* trục chính.

Hầu hết các trục chính của máy đều có thể quay theo hai chiều, *thuận* hoặc *ngược* chiều kim đồng hồ, tùy theo kiểu loại và sự gá lắp dụng cụ cắt. Sự quay trục chính cần được chuyên biệt trong chương trình, bổ sung cho hàm tốc độ trục chính. Hệ điều khiển cung cấp hai hàm M điều khiển chiều quay trục chính – M03 và M04

CHIỀU QUAY TRỤC CHÍNH

Các thuật ngữ *phải* và *trái*, *lên* và *xuống*, *thuận* và *ngược* chiều kim đồng hồ đều được *so sánh* với quy chiếu cho trước. Để quy định sự quay trục chính là thuận (CW) và ngược (CCW) chiều kim đồng hồ, cần thiết lập phương pháp quy chiếu tiêu chuẩn, trong trường hợp này được gọi là điểm quy chiếu (điểm chuẩn).

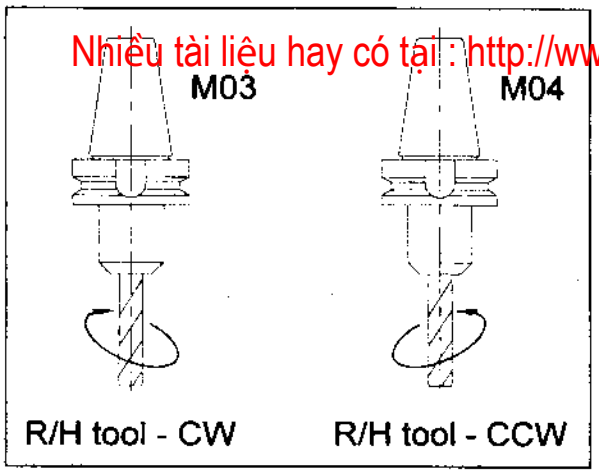
Chiều quay trục chính luôn luôn liên quan đến điểm quy chiếu được thiết lập từ phía trục chính của máy. Phần của máy chứa trục chính, được gọi là đầu máy. Quan sát từ vùng đầu máy theo *chiều dọc theo đường tâm trục chính hướng về mặt cuối trục chính*, thiết lập điểm quy chiếu để xác định chiều quay CW và CCW của trục chính. Đối với khoan CNC, máy phay và trung tâm gia công CNC, điểm quy chiếu này khá đơn giản. Đối với máy tiện cũng áp dụng các nguyên tắc tương tự.

Chiều phay

Sẽ không thực tế nếu nhìn xuống theo đường tâm trục chính, hướng vuông góc đến chi tiết gia công. Điểm quy chiếu tiêu chuẩn là từ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



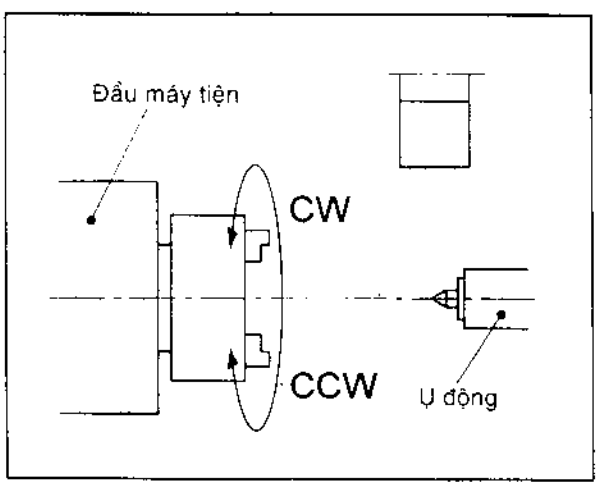
Hình 11.1. Chiều quay trục chính trên máy phay

vị trí người vận hành quay mặt về phía máy phay đứng. Dựa trên điểm quy chiếu này, có thể dễ dàng xác định sự quay trục chính *thuận* và *ngược* chiều kim đồng hồ (Hình 11.1).

Chiều tiện

Phương pháp xác định chiều tiện cũng tương tự chiều phay. Người vận hành quay về phía máy tiện, bên trái là đầu máy, bên phải là ụ động. Hình 11.2 minh họa chiều quay trục chính trên máy tiện CNC.

Tuy các mô tả CW và CCW trong minh họa trên Hình 11.2 là ngược với chiều các mũi tên, nhưng chúng là đúng. Lý do là có hai điểm quy chiếu khả dĩ và đều dùng đường tâm trục chính làm trục chiếu. Chỉ một trong hai điểm chiếu này phù hợp với định nghĩa tiêu chuẩn và điểm đó là đúng. Định nghĩa chiều quay trục chính trên máy tiện là *hoàn toàn* như trên trung tâm gia công.



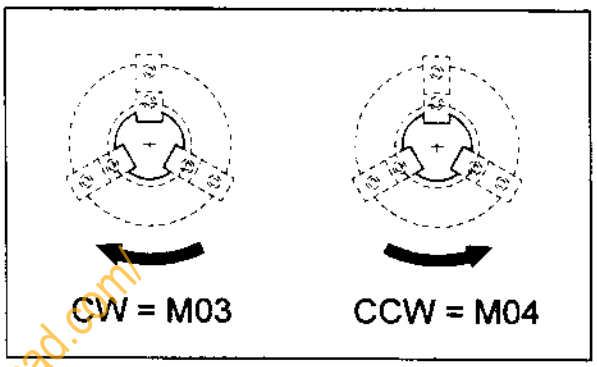
Hình 11.2. Chiều quay trục chính trên máy tiện. Các chiều CW và CCW có vẻ bị đảo ngược.

Để thiết lập chiều quay trục chính là CW và CCW, cần chiếu từ đầu máy tiện hướng đến mặt trục chính.

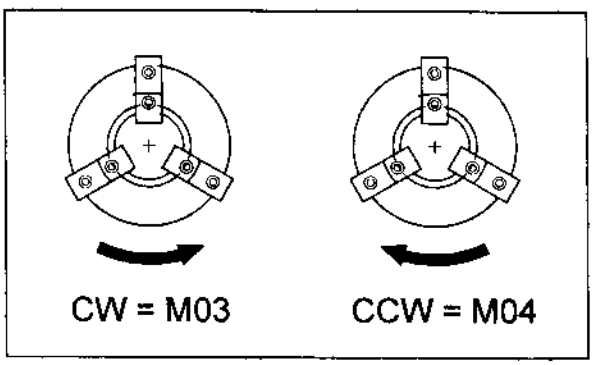
Phương pháp thứ nhất thiết lập điểm quy chiếu bắt đầu ở vùng *đầu máy* tiện. Từ vị trí này, chiếu đến ụ động, các chiều thuận (CW) và ngược (CCW) sẽ được thiết lập một cách chính xác.

Phương pháp thứ hai thiết lập điểm quy chiếu bắt đầu từ vùng *ụ động*, hướng đến mâm cặp, đây là phương pháp *sai*.

Bạn hãy so sánh hai minh họa – Hình 11.3 biểu thị điểm chiếu từ đầu máy (mâm cặp), Hình 11.4 biểu thị điểm chiếu từ ụ động, các mũi tên phải đảo chiều.



Hình 11.3. Chiều quay trục chính nhìn từ đầu mâm cặp.



Hình 11.4. Chiều quay trục chính nhìn từ ụ động.

Các đặc tính chiều quay

Nếu chiều quay trục chính là thuận (CW), cần sử dụng hàm M03 trong chương trình, nếu quay ngược (CCW) cần sử dụng M04.

Do tốc độ trục chính S trong chương trình phụ thuộc vào hàm chiều quay trục chính M03 hoặc M04, quan hệ của chúng trong chương trình là rất quan trọng.

Địa chỉ tốc độ trục chính S và hàm chiều quay trục chính M03 hoặc M04 phải được hệ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

điều khiển chấp nhận đồng thời. Nếu thiếu một trong hai lệnh này, bộ điều khiển sẽ không chấp nhận, đặc biệt khi máy bắt đầu hoạt động. Ít nhất có hai phương pháp đúng để lập trình tốc độ và chiều quay trục chính.

- Nếu tốc độ và chiều quay trục chính được lập trình chung trong một block, tốc độ và chiều quay trục chính sẽ đồng thời khởi động.
- Nếu tốc độ và chiều quay trục chính được lập trình trong các block riêng rẽ, trục chính sẽ không quay cho đến khi cả lệnh tốc độ và chiều quay được xử lý.

Khởi động trục chính

Các ví dụ dưới đây sẽ minh họa một số phương pháp để khởi động tốc độ và chiều quay trục chính trong chương trình. Những ví dụ này đều giả thiết *không có sự xác lập* tốc độ trục chính S, thông qua chương trình trước đó hoặc thông qua *Manual Data Input* (MDI – nhập dữ liệu bằng tay). Trên các máy CNC, chưa có sự đăng ký hoặc mặc định tốc độ trục chính khi máy đóng công tắc chính (ON)

➲ Ví dụ A – Ứng dụng phay

```
N1 G20
N2 G17 G40 G80
N3 G90 G00 G54 X14.0 Y9.5
N4 G43 Z1.0 H01 S600 M03 (SPEED WITH ROTATION)
N5 ...
```

Ví dụ này là một trong các định dạng thông dụng đối với các nguyên công phay. Cả tốc độ và chiều quay trục chính đều được xác lập với chuyển động trục Z hướng đến chi tiết gia công. Phương pháp thứ hai là khởi động trục chính với chuyển động XY – block N3 trong ví dụ:

```
N3 G90 G00 G54 X14.0 Y9.5 S600 M03
```

Lựa chọn tùy theo quan điểm của nhà lập trình. G20 trong block riêng rẽ là không cần thiết đối với các bộ điều khiển Fanuc.

➲ Ví dụ B – Ứng dụng phay

```
N1 G20
N2 G17 G40 G80
N3 G90 G00 G54 X14.0 Y9.5 S600    (SPEED ONLY)
N4 G43 Z1.0 H01 M03             (ROTATION STARTS)
N5 ...
```

Ví dụ B là đúng về kỹ thuật nhưng không hợp logic. Không có lợi ích khi chia tốc độ trục chính và chiều quay vào hai block. Điều này làm cho chương trình khó diễn dịch hơn.

➲ Ví dụ C – Ứng dụng phay

```
N1 G20
N2 G17 G40 G80
N3 G00 G90 G54 X14.0 Y9.5 M03 (ROTATION SET)
N4 G43 Z1.0 H01                (NO ROTATION)
N5 G01 Z0.1 F50.0 S600      (ROTATION STARTS)
N6 ...
```

Ví dụ C không sai nhưng thiếu tính thực tế. Sẽ không có nguy hiểm nếu máy đóng công tắc chính (ON) ngay trước khi chạy chương trình này. Mặt khác, M03 *sẽ* kích hoạt chiều quay trục chính, nếu chương trình khác đã được xử lý trước đó. Điều này có thể đưa đến tình huống nguy hiểm. Do đó, bạn hãy tuân theo nguyên tắc đơn giản:

Lập trình M03 hoặc M04 cùng với hoặc sau địa chỉ S, không đặt trước S.

➲ Ví dụ D – Ứng dụng tiện với G50

```
N1 G20
N2 G50 X13.625 Z4.0 T0100
N3 G96 S420 M03 (SPEED SET - ROTATION STARTS)
N4 ...
```

Đây là định dạng được ưa thích trên các máy tiện CNC, nếu sử dụng phương pháp xác lập G50 kiểu cũ. Do tốc độ trục chính được xác lập là CSS – *Constant Surface Speed* (tốc độ bề mặt không đổi) - hệ thống điều khiển sẽ tính toán số vòng quay thực trong một phút (r/min) dựa trên giá trị CSS 420ft/min và đường kính chi tiết hiện hành ở vị trí X13.625. Ví dụ E là đúng về kỹ thuật nhưng có thể gây nguy hiểm do M03 được lập trình trong block trước địa chỉ S.

➲ Ví dụ E - Ứng dụng tiện với G50

```
N1 G20
N2 G50 X13.625 Z4.0 T0100 M03 (ROTATION SET)
N3 G00 X6.0 Z0.1                (NO ROTATION)
N4 G96 G01 Z0 F0.04 T0100 S420 (ROTAT. STARTS)
N5 ...
```

➲ Ví dụ F – Ứng dụng tiện không dùng G50

```
N1 G20 T0100
N2 G96 S420 M03 (SPEED SET - ROTATION STARTS)
N3 G00 ...
```

Trong ví dụ F, dùng cho các bộ điều khiển hiện đại (do G50 không còn được sử dụng làm lệnh đăng ký vị trí), tốc độ trục chính của máy sẽ được tính toán với giá trị bù dao cắt được lưu trong thanh ghi *Work Geometry Offset* (bù dạng hình học) của hệ thống điều khiển. Hệ thống này sẽ tính toán số vòng / phút khi block N2 được thực thi.

Các ví dụ này chỉ là các phương pháp *đúng về kỹ thuật* để khởi động trục chính. Tất cả đều chứa sự quay được lựa chọn ở đầu chương trình, bao quát cả các ứng dụng tiện và phay. Sự khởi động trục chính ở đầu chương trình là có ý đồ, do đối với dụng cụ cắt thứ nhất trong chương trình hoàn toàn chưa có địa chỉ tốc độ và lệnh chiều quay được kích hoạt (thường được thực hiện từ dụng cụ cắt trước đó). Tuy nhiên, bộ điều khiển có thể còn lưu tốc độ trục chính và chiều quay từ dụng cụ cuối cùng của chương trình gia công *trước đó*.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Dụng cụ bất kỳ *tiếp sau* dụng cụ cắt thứ nhất thường giả định sự chọn tốc độ và chiều quay đã lập trình cho dụng cụ trước đó. Nếu chỉ lập trình tốc độ trục chính cho dụng cụ kế tiếp, mà *không có* chiều quay, dụng cụ cắt này sẽ lấy chiều quay được lập trình *gần nhất*. Nếu chỉ chiều quay M03 hoặc M04 được lập trình, tốc độ trục chính S sẽ hoàn toàn như dụng cụ cắt trước đó.

Bạn cần cẩn thận nếu chương trình chứa các hàm dừng chương trình M00 hoặc M01, hoặc hàm dừng trục chính M05. Các hàm này đều dừng trục chính một cách tự động. Điều đó có nghĩa là cần bảo đảm tuyệt đối sự quay trục chính xảy ra đúng thời điểm, đúng tốc độ và đúng chiều. Bạn phải luôn luôn lập trình tốc độ và chiều quay trục chính trong cùng một block và cho *từng* dụng cụ cắt. Cả hai hàm này liên kết với nhau một cách logic, cần được đặt chung trong block để có cấu trúc chương trình hợp lý và an toàn.

DỪNG TRỤC CHÍNH

Nói chung, hầu hết các nguyên công đều đòi hỏi trục chính quay với tốc độ xác định. Ví dụ, trước khi lập trình sự thay dao hoặc đảo chiều chi tiết ở giữa chương trình, bạn cần dừng trục chính. Trục chính cũng phải dừng lại trong quá trình gia công ren bằng taro và ở cuối chương trình. Một số hàm M sẽ dừng quay trục chính một cách tự động (các hàm M00, M01, M02, và M30). Sự quay trục chính cũng sẽ tự động dừng lại trong một số chu kỳ cố định. Để kiểm soát toàn bộ chương trình, sự dừng trục chính phải được chuyên biệt trong chương trình. Trong khi lập trình, bạn không nên tính đến các hàm khác để dừng trục chính. Có một hàm đặc biệt để dừng trục chính trong chương trình.

Để dừng quay trục chính, bạn hãy dùng hàm M05. Hàm này sẽ dừng quay trục chính thuận hoặc ngược chiều kim đồng hồ. Do hàm M05 chỉ dừng trục chính (khác với các hàm khác cũng dừng trục chính như M00, M01, M02, M30, ...), hàm M05 được dùng trong các tình huống, cần dừng trục chính mà *không* tác động đến mọi hoạt động khác đã được lập trình. Các ví dụ bao gồm đổi chiều khi tarô ren, chuyển động dao cắt đến vị trí phân độ, vị trí thay ổ dao, hoặc sau khi trả về zero máy, tùy theo nguyên công cụ thể. Bạn không cần hàm M05, khi sử dụng một trong các hàm M khác tự động dừng trục chính. Mặt khác, thứ tự của M05 trong block không quá nghiêm ngặt. Phương pháp này có thể làm cho chương trình hơi dài hơn nhưng dễ hiểu và dễ đọc hơn, đặc biệt đối với người vận hành CNC còn ít kinh nghiệm.

Hàm dừng trục chính có thể được lập trình trong block riêng, ví dụ:

```
N120 M05
```

hoặc trong block chương trình có chuyển động dao cắt, chẳng hạn:

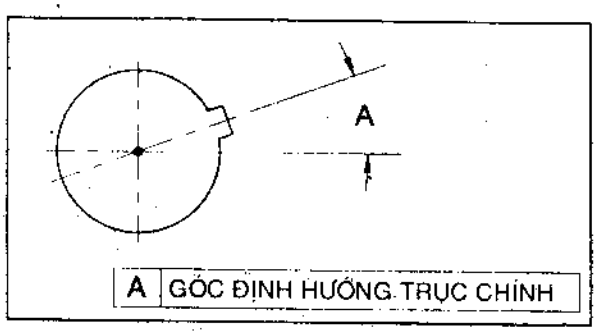
```
N120 21.0 M05
```

Chuyển động này luôn luôn hoàn tất trước, *sau đó* trục chính sẽ dừng lại. Đây là tính năng an toàn được xây dựng bên trong hệ điều khiển. Bạn cần nhớ lập trình M03 hoặc M04 để khôi phục chiều quay trục chính.

SỰ ĐỊNH HƯỚNG TRỤC CHÍNH

Hàm M cuối cùng liên quan đến hoạt động trục chính là M19. Hàm này được sử dụng thường xuyên để đưa trục chính của máy vào vị trí định hướng. Các mã M khác có thể có hiệu lực, tùy theo hệ điều khiển, ví dụ M20 trên một số bộ điều khiển. Hàm định hướng trục chính là hàm rất chuyên biệt, ít khi xuất hiện trong chương trình. Hàm M19 thường được sử dụng chủ yếu trong khi gá lắp, hoặc trong chế độ *Manual Data input* (MDI). Hàm này dùng cho các hệ thống phay, chỉ các máy tiện CNC được trang bị đặc biệt mới sử dụng hàm này. Hàm M19 chỉ được sử dụng khi trục chính tĩnh tại, thường sau khi dừng trục chính, khi hệ thống điều khiển thực thi hàm M19, sẽ xảy ra hoạt động như sau:

Trục chính sẽ hơi xoay theo cả hai chiều, thuận và ngược, và sau khoảng thời gian ngắn, sẽ kích hoạt cơ cấu khóa bên trong. Trong một số trường hợp, sự khóa này sẽ phát ra âm thanh. Trục chính bị khóa ở vị trí *chính xác*, không thể quay bằng tay. Vị trí khóa chính xác do nhà chế tạo máy công cụ xác định, biểu thị bằng gốc xác lập (Hình 11.5).



Hình 11.5. *Góc định hướng trục chính do nhà chế tạo máy xác định, không thể thay đổi.*

Trong vận hành máy công cụ, hàm M19 cho phép người vận hành máy đặt dụng cụ cắt vào trục chính bằng tay và bảo đảm sự định hướng thích hợp cho gá lắp dao. Các chương sau sẽ

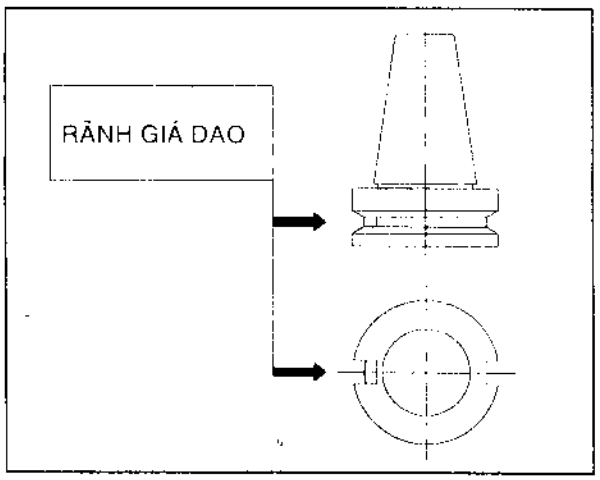
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

trình bày chi tiết về định hướng trục chính và ứng dụng thực tế, ví dụ trong các chu kỳ doa một lưỡi cắt.

CẢNH BÁO – Định hướng giá lắp dao sai có thể gây hư hại cho máy hoặc chi tiết.

Nhiều trung tâm gia công (không phải tất cả) sử dụng giá lắp dao chỉ có thể vào ổ dao theo một chiều. Để đạt được điều này, giá dao có rãnh đặc biệt, khớp với thiết kế bên trong của trục chính (Hình 11.6). Để tìm phía giá dao có rãnh, có một vệt lõm nhỏ ở phía có rãnh.



Hình 11.6. *Rãnh trong giá dao được dùng để định hướng chính xách dao cắt với trục chính – đa số các máy đều có tính năng này.*

Đối với dao nhiều lưỡi cắt, chẳng hạn mũi khoan, phay mặt đầu, dao chuốt, ... định hướng cắt với vị trí trục chính dừng là không quá quan trọng. Tuy nhiên, đối với dao một lưỡi cắt, định hướng lưỡi cắt trong khi gá lắp là rất quan trọng, đặc biệt khi sử dụng các chu kỳ cố định sử dụng định hướng trục chính *theo rãnh*, là G76 và G87, dao rút ra khỏi lỗ gia công nhưng *không quay*. Để tránh hư hại lỗ doa, cần điều khiển sự rút dao. Định hướng trục chính sẽ bảo đảm dao cắt rút ra khỏi lỗ doa theo chiều đúng, do đó cần gá lắp dao một cách chính xác.

Các máy cho phép gá lắp giá dao vào trục chính theo cả hai chiều vẫn yêu cầu xác lập dao thích hợp khi sử dụng các chu kỳ cố định G76 hoặc G87 trong chương trình.

TỐC ĐỘ TRỤC CHÍNH – R/MIN

Khi lập trình trung tâm gia công CNC, bạn hãy gán tốc độ trục chính trực tiếp theo *số vòng quay trên một phút* (r/min). Ví dụ, block cơ bản chứa tốc độ trục chính 200 r/min, đòi hỏi mục nhập dữ liệu như sau:

```
N230 S200 M03
```

Định dạng này là điển hình trên các bộ điều khiển phay, khi không sử dụng tốc độ chu vi (vận tốc dài), không cần sử dụng lệnh chuẩn bị đặc biệt để nêu rõ xác lập r/min, do đây là mặc định của bộ điều khiển. Giá trị r/min phải có số gia tối thiểu là một. Các giá trị phân số hoặc thập phân đều không được phép, và r/min phải trong khoảng đặc tính kỹ thuật của máy.

Vài trung tâm gia công có thể được trang bị với tùy chọn có sự lựa chọn tốc độ trục chính *kép – r/min trực tiếp* và *tốc độ chu vi* (vận tốc dài). Trong trường hợp này, và cũng như trong lập trình máy tiện, cần dùng lệnh chuẩn bị để chọn *chế độ* vận tốc. G96 được dùng cho tốc độ chu vi (vận tốc dài) và G97 chọn r/min (vận tốc góc) trực tiếp.

TỐC ĐỘ TRỤC CHÍNH – BỀ MẶT

Lập trình tốc độ trục chính dựa trên *vật liệu* được gia công và *đường kính dụng cụ cắt* (trung tâm gia công) hoặc *đường kính chi tiết* (máy tiện). Nguyên tắc chung là đường kính càng lớn tốc độ r/min của trục chính càng thấp. Không được ước đoán tốc độ trục chính, *phải luôn luôn tính toán giá trị này*. Sự tính toán sẽ bảo đảm tốc độ trục chính *tỷ lệ thuận với đường kính được lập trình*. Tốc độ trục chính sai sẽ tác động xấu đến cả dụng cụ cắt và chi tiết gia công.

Tính gia công của vật liệu

Để tính toán tốc độ trục chính, vật liệu thường có tính gia công định mức đối với vật liệu dụng cụ cắt. Giá trị định mức này được tính theo phần trăm của một số vật liệu phổ biến, chẳng hạn thép carbon trung bình, hoặc trực tiếp theo *tốc độ chu vi* hoặc *tốc độ bề mặt*. Tốc độ bề mặt được tính theo *feet trên phút (ft/min)* đối với hệ Anh, và hệ mét là *mét trên phút (m/min)*. Viết tắt cũ của ft/min là *FPM – Feet Per Minute*. Giá trị tốc độ bề mặt biểu thị mức độ khó gia công với *vật liệu dụng cụ cắt cho trước*. Tốc độ này càng thấp, độ khó gia công của vật liệu chế tạo chi tiết càng cao.

Ở đây có sự nhấn mạnh *"vật liệu dụng cụ cắt cho trước"*. Để mọi so sánh có ý nghĩa, cần thực hiện với cùng kiểu dụng cụ cắt, ví dụ, tốc độ bề mặt đối với dao cắt làm bằng thép dụng cụ cắt gọt tốc độ cao (thép gió) sẽ thấp hơn nhiều so với hợp kim cứng trên cơ sở carbides.

Dựa trên tốc độ bề mặt và đường kính dao cắt (hoặc đường kính chi tiết gia công trên máy tiện), tốc độ trục chính của máy có thể được tính theo *số vòng quay/phút*, sử dụng công thức riêng cho hệ Anh hoặc riêng cho hệ mét.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tốc độ bề mặt tăng đối với các vật liệu mềm và giảm đối với vật liệu cứng

Thép gió chậm hơn so với hợp kim cứng

Tốc độ trục chính - đơn vị Anh

Để tính tốc độ trục chính theo *r/min*, cần biết *tốc độ chu vi* (vận tốc dài) của vật liệu đối với kiểu dụng cụ cắt và đường kính của dụng cụ hoặc chi tiết gia công:

$$r/min = \frac{12 \times ft/min}{\pi \times D}$$

Trong đó:

r/min:	Tốc độ trục chính (số vòng quay/phút)
12:	Hệ số đổi feet sang inch
ft/min:	Tốc độ chu vi (feet/phút)
D:	đường kính (dao cắt khi phay, chi tiết gia công khi tiện), tính theo inch

➲ **Ví dụ:**

Tốc độ chu vi của vật liệu gia công được chọn là 150ft/min, đường kính dao cắt là 1.75 inch.

r/min = (12 x 150) / (3.145 x 1.75)
= 327.4 = 327 r/min

Nhiều ứng dụng lập trình sử dụng công thức rút gọn nhưng vẫn chính xác:

$$r/min = \frac{3.82 \times ft/min}{D}$$

Đối với các tính toán không yêu cầu cao về độ chính xác, hằng 3.82 có thể làm tròn là 4. Khi tính toán cần áp dụng đúng các đơn vị đo để bảo đảm kết quả chính xác.

Không được dùng chung các đơn vị hệ Anh và hệ mét trong một chương trình

Tốc độ trục chính - đơn vị hệ mét

Khi dùng hệ mét trong chương trình, bản chất của công thức nêu trên không thay đổi, chỉ khác các đơn vị:

$$r/min = \frac{1000 \times m/min}{\pi \times D}$$

Trong đó:

r/min:	Tốc độ trục chính (số vòng quay/phút)
1000:	Hệ số đổi mét sang milimét
m/min:	Tốc độ chu vi tính theo mét / phút.
D:	Đường kính tính theo mm (dao cắt khi phay, chi tiết khi tiện)

➲ **Ví dụ:**

Tốc độ bề mặt là 30 m/min và đường kính dao cắt là 15 mm

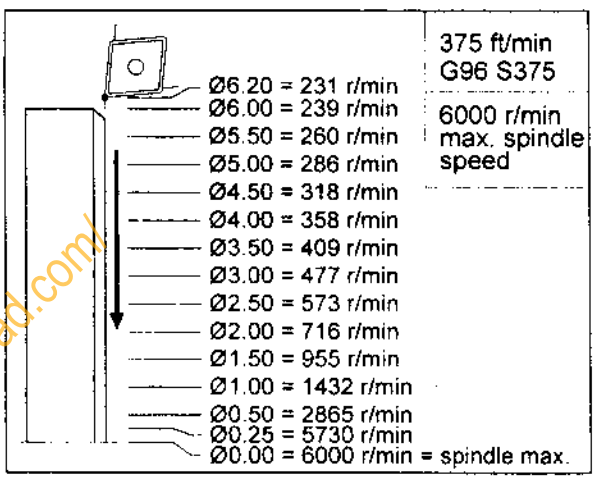
r/min = (1000 x 30) / (3.1415 x 15)
= 636.6 = 637 r/min

Có thể áp dụng công thức rút gọn khi chia 1000 cho π.

$$r/min = \frac{318.3 \times m/min}{D}$$

TỐC ĐỘ BỀ MẶT KHÔNG ĐỔI

Trên các máy tiện CNC, quá trình gia công khác với quá trình phay. Dao tiện không có đường kính còn đường kính dao doa không có quan hệ với tốc độ trục chính, do đó *đường kính chi tiết gia công* được dùng để tính toán tốc độ trục chính. Khi chi tiết đang được gia công, đường kính giảm dần, chẳng hạn khi tiện thô (Hình 11.7), lập trình tốc độ trục chính theo r/min là phi thực tế, do đó cần sử dụng *tốc độ bề mặt* trực tiếp trong chương trình tiện.



Hình 11.7. *Ví dụ về tiện mặt trục ngoài sử dụng chế độ tốc độ bề mặt không đổi G96*

Chọn tốc độ bề mặt chỉ là một nửa vấn đề. Nửa còn lại là thông báo sự lựa chọn này cho hệ thống điều khiển. Hệ thống điều khiển cần được xác lập theo chế độ *tốc độ bề mặt*, không theo chế độ *r/min*. Các nguyên công khoan, chuốt,... là phổ biến trên máy tiện và đòi hỏi r/min, cần chuyên biệt sự lựa chọn giữa *tốc độ bề mặt* và *số vòng quay/phút*. Điều này được thực hiện với các lệnh G96 và G67 đặt *trước* hàm trục chính.

`G96 S M03` *Chọn tốc độ bề mặt*

`G97 S M03` *Chọn số vòng quay/phút*

Đối với phay, thường không cần sự phân biệt này, do tốc độ trục chính mặc định là *r/min*.

Bằng cách lập trình lệnh tốc độ bề mặt G96 đối với tiện và doa, bộ điều khiển nhập chế độ đặc biệt, được gọi là *Constant Surface Speed – CSS* (tốc độ bề mặt không đổi). Trong chế độ này, số vòng quay thực tế của trục chính sẽ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tăng hoặc giảm một cách tự động tùy theo giá trị đường kính đang được cắt gọt (đường kính hiện hành). *Constant Surface Speed* tự động được thiết lập trong hệ điều khiển khả dụng cho hầu hết các máy tiện CNC. Tính năng này không chỉ tiết kiệm thời gian lập trình, mà còn cho phép dụng cụ có thể cắt gọt *lượng vật liệu không đổi* vào mọi thời điểm, do đó giảm tốc độ mòn lưỡi cắt và cải thiện độ bóng bề mặt.

Hình 11.7 minh họa các ví dụ về tiện mặt trụ ngoài bắt đầu ở X6.2 (Ø6.2) và tiến dần vào đường tâm của chi tiết. Chương trình sử dụng G96 S375 và 6000 r/min là tốc độ trục chính cực đại của máy tiện.

Dù đã nêu ra các tốc độ được chọn, nhưng cùng với số vòng quay/phút tương ứng, quá trình cập nhật là liên tục. Bạn hãy chú ý sự tăng nhanh r/min khi dao cắt tiến gần đến đường tâm. Khi dao cắt tiến đến XO (Ø0.0), tốc độ trục chính đạt giá trị cực đại, trong khoảng tốc độ hiện hành. Do tốc độ này có thể quá cao trong một số trường hợp, hệ điều khiển cho phép xác lập tốc độ cực đại theo yêu cầu.

Để lập trình tốc độ chu vi (vận tốc dài) cho máy tiện CNC, ở đây có vài lựa chọn. Trong ba ví dụ kế tiếp, bạn hãy xem xét các lựa chọn tốc độ. Các hàm đổi tốc độ (số bánh răng hộp số) được bỏ qua trong các ví dụ đó.

➲ Ví dụ 1:

Tốc độ bề mặt được xác lập ngay sau xác lập tọa độ, sử dụng lệnh G50 (hoặc G92)

```
N1 G20
N2 G50 X16.0 Z5.0 T0100
N3 G96 S400 M03
...
```

Trong ứng dụng này, tốc độ thực của trục chính sẽ dựa trên đường kính hiện hành 16 inch, kết quả là 95 r/min trong block N3. Trong một số trường hợp, tốc độ này có thể quá thấp. Bạn hãy xét ví dụ kế tiếp:

➲ Ví dụ 2:

Trên các máy tiện CNC lớn, xác lập G50 của đường kính trục X có thể khá lớn, ví dụ Ø24.0 inch. Trong ví dụ trước, đường kính đích của chuyển động dao kế tiếp là không quan trọng, nhưng trong ví dụ 2 điều đó rất quan trọng. Ví dụ:

```
N1 G20
N2 G50 X24.0 Z5.0 T0100
N3 G96 S400 M03
N4 G00 X20.0 T0101 M08
...
```

Trong ví dụ 2, vị trí dao cắt ban đầu là X24.0 và chuyển động dao kết thúc tại X20.0, cả hai giá trị này đều là đường kính, do đó giá trị chuyển động thực là 2.0 inch. Tại X24.0, trục chính quay với tốc độ 64 r/min, tại X20.0 là 76 r/min. Sự chênh lệch là rất nhỏ, do đó không đòi hỏi sự lập trình đặc biệt. Tuy nhiên sẽ có sự khác biệt rõ rệt nếu vị trí khởi đầu có đường kính lớn, và dao cắt chuyển động đến đường kính đích có giá trị nhỏ.

➲ Ví dụ 3a:

Từ vị trí ban đầu Ø24.0 inch, dao cắt sẽ dịch chuyển đến đường kính 2.0 inch khá nhỏ:

```
N1 G20
N2 G50 X24.0 Z5.0 T0100
N3 G96 S400 M03
N4 G00 X2.0 T0101 M08
...
```

Tốc độ trục chính khi khởi động chương trình (block N3) hoàn toàn như ví dụ trước, 64 r/min. Trong block kế tiếp (N4), tốc độ tính toán cho Ø 2.0 inch là 764 r/min, do hệ điều khiển tính toán một cách tự động. Đây là sự thay đổi (tăng tốc độ trục chính) khá lớn, có thể gây ra tác động xấu đến một số máy tiện CNC. Điều gì sẽ xảy ra nếu dao cắt đạt đến Ø2.0 *trước khi* tốc độ trục chính đạt đến giá trị 764 r/min? Dao cắt có thể bắt đầu cắt gọt kim loại với tốc độ thấp hơn nhiều so với yêu cầu. Để giải quyết vấn đề này, cần chỉnh sửa chương trình CNC:

➲ Ví dụ 3b:

Sự chỉnh sửa được thực hiện trong block N3. Thay vì lập trình chế độ tốc độ bề mặt không đổi, bạn hãy lập trình *r/min trực tiếp* cho đích Ø2.0 inch, dựa trên tốc độ bề mặt 400ft/min. Cần tính giá trị r/min trước, sau đó xác lập CSS sẽ được lập trình trong block kế tiếp.

```
N1 G20
N2 G50 X24.0 Z5.0 T0100
N3 G97 S764 M03
N4 G00 X2.0 T0101 M08
N5 G96 S400
....
```

Trong ví dụ này, tại Ø 24.0 (X24.0 trong N2), giá trị r/min thực chỉ là 64 r/min. Tại Ø 2.0 (X2.0 trong N4), r/min sẽ là 764. Dao cắt có thể đạt đến X2.0 trước khi tốc độ trục chính tăng đến 764 r/min, nếu chưa được tính toán và lập trình trước.

Kỹ thuật này chỉ hữu ích nếu máy tiện CNC không hỗ trợ sự trễ thời gian tự động. Nhiều máy tiện hiện đại có bộ thời chuẩn bên trong, buộc dao cắt phải chờ trước khi cắt gọt, cho đến khi trục chính đạt được tốc độ theo yêu cầu.

Nhiều máy tiện CNC ngày nay *không* sử

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

dụng xác lập (bù mũi dao) bằng xác lập *Geometry Offset* (bù dạng hình học). Trong trường hợp này, có thể chưa biết đường kính ban đầu ở vị trí zero của máy, bạn nên lập trình dwell ngắn trước khi thực sự cắt gọt.

Xác lập tốc độ trục chính cực đại

Khi máy tiện CNC vận hành với chế độ *Constand Surface Speed*, tốc độ trục chính có quan hệ trực tiếp với đường kính chi tiết hiện hành. Đường kính phôi càng nhỏ, tốc độ trục chính càng cao. Do đó câu hỏi là – điều gì sẽ xảy ra nếu đường kính đạt giá trị *zero*? Đường như không thể lập trình với đường kính zero, nhưng ít nhất có hai trường hợp xét đến khả năng này.

Trong trường hợp thứ nhất, đường kính zero được lập trình cho mọi *thao tác đường tâm*. Mọi nguyên công khoan, khoan tâm, tarô ren, ... đều được lập trình với đường kính zero (X0). Các nguyên công này luôn luôn lập trình theo chế độ r/min trực tiếp, sử dụng lệnh G97. Trong chế độ G97, tốc độ trục chính được điều khiển trực tiếp, r/min không thay đổi.

Trường hợp thứ hai của đường kính zero là khi *vạt mặt ngoài* cho đến tận đường tâm. Đây là tình huống khác. Đối với mọi thao tác ở X0, đường kính cắt không thay đổi do r/min trực tiếp được lập trình. Trong khi vạt mặt ngoài, đường kính liên tục giảm cho đến khi dao cắt đạt đến đường tâm trục chính, nhưng sẽ *không có* giá trị 0 r/min ở đường tâm trục chính trong chế độ G96 (Hình 11.7).

Mỗi khi chế độ CSS hoạt động và dao cắt đạt đến đường tâm trục chính ở X0, kết quả thường là tốc độ trục chính *cao nhất* khả dụng *trong khoảng tốc độ hoạt động*. Điều này có vẻ nghịch lý, nhưng đó là thực tế. Tình huống đó là chấp nhận được khi chi tiết được gá lắp hợp lý, không chìa ra quá xa từ mâm cặp hoặc đồ gá, dao cắt có độ cứng vững cao... Khi chi tiết được lắp trong đồ gá hoặc gá lắp lệch tâm, chi tiết chìa ra quá dài, hoặc có các điều kiện bất lợi khác, tốc độ trục chính cực đại ở đường tâm có thể *quá cao*, ảnh hưởng đến tính an toàn khi gia công.

Có một giải pháp đơn giản cho vấn đề này, sử dụng tính năng lập trình của Fanuc và các bộ điều khiển khác. Chế độ CSS có thể sử dụng với *giới hạn cao nhất được xác lập trước*, chuyên biệt theo số vòng quay/phút. Hàm chương trình để xác lập tốc độ trục chính cực đại là G50. Xác lập cực đại này đôi khi được gọi là *sự chặn tốc độ trục chính cực đại*. Bạn không nên nhầm lẫn G50 với ý nghĩa thứ hai của hàm này là xác lập trước thanh ghi vị trí. Dưới đây là ví dụ:

```
O1201  (SPINDLE SPEED CLAMP)
N1 G20 T0100
N2 G50 X9.0 Z5.0 S1500        (1500 R/MIN MAX)
N3 M42                      (HIGH SPINDLE RANGE)
N4 G96 S400 M03              (CSS AND 400 FT/MIN)
N5 G00 G41 X5.5 Z0 T0101 M08
N6 G01 X-0.07 F0.012       (BELOW CENTER LINE)
N7 G00 Z0.1
N8 G40 X9.0 Z5.0 T0100
N9 M01
```

Điều gì thực sự xảy ra trong chương trình O1201? Block N1 chọn đơn vị đo theo hệ Anh. Block N2 có *hai ý nghĩa*:

- Chỉ xác lập vị trí tọa độ dao cắt:

  `G50 X9.0 Z5.0`

- *Đồng thời* xác lập r/min cực đại là 1500

  `G50 X9.0 Z5.0 S1500`

Block N3 chọn khoảng tốc độ trục chính; block N4 xác lập chế độ CSS, sử dụng tốc độ bề mặt là 400 ft/min. Chiều quay trục chính M03 được gọi trong cùng block này (N4). Trong block N5, dao cắt thực hiện chuyển động nhanh đến Ø5.5 và mặt ngoài của chi tiết gia công. Trong khi chuyển động nhanh, sẽ kích hoạt hàm chất làm nguội và bù bán kính mũi dao. Tốc độ trục chính ở Ø5.5 sẽ là 278 r/min, sử dụng công thức đã nêu ở phần trước. Block N6 điều khiển sự tiện mặt ngoài. Với tốc độ ăn dao 0.0012 in/rev, lưỡi cắt sẽ tiện phôi đến đường tâm. Trong thực tế, điểm cuối được lập trình trên phía kia của đường tâm trục chính. Kích cỡ bán kính mũi dao cần được xét đến khi lập trình với sự bù bán kính mũi dao và đến đường tâm trục chính. Phần sau sẽ giải thích quá trình này trong khi cắt gọt.

Block N7 dịch chuyển mũi dao ra khỏi mặt ngoài.100 inch với tốc độ nhanh. Trong hai block còn lại, dao sẽ chuyển động nhanh đến vị trí phân độ với sự xóa bù bán kính trong N8 và dừng chương trình tùy chọn trong N9.

Bạn hãy suy nghĩ điều sẽ xảy ra trong các block N5 và N6. Trục chính quay với tốc độ 278 r/min ở Ø5.5. Do chế độ CSS có hiệu lực, khi lưỡi dao cắt chi tiết, đường kính này trở nên nhỏ dần trong khi *r/min liên tục tăng*.

Nếu *không có* giới hạn tốc độ trục chính cực đại trong block N2, tốc độ trục chính ở đường tâm sẽ tương ứng *r/min cực đại* trong khoảng tốc độ M42, có thể đến 3500 r/min hoặc cao hơn.

Với giới hạn tốc độ trục chính cực đại được xác lập trước là 1500 r/min, trục chính sẽ tăng tốc độ liên tục, nhưng chỉ đạt đến 1500 r/min

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

và *giữ nguyên* cho đến khi trong phần cắt gọt kế tiếp.

Tại bộ điều khiển, người vận hành có thể dễ dàng thay đổi giá trị giới hạn cực đại, phản ánh các điều kiện gia công thực tế hoặc tối ưu hóa chế độ cắt.

Tốc độ trục chính được xác lập trước (hoặc giới hạn) theo xác lập *r/min* cực đại, bằng cách lập trình hàm S chung với lệnh chuẩn bị G50. Nếu hàm S trong block không có G50, bộ điều khiển sẽ diễn dịch là tốc độ trục chính mới (CSS hoặc r/min), hoạt động từ block đó trở đi. *Lỗi này có thể rất nghiêm trọng!*

Bạn cần thận trọng khi xác lập trước r/min cực đại cho trục chính.

Tốc độ trục chính cực đại có thể được giới hạn trong block riêng hoặc block có xác lập tọa độ dao cắt hiện hành. Trong ví dụ O1201, block N2 chứa cả hai xác lập này. Nói chung, xác lập kết hợp rất hữu ích ở phần bắt đầu của dụng cụ cắt, xác lập trong block riêng chỉ hữu dụng nếu phát sinh yêu cầu thay đổi tốc độ trục chính cực đại ở giữa chừng của dụng cụ cắt, ví dụ giữa tiện mặt trụ và tiện côn sử dụng cùng một dao cắt.

Để lập trình lệnh G50 trong block riêng, ở vị trí bất kỳ trong chương trình, bạn chỉ cần đặt lệnh này chung với giá trị tốc độ trục chính được xác lập trước. Loại block này sẽ *không có hiệu lực* đối với xác lập tọa độ bất kỳ, mà chỉ biểu thị ý nghĩa khác của lệnh G50. Ví dụ dưới đây nêu rõ ứng dụng lệnh G50 cho cả hai xác lập tọa độ và/hoặc xác lập trước tốc độ trục chính cực đại.

```
N12 G50 X20.0 Z3.0 S1500      Ý nghĩa kép
N38 G50 S1250                 Ý nghĩa đơn
N15 G50 X8.5 Z2.5             Ý nghĩa đơn
N40 G50 Z4.75 S700            Ý nghĩa kép
```

Từ các ví dụ này, bạn dễ dàng hiểu được ý nghĩa công dụng của G50. Lệnh này có hai ý nghĩa hoàn toàn độc lập nhau, có thể được lập trình chung trong một block hoặc trong hai block riêng rẽ.

Nếu máy tiện CNC hỗ trợ G92 thay vì G50, bạn cần nhớ chúng có ý nghĩa và công dụng hoàn toàn như nhau. Trên máy tiện, G50 thông dụng hơn G92 nhưng phương pháp lập trình là như nhau.

Tính toán đường kính chi tiết trong chế độ CSS

Thông tin về *giá trị đường kính* giới hạn tốc độ trục chính là rất hữu ích. Điều này có thể ảnh hưởng đến giá trị xác lập trước đối với tốc độ trục chính cực đại. Để tìm giá trị đường kính tại đó *Constant Surface Speed* trở nên cố định, bạn cần áp dụng công thức:

$$D = \frac{12 \times ft/min}{\pi \times r/min}$$

Trong đó

D:	Đường kính để CSS cố định (inch)
12:	Hệ số chuyển đổi feet sang inch
ft/min:	Tốc độ bề mặt hoạt động
r/min:	Tốc độ trục chính cực đại xác lập trước.

➲ Ví dụ - Đơn vị Anh

Nếu giá trị xác lập trước trong chương trình là G50 S1000 và tốc độ bề mặt được chọn là G96 S350, CSS sẽ được giới hạn khi đạt đến ∅ 1.3369 inch.

D = (12 × 350) / (π × 1000)
= 1.3369015
= ∅1.3369

Công thức này có thể rút gọn thành:

$$D = \frac{3.82 \times ft/min}{r/min}$$

Công thức tính đường kính theo hệ mét:

$$D = \frac{1000 \times m/min}{\pi \times r/min}$$

Trong đó:

D:	Đường kính để CSS cố định (mm)
1000:	Hệ số chuyển đổi mét sang milimét
m/min:	Tốc độ bề mặt hoạt động
r/min:	Tốc độ trục chính cực đại xác lập trước.

Công thức rút gọn:

$$D = \frac{318.3 \times m/min}{r/min}$$

➲ Ví dụ - Đơn vị mét

Nếu giá trị xác lập trước trong chương trình là G50 và S1200 và tốc độ bề mặt được chọn là G96 S165, CSS được giới hạn khi đạt đến ∅ 43.768mm

D = (1000 × 165) / (π × 1200)
= 43.767609 = ∅43.768 mm

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

12 ĐIỀU KHIỂN TỐC ĐỘ CẮT

Tốc độ cắt là đại lượng lập trình gần gũi nhất với hàm trục chính. Hàm trục chính điều khiển tốc độ và chiều quay trục chính, tốc độ ăn dao (cắt) điều khiển *tốc độ chuyển động của dụng cụ cắt*, thường để cắt gọt lượng dư gia công (phôi). *Sự định vị nhanh*, đôi khi còn gọi là *chuyển động nhanh* hoặc *chuyển động ngang nhanh*, không được coi là tốc độ ăn dao thực sự, sẽ được trình bày chi tiết trong Chương 19.

BIỂU KHIỂN TỐC ĐỘ CẮT

Tốc độ cắt là tốc độ tại đó dụng cụ cắt loại bỏ phần vật liệu dư bằng tác động cắt gọt.

Tác động cắt gọt có thể là chuyển động quay của dụng cụ cắt (ví dụ, khoan, phay,...), chuyển động quay của phôi (nguyên công tiện), hoặc tác động khác (cắt bằng ngọn lửa, cắt bằng laser, phóng điện,...). Hàm tốc độ cắt được dùng trong chương trình CNC để chọn giá trị tốc độ cắt thích hợp cho nguyên công cắt gọt cụ thể.

Trong lập trình CNC, có thể sử dụng một trong hai kiểu tốc độ cắt.

- ❑ Tốc độ ăn dao tính theo phút
- ❑ Tốc độ ăn dao theo vòng quay.

Các kiểu máy phổ biến, trung tâm gia công và máy tiện CNC, có thể được lập trình theo một trong hai chế độ đó. Trong thực tế, thường sử dụng *lượng ăn dao/phút* trên trung tâm gia công và *lượng ăn dao/vòng quay* trên máy tiện.

Bảng dưới đây liệt kê mã G dùng cho trung tâm gia công và máy tiện.

LƯỢNG ĂN DAO	Phay	Tiện nhóm A	Tiện nhóm B	Tiện nhóm C
Phút	G94	G98	G94	G94
Vòng quay	G95	G99	G95	G95

Mã G *Nhóm A* được dùng phổ biến trên các bộ điều khiển Fanuc và trong sách này

Ngoài ra còn có *tốc độ ăn dao thời gian ngược*, nhưng rất ít sử dụng trong thực tế.

HÀM TỐC ĐỘ CẮT

Địa chỉ gọi hàm tốc độ cắt trong chương trình là địa chỉ F, tiếp theo là số các chữ số. Số lượng chữ số sau địa chỉ F tùy thuộc chế độ ăn dao và ứng dụng của máy công cụ, có thể cho phép dấu thập phân.

Lượng ăn dao tính theo phút

Đối với các ứng dụng phay, mọi lượng ăn dao theo chế độ nội suy tuyến tính và nội suy vòng đều được lập trình theo *inch per minutes (in/min)* hoặc *milimét/minute (mm/min)*. Giá trị lượng ăn dao là khoảng dao cắt hành trình trong trong một phút. Giá trị này có tính chế độ và chỉ bị xóa bằng địa chỉ F khác. Ưu điểm chính của tốc độ ăn dao tính theo phút là giá trị này *không* phụ thuộc vào tốc độ trục chính, rất hữu ích trong các nguyên công phay sử dụng nhiều cỡ đường kính dao. Viết tắt tiêu chuẩn của lượng ăn dao tính theo phút là:

- ❑ inch per minute in/min (hoặc ipm)
- ❑ Milimetes per minute mm/min

Định dạng phổ biến của lượng ăn dao/phút theo hệ Anh là F3.1 và hệ mét là F4.1. Ví dụ tốc độ ăn dao 15.5 in/min sẽ được lập trình là F15.5. Trong hệ mét, tốc độ ăn dao 250 mm/min, trong chương trình sẽ là F250.0. Một số máy đặc biệt có thể có định dạng địa chỉ F hơi khác.

Đại lượng quan trọng cần nhớ về tốc độ ăn dao là *khoảng* các giá trị ăn dao khả dụng. Khoảng ăn dao của hệ thống điều khiển *luôn luôn vượt quá* khoảng giá trị trong hệ thống tự động của máy. Ví dụ, khoảng ăn dao của hệ thống Fanuc là.001 – 24.000.0 in/min, hoặc 0.0001 – 240000.0 mm/min. Bạn hãy chú ý, sự khác biệt giữa hệ Anh và hệ mét chỉ là vị trí dấu thập phân, *không phải là sự chuyển đổi thực sự*. Trong lập trình, chỉ những tốc độ ăn dao *trong* khoảng cho phép mới được sử dụng, giá trị này luôn luôn nhỏ hơn giá trị trong hệ điều khiển.

Trong nguyên công phay, lệnh lập trình (mã G) đối với *lượng ăn dao/phút* là G94. Đối với hầu hết các máy, giá trị này được xác lập một cách tự động, theo mặc định của hệ thống và không cần viết trong chương trình. Đối với các nguyên công tiện, lượng ăn dao/phút hầu như không sử dụng. Trong *Nhóm A*, mã G đối với lượng ăn dao/phút là G98, các *Nhóm B và C* là G94. Các máy tiện CNC chủ yếu sử dụng chế độ *lượng ăn dao/vòng quay*.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lượng ăn dao theo số vòng quay

Đối với máy tiện CNC, lượng ăn dao không đo theo đơn vị thời gian, chỉ đo theo khoảng cách thực tế về hành trình của dao cắt trong một vòng quay của trục chính. *Lượng ăn dao theo số vòng quay* được dùng chủ yếu trên máy tiện (G99 trong Nhóm A). Giá trị này có tính chế độ, chỉ xóa bằng hàm ăn dao khác (thường là G98). Máy tiện còn có thể được lập trình theo *lượng ăn dao/phút* (G98) để điều khiển tốc độ ăn dao khi trục chính không quay. Khi viết tắt, tiêu chuẩn của *lượng ăn dao/vòng quay* là:

- Inch per revolution — in/rev (hoặc ipr)
- Milimeters per revolution — mm/rev

Định dạng phổ biến đối với lượng ăn dao/vòng quay là 4 chữ số thập phân trong hệ Anh và ba chữ số thập phân trong hệ mét. Ví dụ, tốc độ ăn dao 0.083333 in/rev sẽ được áp dụng trong chương trình CNC là F0.0833 trên hầu hết các bộ điều khiển; 0.42937 mm/rev sẽ được lập trình là F0.429. Nhiều hệ thống hiện đại chấp nhận lượng ăn dao đến sáu chữ số thập phân theo hệ Anh và 5 chữ số thập phân theo hệ mét.

Cần cẩn thận khi làm tròn các giá trị lượng ăn dao (chiều sâu cắt). Đối với các nguyên công tiện và phay, chỉ cần làm tròn hợp lý là đủ. Đối với tiện ren một lưỡi cắt, độ chính xác lượng ăn dao là rất quan trọng, đặc biệt đối với các ren dài hoặc rất mịn. Một số bộ điều khiển Fanuc có thể được lập trình với độ chính xác của lượng ăn dao đến 6 chữ số thập phân khi tiện ren.

Lệnh lập trình đối với lượng ăn dao/vòng quay là G99. Đối với hầu hết các máy tiện, đây là mặc định hệ thống, do đó không cần viết trong chương trình, trừ khi được sử dụng chung với lệnh G98.

Nói chung, *lượng ăn dao/phút* (G98) được dùng trong chương trình tiện CNC phổ biến hơn so với *lượng ăn dao/vòng quay* (G95) trong chương trình phay. Lý do là trên máy tiện, lệnh này điều khiển tốc độ ăn dao trong khi trục chính *không* quay. Ví dụ, trong nguyên công chuốt hoặc xọc, cữ chặn chi tiết được dùng để "đẩy" dao vào vị trí chính xác trong mâm cặp, hoặc bộ đồ gá "kéo" dao ra ngoài. Sự chạy dao nhanh sẽ trở nên quá nhanh và không thể áp dụng lượng ăn dao / vòng quay, do đó phải dùng lượng ăn dao / phút. Trong các trường hợp tương tự, cần dùng các lệnh G98 và G99 trong chương trình. Cả hai lệnh này đều có tính chế độ và xóa lẫn nhau.

LỰA CHỌN LƯỢNG ĂN DAO

Để chọn lượng ăn dao tối ưu, phù hợp nhất cho chế độ cắt cụ thể, cần sử dụng kiến thức về công nghệ gia công. Đây là phần quan trọng khi lập trình và cần được thực hiện cẩn thận. Lựa chọn lượng ăn dao tùy thuộc vào nhiều yếu tố, một số yếu tố quan trọng bao gồm:

- Tốc độ trục chính – tính theo rev / min.
- Đường kính dao [M] hoặc bán kính mũi dao [T].
- Yêu cầu độ bóng bề mặt của chi tiết.
- Dạng hình học của dụng cụ cắt.
- Các lực cắt.
- Gá lắp chi tiết.
- Khoảng chìa ra của dao cắt.
- Chiều dài chuyển động cắt.
- Phương pháp phay.
- Số lưỡi cắt (khoan, phay).
- Các yêu cầu an toàn.

Yêu cầu về an toàn luôn luôn là trách nhiệm lập trình số một, để bảo đảm an toàn cho người và thiết bị. Lượng ăn dao và tốc độ an toàn chỉ là hai trong nhiều vấn đề cần đặc biệt chú ý trong lập trình CNC.

TĂNG TỐC VÀ GIẢM TỐC

Trong khi gia công bao hình (biên dạng), chiều chuyển động cắt thay đổi liên tục, điều này là bình thường với các giao điểm, điểm tiếp tuyến và các khoảng hở. Trong bao hình, ví dụ để lập trình góc nhọn trên chi tiết, chuyển động dao dọc theo trục X trong một block sẽ cần đổi thành chuyển động dọc theo trục Y trong block kế tiếp. Để thay đổi chuyển động cắt, bộ điều khiển phải *dừng chuyển động trục X* trước, sau đó *khởi động chuyển động Y*. Do không thể *khởi động* tức thời với tốc độ ăn dao toàn phần, mà không *tăng tốc*, đồng thời không thể *dừng* ngay sự ăn dao mà không *giảm tốc độ*, sai số cắt gọt sẽ xảy ra. Sai số này có thể làm cho góc nhọn bị cắt quá mức, đặc biệt khi tốc độ ăn dao rất cao hoặc góc quá nhỏ. Điều này chỉ xảy ra trong khi sử dụng chuyển động cắt với lệnh G01, G02, G03, nhưng không xảy ra trong chế độ chuyển động nhanh G00. Trong khi chuyển động nhanh, sự giảm tốc là tự động và dao luôn luôn cách xa chi tiết gia công.

Trong tiến trình gia công CNC, sai số này có thể ít xảy ra và dù có cũng sẽ trong khoang dung sai cho phép.

Nếu cần hiệu chỉnh sai số này, các bộ điều khiển Fanuc cung cấp hai lệnh, cho phép giải quyết vấn đề:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

G09	Dừng chính xác	(chỉ một block)
G61	Chế độ dừng chính xác	(chế độ)

Sự dừng chính xác sẽ làm tăng chu kỳ gia công. Đối với những chương trình sử dụng trên các máy cũ, điều này có thể được yêu cầu trong một số trường hợp.

Lệnh dừng chính xác

Một trong hai lệnh điều khiển lượng ăn dao khi gia công qua các góc là G09 – *Exact Stop* (dừng chính xác). Đây là lệnh *không có tính chế độ* và cần lặp lại trong từng block mỗi khi cần thiết.

Trong ví dụ chương trình O1301, không có sự tăng tốc và giảm tốc. Điều này có thể gây ra sự gia công các góc không đều, do tốc độ ăn dao tương đối cao F90.0 (in/min)

```
O1301 (CẮT BÌNH THƯỜNG)
...
N13 G00 X15.0 Y12.0
N14 G01 X19.0 F90.0
N15 Y16.0
N16 X15.0
N17 Y12.0
...
```

Bằng cách bổ sung lệnh dừng chính xác G09 vào chương trình, sự chuyển động trong block đó sẽ hoàn tất *trước khi* khởi đầu chuyển động theo trục kế tiếp.

```
O1302 (CẮT G09)
...
N13 G00 X15.0 Y12.0
N14 G09 G01 X19.0 F90.0
N15 G09 Y16.0
N16 G09 X15.0
N17 Y12.0
...
```

Ví dụ O1302 bảo đảm góc sắc nét ở cả ba vị trí của chi tiết. Nếu chỉ yêu cầu một góc chính xác, cần lập trình G09 trong block kết thúc tại góc đó (chương trình O1303)

```
O1303 (CẮT G09)
...
N13 G00 X15.0 Y12.0
N14 G01 X19.0 F90.0
N15 G09 Y16.0
N16 X15.0
N17 Y12.0
...
```

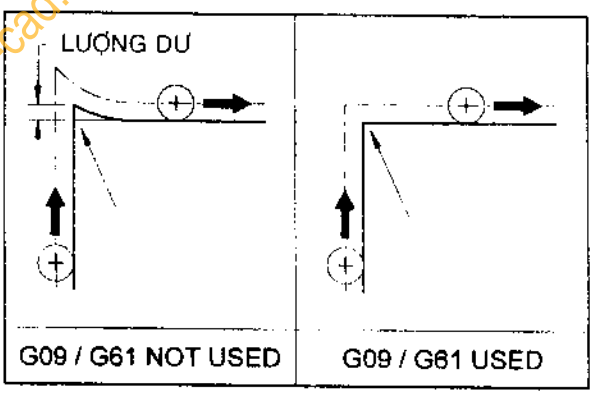
Lệnh G09 chỉ hữu dụng nếu chỉ vài block đòi hỏi giảm tốc ở các góc nhỏ. Đối với chương trình đòi hỏi chính xác tất cả các góc, sự lặp lại thường xuyên lệnh G09 là không hiệu quả.

Lệnh chế độ dừng chính xác

Lệnh thứ hai chỉnh sửa sai số ở các góc sắc nét là G61 - *Exact Stop Mode* (dừng chính xác). Lệnh này hiệu quả hơn G09 nhưng có chức năng tương tự. Khác biệt chính là lệnh G61 có *tính chế độ*, duy trì hiệu lực cho đến khi được xóa bằng lệnh chế độ cắt G64. Lệnh G61 rút ngắn thời gian lập trình nhưng không giảm thời gian chu kỳ. Lệnh này thay cho lệnh G09 nếu khi cần lặp lại nhiều lần lệnh G09 trong chương trình.

```
O1304 (CẮT G61)
...
N13 G00 X15.0 Y12.0
N14 G61 G01 X19.0 F90.0
N15 Y16.0
N16 X15.0
N17 Y12.0
N18 G64
...
```

Bạn hãy chú ý, chương trình O1304 có kết quả đồng nhất với O1301. Trong cả hai trường hợp, kiểm tra sự dừng chính xác áp dụng cho mọi chuyển động cắt – không theo chế độ trong O1301 và có tính chế độ trong O1304. Ngoài ra, bạn cần chú ý block N18, sử dụng lệnh G64 – *chế độ cắt bình thường*. Chế độ này là mặc định khi mở máy và thường không cần lập trình. Hình 12.1 minh họa chuyển động dụng cụ cắt có và không có lệnh G09/G61 ... lượng dư ở góc được phóng đại để minh họa, trong thực tế giá trị này thấp hơn nhiều.



Hình 12.1. Điều khiển lượng ăn dao khi gia công góc, với lệnh G09 hơn G61

Sự vượt qua góc một cách tự động

Trong khi sự bù bán kính mũi dao có hiệu lực đối với dao phay, tốc độ ăn dao ở các điểm đổi hướng (góc) thường *không bị* vượt qua. Trong trường hợp tương tự, lệnh chuẩn bị G62 có thể được dùng để tự động vượt qua lượng ăn dao ở các góc của chi tiết. Lệnh này có hiệu lực cho đến khi lệnh G61 (chế độ dừng chính xác), lệnh G68 (tarô ren), hoặc lệnh G64 (chế độ cắt) được lập trình.

Chế độ tarô ren

Lập trình với chế độ tarô ren G63 sẽ làm cho hệ thống điều khiển bỏ qua xác lập bất kỳ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

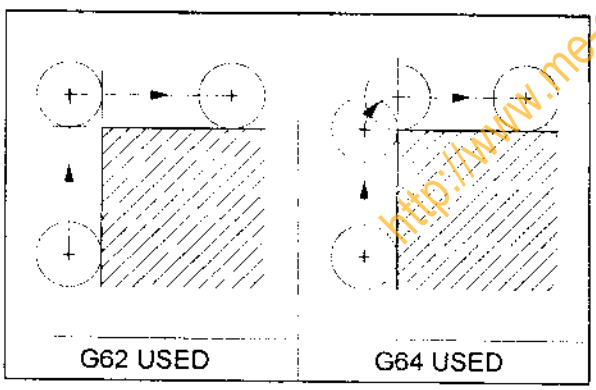
của chế độ cắt hiệu dụng tới tốc độ lập trình. Ngoài ra, độ chính xác lập 100%. G63 còn xóa chức năng của phím duy trì ăn dao trên bảng điều khiển. Chế độ tarô ren có thể được xóa bằng lập trình lệnh G61 (chế độ dừng chính xác), lệnh G62 (chọn chế độ vượt qua góc một cách tự động), hoặc lệnh G64 (chọn chế độ cắt).

Chế độ cắt

Khi chế độ cắt G64 được lập trình hoặc được kích hoạt, do mặc định hệ thống, sẽ biểu thị chế độ cắt *bình thường*. Khi lệnh này hoạt động, lệnh dừng chính xác G61, lệnh vượt qua góc một cách tự động G62, và lệnh tarô ren G63 đều không được thực thi. Điều này có nghĩa là sự tăng tốc và giảm tốc đều được thực hiện một cách bình thường và sự vượt qua lượng ăn dao sẽ có hiệu lực. Đây là chế độ mặc định thông dụng nhất đối với hệ thống điều khiển.

Chế độ cắt có thể xóa bằng cách lập trình G61, G62, hoặc G63.

Lệnh G64 thường không cần lập trình, trừ khi một hoặc nhiều chế độ ăn dao khác được dùng trong cùng một chương trình. Hình 12.2 minh họa sự so sánh các chế độ G62 và G64.



***Hình 12.2.** Chế độ vượt qua góc G62 và chế độ cắt mặc định G64.*

SỰ ĂN DAO KHÔNG ĐỔI

Sự nội suy đường tròn sẽ được trình bày trong Chương 28. Chương này chỉ giải thích chi tiết và đưa ra các ví dụ về duy trì tốc độ cắt không đổi đối với các cung trong và ngoài, theo quan điểm *thực tiễn*. Ở đây sẽ tập trung vào *kiến thức* về tốc độ ăn dao không đổi, chưa đề cập đến các *ứng dụng*.

Trong lập trình, nói chung cần tính toán các giá trị tọa độ của mọi điểm thay đổi biên dạng (đường bao) dựa trên *bản vẽ* chi tiết. Bán kính dao cắt tạo ra *đường tâm* của quỹ đạo dụng cụ cắt thường bị xem nhẹ. Khi lập trình các cung theo kích thước bản vẽ, thay vì đường tâm của dao cắt, tốc độ ăn dao áp dụng cho cung được lập trình luôn luôn liên quan với bán kính *được lập trình*, *thay vì* cắt bán kính thực ở tâm dao cắt.

Khi sự bù bán kính dao cắt hoạt động và quỹ đạo dụng cụ cắt của cung được bù theo bán kính dao cắt, bán kính cung *thực* được cắt có thể *nhỏ hơn* hoặc *lớn hơn*, tùy theo giá trị bù cho chuyển động dụng cụ cắt.

Điều quan trọng cần hiểu là bán kính cắt *hiệu dụng* sẽ *giảm* kích cỡ đối với mọi cung trong và sẽ *tăng* đối với mọi cung ngoài. Do tốc độ cắt không thay đổi một cách tự động trong chế độ bù bán kính dao, bạn cần điều chỉnh trong chương trình. Nói chung, sự điều chỉnh này là không cần thiết, trừ trường hợp đòi hỏi độ bóng bề mặt cao hoặc bán kính dao quá lớn. Điều này chỉ áp dụng cho chuyển động tròn không dùng cho chuyển động cắt tuyến tính.

Tốc độ cắt theo chế độ tròn

Sự xác lập tốc độ ăn dao theo chuyển động tròn thường tương tự như tốc độ ăn dao tuyến tính. Hầu hết các chương trình không thay đổi tốc độ ăn dao theo chế độ chuyển động tròn hoặc tuyến tính. Nếu yêu cầu độ bóng bề mặt cao, tốc độ ăn dao "bình thường" phải được điều chỉnh *cao hơn* hoặc *thấp hơn* với sự xem xét về bán kính dao, kiểu cắt bán kính (cung trong hay ngoài) và các điều kiện cắt. Bán kính dao cắt càng lớn, càng cần hiệu chỉnh tốc độ cắt đối với các cung được lập trình.

Trong trường hợp cắt gọt cung, quỹ đạo dao cắt tương đương (sau khi áp dụng bù bán kính dao) có thể *lớn hơn* hoặc *nhỏ hơn nhiều* so với cung được lập trình theo các kích thước bản vẽ.

Tốc độ cắt đối với các chuyển động cung được bù luôn luôn dựa trên tốc độ cắt chuyển động tuyến tính (Chương 28). Công thức tiêu chuẩn để tính tốc độ ăn dao tuyến tính:

$$F_t = r/min \times F_t \times n$$

Trong đó

F_t:	Tốc độ ăn dao tuyến tính (in/min hoặc mm/min).
r/min:	Tốc độ trục chính
F_t:	Tốc độ ăn dao/răng (lưỡi cắt)
n:	Số lượng biên cắt gọt (số lưỡi cắt)

Dựa trên công thức tốc độ ăn dao tuyến tính, các điều chỉnh lượng ăn dao theo cung phụ thuộc vào *phía* của cung được gia công – *trong* hay *ngoài*. Tốc độ ăn dao tuyến tính phải *tăng* đối với cung ngoài và *giảm* đối với cung trong.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Đối với cung *ngoài*, tốc độ ăn dao được điều chỉnh *tăng*, đến giá trị cao hơn:

$$F_o = \frac{F \times (R + r)}{R}$$

Trong đó:

F_o: Tốc độ ăn dao đối với cung ngoài
F_l: Tốc độ ăn dao tuyến tính
R: Bán kính ngoài của chi tiết
r: Bán kính dao cắt.

Đối với các cung *trong*, tốc độ ăn dao được điều chỉnh *giảm*, đến giá trị thấp hơn:

$$F_i = \frac{F_l \times (R - r)}{R}$$

Trong đó:

F_i: Tốc độ ăn dao đối với cung trong
F_l: Tốc độ ăn dao tuyến tính
R: Bán kính ngoài của chi tiết
r: Bán kính dao cắt.

TỐC ĐỘ ĂN DAO CỰC ĐẠI

Tốc độ ăn dao cực đại có thể lập trình cho các máy CNC do nhà chế tạo máy xác định, không phải do nhà sản xuất bộ điều khiển. Ví dụ, tốc độ cắt cực đại trên một máy nào đó có thể chỉ 393 in/min, dù hệ thống CNC có thể hỗ trợ tốc độ cắt cao hơn vài lần. Điều này có thể áp dụng cho mọi bộ điều khiển, nhưng cần có các xem xét lập trình bổ sung đối với máy tiện CNC khi *lượng ăn dao/vòng quay* là phương pháp chính để lập trình dụng cụ cắt.

Các yếu tố tác động đến tốc độ cực đại

Tốc độ cắt cực đại / vòng quay luôn luôn bị giới hạn theo *tốc độ trục chính được lập trình (r/min)* và *tốc độ chạy dao ngang nhanh cực đại* của máy tiện CNC. Trong một số trường hợp tốc độ cắt / vòng quay được lập trình quá cao một cách vô lý. Vấn đề này rất phổ biến khi tiện ren với dao một lưỡi cắt.

Máy CNC không thể cung cấp tốc độ cắt cao hơn thiết kế, kết quả sẽ không chính xác. Sai số ren lớn đến mức không thể chấp nhận được. Khi sử dụng tốc độ cắt quá cao và tốc độ trục chính quá nhanh trong một chương trình, cần kiểm tra xem tốc độ cắt cuối có vượt quá tốc độ cắt cực đại được phép trên máy đó không. Có thể tính *tốc độ ăn dao/vòng quay cực đại* bằng công thức:

$$F_{max} = \frac{R_{max}}{r/min}$$

Trong đó:

F_{max}: Tốc độ cắt cực đại được phép in/rev
R_{max}: Tốc độ cắt cực đại được chọn từ trục X và trục Z
r/min: Tốc độ trục chính tính theo vòng/phút

R_{max} tính theo *in/min* hoặc *mm/min* tùy theo hệ đơn vị được chọn. Chương 37 sẽ trình bày chi tiết về các giới hạn tốc độ cắt khi tiện ren.

FEEDHOLD VÀ OVERRIDE

Trong khi chạy chương trình, tốc độ cắt được lập trình có thể tạm thời dừng lại hoặc thay đổi bằng cách dùng một trong hai tính năng khả dụng của hệ thống điều khiển. Thứ nhất là công tắc *feedhold* và thứ hai là công tắc *feedrate override*. Cả hai công tắc này đều là tiêu chuẩn và cho phép người vận hành CNC điều khiển bằng tay tốc độ cắt lập trình trong khi thực thi chương trình, chúng được bố trí trên bảng vận hành.

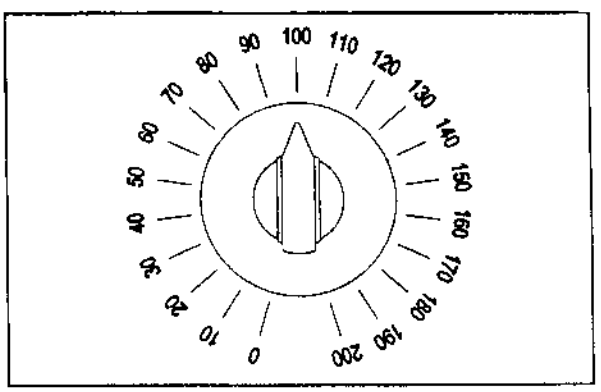
Công tắc Feedhold

Đây là nút nhấn có thể chuyển đổi giữa hai chế độ *Feedhold ON* và *Feedhold OFF*, sử dụng cho cả *lượng ăn dao/phút* và *lượng ăn dao/vòng quay*. Trên nhiều bộ điều khiển, công tắc feedhold không chỉ dừng ăn dao khi G01, G02, hoặc G03 có hiệu lực, mà còn dừng cả chuyển động nhanh G00. Các tính năng khác của chương trình vẫn duy trì hiệu lực trong suốt trạng thái feedhold.

Đối với một số nguyên công cắt gọt xác định, tính năng feedhold (tạm dừng ăn dao) *bị mất hiệu lực một cách tự động*. Điều này là điển hình đối với tarô ren và tiện ren, G92 và G76 trên máy tiện.

Công tắc Feedrate Override

Feedrate Override (vượt qua tốc độ ăn dao) thường được điều khiển bằng công tắc xoay đặc biệt trên bảng điều khiển của máy CNC (Hình 12.3).



Hình 12.3. Công tắc Feedrate Override.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trên công tắc xoay có cách vạch chia, biểu thị số *phần trăm tốc độ cắt được lập trình*. Khoảng vượt qua tốc độ này là 0% đến 200%, trong đó 0% có thể là hoàn toàn không chuyển động hoặc chuyển động chậm nhất, tùy theo loại máy. Xác lập 200% *tăng gấp đôi* tốc độ ăn dao đã lập trình. Ví dụ, tốc độ ăn dao lập trình là 12.0 in/min (F12.0) được coi là 100%. Nếu công tắc override được chỉnh đến vạch 80%, tốc độ cắt thực sẽ là 9.6 in/min; nếu chỉnh đến 110%, tốc độ cắt thực sẽ là 13.2 in/min.

Nguyên tắc đơn giản này cũng áp dụng cho hệ mét. Ví dụ tốc độ cắt lập trình là 300 mm/min, tương ứng 100%. Với xác lập 80%, tốc độ cắt thực là 240 mm/min, 110% sẽ là 330 mm/min.

Công tắc feed override cũng áp dụng cho *lượng ăn dao/vòng quay*. Ví dụ tốc độ cắt lập trình là .014 in/rev, tốc độ cắt thực là 0.126 in/rev với xác lập là 90% và 0.182 in/rev với xác lập 130%. Nếu yêu cầu chính xác lượng ăn dao / vòng quay trục chính, bạn cần cẩn thận với các xác lập override. Ví dụ tốc độ cắt lập trình là F0.012 tính theo *in/rev*, sự thay đổi một vạch chia trên công tắc xoay sẽ *tăng* hoặc *giảm* giá trị lập trình đúng 10%. Do đó tốc độ cắt sẽ là .0108 tại 90%, .0120 tại 100%, .0132 tại 110%,... Trong hầu hết các trường hợp, có thể không yêu cầu tốc độ cắt chính xác, nhưng bạn cần nhớ, một số tốc độ cắt không thể xác lập trên công tắc xoay, chẳng hạn.115in/min, do các vạch chia cố định 10% trên công tắc override.

Trong chế độ tiện ren một lưỡi cắt G32, công tắc feedrate override *không có hiệu lực*. Feedrate override cũng không có hiệu lực đối với các chu kỳ tarô ren G84 và G74 trên các trung tâm gia công, các chu kỳ tiện ren một lưỡi cắt G92 và G76 trên các máy tiện. Nếu chế độ tarô ren được sử dụng trên hệ thống phay với lệnh G63, cả *feedrate override* và *feedhold* đều không có hiệu lực - *trong toàn bộ chương trình*!

Hệ thống điều khiển cung cấp hai tính năng vượt qua tốc độ cắt đối với các chuyển động cắt *khác với* các chu kỳ tarô ren, hoặc tiện ren, chúng là M48 và M49. Đây là các tính năng *lập trình*, nhưng có thể không có trên một số bộ điều khiển.

Các tính năng vượt qua tốc độ cắt

Tuy hàm tốc độ cắt sử dụng địa chỉ F, nhưng vẫn có thể sử dụng hai hàm M trong chương trình để xác lập *feedrate override ON* hoặc *OFF*. Trên bảng vận hành, công tắc được dùng để điều khiển feedrate override. Nếu người vận hành CNC quyết định tốc độ cắt lập trình tạm thời *tăng* hoặc *giảm*, công tắc này là rất hữu ích. Mặt khác, trong khi gia công, trong đó tốc độ cắt *phải* được sử dụng theo giá trị đã lập trình, công tắc override chỉ được phép xác lập theo giá trị 100%.

Ví dụ điển hình là các nguyên công tarô ren *không dùng* các chu kỳ, chỉ áp dụng các lệnh chuẩn bị G00 và G01. Các hàm M48 và M49 được dùng cho các mục đích đó.

M48	Hàm xóa feedrate override là OFF, có nghĩa là feedrate override *hoạt động*
M49	Hàm xóa feedrate override là ON, có nghĩa là feedrate override *không hoạt động*.

Hàm M48 cho phép người vận hành CNC sử dụng công tắc feedrate một cách tự do; hàm M49 sẽ làm cho tốc độ cắt được thực thi *như đã lập trình*, bất kể công tắc feedrate override được xác lập ON trên bảng điều khiển. Công dụng chủ yếu của hai hàm này là tarô ren hoặc tiện ren *không dùng* chu kỳ, trong đó *phải* luôn luôn duy trì tốc độ cắt chính xác đã lập trình. Ví dụ dưới đây là minh hoạc kỹ thuật lập trình này.

```
N10 S500 M03                    (USING TAP 12 TPI)
...
N14 G00 X5.0 Y4.0 M08
N15 Z0.25
N16 M49          (DISABLE FEEDRATE OVERRIDE)
N17 G01 Z-0.625 F41.0 M05
N18 Z0.25 M04
N19 M48          (ENABLE FEEDRATE OVERRIDE)
N20 G00 X.. Y.. M05
N21 M03
...
```

Sự tarô ren xảy ra giữa các block N16 và N19, sự vượt qua tốc độ cắt bị mất hiệu lực trong các block đó.

ĐỊA CHỈ E TRONG TIỆN REN

Một số máy tiện CNC cũ sử dụng địa chỉ tốc độ cắt E khi tiện ren thay cho địa chỉ F.

Hàm tốc độ cắt E tương tự hàm F, chuyên biệt hướng ren theo *lượng ăn dao/vòng quay*, với đơn vị *in/rev* hoặc *mm/rev*, nhưng có *độ chính xác cao hơn* (số chữ số thập phân lớn hơn). Trên hệ điều khiển Fanuc kiểu cũ, ví dụ 6T, khoảng bước ren là:

- **Anh – Bộ điều khiển Fanuc 6T**

  F = 0.0001 *đến* 50.0000 in/rev

  E = 0.000001 *đến* 50.0000 in/rev

- **Mét – bộ điều khiển Fanuc 6T**

  F= 0.001 *đến* 500.000mm/rev

  F= 0.0001 *đến* 500.0000 mm/rev

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trên các bộ điều khiển hiện đại, FS – 0/ 10/ 11/ 15/ 16T, các khoảng cũng tương tự (không có địa chỉ E), nhưng phương pháp an toàn nhất để tìm các khoảng khả dụng là đọc bảng đặc tính kỹ thuật của hệ thống điều khiển.

Địa chỉ E là dư trên các bộ điều khiển hiện đại, được duy trì để bảo đảm tính tương thích với các chương trình cũ có thể còn được sử dụng trên các máy có bộ điều khiển mới. Các khoảng tốc độ cắt ren khả dụng thay đổi giữa các hệ thống điều khiển khác nhau, tùy theo loại ren và đơn vị được dùng trong chương trình.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

13 HÀM DỤNG CỤ CẮT

Máy CNC sử dụng bộ thay dao tự động phải có hàm dụng cụ cắt đặc biệt (hàm T) có thể được dùng trong chương trình. Hàm này điều khiển hành vi của *dụng cụ cắt*, tùy theo kiểu máy công cụ. Có các khác biệt rõ rệt giữa hàm T được dùng trên trung tâm gia công CNC và trên máy tiện CNC. Ngoài ra còn có các khác biệt giữa các bộ điều khiển đối với cùng kiểu máy công cụ. Địa chỉ lập trình thông dụng đối với hàm dụng cụ cắt là địa chỉ T.

Đối với các trung tâm gia công CNC, hàm T thường chỉ điều khiển *số dao cắt*. Đối với máy tiện CNC, hàm này điều khiển sự phân độ *số ổ dao* và *số bù dao cắt*.

HÀM T TRÊN TRUNG TÂM GIA CÔNG

Mọi trung tâm gia công CNC đứng và ngang đều có tính năng *Automatic Tool Changer* (bộ thay dao tự động), viết tắt là *ATC*. Trong chương trình hoặc chế độ MDI (nhập dữ liệu bằng tay) trên máy, tính năng này sử dụng hàm T, trong đó địa chỉ T là số dụng cụ cắt do nhà lập trình chọn. Các chữ số tiếp theo xác định số dụng cụ cắt. Trên máy CNC với sự thay dao bằng tay, có thể hoàn toàn không cần dùng hàm dụng cụ cắt.

Trước khi bắt đầu lập trình cho trung tâm gia công CNC, bạn cần biết *kiểu lựa chọn dao cắt* của máy đó. Có hai kiểu chọn dao chính được dùng trong quá trình thay dao tự động:

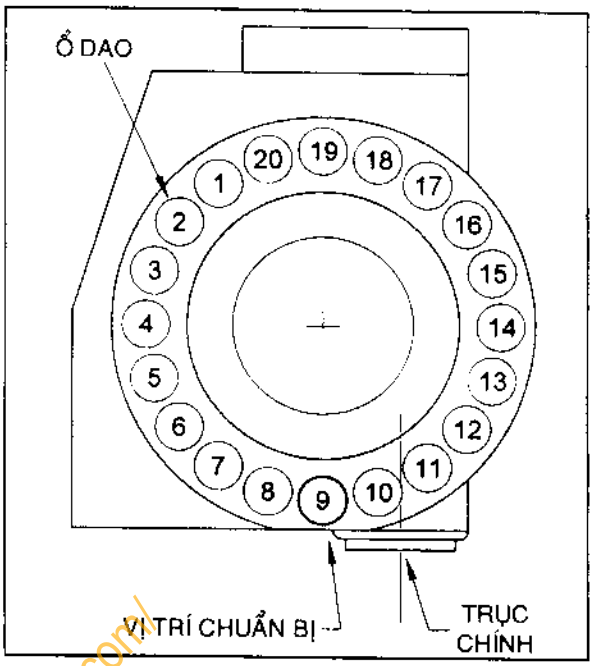
- Kiểu cố định
- Kiểu nhớ ngẫu nhiên

Để hiểu sự khác biệt giữa chúng, bước thứ nhất là hiểu các nguyên tắc chung về lưu giữ dụng cụ cắt và lựa chọn dao cắt, được dùng trên các trung tâm gia công CNC hiện đại.

Hộp giữ dao cắt

Trung tâm CNC (đứng hoặc ngang) được thiết kế với *hộp giữ dao* đặc biệt (đôi khi còn được gọi là *hộc dao cắt*) chứa tất cả các dao cắt cần thiết trong chương trình. Hộp này không phải là nơi cất giữ dao lâu dài, nhưng nhiều người vận hành máy thường giữ các dụng cụ cắt gọt được sử dụng thường xuyên, nếu có thể. Hộp chứa 20 dao cắt được nêu trên Hình 13.1.

Dung lượng của hộp chứa dao có thể là 10, 20, hoặc hàng trăm dao trên các máy đặc biệt.



Hình 13.1. Mặt bên của hộp chứa 20 dao cắt.

Trung tâm gia công trung bình có 20 đến 40 dao, trung tâm lớn có thể nhiều hơn. Hộp này thường có dạng hình tròn hoặc oval, chứa một số xác định *các ổ dao*, nơi lắp giá dao và dao cắt trong quá trình xác lập máy. Các ổ dao được đánh số theo thứ tự liên tiếp. Điều quan trọng cần biết là *chỉ số ổ dao là cố định* cho từng ổ dao. Hộp dao có thể được vận hành bằng tay khi gá lắp và tự động, thông qua chương trình CNC hoặc MDI. Số lượng ổ dao là số dụng cụ cắt tối đa có thể thay một cách tự động trên trung tâm gia công.

Trong khoảng hành trình của hộp dao có một vị trí đặc biệt, được dùng để thay dao tự động. Vị trí này thẳng hàng với bộ thay dao và thường được gọi là vị trí *chờ*, vị trí *chuẩn bị*, vị trí *sẵn sàng*, hoặc vị trí *thay dao*.

Sự chọn dao cố định

Trung tâm gia công sử dụng chọn dao cố định đòi hỏi người vận hành lắp tất cả các dao vào ổ chứa dao *tương ứng với số dao*. Ví dụ, dao số 1 (được gọi là T01 trong chương trình) *phải* được đặt vào ổ chứa dao số 1, dao số 7 (T07 trong chương trình) phải trong ổ dao số 7, v.v...

Ổ dao được lắp trên một phía của máy CNC, thường cách xa khu vực gia công. Với sự

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

lựa chọn dao cố định, hệ điều khiển không thể xác định chỉ số dụng cụ trong số ổ dao ở thời điểm bất kỳ. Người vận hành CNC phải lắp đúng chỉ số dao vào ổ dao có số tương ứng trong quá trình xác lập máy. Kiểu chọn dao này được dùng chủ yếu trên các trung tâm gia công CNC kiểu cũ hoặc một số trung tâm CNC rẻ tiền.

Lập trình dao cắt khá dễ dàng - mỗi khi chỉ số hàm T được dùng trong chương trình, số dụng cụ cắt đó sẽ được chọn khi thay dao. Ví dụ:

```
N67 T04 M06
```

hoặc

```
N67 M06 T04
```

hoặc

```
N67 T04
N68 M06
```

đơn giản có nghĩa là đưa dao số 4 vào trục chính. Điều gì sẽ xảy ra đối với dao trong trục chính vào thời điểm đó? Hàm thay dao M06 sẽ làm cho dao đang hoạt động trở về ổ dao tương ứng với dao đó, trước khi đưa dao mới vào vị trí. Nói chung, bộ thay dao sẽ chọn đường ngắn nhất để chọn dao mới.

Ngày nay, kiểu chọn dao này được coi là thiếu thực tế, chi phí cao và tốn thời gian. Có sự tổn thất thời gian rõ rệt trong khi thay dao, do máy công cụ phải chờ cho đến khi tìm được dao mới và đưa dao đó đến trục chính. Nhà lập trình có thể cải thiện hiệu suất bằng cách chọn các dao và gán chỉ số dao một cách cẩn thận, không cần theo đúng thứ tự sử dụng. Các ví dụ trong phần này dựa trên phương pháp lựa chọn dụng cụ cắt hiện đại hơn, được gọi là *nhớ ngẫu nhiên*.

Lựa chọn dao cắt nhớ ngẫu nhiên

Tính năng này được sử dụng phổ biến trên các trung tâm gia công hiện đại, lưu giữ mọi dụng cụ cắt cần thiết để gia công chi tiết trong các ổ dao của hộp chứa dao, cách xa khu vực gia công. Nhà lập trình CNC xác định từng dụng cụ cắt bằng chỉ số T, thường theo thứ tự sử dụng. Sự gọi chỉ số dụng cụ cắt do chương trình thực hiện sẽ chuyển dụng cụ đó đến vị trí *chờ* bên trong hộp chứa dao. Điều này có thể xảy ra một cách đồng thời, trong khi máy đang dùng dụng cụ khác để cắt gọt chi tiết. Sự thay dao thực tế sẽ xảy ra vào đúng thời điểm thích hợp. Đây là khái niệm về *sự chờ dao cắt kết tiếp* - với hàm T biểu thị *dao cắt kế tiếp*, không phải dao cắt hiện hành. Trong chương trình, dao cắt kế tiếp có thể được chuẩn bị sẵn sàng bằng cách lập trình vài block đơn giản:

```
T04                          (DAO 4 CHUẨN BỊ)
   <...Gia công với dao trước đó...>
...
M06                (THAY DAO - T04 TRONG TRỤC CHÍNH)
T15                          (DAO KẾ TIẾP CHUẨN BỊ)
...
   <... Gia công với dao 4 - T04...>
...
```

Trong block thứ nhất, dao T04 được gọi vào vị trí chờ của hộp dao, trong khi dao trước đó vẫn còn cắt gọt. Khi sự cắt gọt này hoàn tất sẽ xảy ra sự thay dao thực sự, T04 trở thành dao hoạt động. Ngay khi đó, hệ thống CNC sẽ tìm dao kế tiếp (T15 trong ví dụ nêu trên), và đưa dao này vào vị trí chờ, trong khi T04 đang cắt gọt.

Ví dụ này cho thấy hàm T hoàn toàn *không* thực hiện sự thay dao thực sự. Đối với điều đó cần sử dụng và lập trình *hàm thay dao tự động* M06.

Bạn không nên nhầm lẫn ý nghĩa của địa chỉ T được dùng với sự lựa chọn dao cố định với hàm T trong sự lựa chọn dao ngẫu nhiên. Địa chỉ T cố định có nghĩa là *chỉ số dao cắt thực tế của ổ dao*, địa chỉ T ngẫu nhiên có nghĩa là *chỉ số dao cắt của dao kế tiếp*. Sự gọi dao này được lập trình sớm hơn, hệ thống điều khiển có thể tìm dao đó trong khi dao hiện hành đang cắt gọt.

Đăng ký chỉ số dao cắt

Máy tính nói chung, và hệ thống CNC nói riêng, có thể xử lý dữ liệu rất nhanh với độ chính xác rất cao. Đối với CNC, dữ liệu cần thiết phải được nhập trước để máy tính hoạt động hiệu quả. Trong phương pháp lựa chọn dao ngẫu nhiên, người vận hành CNC tự do đặt *dao cắt bất kỳ* vào *ổ dao tùy ý của hộp chứa dao*, khi xác lập được đăng ký (ghi) vào bộ CNC, dưới dạng tham số của hệ thống điều khiển. Bạn không cần quá bận tâm về các tham số điều khiển, chỉ cần chấp nhận chúng là tập hợp các xác lập hệ thống. Sự đăng ký các chỉ số dao có màn hình nhập riêng.

Trong khi xác lập máy, người vận hành CNC đặt các dao cắt cần thiết vào ổ dao của hộp giữ dao, ghi lại các chỉ số (chỉ số dụng cụ cắt và chỉ số ổ dao tương ứng), và đăng ký thông tin này vào hệ thống. Thao tác này là một phần trong xác lập máy và có thể dùng một số đi tắt.

Định dạng lập trình

Định dạng lập trình đối với hàm T được dùng trên các hệ thống phay phụ thuộc vào số lượng tối đa các dụng cụ cắt khả dụng đối với

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

máy CNC. Hầu hết các trung tâm gia công đều có số đầu dưới 100, nhưng có máy rất lớn có hộp chứa dao (với vài trăm dao). Trong các ví dụ dưới đây sẽ sử dụng hàm T hai chữ số, trong khoảng T01 đến T99.

Trong chương trình, lệnh T01 sẽ gọi dao được xác định trong bảng dụng cụ là dao số 1; T02 gọi dao số 2; T20 gọi dao số 20, v.v... Có thể bỏ qua các số zero đứng trước, nếu cần thiết, chẳng hạn T01 có thể viết là T1; T02 là T2,... Các chữ số zero đứng sau *luôn luôn được ghi rõ*, ví dụ T20 phải viết là T20, nếu bỏ qua chữ số zero này, hệ thống sẽ nhầm lẫn, không được viết T2 thay cho T20 (do T2 tương ứng T02, không phải là T20).

Ổ dao rỗng

Trong thực tế, đôi khi yêu cầu trục chính không có dụng cụ cắt. Đối với mục đích này có thể gán ổ dao rỗng (không chứa dao cắt). Ổ dao đó phải có chỉ số đặc thù, dù không có dao được sử dụng. Nếu ổ dao trên hộp chứa dao hoặc trục chính không có dụng cụ cắt, cần có chỉ số dao *rỗng* để duy trì tính liên tục của các lần thay dao từ chi tiết này sang chi tiết khác. Ổ dao không chứa dao này thường được gọi là ổ dao *rỗng*.

Chỉ số của ổ dao rỗng (dao giả) cần chọn lớn hơn số lượng dao tối đa. Ví dụ, nếu trung tâm gia công có 24 ổ dao, ổ dao rỗng phải có chỉ số T25 hoặc cao hơn. Trong thực tế nên chọn ổ dao này theo chỉ số lớn nhất trong khoảng định dạng hàm T. Ví dụ, với định dạng hai chữ số, dao giả sẽ có chỉ số là T99; với định dạng ba chữ số, dao đó sẽ là T999. Số này rất dễ nhớ và dễ nhận biết trong chương trình.

Về nguyên tắc, không nên dùng ký hiệu T00 cho ổ dao *rỗng* (dao giả) - mọi dao *chưa được gán* đều có thể đăng ký là T00. Tuy nhiên, một số máy công cụ cho phép sử dụng T00 mà không gây ra sự nhầm lẫn.

HÀM THAY DAO – M06

Hàm dụng cụ cắt T, áp dụng cho trung tâm gia công CNC, sẽ *không* thực sự thay dao, hàm M06 phải được dùng trong chương trình để thực hiện điều đó. Mục đích của hàm thay dao là trao đổi dao trong trục chính với dao ở vị trí chờ. Công dụng của hàm T đối với hệ thống phay là quay hộp chứa dao và đặt dao đã chọn vào vị trí chờ, nơi thực hiện sự thay dao. *Sự tìm kiếm dao kế tiếp* xảy ra trong khi bộ điều khiển xử lý các block tiếp sau sự gọi hàm T.

➲ Ví dụ

```
N81 T01        T01 sẵn sàng, được đưa vào vị trí chờ
N82 M06        Đưa T01 vào trục chính.
N83 T02        T02 sẵn sàng, được đưa vào vị trí chờ
```

Ba block này tương đối đơn giản. Trong block N81, dao được lập địa chỉ là T01 trong chương trình sẽ được đưa vào vị trí chờ. Block kế tiếp, N82 sẽ kích hoạt sự thay dao – dao T01 được đưa vào trục chính, sẵn sàng cắt gọt. Tiếp sau sự thay dao là T02 trong block N83. Block này làm cho hệ điều khiển tìm dao kế tiếp, T02, để đưa vào vị trí chờ. Sự tìm kiếm này xảy ra *đồng thời* với dữ liệu chương trình tiếp sau block N83, thường là sự chuyển động dao đến vị trí cắt gọt trên chi tiết gia công. Ở đây không có sự tổn thất thời gian, phương pháp này bảo đảm thời lượng thay dao luôn luôn như nhau (được gọi là *thời gian phoi - phoi*).

Một số nhà lập trình thường muốn rút ngắn chương trình bằng cách lập trình lệnh thay dao *cùng với* sự tìm kiếm dao kế tiếp trong một block. Phương pháp này tiết kiệm một block cho từng dao trong chương trình.

```
N81 T01
N82 M06 T02
```

Kết quả là như nhau, sự lựa chọn tùy theo sở thích cá nhân.

Một số máy công cụ sẽ không chấp nhận kiểu rút gọn lập trình ba block. Nếu có ý nghi ngờ, bạn nên sử dụng kiểu ba block.

Điều kiện thay dao

Trước khi gọi hàm thay dao M06 trong chương trình, bạn cần tạo các điều kiện an toàn. Hầu hết các máy đều có đèn ở bảng điều khiển để quan sát dao cắt ở đúng vị trí thay dao.

Sự thay dao tự động và an toàn chỉ có thể xảy ra nếu thỏa các điều kiện dưới đây:

- ❑ Các trục máy đều trở về zero
- ❑ Trục chính phải lùi lại hoàn toàn
  (a) Trong trục Z tại zero máy đối với máy phay đứng
  (b) Trong trục Y tại zero máy đối với máy phay ngang.
- ❑ Các vị trí trục X và Y của dao phải được chọn ở vị trí hở.
- ❑ Dao kế tiếp phải được chọn trước bằng hàm T.

Chương trình mẫu minh họa sự thay dao ở giữa chương trình (T03 thay cho T02) được nêu trên các Hình 13.2 đến 13.4.

➲ Chương trình minh họa

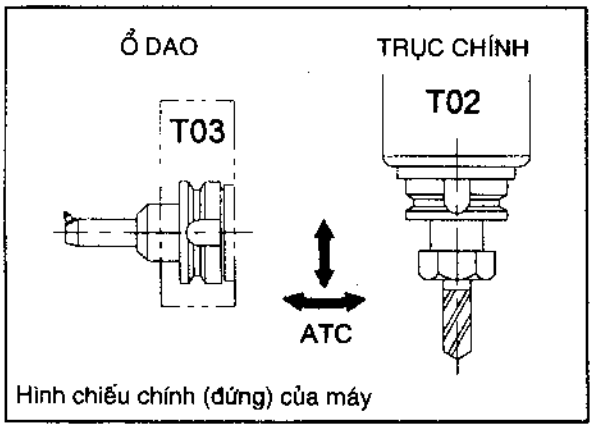
```
N51 ...                          (...T02 IN SPINDLE)
N52 ... T03 (... T03 READY FOR TOOL   CHANGE)
...                              (MACHINING WITH T02)
N75 G00 Z1.0                     (RETRACT FROM DEPTH)
N76 G28 Z1.0 M05                     (T02 COMPLETED)
N77 M01                              (OPTIONAL STOP)
                             (BLANK LINE BETWEEN TOOLS)
N78 T03                              (T03 CALL REPEATED)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

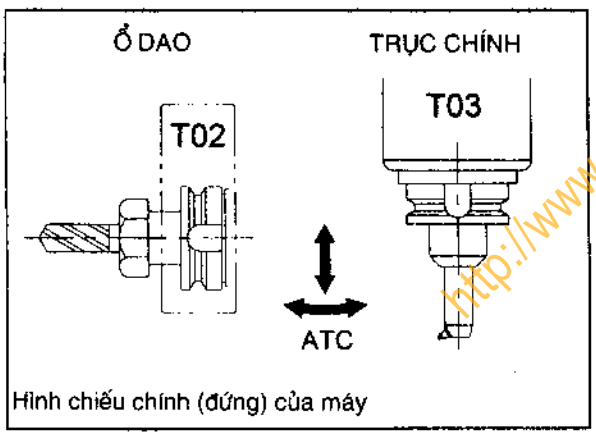
Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N79 M06 T02 (CUT WITH THE SECOND)
N80 G90 G54 G00 X-18.56 Y14.43 S700 M03 T04
N81 ...                    (MACHINING WITH T03)
```

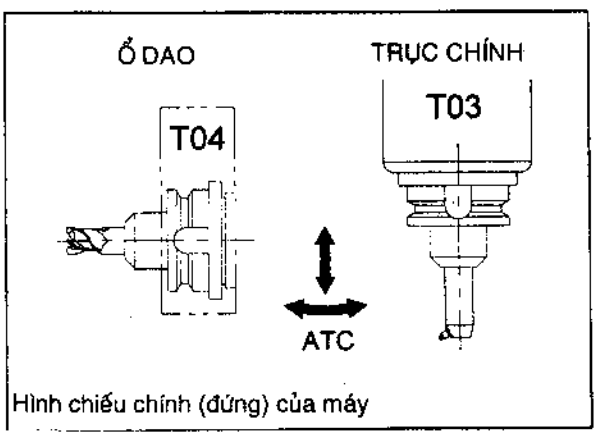
Trong ví dụ này, block N76 biểu thị kết thúc gia công, sử dụng dao T02, dao T02 chuyển đến vị trí zero máy trên trục Z, đồng thời dừng trục chính. Hàm dừng trục chính tùy chọn M01 được đưa vào block kế tiếp, N77.



Hình 13.2. *Sự thay dao tự động (ATC) – Các block N51 đến N78 (trạng thái hiện hành)*



Hình 13.3. *Sự thay dao tự động (ATC) – block N79 (thực sự thay dao)*



Hình 13.4. *Sự thay dao tự động (ATC) – block N80 (dao kế tiếp ở vị trí chờ)*

trong block tiếp theo, N78, lặp lại sự gọi T03, điều này là không cần thiết, nhưng có thể rất hữu ích để sau này gọi lại dao cắt đó. Block N79 là sự thay dao thực sự. T02 trong trục chính sẽ được thay bằng T03 hiện đang ở vị trí chờ.

Cuối cùng trong block N80, chuyển động nhanh trong các trục X và Y biểu thị chuyển động thứ nhất của T03, với trục chính ON. Bạn hãy chú ý, T04 ở cuối block này. Để tiết kiệm thời gian, dao kế tiếp phải được đưa vào vị trí chờ ngay sau khi thay dao.

Ngoài ra, bạn cần chú ý khi T02 hoàn tất trong block N77, dao này *vẫn còn trong trục chính*! Một số nhà lập trình không áp dụng phương pháp này. Nếu sự thay dao kết thúc ngay sau block G28 (trở về zero máy) và *trước* block M01, sẽ khó khăn hơn cho người vận hành khi cần lặp lại dao mới vừa hoàn tất gia công.

BỘ THAY DAO TỰ ĐỘNG – ATC

Trong một số ví dụ đã sử dụng *Automatic Tool Changer* (ATC). Có nhiều thiết kế ATC trên các máy CNC và chúng khác nhau tùy theo nhà chế tạo máy, do đó các phương pháp lập trình cũng khác nhau. Bộ thay dao sau khi xác lập sẽ tự động phân độ dao cắt đã lập trình theo thứ tự thích hợp. Tất cả đều do chương trình điều khiển. Nhà lập trình và người vận hành phải nắm vững kiểu loại ATC trên mọi trung tâm gia công trong xưởng cơ khí.

Hệ thống ATC thông dụng

Hệ thống *Automatic Tool Change* có thể có thanh hai nhánh, một nhánh đưa dao đến, nhánh kia lấy dao ra, dựa trên sự lựa chọn *nhớ ngẫu nhiên*, nghĩa là dao kế tiếp có thể chuyển đến vị trí chờ và sẵn sàng thay dao, trong khi dao hiện hành đang cắt gọt. Tính năng này bảo đảm thời lượng thay dao không đổi. Thời gian của chu kỳ thay dao có thể rất nhanh trên các máy CNC hiện đại, thường dưới 1 giây.

Số lượng dao đối đa có thể tải vào hộp chứa dao thay đổi trong khoảng rộng, từ 10 đến 400, thậm chí cao hơn. Trung tâm gia công đứng CNC cỡ nhỏ thường có 10 đến 30 dao.

Ngoài các tính năng thay dao, nhà lập trình và người vận hành CNC cần chú ý các vấn đề kỹ thuật khác có thể ảnh hưởng đến sự thay dao trong chương trình điều khiển. Chúng liên quan với các đặc tính vật lý của dao cắt khi được lắp vào giá dao:

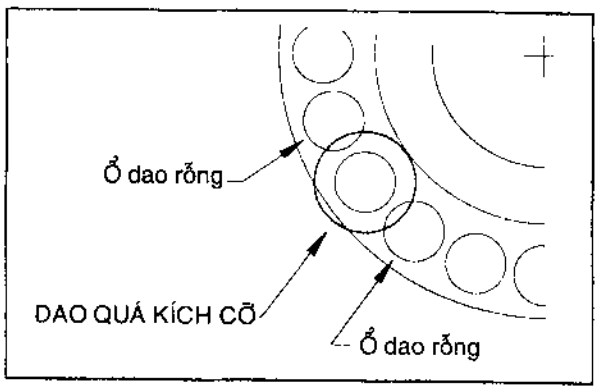
- Đường kính dao cực đại
- Chiều dài dao cực đại
- Trọng lượng dao cực đại

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Đường kính dao cực đại

Đường kính dao cực đại, có thể sử dụng mà không cần các xem xét đặc biệt, là do nhà chế tạo máy công cụ quy định, với giả thiết đường kính cực đại của cỡ dao xác định có thể được sử dụng trong *từng* ổ dao của hộp chứa dao. Nhiều nhà chế tạo máy cho phép sử dụng đường kính dao hơi lớn hơn với điều kiện hai ổ dao *kế cận* đều rỗng (Hình 13.5).



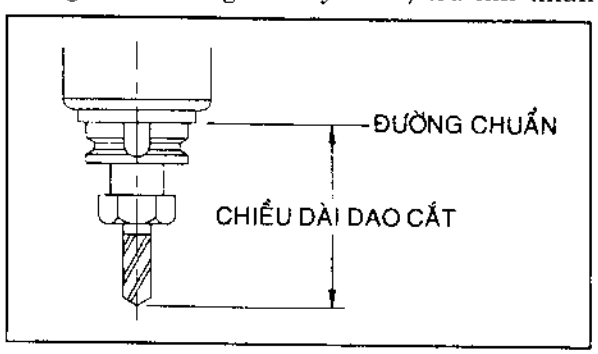
Hình 13.5. *Hai ổ kế cận phải rỗng nếu dùng dao có đường kính lớn.*

Ví dụ, đặc tính kỹ thuật của máy cho phép đường kính dao cực đại *với* các dao kế cận là 4 inch (100mm) Nếu cả hai ổ dao kế cận đều rỗng, đường kính dao cực đại có thể đến 5.9 inch (150mm). Bằng cách dùng các dao cắt lớn hơn đường kính theo quy định, số lượng dao thực sự trong ổ dao sẽ *giảm* tương ứng.

Các ổ dao kế cận phải rỗng để sử dụng dao quá khổ

Chiều dài dao cực đại

Chiều dài dao trong quan hệ với ATC, là phần nhô ra của dao cắt từ đường chuẩn trục chính hướng đến chi tiết. Chiều dài dao càng lớn, khoảng hở trục Z trong khi thay dao càng quan trọng. Sự tiếp xúc bất kỳ của dao với máy, với đồ gá, hoặc với chi tiết gia công đều không được phép. Điều này rất nguy hiểm, không đủ thời gian để dừng chu kỳ ATC, trừ khi nhấn *công tắc Emergency*, nhưng cũng có thể không kịp Hình 13.6 minh họa chiều dài dao.



Hình 13.6. *Chiều dài dao cắt.*

Trọng lượng dao cực đại

Hầu hết các nhà lập trình đều xét đường kính và chiều dài dụng cụ cắt khi viết chương trình mới. Tuy nhiên, một số nhà lập trình có thể quên xem xét *trọng lượng* toàn phần của dao cắt. Trọng lượng dao nói chung không gây ra sự khác biệt trong lập trình, do hầu hết các dao đều nhẹ hơn trọng lượng cực đại cho phép. Bạn cần nhớ ATC là thiết bị cơ học, do đó có các giới hạn tải xác định. Trọng lượng dao luôn luôn là *trọng lượng tổng* của dao *và* giá dao kể cả vòng lót, vít,...

Trong khi gá lắp không được vượt quá trọng lượng dao cho phép

Ví dụ, trung tâm gia công CNC có thể có trọng lượng dao tối đa cho phép là 22 pounds (10kg). Nếu cần sử dụng dao *chỉ hơi nặng hơn*, chẳng hạn 24 lb (10.8 kg), bạn cũng không được dùng ATC, bạn phải thay dao đó bằng tay. Trục chính của máy có thể chịu được dao hơi nặng hơn nhưng ATC không thể. Do thuật ngữ "hơi nặng hơn" chỉ có tính tương đối, cách tốt nhất trong trường hợp này là *không vượt quá trọng lượng dao tối đa cho phép*. Nếu có nghi ngờ, bạn hãy xem xét bảng các đặc tính kỹ thuật của máy.

Chu kỳ ATC

Nhà lập trình có thể không cần biết mọi chi tiết liên quan đến sự vận hành thực tế của bộ thay dao tự động (ATC), dù kiến thức này rất hữu ích trong nhiều ứng dụng. Mặt khác, người vận hành máy CNC phải biết chi tiết *từng bước* trong chu kỳ ATC.

Sự trình bày dưới đây sẽ nêu rõ chu kỳ ATC trên trung tâm gia công CNC kiểu đứng, nhưng có thể hơi khác đối với một số máy. Bạn cần nghiên cứu kỹ từng bước trong quy trình vận hành của ATC, kiến thức này sẽ giúp giải quyết vấn đề kẹt dao trong khi thay dao. Sự tổn thất thời gian do kẹt dao là hoàn toàn có thể tránh được. Một số máy có chu kỳ từng bước khả dụng với công tắc xoay đặc biệt, thường ở gần hộp chứa dao.

Trong ví dụ này, bộ thay dao với hệ thống quay hai nhánh sẽ được sử dụng, lấy dao cắt từ vị trí chờ thay cho dao đang trong trục chính của máy.

ATC là quy trình thực thi các bước theo thứ tự khi hàm thay dao M06 được lập trình. Mọi bước được trình bày đều có tính cơ bản, nhưng

http://www.me-cad.com/

 Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

không nhất thiết phải là tiêu chuẩn đối với từng trung tâm gia công CNC.

1. Trục chính định hướng.
2. Ổ dao đi xuống.
3. Thanh hai nhánh quay 60^0 CCW.
4. Nhả kẹp chặt dao (trong hộp chứa dao và trục chính).
5. Thanh hai nhánh dịch chuyển xuống
6. Thanh hai nhánh quay 180^0 CW.
7. Thanh hai nhánh đi lên.
8. Dao được kẹp chặt.
9. Thanh hai nhánh quay 60^0 CW.
10. Thanh răng trở về.
11. Ổ dao đi lên.

Ví dụ này chỉ cung cấp thông tin tổng quát – với tính logic có thể thích ứng với từng máy công cụ. Sổ tay hướng dẫn sử dụng máy thường nêu rõ các chi tiết về ATC.

Bất kể máy công cụ được sử dụng, để thực hiện ATC một cách chính xác cần có 2 điều kiện:

- Trục chính phải dừng lại (với hàm M05).
- Trục thay dao phải ở vị trí gốc (vị trí quy chiếu máy).

Đối với trung tâm gia công CNC kiểu đứng, trục thay dao là trục Z, trung tâm CNC kiểu ngang là trục Y. Hàm M06 cũng dừng trục chính, nhưng bạn không được tính đến hàm này. Bạn phải dừng trục chính với hàm M05 (spindle stop) *trước khi* thực thi chu kỳ thay dao.

http://www.me-cad.com/

Sự vận hành MDI

Từng bước của chu kỳ thay dao còn có thể được thực thi thông qua MDI (*Manual Data Input* – nhập dữ liệu bằng tay), sử dụng các hàm M đặc biệt. Các hàm này chỉ được dùng trong sửa chữa, bảo dưỡng, thông qua sự vận hành MDI và *không thể* sử dụng trong chương trình CNC. Lợi ích của tính năng này là có thể tìm nguyên nhân và giải quyết vấn đề phát sinh khi thay dao. Bạn nên xem sổ tay hướng dẫn của từng máy để biết chi tiết về các hàm này.

LẬP TRÌNH ATC

Có thể có nhiều khả năng liên quan với bộ thay dao tự động. Một số khả năng quan trọng gồm số lượng dao được dùng, chỉ số dao được đăng ký cho trục chính (nếu có) khi bắt đầu quy trình gia công, có cần thay dao bằng tay không, có dùng dao quá khổ không,...

Trong các ví dụ kế tiếp, một số tùy chọn sẽ được giới thiệu, các ví dụ này có thể được sử dụng một cách trực tiếp, nếu máy CNC sử dụng đúng định dạng đó, hoặc chúng có thể được *chỉnh sửa* phù hợp với môi trường làm việc cụ thể. Đối với các ví dụ dưới đây, một số điều kiện phải được thiết lập để hiểu chủ đề lập trình thay dao.

Để lập trình ATC thành công, toàn bộ điều cần thiết là định dạng lập trình cho *ba dao cắt*, dao *thứ nhất* được dùng, các dao được dùng ở *giữa* chương trình và dao *cuối cùng* được dùng trong chương trình. Để hiểu rõ toàn bộ khái niệm này, các ví dụ sẽ chỉ sử dụng bốn chỉ số dao – mỗi chỉ số dao sẽ *biểu thị* một trong bốn định dạng lập trình khả dụng.

- T01 ... Sự gán dao biểu thị dao thứ nhất được dùng trong chương trình CNC.
- T02 ... Sự gán dao biểu thị dao bất kỳ trong chương trình CNC giữa dao thứ nhất và dao cuối.
- T03 ... Sự gán dao biểu thị dao cuối cùng trong chương trình CNC.
- T99 ... Sự gán dao biểu thị ổ dao rỗng.

Trong tất cả các ví dụ đó, sẽ luôn luôn sử dụng ba dao cắt, ổ dao rỗng chỉ dùng khi có yêu cầu. Các ví dụ này sẽ minh họa khái niệm của nhiều ứng dụng ATC khả dĩ. Một tình huống có thể xảy ra là chỉ sử dụng *một dao cắt* trong chương trình CNC.

Lập trình một dao cắt

Một số nguyên công đặc biệt có thể chỉ yêu cầu một dao cắt để gia công. Trong trường hợp đó, dao cắt thường được lắp trong trục chính khi xác lập máy do đó trong chương trình không cần gọi hoặc thay dao khác:

```
O1401 (FIRST TOOL IN THE SPINDLE AT     START)
N1 G20                                 (INCH MODE)
N2 G17 G40 G80                         (SAFE BLOCK)
N3 G90 G54 G00 X.. Y.. S.. M03 (TOOL   MOTION)
...
```

<...T01 hoạt động...>

```
...
N26 G00 Z.. M09              (T01 MACHINING DONE)
N27 G28 Z.. M05                    (T01 TO Z-HOME)
N28 G00 X.. Y..                 (SAFE XY POSITION)
N29 M30                           (END OF PROGRAM)
%
```

Trừ khi thay chi tiết gia công, dao này luôn luôn trong trục chính để cắt gọt.

Lập trình nhiều dao cắt

Gia công chi tiết sử dụng nhiều dao là phương pháp phổ biến trên các máy công cụ CNC. Từng dao được đưa vào trục chính khi có yêu cầu, sử dụng các quy trình CNC khác nhau.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Theo quan điểm lập trình, các phương pháp thay dao này không tác động đến phần cắt gọt của chương trình, chỉ tác động đến *sự khởi đầu* của dao (trước khi gia công) và *sự kết thúc* của dao (sau khi gia công).

Như đã đề cập, dao được yêu cầu chỉ có thể được thay tự động nếu trục Z ở zero máy (trung tâm gia công đứng) hoặc trục Y ở zero máy (trung tâm gia công ngang). Vị trí dao trong các trục còn lại chỉ quan trọng đối với sự an toàn khi thay dao, không được phép có sự tiếp xúc của dao cắt với máy, đồ gá, hoặc chi tiết. Các ví dụ tiếp theo được định dạng cho các máy kiểu đứng. Một số chương trình sử dụng trả về zero máy cho *tất cả* các trục khi kết thúc dao cắt cuối cùng, ví dụ:

```
...
N393 G00 Z.. M09          (CURRENT TOOL WORK DONE)
N394 G28 Z.. M05          (CURRENT TOOL TO Z HOME)
N395 G28 X.. Y..          (CURRENT TOOL TO XY HOME)
N396 M30                  (END OF PROGRAM)
%
```

Về kỹ thuật, không có gì sai trong phương pháp này, nhưng có thể tốn nhiều thời gian khi gia công hàng loạt nhiều chi tiết. Phương pháp thường dùng là thực hiện thay dao *phía trên* vị trí dao cắt cuối cùng, hoặc dịch chuyển dao *ra xa* chi tiết, đến vị trí *an toàn*. Phương pháp này được minh họa trong các ví dụ giới thiệu các phương pháp khởi đầu chương trình, liên quan với các phương pháp thay dao khác nhau.

Theo dõi các dao cắt

Nếu thao tác thay dao là đơn giản, cũng có thể dễ dàng theo dõi vị trí từng dao vào thời điểm bất kỳ. Trong các ví dụ dưới đây sẽ thực hiện sự thay dao phức tạp hơn. Sự theo dõi dao nào đang chờ, dao nào đang trong trục chính, có thể được thực hiện bằng bảng ba cột với chỉ số block, dao chờ và dao trong trục chính.

Chỉ số block	Dao chờ	Dao trong trục chính

Để ghi vào bảng này, bạn hãy bắt đầu từ đỉnh chương trình và tìm từng lần xuất hiện địa chỉ T và hàm M06. Mọi dữ liệu khác đều không liên quan. Trong ví dụ O142, bảng này sẽ được ghi để minh họa công dụng thực tiễn của bảng.

Dao bất kỳ trong trục chính – không phải dao thứ nhất

Đây là phương pháp lập trình ATC thông dụng nhất. Người vận hành đặt tất cả các dao vào hộp chứa dao, ghi các xác lập nhưng để lại dao cuối cùng trong trục chính. Trên hầu hết các máy, dao này không phải là dao thứ nhất. nhà lập trình tương hợp phương pháp thay dao này trong chương trình. Ví dụ tiếp theo có lẽ là một trong các đoạn chương trình hữu dụng nhất trong công việc hàng ngày. Mọi hoạt động đều được liệt kê trong phần ghi chú:

```
O1402              (ANY TOOL IN SPINDLE AT START)
                   (**** NOT THE FIRST TOOL ****)
N1 G20                                 (INCH MODE)
N2 G17 G40 G80 T01                  (GET T01 READY)
N3 M06                             (T01 TO SPINDLE)
N4 G90 G54 G00 X.. Y.. S.. M03 T02
                                     (T02 READY)
N5 G43 Z.. H01 M08                 (APPROACH WORK)
...
```

<... T01 đang làm việc... >

```
...
N26 G00 Z.. M09              (T01 MACHINING DONE)
N27 G28 Z.. M05                 (T01 TO Z HOME)
N28 G00 X.. Y..              (SAFE XY POSITION)
N29 M01                        (OPTIONAL STOP)
N30 T02                     (T02 CALL REPEATED)
N31 M06                        (T02 TO SPINDLE)
N32 G90 G00 G54 X.. Y.. S.. M03 T03
                                     (T03 READY)
N33 G43 Z.. H02 M08                (APPROACH WORK)
...
```

<... T02 đang làm việc... >

```
...
N46 G00 Z.. M09              (T02 MACHINING DONE)
N47 G28 Z.. M05                 (T02 TO Z HOME)
N48 G00 X.. Y..              (SAFE XY POSITION)
N49 M01                        (OPTIONAL STOP)
N50 T03                     (T03 CALL REPEATED)
N51 M06                        (T03 TO SPINDLE)
N52 G90 G00 G54 X.. Y.. S.. M03 T01
                                     (T01 READY)
N53 G43 Z.. H03 M08                (APPROACH WORK)
..
```

<... T03 đang làm việc ... >

```
...
N66 G00 Z.. M09              (T03 MACHINING DONE)
N67 G28 Z.. M05                 (T03 TO Z HOME)
N68 G00 X.. Y..              (SAFE XY POSITION)
N69 M30                        (END OF PROGRAM)
%
```

Bảng 13.1 nêu rõ trạng thái của các dao cắt *chỉ* đối với *chi tiết thứ nhất*, "?" biểu thị chỉ số dao cắt bất kỳ.

Bảng 13.1. *Thứ tự dao gia công chi tiết thứ nhất*

Chỉ số block	Dao đang chờ	Dao trong trục chính
N1	?	?
N2	T01	?
N3	?	T01
N4	T02	T01
T01 ĐANG LÀM VIỆC		
N30	T02	T01
N31	T01	T02

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chỉ số block	Dao đang chờ	Dao trong trục chính
N32	T03	T02
T02 ĐANG LÀM VIỆC		
N50	T03	T02
N51	T02	T03
N52	T01	T03
T03 ĐANG LÀM VIỆC		

Khi *chi tiết thứ hai* được gia công và chi tiết kế tiếp bất kỳ, sự theo dõi dao là liên tục và được đơn giản hóa. Bạn hãy so sánh Bảng 13.2 với Bảng 13.1, trên Bảng 13.2 không có dấu hỏi "?".

Bảng 13.2. *Thứ tự dao gia công chi tiết kế tiếp.*

Chỉ số block	Dao đang chờ	Dao trong trục chính
N1	T01	T03
N2	T01	T03
N3	T03	T01
N4	T02	T01
T01 ĐANG LÀM VIỆC		
N30	T02	T01
N31	T01	T02
N32	T03	T02
T02 ĐANG LÀM VIỆC		
N50	T03	T02
N51	T02	T03
N52	T01	T03
T03 ĐANG LÀM VIỆC		

Các ví dụ đã nêu ở đây đều sử dụng phương pháp này hoặc có thay đổi chút ít. Đối với hầu hết các công việc, có lẽ không cần thay dao ở vị trí an toàn XY, nếu khu vực làm việc không bị cản trở. Bạn hãy nghiên cứu phương pháp này trước để hiểu rõ hơn về logic của một số phương pháp tiên tiến hơn.

Bạn hãy chú ý một số chú thích đối với chương trình ví dụ O1402. Bạn cần luôn luôn lập trình dừng tùy chọn M01 *trước* sự thay dao – sẽ dễ dàng hơn khi dùng lại dao này, nếu cần. Ngoài ra bạn hãy chú ý phần bắt đầu của từng dao đều có sự tìm kiếm dao kế tiếp. Dao trong block chứa chuyển động thứ nhất đã được gọi sẵn – bạn hãy so sánh block N4 với N30 và N32 với N50. Có hai lý do để lặp lại sự tìm kiếm dao ở phần đầu của từng dao. Thứ nhất, chương trình dễ đọc hơn (biết dao sẽ đến trục chính); thứ hai, cho phép *sự lặp lại* dao đó, bất kể dao nào đang trong trục chính.

Dao thứ nhất trong trục chính

Chương trình cũng có thể bắt đầu với dao thứ nhất trong trục chính, đây là sự thực hành phổ biến khi lập trình ATC. Dao thứ nhất trong chương trình phải được tải vào trục chính khi xác lập máy. Trong chương trình, dao thứ nhất được gọi đến vị trí chờ trong khi dao *cuối cùng*, không phải dao *đầu tiên*. Sau đó, sự thay dao sẽ được yêu cầu trong các block cuối của chương trình. Dao thứ nhất trong chương trình phải là dao *đầu tiên* đối với *mọi chi tiết* trong một mẻ (đợt) gia công.

```
O1403          (FIRST TOOL IN SPINDLE AT START)
N1 G20                                (INCH MODE)
N2 G17 G40 G80 T02                 (GET T02 READY)
N3 G90 G54 G00 X.. Y.. S.. M03
N4 G43 Z.. H01 M08                 (APPROACH WORK)
...
```

< ... T01 đang làm việc... >

```
...
N26 G00 Z.. M09              (T01 MACHINING DONE)
N27 G28 Z.. M05                    (T01 TO Z HOME)
N28 G00 X.. Y..                 (SAFE XY POSITION)
N29 M01                            (OPTIONAL STOP)

N30 T02                          (T02 CALL REPEATED)
N31 M06                             (T02 TO SPINDLE)
N32 G90 G54 G00 X.. Y.. S.. M03 T03
                                          (T03 READY)
N33 G43 Z.. H02 M08                (APPROACH WORK)
...
```

< ... T02 đang làm việc ... >

```
...
N46 G00 Z.. M09              (T02 MACHINING DONE)
N47 G28 Z.. M05                    (T02 TO Z HOME)
N48 G00 X.. Y..                 (SAFE XY POSITION)
N49 M01                            (OPTIONAL STOP)

N50 T03                          (T03 CALL REPEATED)
N51 M06                             (T03 TO SPINDLE)
N52 G90 G54 G00 X.. Y.. S.. M03 T01
                                          (T01 READY)
N53 G43 Z.. H03 M03                (APPROACH WORK)
...
```

< ... T03 đang làm việc ... >

```
...
N66 G00 Z.. M09              (T03 MACHINING DONE)
N67 G28 Z.. M05                    (T03 TO Z HOME)
N68 G00 X.. Y..                 (SAFE XY POSITION)
N69 M06                            (T01 TO SPINDLE)
N70 M30                            (END OF PROGRAM)
%
```

Phương pháp này vẫn có nhược điểm. Do luôn luôn có dao trong trục chính, dao có thể gây cản trở trong khi gá lắp hoặc thay chi tiết. Giải pháp là lập trình sự thay dao sao cho không có dao trong trục chính trong khi gá lắp chi tiết (điều kiện *trục chính rỗng*)

Không có dao trong trục chính

Trục chính rỗng khi bắt đầu và kết thúc gia công chi tiết sẽ có năng suất thấp hơn so với khi khởi động đã có dao thứ nhất trong trục chính. Thêm một lần thay dao sẽ tăng thời gian chu kỳ. Trục chính rỗng khi khởi đầu chỉ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

được khi dùng kiểu lập trình này, chẳng hạn khôi phục khoảng trống phía trên chi tiết nơi dao chiếm chỗ. Khoảng trống này có thể có ích, ví dụ để tháo chi tiết ra ngoài bằng móc treo hoặc tời nâng. Định dạng lập trình trong trường hợp đó không mấy khác biệt so với ví dụ trước, nhưng phải *thêm một lần thay dao* ở cuối chương trình. Sự thay dao này đưa dao thứ nhất trở lại trục chính, để bảo đảm chương trình lặp lại một cách ổn định.

```
O1404              (NO TOOL IN SPINDLE AT START)
N1 G20                                (INCH MODE)
N2 G17 G40 G80 T01                 (GET T01 READY)
N3 M06                             (T01 TO SPINDLE)
N4 G90 G54 G00 X.. Y.. S.. M03 T02
                                      (T02 READY)
N5 G43 Z.. H01 M08                 (APPROACH WORK)
...
```

< ... T01 đang làm việc ... >

```
...
N26 G00 Z.. M09           (T01 MACHINING DONE)
N27 G28 Z.. M05                 (T01 TO Z HOME)
N28 G00 X.. Y..               (SAFE XY POSITION)
N29 M01                         (OPTIONAL STOP)
N30 T02                      (T02 CALL REPEATED)
N31 M06                         (T02 TO SPINDLE)
N32 G90 G54 G00 X.. Y.. S.. M03 T03
                                      (T03 READY)
N33 G43 Z.. H02 M08            (APPROACH WORK)
...
```

<... T02 đang làm việc ...>

```
...
N46 G00 Z.. M09           (T02 MACHINING DONE)
N47 G28 Z.. M05                 (T02 TO Z HOME)
N48 G00 X.. Y..               (SAFE XY POSITION)
N49 M01                         (OPTIONAL STOP)
N50 T03                      (T03 CALL REPEATED)
N51 M06                         (T03 TO SPINDLE)
N52 G90 G54 G00 X.. Y.. S.. M03 T99
                                      (T99 READY)
N53 G43 Z.. H03 M08            (APPROACH WORK)
...
```

<... T03 đang làm việc... >

```
...
N66 G00 Z.. M09           (T03 MACHINING DONE)
N67 G28 Z.. M05                 (T03 TO Z-HOME)
N68 G00 X.. Y..               (SAFE XY POSITION)
N69 M06                         (T99 TO SPINDLE)
N70 M30                         (END OF PROGRAM)
%
```

Dao thứ nhất trong trục chính với sự thay dao bằng tay

Trong ví dụ kế tiếp, dao thứ hai đại diện cho dao *bất kỳ ở khoảng giữa* chương trình sử dụng không dưới ba dao cắt. Dao này có thể quá nặng hoặc quá dài và không thể phân độ thông qua chu kỳ ATC, cần lắp bằng tay. Sự thay dao này có thể do người vận hành thực hiện, nhưng chỉ với *chương trình hỗ trợ thay dao bằng tay*. Để đạt được điều này, cần sử dụng sự dừng chương trình M00 với dòng ghi chú hợp lý, mô tả lý do tạm dừng. Sự dừng tùy chọn M01 không an toàn, bạn nên dùng M00, do lệnh này *luôn luôn* dừng máy không cần sự can thiệp của người vận hành.

Bạn hãy theo dõi ví dụ trước một cách cẩn thận, để hiểu phương pháp thay dao bằng tay khi dao thứ nhất trong trục chính. T02 trong ví dụ này sẽ được người vận hành CNC thay bằng tay.

```
O1405        (FIRST TOOL IN SPINDLE AT START)
N1 G20                                (INCH MODE)
N2 G17 G40 G80 T99                 (GET T99 READY)
N3 G90 G54 G00 X.. Y.. S.. M03
N4 G43 Z.. H01 M08                 (APPROACH WORK)
...
```

<...T01 đang làm việc...>

```
...
N26 G00 Z.. M09           (T01 MACHINING DONE)
N27 G28 Z.. M05                 (T01 TO Z HOME)
N28 G00 X.. Y..               (SAFE XY POSITION)
N29 M01                         (OPTIONAL STOP)
N30 T99                      (T99 CALL REPEATED)
N31 M06                         (T99 TO SPINDLE)
N32 T03                            (T03 READY)
N33 M00          (STOP AND LOAD T02 MANUALLY)
N34 G90 G54 G00 X.. Y.. S.. M03
                                   (NO NEXT TOOL)
N35 G43 Z.. H02 M08            (APPROACH WORK)
...
```

<... T02 đang làm việc... >

```
...
N46 G00 Z.. M09           (T02 MACHINING DONE)
N47 G28 Z.. M05                 (T02 TO Z HOME)
N48 G00 X.. Y..               (SAFE XY POSITION)
N49 M19                   (SPINDLE ORIENTATION)
N50 M00          (STOP AND UNLOAD T02 MANUALLY)
N51 T03                      (T03 CALL REPEATED)
N52 M06                         (T03 TO SPINDLE)
N53 G90 G54 G00 X.. Y.. S.. M03 T01
                                      (T01 READY)
N54 G43 Z.. H03 M08            (APPROACH WORK)
...
```

< ... T03 đang làm việc ... >

```
...
N66 G00 Z.. M09           (T01 MACHINING DONE)
N67 G28 Z.. M05                 (T03 TO Z HOME)
N68 G00 X.. Y..               (SAFE XY POSITION)
N69 M01                         (OPTIONAL STOP)
N70 M06                         (T01 TO SPINDLE)
N71 M30                         (END OF PROGRAM)
%
```

Bạn hãy chú ý hàm M19. Hàm này định hướng trục chính đến đúng vị trí như khi sử dụng chu kỳ thay dao tự động. Người vận hành CNC sau đó có thể thay dao hiện hành bằng dao kế tiếp và vẫn duy trì hướng vị trí dao. Điều này là rất quan trọng đối với các chu kỳ doa khi lưỡi cắt cần được định vị cách xa bề mặt đã gia công. Nếu dùng cán doa dạng thanh, cần chỉnh thẳng hàng với đầu cắt.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Không có dao trong trục chính với sự thay dao bằng tay

Chương trình mẫu dưới đây là biến thể của ví dụ trước, nhưng không có dao trong trục chính khi chương trình khởi động.

```
O1406          (NO TOOL IN SPINDLE AT START)
N1 G20                          (INCH MODE)
N2 G17 G40 G80 T01           (GET T01 READY)
N3 M06                      (T01 TO SPINDLE)
N4 G90 G54 G00 X.. Y.. S.. M03 T99
                                (T99 READY)
N5 G43 Z.. H01 M08          (APPROACH WORK)
...
```

<...T01 đang làm việc...>

```
...
N26 G00 Z.. M09          (T01 MACHINING DONE)
N27 G28 Z.. M05               (T01 TO Z HOME)
N28 G00 X.. Y..            (SAFE XY POSITION)
N29 M01                       (OPTIONAL STOP)

N30 T99                    (T99 CALL REPEATED)
N31 M06                       (T99 TO SPINDLE)
N32 T03                          (T03 READY)
N33 M00          (STOP AND LOAD T02 MANUALLY)
N34 G90 G54 G00 X.. Y.. S.. M03
                              (NO NEXT TOOL)
N35 G43 Z.. H02 M08         (APPROACH WORK)
...
```

< ... T02 đang làm việc ... >

```
...
N46 G00 Z.. M09          (T02 MACHINING DONE)
N47 G28 Z.. M05               (T02 TO Z HOME)
N48 G00 X.. Y..            (SAFE XY POSITION)
N49 M19                  (SPINDLE ORIENTATION)
N50 M00        (STOP AND UNLOAD T02 MANUALLY)

N51 T03                    (T03 CALL REPEATED)
N52 M06                       (T03 TO SPINDLE)
N53 G90 G54 G00 X.. Y.. S.. M03 T99
                                (T99 READY)
N54 G43 Z.. H03 M08         (APPROACH WORK)
...
```

<... T03 đang làm việc... >

```
...
N66 G00 Z.. M09          (T03 MACHINING DONE)
N67 G28 Z.. M05               (T03 TO Z HOME)
N68 G00 X.. Y..            (SAFE XY POSITION)
N69 M01                       (OPTIONAL STOP)
N70 M06                       (T99 TO SPINDLE)
N71 M30                       (END OF PROGRAM)
%
```

Dao thứ nhất trong trục chính và dao quá khổ

Đôi khi cần phải dùng dao có đường kính hơi lớn hơn so với đặc tính kỹ thuật của máy cho phép. Trong trường hợp đó, dao quá khổ *phải* trở về đúng ổ dao cũ trong hộp chứa dao và hai ổ dao kế cận phải rỗng. Bạn *không được dùng dao quá nặng*. Trong ví dụ O1407, dao lớn là T02.

```
O1407       (FIRST TOOL IN SPINDLE AT START)
N1 G20                          (INCH MODE)
N2 G17 G40 G80 T99           (GET T99 READY)
N3 G90 G54 G00 X.. Y.. S.. M03
N4 G43 Z.. H01 M08          (APPROACH WORK)
...
```

<... T01 đang làm việc... >

```
...
N26 G00 Z.. M09          (T01 MACHINING DONE)
N27 G28 Z.. M05               (T01 TO Z HOME)
N28 G00 X.. Y..            (SAFE XY POSITION)
N29 M01                       (OPTIONAL STOP)

N30 T99                    (T99 CALL REPEATED)
N31 M06                       (T99 TO SPINDLE)
N32 T02                          (T02 READY)
N33 M06                       (T02 TO SPINDLE)
N34 G90 G54 G00 X.. Y.. S.. M03
                              (NO NEXT TOOL)
N35 G43 Z.. H02 M08         (APPROACH WORK)
...
```

<... T02 đang làm việc ... >

```
...
N46 G00 Z.. M09          (T02 MACHINING DONE)
N47 G28 Z.. M05               (T02 TO Z HOME)
N48 G00 X.. Y..            (SAFE XY POSITION)
N49 M01                       (OPTIONAL STOP)

N50 M06       (T02 OUT OF SPINDLE TO THE SAME
                                        POT)
N51 T03                          (T03 READY)
N52 M06                       (T03 TO SPINDLE)
N53 G90 G54 G00 X.. Y.. S.. M03 T01
                                (T01 READY)
N54 G43 Z.. H03 M08         (APPROACH WORK)
...
```

< ... T03 đang làm việc ... >

```
...
N66 G00 Z.. M09          (T03 MACHINING DONE)
N67 G28 Z.. M05               (T03 TO Z HOME)
N68 G00 X.. Y..            (SAFE XY POSITION)
N69 M01                       (OPTIONAL STOP)
N70 M06                       (T01 TO SPINDLE)
N71 M30                       (END OF PROGRAM)
%
```

Không có dao trong trục chính và dao quá khổ

Đây là phiên bản thay dao kế tiếp, giả thiết không có dao trong trục chính khi chương trình khởi động, với dao kế tiếp *lớn hơn* một cách hợp lý so với đường kính dao cực đại cho phép. Trong trường hợp này, dao quá khổ phải trở về *đúng* ổ dao cũ. Điều quan trọng là các ổ dao kế bên phải rỗng (không chứa dao).

Cả hai dao kế cận trên hộp chứa dao phải rỗng để ở giữa có thể chứa dao quá khổ

Trong ví dụ O1408, T02 là dao quá khổ

```
O1408          (NO TOOL IN SPINDLE AT START)
N1 G20                          (INCH MODE)
N2 G17 G40 G80 T01           (GET T01 READY)
N3 M06                      (T01 TO SPINDLE)
N4 G90 G54 G00 X.. Y.. S.. M03 T99
                                (T99 READY)
N5 G43 Z.. H01 M08          (APPROACH WORK)
...
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

<... T01 đang làm việc... >

```
...
N26 G00 Z.. M09          (T01 MACHINING DONE)
N27 G28 Z.. M05              (T01 TO Z HOME)
N28 G00 X.. Y..           (SAFE XY POSITION)
N29 M01                     (OPTIONAL STOP)

N30 T99                   (T99 CALL REPEATED)
N31 M06                     (T99 TO SPINDLE)
N32 T02                         (T02 READY)
N33 M06                     (T02 TO SPINDLE)
N34 G90 G54 G00 X.. Y.. S.. M03 (NO NEXT TOOL)
N35 G43 Z.. H02 M08          (APPROACH WORK)
...
```

<... T02 đang làm việc... >

```
...
N46 G00 Z.. M09          (T02 MACHINING DONE)
N47 G28 Z.. M05              (T02 TO Z HOME)
N48 G00 X.. Y..           (SAFE XY POSITION)
N49 M01                     (OPTIONAL STOP)

N50 M06   (T02 OUT OF SPINDLE TO THE SAME POT)
N51 T03                         (T03 READY)
N52 M06                     (T03 TO SPINDLE)
N53 G90 G54 G00 X.. Y.. S.. M03 T99 (T99 READY)
N54 G43 Z.. H03 M08          (APPROACH WORK)
...
```

<... T03 đang làm việc ...>

```
...
N66 G00 Z.. M09          (T03 MACHINING DONE)
N67 G28 Z.. M05              (T03 TO Z HOME)
N68 G00 X.. Y..           (SAFE XY POSITION)
N69 M01                     (OPTIONAL STOP)
N70 M06                     (T99 TO SPINDLE)
N71 M30                     (END OF PROGRAM)
%
```

Các ví dụ nêu trên minh họa một số phương pháp lập trình ATC. Nhiệm vụ này không khó nếu biết rõ cơ chế thay dao trên trung tâm gia công.

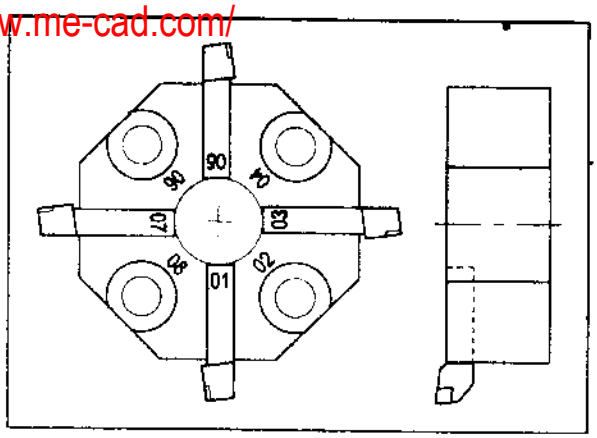
HÀM T ĐỐI VỚI MÁY TIỆN

Cho đến phần này đã trình bày hàm dụng cụ cắt áp dụng cho trung tâm gia công CNC. Các máy tiện cũng sử dụng hàm T nhưng với cấu trúc hoàn toàn khác.

Trạm dao trên máy tiện

Máy tiện băng máy nghiêng sử dụng hộc dao đa giác giữ tất cả các dao cắt trong và ngoài trên các ổ dao đặc biệt. *Các trạm dao này* tương tự hộp chứa dao trên trung tâm gia công, có thể chứa 8, 10, 12 dao hoặc nhiều hơn (Hình 13.7)

Do mọi dao cắt đều được giữ trong một trạm dao, dao được chọn để cắt gọt sẽ luôn luôn *dịch chuyển cùng với* tất cả các dao khác vào khu vực làm việc. Đây có thể là thiết kế đã lạc hậu nhưng vẫn còn thông dụng trong công nghiệp. Do có thể xảy ra khả năng va chạm giữa dao và máy hoặc chi tiết gia công, cần rất cẩn thận không chỉ với dao cắt *đang hoạt động* mà còn

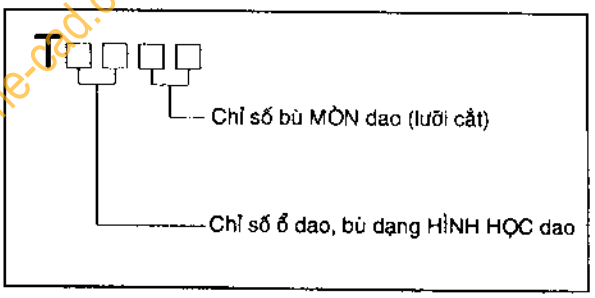


Hình 13.7. *Trạm dao hình bát giác (8-cạnh) trên máy tiện.*

với tất cả các dao khác trong trạm dao để tránh sự va chạm không đáng có.

Phân độ dao cắt

Để lập trình sự thay dao hoặc để *phân độ* dao cắt vào vị trí hoạt động, hàm T phải được lập trình theo định dạng thích hợp. Đối với máy tiện CNC, định dạng này gọi địa chỉ T, tiếp theo là bốn chữ số (Hình 13.8).



Hình 13.8. *Cấu trúc chỉ số dao 4-chữ số đối với máy tiện CNC.*

Điều quan trọng là hiểu rõ hàm này. Bốn chữ số được chia thành *hai cặp chữ số*, thay vì bốn chữ số riêng rẽ. Có thể bỏ qua các số zero đứng trước trong từng cặp chữ số. Mỗi cặp chữ số đều có ý nghĩa riêng:

Cặp *thứ nhất* (chữ số thứ nhất và thứ hai) điều khiển trạm phân độ dao cắt và sự bù dạng hình học.

➲ **Ví dụ**

T01xx – chọn dao cắt lắp trong vị trí 1 và kích hoạt bù dạng hình học số 1.

Cặp *thứ hai* (chữ số thứ ba và thứ tư) điều khiển bù mòn dao cắt được dùng với dao đã chọn.

➲ **Ví dụ**

Txx01 – chọn số đăng ký bù mòn dao là số 1

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Thường có thói quen tương hợp các cặp này nếu có thể. Ví dụ hàm T0101 sẽ chọn số 1 trên trạm dao, số 1 bù dạng hình học và số 1 bù mòn dao tương ứng. Định dạng này dễ nhớ nhưng chỉ được sử dụng nếu có một số bù được gán cho chỉ số dao cắt.

Nếu *hai hoặc nhiều số bù mòn dao* được dùng cho một dao, không thể tương hợp hai cặp số. Trong trường hợp đó, cần lập trình *hai hoặc nhiều* số bù mòn dao cho *cùng một* chỉ số trạm dao.

➲ **Ví dụ:**

T0101 Trạm dao 01,
bù dạng hình học 01 và bù mòn dao 01

➲ **Ví dụ:**

T0111 Trạm dao 01,
bù dạng hình học 01 và bù mòn dao 11

Cặp chữ số thứ nhất luôn luôn là chỉ số trạm dao *và* chỉ số bù hình học. Các ví dụ này giả định bù mòn dao 11 không dùng cho dao cắt khác. Nếu dao 11 được dùng với độ bù 11, cần chọn chỉ số bù mòn dao thích hợp khác, ví dụ 21, và lập trình là T0121. Hầu hết các bộ điều khiển đều có không dưới 32 số đăng ký bù dạng hình học và 32 số đăng ký bù mòn dao.

Các giá trị bù có thể áp dụng cho chương trình tiện CNC bằng cách đăng ký giá trị của chúng vào *offset registers* (đăng ký độ bù).

ĐĂNG KÝ BÙ DAO CẮT

Thuật ngữ *Offset* (bù) được dùng cho hai trường hợp *geometry offset* (bù dạng hình học) và *wear offset* (bù mòn dao). Vậy *offset* là gì? Sự khác biệt giữa chúng?

Trên hiển thị OFFSET của bộ điều khiển Fanuc, có sự lựa chọn giữa hai màn hình, chúng có hình thức *rất* giống nhau. Thứ nhất là màn hình *Geometry Offset* và thứ hai *Wear Offset*. Các Hình 13.9 và 13.10 minh họa hai màn hình này với các mục nhập mẫu.

OFFSET - GEOMETRY

No.	X-OFFSET	Z-OFFSET	Radius	Tip
01	-8.4290	-16.4820	0.0313	3
02	-8.4570	-14.7690	0.0000	0
03	-8.4063	-16.3960	0.0156	3
04	-8.4570	-12.6280	0.0000	0
05	-8.4350	-16.4127	0.0000	0
06	-9.8260	-13.2135	0.0313	2
07	0.0000	0.0000	0.0000	0

Hình 13.9. *Ví dụ về màn hình hiển thị GEOMETRY*

OFFSET - WEAR

No.	X-OFFSET	Z-OFFSET	Radius	Tip
01	0.0000	0.0000	0.0313	3
02	0.0000	0.0150	0.0000	0
03	0.0036	0.0000	0.0156	3
04	0.0000	-0.0250	0.0000	0
05	0.0010	-0.0022	0.0000	0
06	-0.0013	0.0000	0.0313	2
07	0.0000	0.0000	0.0000	0

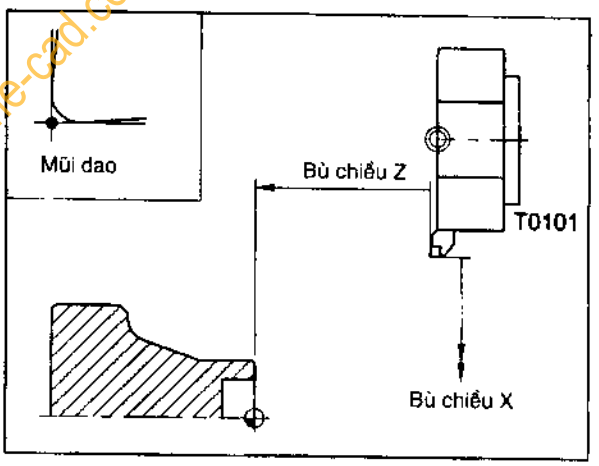
Hình 13.10. *Ví dụ về màn hình hiển thị WEAR*

Geometry Offset

Chỉ số bù hình học *luôn luôn* trùng với chỉ số trạm dao. Người vận hành đo và điền các giá trị bù hình học cho mọi dao cắt được dùng trong chương trình.

Giá trị bù GEOMETRY (hình học) luôn luôn đo từ vị trí zero của máy

Khoảng cách từ vị trí zero máy phản ánh khoảng cách từ điểm quy chiếu dao đến điểm quy chiếu chi tiết gia công. Hình 13.11 minh họa sự đo giá trị bù hình học cho dao cắt *ngoài*.



Hình 13.11. *Bù hình học dao cắt (tiện) ngoài*

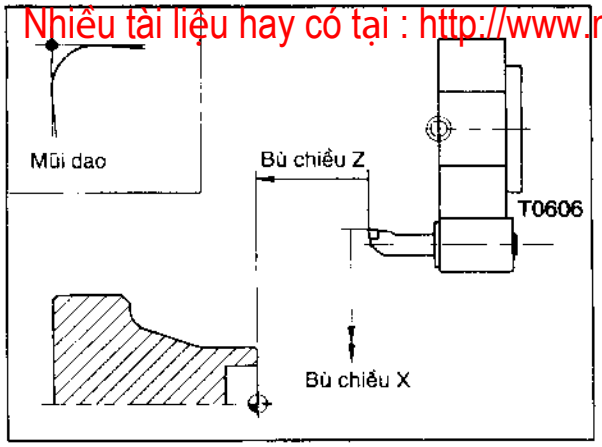
Mọi giá trị X đều có giá trị *đường kính* và được lưu theo số *âm* đối với máy tiện sau, kiểu băng máy nghiêng. Các giá trị trục X cũng là âm (có thể dùng giá trị dương nhưng không thuận tiện). Phương pháp tính giá trị bù hình học được trình bày trong công nghệ chế tạo máy, không thuộc lĩnh vực lập trình.

Hình 13.12 minh họa phương pháp đo độ bù hình học cho dao cắt *trong*.

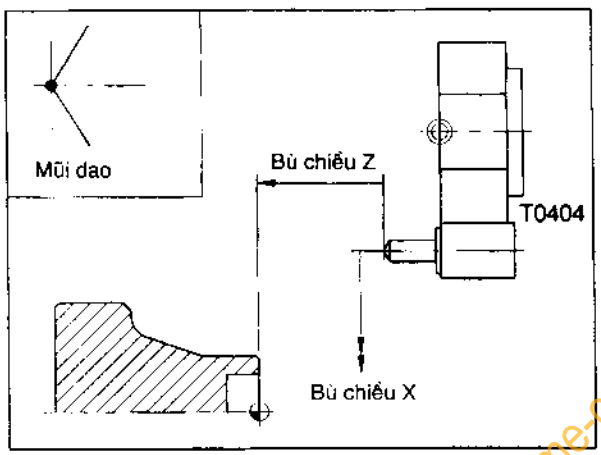
Khả năng thứ ba liên quan đến sự bù hình học được minh học trên Hình 13.13, nêu rõ sự bù hình học áp dụng cho dụng cụ cắt được dùng trên đường tâm trục chính (tại vị trí X0). Các

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 13.12. Bù hình học dao cắt (doa) trong



Hình 13.13. Bù hình học dao cắt theo đường tâm (khoan)

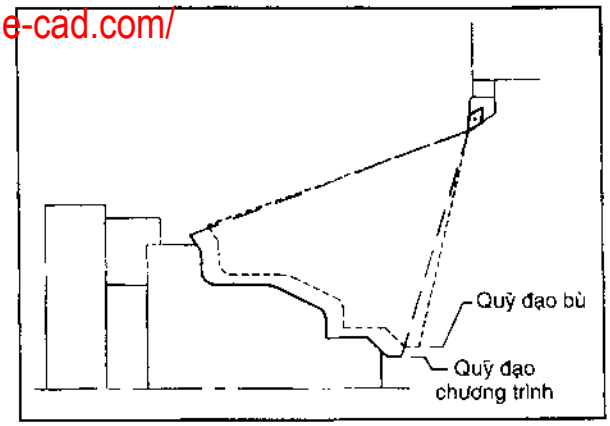
dụng cụ cắt này bao gồm mũi khoan tâm, mũi khoan, tarô ren, chuốt,.. Giá trị bù theo trục X của chúng sẽ luôn luôn như nhau.

Wear Offset

Khi viết chương trình CNC, các kích thước được dùng trong chương trình là kích thước trên bản vẽ hoàn tất. Ví dụ, kích thước 3.00 inch được lập trình là X3.0. Số này không phản ánh các dung sai kích thước. Các mục nhập chương trình X3.0, X3.00, X3.000, và X3.0000 đều có kết quả hoàn toàn *như nhau*. Cần làm gì để duy trì các dung sai kích thước? Cần làm gì với dao cắt bị mòn nhưng vẫn còn tốt để gia công thêm vài chi tiết? Câu trả lời là điều chỉnh, *tinh chỉnh*, quỹ đạo dao cắt để phù hợp với các điều kiện cắt gọt. Bản thân chương trình không thay đổi, chỉ áp dụng *Wear Offset* (bù mòn dao) cho dụng cụ cắt đã chọn.

Giá trị bù MÒN DAO là hiệu số giữa giá trị *được lập trình* và kích cỡ đo *thực tế* của chi tiết gia công.

Giá trị bù mòn dao chỉ có một mục đích – bù giữa giá trị lập trình, ví dụ đường kính 3.0



Hình 13.14. Quỹ đạo dao được lập trình và quỹ đạo dao với sự bù mòn dao

inch, và kích thước thực *được đo* khi kiểm tra, ví dụ 3.004. Hiệu số -.004 được nhập vào bộ đăng ký bù mòn dao. Đây là giá trị bù chuyên biệt *cặp chữ số thứ hai* của hàm dụng cụ cắt trong chương trình. Do chương trình sử dụng các giá trị đường kính theo trục X, giá trị bù cũng được nhập cho đường kính. Các chi tiết này rất hữu ích đối với người vận hành CNC và nhà lập trình.

Điều chỉnh wear offset

Để minh họa khái niệm điều chỉnh bù mòn dao trên máy tiện, T0404 trong chương trình sẽ được dùng làm ví dụ. Mục đích là đạt được đường kính ngoài 3.0 in và dung sai ±.0005. Giá trị ban đầu của độ bù mòn dao trong bộ đăng ký TXX04 sẽ là zero. Phần này của chương trình có dạng như sau:

```
N31 M01
...
N32 T0400 M42
N33 G96 S450 M03
N34 G00 G42 X3.0 Z0.1 T0404 M08
N35 G01 Z-1.5 F0.012
N36 ...
```

Khi kiểm tra (đo) chi tiết gia công, kết quả sẽ là một trong ba khả năng:

- ❑ Kích thước đúng
- ❑ Kích thước dư
- ❑ Kích thước thiếu

Nếu chi tiết được đo *đúng kích thước*, sẽ không cần sự can thiệp. Sự gá lắp dao và chương trình hoạt động chính xác. Nếu chi tiết *quá kích thước*, thường có thể cắt lại khi gia công đường kính *ngoài*. Đối với *đường kính trong*, sẽ áp dụng ngược lại. Sự cắt lại có thể gây hư hại độ bóng bề mặt. Nếu chi tiết bị *thiếu kích thước*, sẽ trở thành phế phẩm. Mục đích là tránh thiếu kích thước cho các chi tiết

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

kế tiếp. Bảng dưới đây liệt kê các khả năng với mọi khả năng có thể xảy ra:

Đo	Đường kính ngoài	Đường kính trong
ĐÚNG kích thước	Tốt	Tốt
DƯ kích thước	Có thể cắt lại	Phế phẩm
THIẾU kích thước	Phế phẩm	Có thể cắt lại

Dù chi tiết bị thiết hay dư kích thước, vẫn có biện pháp để tránh điều này xảy ra. Đó là *điều chỉnh giá trị bù mòn dao*. Ví dụ này dùng cho đường kính *ngoài*.

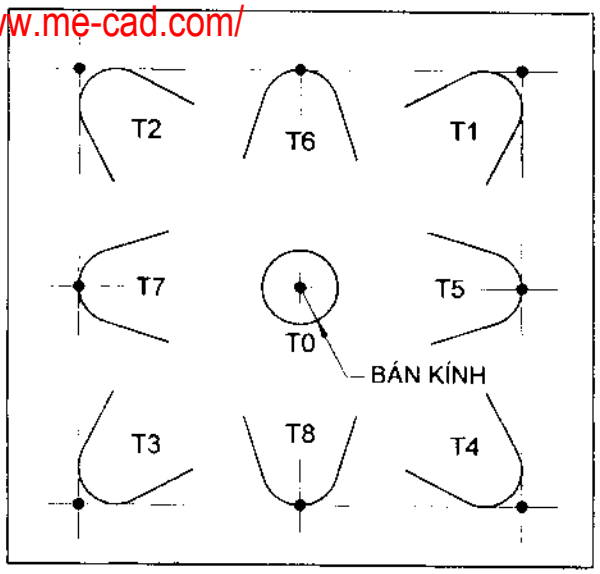
Đường kính ngoài X3.0 trong ví dụ này, khi đó có thể có kết quả là 3.004, có nghĩa là quá kích thước (đường kính) 0.004 in. Người vận hành, có trách nhiệm điều chỉnh giá trị bù, sẽ thay đổi giá trị 0.0000 hiện hành trong thanh ghi X bù mòn dao là 0.0040. Lần cắt kế tiếp sẽ làm cho chi tiết có kết quả đo trong phạm vi dung sai cho phép.

Nếu chi tiết bị thiếu kích thước, ví dụ 2.9990 in, giá trị bù mòn dao phải điều chỉnh là +.0010 theo chiều dương trên trục X, chi tiết vừa đo bị loại bỏ.

Nguyên tắc điều chỉnh bù mòn dao là logic. Nếu đường kính đã gia công *lớn hơn* kích thước bản vẽ cho phép, sự bù mòn dao được chỉnh theo giá trị *âm*, hướng đến đường tâm trục chính, và ngược lại. Nguyên tắc này áp dụng cho đường kính trong và đường kính ngoài. Sự khác biệt thực tiễn duy nhất là đường kính ngoài bị dư và đường kính trong bị hụt có thể cắt lại. Chương 33 sẽ trình bày chi tiết về bù mòn dao.

Các xác lập R và T

Các mục cuối cùng là cột R và cột T (*Geometry* và Wear). Các cột màn hình bù này chỉ hữu dụng trong khi gá lắp. *Cột R* là cột bán kính, *cột T* là cột định hướng mũi dao cắt (Hình 13.15).



Hình 13.15. *Các chỉ số định hướng mũi dao tùy ý được dùng với sự bù bán kính mũi dao (chế độ G41 hoặc G42)*

Nguyên tắc chính sử dụng các cột R và T là chúng chỉ có hiệu lực trong chế độ bù bán kính mũi dao. Nếu không có G41 hoặc G42 trong chương trình, giá trị trong các cột này sẽ không xuất hiện. Nếu dùng lệnh G41/G42, các giá trị khác zero đối với dao cắt đó phải được xác lập trong cả hai cột. Cột R yêu cầu *bán kính mũi dao* của dụng cụ cắt, cột T yêu cầu *chỉ số định hướng mũi dao* của dụng cụ cắt. Cả hai lệnh sẽ được trình bày chi tiết trong Chương 29. Các bán kính mũi dao phổ biến khi tiện và doa bao gồm:

1/64 inch = .0156 hoặc 0.4 mm

1/32 inch = .0313 hoặc 0.8 mm

3/64 inch = .0469 hoặc 1.2 mm

Các chỉ số mũi dao là tùy ý và biểu thị chỉ số định hướng dao cắt được dùng để tính toán giá trị bù bán kính mũi dao, bất kể xác lập dao trong trạm chứa dao cắt.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

14 CÁC ĐIỂM QUY CHIẾU

Trong các chương trước đã trình bày quan hệ cơ bản giữa hình học máy và gá lắp chi tiết, nhà lập trình CNC làm việc trong môi trường chính xác, các quan hệ toán học là rất quan trọng.

Có ba môi trường chính trong lập trình đòi hỏi thiết lập các quan hệ toán học:

Môi trường	Quan hệ
Máy	Máy công cụ + hệ thống điều khiển (bộ CNC)
Chi tiết	Chi tiết gia công + Bản vẽ + Vật liệu
Dao cắt	Giá giữ dao + dụng cụ cắt

Mỗi môi trường này đều độc lập với nhau. Nếu quan hệ là chưa rõ ràng, bạn hãy xem xét *các nguồn* của từng môi trường:

- MÁY CÔNG CỤ của nhà chế tạo, thường không có bộ điều khiển và dụng cụ cắt.

...môi trường này luôn luôn kết hợp với ...

- HỆ THỐNG ĐIỀU KHIỂN được chế tạo từ công ty chuyên môn hóa về ứng dụng điện tử cho máy công cụ. Họ thường không chế tạo máy công cụ hoặc dụng cụ cắt.
- CHI TIẾT (gia công) là thiết kế kỹ thuật đặc thù được thực hiện trong công ty không chế tạo máy công cụ, hệ thống điều khiển, dao cắt và giá giữ dao.
- DỤNG CỤ CẮT do công ty chuyên môn chế tạo, có hoặc không chế tạo các cán dao. Các công ty này không chế tạo máy công cụ và hệ thống điều khiển.

Các nguồn này phải tương hợp với nhau khi khách hàng mua máy công cụ. Thiết kế kỹ thuật (chi tiết) phải được gia công trên máy công cụ từ một nhà chế tạo, sử dụng hệ thống điều khiển của nhà chế tạo khác, dao cắt của công ty khác, và giá giữ dao từ nguồn thứ tư. Các nguồn này tương tự dàn nhạc gồm các nhạc công chưa từng chơi cùng nhau. Trong cả hai trường hợp, cần tạo ra sự hài hòa và tương hợp.

Điểm chung ở đây là cả ba môi trường này đều cần có "đội ngũ công tác". Họ phải cùng làm việc và *tương tác* với nhau.

Đối với các mục đích lập trình, các quan hệ và tương tác này đều dựa trên một mẫu số chung cho *từng* môi trường – *điểm quy chiếu*.

Điểm quy chiếu là vị trí cố định hoặc được chọn tùy ý trên máy, dụng cụ cắt, hoặc chi tiết (chuẩn máy, chuẩn định vị, hoặc chuẩn gia công). Điểm quy chiếu cố định là vị trí chính xác dọc theo hai hoặc nhiều trục, được thiết kế trong khi chế tạo hoặc gá lắp. Một số điểm quy chiếu do nhà lập trình thiết lập trong quá trình viết chương trình. Trong ba môi trường này, cần có ba điểm quy chiếu, mỗi điểm dùng cho một trong các nhóm:

- Điểm quy chiếu *máy* ...zero máy hoặc chuẩn máy
- Điểm quy chiếu *chi tiết* ...zero chương trình hoặc chuẩn chi tiết
- Điểm quy chiếu *dao* ...đỉnh dao hoặc chuẩn dao.

CÁC NHÓM ĐIỂM QUY CHIẾU

Nhóm thứ nhất là *máy công cụ CNC*, kết hợp giữa máy và hệ thống điều khiển. Những giá trị số liên quan đến máy công cụ bao gồm các kích thước, thông số kỹ thuật, các tham số, các khoảng, các định mức,... khi chi tiết được lắp vào đồ gá trên bàn máy, trong mâm cặp máy tiện, ống lót, tấm định vị,.. sẽ có nhóm số thứ hai cần khảo sát. *Các đặc tính chi tiết*, chẳng hạn kích cỡ, chiều cao, trọng lượng, đường kính, hình dạng,.. đều là đặc thù đối với quá trình gia công. Cuối cùng, nhóm số thứ ba liên quan đến *dụng cụ cắt*. Mỗi dụng cụ cắt đều có các tính năng riêng và các tính năng chung với các dụng cụ cắt khác.

Mọi giá trị số khả dụng đều có ý nghĩa – chúng không chỉ là các số – chúng là các *giá trị* thực, người vận hành và nhà lập trình phải làm việc một cách độc lập và hợp tác với nhau.

Quan hệ các nhóm điểm quy chiếu

Cốt lõi để lập trình CNC thành công là làm cho cả ba nhóm quy chiếu hài hòa với nhau. Mục đích này chỉ có thể đạt được khi hiểu rõ về các nguyên lý của điểm quy chiếu và cách thức hoạt động của chúng. Mỗi điểm quy chiếu có thể có hai đặc tính:

- Điểm quy chiếu *cố định*.
- Điểm quy chiếu *động* hoặc *linh hoạt*.

Điểm quy chiếu cố định do nhà chế tạo máy công cụ xác lập như một phần trong thiết kế máy và người sử dụng máy không thể thay đổi. Máy CNC có ít nhất một điểm quy chiếu cố định. Khi cần quyết định các điểm quy chiếu cho chi tiết hoặc dao cắt, nhà lập trình có thể

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

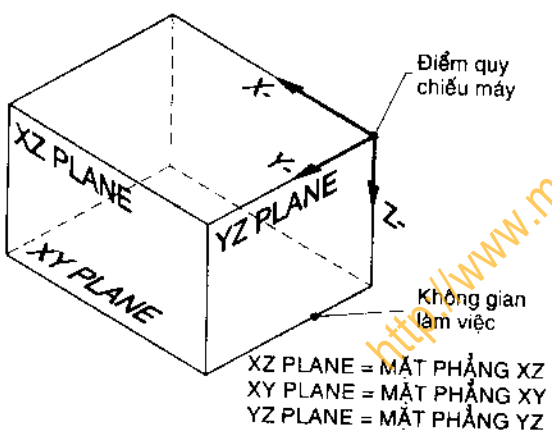
Nhiều tài liệu hay có tại : http://www.me-cad.com/

lựa chọn tương đối tự do. Điểm quy chiếu chi tiết (điểm zero chương trình) luôn luôn là điểm linh hoạt (động), vị trí thực của điểm này do nhà lập trình quyết định. Điểm quy chiếu đối với dao cắt đã lắp có thể là cố định hoặc linh hoạt, tùy theo thiết kế máy.

ĐIỂM QUY CHIẾU MÁY

Điểm zero máy, thường được gọi là *zero máy*, *chuẩn máy*, hoặc *vị trí quy chiếu máy*, là điểm gốc của hệ tọa độ máy. Vị trí của điểm này có thể khác nhau tùy theo nhà chế tạo máy, nhưng khác biệt rõ nhất là giữa các kiểu máy đứng hoặc ngang.

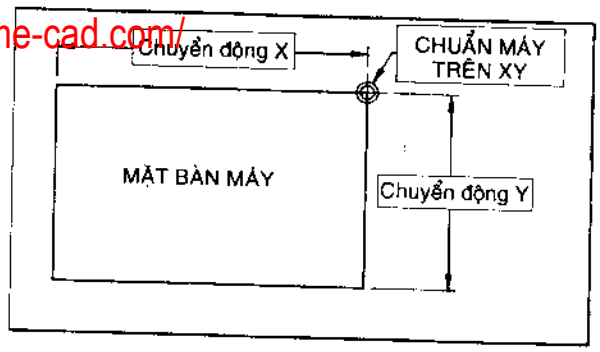
Trên tất cả các máy CNC sử dụng hệ tọa độ chuẩn, zero máy ở đầu *dương* của từng khoảng hành trình trục. Đối với trung tâm gia công đứng ba trục, bạn hãy quan sát phần mặt phẳng XY, thẳng xuống từ vị trí dao (đỉnh dao). Bạn hãy quan sát mặt phẳng XZ và mặt phẳng YZ. Ba mặt phẳng này vuông góc với nhau tạo thành *khối lập phương làm việc* hoặc *không gian làm việc* (Hình 14.1).



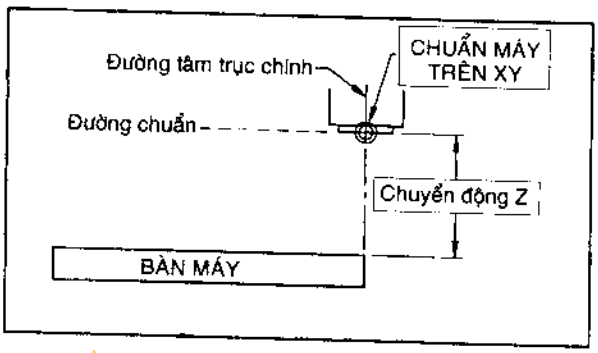
Hình 14.1. *Điểm quy chiếu máy và định hướng trục của trung tâm gia công đứng.*

Không gian làm việc đã nêu chỉ có tác dụng để hiểu về vùng làm việc tổng quát của máy CNC. Đối với lập trình và gá lắp, hầu hết các công việc đều được thực hiện mỗi lần với một hoặc hai trục. Để hiểu vùng làm việc và điểm zero máy trong mặt phẳng, bạn hãy quan sát máy từ phía trên xuống (mặt phẳng máy XZ) và từ phía trước (mặt phẳng máy YZ). Các Hình 14.2 và 14.3 minh họa hai phép chiếu này.

Ngoài ra, bạn hãy chú ý trong hình chiếu trước, đường nét đứt được gọi là *đường chuẩn*. Đây là vị trí tưởng tượng để lắp thân côn của cán dao và do nhà chế tạo máy xác lập. Bên trong trục chính là đoạn côn được gia công



Hình 14.2. *Mặt chiếu bằng của máy gia công đứng khi chiếu về phía bàn máy.*



Hình 14.3. *Mặt chiếu của máy gia công đứng khi chiếu từ phía trước.*

chính xác để lắp giá dao với dụng cụ cắt. Giá dao bất kỳ lắp trong trục chính đều có cùng vị trí. Chuyển động Z được minh họa sẽ rút ngắn với khoảng cách bằng phần dụng cụ cắt chìa ra ngoài. Vấn đề này sẽ được trình bày ở cuối Chương 14.

Trả về zero máy

Trong chế độ vận hành bằng tay, người vận hành CNC đưa các trục đến vị trí zero máy, và chịu trách nhiệm đăng ký vị trí này vào hệ thống điều khiển, nếu cần thiết. Bạn không được tắt máy khi các trục *đến rất gần* hoặc *ngay tại* vị trí zero máy. Ở quá gần vị trí này sẽ làm cho sự trả về zero máy trở nên khó khăn sau khi đóng điện (ON) cho máy. Khoảng 1 inch (25 mm) cách zero máy cho từng trục là vừa đủ. Quy trình để đạt đến zero máy gồm các bước sau:

1. Đóng điện cho hệ thống (cả máy và bộ điều khiển).
2. Chọn chế độ trả về zero máy
3. Chọn trục thứ nhất để dịch chuyển (thường là trục Z)
4. Lặp lại cho các trục còn lại.
5. Kiểm tra các đèn báo "vị trí zero"
6. Kiểm tra màn hình hiển thị vị trí
7. Xác lập hiển thị về zero, nếu cần thiết.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chú ý: với các lý do an toàn, đối với máy

tâm gia công, trục thứ nhất được chọn thường là trục Z, và đối với máy tiện là trục X. Trong cả hai trường hợp, trục phải dịch chuyển *ra xa* chi tiết gia công, đến vùng không gian trống. Khi trục đạt đến vị trí zero máy, đèn trên bảng điều khiển sẽ sáng lên, khẳng định cho từng trục. Tuy máy đã sẵn sàng để sử dụng, nhưng người vận hành còn phải thực hiện thêm một bước. Trên màn hình hiển thị *vị trí*, vị trí thực phải được xác lập theo giá trị zero cho từng trục, nếu hệ điều khiển không tự động xác lập vị trí zero. Nút POS trên bảng dữ liệu cho phép chọn màn hình hiển thị vị trí.

ĐIỂM QUY CHIẾU CHI TIẾT

Chi tiết sẵn sàng gia công được lắp trong phạm vi các giới hạn chuyển động của máy. Từng chi tiết phải được lắp trong thiết bị hoặc đồ gá an toàn, thích hợp với nguyên công được yêu cầu và không thay đổi vị trí đối với mọi chi tiết trong loạt gia công. Vị trí cố định của đồ gá là rất quan trọng để bảo đảm tính ổn định và tính chính xác, sao cho mọi chi tiết trong cùng một loạt gia công đều như nhau. Sau khi thiết lập các bước công nghệ, cần chọn điểm quy chiếu chi tiết (chuẩn gia công, chuẩn định vị).

Điểm quy chiếu này được dùng trong chương trình để thiết lập quan hệ với điểm quy chiếu máy. Điểm quy chiếu của dụng cụ cắt, và kích thước bản vẽ.

Điểm quy chiếu chi tiết thường được gọi là *zero chương trình* hoặc *zero chi tiết*. Do điểm tọa độ biểu thị zero chương trình có thể được nhà lập trình chọn bất kỳ, đây không phải là điểm cố định, và được gọi là điểm *linh hoạt*.

Chọn zero chương trình

Khi chọn zero chương trình, quyết định chính được thực hiện sẽ ảnh hưởng đến sự gá lắp chi tiết và hiệu suất gia công, do đó cần đặc biệt chú ý mọi yếu tố về lựa chọn zero (điểm chuẩn chương trình).

Về lý thuyết, điểm zero chương trình có thể được chọn tùy ý, nhưng điều này là không nên, dù đúng về toán học. Trong các giới hạn thực tiễn gia công, cần xét các khả năng để đưa đến kết quả tối ưu. Có ba yếu tố chính cần xem xét khi chọn zero chương trình.

- ❑ Độ chính xác gia công.
- ❑ Sự thuận tiện khi gá lắp và gia công.
- ❑ An toàn khi gia công.

Độ chính xác gia công

Độ chính xác gia công là yêu cầu cao nhất – mọi chi tiết đều phải được gia công chính xác theo các yêu cầu kỹ thuật trên bản vẽ, và bảo đảm tính lắp lẫn. Mọi chi tiết trong một loạt gia công và các loạt kế tiếp cũng phải như nhau (hầu như đồng nhất với nhau).

Thuận tiện khi gá lắp và gia công

Chỉ xét sự thuận tiện về gá lắp và gia công sau khi bảo đảm độ chính xác gia công. Mọi người đều mong muốn công việc dễ dàng hơn. Nhà lập trình CNC kinh nghiệm luôn luôn nghĩ đến hiệu quả của chương trình trong xưởng cơ khí.

An toàn lao động

An toàn luôn là yếu tố rất quan trọng trong sản xuất. Khi chọn zero chương trình cần xem xét kỹ các vấn đề an toàn trong gia công chi tiết.

Các yếu tố nêu trên là chung cho trung tâm gia công và máy tiện khi chọn zero chương trình, tuy nhiên các khác biệt về thiết kế chi tiết cũng có ảnh hưởng lớn đối với sự lựa chọn zero chương trình.

http://www.me-cad.com/

Zero chương trình – trung tâm gia công

Trung tâm gia công CNC cung cấp nhiều phương pháp xác lập và gá lắp. Tùy theo loại nguyên công, các phương pháp gá lắp phổ biến bao gồm sử dụng ngàm kẹp, mâm cặp, tấm lót và đồ gá. Ngoài ra, hệ thống phay CNC còn cho phép gá lắp nhiều chi tiết để tăng thêm các khả năng lựa chọn. Để chọn zero chương trình, cần xét cả ba trục máy. Trung tâm gia công với các trục bổ sung đòi hỏi điểm zero cho từng trục đó, ví dụ trục xoay, trục phân độ.

Nhà lập trình CNC xác định phương pháp gá lắp chi tiết cụ thể thường với sự hợp tác của người vận hành máy. Nhà lập trình CNC còn chọn vị trí zero cho từng chương trình. Quá trình lựa chọn zero chương trình bắt đầu bằng sự đánh giá bản vẽ, nhưng trước hết cần hoàn tất hai bước.

Bước 1.	Nghiên cứu các kích thước bản vẽ, kích thước nào là cơ bản
Bước 2.	Quyết định phương pháp định vị và kẹp chặt chi tiết.

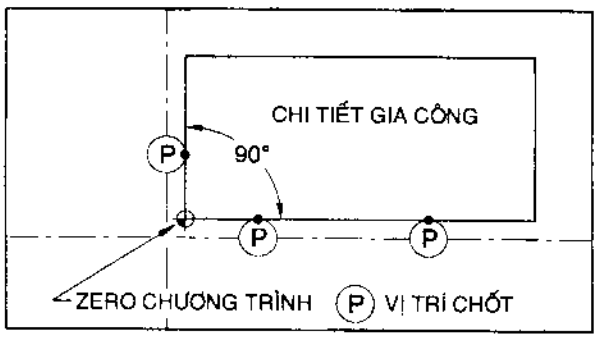
Zero chương trình thường được ghi trong bản vẽ. Trong quy trình gá lắp bất kỳ, bạn cần bảo đảm mọi dung sai và kích thước cơ bản đều được duy trì đồng bộ giữa các chi tiết.

Gá lắp đơn giản nhất trên bàn máy là giá

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

đỡ chi tiết, các bộ phận kẹp chặt và các bề mặt định vị. Các mặt định vị phải cố định trong khi gia công và dễ dàng đo đạc. Gá lắp kiểu này thường dựa trên kỹ thuật *ba chốt* định vị. Hai chốt tạo thành một hàng và chốt thứ ba theo phương pháp tuyến, tạo thành góc 90° cho hai mặt định vị (Hình 14.4).

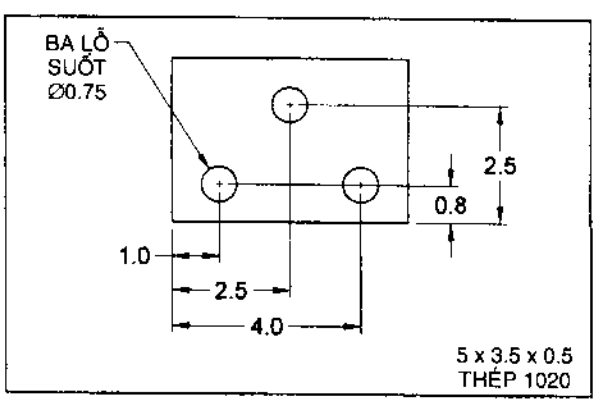


Hình 14.4. *Kỹ thuật ba chốt định vị chi tiết (cả ba chốt có cùng đường kính)*

Do chi tiết chỉ tiếp xúc một điểm với từng chốt định vị, sự gá lắp là rất chính xác. Sự kẹp chặt thường được thực hiện với các kẹp định vị và song song. Cạnh đáy và cạnh trái của chi tiết song song với các trục máy và vuông góc với nhau. Zero chương trình (chuẩn định vị chi tiết) là giao điểm của hai cạnh định vị.

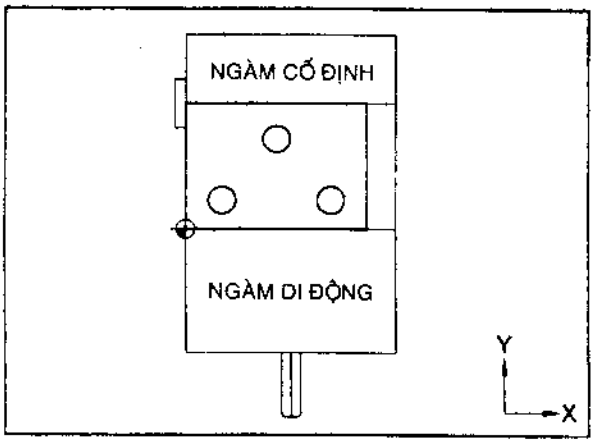
Kỹ thuật ba chốt là chung cho mọi gá lắp, không sử dụng các chốt thực. Nếu chi tiết được lắp trong ngàm kẹp, sự định vị cũng tương tự. Các ngàm kẹp phải song song hoặc vuông góc với trục máy và vị trí cố định phải được thiết lập với cữ chặn hoặc phương pháp định vị khác.

Do ngàm kẹp là thiết bị định vị thông dụng nhất cho các chi tiết nhỏ, ngàm kẹp được dùng làm ví dụ thực tiễn để minh họa phương pháp chọn zero chương trình. Hình 14.5 minh họa bản vẽ kỹ thuật đơn giản, có đủ các kích thước, ghi chú, và các đặc tính vật liệu.



Hình 14.5. *Bản vẽ minh họa phương pháp chọn zero chương trình.*

Đối với ví dụ này, sẽ không có vấn đề gì khi lập trình điểm quy chiếu ở vị trí bất kỳ ngoại trừ góc dưới bên trái của chi tiết. Đây là điểm gốc của bản vẽ và cũng là điểm gốc của chi tiết. Điểm này thỏa *Bước 1* trong quy trình chọn zero chương trình. *Bước 2*, thực chất là chọn thiết bị kẹp chặt chi tiết. Sự gá lắp chi tiết trong ngàm kẹp trên máy CNC được minh họa trên Hình 14.6.



Hình 14.6. *Lắp chi tiết trong ngàm kẹp máy. Phiên bản 1*

Trong gá lắp, tạm gọi là *Phiên bản 1*, chi tiết được định vị giữa ngàm kẹp và cữ chặn bên trái. Định hướng của chi tiết hoàn toàn như trên bản vẽ, do đó mọi kích thước bản vẽ sẽ xuất hiện trong chương trình đều sử dụng đúng các kích thước trên bản vẽ. Điều này có vẻ là gá lắp đúng, nhưng thiếu tính thực tiễn.

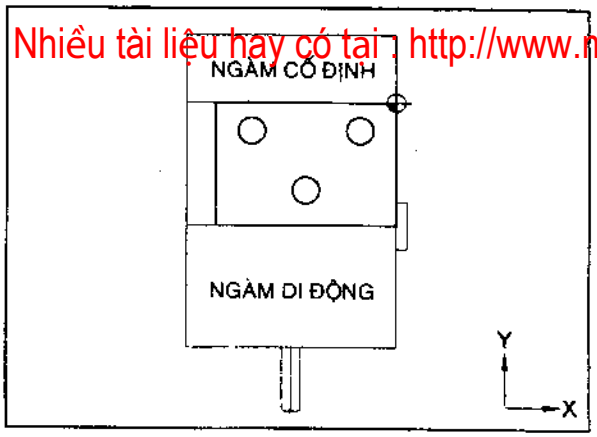
Điều chưa được xem xét là kích thước ban đầu của phôi thép. Bản vẽ quy định phôi là khối thép chữ nhật 5.00 × 3.50. Đây chỉ là các kích thước chung, chúng có thể có dung sai ± 0.1 hoặc cao hơn.

Bạn hãy kết hợp dung sai cho phép với thiết kế ngàm kẹp, trong đó, *một ngàm cố định* và *một ngàm di động*. Vấn đề sẽ dễ nhận thấy hơn. *Quy chiếu trục Y là ngược với ngàm di động.*

Cạnh của zero chương trình phải là ngàm *cố định*. Nhiều nhà lập trình sử dụng ngàm di động làm cạnh quy chiếu, điều này là không đúng. Lợi ích của lập trình trong góc phần tư thứ nhất (mọi giá trị tuyệt đối đều là dương) là rất hấp dẫn, nhưng có thể đưa đến kết quả gia công không chính xác, trừ khi phôi là đồng nhất 100% đối với mọi chi tiết (ít xảy ra trong thực tế). Gá lắp theo *Phiên bản 1* có thể được cải tiến rõ rệt bằng cách quay chi tiết 180° và đặt thẳng hành cữ chặn ở phía đối diện. (Hình 14.7).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

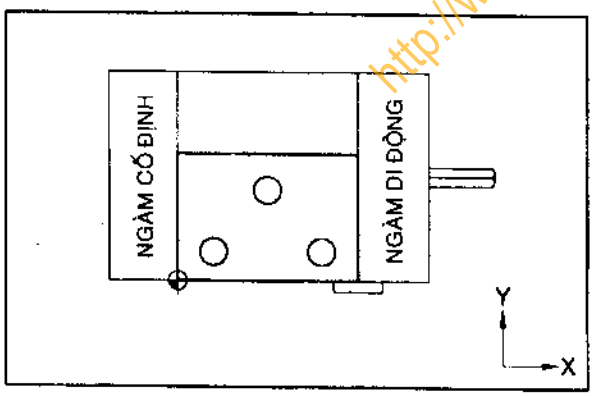
Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 14.7. Lắp chi tiết trong ngàm kẹp máy – Phiên bản 2

Trong Phiên bản 2, kết quả đồng nhất với bản vẽ. Sự định hướng chi tiết 180^0 gây ra vấn đề khác – *chi tiết được định vị trong góc phần tư thứ ba*. Mọi giá trị X và Y đều là số *âm*. Các kích thước bản vẽ có thể được sử dụng trong chương trình nhưng theo số âm. Bạn không được quên dấu âm. Nếu phải chọn giữa Phiên bản 1 và Phiên bản 2, bạn hãy chọn *Phiên bản 2* và bảo đảm mọi dấu âm đều được lập trình chính xác.

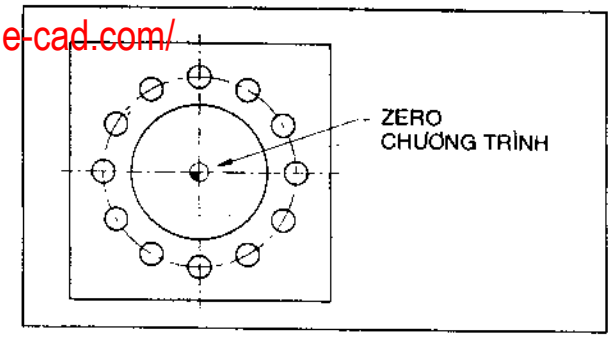
Phiên bản 3 có lẽ là tối ưu, chương trình gia công sẽ có mọi kích thước trong góc phần tư thứ nhất và đúng theo bản vẽ. Ngoài ra cạnh quy chiếu của chi tiết sẽ ngược với ngàm kẹp cố định. Giải pháp là bạn quay *bộ ngàm kẹp* 90^0 và định vị chi tiết như trên Hình 14.8, nếu có thể.



Hình 14.8. Chi tiết lắp trong bộ ngàm kẹp máy – Phiên bản 3

Để chọn zero máy đối với trục Z, bạn nên chọn mặt trên (đỉnh) của chi tiết hoàn tất. Điều này sẽ làm cho trục Z là dương *phía trên* mặt này và là âm *phía dưới* mặt đó. Phương pháp kế tiếp là chọn mặt đáy (dưới) của chi tiết, nơi chi tiết được định vị trong đồ gá.

Các đồ gá đặc biệt có thể được dùng để gá lắp chi tiết. Để định vị chi tiết phức tạp, cần có đồ gá chuyên dùng. Trong nhiều ứng dụng của đồ gá, vị trí zero chương trình có thể được chọn ngay trong đồ gá, cách xa chi tiết gia công.



Hình 14.9. Zero chương trình đối với các đối tượng tròn thường là tâm điểm.

Sự chọn zero chương trình đối với các chi tiết tròn hoặc sơ đồ hình tròn (ví dụ, các lỗ trên chu vi hình tròn), zero chương trình hữu dụng nhất là tâm đường tròn (Hình 14.9).

Chương 39 sẽ trình bày lệnh G52, có thể giải quyết nhiều vấn đề liên quan với zero chương trình ở tâm hình tròn.

Zero chương trình – máy tiện

Trên máy tiện CNC, sự chọn zero chương trình khá đơn giản, do chỉ cần xét hai trục là trục đứng X và trục ngang Y. Do thiết kế của máy tiện, zero chương trình theo trục X luôn luôn là đường tâm trục chính.

Trên máy tiện CNC, zero chương trình đối với trục X PHẢI trên đường tâm trục chính

Đối với trục Z, có thể chọn một trong ba phương pháp:

- ❑ Mặt mâm cặp ... mặt chính của mâm cặp
- ❑ Mặt ngàm kẹp ... mặt định vị của các ngàm kẹp
- ❑ Mặt chi tiết ... mặt đầu của chi tiết hoàn tất

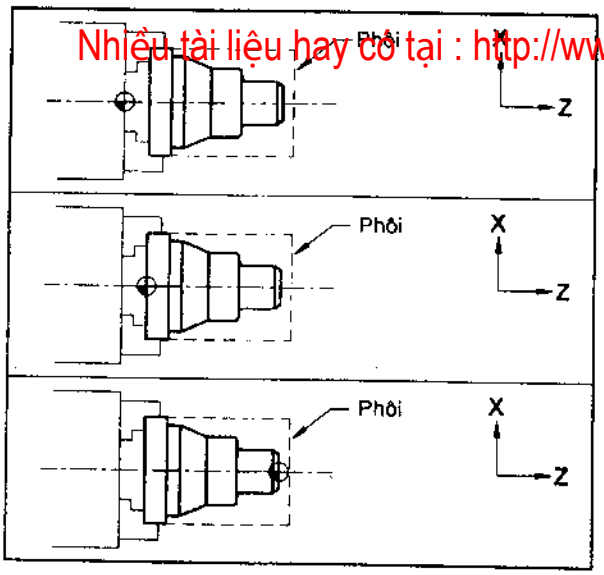
Trong gá lắp, mặt mâm cặp chỉ có một ưu điểm, mặt này có thể chạm vào cạnh dao cắt, sử dụng các cữ lá để tránh phoi tiện vướng vào dao cắt. Nhược điểm là, trừ khi chi tiết tựa vào mặt mâm cặp, cần có thêm các tính toán về tọa độ và khó sử dụng các kích thước trên bản vẽ.

Mặt ngàm kẹp hoặc đồ gá là giải pháp thuận tiện hơn. Mặt này cũng có thể chạm vào dao cắt và cố định đối với mọi chi tiết. Vị trí này có thể có ích khi gia công các hình dạng không đều, chẳng hạn vật đúc, vật rèn...

Nhiều chi tiết tiện cần gia công ở cả hai mặt đầu. Trong khi gia công mặt thứ nhất, phôi để gia công mặt thứ hai phải được cộng thêm từng giá trị Z. Đây là lý do chính để nhà lập trình không chọn zero chương trình trên mặt ngàm kẹp hoặc đồ gá, trừ các trường hợp đặc biệt.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 14.10. Lựa chọn zero chương trình trên máy tiện CNC- đường tâm là X0

Phương pháp thông dụng nhất là chọn zero chương trình trên *mặt đầu* của chi tiết *hoàn tất*. Đây không phải là sự lựa chọn hoàn hảo, nhưng có nhiều ưu điểm. Nhược điểm duy nhất là trong khi gá lắp chưa có bề mặt hoàn tất. Nhiều người vận hành máy cộng thêm chiều rộng của bề mặt thô (chưa gia công) hoặc cắt một mặt nhỏ để dao tiếp xúc.

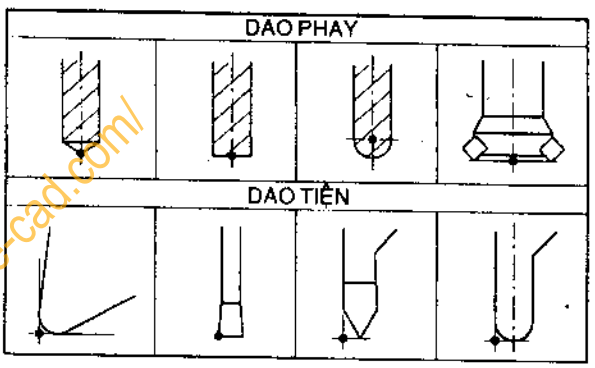
Các ưu điểm của zero chương trình ở mặt đầu? Thứ nhất là nhiều kích thước bản vẽ dọc theo trục Z có thể chuyển trực tiếp vào chương trình, thường với giá trị *âm*. Tuy điều này phụ thuộc vào phương pháp lập kích thước nhưng trong nhiều trường hợp, nhà lập trình CNC sẽ có lợi. Thứ hai, có lẽ quan trọng nhất, là giá trị Z âm của chuyển động dao biểu thị khu vực làm việc, giá trị dương là vùng không gian trống. Trong khi viết chương trình, có thể dễ dàng quên dấu âm của các chuyển động cắt theo trục Z. Lỗi sai này, nếu không phát hiện kịp thời, sẽ đưa dao vào chi tiết, ụ động có thể sẽ là vật cản. Đây là vị trí sai nhưng còn đỡ hơn là va đập vào chi tiết. Các ví dụ ở đây sử dụng zero chương trình ở *mặt đầu chi tiết đã gia công*, trừ khi có các chuyên biệt khác.

ĐIỂM QUY CHIẾU DỤNG CỤ CẮT

Điểm quy chiếu cuối cùng liên quan đến dụng cụ cắt. Trong các nguyên công phay, điểm quy chiếu dụng cụ cắt thường là giao điểm của đường tâm dao cắt và lưỡi cắt ở vị trí thấp nhất.

Trong tiện và doa, điểm quy chiếu dụng cụ cắt phổ biến nhất là đỉnh cắt tưởng tượng của mảnh dao cắt, do hầu hết các dao cắt đều có lưỡi cắt với bán kính xác định.

Đối với dao cắt kiểu mũi khoan hoặc dao nhiều lưỡi cắt được dùng để phay hoặc tiện, điểm quy chiếu luôn luôn là đỉnh dụng cụ cắt, đo theo trục Z (Hình 14.11).



Hình 14.11. Các điểm quy chiếu dụng cụ cắt

Cả ba nhóm điểm quy chiếu đều liên quan với nhau. Sai số trong một số xác lập sẽ có tác hại đối với điểm quy chiếu khác. Kiến thức về điểm quy chiếu là rất quan trọng để hiểu các lệnh đăng ký, dạng hình học máy, và các giá trị bù.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

15 CÁC LỆNH ĐĂNG KÝ

Ba điểm quy chiếu khả dụng trong lập trình CNC phải đồng bộ và hài hòa với nhau để bảo đảm độ chính xác gia công. Khi có các điểm quy chiếu đối với *chi tiết* (zero chương trình) và đối với *dụng cụ cắt* cần có thêm phương tiện để liên hệ chúng với nhau. Đây phải là phương tiện *"báo cho"* hệ điều khiển biết vị trí chính xác của từng dao cắt, trong khu vực gia công của máy, *trước khi* có thể sử dụng dao cắt. Phương pháp cổ điển để thực hiện điều này là *đăng ký* (ghi) vị trí hiện hành của dao cắt vào bộ nhớ của hệ điều khiển, thông qua chương trình. Phương pháp này đòi hỏi sử dụng lệnh có tên là *Position Register* (đăng ký vị trí).

LỆNH POSITION REGISTER

Lệnh chuẩn bị để đăng ký vị trí dao cắt là G92 dùng cho trung tâm gia công và G50 dùng cho máy tiện:

G92	Lệnh đăng ký vị trí (dùng cho nguyên công phay)
G50	Lệnh đăng ký vị trí (dùng cho nguyên công tiện)

Một số máy tiện CNC cũng sử dụng lệnh G92, nhưng các máy tiện có bộ điều khiển Fanuc hoặc bộ điều khiển tương tự thường sử dụng lệnh G50. Trong các ứng dụng thực tiễn, hai lệnh G92 và G50 có cùng ý nghĩa, do đó phần trình bày dưới đây sẽ áp dụng chung cho cả hai. Phần đầu của chương này sẽ tập trung vào các ứng dụng phay sử dụng lệnh G92, phần sau sẽ nói về lệnh G50 trên máy tiện.

Trong lập trình CNC hiện đại, hai lệnh đăng ký được thay bằng tính năng linh hoạt và tinh vi hơn được gọi là *Work Offset* (G54 đến G59) (Chương 17), và *Tool Length Offset* (G43) (Chương 18). Tuy nhiên, có thể vẫn còn vài máy công cụ cũ chưa có chuỗi các lệnh G54. Ngoài ra, nhiều công ty sử dụng các chương trình cũ chạy trên các máy CNC hiện đại. Trong các trường hợp đó, bạn cần hiểu rõ lệnh đăng ký vị trí. Lệnh này hơi khó hiểu đối với một số nhà lập trình và người vận hành, nhưng trong thực tế là lệnh rất đơn giản.

Trước hết, bạn hãy xem định nghĩa chi tiết của lệnh này. Định nghĩa chung chỉ chuyên biệt *Position Register Command* (lệnh đăng ký vị trí), nhưng không nêu rõ ý nghĩa.

Định nghĩa đăng ký vị trí

Định nghĩa chi tiết hơn về lệnh đăng ký vị trí có thể như sau:

Lệnh đăng ký vị trí xác lập vị trí dao theo khoảng cách và chiều TỪ ... zero chương trình ĐẾN ... vị trí dao hiện hành, đo dọc theo các trục.

Chú ý, định nghĩa này không đề cập *zero máy*, chỉ nêu ra *vị trí dao cắt hiện hành*. Đây là sự phân biệt rất quan trọng. Vị trí dao cắt hiện hành *có thể* ở zero máy, nhưng cũng *có thể* ở vị trí khác, trong các giới hạn hành trình trên từng trục máy.

Ngoài ra, bạn cần chú ý sự nhấn mạnh chiều *từ – đến*. Theo định nghĩa, khoảng cách này là một chiều, giữa zero chương trình và vị trí dao hiện hành. Chiều này luôn luôn *từ* zero chương trình *đến* vị trí dao cắt, không bao giờ ngược lại. Trong chương trình, cần bảo đảm dấu đúng (dương, âm, zero) của từng giá trị trục.

Sự đăng ký vị trí chỉ có thể áp dụng trong chế độ lập trình tuyệt đối, trong khi lệnh G90 có hiệu lực, không sử dụng trong chế độ lập trình số gia (tương đối) G91. Trong lập trình thực tiễn, hầu như mọi chương trình được viết theo chế độ số gia đều bắt đầu với chế độ tuyệt đối để đạt đến vị trí dao cắt thứ nhất.

Định dạng lập trình

Dữ liệu vị trí dao cắt liên quan với lệnh G92 sẽ được *đăng ký* (lưu) trong bộ nhớ của hệ thống điều khiển.

Định dạng đối với lệnh G92 như sau:

```
G92 X.. Y.. Z..
```

Trong mọi trường hợp, địa chỉ của từng trục chuyên biệt khoảng cách *từ zero chương trình đến điểm quy chiếu dao* (mũi dao). Nhà lập trình cung cấp tất cả các tọa độ dựa trên chuẩn chương trình (zero chương trình). Trục bổ sung bất kỳ cũng được đăng ký với G92, ví dụ trục B để phân độ bàn máy trên trung tâm gia công ngang.

Xác lập vị trí dao cắt

Công dụng duy nhất của lệnh G92 là *đăng ký vị trí dao cắt hiện hành vào bộ nhớ của hệ thống điều khiển.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trong block chứa lệnh G92 không được phép có chuyển động máy

Có thể thấy tác dụng của G92 trên màn hình hiển thị *vị trí tuyệt đối*. Ở mọi thời điểm, sự hiển thị vị trí tuyệt đối luôn luôn có vài giá trị cho từng trục, chúng có thể là zero hoặc số bất kỳ. Khi lệnh G92 được thực thi, mọi giá trị hiện hành trên màn hiển thị sẽ *được thay* bằng các giá trị chuyên biệt theo G92. Nếu có trục không được chuyên biệt với G92, sẽ không có sự thay đổi hiển thị đối với trục đó. Tại máy, người vận hành có trách nhiệm tương hợp xác lập dao cắt thực với các giá trị chuyên biệt trong lệnh G92.

ỨNG DỤNG CÁC TRUNG TÂM GIA CÔNG

Trong lập trình các trung tâm gia công CNC *không* có tính năng *Work Coordinate System* – hệ tọa độ gia công (còn gọi là *Work Offsets* – bù chi tiết gia công), *Position Register* phải được thiết lập cho từng trục và từng dao. Có hai phương pháp.

- Vị trí dao cắt được xác lập tại *zero máy*.
- Vị trí dao cắt được xác lập *cách xa zero máy*.

Xác lập dao ở zero máy

Phương pháp thứ nhất đòi hỏi vị trí zero máy cũng là vị trí thay dao đối với tất cả các trục. Điều này không cần thiết và *không* thực tiễn. Bạn hãy xét điều này và tìm hiểu tại sao điều đó thiếu thực tiễn.

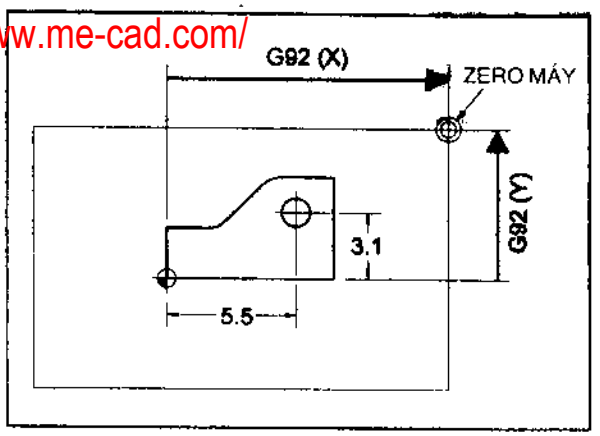
Chương trình thường được viết ở cách xa máy, nhưng phải chuyên biệt vị trí chi tiết trên bàn máy:

```
G92 X12.0 Y7.5 Z8.375
```

Các số trong ví dụ này dường như vô hại. Nhưng bạn hãy xét người vận hành CNC ở bên máy, đang gá lắp chi tiết (không có đồ gá chuyên dùng), *chính xác* 12.0 inch cách zero máy theo trục X. Đồng thời người vận hành phải lắp chi tiết này *chính xác* 7.5 inch cách zero máy theo trục Y và cách 8.375 inch theo trục Z.

Đây hầu như là nhiệm vụ không thể thực hiện, đặc biệt khi không có đồ gá chuyên dùng, và rất mất thời gian. Ở đây *không cần* các số này, chúng hoàn toàn là tùy ý; X12.0 có thể là X12.5 mà không hề có ảnh hưởng gì. Toàn bộ khó khăn này chỉ do nhà lập trình đã chọn điểm quy chiếu zero máy làm vị trí thay dao (chủ yếu theo các trục X và Y)

Hình 15.1 minh họa xác lập G92 dựa trên xác lập dao ở vị trí zero máy. Phương pháp bắt đầu chương trình ở zero máy là rất hữu dụng.

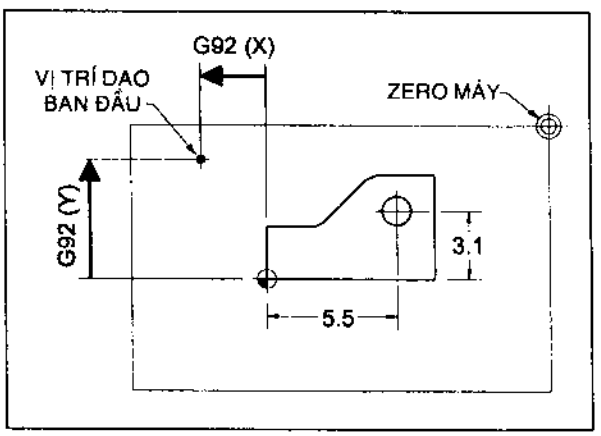


Hình 15.1. Đăng ký vị trí dao hiện hành ở zero máy (chỉ nêu các trục X và Y)

Đây có thể là ưu điểm, ví dụ, nếu đồ gá chuyên dùng được lắp cố định trên bàn máy. Tấm lót với lưới định vị thường được sử dụng, cũng có thể dùng một hoặc nhiều bộ ngàm kẹp cố định. Phương pháp gá lắp này có nhiều biến thể.

Xác lập dao cách xa zero máy

Phương pháp này có thể loại bỏ khó khăn của phương pháp nêu trên, cho phép nhà lập trình chọn vị trí dao XY ở điểm bất kỳ trong phạm vi các giới hạn hành trình máy (trước hết cần xét các yêu cầu an toàn) và sử dụng vị trí đó làm vị trí thay dao theo các trục X, Y. Do không cần zero máy, người vận hành có thể gá lắp chi tiết ở vị trí bất kỳ trên bàn máy, trong các giới hạn hành trình của từng trục máy. Hình 15.2 minh họa ví dụ về xác lập dao theo chiều âm trên trục X và chiều dương trên trục Y.



Hình 15.2. Xác lập đăng ký vị trí dao hiện hành từ zero máy (chỉ nêu các trục X và Y).

Để đưa dao vào vị trí bắt đầu thay dao, người vận hành phải dịch chuyển dao *từ zero chương trình* theo các giá trị chuyên biệt trong lệnh G92. Điều này dễ thực hiện hơn và hiệu quả hơn so với xác lập theo zero máy.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

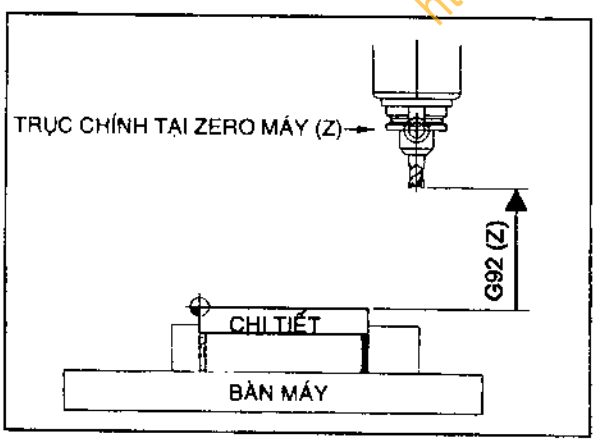
Sau khi thiết lập vị trí của tập dao máy, mọi dao trong chương trình đều trở về vị trí này để thay dao. Vị trí thay dao tự động trên trục Z của trung tâm gia công đứng phải được lập trình ở zero máy, đây là vị trí thay dao tự động *duy nhất*. Do đó nội dung nêu trên thực tế chỉ áp dụng cho các trục X, Y. Bất kể vị trí dao, xác lập G92 phải như nhau cho mọi dụng cụ cắt, trừ khi có lý do chính đáng để thay đổi.

Nhược điểm duy nhất của phương pháp này là vị trí dao mới chỉ được hệ thống điều khiển ghi nhớ trong khi được cung cấp điện. Khi tắt máy (hoặc mất điện) vị trí thay dao này sẽ bị mất. Nhiều người vận hành CNC có kinh nghiệm giải quyết vấn đề này đơn giản bằng cách tìm khoảng cách thực từ zero máy đến vị trí thay dao, ghi lại cho từng lần gá lắp và sau đó dịch chuyển dao theo khoảng cách đó sau khi có điện trở lại, chẳng hạn khi bắt đầu ca làm việc mới.

Đăng ký vị trí theo trục Z

Đối với phay đứng, để thực hiện sự thay dao tự động, trục Z phải trở về zero máy. Giá trị đăng ký vị trí này được đo từ zero máy của trục Z (thường là mặt trên hoàn tất), đến đỉnh quy chiếu dao, trong khi trục Z ở vị trí zero máy.

Nói chung, từng dao đều có giá trị Z *khác* trong lệnh G92, với giả thiết các dao có chiều dài khác nhau. Về nguyên tắc, các xác lập XY sẽ không đổi. Hình 15.3 minh họa xác lập đối với lệnh G92 dọc theo trục Z. Ví dụ O1601 minh họa cho khái niệm này.



Hình 15.3. *Xác lập đăng ký vị trí dao hiện hành tại zero máy đối với trục Z (từng dao có xác lập riêng)*

Ví dụ lập trình

Để minh họa phương pháp sử dụng lệnh đăng ký trong chương trình trên trung tâm gia công đứng, cần tuân thủ một số nguyên tắc.

- Dụng cụ cắt số 1 là dao được thay đầu tiên
- G92 phải được thiết lập trước mọi chuyển động dao bất kỳ
- Dao phải trở về vị trí G92 sau khi hoàn tất sự cắt gọt.

Cả ba nguyên tắc này đều được tuân thủ trong chương trình mẫu:

```
O1601                                (PROGRAM NUMBER)
N1 G20                             (SET ENGLISH UNITS)
N2 G17 G40 G80 G90 T01              (GET TOOL 1 READY)
N3 M06                             (TOOL 1 TO SPINDLE)
N4 G92 X9.75 Y6.5 Z11.0              (SET CURRENT XY)
N5 G00 X1.0 Y0.5 S800 M03          (MOVE TO POSITION)
N6 Z0.1 M08                      (MOVE TO CLEAR ABOVE)
N7 G01 Z-0.55 F5.0                    (FEED TO DEPTH)
N8 X3.0 Y4.0 F7.0                        (CUT A SLOT)
N9 G00 Z11.0 M09          (RAPID TO Z MACHINE ZERO)
N10 X9.75 Y6.5 M05 (RAPID TO XY SET POSITION)
N11 M01                  (OPTIONAL STOP FOR TOOL 1)
...
```

Đây là ví dụ đơn giản khi viết nhưng khó xác lập trên máy. Vào thời điểm này bạn chưa cần biết các mục nhập chương trình, phần sau sẽ giải thích rõ.

Bạn chú ý – vị trí xác lập trục Z luôn luôn được biết – *tại zero máy*. Bất kể sự thay dao được thực hiện theo các trục XY, tại zero máy hay cách xa điểm đó, định dạng chương trình vẫn như nhau, chỉ *ý nghĩa* của các giá trị là khác nhau. Chỉ một dao được sử dụng, nhưng nói chung từng dao đều có giá trị Z riêng để đăng ký vị trí, do các dao có chiều dài khác nhau.

ỨNG DỤNG TRÊN MÁY TIỆN

Đối với máy tiện CNC có bộ điều khiển Fanuc hoặc tương tự, lệnh G50 được dùng thay cho G92.

```
G50 X.. Z..
```

Nếu G92 được dùng cho máy tiện, lệnh này sẽ như sau:

```
G92 X.. Z..
```

Lệnh này có cùng định nghĩa và nguyên tắc như đối với nguyên công phay – biểu thị khoảng cách *từ* zero chương trình *đến* vị trí dao hiện hành dọc theo các trục.

Không được có chuyển động máy trong block chứa lệnh G50 hoặc G92

Các lệnh G50 và G92 là đồng nhất, nhưng thuộc hai nhóm mã G khác nhau. Fanuc cung cấp ba nhóm mã G để điều khiển máy tiện. Về lịch sử, các bộ điều khiển của Nhật thường dùng lệnh G50, còn Hoa Kỳ sử dụng G92. Sản phẩm hợp tác Hoa Kỳ, Nhật là GE Fanuc (General Electric và Fanuc), được dùng phổ biến trong công nghiệp Bắc Mỹ, sử dụng lệnh G50.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lập trình đăng ký vị trí cho các ứng dụng phay tiện hầu như hoàn toàn tương tự với lệnh G92 trên máy phay. Tuy nhiên, do thiết kế của máy tiện CNC, mọi dao cắt đều được lắp trên hộc dao (bàn xe dao), cần phải xét *phần chìa ra* của từng dao (trên cả hai trục) từ giá dao. Ngoài ra, cần tránh sự va chạm, do mọi dao được lắp nhưng chưa đến lượt cắt gọt sẽ chuyển động đồng thời với dao đang hoạt động. Trong nguyên công phay, các dao chưa hoạt động đều được lắp trong hộp chứa dao do đó rất an toàn. Hiện có một số thiết kế mới cho máy tiện CNC với bộ thay dao tương tự như trên máy phay.

Gá lắp dao

Quyết định lập trình quan trọng nhất khi tiện là sự gá lắp. Ở đây có nhiều lựa chọn với các ưu điểm và nhược điểm riêng.

Có lẽ phương pháp thực tiễn nhất đối với gá lắp tiện là sử dụng vị trí thay dao cho mọi dao cắt tương ứng vị trí zero máy. Bàn xe dao dễ dàng dịch chuyển đến vị trí này, chỉ cần sử dụng các công tắc trên bảng điều khiển. Vị trí đăng ký được đo đến zero máy có một nhược điểm lớn, có thể *quá xa* đối với nhiều chi tiết, đặc biệt là trên các máy tiện lớn dọc theo trục Z. Bạn hãy hình dung chuyển động dao dọc theo trục Z khoảng 30 in chỉ để phân độ bàn xe dao và 30 in chuyển động trở lại để tiếp tục chu kỳ cắt gọt. Điều này hoàn toàn không hiệu quả, tuy nhiên luôn luôn có giải pháp thích hợp.

Phương pháp hiệu quả hơn nhiều là chọn vị trí phân độ dao cắt *gần chi tiết* đến mức ngắn nhất cho phép. Vị trí này phải dựa trên dụng cụ cắt *dài nhất* lắp trên bàn xe dao (thường là dao tiện mặt trong), dù dao đó có được dùng trong chương trình hay không. Nếu có đủ khoảng trống cho dao dài nhất, thì cũng sẽ đủ cho mọi dao còn lại.

Sự dung hòa giữa hai phương pháp nêu trên là chỉ duy trì vị trí phân độ dao tại zero máy trên trục X (thường không quá xa) và chỉ cần thiết lập vị trí trục Z.

Trên máy tiện CNC, bạn cần nhớ sự sắp xếp của tất cả các dao trên bàn xe dao, để tránh va chạm với mâm cặp, chi tiết, hoặc với máy.

Ngoài ra còn có các phương pháp khác có thể áp dụng để lắp dao trên máy tiện sử dụng lệnh G50, nhưng ít được sử dụng.

Các nhóm xác lập ba dao cắt

Trên máy tiện CNC, được trang bị với hộc dao đa giác (6 đến 14 ổ dao), mọi dao cắt đều được lắp vào ổ dao riêng trong hộc dao. Khi [illegible], chỉ có dao được chọn là trong ổ dao hoạt động. Khi đánh giá kiểu dụng cụ cắt trên máy tiện, có thể thấy chỉ có ba nhóm dao cắt, dựa trên kiểu nguyên công tiện cơ bản.

- Dao cắt làm việc trên *đường tâm* chi tiết.
- Dao cắt làm việc theo *mặt ngoài* chi tiết.
- Dao cắt làm việc theo *mặt trong* chi tiết.

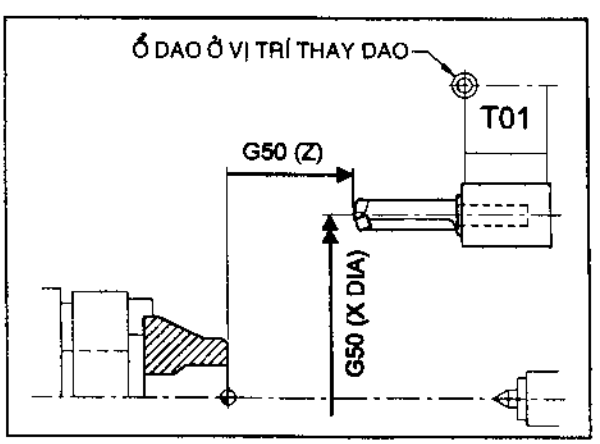
Nếu hiểu rõ sự đăng ký vị trí cho từng nhóm, sẽ dễ dàng áp dụng trong dao cắt bất kỳ trong nhóm, bất kể số lượng dao được sử dụng.

Xác lập dao cắt đường tâm

Các dao cắt theo đường tâm trên máy tiện bao gồm mũi khoan tâm, khoan điểm, mũi khoan xoắn tiêu chuẩn, mũi khoan sâu (gắn các mảnh hợp kim), tarô ren, chuốt, ... Mọi dụng cụ cắt thuộc nhóm này đều có một đặc điểm chung, đỉnh dao luôn luôn định vị theo đường tâm trục chính trong khi cắt gọt. Các dụng cụ cắt này phải được lắp chính xác 90^0 với mặt đầu của chi tiết (song song với trục Z)

Giá trị đăng ký vị trí theo trục X là từ đường tâm trục chính (X0) đến đường tâm dụng cụ cắt. Đối với trục Z, giá trị đăng ký được đo từ zero chương trình đến đỉnh dụng cụ cắt. Nói chung, các dao cắt theo đường tâm có phần chìa ra khá dài – nghĩa là giá trị G50 của chúng theo trục Z tương đối nhỏ khi so với dao cắt ngoài.

Hình 15.4 minh họa sự gá lắp dao cắt theo đường tâm sử dụng mũi khoan với mảnh chắp.



Hình 15.4. Xác lập G50 cho dao cắt theo đường

Xác lập dao tiện ngoài

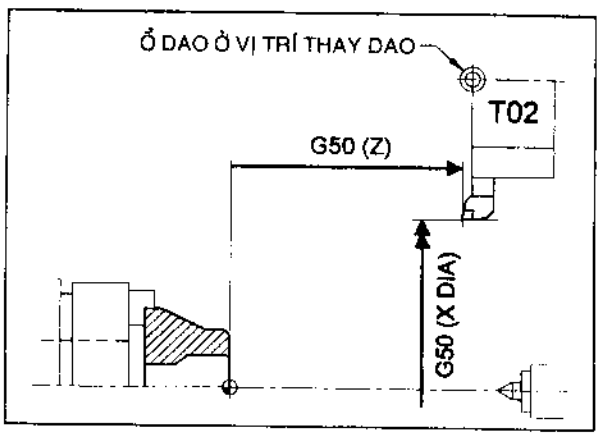
Đối với các nguyên công tiện ngoài, chẳng hạn tiện thô và tiện tinh đường kính ngoài, chà nhám, tiện côn, tiện ren, tiện rãnh, cắt đứt... dao tương đối nhỏ và tiến đến chi tiết với không gian khá rộng.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Giá trị đăng ký vị trí được đo từ zero chương trình đến đầu mũi dao (hoặc mảnh hợp kim). Nếu dùng dao tiện ren hoặc cắt rãnh, giá trị G50 thường được đo từ bên trái mảnh hợp kim, chủ yếu để bảo đảm an toàn.

Hình 15.5 minh họa gá lắp tiện mặt ngoài.

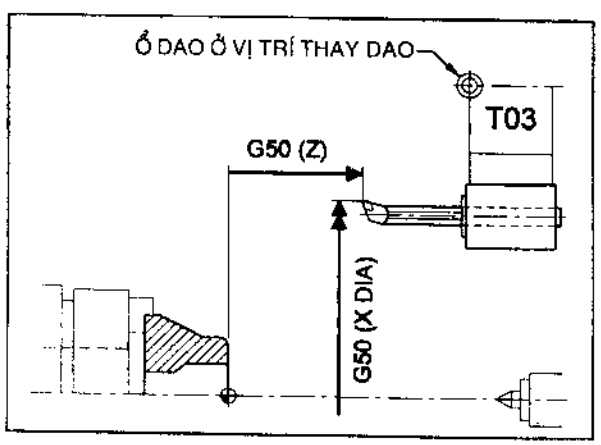


Hình 15.5. Xác lập G50 cho dao tiện mặt ngoài

Xác lập dao tiện trong

Dao tiện trong thực hiện gia công mặt trong của chi tiết, chẳng hạn trong lỗ đã khoan, doa lỗ, tiện rãnh trong, tiện ren trong... Các nguyên tắc gá lắp dọc theo trục Z đối với dao tiện trong hoàn toàn như dao tiện ngoài cùng chủng loại.

Dọc theo trục X, xác lập đăng ký vị trí dao phải được thực hiện đến đỉnh tưởng tượng của lưỡi cắt (mảnh dao hợp kim). Hình 15.6 minh họa xác lập đăng ký vị trí dao gia công lỗ (dao doa).



Hình 15.6. Xác lập G50 cho dao tiện trong

Hình 15.4, 15.5 và 15.6 minh họa thứ tự ba nguyên công (khoan – tiện - doa) thực hiện trên máy tiện CNC. Chú ý, vị trí hộc dao được xác định là *vị trí thay dao*, không bắt buộc phải là vị trí zero máy. Điều đó có nghĩa là G50 có thể được xác lập ở nơi bất kỳ trong các giới hạn hành trình của máy, kể cả ở zero máy.

Vì các lý do an toàn, dao không được chìa ra từ ổ dao vào vùng Z có giá trị âm – bên trái của mặt đầu chi tiết. Nhiều máy tiện CNC có hành trình hơi dài *vượt quá* zero máy trên trục Z (1-2 inch, 25 – 50 mm). Đôi khi có thể vào vùng này để thay dao an toàn cho các dao rất dài. Tuy nhiên, đây là phương pháp lập trình tiên tiến và đòi hỏi các đánh giá an toàn nghiêm ngặt. Hầu như không có vòng mở rộng đối với trục Z phía trên vị trí máy (chỉ khoảng 0.2 inch, 0.5 mm)

Vấn đề an toàn kế tiếp liên quan đến các dao dài là khoảng hở trong vùng kẹp chặt chi tiết, kể cả mâm cặp và ngàm kẹp. Bạn cần bảo đảm chỉ kéo dài các dao khi công việc thực sự đòi hỏi điều đó.

Chi tiết đỉnh góc

Dao tiện CNC thường có mảnh hợp kim có thể phân độ, với bán kính góc để tăng độ bền và độ bóng bề mặt gia công. Khi lệnh đăng ký vị trí được dùng cho dao có bán kính mũi, nhà lập trình cần biết (và báo cho người vận hành CNC), G50 ứng với *lưỡi cắt nào*. Trong nhiều trường hợp, sự lựa chọn là đơn giản. Giá trị G50 được đo từ zero máy đến giao điểm của trục tiếp tuyến X và Z. Tùy theo hình dạng dao và định hướng dao trong ổ dao, xác lập G50 sẽ có giá trị tương ứng. Hình 15.7 minh họa vài xác lập đối với các định hướng dao và có bán kính góc, gồm hai dao cắt rãnh.

Ví dụ lập trình

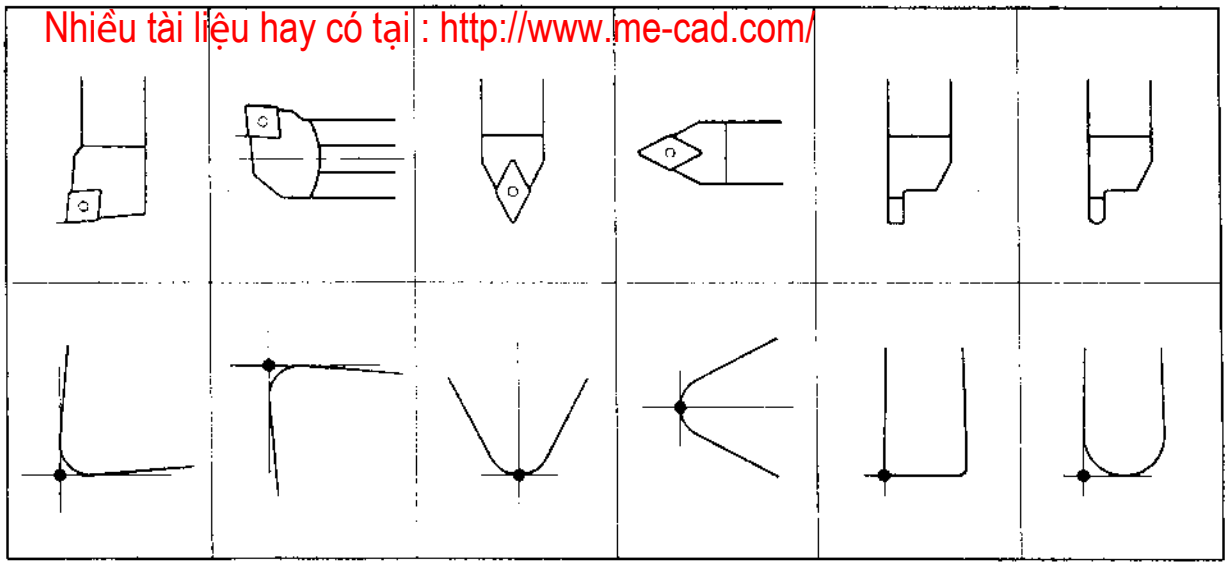
Ví dụ này minh họa phương pháp sử dụng lệnh G50 trên máy tiện có nhiều nét tương tự chương trình trên trung tâm gia công. Thứ nhất, sự thay dao được thực hiện, tiếp sau xác lập G40 cho dao được chọn. Khi dao đó tiến hành gia công, sẽ trở về đúng vị trí tuyệt đối được chuyên biệt trong block chứa G50. Trong ví dụ này có hai dao, dao thứ nhất được lập trình để tiện bề mặt (vạt mặt), dao thứ hai được lập trình để tiện đường kính 2.5 inch.

```
O1602
N1 T0100
N2 G50 X7.45 Z5.5
N3 G96 S400 M03
N4 G00 X2.7 Z0 T0101 M08
N5 G01 X-0.07 F0.007
N6 G00 Z0.1 M09
N7 X7.45 Z5.5 T0100
N8 M01

N9 T0200
N10 G50 X8.3 Z4.8
N11 G96 S425 M03
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 15.7. *Xác lập đăng ký vị trí G50 đối với các định hướng đầu dao thông dụng – chấm đen đậm biểu thị các tọa độ XY do G50 X..Y xác lập.*

```
N12 g00 X2.5 Z0.1 T0202 M08
N13 Z-1.75 F0.008
N14 G00 X2.7 M09
N15 X8.3 Z4.8 T0200
N16 M30
%
```

Chú ý các block N2 và N7 dùng cho dao thứ nhất, N10 và N15 cho dao thứ hai. Đối với từng dao, các cặp XZ này của các block là như nhau. Điều chương trình "báo cho" hệ điều khiển biết ở đây là block N2 chỉ đăng ký vị trí dao hiện hành, còn N7 thực sự trả dao về đúng vị trí cũ. Đối với dao thứ hai, block N10 đăng ký vị trí dao hiện hành, block N15 buộc dao phải trở về vị trí đó.

Các block quan trọng cần xem xét là N7 và N10. Block N7 là *vị trí thay dao* của dao thứ nhất, block N10 là *đăng ký vị trí dao* của dao thứ hai – *cả hai dao này ở cùng vị trí trên hộc dao*. Sự khác biệt giữa các giá trị XZ phản ánh sự khác biệt chiều dài chìa ra của từng dao trên ổ dao. Toàn bộ điều được thực hiện với lệnh G50 là báo cho hệ điều khiển *vị trí đầu dao tính từ zero chương trình*. Bạn cần nhớ kỹ điều đó.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

16 BÙ VỊ TRÍ

Quan hệ giữa các điểm quy chiếu trong lập trình CNC được biểu thị theo các giá trị số được xác lập trước. Các số này, các giá trị chuyên biệt được yêu cầu *trước khi* thực sự tiến hành xác lập máy. Trong khi lập trình gia công, nhiều kích thước đã biết một cách chính xác, số khác chỉ biết gần đúng, và một số kích thước hoàn toàn chưa biết. Nhiều kích thước đã biết có thể thay đổi giữa các nguyên công. Nếu không có cách thức hiệu chỉnh, sẽ hầu như không thể xác lập máy một cách chính xác và hiệu quả. Các bộ điều khiển hiện đại cung cấp nhiều tính năng giúp cho sự lập trình và xác lập máy trở nên dễ dàng hơn, nhanh và chính xác hơn. Các hệ tọa độ, bù vị trí và bù dạng hình học, là những công cụ hỗ trợ đắc lực được dùng trong lập trình để hiệu chỉnh các sai số.

Một trong các kỹ thuật lập trình cũ nhất được dùng khi viết chương trình là *bù vị trí*. Sử dụng các hàm bù vị trí, vị trí *thực* của dụng cụ cắt được bù cho vị trí *lý thuyết* của dao đó.

Đây chỉ là một trong nhiều phương pháp hiệu chỉnh khả dụng cho nhà lập trình và người vận hành máy. Trên các hệ thống CNC hiện đại, phương pháp này vẫn được sử dụng để tương thích với các chương trình cũ. Ngày nay, kỹ thuật này thực sự không cần thiết, đã được thay bằng *Work Offsets* (*Work Coordinate System*) linh hoạt và hiệu quả hơn, sẽ được trình bày trong Chương 18. Chương này chỉ đề cập một số ứng dụng lập trình có thể có ích khi sử dụng phương pháp bù vị trí kiểu cũ.

MÔ TẢ

Mục đích chính của bù vị trí là hiệu chỉnh chênh lệch giữa vị trí zero máy và vị trí zero chương trình của dụng cụ cắt. Trong thực tế, điều này được dùng trong các trường hợp như khoảng cách giữa hai điểm quy chiếu có thể biến đổi hoặc hoàn toàn chưa biết. Ví dụ, khi gia công vật đúc, zero chương trình được lấy từ bề mặt vật đúc có thể thay đổi rõ rệt. Sử dụng bù vị trí sẽ không cần thường xuyên thay đổi chương trình hoặc chỉnh lại sự gá lắp. Nói chung, chi tiết được lắp trong đồ gá trên bàn máy và toàn bộ gá lắp sẽ được bù. Vì lý do đó, sự bù vị trí đôi khi được gọi là *bù đồ gá* hoặc *bù bàn máy*.

Tương tự một số hàm khác, sự bù vị trí là phương pháp lập trình đòi hỏi giá trị nhập của người vận hành máy CNC. Nhà lập trình chuyên biệt kiểu bù và số đăng ký bộ nhớ, người vận hành nhập giá trị thực ở máy, sử dụng màn hình hiển thị tương ứng, trong khi gá lắp chi tiết.

Các lệnh lập trình

Trên Fanuc và các bộ điều khiển tương tự, có bốn lệnh chuẩn bị (mã G) khả dụng để lập trình các hàm bù vị trí.

G45	*Tăng đơn* theo hướng lập trình với giá trị bù
G46	*Giảm đơn* theo hướng lập trình với giá trị bù
G47	*Tăng kép* theo hướng lập trình với giá trị bù
G48	*Giảm kép* theo hướng lập trình với giá trị bù

Những định nghĩa này dựa trên các giá trị bù *dương* được lưu trong bộ ghi điều khiển. Nếu các giá trị lưu là âm, ý nghĩa của tất cả các định nghĩa chỉ có hiệu lực khi đổi dấu. Cả bốn lệnh chuẩn bị này đều không có tính chế độ và chỉ có hiệu lực bên trong block của chúng. Nếu được yêu cầu trong nhiều block, chúng phải được lặp lại trong từng block kế tiếp.

Định dạng lập trình

Mỗi mã G (từ G45 đến G48) đều liên quan đến giá trị bù vị trí đặc thù, được lập trình với địa chỉ H. Địa chỉ H trỏ đến vị trí số lưu trữ trong bộ nhớ của hệ thống điều khiển. Trên hầu hết các bộ điều khiển Fanuc, ký tự lập trình có thể thể là chữ D với cùng ý nghĩa như H. Địa chỉ H hoặc D được dùng trong chương trình là tùy thuộc vào xác lập thực của tham số trong hệ thống điều khiển.

Định dạng lập trình của hàm bù vị trí thường như sau:

```
G91 G00 G45 X.. H..
```

hoặc

```
G91 G00 G45 X..D..
```

Trong đó mã G tương ứng (G45 đến G48) được nối tiếp bằng vị trí đích và số khu vực trong bộ nhớ (sử dụng địa chỉ H hoặc D)

Bạn hãy chú ý định dạng nêu trên sử dụng các chế độ *số gia* (tương đối) và *chuyển động nhanh* và chỉ *một trục*. Nói chung, giá trị bù thường được áp dụng cho cả hai trục X và Y. Tuy nhiên, chỉ một giá trị đo được lưu theo số H hoặc D. Do hầu như chắc chắn giá trị bù sẽ khác nhau đối với từng trục, chúng cần chuyên biệt trong các block riêng rẽ với hai giá trị bù H (hoặc D) *khác nhau*, ví dụ:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

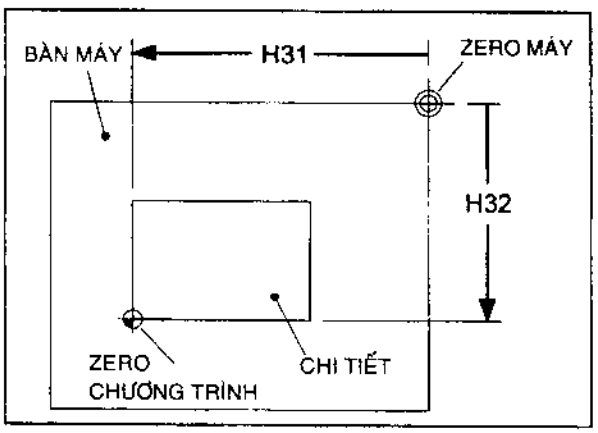
```
G91 G00 G45 X.. H31          (H31 STORES THE X VALUE)
G45 Y.. H32                  (H32 STORES THE Y VALUE)
```

hoặc

```
G91 G00 G45 X.. D31          (D31 STORES THE X VALUE)
G45 Y.. D32                  (D32 STORES THE Y VALUE)
```

Địa chỉ H còn được dùng với kiểu bù khác, được gọi là *bù chiều dài dao*, được trình bày trong Chương 18. Địa chỉ D cũng được dùng với kiểu bù khác, *bù bán kính dao* (Chương 29).

Mã G chuẩn bị sẽ xác định cách thức diễn dịch địa chỉ D hoặc H. Trong các ví dụ sẽ sử dụng địa chỉ H (Hình 16.1).



Hình 16.1. *Bù vị trí – khái niệm chung*

Chế độ số gia

Có thể phát sinh câu hỏi tại sao chuyển động bù là theo chế độ số gia. Bạn cần nhớ mục đích của bù vị trí là hiệu chỉnh sự chênh lệch giữa zero máy và zero chương trình. Công dụng chính là khi khởi đầu chuyển động dao *từ* vị trí zero máy. Theo mặc định, và không có sự bù bất kỳ, các xác lập tọa độ hoặc bù hoạt động, zero máy là zero tuyệt đối, là zero *duy nhất* hệ thống điều khiển "biết" vào thời điểm đó.

Bạn hãy xem ví dụ gồm vài block, được lập trình ở đầu chương trình với lệnh bù vị trí:

```
N1 G20
N2 G17 G80 T01
N3 M06
N4 G90 G00 G45 X0 H31                (NO X MOTION)
N5 G45 Y0 H32                        (NO Y MOTION)
N6 ...
```

Ví dụ này minh họa chuyển động từ zero máy (vị trí dao hiện hành) đến zero chương trình, là vị trí đích, dọc theo các trục XY. Bạn hãy chú ý xác lập chế độ tuyệt đối G90 trong block N4. Giả sử hệ thống điều khiển được xác lập theo H31 = -12.0000 inch. Bộ điều khiển sẽ đánh giá block này và diễn dịch theo ý đồ của nhà lập trình để đến zero tuyệt đối, do G90 chuyên biệt. Bộ điều khiển kiểm tra vị trí hiện hành, nhận thấy vị trí đó đã ở zero tuyệt đối và do đó *không có làm gì cả*. Ở đây sẽ *không có chuyển động*, bất kể xác lập giá trị bù, nếu chuyển động tuyệt đối được lập trình đến vị trí đích X0 hoặc Y0. Nếu G90 được đổi thành G91, từ chế độ tuyệt đối sang số gia (tương đối), sẽ có chuyển động dọc theo chiều *âm* của trục X, với khoảng cách đúng 12 inch, và sẽ có chuyển động tương tự dọc theo trục Y, trong block N5. Kết luận? *Bạn hãy dùng các lệnh bù vị trí chỉ theo chế độ số gia G91.*

Tính toán chiều dài chuyển động

Bạn hãy xem xét kỹ hơn cách thức hệ thống điều khiển diễn dịch block bù vị trí. Sự diễn dịch cách thức hệ điều khiển xử lý các số là rất quan trọng để hiểu cơ chế hoạt động của sự bù vị trí. Sự tăng đơn được lập trình với lệnh G45 và giảm đơn được lập trình với G46. Tại thời điểm này chưa cần đề cập các lệnh G47 và G48. Do cả hai lệnh G45 và G46 đều có quan hệ với một trục và với một địa chỉ H, cần đánh giá mọi khả năng kết hợp có thể xảy ra.

- Hoặc G45 (*tăng*) hoặc G46 (*giảm*) được lập trình.
- Trục đích có thể có giá trị *zero*, *dương*, hoặc *âm*.
- Lượng bù có thể có giá trị *zero*, *dương*, hoặc *âm*.

Trong lập trình, điều quan trọng là xác lập các tiêu chuẩn, và tuân thủ các tiêu chuẩn đó. Ví dụ, trên trung tâm gia công đứng, giá trị bù được đo *từ* zero máy *đến* zero chương trình. Điều này có nghĩa là chiều âm từ điểm quan sát của người vận hành. Quyết định logic sẽ là xác lập các giá trị bù âm được dùng làm tiêu chuẩn.

Điều rất quan trọng là hiểu *cách thức* bộ điều khiển diễn dịch thông tin trong block. Trong bù vị trí, bộ điều khiển đánh giá giá trị lưu trong bộ nhớ được địa chỉ H (hoặc D) gọi ra. Nếu giá trị này là zero, sự bù không xảy ra. Nếu giá trị của H được lưu là âm, bộ điều khiển cộng giá trị này vào giá trị của vị trí trục đích, kết quả sẽ là chiều và độ dài chuyển động. Ví dụ, giả sử địa chỉ 431 lưu giá trị -15.0 inch, vị trí hiện hành của máy là vị trí zero, và xác lập trục trên bộ điều khiển cũng là zero, block:

```
G91 G00 G45 X0 431
```

sẽ được diễn dịch như sau:

```
-15.0 + 0 = -15.0000
```

Kết quả sẽ là chuyển động âm 15.0 inch dọc theo trục X.

Nếu vị trí đích trên trục X không phải zero, và là số dương, công thức hoàn toàn tương tự:

```
G91 G00 G45 X1.5 431
```

sẽ được diễn dịch như sau:

```
-15.0 + 1.5 = -13.5000
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều khi, yêu cầu này là không thỏa mãn

```
G91 G00 G45 X-1.5 H31
```

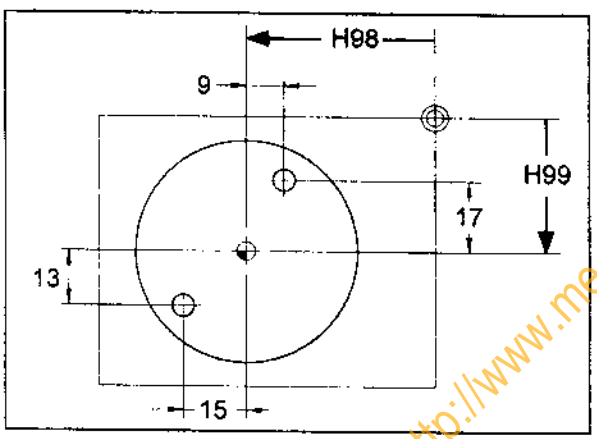
Ở đây, chuyển động sẽ tiến theo chiều *dương* của trục X, kết quả sẽ là vượt quá hành trình trên trục X. Do giá trị X là âm, lệnh G45 không được phép sử dụng, thay vào đó bạn *phải dùng* lệnh G46.

```
G01 G00 G46 X-1.5 H31
```

sẽ được diễn dịch như sau:

```
-15.0 + (-1.5) = -15.0000 - 1.5= -16.5000
```

Lệnh G45 được đưa ra khỏi chương trình và giá trị bù âm được đổi thành giá trị dương. Điều này dễ gây nhầm lẫn và không ổn định, nhưng vẫn bảo đảm hoạt động tốt. Để xem xét các khả năng, chương trình O1701 dịch chuyển từ zero máy đến các vị trí khác và trở về zero máy (lệnh G28 bảo đảm sự trở về zero máy, sẽ được giải thích trong Chương 20).



Hình 16.2. *Bù vị trí áp dụng cho các vị trí đích khác nhau: zero, dương và âm (chương trình O1601)*

Hình 16.2 minh họa chương trình O1601. Logic áp dụng cho các trục X và Y là hoàn toàn như nhau. Chương trình được viết theo đơn vị hệ mét và được kiểm nghiệm trên bộ điều khiển Fanuc 11M, với địa chỉ H (địa chỉ D cũng cho kết quả tương tự). Các giá trị bù vị trí H98 và H99 được xác lập như sau:

```
H98 = -250.000
H99 = -150.000
```

cho các trục X và Y. Các lệnh chế độ không cần lặp lại:

```
O1601 (G45 AND G46 TEST)
N1 G21 G17
N2 G92 X0 Y0 Z0
N3 G90 G00 G45 X0 H98          (ABS X0 TARGET)
N4 G46 Y0 H99                  (ABS Y0 TARGET)
N5 G28 X0 Y0

N6 G91 G00 G4 X0 H98           (INC X0 TARGET)
N7 G46 Y0 H99                  (INC Y0 TARGET)
N8 G28 X0 Y0

N9 G90 G00 G45 X9.0 H98        (ABS X+ TARGET)
N10 G46 Y17.0 H99              (ABS Y+ TARGET)
N11 G28 X0 Y0

N12 G91 G00 G45 X9.0 H98       (INC X+ TARGET)
N13 G46 Y17.0 H99              (INC Y+ TARGET)
N14 G28 X0 Y0

N15 G90 G00 G45 X-15.0 H98     (ASS X- TARGET)
N16 G46 Y-13.0 H99             (ABS Y- TARGET)
N17 G28 X0 Y0

N18 G91 G00 G45 X-15.0 H98     (INC X- TARGET)
N19 G46 Y-13.0 H99             (INC Y- TARGET)
N20 G28 X0 Y0
N21 M30
%
```

Hệ thống điều khiển từng block chuyển động một cách riêng rẽ – hoặc theo đúng ý định của nhà lập trình hoặc không đúng (ký hiệu O/T có nghĩa là *vượt quá hành trình*, trước ký hiệu này là trục và chiều bị vượt quá hành trình):

```
N3   G90  →  G45  →  0  .  .  .  no motion
N4   G90  →  G46  →  0  .  .  .  no motion
N6   G91  →  G45  →  0  .  .  .  X-250.0
N7   G91  →  G46  →  0  .  .  .  Y+ O/T

N9   G90  →  G45  →  +  .  .  .  X-241.0
N10  G90  →  G46  →  +  .  .  .  Y+ O/T
N12  G91  →  G45  →  +  .  .  .  X-241.0
N13  G91  →  G46  →  +  .  .  .  Y+ O/T

N15  G90  →  G45  →  -  .  .  .  X+ O/T
N16  G90  →  G46  →  -  .  .  .  Y-163.0
N18  G91  →  G45  →  -  .  .  .  X+ O/T
N19  G91  →  G46  →  -  .  .  .  Y-163.0
```

Bù vị trí theo trục Z

Tính năng bù vị trí chỉ áp dụng cho các trục X và Y, và không áp dụng cho trục Z. Trong hầu hết các trường hợp, trục Z được điều khiển bằng bù khác – được gọi là *bù chiều dài dụng cụ cắt* (Chương 18). Nếu trục Z được lập trình với lệnh G45 hoặc G46, trục đó cũng sẽ bị tác động.

Sử dụng G47 và G48

Trong các ví dụ đã nêu, tính năng bù vị trí chỉ được dùng giữa zero máy và zero chương trình, là phương pháp xác định một cách chính xác vị trí của chi tiết trên bàn máy. Sự tăng đơn sử dụng lệnh G45 và sự giảm đơn sử dụng G46, do chúng là các lệnh duy nhất cần sử dụng.

Các lệnh G47 (tăng kép) và G48 (giảm kép) chỉ cần thiết cho sự bù bán kính được đơn giản hóa *đến mức* cao nhất, và chúng đã lạc hậu nhưng vẫn còn sử dụng.

Phay mặt

Trong phần kế tiếp (Chương 17), các nguyên lý phay mặt sẽ được trình bày chi tiết. Trong chương đó sẽ nêu rõ ví dụ về phương pháp bù vị trí được áp dụng để bù đường kính dao phay mặt ở vị trí trống trải, bất kể kích thước của dao. Đây có lẽ là khả năng duy nhất có thể sử dụng các lệnh G45 và G46 trong lập trình hiện đại.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

17 BÙ CHI TIẾT

Sử dụng phương pháp *Work Offset* (bù chi tiết) để định vị dụng cụ cắt dựa trên zero máy nhanh và hiệu quả hơn nhiều so với các phương pháp cũ sử dụng các hàm bù vị trí G45 và G46 (Chương 16). Bù chi tiết còn được gọi là *Work Coordinate System* (hệ tọa độ làm việc) hoặc *Fixture Offsets* (bù đồ gá). Bù chi tiết hiệu quả hơn so với sử dụng các lệnh đăng ký vị trí G92 (hệ thống phay) hoặc G50 (hệ thống tiện). Nhà lập trình CNC nếu không biết ý nghĩa của các hàm *bù vị trí* hoặc các lệnh *đăng ký vị trí*, có lẽ chỉ làm việc với các máy CNC hiện đại. Tuy nhiên, vẫn còn nhiều máy CNC trong công nghiệp sử dụng các hàm đã lạc hậu đó. Hiểu biết về chúng sẽ tăng số lượng công cụ lập trình khả dụng.

Chương này trình bày các phương pháp hiện đại phối hợp quan hệ giữa vị trí zero máy và điểm quy chiếu zero chương trình. Trong các ví dụ sẽ sử dụng tính năng *Work Coordinate System* trên hệ thống điều khiển hiện đại. Bạn hãy coi sự bù chi tiết là sự *chỉnh thẳng hàng* giữa hai hoặc nhiều hệ tọa độ

VÙNG LÀM VIỆC KHẢ DỤNG

Trước khi tìm hiểu chi tiết, bạn cần biết về hệ tọa độ làm việc – hoặc *bù chi tiết*. Bù chi tiết là phương pháp cho phép nhà lập trình CNC lập trình gia công mà không cần biết vị trí chính xác của chi tiết trên bàn máy. Đây là cách tiếp cận tương tự phương pháp bù vị trí nhưng linh hoạt và tiên tiến hơn. Trong hệ thống bù chi tiết, đến sáu chi tiết có thể được gá lắp trên bàn máy, mỗi chi tiết có chỉ số bù riêng. Nhà lập trình có thể dịch chuyển dụng cụ cắt từ chi tiết này đến chi tiết khác một cách dễ dàng. Để đạt được điều này, cần có lệnh chuẩn bị để kích hoạt sự bù chi tiết, phần còn lại sẽ do hệ thống điều khiển thực hiện. Hệ thống sẽ tự động thực hiện sự điều chỉnh bất kỳ về sự khác biệt giữa hai vị trí chi tiết.

Khác với hàm bù chi tiết, hai, ba, hoặc nhiều trục có thể dịch chuyển đồng thời với sự bù chi tiết, dù trục Z trên trung tâm gia công được điều khiển một cách độc lập, sử dụng các lệnh bù chiều dài dụng cụ cắt G43 hoặc G44. Các lệnh liên quan với sự bù trục sẽ được trình bày chi tiết trong Chương 18.

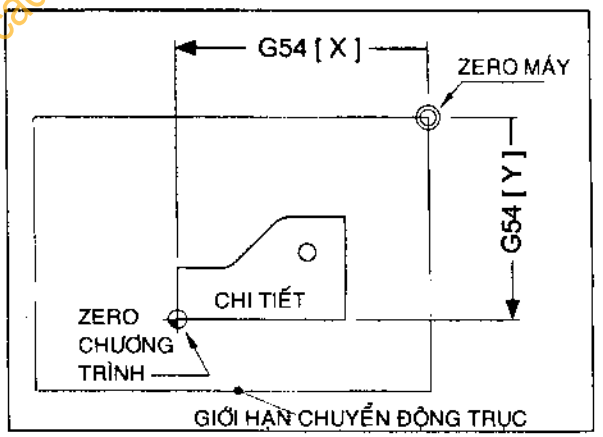
Trong bù vị trí, để chuyển sự gia công từ chi tiết này sang chi tiết khác trong cùng một xác lập chung, chương trình chứa chỉ số bù khác từ zero chương trình của chi tiết trước đó. Sử dụng phương pháp bù chi tiết, *mọi zero chương trình đều được đo từ vị trí zero máy*, thường đến sáu giá trị bù, đôi khi cao hơn.

Sáu hệ tọa độ làm việc – hoặc *bù chi tiết* – khả dụng trên hệ thống điều khiển Fanuc, được gán các lệnh chuẩn bị như sau

G54 G55 G56 G57 G58 G59

Khi hệ điều khiển bắt đầu hoạt động, hệ tọa độ mặc định thường là G54.

Về cơ bản, sự bù chi tiết thiết lập sáu vùng làm việc độc lập, đây là tính năng tiêu chuẩn. Các giá trị nhập vào bộ CNC luôn luôn là các khoảng cách được đo *từ zero máy đến zero chương trình*. Do có đến sáu vùng làm việc, có thể xác định đến sáu vị trí zero chương trình độc lập. Hình 17.1 nêu rõ các quan hệ cơ bản, sử dụng xác lập G54 mặc định.



Hình 17.1. *Các quan hệ cơ bản, sử dụng xác lập G54 mặc định*

Các quan hệ cơ bản của sự bù chi tiết mặc định (G54) cũng áp dụng theo cùng cách thức đối với năm lệnh bù chi tiết còn lại (G55 đến G59). Các giá trị lưu trong hệ thống điều khiển luôn luôn được đo từ vị trí zero máy *đến* zero chương trình của từng chi tiết, theo sự xác định của nhà lập trình CNC.

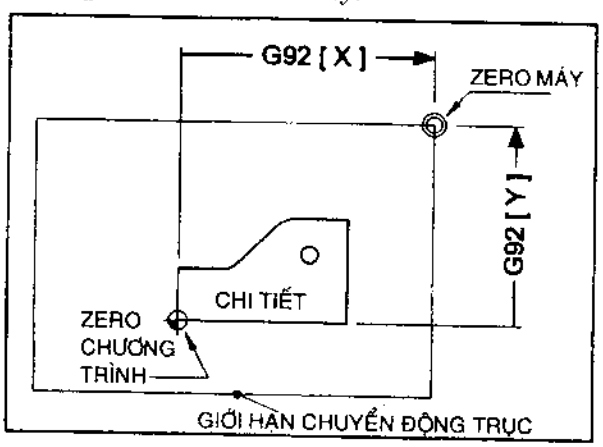
Khoảng cách *từ* zero máy *đến* zero chương trình của từng vùng làm việc được đo riêng rẽ theo các trục X và Y, được nhập vào bộ ghi bù chi tiết tương ứng của hệ điều khiển. Bạn hãy chú ý, chiều đo là *từ* zero máy *đến* zero chương

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

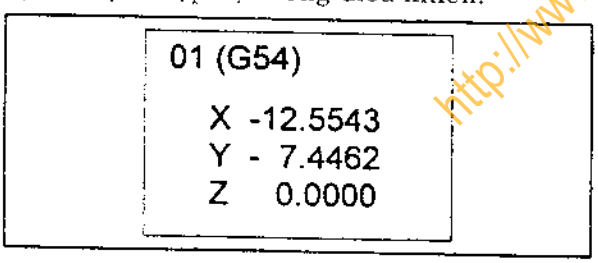
trình. Nếu chiều này là âm, dấu âm phải được nhập vào màn hình bù.

Để so sánh với lệnh đăng ký vị trí G92, Hình 17.2 minh họa sự gá lắp chi tiết với phương pháp cũ sử dụng G92 và zero máy là điểm khởi đầu. Bạn hãy chú ý sự gán các mũi tên là *ngược chiều*, cho biết chiều đo *từ* zero chương trình *đến* zero máy.



***Hình 17.2.** Các quan hệ cơ bản của lệnh đăng ký vị trí G92*

Đối với các lệnh bù chi tiết G54 đến G59, mục nhập vào bộ ghi vị trí bù tọa độ sẽ là giá trị âm trên trục X, giá trị âm trên trục Y, là zero trên trục Z, đối với hầu hết các trung tâm gia công đứng. Điều này do người vận hành CNC thực hiện tại máy. Hình 17.3 minh họa ví dụ về mục nhập hệ thống điều khiển.



***Hình 17.3.** Mục nhập dữ liệu đối với hệ tọa độ làm việc G54*

Bằng cách dùng các xác lập G54 đến G59 trong chương trình, hệ điều khiển chọn các khoảng cách đo đã lưu và dao cắt có thể chuyển động đến vị trí bất kỳ bên trong vùng làm việc đã chọn đồng thời theo cả hai trục X và Y, mỗi khi cần thiết.

Vị trí chi tiết trên bàn máy thường được biết trong khi lập trình. Mục đích chính của bù chi tiết đồng bộ hóa vị trí thực của chi tiết trong quan hệ với vị trí zero máy.

Bù chi tiết bổ sung

Tiêu chuẩn sáu hệ tọa độ làm việc (bù chi tiết) là đủ cho nhiều dạng gia công. Tuy nhiên có một số công việc có thể đòi hỏi gia công với nhiều điểm quy chiếu chương trình, ví dụ chi tiết khối đa diện trên bàn máy gia công ngang. Hiện có các tùy chọn nào, ví dụ nếu chi tiết gia công đòi hỏi đến mười hệ tọa độ làm việc?

Fanuc cung cấp - *ở dạng tùy chọn* – đến 48 sự bù chi tiết bổ sung, tổng cộng là 54 (6+48). Nếu tùy chọn này khả dụng trên hệ thống CNC, sự bù bất kỳ trong 48 sự bù này có thể được sử dụng bằng cách lập trình mã G đặc biệt:

G54.1 P..	Chọn bù chi tiết bổ sung với P = 1 đến 48

➲ **Ví dụ: G54.1 P..**

G54.1 P1	Chọn bù chi tiết bổ sung 1
G54.1 P2	Chọn bù chi tiết bổ sung 2
G54.1 P3	Chọn bù chi tiết bổ sung 3
G54.1 PX..	Chọn bù chi tiết bổ sung X..
G54.1 P48	Chọn bù chi tiết bổ sung 48

Sử dụng các lệnh bù chi tiết bổ sung trong chương trình hoàn toàn như các lệnh tiêu chuẩn:

```
N2 G90 G00 G54 P1 X5.5 Y3.1 S1000 M03
```

Hầu hết các bộ điều khiển Fanuc cho phép bỏ qua phần thập phân của G54.1, mà không gây ra vấn đề gì:

```
N2 G90 G00 G54 P1 Y3.1 S1000 M03
```

Sự hiện diện của hàm P1 đến P48 bên trong block sẽ chọn sự bù chi tiết *bổ sung*. Nếu không có tham số P1 đến P48, hệ điều khiển sẽ chọn lệnh G54 mặc định.

KHỞI ĐẦU VÀ MẶC ĐỊNH BÙ CHI TIẾT

Nếu không có lệnh bù chi tiết được chuyên biệt trong chương trình và hệ điều khiển hỗ trợ sự bù chi tiết, bộ điều khiển sẽ tự động chọn G54 – đây là *sự lựa chọn mặc định bình thường*. Trong lập trình, bạn nên chọn lệnh bù chi tiết và các hàm mặc định khác, dù G54 mặc định được dùng thường xuyên trong các chương trình. Người vận hành máy sẽ có cảm tình hơn với chương trình CNC. Bạn cần nhớ, bộ điều khiển vẫn có các tọa độ chi tiết chính xác được lưu trong bộ ghi G54

Trong chương trình, sự bù chi tiết có thể được thiết lập theo hai cách – trong block riêng, không có thông tin bổ sung, ví dụ:

```
N1 G54
```

Lệnh bù chi tiết có thể được lập trình như một phần của block khởi đầu, thường ở đầu chương trình hoặc ở đầu của từng dụng cụ cắt:

```
N1 G17 G40 G80 G54
```

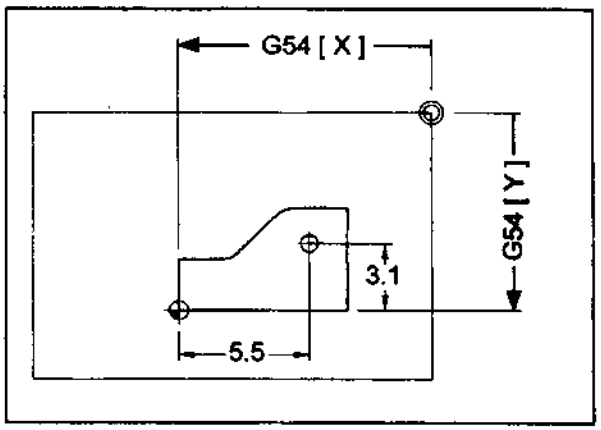

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ứng dụng phổ biến nhất là lập trình mã G tương ứng trong từng block như chuyển động dụng cụ cắt thứ nhất:

```
N40 G00 G54 X5.5 Y3.1 S1500 M03
```

Hình 17.4 minh họa kỹ thuật này. Trong block N40 nêu trên, vị trí tuyệt đối của dao cắt là X5.5 Y3.1, *bên trong* G54. Điều gì *thực sự* xảy ra khi xử lý block này?



Hình 17.4. *Chuyển động trực tiếp của dao cắt đến vị trí cho trước sử dụng lệnh G54*

Bạn hãy chú ý, không có các giá trị X hoặc Y liên quan với G54 trong Hình 17.4; thực tế không cần các giá trị đó. Người vận hành CNC đặt chi tiết ở *vị trí thích hợp bất kỳ* trên bàn máy, kẹp chặt, xác định khoảng cách giữa zero chương trình và zero máy, sau đó nhập các giá trị đó vào bộ ghi điều khiển, dưới tiêu đề G54; có thể nhập bằng tay hoặc tự động.

Giả sử xét thời điểm sau khi gá lắp, các khoảng cách đo từ zero máy đến zero chương trình là X-12.5543 và Y-7.4462. Máy tính sẽ xác định chuyển động thực bằng phép toán đơn giản – luôn là phép *cộng* giá trị đích X lập trình với giá trị X đã đo, giá trị Y lập trình với giá trị Y đã đo.

Chuyển động thực của dụng cụ cắt trong block N40 sẽ là:

```
X=   -12.5543 + 5.5 = -7.0543
Y=    -7.4462 + 3.1 = -4.3462
```

Các tính toán này hoàn toàn không cần thiết trong lập trình hàng ngày, chúng chỉ hữu ích để hiểu phương pháp bộ điều khiển diễn dịch dữ liệu đã cho.

Toàn bộ tính toán này là rất ổn định, có thể đưa vào công thức đơn giản. Để đơn giản, các xác lập của sự bù EXT (ngoài hoặc chung) không gộp vào công thức này, chúng sẽ được giải thích ở cuối chương.

```
A = M + P
```

A: chiều dài chuyển động thực (khoảng cách được hiển thị)
M: khoảng cách đo từ zero máy
P: Vị trí đích tuyệt đối được lập trình (giá trị trục)

Bạn cần rất cẩn thận khi cộng giá trị âm, do đây là tổng đại số.

Chẳng hạn:

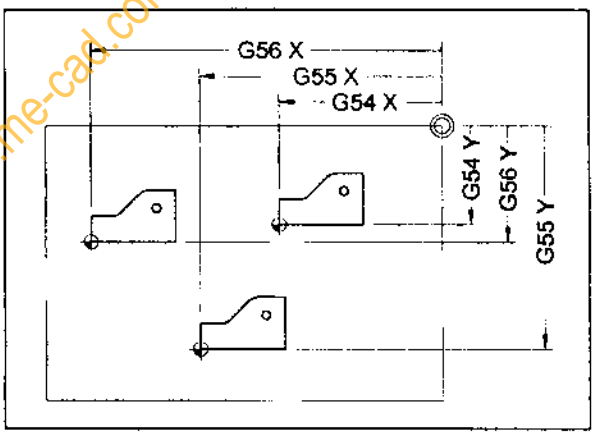
$-10 + (-12) = -10 - 12 = -22$

Nếu có sự bù chi tiết khác được lập trình, sự bù đó sẽ tự động thay thế giá trị bù cũ, *trước* khi xảy ra chuyển động thực của dao cắt.

Thay đổi giá trị bù chi tiết

Trong chương trình CNC, có thể sử dụng một, hai, hoặc tất cả các lệnh bù chi tiết khả dụng. Nếu sử dụng tất cả các lệnh bù chi tiết, xác lập bù sẽ lưu *khoảng cách từ zero máy đến zero chương trình của từng chi tiết trong nhóm gá lắp.*

Ví dụ, nếu có ba chi tiết lắp trên bàn máy, mỗi chi tiết đều có vị trí zero chương trình riêng liên quan với một mã G bù chi tiết.



Hình 17.5. *Sử dụng nhiều lệnh bù chi tiết trong một gá lắp và một chương trình*

Bạn hãy so sánh mọi chuyển động khả dĩ trên Hình 17.5.

```
G90 G00 G54 X0 Y0
```

... là chuyển động nhanh từ vị trí dao cắt *hiện hành* đến vị trí zero chương trình của chi tiết *thứ nhất*.

```
G90 G00 G55 X0 Y0
```

... là chuyển động nhanh từ vị trí dao cắt *hiện hành* đến vị trí zero chương trình của chi tiết *thứ hai*.

```
G90 G00 G56 X0 Y0
```

... là chuyển động nhanh từ vị trí dao cắt

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

hiện hành đến vị trí zero chương trình của chi tiết *thứ hai*.

Vị trí đích *không hẳn* là zero chi tiết (zero chương trình) như trong ví dụ nêu trên. Nói chung, dao sẽ chuyển động đến đúng vị trí cắt gọt thứ nhất, để tiết kiệm thời gian chu kỳ. Chương trình ví dụ dưới đây sẽ minh họa kỹ thuật đó.

Trong ví dụ này, lỗ sẽ được khoan điểm trên từng chi tiết (có 3 chi tiết) đến chiều sâu tính toán Z-0.17 (Chương trình O1701). Bạn hãy nghiên cứu tính đơn giản của sự chuyển tiếp giữa các mã G bù chi tiết, không có sự xóa, chỉ có mã G mới, sự bù chi tiết mới. Bộ điều khiển sẽ thực hiện phần còn lại.

```
O1701
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X5.5 Y3.1 S1000 M03 (G54 USED)
N4 G43 Z0.1 H01 M08
N5 G99 G82 R0.1 Z-0.14 P100 F8.0
N6 G55 X5.5 Y3.1                (SWITCH TO G55)
N7 G56 X5.5 Y3.1                (SWITCH TO G56)
N8 G80 Z1.0 M09
N9 G91 G54 G28 Z0 M05           (SWITCH TO G54)
N10 M01
...
```

Các block N3 đến N5 liên quan với chi tiết thứ nhất, bên trong lệnh bù G54. Block N6 sẽ khoan điểm chi tiết thứ hai, bên trong lệnh bù G55, block N7 sẽ khoan điểm chi tiết thứ ba trong lệnh bù G56. Bạn hãy chú ý sự trả về lệnh G54 trong block N9.

Sự trả về hệ tọa độ mặc định là không cần thiết, nhưng nên lập trình khi dao hoàn tất sự cắt gọt. Sự lựa chọn bù chi tiết có tính chế độ, bạn hãy chú ý sự chuyển tiếp giữa các dụng cụ cắt từ mã G bù này sang mã G bù khác. Trở về G54 mặc định sẽ rất hữu ích ở cuối từng dao cắt.

Nếu tất cả các block này đều trong *cùng một* chương trình, bộ điều khiển sẽ tự động xác định *sự khác biệt* (hiệu số) giữa vị trí dao hiện hành và cùng vị trí dao đó trong mã G bù chi tiết kế tiếp. Đây là ưu điểm lớn nhất khi sử dụng các lệnh bù chi tiết, hơn hẳn sự bù vị trí và đăng ký vị trí. Mọi chi tiết được gá lắp có thể là đồng nhất hoặc khác nhau, khi chúng ở cùng các vị trí đối với toàn bộ nhóm gá lắp.

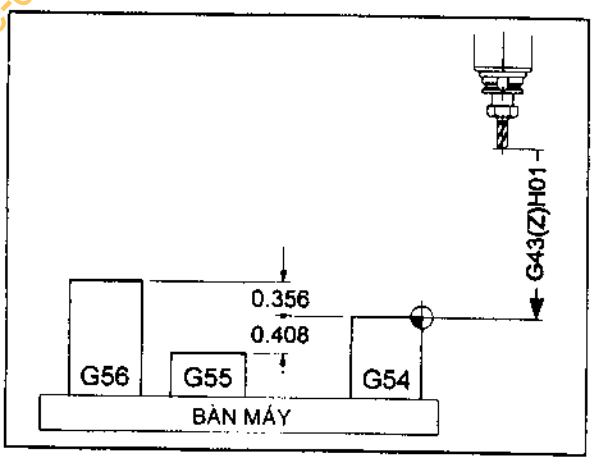
Ứng dụng trục Z

Cho đến đây vẫn chưa trình bày về trục Z trong nội dung liên quan đến sự bù chi tiết. Điều này không phải ngẫu nhiên, nhưng là hoàn toàn cố ý. Tuy sự bù chi tiết được chọn bất kỳ đều có thể áp dụng cho trục Z, và cùng logic như các trục X, Y, nhưng ở đây có phương pháp điều khiển trục Z tốt hơn. Phương pháp điều khiển trục Z dưới dạng các lệnh G43 và G44 liên quan chặt chẽ với *sự bù chiều dài dụng cụ cắt*, đôi khi được gọi là *sự bù dao*. Vấn đề này sẽ được trình bày riêng cho Chương 18. Trong hầu hết các ứng dụng lập trình, sự bù chi tiết chỉ được dùng trong mặt phẳng XY. Đây là xác lập hệ điều khiển thông dụng và có thể biểu thị bằng ví dụ dưới đây với các giá trị lưu trong bộ ghi điều khiển:

```
(G54) X-8.761 Y-7.819 Z0
(G55) X-15.387 Y-14.122 Z0
(G56) X-22.733 Y-8.352 Z0
(G57) ...
```

Mục nhập Z0 là rất quan trọng trong các ví dụ nêu trên và trong sự điều khiển máy. Z0 có nghĩa là xác lập tọa độ đối với giá trị Z (chiều cao của chi tiết) không thay đổi giữa các chi tiết, dù xác lập XY có thể thay đổi.

Trường hợp duy nhất cần xét trục Z trong xác lập bù chi tiết là khi chiều cao của các chi tiết trong cùng một nhóm gá lắp có các giá trị *khác nhau*. Nếu giá trị Z thay đổi, sự thay đổi đó phải được xét bằng cách hiệu chỉnh xác lập bộ ghi tọa độ trong bộ điều khiển. Điều này do người vận hành CNC chịu trách nhiệm thực hiện, nhưng nhà lập trình phải nắm vững.



Hình 17.6. *Xác lập các mã bù chi tiết khi chiều cao chi tiết khác nhau*

Hình 17.6 minh họa các khả năng được dùng cho các chi tiết có chiều cao khác nhau trong cùng nhóm gá lắp dụng cụ cắt. Sự khác biệt giữa các chiều cao chi tiết có thể dễ dàng nhận biết trên bản vẽ kỹ thuật hoặc đo đạc.

Nếu ví dụ về nhiều lệnh bù chi tiết đối với xác lập XY được chỉnh sửa để bù cả trục Z, sự bù chi tiết có thể được xác lập cho các chi tiết trong cùng nhóm gá lắp, nhưng có chiều cao khác nhau. Chiều cao biến thiên này do trục Z

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

điều khiển. Kết quả của xác lập trục Z phản ánh mối sai biệt dao giữa bề mặt trục Z được đo cho một chi tiết và bề mặt trục Z đo với chi tiết khác. Dựa theo dữ liệu trong ví dụ nêu trên, kết hợp với các giá trị Z trên Hình 17.6, các xác lập của hệ điều khiển sẽ như sau:

```
(G54) X-8.761 Y-7.819 Z0
(G55) X-15.387 Y-14.122 Z-0.408
(G56) X-22.733 Y-8.352 Z0.356
```

Điều quan trọng cần biết về điều khiển trục Z trong lệnh bù chi tiết đã chọn là sự điều khiển này có quan hệ *rất chặt chẽ* với sự bù chiều dài dụng cụ cắt, sẽ được trình bày trong Chương 18. Giá trị lưu của xác lập trục Z bên trong lệnh bù chi tiết sẽ áp dụng cho chuyển động dao thực sự và được dùng để *điều chỉnh* chuyển động này theo xác lập của lượng bù chiều dài dao.

Ví dụ, nếu lượng bù chiều dài dao của dụng cụ cắt được đo là Z-10.0, chuyển động thực của dao đó đến zero chương trình dọc theo trục Z sẽ là -10.0 bên trong lệnh G54, -10.408 trong G55 và -9.644 trong G56, tất cả đều sử dụng ví dụ minh họa trên Hình 17.6.

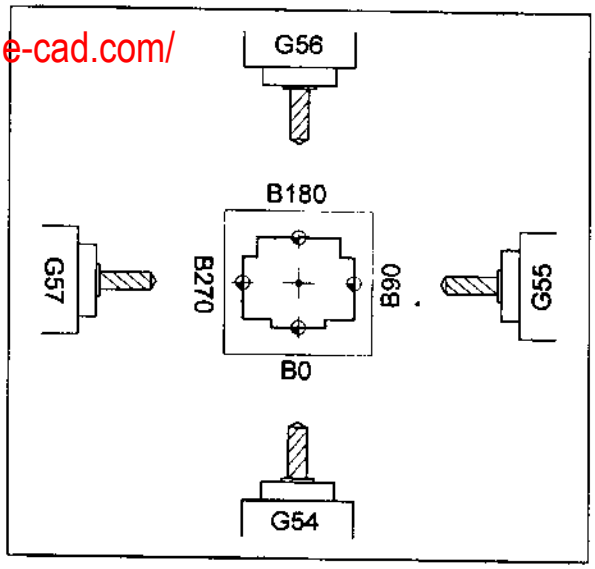
ÁP DỤNG TRÊN MÁY NGANG

Gia công nhiều chi tiết trong một nhóm gá lắp được thực hiện thường xuyên trên các trung tâm gia công CNC đứng. Kỹ thuật sử dụng nhiều lệnh bù chi tiết đặc biệt hữu ích trên các trung tâm gia công ngang, chẳng hạn phay và doa lỗ, nhiều bề mặt chi tiết được gia công trong một lần gá lắp.

Gia công hai, ba, hoặc nhiều bề mặt chi tiết trên trung tâm gia công ngang là công việc thường ngày trong các xưởng cơ khí hiện đại. Đối với mục đích này, sự lựa chọn bù chi tiết là rất quan trọng. Ví dụ, *zero chương trình* tại điểm xoay của bàn phân độ có thể xác lập cho các trục X và Y. Xác lập chương trình của trục Z có thể ở cùng vị trí (điểm xoay của bàn phân độ) hoặc trên bề mặt của từng vị trí phân độ. Sự bù chi tiết xử lý ứng dụng này rất chuẩn, đến sáu bề mặt với khoảng các mã G tiêu chuẩn.

Không có sự khác biệt trong ứng dụng lập trình – sự chuyển đổi giữa các lệnh G bù chi tiết hoàn toàn như trên trung tâm gia công đứng. Sự thay đổi duy nhất là trục Z sẽ lùi đến vị trí trống và sự phân độ bàn máy thường được lập trình giữa sự thay đổi bù chi tiết.

Hình 17.7 minh họa xác lập điển hình cho 4 bề mặt của chi tiết trong đó Z0 ở đỉnh của từng bề mặt đó. Số bề mặt tốt đa tương ứng số vị trí



Hình 17.7. *Ví dụ về bù chi tiết áp dụng cho trung tâm gia công ngang.*

phân độ bàn mày. Trong các trường hợp đó, phương pháp lập trình là tương tự nhau nếu Z0 ở tâm của bàn phân độ, đây cũng là ứng dụng phổ biến (Chương 45).

BÙ CHI TIẾT NGOÀI

Khi quan sát kỹ màn hình bù chi tiết bạn sẽ thấy một sự bù chi tiết đặc biệt ứng với một trong các ký hiệu dưới đây:

- 00 (EXT)
- 00 (COM)

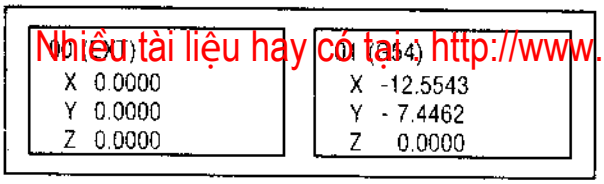
Hai số zero – 00 – cho biết sự bù này *không phải* là một trong 6 lệnh bù tiêu chuẩn G54 đến G59. Các lệnh bù này được ghi theo số từ 01 đến 06. Ký hiệu 00 còn cho biết đây không phải là sự bù lập trình được, ít nhất là không sử dụng các phương pháp lập trình CNC tiêu chuẩn. Tùy chọn *Fanuc Macro B* cho phép lập trình sự bù nêu trên.

Viết tắt *EXT* là từ *External* (ngoài) và *COM* là từ *Common* (chung). Bộ điều khiển có thể có một trong hai kí hiệu đó, nhưng không có cả hai. COM được ghi trên các bộ điều khiển cũ, còn EXT trên các bộ điều khiển hiện đại hơn. Lý do? Với sự bùng nổ thị trường máy tính cá nhân, viết tắt *COM* trở thành viết tắt chuẩn của *Communication* (truyền thông). Bộ điều khiển Fanuc hỗ trợ nhiều phương pháp truyền thông, kể cả nối kết với máy tính cá nhân, do đó COM được thay bằng EXT để tránh nhầm lẫn.

Hai ký hiệu này là như nhau, cả về công dụng và sự bù. Trên màn hiển thị sự bù đặc biệt này thường *ở trước* hoặc *trên* lệnh bù G54 (Hình 17.8).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 17.8. Ví dụ về hiển thị bù chi tiết EXT

Sự khác biệt chính giữa sự bù chi tiết EXT và COM là chúng không lập trình được với mã G bất kỳ. Xác lập của sự bù này thường là zero cho tất cả các trục. Xác lập khác zero sẽ kích hoạt sự bù chi tiết này theo cách thức rất quan trọng,

******** QUAN TRỌNG ********
Xác lập bất kỳ của sự bù chi tiết EXT luôn luôn tác động đến mọi xác lập bù được dùng trong chương trình CNC

Cả sáu lệnh bù chi tiết tiêu chuẩn, và mọi lệnh bù bổ sung, đều bị tác động từ các giá trị xác lập trong sự bù ngoài, dựa trên xác lập của từng trục. Do mọi hệ tọa độ lập trình đều bị ảnh hưởng, sự bù đặc biệt này được gọi là *Common Work Offset* hoặc *External Work Offset*.

CÁC ỨNG DỤNG TIỆN

Ngay từ đầu, hệ tọa độ làm việc chỉ được thiết kế cho các trung tâm gia công CNC, sau đó mới được áp dụng cho các máy tiện CNC. Sử vận hành, cả logic và thực tế, trên máy tiện là đồng nhất với trung tâm gia công. Sử dụng sự bù chi tiết trên máy tiện CNC cho phép loại bỏ lệnh G50 hoặc G92, sự gá lắp gia công tiện CNC trở nên nhanh hơn và dễ dàng hơn.

Các kiểu bù

Khác biệt chính trong ứng dụng bù chi tiết trên máy tiện là hầu như chỉ cần một sự bù chi tiết. Hai sự bù chi tiết là có thể, ba sự bù trở lên chỉ dùng cho các gá lắp phức tạp và đặc biệt. Từ G54 đến G59 đều khả dụng trên các máy tiện CNC hiện đại và thường bỏ qua sự lựa chọn bù chi tiết trong chương trình, trừ khi sử dụng từ hai lệnh bù chi tiết trở lên. Điều đó có nghĩa là về nguyên tắc nhà lập trình chỉ phụ thuộc vào xác lập mặc định G54.

Hai tính năng bù đặc biệt trên các hệ thống điều khiển hiện đại là bù *Geometry* (hình học) và bù *Wear* (mòn dao), trên cùng màn hình hiển thị hoặc trên các màn hình riêng rẽ, tùy theo kiểu loại của bộ điều khiển.

Bù hình học

Bù hình học (Geometry Offset) là tương đương với bù chi tiết (work offset) trên các bộ điều khiển phay. Lệnh bù này biểu thị khoảng cách từ điểm quy chiếu dao cắt đến zero chương trình, được đo từ zero máy dọc theo trục đã chọn. Nói chung, trên máy tiện với hộc chứa dao *phía trên* đường tâm trục chính, bù hình học cho cả trục X và trục Z sẽ là giá trị âm. Hình 17.9 minh họa các giá trị bù hình học cho mũi khoa, dao tiện và dao doa (T01, T02, T03).

GEOMETRY OFFSET

No.	X OFFSET	Z OFFSET	RADIUS	TIP
01	-8.6330	-2.3630	0.0000	0
02	-8.6470	-6.6780	0.0469	3
03	-9.0720	-2.4950	0.0313	2
04	0.0000	0.0000	0.0000	0
05	0.0000	0.0000	0.0000	0

Hình 17.9. Mục nhập dữ liệu bù Hình Học dao tiện

Bù mòn dao

Bù mòn dao cũng được dùng trên các bộ điều khiển phay, nhưng chỉ có bù chiều dài dao và bù bán kính dao, không có bù hệ tọa độ làm việc (bù chi tiết).

Trên các máy tiện CNC, công dụng của bù mòn dao hoàn toàn như trên trung tâm gia công. Điều này bù cho sự mòn dụng cụ cắt và được dùng để tinh chỉnh bù hình học. Về nguyên tắc, sau khi xác lập bù hình học cho dụng cụ cắt, xác lập đó phải không thay đổi. Các điều chỉnh và tinh chỉnh kích thước thực của chi tiết chỉ được thực hiện với sự bù mòn dao.

WEAR OFFSET

No.	X OFFSET	Z OFFSET	RADIUS	TIP
01	0.0000	0.0000	0.0000	0
02	-0.0060	0.0000	0.0469	3
03	0.0000	0.0040	0.0313	2
04	0.0000	0.0000	0.0000	0
05	0.0000	0.0000	0.0000	0

Hình 17.10. Mục nhập dữ liệu bù MÒN dao tiện.

Hình 17.10 minh họa một số mục nhập thông dụng trong bộ ghi bù mòn dao. Các xác lập chỉ số mũi dao và bán kính dao xuất hiện trong cả hai hiển thị và sự hiển thị trên hai màn hình là tự động sau khi nhập giá trị bù. Bán kính mũi dao và chỉ số định hướng đỉnh dao là đặc trưng đối với các bộ điều khiển tiện CNC.

Chỉ số bù và dao

Các dụng cụ cắt trên máy tiện CNC, tương tự trên trung tâm gia công, cũng có các chỉ số. Nói chung, thường chỉ sử dụng một giá trị bù

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tọa độ, nhưng các chỉ số dao là khác nhau. Bạn cần nhớ, chỉ số dao trên máy tiện có *bốn chữ số*, ví dụ T0404:

- Hai chữ số đầu chọn trạm phân độ dao (ổ dao trên hộc chứa dao) *và* số bù hình học. Ở đây không có sự lựa chọn khác. Ví dụ dao trong ổ dao 4, cũng sẽ dùng số bù hình học là 4.
- Hai chữ số cuối là các số đăng ký (ghi) bù mòn dao. Chúng không nhất thiết phải bằng chỉ số dao, nhưng bạn nên làm cho các số này tương hợp với nhau, nếu có thể.

Tùy theo bộ điều khiển và kích cỡ màn hình, sự ghi bù dao có thể có màn hiển thị riêng cho sự bù hình học và bù dao. Các giá trị bù chi tiết (tọa độ làm việc) luôn luôn trong cột bù *Geometry* (hình học).

GÁ LẮP DAO

Trong các minh họa kế tiếp tương tự các minh họa đã nêu trong Chương 15, trình bày công dụng của phương pháp đăng ký G50 (lệnh đăng ký vị trí được dùng trong chương trình). *Bạn hãy so sánh các minh họa đó.*

Xác lập của máy tiện CNC là đồng nhất trong cả hai trường hợp, ngoại trừ phương pháp và mục đích đo vị trí. Mọi minh họa trong các ứng dụng này cũng tương hợp với dữ liệu được nhập vào các màn hình bù hình học dao và bù mòn dao của hệ thống điều khiển.

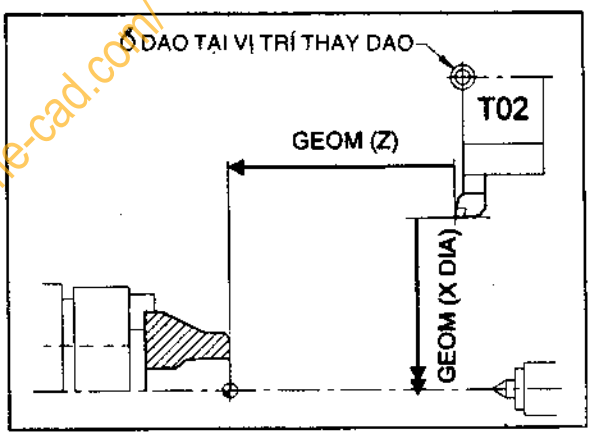
Các giá trị điện hành dọc theo trục X luôn luôn âm, các giá trị dọc theo trục Z thường là âm, cũng có thể có giá trị dương, nhưng điều đó có nghĩa là dao cắt ở phía trên chi tiết và sự thay dao có thể rất nguy hiểm. *Bạn hãy cảnh giác với các tình huống đó!*

Dao cắt gọt theo đường tâm

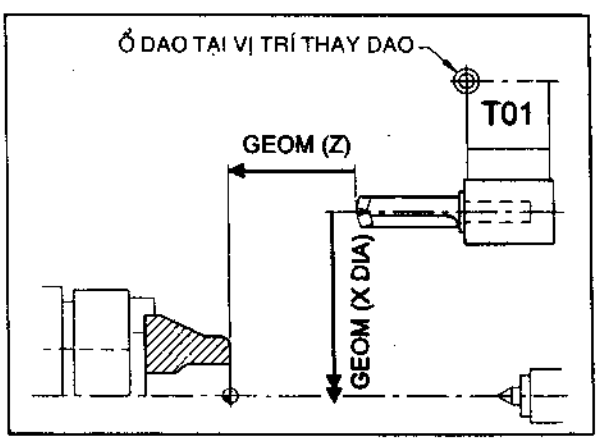
Các dao gia công theo đường tâm trục chính là loại có đỉnh ở đường tâm trong quá trình cắt gọt. Lĩnh vực này bao gồm mọi mũi khoan, chuốt, taro ren, kể cả các dao phay được dùng để làm phẳng đáy lỗ cụt. Tuy nhiên, không phải mọi dao doa đều thuộc nhóm này, do đỉnh dao phay thường không nằm trên đường tâm trục chính trong khi gia công. Các dao theo đường tâm *luôn luôn* được đo từ đường tâm của dao đến đường tâm trục chính dọc theo trục X, và từ đỉnh dao đến zero chương trình dọc theo trục Z (Hình 17.11).

Dao tiện

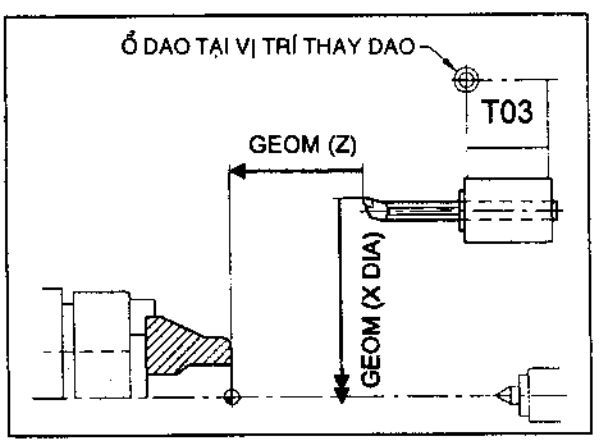
Dao tiện – hoặc *dao tiện ngoài* – được đo từ đỉnh dao tưởng tượng đến zero chương trình, dọc theo trục X (đường kính âm) và theo trục Z, thường cũng là giá trị âm. Bạn cần nhớ nếu mảnh dao hợp kim (tiện hoặc doa) thay đổi bán kính trong *cùng một* ổ dao, giá trị gá lắp cũng thay đổi. Sự thay đổi này có thể rất nhỏ, nhưng cũng đủ gây ra vết xước, do đó bạn cần rất cẩn thận. Đối với nguyên công tiện, bạn cần chú ý bán kính mũi dao có thể thay đổi từ lớn đến nhỏ, ví dụ từ 3/64 (R0.0469) đến 1/32 (R0.0313).



Hình 17.12. *Xác lập bù hình học đối với dao tiện NGOÀI.*



Hình 17.11. *Xác lập bù hình học cho các dao cắt gọt theo ĐƯỜNG TÂM*



Hình 17.13. *Xác lập bù hình học đối với dao gia công mặt TRONG*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Hình 17.12 minh họa xác lập bù hình học đối với dao tiện (ngoài) và Hình 17.13 minh họa xác lập bù hình học đối với dao doa (gia công mặt trong).

Dao doa

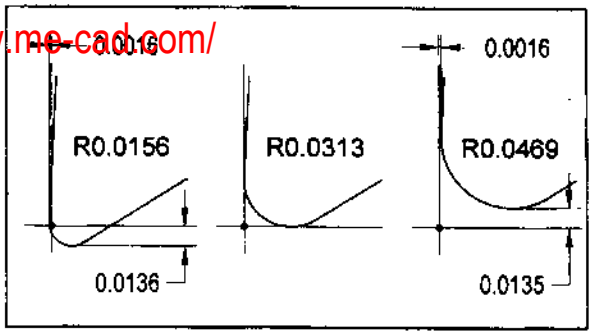
Dao doa – hoặc dao gia công mặt *trong* – luôn luôn được đo từ đỉnh dao tưởng tượng đến zero chương trình dọc theo trục X (thường là đường kính âm) và theo trục Z, cũng là giá trị âm. Trong hầu hết các trường hợp, giá trị X của dao doa thường lớn hơn dao tiện ngoài.

Đối với nguyên công doa, cũng như nguyên công tiện, cần chú ý bán kính mũi dao thay đổi từ lớn đến nhỏ, có thể gây ra phế phẩm.

Điểm lệnh và bù dụng cụ cắt

Vì nhiều lý do, trong khi gia công có thể phải thay mảnh hợp kim cắt, chủ yếu để duy trì chế độ cắt và dung sai kích thước trong phạm vi cho phép. Mảnh dao hợp kim được chế tạo với tiêu chuẩn rất cao, nhưng vẫn có thể có sai lệch kích thước khi mua từ các nguồn khác nhau. Nếu thay mảnh dao hợp kim, cần điều chỉnh giá trị bù mòn dao để bảo đảm độ chính xác gia công.

Cán dao tiếp nhận mảnh hợp kim có cùng kích cỡ và hình dạng nhưng có thể có bán kính mũi dao khác nhau. Bạn cần đặc biệt chú ý khi thay mảnh dao hợp kim khác có bán kính mũi dao lớn hơn hoặc nhỏ hơn. Sự bù mòn dao có thể được điều chỉnh theo cả hai trục với giá trị thích hợp.



Hình 17.14. *Sai số xác lập phát sinh do bán kính mũi dao hợp kim khác nhau lắp trong cùng một cán dao.*

Ví dụ trên Hình 17.14 minh họa xác lập tiêu chuẩn đối với bán kính mũi dao 1/32 (.0313 giữa), sai số xác lập đối với bán kính nhỏ hơn (trái) và lớn hơn (phải). Các kích thước minh họa giá trị sai số đối với mảnh dao cụ thể được nêu trong ví dụ đó.

Khi thay đổi mảnh dao hợp kim, bạn hãy điều chỉnh giá trị bù mòn dao

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

18 BÙ CHIỀU DÀI DAO

Các chương trước đã xem xét hai phương pháp bù cho vị trí thực của *dụng cụ cắt* trong quan hệ với *điểm quy chiếu máy*. Thứ nhất là phương pháp cũ, sử dụng *sự bù vị trí*, thứ hai là phương pháp mới, *hệ tọa độ vị trí (bù chi tiết)*. Trong cả hai trường hợp, chỉ tập trung vào các trục X và Y, chưa chú ý đến trục Z. Mặc dầu trục Z có thể được gộp vào hai phương pháp nêu trên, nhưng kết quả thường có tính thực tiễn không cao. Lý do chính là bản chất của không gian làm việc CNC.

Nói chung, nhà lập trình quyết định sự gá lắp chi tiết trong đồ gá và chọn vị trí thích hợp của zero chương trình XYZ (điểm quy chiếu chi tiết hoặc zero chi tiết). Khi sử dụng bù chi tiết, các trục XY luôn luôn đo từ *điểm quy chiếu máy* đến vị trí *zero chương trình*. Theo định nghĩa chặt chẽ, nguyên tắc này cũng áp dụng cho trục Z. Khác biệt chính là các giá trị đo trên X,Y sẽ *không thay đổi đối với mọi dụng cụ cắt*, bất kể số lượng dao được dùng là một hoặc hàng trăm dao. Điều này hoàn toàn khác với trục Z.

Lý do? *Mỗi dao đều có chiều dài dao riêng.*

NGUYÊN LÝ CHUNG

Bạn có thể phải giải quyết vấn đề chiều dài dụng cụ cắt trong từng chương trình trên trung tâm gia công CNC. Kể từ những ứng dụng CN đầu tiên đã xuất hiện nhiều kỹ thuật lập trình chiều dài dụng cụ cắt. Chúng thuộc một trong hai nhóm cơ bản.

- ❑ Biết chiều dài thực của dụng cụ cắt.
- ❑ Chưa biết chiều dài thực của dụng cụ cắt.

Mỗi nhóm đều đòi hỏi kỹ thuật lập trình riêng. Để hiểu khái niệm chiều dài dao cắt trong lập trình CNC, điều quan trọng là hiểu *chiều dài thực của dao cắt*. Chiều dài này đôi khi được gọi là chiều dài vật lý của dao cắt hoặc đơn giản chỉ là chiều dài dao cắt, có ý nghĩa rất đặc biệt trong lập trình CNC và gá lắp.

Chiều dài thực của dụng cụ cắt

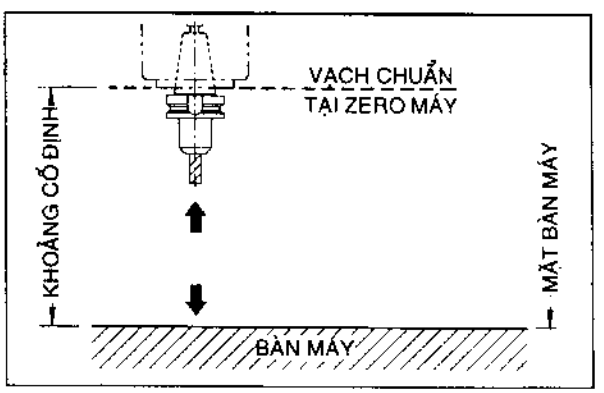
Trước hết, bạn hãy xét dao cắt đơn giản. Với mũi khoan bạn có thể dễ dàng xác định chiều dài bằng dụng cụ đo. Trong ngôn ngữ hàng ngày, mũi khoan 6 inch có chiều dài 6 inch đo từ đầu này đến đầu kia. Trong lập trình CNC, điều đó vẫn đúng nhưng chưa đủ. Mũi khoan hoặc dụng cụ cắt bất kỳ – thường được lắp trong ổ dao và chỉ *một phần* của dao chìa ra ngoài, phần còn lại nằm trong ổ dao. Ổ dao được lắp trong trục chính, bằng hệ thống định vị và kẹp chặt tiêu chuẩn. Các ký hiệu dao, chẳng hạn HSK63, HSK 100, BT40, CAT 50, là các tiêu chuẩn Châu Âu và Hoa Kỳ. Dụng cụ cắt bất kỳ trong nhóm phân loại đều khớp với máy công cụ được thiết kế trong nhóm đó. Đây là một trong tính năng bảo đảm độ chính xác gia công trên máy CNC.

Chiều dài dao đối với lập trình CNC luôn luôn liên hệ chặt chẽ với ổ dao và thiết kế máy công cụ. Do đó, nhà chế tạo thiết lập vị trí chính xác trong trục chính được gọi là *vạch chuẩn*.

Vạch chuẩn

Khi cán dao cùng với dao được lắp vào trục chính của máy CNC, chuôi côn của cán dao sẽ lắp ngược chiều với ống côn trong trục chính và được giữ chặt bằng kẹp rút. Độ chính xác gia công bảo đảm vị trí cố định của cán dao trong trục chính. Vị trí này được dùng để quy chiếu và thường được gọi là *vạch chuẩn*. Đây là đường quy chiếu tưởng tượng được dùng để đo theo trục Z (Hình 18.1).

http://www.me-cad.com/



Hình 18.1. *Hình chiếu trước của trung tâm gia công đứng CNC*

Vạch chuẩn được dùng để đo chính xác chiều dài dụng cụ cắt và chuyển động bất kỳ của dao cắt dọc theo trục Z. Vạch chuẩn do nhà chế tạo máy công cụ xác định và có quan hệ chặt chẽ với một bề mặt chính xác, được gọi là *bàn máy*, thực tế đó là *mặt trên* của bàn máy. Vạch chuẩn là một cạnh của mặt phẳng song song với *mặt trên của bàn máy*.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Mặt trên bàn máy

Mọi trung tâm gia công đều có bàn máy, để lắp đồ gá và chi tiết. Mặt trên của bàn máy được mài chính xác để bảo đảm độ phẳng và độ vuông góc cho chi tiết gá lắp.

Ngoài ra, bàn máy được định vị với khoảng cách xác định tính từ vạch chuẩn. Vị trí của cán dao trong trục chính không thể thay đổi, vị trí của bàn máy (kể cả loại bàn máy dịch chuyển được) cũng không thay đổi trong khi gia công. Bề mặt của bàn máy tạo thành mặt phẳng quy chiếu song song với vạch chuẩn. Sự sắp xếp này cho phép lập trình chính xác chuyển động của dụng cụ cắt dọc theo trục Z.

Có thể định nghĩa bù chiều dài dao cắt như sau:

Bù chiều dài dao cắt là quy trình hiệu chỉnh khoảng cách giữa chiều dài lập trình của dao cắt và chiều dài thực của dao đó.

Lợi ích lớn nhất của bù chiều dài dao trong lập trình CNC là cho phép nhà lập trình thiết kế chương trình hoàn chỉnh, sử dụng số lượng dao cắt theo yêu cầu gia công mà không cần biết chiều dài thực của từng dao.

LỆNH BÙ CHIỀU DÀI DAO

Các hệ thống Fanuc và nhiều bộ điều khiển khác có ba lệnh liên quan đến sự bù chiều dài dao – tất cả đều là mã G.

G43 G44 G49

Cả ba lệnh này đều chỉ áp dụng cho trục Z. Khác với các lệnh bù chi tiết G54 – G59; G43 và G44 luôn luôn kèm theo các đặc tính riêng. Chúng chỉ có thể được sử dụng với chỉ số bù được gán từ địa chỉ H. Tiếp sau địa chỉ H phải có từ 1 đến 3 chữ số, tùy theo số lượng bù khả dụng trong hệ thống.

G43	Bù chiều dài dao dương
G44	Bù chiều dài dao âm
G49	Xóa bù chiều dài dao
H00	Xóa bù chiều dài dao
H..	Chọn số bù chiều dài dao

Nói chung, bù chiều dài dao thường được lập trình trong chế độ tuyệt đối G90. Mục nhập chương trình sẽ là lệnh G43 hoặc G44, tiếp theo là vị trí đích trên trục Z, và chỉ số bù H.

```
N66 G43 Z1.0 H04
```

Đây cũng là block thuận tiện để bổ sung hàm chất làm nguội M08 cho dụng cụ cắt hiện hành:

```
N66 G43 Z1.0 H04 M08
```

Chuyển động dao trong ví dụ nêu trên là 1.0 inch trên zero chi tiết. Hệ điều khiển sẽ tính toán *khoảng cách dịch chuyển*, dựa trên giá trị bù H được người vận hành lưu lại trong khi xác lập chế độ gia công.

Hình 18.2 minh họa màn hình hiển thị bù chiều dài dụng cụ cắt.

TOOL OFFSET (LENGTH)		
No.	GEOMETRY	WEAR
001	-6.7430	0.0000
002	-8.8970	0.0000
003	-7.4700	0.0000
004	0.0000	0.0000
005	0.0000	0.0000
006	0.0000	0.0000
...	...	...

Hình 18.2. *Các giá trị nhập trên màn hình bù chiều dài dụng cụ cắt.*

Bạn cần chú ý, màn hình thực tế có thể khác nhau giữa các bộ điều khiển và một số hệ điều khiển không hiển thị giá trị bù mòn dao. Bù mòn dao (nếu có) chỉ được dùng cho *các điều chỉnh* chiều dài dao theo mục nhập màn hình riêng rẽ.

Lệnh G44 được dùng trong chương trình, có lẽ là lệnh ít được sử dụng nhất trong mọi mã G của Fanuc. Cuối chương này sẽ so sánh lệnh G43 với G44.

Nhiều nhà lập trình CNC có thể không nhận thấy xác lập trục Z trong bù chi tiết (G54 – G59) cũng rất quan trọng đối với bù chiều dài dao. Lý do sẽ được giải thích trong phần trình bày các phương pháp xác lập bù chiều dài dao.

Một số sổ tay hướng dẫn lập trình đề nghị sử dụng các lệnh cũ G45 và G46 để bù chiều dài dao. Tuy điều này đến nay vẫn còn đúng và có thể đã có vài ưu điểm trên bộ điều khiển cũ, nhưng không nên sử dụng chúng. Thứ nhất, các lệnh vị trí ngày càng ít được sử dụng và thứ hai, chúng còn có thể được dùng với các trục X và Y do đó không hoàn toàn biểu thị cho trục Z.

Khoảng cách dịch chuyển theo trục Z

Để diễn dịch cách thức hệ thống CNC sử dụng lệnh bù chiều dài dao, nhà lập trình hoặc người vận hành phải có khả năng tính toán *khoảng cách dịch chuyển* của dụng cụ cắt. Logic của sự bù chiều dài dao khá đơn giản:

- Giá trị bù H sẽ được *cộng* vào vị trí đích Z nếu dùng G43, do G43 được định nghĩa là bù chiều dài dao *dương*.
- Giá trị bù H sẽ được *trừ* khỏi vị trí đích Z nếu dùng G44, do G44 được định nghĩa là bù chiều dài dao *âm*.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Vị trí đích trong cả hai trường hợp đều là giá trị tuyệt đối của trục Z trong chương trình. Nếu biết xác lập trục Z của bù chi tiết (G54 – G59), giá trị H và đích Z, có thể tính chính xác *khoảng cách dịch chuyển*. Hệ điều khiển sẽ sử dụng công thức:

$$Z_d = W_z + Z_t + H$$

Trong đó:

Zd: khoảng cách dịch chuyển theo trục Z (hành trình thực)
W_z: giá trị tọa độ làm việc đối với trục Z.
Z_t: vị trí đích trên trục Z (tọa độ Z)
H: giá trị của số bù H được áp dụng.

➲ **Ví dụ - W_z = 0**

```
G43 Z0.1 H01  .....trong đó:
```

G54 theo trục Z được xác lập là Z0, vị trí đích trên trục Z là .10 và H01 được xác lập là –6.743, khoảng cách dịch chuyển sẽ là:

$$Z_d = 0 + (+.1) + (-6.743)$$
$$= 0 + .1 - 6.743 = -6.643$$

Khoảng cách dịch chuyển hiển thị sẽ là Z-6.643

Để bảo đảm công thức này luôn luôn chính xác, bạn hãy thử thay đổi vài giá trị

➲ **Ví dụ - W_z = .0200**

Trong ví dụ này, chương trình có block

```
G43 Z1.0 H03  ....trong đó:
```

G54 theo trục Z được xác lập là.0200, đích trên trục Z là Z1.0 và giá trị của H03 là –7.47.

$$Z_d = (+.02) + (+1.0) + (-7.47)$$
$$= .02 + 1.0 - 7.47 = -6.45$$

Kết quả là đúng, dao sẽ chuyển động theo trục Z hướng đến chi tiết, và *khoảng cách dịch chuyển* sẽ là Z-6.45

Trong ví dụ cuối, vị trí đích là âm:

➲ **Ví dụ - W_z = .0500**

Block chương trình có tọa độ Z âm

```
G43 Z-0.625 H07  trong đó:
```

G54 theo trục Z được xác lập là.0200, đích trục Z là –0.625 và H07 là –8.28. Tính toán *khoảng cách dịch chuyển* cũng sử dụng công thức nêu trên với các giá trị khác:

$$Z_d = (+.02) + (-.625) + (-8.28)$$
$$= .02 - .625 - 8.28 = -8.885$$

Công thức này vẫn chính xác và có thể dùng để tính toán khoảng cách dịch chuyển bất kỳ theo trục Z.

XÁC LẬP CHIỀU DÀI DAO

Chiều dài của dao được dùng để gia công (gồm dao và cán dao), có thể được xác lập trực tiếp trên máy CNC hoặc cách xa máy. Các tùy chọn xác lập này thường được gọi là xác lập chiều dài dao *trên máy* hoặc *cách xa máy*. Mỗi tùy chọn đều có ưu điểm và nhược điểm riêng. Cả hai đều có quan hệ với vạch chuẩn, khi áp dụng cho chiều dài dao hoặc phần chìa ra ngoài ổ dao. Bạn hãy đánh giá từng tùy chọn đó, so sánh các ưu điểm và nhược điểm của chúng. Việc lựa chọn tùy chọn thích hợp còn phụ thuộc vào nhiều yếu tố.

Cả hai tùy chọn đều đòi hỏi sự tham gia của hai người, hoặc hai kỹ năng chuyên nghiệp, nhà lập trình và người vận hành CNC. Nhà lập trình xác định tất cả các dao được chọn theo chỉ số của chúng (địa chỉ T) và gán các số bù chiều dài dao, ví dụ G43 hoặc G44 với địa chỉ H vào bộ nhớ của hệ thống CNC.

Xác lập chiều dài dao trên máy

Về kỹ thuật, xác lập chiều dài dao *trên máy* là công việc của người vận hành máy. Nói chung, người vận hành lắp dao vào trục chính và đo khoảng hành trình (chạy dao) *từ zero máy* đến *zero chi tiết*. Công việc này tốn nhiều thời gian nếu lắp nhiều dao. Có một số phương pháp gá lắp dao cho phép giảm đi thời gian xác lập chiều dài dao *trên máy*, chẳng hạn phương pháp *dao chính*, sẽ được trình bày trong phần kế tiếp. Ưu điểm của phương pháp này là không đòi hỏi thiết bị bổ sung và không yêu cầu kỹ năng của người vận hành.

Xác lập chiều dài dao cách xa máy

Về kỹ thuật, xác lập chiều dài dao *cách xa máy* là công việc của thợ dụng cụ lành nghề hoặc người vận hành CNC. Do xác lập được thực hiện ở cách xa máy, cần có thiết bị đặc biệt, làm tăng thêm chi phí sản xuất. Thiết bị này có thể là đồ gá đơn giản với chuẩn chiều cao, hoặc phức tạp với hiển thị số.

Đăng ký giá trị bù chiều dài dao

Bất kể phương pháp xác lập chiều dài dao được sử dụng, luôn luôn có giá trị đo biểu thị chiều dài của dụng cụ cắt được chọn. Giá trị chiều dài dao phải được đưa vào chương trình trước khi tiến hành gia công. Người vận hành phải đăng ký giá trị đo vào hệ thống, với tiêu đề thích hợp trên bảng điều khiển.

Hệ điều khiển chứa mục ghi đặc biệt chuyên dùng cho giá trị bù chiều dài dao,

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

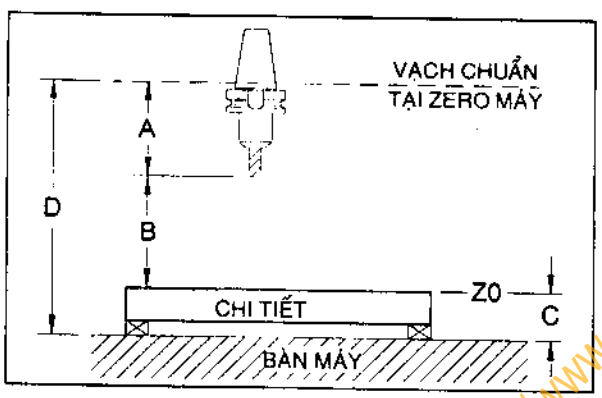
Nhiều tài liệu hay có tại : http://www.me-cad.com/

thường có tiêu đề *Tool length setting, tool length offset, tool length compensation*, hoặc *offset*. Bất kể tên tiêu đề, quy trình xác lập phải bảo đảm chiều dài đo được nhập vào bộ điều khiển để chương trình có thể sử dụng. Chiều dài đo luôn luôn trong các giới hạn hành trình của trục Z nhưng vẫn cho phép đủ khoảng trống để thay dao và thay chi tiết.

Để hiểu về bù chiều dài dao, bạn cần biết chuyển động trục Z và dạng hình học của máy. Trên các trung tâm gia công ngang và đứng, bạn hãy quan sát mặt phẳng XY, là mặt trên của chi tiết. Các nguyên lý là như nhau, nhưng ở đây sẽ tập trung vào trung tâm gia công đứng.

CÁC QUAN HỆ TRÊN TRỤC Z

Để hiểu các nguyên lý chung về bù chiều dài dao, bạn hãy xem xét minh họa về gá lắp trên trung tâm gia công đứng (Hình 18.3).



Hình 18.3. *Các quan hệ trên trục Z của máy, dao cắt, mặt bàn máy và chiều dao chi tiết.*

Hình này biểu thị sự gá lắp trên trung tâm gia công đứng CNC, chiều từ phía trước máy, từ phía người vận hành. Cột trục chính đặt tại vị trí zero máy. Đây là vị trí chuyển đổi giới hạn đối với hành trình trục Z dương và cần thiết để thay dao tự động trên hầu hết mọi trung tâm gia công. Cả bốn kích thước trên Hình 18.3 đều có thể xác định một cách dễ dàng. Chúng luôn luôn được coi là kích thước *đã biết* hoặc kích thước *cho trước* và đều là *cơ sở* để xác lập máy chính xác.

- Khoảng cách giữa vạch chuẩn dao và đỉnh cắt của dao
  ... ***kích thước A trên hình.***
- Khoảng cách giữa đỉnh cắt của dao và Z0 (zero chương trình của chi tiết)
  ... ***kích thước B trên hình.***
- Chiều cao của chi tiết (khoảng cách giữa mặt bàn máy và Z0 của chi tiết)
  ... ***kích thước C trên hình.***
- Tổng của ba kích thước nêu trên (khoảng cách giữa mặt bàn máy và vạch chuẩn dao)
  ... ***kích thước D trên hình.***

Trên Hình 18.3, kích thước D luôn luôn được biết, do đây là khoảng cách được các nhà chế tạo máy CNC xác định. Có thể chưa biết kích thước C (chiều cao chi tiết với các khoảng hở), nhưng có thể dễ dàng xác định khi chuẩn bị gá lắp chi tiết.

Còn lại là kích thước A – khoảng cách giữa vạch chuẩn dao và đỉnh cắt của dao, không có phương pháp xác định kích thước này trừ phương pháp đo thực tế. Trong những ngày đầu của CNC, chiều dài A được cho trước và được đưa vào chương trình. Do sự bất tiện khi tìm kích thước này, nhiều phương pháp mới được dùng trong các máy CNC hiện đại.

Ngày nay, ba phương pháp được xem xét trong lập trình chiều dài dao cắt.

- Phương pháp xác lập trước chiều dài dao
  ... dựa trên thiết bị gá lắp dao bên ngoài
- Phương pháp đo
  ... dựa trên sự đo đạc chiều dài dao trên máy
- Phương pháp dao chính là phương pháp hiệu quả nhất
  ... dựa trên quan hệ với chiều dài của dao dài nhất.

Từng phương pháp đều có các ưu điểm riêng. Nhà lập trình CNC xem xét các ưu điểm đó và chọn phương pháp thích hợp. Các ứng dụng của chúng không liên quan trực tiếp với quá trình lập trình, chỉ thuần túy là *gá lắp* dao trên máy, nhưng nhà lập trình phải hiểu rõ. Bất kể phương pháp được chọn, bạn phải có sự quy chiếu đến xác lập được chọn trong chương trình, thường ở dạng chú thích hoặc thông báo.

Xác lập trước chiều dài dao

Một số người dùng ưa thích *xác lập trước* chiều dài dao cắt *bên ngoài* máy, thay vì trong khi xác lập máy. Đây là phương pháp xác lập chiều dài dao cắt cổ điển. Phương pháp này có một số ưu điểm, đặc biệt là loại bỏ thời gian chết trong khi gá lắp. Ưu điểm thứ hai là trên trung tâm gia công ngang, zero chương trình thường được xác lập trước ở tâm bàn máy xoay hoặc bàn phân độ. Nhược điểm chính của phương pháp này là sự xác lập trước chiều dài dao cách xa máy đòi hỏi thiết bị bên ngoài, được gọi là *đồ gá dao*, làm tăng thêm chi phí.

Sử dụng đồ gá dao, mọi dao cắt đều được lắp vào thiết bị bên ngoài, trong khi máy CNC đang gia công. Không cần đo trên máy khi thay đổi loại chi tiết gia công. Công việc của người

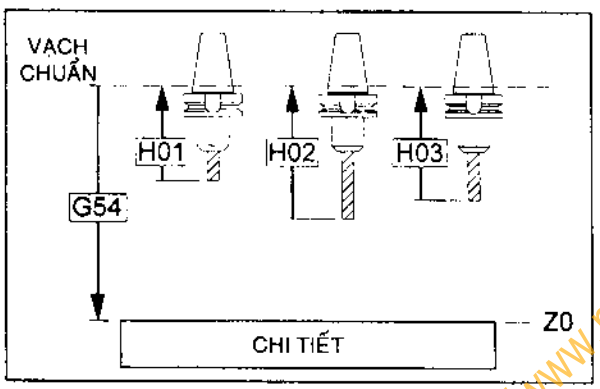
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

vận hành là nhập các giá trị đo vào bộ đăng ký bù. Thậm chí phần này trong quy trình gá lắp có thể được thực hiện thông qua chương trình bằng cách sử dụng lệnh G10 tùy chọn (nếu có).

Phương pháp này đòi hỏi có thợ chuyên môn cao chịu trách nhiệm xác lập trước các dao cắt. Nhiều xưởng cơ khí nhỏ và trung bình sử dụng trung tâm gia công đứng khó có thể gá lắp chi tiết, chủ yếu sử dụng phương pháp đo.

Trong quá trình đo chiều dài dao, khoảng cách từ đỉnh cắt của dao đến vạch chuẩn được xác định một cách chính xác (Hình 18.4). Các dao cắt xác lập trước vươn đến máy được lắp sẵn trong giá dao, được xác định bằng chỉ số dao và danh mục chiều dài dao (xác lập trước) đã đo. Công việc của người vận hành CNC là lắp các dao cần thiết vào hộc dao và đăng ký chiều dài từng dao vào bộ đăng ký bù, sử dụng chỉ số bù thích hợp.



Hình 18.4. *Xác lập trước chiều dài dao ở cách xa máy (phương pháp dùng đồ gá dao), cần sử dụng sự bù chi tiết (G45 - G59).*

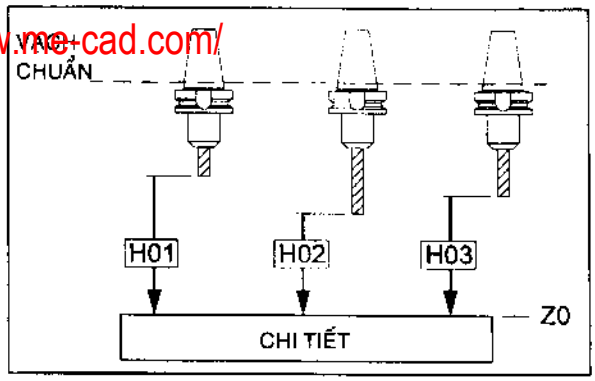
Các kích thước được xác định trước có giá trị *dương*, được đo từ điểm quy chiếu dao đến vạch chuẩn của ổ dao. Vạch chuẩn của máy được mô phỏng trong đồ gá dao để bảo đảm tính tương hợp. Mỗi kích thước sẽ được nhập theo giá trị bù H trong màn hình bù chiều dài dao. Ví dụ, chiều dài dao được xác lập trước theo giá trị 8.5 inch, với chỉ số bù cho dao này là H05. Trên màn hình bù, dưới số 05, người vận hành nhập chiều dài đo là 8.500:

```
04 ...
05 8.500
06 ...
```

Chiều dài dao theo phương pháp đo

Chiều dài dao theo phương pháp đo là rất phổ biến, dù tốn thêm thời gian khi gá lắp. Trên Hình 18.5, mỗi dao được gán một chỉ số H được gọi là *chỉ số bù chiều dài dao*.

Chỉ số này được lập trình theo địa chỉ H với



Hình 18.5. *Phương pháp đo để xác lập bù chiều dài dao.*

chính số đó. Chỉ số H thường tương ứng với chỉ số dao để dễ sử dụng. Quy trình xác lập số đo khoảng cách hành trình dao từ *vị trí zero máy đến vị trí zero chương trình (Z0)*. Khoảng cách này luôn luôn âm và được nhập vào các số bù H tương ứng với menu bù chiều dài dao của hệ điều khiển. Điều quan trọng là các xác lập trục Z đối với bù chi tiết bất kỳ G54 – G59 và sự bù *chung* thường được xác lập theo Z0.0000.

Sử dụng chiều dài dao chính

Sử dụng phương pháp đo để xác lập bù chiều dài dao có thể rút ngắn thời gian rõ rệt bằng cách áp dụng phương pháp đặc biệt được gọi là *dao chính*, thường là dao dài nhất. Đây có thể là dao thực hoặc một thanh dài có đỉnh tròn lắp vào ổ dao. Trong phạm vi hành trình trục Z, "dao" mới này phải chìa ra ngoài xa hơn mọi dao khác.

Các lệnh bù G54 đến G59 và bù gia công *mặt ngoài* thường có giá trị Z xác lập theo 0.0, khi sử dụng phương pháp đo. Xác lập này sẽ thay đổi trong phương pháp chiều dài dao chính. Đo chiều dài chính là rất hiệu quả và đòi hỏi quy trình xác lập riêng, với các bước có thể cần vài chỉnh sửa theo điều kiện thực tế:

1. Lắp dao chính vào trục chính
2. Chuẩn zero cho trục Z và bảo đảm số đo trên màn hình *tương đối* (số gia) là Z0.000 hoặc Z0.000.
3. Đo chiều dài dao chính, sử dụng phương pháp đo đã nêu. Sau khi chạm vào bề mặt đo, *cần giữ nguyên dao ở vị trí đó*.
4. Thay vì đăng ký giá trị đo theo chỉ số bù chiều dài dao, bạn hãy đăng ký theo giá trị bù chi tiết chung hoặc một trong các mã G54 – G59 với xác lập Z! *Đây sẽ là giá trị âm.*
5. Trong khi dao chính chạm vào bề mặt đo, bạn hãy xác lập giá trị trục Z tương đối là *zero*!
6. Đo từng dao còn lại, sử dụng phương pháp đo chiều dài. Số đo sẽ từ đỉnh dao chính thay vì từ zero máy.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

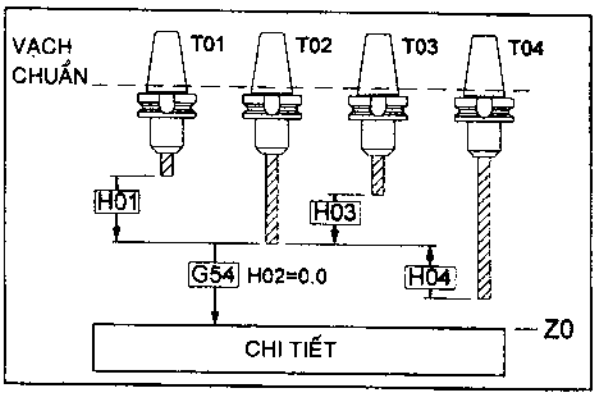
Nhiều tài liệu hay có tại : http://www.me-cad.com/

7. Nhập các giá trị đo vào chỉ số bù H trên màn hình bù chiều dài dao. Đây luôn luôn là giá trị âm đối với mọi dao ngắn hơn dao chính.

➲ **Chú ý:**

Dao chính *không nhất thiết* phải là dao dài nhất. Khái niệm dao dài nhất chỉ nhằm bảo đảm tính an toàn, có nghĩa là mọi dao khác đều ngắn hơn.

Nếu chọn dao khác làm dao chính, quy trình về logic là không thay đổi, ngoại trừ các mục nhập bù H sẽ là *dương* đối với dao cắt bất kỳ *dài hơn* dao chính và là *âm* đối với dao *ngắn hơn* dao chính. Trong trường hợp đặc biệt, dao được đo từ chiều dài đúng bằng dao chính, mục nhập bù cho dao đó sẽ là zero (Hình 18.6).



Hình 18.6. *Bù chiều dài dao sử dụng phương pháp chiều dài dao chính. T02 là dao chính, với xác lập H02 = 0.0*

Sau khi chiều dài dao chính được xác lập và được đăng ký theo trục Z bù chi tiết, bạn hãy nhập khoảng cách từ đỉnh dao mới đến đỉnh dao dao chính, và đăng ký vào chỉ số bù H tương ứng. Nếu dao dài nhất là dao thực, giá trị bù H của dao đó luôn luôn là 0.0.

Ưu điểm lớn nhất của phương pháp này là rút ngắn thời gian xác lập. Nếu một số dao được dùng cho nhiều nhóm chi tiết, chỉ cần xác định lại chiều dài dao chính đối với chiều cao của chi tiết mới bất kỳ trong khi mọi dao còn lại đều không thay đổi. Chúng chỉ liên quan với dao chính.

Sự khác biệt giữa G43 và G44

Từ đầu chương này đã đề cập Fanuc và các hệ thống tương tự có *hai* lệnh chuẩn bị, kích hoạt sự bù chiều dài dao, G43 và G44. Hầu hết các nhà lập trình đều chỉ dùng lệnh G43 trong chương trình và có thể gặp khó khăn khi điển dịch ý nghĩa của lệnh G44, do họ chưa từng dùng lệnh đó. Thực tế có lý do để không sử dụng lệnh G44. Nhà lập trình thường muốn biết *phương pháp* và *thời điểm* sử dụng lệnh này hoặc lệnh kia.

Trước hết, bạn hãy xem xét các định nghĩa trong các tài liệu CNC và các bảng đặc tính kỹ thuật của nhà chế tạo. Nói chung, bạn sẽ gặp các định nghĩa dưới đây, tất cả đều được coi là đúng:

G43 Bù cộng
G44 Bù trừ

G43 Bù chiều dài dao dương
G44 Bù chiều dài dao âm

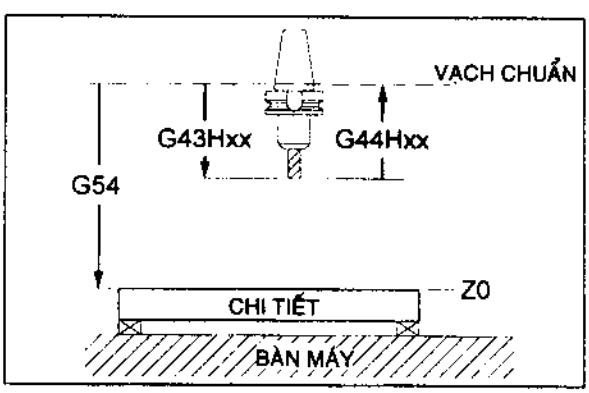
G43 Chiều cộng
G44 Chiều trừ

Các định nghĩa này chỉ đúng nếu xét trong bối cảnh cụ thể. Bối cảnh đó không hoàn toàn rõ ràng. *Cộng đến đâu? Dương của cái gì?* Để tìm bối cảnh cụ thể, bạn hãy suy nghĩ về công dụng của bù chiều dài dao trên máy CNC. *Mục đích của bù chiều dài dao là gì?*

Mục đích chính và quan trọng nhất của bù chiều dài dao là cho phép chương trình CNC được viết ở nơi *cách xa* máy, *cách xa* dụng cụ cắt và đồ gá, và không cần biết chiều dài dao thực trong khi lập trình.

Quá trình này có hai phần – một trong chương trình, và một ở trên máy. Trong chương trình, cần có lệnh G43 hoặc G44, cùng với chỉ số H thích hợp, do nhà lập trình thực hiện. Tại máy, bù chiều dài dao có thể được xác lập trên máy hoặc cách xa máy. Chiều dài dao được đo và giá trị đo được nhập vào bộ điều khiển, đây là công việc của người vận hành. Nếu *đo đạc tại máy* có thể có các biến thể, người lập trình chỉ có lựa chọn là hai mã G.

Hình 18.7 minh họa một trong hai phương pháp xác lập lệnh bù chiều dài dao – cần sử dụng cả lệnh G54 hoặc lệnh bù chi tiết khác.

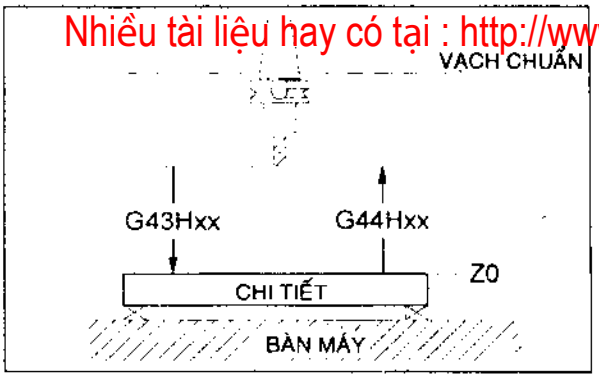


Hình 18.7. *Phương pháp bù chiều dài dao ít được sử dụng. Cần phải xác lập cả bù chi tiết (thường là G54)*

Hình 18.8 minh họa phương pháp thứ hai. Trong trường hợp này, tất cả các lệnh bù chi tiết G54 đến G59 thường có giá trị Z là 0.0.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 18.8. *Phương pháp bù chiều dài dao thông dụng. Không cần dùng xác lập bù chi tiết, lệnh G43 thường được sử dụng.*

Trong các trường hợp đó, chương trình được viết hoàn toàn như nhau (chỉ phương pháp xác lập là khác nhau, phương pháp lập trình *không* thay đổi). Chương trình sẽ chứa lệnh bù chiều dài dao (G43 hoặc G44), tiếp theo là vị trí đích dọc theo trục Z và chỉ số bù H.

```
G43 Z1.0 H06   hoặc   G44 Z1.0 H06
```

Hệ điều khiển không thể đưa ra các điều khiển thích hợp, nếu giá trị đo đối với H06 chưa được lưu trong bộ đăng ký bù. Ví dụ, nếu H06 được đo là 7.6385, sẽ được nhập theo giá trị *âm*, nếu dùng G43, và là giá trị *dương*, nếu dùng G44 (chuyển động chạy dao là như nhau):

```
G43 Z1.0 H06  ....H06 = -7.6385
G44 Z1.0 H06  ....H06 = +7.6385
```

Sự khác biệt giữa G43 và G44 chỉ là dấu âm hoặc dương. Các lệnh này báo cho hệ điều khiển *phương pháp* tính chuyển động thực trên trục Z. Sử dụng G43, giá trị bù H sẽ được *cộng* (+) trong tính toán. Với G44, giá trị bù H sẽ được *trừ* (-). Hành trình thực theo trục Z sẽ là:

```
G43: Z + H06 = (1.0) + (-7.6385) = - 6.6385
G44: Z + H06 = (1.0) - (+7.6385) = - 6.6385
```

Phương pháp đo chiều dài dao được thực hiện *trên* máy sẽ có kết quả bù với các giá trị *âm*. Quy trình xác lập có thể tự động nhập tất cả các giá trị đo vào bộ ghi bù, theo số *âm*. Đây là lý do để G43 là lệnh lập trình bù chiều dài dao. Trong công việc hàng ngày, lệnh G44 ít có tính thực tiễn.

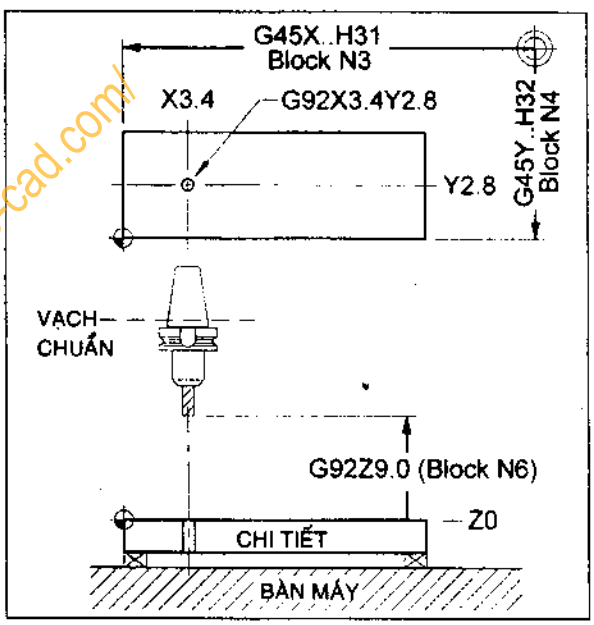
ĐỊNH DẠNG LẬP TRÌNH

Định dạng lập trình bù chiều dài dao tương đối đơn giản và đã được minh họa vài lần. Trong các ví dụ kế tiếp sẽ đưa ra vài ứng dụng tổng quát. Ví dụ thứ nhất sẽ nêu ra phương pháp lập trình khi không có bù chiều dài dao. Kiến thức về sự phát triển bù chiều dài dao qua nhiều năm sẽ giúp bạn áp dụng chúng trong lập trình dễ dàng hơn. Ví dụ thứ hai sẽ so sánh các phương pháp lập trình sử dụng lệnh G92 đã cũ và sử dụng các lệnh G54 - G59. Ví dụ cuối cùng minh họa phương pháp dùng G54 - G59 để viết chương trình đơn giản với ba dao cắt, hiện nay tương đối thông dụng.

Chưa có bù chiều dài dao

Trong thời kỳ đầu của lập trình, chưa có bù chiều dài dao và bù chi tiết. Lệnh đăng ký vị trí G92 là mã G duy nhất được dùng để xác lập vị trí dao hiện hành. Nhà lập trình cần biết *tất cả* và *từng* kích thước do nhà chế tạo chuyên biệt, *tất cả* và *từng* kích thước của chi tiết đang được gá lắp, đặc biệt là khoảng cách từ Z0 đến đỉnh dụng cụ cắt.

Chương trình này đòi hỏi lệnh bù vị trí G45 hoặc G46 theo các trục X và Y, và lệnh đăng ký vị trí G92 theo các trục XYZ. Mỗi chi tiết phải bắt đầu ở zero máy (Hình 18.9).



Hình 18.9. *Xác lập chiều dài dao không có sự bù chiều dài dao – chương trình O1801.*

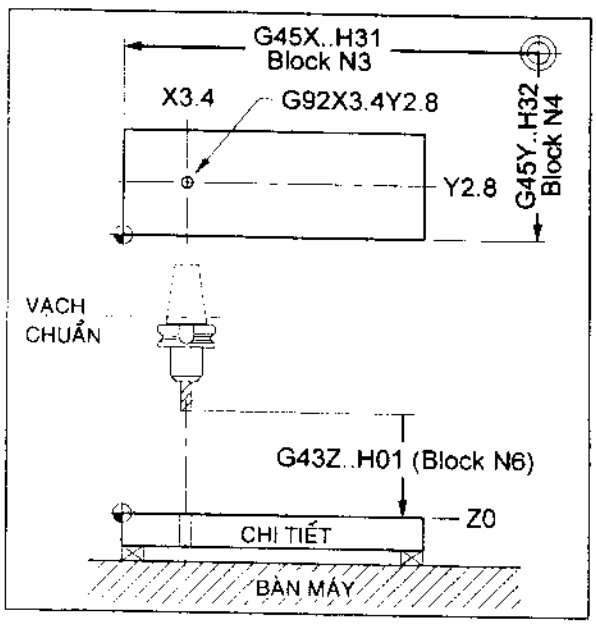
```
O1801
N1 G20                          (INCH MODE SELECTED)
N2 G92 X0 Y0 Z0              (MACHINE ZERO POSITION)
N3 G90 G00 G45 X3.4 H31.        (X POSITION COMP)
N4 G45 Y2.8 H32                 (Y POSITION COMP)
N5 G92 X3.4 Y2.8             (TOOL POS REGISTER XY)
N6 G92 Z9.0                   (TOOL POS REGISTER Z)
N7 S850 M03                     (SPINDLE COMMANDS)
N8 G01 Z0.1 F15.0 M08        (Z APPROACH MOTION)
N9 Z-0.89 F7.0                (Z CUTTING MOTION)
N10 G00 Z0.1 M09              (Z RAPID RETRACT)
N11 Z9.0                   (MACHINE ZERO RETURN Z)
N12 X-2.0 Y10.0               (CLEAR POSITION XY)
N13 M30                         (END OF PROGRAM)
%
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Bù chiều dài dao và G92

Khi bù chiều dài dao trở nên khả dụng, lập trình cũng trở nên dễ dàng hơn. Bù vị trí G45/G46 vẫn được sử dụng, và G92 được xác lập cho cả trục X và trục Y. Tuy nhiên, xác lập G92 đối với trục Z được thay bằng lệnh G43 hoặc G44, với chỉ số bù H (Hình 18.10).



Hình 18.10. Xác lập chiều dài dao với G43 (Z) và G92 (XY) – chương trình O1802

Ngày nay, phương pháp kết hợp bù vị trí G45/G46 và bù chiều dài dao G43/G44 được coi là đã lạc hậu. Chỉ còn G43H.. được sử dụng trong lập trình hiện đại với vị trí đích.

Trong chương trình đã cải tiến, bù chiều dài G43 được áp dụng cho lệnh *chuyển động thứ nhất* của trục Z.

```
O1802
N1 G20                          (INCH MODE SELECTED)
N2 G92 X0 Y0 Z0              (MACHINE ZERO POSITION)
N3 G90 G00 G45 X3.4 H31         (X POSITION COMP)
N4 G45 Y2.8 H32                 (Y POSITION COMP)
N5 G92 X3.4 Y2.8          (TOOL POSITION REGISTER)
N6 G43 Z1.0 H01              (TOOL LENGTH COMP Z)
N7 S850 M03                   (SPINDLE COMMANDS)
N8 G01 Z0.1 F15.0 M08      (Z APPROACH MOTION)
N9 Z-0.89 F7.0               (Z CUTTING MOTION)
N10 G00 Z0.1 M09              (Z RAPID RETRACT)
N11 G28 X3.4 Y2.8 Z1.0 (MACHINE ZERO RETURN)
N12 G49 D00 H00          (OFFSETS CANCELLATION)
N13 M30                         (END OF PROGRAM)
%
```

Khi chương trình được triển khai sử dụng G92, các block N6 và N7, để thuận tiện, có thể kết hợp với nhau:

```
N6 G43 Z1.0 S850 M03 H01
N7 ...
```

Phương pháp này không có tác dụng đối với bù chiều dài dao, chỉ vào thời điểm tại đó trục chính bắt đầu quay. Bù vị trí và bù chiều dài dao *không thể lập trình* trong cùng một block.

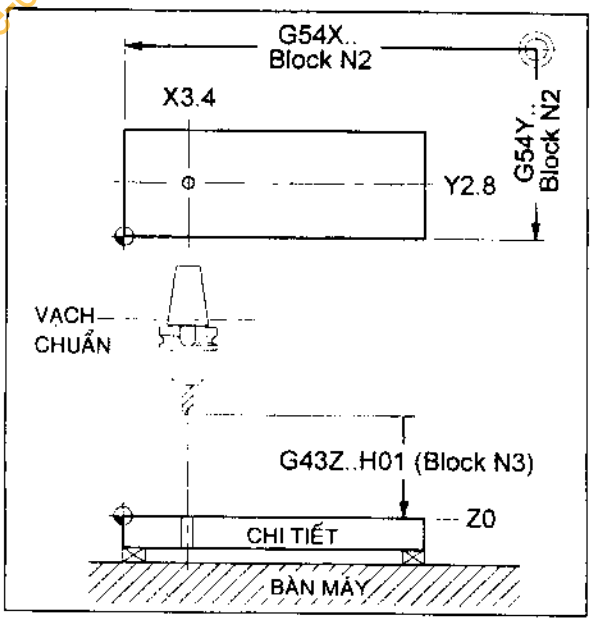
Bạn hãy chú ý, bù vị trí vẫn có hiệu lực trong ví dụ này, do không có bù tọa độ làm việc từ một trong các lệnh G54 – G59

Bù chiều dài dao và G54 - G59

Kỹ thuật lập trình hiện đại có nhiều lệnh và hàm, G54 – G59 là một trong các lệnh đó. Lệnh G92 đã được thay bằng hệ thống bù chi tiết G54 – G59 và các lệnh khác. Nói chung, G92 không được dùng trong chương trình có sự lựa chọn bù chi tiết G54 đến G59.

Dưới đây là chương trình ví dụ sử dụng bù chiều dài dao trong môi trường bù chi tiết G54 – G59

```
O1803
N1 G20                          (INCH MODE SELECTED)
N2 G90 G00 G54 X3.4 Y2.8 (XY TARGET LOCATION)
N3 G43 Z1.0 H01              (TOOL LENGTH COMP Z)
N4 S850 M03                   (SPLNDLE COMMANDS)
N5 G01 Z0.1 F15.0 M08      (Z APPROACH MOTION)
N6 Z-0.89 F7.0               (Z CUTTING MOTION)
N7 G00 Z0.1 M09               (Z RAPID RETRACT)
N8 G28 X3.4 Y2.8 Z1.0   (MACHINE ZERO RETURN)
N9 G49 D00 H00          (OFFSETS CANCELLATION)
N10 M30                         (END OF PROGRAM)
%
```



Hình 18.11. Xác lập chiều dài dao với G43 (Z) và G54 – G59 (XY) – chương trình O1803

Trong ví dụ này (Hình 18.11), sử dụng bù chi tiết G54 đến G59, các block N2, N3, và N4 có thể kết hợp với nhau mà không gây ra vấn đề, có lẽ chỉ làm tăng tốc độ xử lý

```
N2 G90 G00 G54 G43 X3.4 Y2.8 Z1.0 S850 M3 H01
N3 ...
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lệnh G54 sẽ tác động đến tất cả các trục G43 với H01 chỉ tác động đến trục Z. Bảo phải dịch chuyển đến nơi trống.

Bù chiều dài dao và nhiều dao

Hầu hết các chương trình CNC đều có nhiều dao. Ví dụ kế tiếp (độc lập với các bản vẽ nêu trên) minh họa phương pháp phổ biến về nhập bù chiều dài dao cho ba dụng cụ cắt.

Ba lỗ cần được khoan điểm, khoan và taro ren. Vào thời điểm này, bản vẽ và chế độ gia công là không quan trọng, chỉ tập trung vào ứng dụng chiều dài dao G43. Điều quan trọng là cấu trúc chương trình - bạn hãy chú ý, *không có sự thay đổi* trong cấu trúc chương trình của dao bất kỳ, chỉ có giá trị lập trình thay đổi.

```
O1804
N1 G20
N2 G17 G40 G80 T01
N3 M06
N4 G90 G00 G54 X1.0 Y1.5 S1800 M03 T02
N5 G43 Z0.5 H01 M08     (TOOL LG OFFSET FOR T01)
N6 G99 G82 R0.1 Z-0.145 P200 F5.0
N7 X2.0 Y2.5
N8 X3.0 Y1.5
N9 G80 Z0.5 M09
N10 G28 Z0.5 M05
N11 M01

N12 T02
N13 M06
N14 G90 G00 G54 X3.0 Y1.5 S1600 M03 T03
N15 G43 Z0.5 H02 M08 (TOOL LG OFFSET FOR T02)
N16 G99 G81 R0.1 Z-0.89 F7.0
N17 X2.0 Y2.5
N18 X1.0 Y1.5
N19 G80 Z0.5 M09
N20 G28 Z0.5 M05
N21 M01

N22 T03
N23 M06
N24 G90 G00 G54 X1.0 Y1.5 S740 M03 T01
N25 G43 Z1.0 H03 M08 (TOOL LG OFFSET FOR T03)
N26 G99 G84 R0.5 Z-1.0 F37.0
N27 X2.0 Y2.5
N28 X3.0 Y1.5
N29 G80 Z1.0 M09
N30 G28 Z1.0 M05
N31 M30
%
```

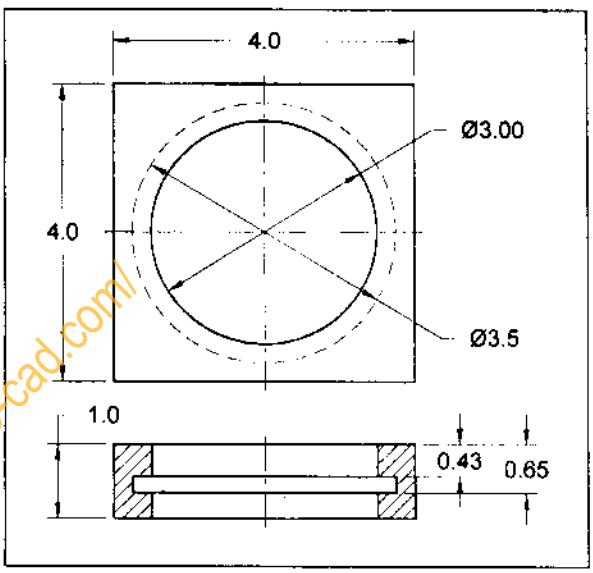
Đây là ví dụ thực tiễn về sử dụng bù chiều dài dao G43 trong chương trình CNC. Tóm lại, bù chiều dài dao G43 đòi hỏi vị trí đích trên trục Z và địa chỉ H cho từng dao. Giá trị bù thực tế được xác lập ở bộ điều khiển trong khi gá lắp. Hai hoặc nhiều giá trị bù chiều dài có thể được dùng cho một dao, nếu cần thiết, đây là vấn đề mới, sẽ được trình bày ở phần sau trong chương này.

Ngoài ra, bạn hãy lưu ý không có chế độ xóa bù chiều dài dao. Điều này sẽ được giải thích ở cuối chương.

THAY ĐỔI BÙ CHIỀU DÀI DAO

Hầu hết các công việc lập trình chỉ yêu cầu một lệnh bù chiều dài dao cho từng dao. Dựa trên nguyên tắc này, bạn có thể ký hiệu *Tool 1* (T01) với bù chiều dài H01, *Tool 2* (T02) với H02. Tuy nhiên, trong một số trường hợp, bù chiều dài dao đối với *cùng một dao* có thể thay đổi. Trong các ứng dụng đó, sẽ có *hai* hoặc *nhiều* giá trị bù chiều dài dao cho một dao.

Ví dụ về thay đổi bù chiều dài dao có thể là chi tiết bất kỳ sử dụng hai hoặc nhiều quy chiếu bản vẽ dọc theo trục Z. Hình 18.12 minh họa khái niệm này với rãnh được lập kích thước theo vị trí chiều sâu đối với đỉnh và đáy (chiều rộng rãnh là .220).



Hình 18.12. *Ví dụ về lập trình nhiều giá trị bù chiều dài dao cho một dao - chương trình O1805*

Dựa trên minh họa này, trước hết cần chọn phương pháp gia công (gia công thô lỗ Ø3.000). Dao phay rãnh rộng .125 sẽ là lựa chọn tốt để cắt biên dạng tròn, sử dụng phương pháp phay đường tròn (Chương 28). Có thể rút ngắn chương trình bằng cách dùng phương pháp chương trình con (Chương 38). Do chiều rộng rãnh .220 lớn hơn dao phay, trong trường hợp này cần cắt hai lần. Đối với lần cắt thứ nhất, dao được định vị theo chiều sâu Z-0.65 (như trên bản vẽ) và cắt ở đáy rãnh. *Cạnh đáy* của dao sẽ đạt đến chiều sâu Z-0.65.

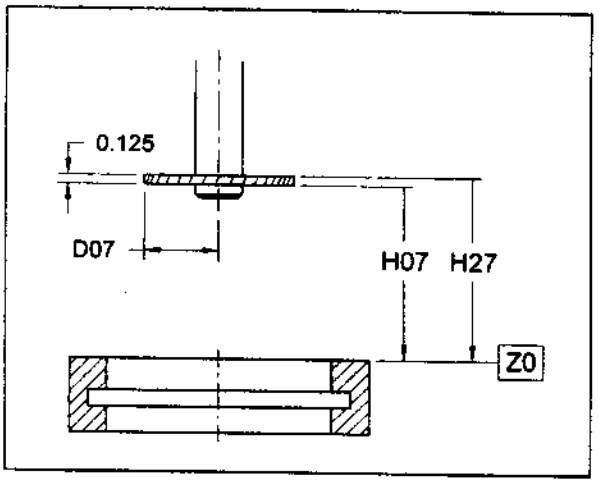
Đối với lần cắt thứ hai, *cạnh trên* của dao phay rãnh được dùng để cắt biên dạng rãnh thứ hai (thực chất là làm rộng rãnh thứ nhất) ở chiều sâu Z-0.43 (như trên bản vẽ).

Bạn hãy lưu ý cạnh *đáy* và cạnh *trên* của dao phay rãnh. Cạnh nào là chuẩn lập trình đối chiều dài dao? Cạnh đáy hay cạnh trên?

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Hình 18.13 minh họa *hai* vị trí quy chiếu được dùng cho một dao. Vì thế chương trình yêu cầu *hai* giá trị bù chiều dài dao, trong minh họa đó là H07 và H27. D07 là bù bán kính dao cắt và .125 là chiều rộng rãnh phay.



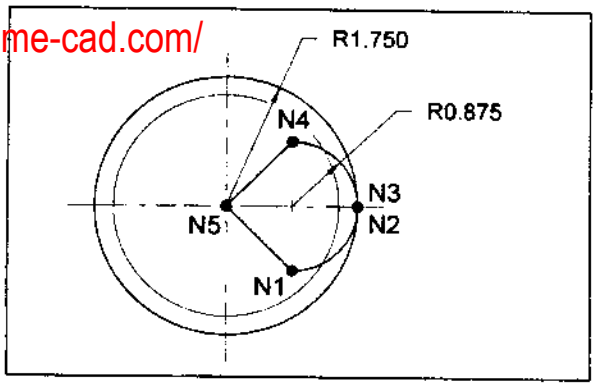
Hình 18.13. *Xác lập hai giá trị bù chiều dài cho một dao. Hiệu giữa hai giá trị bù H07 và H27 là chiều rộng rãnh (.125)*

Có thể sử dụng các phương pháp lập trình khác, ví dụ, tính toán hiệu số bằng tay, nhưng phương pháp sử dụng nhiều giá trị bù chiều dài dao rất hữu dụng khi gia công, cho phép tinh chỉnh chiều rộng rãnh. Điều này được minh họa trong chương trình O1805.

```
O1805
(TWO TOOL LENGTH OFFSETS FOR ONE TOOL)
N1 G20
N2 G17 G40 G80
N3 G90 G00 G54 X0 Y0 S600 M03
N4 G43 Z1.0 H07 M08      (ABOVE JOB CLEARANCE)
N5 G01 Z-0.65 F20.0    (CUTTER EDGE - BOTTOM)
N6 M98 P7000        (CUTTING GROOVE AT Z-0.65)
N7 G43 Z-0.43 H27          (CUTTER EDGE - TOP)
N8 M98 P7000        (CUTTING GROOVE AT Z-0.43)
N9 G00 Z1.0 M09
N10 G28 Z1.0 M05
N11 M30
%

O7000
(CHƯƠNG TRÌNH CON THAY RÃNH TRONG O1805)
N1 G01 G41 X0.875 Y-0.875 D07 F15.0
N2 G03 X1.75 Y0 R0.875 F10.0
N3 I-1.75
N4 X0.875 Y0.875 R0.875 F15.0
N5 G01 G40 X0 Y0
N6 M99
%
```

Trong ví dụ này, bù chiều dài dao H07 được dùng cho cạnh quy chiếu đáy của dao phay rãnh và H27 được dùng cho cạnh quy chiếu trên D07 chỉ được dùng cho bán kính dao. Hình 18.14 minh họa các chuyển động chạy dao trong chương trình con O7000.



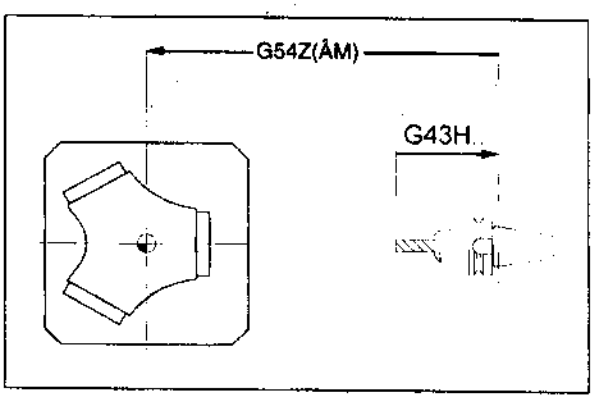
Hình 18.14. *Phay đường tròn – chương trình con O7000. Bắt đầu và kết thúc cắt ở tâm rãnh.*

ÁP DỤNG TRÊN TRUNG TÂM GIA CÔNG NGANG

Mọi ví dụ đã đề cập đều nhắm đến trung tâm gia công đứng. Tuy logic bù chiều dài dao được áp dụng cho trung tâm gia công *bất kỳ*, bất kể hướng trục Z, nhưng vẫn có các khác biệt rõ rệt trong ứng dụng thực tiễn trên trung tâm gia công ngang (Chương 45).

Trung tâm gia công ngang cho phép lập trình quỹ đạo chạy dao trên nhiều bề mặt của chi tiết. Do các mặt có khoảng cách khác nhau đến đỉnh dao (dọc theo trục Z), bù chiều dài dao cho từng mặt cũng khác nhau. Nói chung, cần lập trình các giá trị bù chi tiết khác nhau và các giá trị bù chiều dài dao khác nhau cho từng bề mặt gia công.

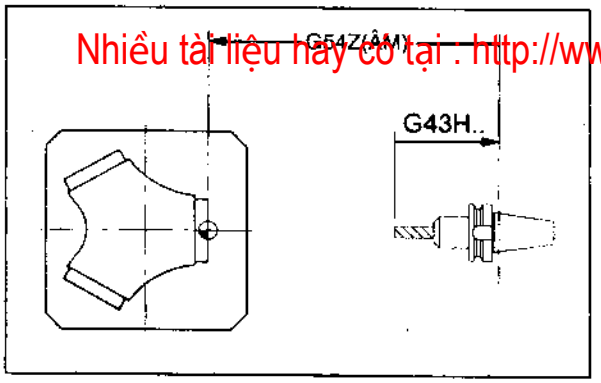
Hai hình minh họa nêu rõ sự xác lập bù chiều dài dao đối với dao được xác lập trước trên trung tâm gia công ngang. Hình 18.15 minh họa zero chương trình ở tâm bàn máy. Hình 18.16 minh họa zero chương trình ở bề mặt chi tiết.



Hình 18.15. *Xác lập bù chiều dài dao đối với dao được xác lập trước. Zero chương trình ở tâm bàn máy*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 18.16. *Xác lập bù chiều dài dao đối với dao được xác lập trước. Zero chương trình ở bề mặt chi*

XÓA BÙ CHIỀU DÀI DAO

Trong lập trình, điều quan trọng là phương pháp tổ chức hợp lý. Điều đó có nghĩa là lệnh chương trình được kích hoạt khi cần thiết và bị mất hiệu lực ngay khi không còn cần thiết. Các lệnh bù chiều dài dao cũng tuân theo quy tắc đó.

Sự xóa bù chiều dài dao có thể được gộp trong chương trình có một lệnh chuẩn bị đặc biệt được dùng để xóa phương pháp bất kỳ được chọn cho bù chiều dài dao, G43 hoặc G44. Lệnh xóa bù chiều dài dao trong chương trình (hoặc thông qua MDI) là G49:

G49	Xóa bù chiều dài dao

Phương pháp dùng lệnh G49 trong một block riêng, ngay trước khi trở về zero máy theo trục Z, ví dụ:

```
N176 G49
N177 G91 G28 Z0
...
```

Phương pháp thứ hai, xóa bù chiều dài dao

```
N53 G91 G28 Z0 H00
```

Trong trường hợp này, lệnh G28 đi kèm với H00. Chú ý, ở đây không dùng lệnh G49 trong block, H00 sẽ xóa bù chiều dài dao. Không có xác lập H00 trên bộ điều khiển, do đó chỉ có nghĩa là xóa bù chiều dài dao.

Chương trình còn có thể được *bắt đầu* với lệnh bù chiều dài dao bị xóa (trong điều khiển chương trình), thường trong dòng lệnh an toàn (block an toàn hoặc block khởi đầu)

```
N1 G20 G17 G40 G80 G49
```

... hoặc biến thể của block nêu trên.

```
N1 G20
N2 G17 G40 G80 G49
```

Phương pháp cuối cùng để xóa bù chiều dài dao là *hoàn toàn không lập trình lệnh xóa bù chiều dài dao.*

Điều đó có vẻ hơi lạ, nhưng vẫn có thể xảy ra. Hầu hết các ví dụ được nêu trong sách này đều *không sử dụng lệnh G49. Tại sao? Điều gì xảy ra ở cuối sự gia công của từng dao?*

Nguyên tắc của Fanuc hoàn toàn rõ ràng, lệnh G28 hoặc G30 (cả hai đều trả dao về zero máy) sẽ xóa chiều dài dao *một cách tự động.*

Ý nghĩa rất đơn giản, nhà lập trình có thể tận dụng ưu điểm của nguyên tắc này và không cần xóa bù chiều dài dao một cách chuyên biệt, nếu máy trở về vị trí thay dao. Điều này là bình thường đối với mọi máy có bộ thay dao tự động.

Phương pháp bất kỳ trong các phương pháp nêu trên sẽ bảo đảm xóa bù chiều dài dao. Có thể có vài khác biệt giữa các nhà chế tạo, bạn nên xem sổ tay hướng dẫn sử dụng máy để biết phương pháp cụ thể.

Có một số máy đòi hỏi sử dụng G49 cho từng dao

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

19 ĐỊNH VỊ NHANH

Máy công cụ CNC không thể cắt gọt liên tục và tạo ra các phoi cắt. Từ thời điểm dụng cụ cắt bắt đầu hoạt động trong chương trình gia công, dao sẽ trải qua nhiều chuyển động – bao gồm các chuyển động cắt gọt và chuyển động không cắt gọt (định vị)

Các chuyển động định vị là cần thiết nhưng không gia công. Không thể loại bỏ hoàn toàn các chuyển động này, chúng cần được quản lý một cách hiệu quả. Đối với mục đích đó, hệ thống CNC cung cấp tính năng được gọi là chuyển động *chạy dao nhanh*, nhằm rút ngắn thời gian định vị dao giữa các thao tác không cắt gọt, khi dao không tiếp xúc với chi tiết gia công. Chuyển động chạy dao nhanh thường gồm 4 kiểu chuyển động:

- Từ vị trí thay dao tiến đến chi tiết gia công
- Từ chi tiết trở về vị trí thay dao
- Chuyển động vòng qua các vật cản (nếu có)
- Chuyển động giữa các vị trí khác nhau trên chi tiết.

CHUYỂN ĐỘNG CHẠY DAO NHANH

Chuyển động chạy dao nhanh, đôi khi còn gọi là *chuyển động định vị*, là phương pháp đưa dụng cụ cắt từ vị trí này đến vị trí khác với *tốc độ nhanh của máy*. Tốc độ nhanh cực đại do nhà chế tạo máy CNC xác định, chỉ xảy ra trong giới hạn hành trình của máy.

Tốc độ nhanh phổ biến đối với nhiều máy CNC lớn là khoảng 450 in/ph (11430 mm/ph) Các máy hiện đại có tốc độ đến 1500 in/ph (38100 mm/ph) hoặc cao hơn, tốc độ này có thể là như nhau hoặc khác nhau trên từng trục. Tốc độ khác, thường được gán cho trục Z, còn các trục X và Y có tốc độ như nhau.

Chuyển động nhanh có thể được thực thi trên một trục, hoặc phối hợp đồng thời trên hai hoặc nhiều trục, được lập trình theo chế độ lập trình kích thước tuyệt đối hoặc tương đối (số gia) và được sử dụng khi trục chính quay hoặc đứng yên. Trong khi thực thi chương trình, người vận hành CNC có thể tạm dừng chuyển động nhanh bằng cách nhấn phím Feedhold trên bảng điều khiển, hoặc xác lập công tắc vượt qua Feedhold theo zero hoặc theo tốc độ giảm. Phương pháp thứ hai điều khiển tốc độ chuyển động nhanh là sử dụng hàm *chạy không*, chủ yếu trong quá trình gá lắp dao và chi tiết.

Lệnh G00

Lệnh chuẩn bị G00 được dùng trong chương trình CNC để khởi động chế độ chuyển động nhanh. Hàm ăn dao F là *không cần thiết* với G00, và, nếu được lập trình, sẽ bị bỏ qua trong khi chuyển động nhanh (trong chế độ G00). Sự ăn dao này sẽ được lưu trong bộ nhớ và sẽ có hiệu lực ngay khi xuất hiện chuyển động cắt gọt (G01, G02, G03, ...) trừ khi hàm F mới, được lập trình với chuyển động cắt.

➲ **Ví dụ A:**

```
N21 G00 X24.5 F30.0
N22 Y12.0
N23 G01 X30.0
```

Trong block N21, chỉ có chuyển động nhanh được thực thi. Lượng ăn dao 30.0in/min sẽ bị bỏ qua trong block này, được lưu lại để sử dụng. Block N22 cũng trong chế độ chuyển động nhanh do G00 là lệnh có tính chế độ. Block cuối, N23 là chuyển động tuyến tính (cắt gọt), đòi hỏi sự ăn dao. Do không có lượng ăn dao được gán trong block này, lượng ăn dao *đã lập trình gần nhất* sẽ được sử dụng. Lệnh được chuyên biệt trong block N21, trở thành lượng ăn dao *hiện hành* trong block N23, đó là F30.0.

➲ **Ví dụ B:**

```
N21 G00 X24.5 F30.5
N22 Y12.0
N23 G01 X30.0 F20.0
```

Trong block N21 lệnh G00 có tính chế độ và vẫn có hiệu lực cho đến khi bị xóa bằng lệnh khác trong cùng nhóm mã. Trong ví dụ B, lệnh G01 trong block N23 xóa chế độ chuyển động nhanhv à chuyển sang chế độ tuyến tính. Ngoài ra, lượng ăn dao được lập trình lại là 20.0 in/min bắt đầu từ block N23. Tốc độ ăn dao F30.0 trong block N21 bị xóa.

Tốc độ chạy dao nhanh được đo theo *khoảng cách với các đơn vị hiện hành chuyển động trong một phút* (*in/min* hoặc *mm/min*). Tốc độ cực đại luôn luôn do nhà chế tạo máy xác định, *không* phụ thuộc vào hệ thống điều khiển hoặc chương trình. Giới hạn do nhà chế tạo máy đưa ra thường trong khoảng 300–1500 in/min (7620–38100 mm/min) hoặc cao hơn. Do chuyển động theo thời gian là độc lập với sự quay của trục chính, tốc độ này có thể được áp dụng vào thời điểm bất kỳ, bất kể chế độ quay trục chính (M03, M04, M05).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tùy theo thiết kế máy CNC, tốc độ chuyển động nhanh có thể như nhau trên tất cả các trục, hoặc mỗi trục có giá trị cực đại riêng. Tốc độ nhanh cực đại đối với trung tâm gia công có thể đến 1181 in/min (30000 mm/min) trên các trục X và Y; và 945 in/min (24000 mm/min) trên trục Z. Đối với máy tiện CNC, các tốc độ này chậm hơn, ví dụ 197in/min (5000mm/min) đối với trục X, và 394 in/min (10000mm/min) trên trục Z. Các máy hiện đại có thể có tốc độ cao hơn.

QUỸ ĐẠO DAO CHUYỂN ĐỘNG NHANH

Mọi chuyển động trong chế độ G00 đều là chuyển động nhanh không theo đường tròn (chuyển động tròn hoặc xoắn thường không thể thực hiện với tốc độ cao). Chuyển động tuyến tính thực của dao giữa hai điểm chưa hẳn là đường ngắn nhất theo dạng đường thẳng. Quỹ đạo dao lập trình và quỹ đạo dao thực có thể khác nhau, phụ thuộc vào nhiều yếu tố:

- Số lượng trục được lập trình đồng thời.
- Chiều dài thực của chuyển động trên từng trục.
- Tốc độ chạy dao nhanh của từng trục.

Do mục đích duy nhất của chuyển động nhanh là tiết kiệm thời gian (chuyển động từ vị trí dao hiện hành đến vị trí dao đích), quỹ đạo dao khác với biên dạng chi tiết gia công. Vì các lý do an toàn, bạn cần chú ý quỹ đạo chạy dao nhanh, đặc biệt khi lập trình đồng thời cho hai hoặc nhiều trục. Trên quỹ đạo chuyển động nhanh phải không có chướng ngại.

Nếu có chướng ngại vật giữa hai điểm bất kỳ trong quỹ đạo dao, bộ điều khiển *không thể giúp dao* tránh chướng ngại vật đó, đơn giản do bộ điều khiển không có phương tiện phát hiện chướng ngại vật. Nhà lập trình chịu trách nhiệm bảo đảm mọi chuyển động dao (kể cả chạy dao nhanh) xảy ra không gặp chướng ngại vật.

Các chướng ngại vật có thể can thiệp vào chuyển động dao bao gồm:

- ĐỐI VỚI TRUNG TÂM GIA CÔNG
  Kẹp, ngàm định vị, đồ gá, bàn máy quay hoặc phân độ, chi tiết gia công, ...
- ĐỐI VỚI MÁY TIỆN
  Ụ động, mũi chống tâm, mâm cặp, gối tựa, tâm sống, tấm mặt, đồ gá, dao khác, chi tiết gia công, ...

Các chướng ngại vật khác có thể phát sinh từ phương pháp gá lắp và định vị, thiết kế máy, phương pháp lắp dao, ...

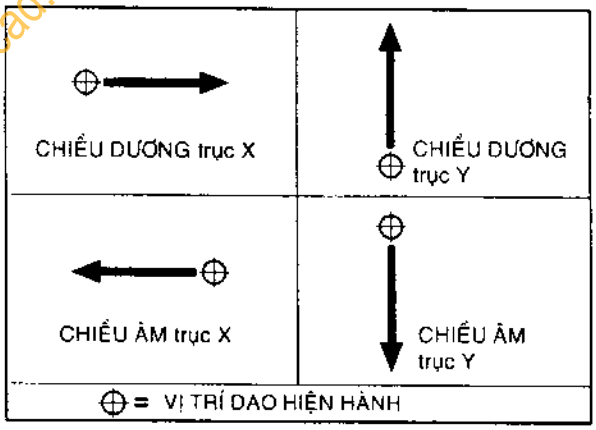
Luôn luôn chú ý đến các chướng ngại vật trong quá trình chuyển động nhanh!

Mặc dầu chướng ngại vật có thể hiện diện trên hành trình các chuyển động cắt trong chế độ G01, G02 hoặc G03 (ví dụ, tiện mặt ngoài hướng đến ụ động trên máy tiện) nhưng hầu hết các vấn đề chỉ xảy ra trong chế độ chạy dao nhanh G00, G29, G30, và các chu kỳ cố định G81 đến G89, G73 và G76. Trong khi chuyển động nhanh, quỹ đạo dao có thể dễ dự đoán hơn so với chuyển động cắt. Bạn cần nhớ, mục đích duy nhất của chuyển động nhanh là đưa *nhanh* dao từ vị trí này đến vị trí khác, nhưng không nhất thiết phải là đường thẳng.

Để vòng qua các chướng ngại vật và vẫn bảo đảm chuyển động nhanh an toàn trong chương trình, bạn hãy khảo sát các tùy chọn khả dụng trong lập trình chuyển động nhanh.

Chuyển động theo một trục

Chuyển động dao bất kỳ được lập trình một cách chuyên biệt mỗi lần chỉ cho một trục luôn luôn là đường thẳng dọc theo trục đã chọn. Nói cách khác, từng chuyển động nhanh song song với một trong các trục khả dụng phải được lập trình trong block riêng. Chuyển động này luôn luôn tương đương với khoảng cách ngắn nhất giữa các điểm đầu và cuối của chuyển động (Hình 19.1).



Hình 19.1. Chuyển động theo 1 trục đối với gia công tâm (trục XY)

Các block chương trình liên tiếp, mỗi block chỉ chứa chuyển động trên một trục, có thể gộp trong chương trình để tránh các chướng ngại khi gia công. Phương pháp lập trình này được ưa chuộng trong các trường hợp biết vị trí *chính xác* hoặc *gần đúng* của một số chướng ngại vật (chẳng hạn đồ gá hoặc kẹp định vị) trong khi chuẩn bị chương trình.

Chuyển động nhiều trục

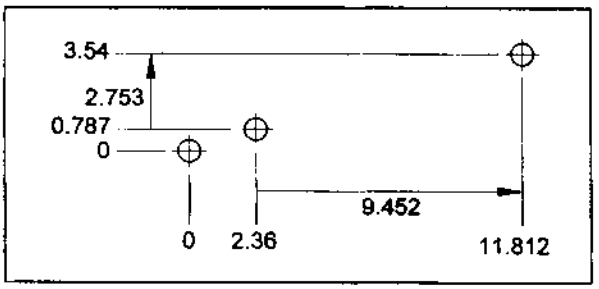
Bạn đã biết lệnh G00 làm cho dao cắt chuyển động nhanh. Nếu đây là chuyển động đồng thời trên hai hoặc nhiều trục, quỹ đạo

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

nhanh động lập trình và chuyển động thực tế của dao có thể, và thường, khác với chuyển động lý thuyết đã lập trình (chuyển động mong muốn).

Về lý thuyết, chuyển động dọc theo hai trục bất kỳ, tương đương với *chuyển động thẳng theo đường chéo*. Tuy nhiên chuyển động thực có thể hoặc không thể theo đúng quỹ đạo đường chéo. Bạn hãy xét ví dụ trên Hình 19.2.



Hình 19.2. Bản vẽ minh họa ví dụ chuyển động

Vị trí dao hiện hành (điểm bắt đầu) có tọa độ X2.36 và Y0.787. Chuyển động dao kết thúc ở X11.812 Y3.54. Nếu xét theo chuyển động số gia (tương đối), dao cắt phải chuyển động 9.452 in theo trục X và 2.753 in theo Y.

Nếu tốc độ nhanh trên cả hai trục là như nhau (đây là điều thường xảy ra), ví dụ 394 in/min, sẽ cần:

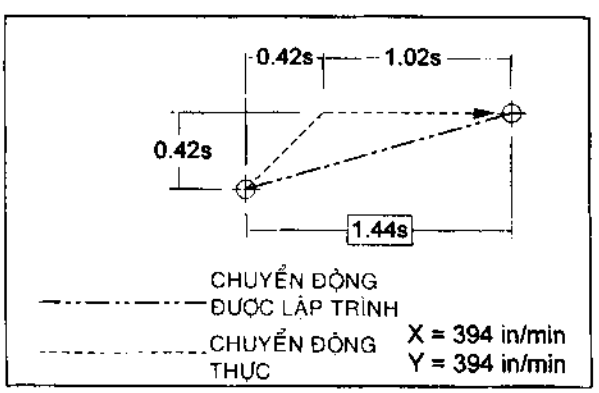
```
(9.452 x 60) / 394 = 1.44 giây
```

để hoàn tất chuyển động theo trục X, nhưng chỉ cần:

```
(2.753 x 60) / 394 = 0.42 giây
```

để hoàn tất chuyển động theo trục Y. Do chuyển động này chỉ hoàn tất khi *cả hai* trục đều đạt đến điểm cuối, rõ ràng quỹ đạo thực của dao sẽ *khác* với quỹ đạo lập trình.

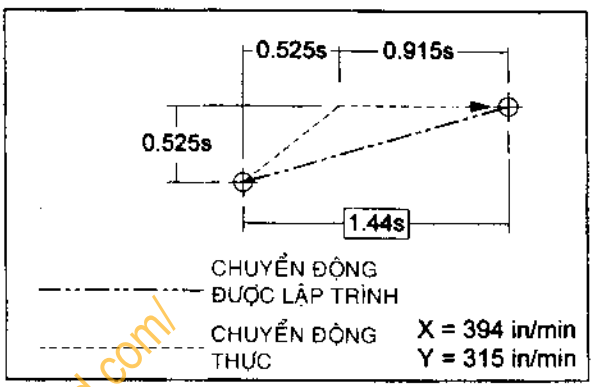
Hình 19.3 minh họa sự phối hợp chuyển động góc và chuyển động thẳng tạo thành quỹ đạo thực của dao. Dao này bắt đầu với tốc độ



Hình 19.3. Sự lệch chuyển động nhanh – tốc độ trên cả hai trục là như nhau

(10000 mm/min) đồng thời trên *cả hai* trục, với chuyển động kết quả là 45^0. Tổng thời gian cần thiết để đến điểm cuối là 1.44 giây, là thời gian cần thiết dài nhất đối với *một trong hai* trục để đạt đến vị trí đích. Sau 0.42 giây, vị trí đích trên trục Y đã đạt được, nhưng vẫn còn 1.02 giây để hoàn tất chuyển động dọc theo trục X. Cần đạt đến vị trí đích trên cả hai trục, do đó dao chỉ tiếp tục chạy theo trục X (trong 1.02 giây) để đến vị trí cuối.

Ví dụ thứ hai, cũng sử dụng các tọa độ vị trí như trên Hình 19.2, minh họa tình huống khác, với tốc độ chuyển động nhanh *khác nhau* giữa hai trục (Hình 19.4).



Hình 19.4. Sự lệch chuyển động nhanh – tốc độ khác nhau trên hai trục.

Trong ví dụ này, ít xảy ra trong thực tế, tốc độ trên trục X là 394 in/min (10000mm/min) và trên trục Y là 315 in/min (8000 mm/min). Dao cần:

```
(9.452 x 60) / 394 = 1.44 giây
```

để hoàn tất trên trục X và chỉ cần:

```
(2.753 x 60) / 315 = 0.525 giây
```

để hoàn tất chuyển động trên trục Y. Trong trường hợp này, chuyển động kết quả cũng có sự khởi đầu theo góc, nhưng *không phải* là góc 45^0 do tốc độ khác nhau trên các trục. Trong 0.525 giây (chung cho cả hai trục), hành trình trên trục X là:

```
0.525/60 x 394 = 3.448 in
```

nhưng trên trục Y chỉ là:

```
0.525 / 60 x 315 = 2.753 in
```

Chuyển động kết quả theo góc $38,605^0$. Góc xuất phát thực có thể không cần biết, nhưng sẽ giúp bạn tính toán các chuyển động nhanh trong một số vùng rất hẹp của chi tiết. Bạn chỉ cần dùng vài hàm lượng giác đơn giản để tính toán quỹ đạo chạy dao nhanh nếu biết tốc độ chuyển động nhanh.

Hai ví dụ nêu trên minh họa chuyển động

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

góc (nghiêng) (hoặc theo hai trục tiện thẳng) là chuyển động thẳng trên một trục. Biểu diễn đồ thị của các chuyển động này là đường cong, rất phổ biến trong thực tế.

Chuyển động nhanh ngược chiều

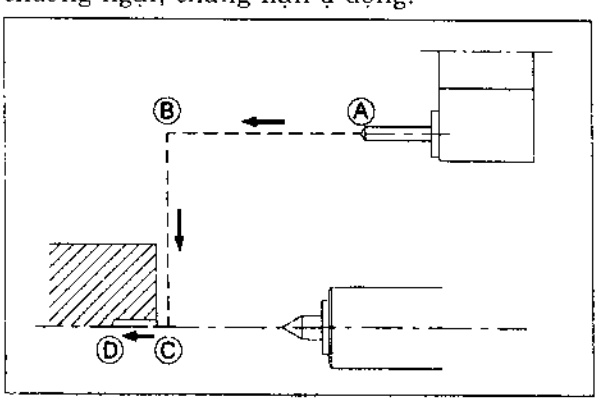
Chuyển động nhanh cần được xét theo hai chiều, *tiến đến* chi tiết, và *lùi về* vị trí thay dao. Đây là cách thức dụng cụ cắt thường được lập trình – chuyển động bắt đầu từ vị trí xác định và trở về vị trí đó, khi hoàn tất toàn bộ hoạt động cắt gọt. Đây không phải là phương pháp bắt buộc, nhưng có tính tổ chức cao, cho phép lập trình dễ dàng hơn.

Cho đến phần này bạn đã xét chuyển động nhanh *trước khi* cắt gọt thực sự, bắt đầu từ vị trí thay dao. Khi hoàn tất cắt gọt, cần có chuyển động nhanh để trả dao về vị trí thay dao.

Khảo sát này đối với tiện thường quan trọng hơn phay, do bản chất lập trình của hai kiểu máy. Trong nguyên công tiện, chuyển động tiến có thể dọc theo trục Z trước để tránh va chạm với ụ động, *sau đó* theo trục X. Chuyển động lùi sẽ theo trục X trước, *sau đó* theo trục Z, để bảo đảm an toàn khi trở về vị trí thay dao.

Ứng dụng của kiểu lập trình này có thể rất hữu ích sau khi sử dụng chu kỳ gia công (chẳng hạn tiện, doa, cắt ren, ...) với điểm bắt đầu cũng là điểm kết thúc chu kỳ.

Từ Hình 19.5 có thể thấy, thay vì lập trình chuyển động trực tiếp từ vị trí ổ dao đến vị trí cắt gọt (từ A đến C), chuyển động dao được chia thành nhiều phần. Chuyển động tiến đến chi tiết sẽ theo thứ tự A đến B đến C, với tốc độ nhanh. Từ C đến D là quá trình cắt gọt thực sự. Sau khi hoàn tất cắt gọt, dao sẽ chuyển động nhanh theo thứ tự ngược lại, trở về vị trí ban đầu. Chuyển động nhanh sẽ từ D đến C đến B và đến A. Điều này là cần thiết để tránh các chướng ngại, chẳng hạn ụ động.



Hình 19.5. *Ví dụ về chuyển động lùi dao nhanh trên máy tiện CNC, được dùng để tránh các chướng ngại, chẳng hạn ụ động.*

Kỹ thuật lập trình từng trục riêng rẽ trong từng block của chương trình, chủ yếu chỉ nhằm tránh các chướng ngại tiềm ẩn trên quỹ đạo chạy dao, đặc biệt để bảo đảm an toàn. Phương pháp lập trình này đòi hỏi thời gian chu kỳ hơi dài hơn so với chuyển động nhanh đồng thời trên nhiều trục. Để so sánh sự khác biệt, bạn hãy xét chuyển động nhanh ba trục, thường áp dụng trên máy phay.

Ví dụ, tốc độ chuyển động nhanh là 394 in/min (10000mm/min) trên từng trục. Chuyển động xảy ra giữa hai vị trí tọa độ X2.36 Y0.787 Z0.2 (điểm đầu) và X11.812 Y354 Z1.0 (điểm cuối).

Thời gian cần thiết để dao chạy trên từng trục có thể tính là:

- Thời gian trục X
  ((11.812 – 2.36) x 60) / 394 = 1.440 giây
- Thời gian trục Y
  ((3.54 – 0.787) x 60) / 394 = 0.420 giây
- Thời gian trục Z
  ((1.0 – 0.2) x 60) / 394 = 0.121 giây

Nếu chuyển động đồng thời trên cả ba trục, thời gian toàn phần để định vị là 1.140 giây là thời gian dài nhất cần thiết cho trục bất kỳ để đến điểm cuối. Block chương trình sẽ như sau:

```
G00 X11.812 Y3.54 Z1.0
```

Nếu chuyển động này được tách ra thành ba block riêng rẽ, thời gian toàn phần sẽ là tổng của ba thời gian cần thiết ứng với từng trục.

```
1.44 + 0.42 + 0.121 = 1.981 giây
```

dài hơn khoảng 37.5%. Số phần trăm này thay đổi tùy thuộc tốc độ chuyển động nhanh và chiều dài hành trình, được đo trên từng trục máy. Các block chương trình được viết riêng rẽ như sau:

```
G00 X11.812
Y3.54
Z1.0
```

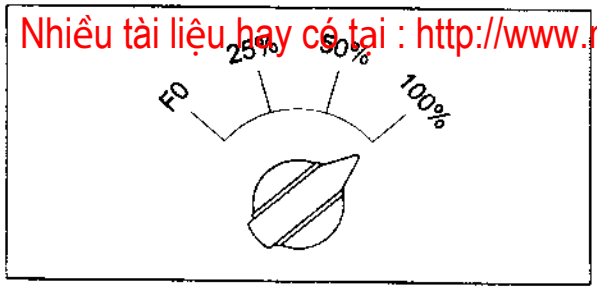
Chú ý, tính chế độ của lệnh chuyển động nhanh G00 không cần lặp lại trong các block kế tiếp.

GIẢM TỐC ĐỘ CHẠY DAO NHANH

Trong khi gá lắp chi tiết hoặc kiểm chứng chương trình mới trên máy, người vận hành CNC có thể chọn *tốc độ chạy dao nhanh chậm hơn* so với giá trị cực đại được nhà chế tạo máy thiết lập. Sự điều chỉnh này được thực hiện thông qua *công tắc rapid override* trên bảng điều khiển. Công tắc này thường có bốn vị trí

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 19.6. Công tắc vượt qua chuyển động nhanh xác lập với tốc độ 100%

lựa chọn, tùy theo máy và kiểu bộ điều khiển (Hình 19.6).

Các vị trí thứ hai, thứ ba, và thứ tư trên công tắc này được định mức theo *số phần trăm tốc độ nhanh thực tế* - 25%, 50%, và 75%, do nhà chế tạo máy xác lập. Xác lập thứ nhất, F0 (hoặc F1) là tốc độ chạy dao nhanh được xác định thống qua *tham số hệ thống điều khiển*. Xác lập F0 (hoặc F1) luôn luôn thấp hơn các xác lập khác, dưới 25%.

Cấu hình của công tắc rapid override có thể khác nhau tùy theo nhà chế tạo máy. Trên một số máy, chuyển động nhanh có thể dừng hoàn toàn, trên các máy khác, dao sẽ chuyển động với số phần trăm thấp nhất và không thể dừng nếu chỉ dùng công tắc này.

Trong sản xuất thực tế, sau khi kiểm chứng và tối ưu hóa chương trình để có năng suất và hiệu suất cao nhất, công tắc này phải xác lập ở vị trí 100% để rút ngắn thời gian chu kỳ.

CÔNG THỨC CHUYỂN ĐỘNG NHANH

Các tính toán về chuyển động chạy dao nhanh có thể biểu thị bằng công thức và được sử dụng vào thời điểm bất kỳ bằng cách thay các tham số đã biết. Quan hệ giữa tốc độ chạy dao nhanh, chiều dài hành trình, và thời gian được biểu thị theo ba công thức.

$$T = \frac{L \times 60}{R}$$

$$R = \frac{L \times 60}{T}$$

$$L = \frac{T \times R}{60}$$

Trong đó: T: Thời gian, tính theo giây.
R: Tốc độ chạy dao nhanh trên trục cho trước - in/min hoặc mm/min.
L: chiều dài chuyển động - inch hoặc mm

Các đơn vị phải thống nhất với hệ thống đo được sử dụng trong chương trình, theo hệ Anh (inch, inch/min) hoặc theo hệ mét (mm, mm/min)

TIẾN ĐẾN CHI TIẾT

Hình 19.5 minh họa sự tiến dao an toàn áp dụng cho máy tiện CNC. Đối với trung tâm gia công CNC, cũng cần chú ý đặc biệt về tính an toán khi dao tiến đến chi tiết. Bạn cần nhớ, các nguyên tắc chung về chuyển động nhanh cần được xem xét cho mọi máy công cụ. Khi tiến đến chi tiết với tốc độ nhanh, thời gian chu kỳ có thể giảm đôi chút bằng cách duy trì khoảng hở chi tiết theo giá trị an toàn nhỏ nhất. Bạn hãy khảo sát một số vấn đề.

Trong ví dụ dưới đây, dao tiến đến chi tiết được thực hiện dọc theo trục Z, với khoảng hở 0.05 in (1.27mm) trong block N315.

```
N314 G90 G54 G00 X10.0 Y8.0 S1200 M03
N315 G43 Z0.05 H01
N316 G01 Z-1.5 F12.0
```

Không có gì sai với phương pháp lập trình, nếu dao cắt được xác lập đúng và chiều cao chi tiết là như nhau giữa các chi tiết. Khoảng hở 0.05 in (1.27 mm) là rất nhỏ. Mặt khác, người vận hành CNC thiếu kinh nghiệm có thể không cảm thấy thoải mái với khoảng hở nhỏ này. Nếu sự thoải mái của người vận hành là yếu tố quan trọng góp phần vào năng suất chung, có thể chia chuyển động trục Z vào hai block riêng rẽ.

```
N314 G90 G54 G00 X10.0 Y8.0 S1200 M03
N315 G43 Z0.5 H01
N316 G01 Z0.05 F100.0
N317 Z-1.5 F12.0
```

Trong phương pháp này, chuyển động nhanh ban đầu được lập trình với vị trí 0.500 in phía trên chi tiết (N315). Sau đó chuyển động tiếp tục đến điểm bắt đầu cắt gọt, sử dụng nội suy *tuyến tính* G01 trong block N316. Do đây vẫn là chuyển động tự do, sự cắt gọt chưa xảy ra, có thể dùng tốc độ ăn dao lớn để rút ngắn thời gian.

Mặc dầu thời gian cắt gọt tăng đôi chút, nhưng đồng thời, người vận hành CNC sẽ có cơ hội sử dụng công tắc vượt qua sự ăn dao để kiểm tra chi tiết thứ nhất (có thể được dùng trong chế độ block đơn). Khi chương trình đã được kiểm chứng và tối ưu hóa, tốc độ ăn dao lớn trong chuyển động không cắt gọt sẽ làm tăng nhanh sự vận hành đồng thời cung cấp thêm khoảng hở an toàn. Chương trình với sự phân chia chuyển động trục Z có thể được tối ưu hóa, dù điều này có thể chưa phải là tốt nhất đối với các gia công lặp lại, do sự gá lắp luôn luôn "mới" đối với sự lặp lại bất kỳ. Tuy nhiên, có thể sẽ rất hữu dụng khi gia công hàng loạt lớn (hàng ngàn chi tiết cùng loại).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

20 TRẢ VỀ ZERO MÁY

Khả năng của hệ thống điều khiển đưa dụng cụ cắt từ vị trí bất kỳ đến vị trí quy chiếu máy (chuẩn máy) là tính năng cơ bản của mọi hệ thống CNC hiện đại. Nhà lập trình và người vận hành hiểu khái niệm *vị trí quy chiếu máy* và vị trí *zero máy* hoặc vị trí *chuẩn máy*. Đây là *vị trí của mọi sống trượt máy tại một trong các giới hạn hành trình tối đa trên từng trục*. Vị trí chính xác được nhà chế tạo máy quy định thường *không* thay đổi trong suốt tuổi thọ của máy. Sự trả về vị trí đó là tự động, theo yêu cầu từ băng điều khiển, trong chế độ MDI, hoặc thông qua chương trình.

VỊ TRÍ QUY CHIẾU MÁY

Sự hiện hữu của vị trí quy chiếu máy nhằm các mục đích quy chiếu. Để bảo đảm máy CNC chính xác, không chỉ cần có các bộ phận máy chất lượng cao, mà còn phải có vị trí đặc thù có thể được coi là điểm chuẩn của máy – vị trí zero hoặc vị trí gốc.

Zero máy là vị trí cố định trên máy CNC có thể vươn tới một cách lặp lại nhiều lần, theo yêu cầu, thông qua bảng điều khiển, MDI, hoặc thực thi mã chương trình.

Trung tâm gia công

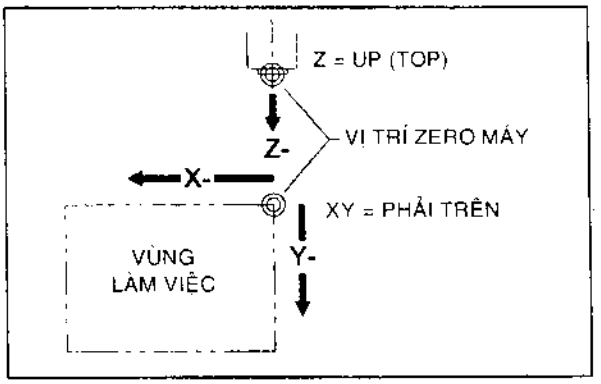
Mặc dầu các trung tâm gia công có thiết kế khác nhau, nhưng chỉ có 4 vị trí có thể là zero máy, trong mặt phẳng XY:

- Góc dưới bên trái của máy.
- Góc trên bên trái của máy.
- Góc dưới bên phải của máy.
- Góc trên bên phải của máy.

Nói chung, thường bắt đầu phần thứ nhất của chương trình mới từ vị trí zero máy. Thông thường, nhiều chương trình thực hiện sự thay dao ở vị trí zero máy và trở về vị trí đó khi hoàn tất sự thực thi chương trình. Một số vị trí nêu trên là không thuận tiện để gá lắp chi tiết trên bàn máy và tháo chi tiết sau khi gia công.

Vị trí quy chiếu máy tiêu chuẩn đối với trung tâm gia công đứng là *góc trên bên phải* của máy khi chiếu vuông góc đến mặt phẳng XY (Hình 20.1).

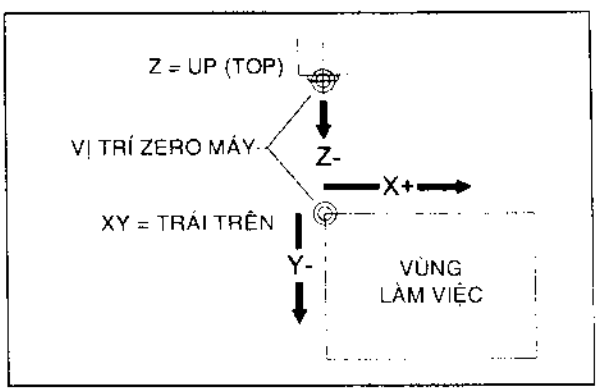
Vị trí zero máy trên trục Z luôn luôn là nơi thực hiện thay dao tự động (ATC). Đây là vị trí



Hình 20.1. *Vị trí zero máy ở góc trên bên phải của trung tâm gia công đứng CNC*

thường được đặt ở khoảng cách an toàn từ bàn máy và vùng làm việc. Đối với hầu hết các máy, zero máy tiêu chuẩn của trung tâm gia công CNC là giới hạn hành trình tối đa của từng trục theo chiều *dương*, tuy nhiên vẫn có thể có các ngoại lệ.

Như minh họa trên Hình 20.2, một số trung tâm gia công đứng CNC có vị trí zero máy ở *góc trên bên trái* của mặt phẳng XY.



Hình 20.2. *Vị trí zero máy ở góc trên bên trái XY của trung tâm gia công đứng CNC.*

Trong cả hai minh họa, mũi tên hiển thị chiều chuyển động dao *hướng đến vùng làm việc*, chuyển động dao từ zero máy theo chiều ngược lại sẽ dẫn đến điều kiện được gọi là *hành trình quá mức* – bạn hãy so sánh hai khả năng:

- Chuyển động dao từ zero máy, nếu zero máy ở góc trên *bên phải*:
  X+Y+Z+ ... chuyển động dao sẽ quá hành trình
- Chuyển động từ zero máy, nếu zero máy ở góc trên *bên trái*:
  X+Y+Z+ ... chuyển động dao sẽ quá hành trình

Nhiều tài liệu hay có tại : http://www.me-cad.com/

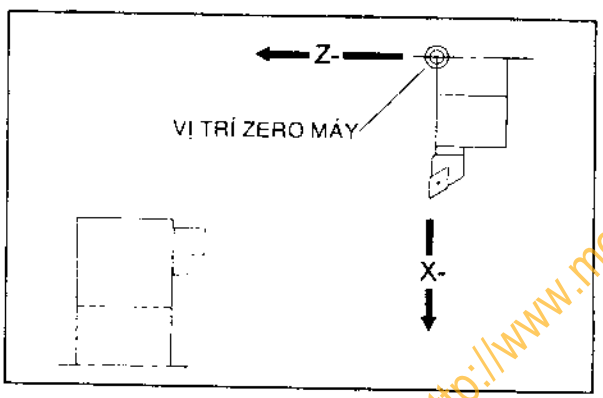
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Hai góc còn lại (dưới bên trái và dưới bên phải của mặt phẳng XY) không được dùng làm zero máy.

Máy tiện

Vị trí quy chiếu máy đối với máy tiện CNC hai trục là không khác với vị trí quy chiếu của trung tâm gia công. Sự tiếp cận dễ dàng của người vận hành đối với chi tiết lắp đặt là yếu tố quyết định. Cả hai trục X và Y đều có vị trí quy chiếu máy ở vị trí xa nhất tính từ chi tiết quay, có nghĩa là cách xa khu vực mâm cặp.

Đối với trục X, vị trí quy chiếu zero máy luôn luôn là giới hạn của hành trình *cách xa đường tâm trục chính*. Đối với trục Z, vị trí quy chiếu luôn luôn là hành trình cực đại *cách xa mâm cặp*. Trong cả hai trường hợp, thường có nghĩa là chiều dương hướng đến zero máy, tương tự như trung tâm gia công. Minh họa trên Hình 20.3 trình bày zero máy trên máy tiện CNC điển hình.



Hình 20.3. *Vị trí zero máy trên máy tiện CNC*

Trong minh họa này, các mũi tên biểu thị chiều chuyển động dao hướng đến vùng làm việc. Chuyển động dao từ zero máy theo chiều ngược lại sẽ gây ra *hành trình quá mức* trên trục đó.

- Chuyển động dao từ zero máy của máy tiện điển hình.
  X+Z+ ... chuyển động dao sẽ quá hành trình

Xác lập các trục máy

Từ các nguồn nêu trên, bạn có thể thấy có quan hệ trực tiếp giữa máy CNC, dụng cụ cắt và chi tiết. Điểm quy chiếu gia công (zero chương trình, zero chi tiết, chuẩn gia công) do nhà lập trình CNC xác lập, điểm quy chiếu dao được xác định theo chiều dài dao ở lưỡi cắt, cũng do nhà lập trình lựa chọn.

Chỉ có điểm quy chiếu máy (chuẩn máy) do nhà chế tạo máy xác định và được thiết lập ở vị trí *cố định*. Đây là yếu tố rất quan trọng.

Zero máy cố định có nghĩa là mọi điểm quy chiếu khác đều phụ thuộc vào vị trí này

Để đạt đến vị trí quy chiếu máy (chuẩn máy) và xác lập các trục máy, ví dụ trong khi lắp chi tiết hoặc đồ gá, có ba phương pháp khả dụng đối với người vận hành CNC:

- *Bằng tay* – sử dụng bảng điều khiển của hệ thống
  Người vận hành sẽ sử dụng các nút hoặc công tắc XYZ (trung tâm gia công) hoặc XZ (máy tiện) chuyên dùng cho mục đích này. Một hoặc nhiều trục máy có thể được kích hoạt đồng thời, tùy theo hệ điều khiển.
- *Sử dụng MDI* – chế độ nhập dữ liệu bằng tay
  Phương pháp này cũng sử dụng bảng điều khiển. Trong trường hợp này, người vận hành máy xác lập chế độ MDI và lập trình chuyển động dao, sử dụng các lệnh chương trình tương ứng (G28, G30)
- *Trong chương trình CNC* – trong khi đang gia công
  Sử dụng cùng các lệnh chương trình như trong chế độ MDI, nhà lập trình CNC, không phải người vận hành, đưa lệnh (hoặc các lệnh) trả về zero máy trong chương trình, tại các vị trí mong muốn.

Khi người vận hành thực hiện sự trở về zero máy, họ thường xác lập các vị trí tương đối và tuyệt đối theo zero trên màn hình hiển thị. Bạn cần nhớ, hiển thị tương đối chỉ có thể xác lập theo zero trên bảng điều khiển và hiển thị tuyệt đối chỉ có thể thay đổi thông qua sự bù chi tiết, chế độ MDI hoặc chương trình gia công.

Đối với phương pháp trả về zero máy theo MDI hoặc trong chương trình CNC, hệ thống CNC cung cấp các lệnh chuẩn bị chuyên biệt.

Các lệnh chương trình

Có 4 lệnh chuẩn bị liên quan đến vị trí quy chiếu zero máy:

G27	*Kiểm tra* sự trở về vị trí quy chiếu zero máy
G28	Trở về vị trí quy chiếu zero máy *sơ cấp*
G29	Trở về *từ* vị trí quy chiếu zero máy
G30	Trở về vị trí quy chiếu zero máy *thứ cấp.*

Trong 4 lệnh nêu trên, G28 hầu như chỉ sử dụng trong lập trình CNC hai hoặc ba trục. Mục đích duy nhất của lệnh này là đưa dao hiện hành trở về vị trí zero máy dọc theo một hoặc nhiều trục chuyên biệt trong block chương trình G28

Nhóm lệnh

Cả 4 lệnh chuẩn bị G27 đến G30 đều thuộc nhóm 00 theo tiêu chuẩn Fanuc quy định các mã G *không chế độ* hoặc chỉ có *tác dụng một lần*. Trong tiêu chuẩn này, từng mã G của

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

nhóm 00 phải lặp lại trong block sử dụng mã đó. Ví dụ, khi lệnh G28 được dùng trong một block cho trục Z sau đó được dùng trong block kế tiếp cho các trục Y và X, cần phải lặp lại trong *từng* block:

```
N230 G28 Z..        (MACHINE ZERO RETURN Z AXIS)
N231 G28 X.. Y..  (MACHINE ZERO RETURN XY AXES)
```

Lệnh G28 trong block N231 *phải* được lặp lại. Nếu sót lệnh này, lệnh *chuyển động gần nhất* được lập trình sẽ có hiệu lực, chẳng hạn G00 hoặc G01.

TRẢ VỀ ZERO MÁY SƠ CẤP

Máy CNC bất kỳ có thể có hơn một điểm quy chiếu zero máy (chuẩn máy) tùy theo thiết kế. Ví dụ, nhiều trung tâm gia công với bộ thay đổi mâm dao có vị trí quy chiếu máy *thứ cấp*, thường được dùng để chỉnh thẳng hàng cả hai mâm dao trái và phải khi thay mâm dao. Hầu hết các máy công cụ đều chỉ sử dụng một vị trí zero máy. Để đến vị trí sơ cấp này, lệnh chuẩn bị G28 được dùng trong chương trình và còn có thể được dùng trong chế độ MDI.

Lệnh G28 dịch chuyển trục hoặc các trục *chuyên biệt* đến vị trí zero máy, luôn luôn theo tốc độ chạy dao nhanh. Điều đó có nghĩa là lệnh G00 là mặc định và không cần lập trình. Trục hoặc các trục của chuyển động mong muốn (với giá trị nào đó) phải *luôn luôn* lập trình. Chỉ những trục được lập trình mới có hiệu lực.

➲ **Ví dụ**

```
N67 G28
```

cho thấy chỉ có G28 được lập trình trong block – đây là lệnh *còn thiếu*. Ít nhất một trục phải được chuyên biệt với lệnh G28, ví dụ:

```
N67 G28 Y..
```

sẽ chỉ đưa trục Y đến vị trí quy chiếu máy, hoặc...

```
N67 G28 Z..
```

sẽ chỉ đưa trục Z đến vị trí quy chiếu zero máy, và

```
N67 G28 X.. Y.. Z..
```

sẽ đưa cả ba trục này về vị trí zero máy. Chuyển động nhiều trục bất kỳ đều đòi hỏi chú ý đặc biệt về khả năng *hành trình bị gấp khúc*.

Điểm trung gian

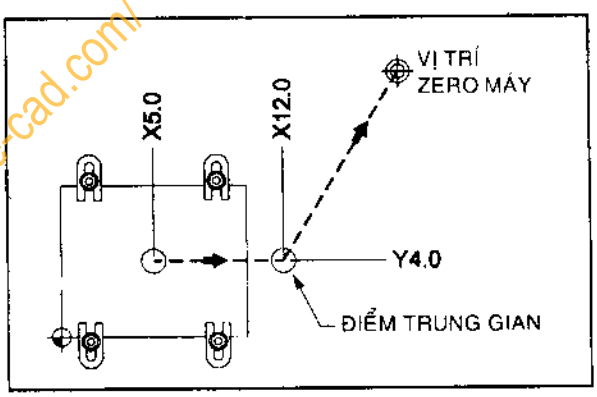
Một trong các yêu cầu cơ bản trong lập trình là thành phần các chữ và số của từ ngữ. Trong chương trình, mỗi chữ cái phải được tiếp theo bằng một hoặc nhiều chữ số. Câu hỏi đặt ra là các trục trong G28 có *các giá trị* nào? Chúng sẽ là điểm *trung gian* trong chuyển động trở về zero máy. Khái niệm chuyển động trung gian trong G28 hoặc G30 là một trong các tính năng lập trình thường bị hiểu sai.

Các lệnh G28 và G30 phải luôn luôn chứa *điểm trung gian* (vị trí dao). Theo định nghĩa và thiết kế của Fanuc, các lệnh G28 / G30 có chuyển động *được thiết lập sẵn* để đến điểm trung gian, *trên đường* về zero máy.

Các giá trị tọa độ của các trục liên quan với lệnh G28 và G30 luôn luôn biểu thị điểm trung gian.

Mục đích của điểm hoặc vị trí trung gian là rút ngắn chương trình, thường là một block. Sự rút gọn này hơi đặc biệt, do đó bạn cần hiểu khái niệm về điểm trung gian.

Khi lệnh G28 hoặc G30 được dùng trong chương trình, *ít nhất phải chuyên biệt một trục* trong block. Giá trị của trục đó là điểm trung gian, do hệ điều khiển diễn dịch. Các chế độ tuyệt đối và số gia G90 và G91 tạo ra sự khác biệt lớn khi diễn dịch hành vi của G28 hoặc G30, sẽ được trình bày ngắn gọn dưới đây.



Hình 20.4. *Điểm trung gian đối với trở về zero máy – theo các trục X và Y.*

Chuyển động dao trên Hình 20.4 là từ lỗ tâm của chi tiết. Trong chuyển động đó, dao có thể va chạm với kẹp định vị góc trên bên phải trên đường đến zero máy nếu chuyển động đến vị trí này được lập trình trực tiếp. Trên hình chỉ xét các trục X và Y. *Điểm trung gian* có thể được lập trình ở vị trí an toàn, mà không làm cho chương trình dài hơn. Chương trình *không có* điểm trung gian có thể được cấu trúc như sau:

```
G90
...
G00 X5.0 Y4.0              (MACHINED HOLE)
G28 X5.0 Y4.0        (MACHINE ZERO MOTION)
...
```

Chương trình *có* điểm trung gian ở vị trí an toàn sẽ thay đổi đôi chút.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
G90
...
G00 X5.0 Y4.0                    (MACHINED HOLE)
G28 X12.0 Y4.0                   (MACHINE ZERO MOTION)
...
```

Các ví dụ đó nêu rõ lý do về chuyển động kép. Điều này rất đơn giản, chỉ để *tiết kiệm một block chương trình*. Mục đích là sử dụng một block của chương trình để đạt được hai chuyển động, bình thường cần đến hai block. Chương trình an toàn cũng có thể như sau:

```
G90
...
G00 X5.0 Y4.0                    (MACHINED HOLE)
X12.0                            (SAFE LOCATION)
G28 X12.0 Y4.0                   (MACHINE ZERO MOTION)
...
```

sẽ có cùng kết quả, nhưng *có thêm một block*

Ví dụ, sử dụng điểm trung gian, dao có thể được lập trình để tránh chướng ngại trên đường trở về zero máy. Nếu được lập trình cẩn thận, vị trí trung gian có thể rất hữu ích. Nói chung, sẽ thực tiễn hơn khi làm cho điểm trung gian bằng zero và dịch chuyển dụng cụ cắt đến zero máy một cách trực tiếp. Điều này được thực hiện bằng cách chuyên biệt điểm trung gian *trùng với* vị trí dao hiện hành trong chế độ *tuyệt đối* hoặc chuyên biệt chuyển động dao là zero trong chế độ *số gia*.

Chế độ tuyệt đối và chế độ số gia

Có sự khác biệt cơ bản trong lập trình lệnh trở về zero máy G28 hoặc G30 trong các chế độ *tuyệt đối* và *số gia*. Bạn cần nhớ sự khác biệt giữa hai nhóm lệnh:

`G90 G00 X0 Y0 Z0` *và* `G91 G00 X0 Y0 Z0`

Hệ điều khiển diễn dịch từng tọa độ X0Y0Z0 một cách khác nhau. Cần nhắc lại, địa chỉ tiếp theo là số không, ví dụ X0, có nghĩa là *vị trí tại điểm quy chiếu chương trình*, nếu chế độ là tuyệt đối, sử dụng lệnh G90. Nếu chế độ là số gia, sử dụng lệnh G91. X0 có nghĩa là *không có chuyển động* trên trục được chuyên biệt.

Hầu hết các máy tiện CNC đều sử dụng trục U và W cho chuyển động số gia (dựa trên các trục X và Y tuyệt đối), với cùng logic ứng dụng nêu trên. Các tọa độ trục tuyệt đối sẽ được diễn dịch là *vị trí dao được lập trình*, các tọa độ số gia nêu rõ *chuyển động dao được lập trình*.

Bạn hãy so sánh hai ví dụ chương trình dưới đây – chúng là như nhau – *đồng nhất* về chuyển động thực của dụng cụ cắt:

```
(---> G28 USED IN THE ABSOLUTE MODE)
G90
...
N12 G01 Z-0.75 F4.0 M08
...
N25 G01 X9.5 Y4.874
N26 G28 Z-0.75 M09 (G28 IN ABSOLUTE MODE)
...


(---> G28 USED IN THE INCREMENTAL MODE)
G90
...
N12 G01 Z-0.75 F4.0 M08
...
N25 G01 X9.5 Y4.874
N26 G91 G28 Z0 M09 (G28 IN INCREMENTAL MODE)
...
```

Phương pháp nào tốt hơn? Do cả hai phương pháp đều cho kết quả như nhau, sự lựa chọn sẽ dựa trên tình huống cụ thể hoặc ý thích cá nhân. Chuyển sang chế độ số gia cũng sẽ có ích, do vị trí dao hiện hành có thể không phải lúc nào cũng biết. Nhược điểm của phương pháp này là G91 chỉ là xác lập tạm thời và phải trở lại chế độ G90, được dùng chủ yếu trong chương trình.

Nếu không trở về chế độ tuyệt đối một cách thích hợp có thể sẽ gây ra các lỗi nghiêm trọng

Chế độ lập trình tuyệt đối chuyên biệt vị trí dao hiện hành từ zero chương trình – điều này được thực hiện vào *mọi thời điểm*. Nhiều ví dụ được giới thiệu ở đây sử dụng chế độ lập trình tuyệt đối và đây là chế độ lập trình tiêu chuẩn đối với hầu hết các chương trình.

Tuy nhiên, cũng có trường hợp trở về zero máy theo chế độ số gia có ưu điểm thực tiễn, khi nhà lập trình *không biết* vị trí dao hiện hành. Tình huống này thường xảy ra khi sử dụng chương trình con, chế độ số gia được dùng lặp lại nhiều lần để chuyển dao đến và vị trí XY khác. Ví dụ, vị trí *chính xác* của dao cắt khi chu kỳ khoan hoàn tất trong block N35 trong ví dụ dưới đây?

```
G90
...
N32 G99 G81 X1.5 Y2.25 R0.1 Z-0.163 F12.0
N33 G91 X0.3874 Y0.6482 L7       (REPEAT 7 TIMES)
N34 G90 G80 Z1.0 M09             (CANCEL CYCLE)
N35 G28 (X???? Y????) Z1.0  (UNKNOWN POSITION)
...
```

Có thật sự cần tìm vị trí tuyệt đối không? Có lẽ là không. Bạn hãy xem một số ví dụ khác. Khi trong chế độ tuyệt đối G90, các giá trị tọa độ trục xác định *vị trí* điểm trung gian. Khi chế độ số gia G91 được lập trình, các giá trị tọa độ xác định khoảng cách thực và chiều của *chuyển động* trung gian. Trong cả hai trường hợp, chuyển động trung gian của dao sẽ được thực thi trước. Sau đó, và chỉ sau đó, mới trở về đúng vị trí quy chiếu zero máy.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Giả sử vị trí dao hiện hành là X5.0 và Y1.0 (vị trí tuyệt đối). Trong chương trình, các giá trị XY của lệnh G28 tiếp sau block vị trí là rất quan trọng:

```
G90
...
N12 G00 X5.0 Y1.0
N13 G28 X0 Y0
...
```

Trong ví dụ này, lệnh G28 chuyên biệt dụng cụ cắt phải đến vị trí zero máy – X0Y0 trong block N13. Do lệnh G28 chỉ liên hệ với zero máy, sẽ có thể có lý khi giả thiết X0Y0 liên hệ với zero máy, thay vì zero (chuẩn) chi tiết. *Điều này là không đúng.*

X0Y0 chỉ quy chiếu *theo điểm, qua đó dao sẽ đến vị trí zero máy*. Điểm này là *vị trí trung gian* đối với lệnh trả về zero máy. Điểm trung gian này được gán các tọa độ liên quan đến chi tiết (trong chế độ tuyệt đối). Dụng cụ cắt sẽ chuyển đến zero chương trình *trước khi* tiếp tục đến zero máy, do đó trong một block có hai chuyển động dao. Điều này dường như không hẳn là chuyển động theo ý đồ ban đầu.

Ví dụ nêu trên có thể thay đổi, để loại bỏ chuyển động trung gian, hoặc chuyển động này được *xác định theo vị trí dao hiện hành*. Chuyển động trung gian về lý thuyết là không thể loại bỏ, nhưng có thể lập trình theo khoảng cách zero (bằng không).

```
G90
...
N12 G00 X5.0 Y1.0
N13 G28 X5.0 Y1.0
...
```

Bằng cách này, điểm trung gian trở thành *vị trí dao hiện hành*, kết quả là chuyển động *trực tiếp* đến zero máy. Lý do là vị trí dao trung gian trùng với vị trí dao hiện hành. Định dạng lập trình này không cần sử dụng các giá trị trục có tính chế độ. Trong chương trình gia công, X5.0Y1.0 trong block N13 phải được lặp lại, trong khi chế độ tuyệt đối G90 vẫn còn hiệu lực.

Trong các trường hợp khi *chưa biết* vị trí dao hiện hành, trả về zero máy được thực hiện trong *chế độ số gia*. Khi đó, cần thay đổi tạm thời sang chế độ số gia và lập trình chuyển động với chiều dài zero (bằng không) cho từng trục được chuyên biệt:

```
G90
...
N12 G00 X5.0 Y1.0
N13 G91 G28 X0 Y0
N14 G90 ...
...
```

Điểm quan trọng ở đây là luôn luôn trở lại chế độ tuyệt đối ngay khi có thể, để tránh diễn dịch sai các dữ liệu chương trình kế tiếp.

Tóm lại, điểm trung gian không thể loại bỏ khỏi block G28/G30. Nếu tình huống yêu cầu trở về zero máy mà không đi qua điểm trung gian riêng rẽ, bạn hãy sử dụng chuyển động dao zero hướng đến điểm trung gian. Phương pháp này tùy thuộc vào lệnh G90 hay lệnh G91 hoạt động vào thời điểm đó.

- Trong chuyển động với chế độ tuyệt đối G90 trở về zero máy, vị trí tọa độ dao hiện hành phải được lặp lại cho từng trục chuyên biệt với lệnh G28
- Trong chuyển động với chế độ số gia G91 trở về zero máy, chuyển động dao hiện hành phải bằng không (zero) cho từng trục chuyên biệt với lệnh G28

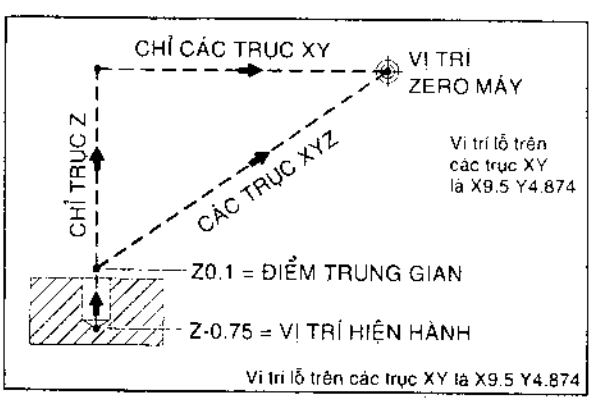
Trở về vị trí chiều sâu Z

Một ví dụ phổ biến về sử dụng vị trí dao trung gian trong block chương trình là từ lỗ sâu hoặc hốc trả về zero máy. Trong ví dụ kế tiếp, chỉ nhằm để giải thích, sử dụng các chuyển động dao đều đặn thay cho chu kỳ khoan, để rút dao ra khỏi lỗ khoan. Trong ví dụ này, vị trí XY hiện hành là X9.5Y4.874 và nguyên công khoan sẽ được mô phỏng trong các block riêng rẽ.

```
...
N21 G90 G00 G54 X9.5 Y4.874 S900 M03
N22 G43 Z0.1 H01 M08
N23 G01 Z-0.45 F10.0
N24 G00 Z-0.43
N25 G01 Z-0.75
...
```

Trong block N25, dao ở đáy lỗ, vị trí dao hiện hành có tọa độ tuyệt đối là X9.5Y4.874 Z-0.75. Toàn bộ sự cắt gọt đã hoàn tất, dao cần trở về zero máy trên cả ba trục. Vì lý do an toàn, trục Z phải chuyển động trước. Có nhiều tùy chọn, trong đó 3 tùy chọn phổ biến là:

- Lùi trục Z phía trên chi tiết trong một block, sau đó trả các trục XYZ về zero máy.



Hình 20.5. *Trả về zero máy từ lỗ sâu – phay.*

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Lùi trục Z về zero máy, sau đó lùi các trục XY trong block kế tiếp.
- Trả các trục XYZ về zero máy một cách trực tiếp từ vị trí dao hiện hành (ở đáy lỗ).

Hình 20.5 minh họa ba tùy chọn này.

➲ Tùy chọn 1

Để lùi trục Z phía trên chi tiết trong một block, sau đó trả các trục XYZ về vị trí zero máy, thường sử dụng:

```
N26 G00 Z0.1 M09
```

Tiếp sau block này phải là sự trở về zero máy, dọc theo trục Z.

```
N27 G00 Z0.1 M05
```

Chương trình đầy đủ đối với *Tùy chọn 1* sẽ là:

```
...
N21 G90 G00 G54 X9.5 Y4.874 S900 M03
N22 G43 Z0.1 H01 M08
N23 G01 Z-0.45 F10.0
N24 G00 Z-0.43
N25 G01 Z-0.75
N26 G00 Z0.1 M09
N27 G28 Z0.1 M05
N28 G28 X9.5 Y4.874
N29 M01
```

➲ Tùy chọn 2

Để lùi hẳn trục Z về zero máy, sau đó lùi các trục XY trong block kế tiếp, biến thể của *Tùy chọn 1*, trước hết lùi trục Z về zero máy:

```
N26 G28 Z-0.75 M09
```

Sau đó trả các trục XY về zero máy:

```
N27 G28 X9.5 Y4.874
```

Chương trình đầy đủ đối với *Tùy chọn 2* sẽ là:

```
...
N21 G90 G00 G54 X9.5 Y4.874 S900 M03
N22 G43 Z0.1 H01 M08
N23 G01 Z-0.45 F10.0
N24 G00 Z-0.43
N25 G01 Z-0.75
N26 G28 Z0.75 M09
N27 G28 X9.5 Y4.874 M05
N28 M01
```

➲ Tùy chọn 3

Để đưa cả ba trục về zero máy trực tiếp từ vị trí dao hiện hành (trong khi dao vẫn ở đáy lỗ) chỉ cần *một* block trả về zero.

```
N26 G28 X9.5 Y4.874 Z0.1 M09
```

Đây là phương pháp lập trình theo thiết kế của các bộ điều khiển Fanuc, tuy một số nhà lập trình có thể không đồng ý với Fanuc. Chương trình đầy đủ với *Tùy chọn 3* như sau:

```
...
N21 G90 G00 G54 X9.5 Y4.874 S900 M03
N22 G43 Z0.1 H01 M08
N23 G01 Z-0.45 F10.0
N24 G00 Z-0.43
N25 G01 Z-0.75 M09
N26 G28 X9.5 Y4.874 Z0.1 M05
N27 M01
```

Chuyển động trở về zero máy thực hiện theo hai bước:

Bước 1: Trục Z chuyển động nhanh đến vị trí Z0.1

Bước 2: Cả ba trục trở về zero máy

Ngoài ra, bạn cần lưu ý sự sắp xếp lại các hàm M09 và M05. Sự tắt chất làm nguội trước là thực tiễn hơn so với dừng trục chính.

So sánh các tùy chọn nêu trên bạn có thể rút ra các kết luận:

- ***TÙY CHỌN 1...***

  ... chỉ an toàn hợp lý, nhưng rất hiệu quả về thời gian chu kỳ, có thể có khả năng có chướng ngại trong chuyển động ba trục đến zero máy.

- ***TÙY CHỌN 2...***

  ... hiệu quả kém hơn so với tùy chọn trước, nhưng là tùy chọn an toàn nhất.

- ***TÙY CHỌN 3...***

  ... hiệu quả cao nhất về thời gian chu kỳ, nhưng sai số vị trí có thể gây ra va chạm.

Trả về các trục cần thiết đối với ATC

Nếu mục đích duy nhất của trả về zero máy là thực hiện sự thay dao tự động, chỉ cần dịch chuyển trên một số trục. Đối với trung tâm gia công đứng, chỉ yêu cầu trục Z để thay dao:

```
G91 G28 Z0 M06
```

Trung tâm gia công ngang chỉ yêu cầu trục Y đạt đến vị trí quy chiếu để thay dao tự động. Để an toàn và thuận tiện, trục Z thường được lập trình, cùng với trục Y, để tránh va chạm với dao kế cận trong hộp dao:

```
G91 G28 Y0 Z0 M06
```

Trong cả hai ví dụ, hàm thay dao M06 sẽ *không* có hiệu lực, cho đến khi đạt được vị trí quy chiếu zero máy. Hàm M06 có thể được lập trình trong block riêng, nếu muốn.

Các trục quay hoặc phân độ cũng có điểm quy chiếu riêng và được dùng với lệnh G28 hoàn toàn như các trục tuyến tính. Ví dụ, trục B sẽ trả về vị trí zero máy trong block:

```
G91 G28 B0
```

Nếu an toàn, trục B có thể được lập trình đồng thời với trục khác:

```
G91 G28 X0 B0
```

Ký hiệu chế độ tuyệt đối tuân theo cùng các quy tắc cho trục quay hoặc phân độ như các trục tuyến tính.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trở về zero đối với máy tiện CNC

Đối với nguyên công tiện CNC, cũng có thể dùng lệnh G28, thường là khi gá lắp. Ứng dụng chung của trả về zero máy cũng được sử dụng, khi ít nhất một trục bắt đầu và kết thúc ở vị trí zero máy. Điều này thường xảy ra với trục X, nhưng ít xảy ra với trục Z, do có thể quá xa trên một số máy tiện cỡ lớn.

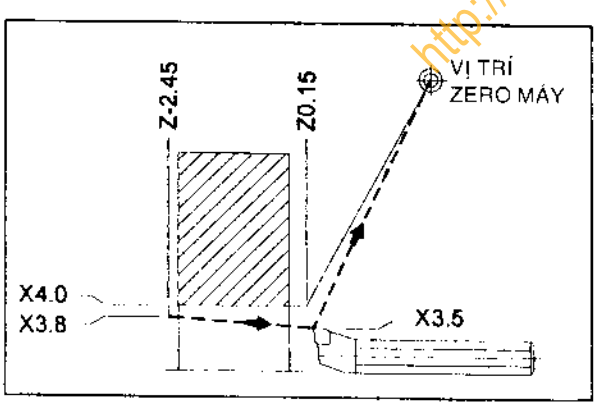
Nói chung, chương trình tiện CNC sẽ được thiết kế sao cho sự gia công chi tiết thứ nhất sẽ bắt đầu từ zero máy, nhưng chi tiết kế tiếp sẽ được gia công từ vị trí thay dao an toàn. Phương pháp này chỉ có tính thực tiễn nếu chương trình sử dụng sự bù hình học thay vì xác lập G50 cũ. Phương pháp trở về zero máy thông dụng nhất trên máy tiện là phương pháp trực tiếp không có điểm trung gian, do không cần G91, sai số ít xảy ra.

```
N78 G28 U0
N79 G28 W0
```

Hai block này sẽ trả dụng cụ cắt trở về zero máy theo chế độ số gia, do đó không áp dụng chuyển động trung gian. Sẽ an toàn hơn khi dịch chuyển trục X trước, sử dụng chế độ số gia U, sau đó là trục Z, sử dụng chế độ số gia W. Nếu vùng làm việc có khoảng trống (chú ýụ động), cả hai trục X và Z đều có thể đồng thời trở về zero máy:

```
N78 G28 U0 W0
```

Hình 20.6 minh họa sự rút dao ra khỏi lỗ, khi hoàn tất gia công.



Hình 20.6. *Trả về zero máy từ lỗ sâu, gia công trên máy tiện.*

Khi sử dụng lệnh đăng ký vị trí G50, xác lập XY phải được biết đối với lệnh này. Trong trường hợp đó, các nguyên tắc lập trình trả về zero máy là rất giống nhau. Giả sử vị trí zero máy có tọa độ X10.0Z3.0, chương trình đối với dao doa có thể viết theo hai cách – một *không* sử dụng lệnh G28 và một *có* sử dụng lệnh G28.

➲ Ví dụ 1

Ví dụ thứ nhất không sử dụng lệnh trả về zero máy G28

```
N1 G20 (EXAMPLE 1)
...
N58 G50 X10.0 Z3.0 S1000 (OLDER METHOD ONLY)
N59 G00 T0300 M42
N60 G96 S400 M03
N61 G00 G41 X4.0 Z0.15 T0303 M08
N62 G01 Z-2.45 F0.012
N63 X3.8 M09
N64 G00 G40 X3.5 Z0.15 M05
N65 X10.0 Z3.0 T0300
N66 M01
```

➲ Ví dụ 2

Ví dụ này sử dụng lệnh trả về zero máy G28, để đạt được cùng vị trí đích:

```
N1 G20 (EXAMPLE 2)
...
N58 G50 X10.0 Z3.0 S1000 (OLDER METHOD ONLY)
N59 G00 T0300 M42
N60 G96 S400 M03
N61 G00 G41 X4.0 Z0.15 T0303 M08
N62 G01 Z-2.45 F0.012
N63 G40 X3.8 M09
N64 G28 X3.5 Z0.15 M05 T0300
N65 M01
```

Hầu hết các nhà lập trình CNC đều cảm thấy thuận tiện hơn với ví dụ thứ nhất và sự tiết kiệm một block chương trình chưa đủ để họ thay đổi phương cách lập trình. Ví dụ 2 có thể được lập trình cả trong chế độ số gia, sử dụng các địa chỉ U và W, nhưng tính thực tiễn không cao.

LỆNH KIỂM TRA VỊ TRÍ TRẢ VỀ

Lệnh chuẩn vị G27 thực hiện chức năng kiểm tra. Mục đích duy nhất là kiểm tra (có ý nghĩa *xác nhận*) vị trí lập trình trong block chứa G27 có ở đúng điểm quy chiếu zero máy hay không. Nếu đúng, đèn báo trên bảng điều khiển đối với từng trục đạt đến vị trí này sẽ sáng. Nếu vị trí đạt được không phải là zero máy, sự xử lý chương trình sẽ tạm dừng với điều kiện lỗi hiển thị trên màn hình sẽ cảnh báo.

Nếu vị trí khởi động dao được lập trình tại quy chiếu zero máy (chuẩn máy), thì nên trở lại vị trí này khi hoàn tất sự gia công với dao đó. Điều này thường được thực hiện trên các máy tiện CNC, nếu sự thay (phân độ) dao thường xảy ra trong cùng một vị trí, dù vị trí đó chưa hẳn là zero máy. Nói chung, vị trí an toàn là gần chi tiết gia công.

Định dạng đối với lệnh G27 là:

```
G27 X.. Y Z..
```

Trong đó phải chuyên biệt ít nhất một trục.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Khi được dùng trong chương trình thông thường, lệnh G27 cắt sẽ tự động chạy nhanh (không cần G00) đến vị trí được chuyên biệt theo các trục trong block G27. Chuyển động này có thể theo chế độ số gia hoặc chế độ tuyệt đối. Chú ý, không sử dụng lệnh G28.

```
N1 G20
N2 G50 X7.85 Z2.0          (OLDER METHOD ONLY)
N3 G00 T0400 M42
N4 G96 S350 M03
N5 G00 G42 X4.125 Z0.1 T0404 M08
N6 G01 Z-1.75 F0.012
N7 U0.2 F0.04
N8 G27 G40 X7.85 Z2.0 T0400 M09
N9 M01
```

Trong ví dụ này, block N8 chứa G27, nhưng không có G00 hoặc G28. Block này ra lệnh cho máy CNC trở về vị trí X7.85Z2.0 và kiểm tra khi đến vị trí đích, xem đó có phải là zero máy theo *mọi trục* (hai trục trong ví dụ này) đã chuyên biệt hay không. Đèn xác nhận sẽ sáng nếu vị trí zero máy được khẳng định. Nếu không, chương trình sẽ dừng lại, cho đến khi khắc phục được nguyên nhân.

Bạn hãy so sánh điểm bắt đầu trong block N2 và điểm kết thúc trong N8. Giả sử vị trí này ở điểm quy chiếu zero máy theo cả hai trục X và Z, ví dụ nêu trên sẽ xác nhận vị trí trong block N8. Giả sử có sai sót nhỏ khi viết N8, giá trị X được nhập là X7.58 thay vì X7.85:

```
N8 G27 G40 X7.58 Z2.0 T0400 M09
```

Trong trường hợp này, hệ điều khiển sẽ đưa ra điều kiện lỗi. Lỗi này hiển thị tự động trên màn hình (cảnh báo). Hệ thống sẽ *không* xử lý phần còn lại của chương trình, cho đến khi lỗi được chỉnh sửa. Đèn báo *Cycle Start* sẽ tắt để tìm *nguyên nhân* của vấn đề. Khi tìm nguyên nhân, bạn hãy kiểm tra *cả hai vị trí*, block vị trí khởi đầu và block vị trí cuối. Lỗi có thể xảy ra ở một trong hai block đó. Ngoài ra, bạn cần lưu ý trục bất kỳ *không* được chuyên biệt trong block sẽ không được kiểm tra vị trí thực của trục đó.

Điểm quan trọng kế tiếp là sự xóa bù bán kính dao và bù dao. Lệnh chuẩn bị G27 cần được lập trình với lệnh G40 và Txx00 có hiệu lực (G49 hoặc H00). Nếu sự bù dao hoặc bù bán kính dao vẫn có hiệu lực, sự kiểm tra sẽ không thể hoàn chỉnh, do điểm quy chiếu dao bị lệch theo giá trị bù.

Dưới đây là cách thức chương trình thứ nhất (ví dụ 1) nêu trên có thể được chỉnh sửa để chấp nhận lệnh G27. Bạn hãy lưu ý G27 sẽ chỉ chuyển động đến các tọa độ đã chuyên biệt, *không* đến điểm trung gian bất kỳ. Block N65 sẽ trở thành block kiểm tra thực sự. Hệ điều khiển sẽ chỉ chuyển các trục máy đến X10.0Y3.0 và kiểm tra (xác định) xem vị trí này có phải là điểm quy chiếu zero máy hay không. Đây là lý do Ví dụ 1 có thể được chỉnh sửa, nhưng không đúng với Ví dụ 2.

```
N1 G20
...
N58 G50 X10.0 Z3.0 S1000 (OLDER METHOD ONLY)
N59 G00 T0300 M42
N60 G96 S400 M03
N61 G00 G41 X4.0 Z0.15 T0303 M08
N62 G01 Z-2.45 F0.012
N63 X3.0 M09
N64 G00 G40 X3.5 Z0.15 M05
N65 G27 X10.0 Z3.0 T0300
N66 M01
```

Sự kiểm tra trả về điểm quy chiếu máy có thể được thực hiện trong chế độ số gia hoặc chế độ tuyệt đối. Nhóm lệnh tuyệt đối trong block N65 (ví dụ nêu trên) có thể được thay bằng chế độ số gia.

```
N65 G27 U6.5 W2.85 T0300
```

Đây là nhược điểm của lệnh này. Cái giá phải trả khi sử dụng lệnh kiểm tra là sự tổn thất thời gian chu kỳ. Do sự giảm tốc độ chuyển động dao được thiết lập trong lệnh này từ bộ điều khiển, có thể cần 1-3 giây để thực thi lệnh G27. Đây có thể là tổn thất khá lớn, nếu có nhiều dao sử dụng kiểm tra G27 trong từng chương trình.

Lệnh G27 ít khi được dùng với xác lập *bù hình học* cho dụng cụ cắt, là phương pháp thông dụng hiện nay. Lệnh G50 đã cũ, không dùng trên các máy tiện CNC hiện đại, nhưng nhiều máy tiện trong công nghiệp vẫn còn cần sử dụng xác lập G50.

TRỞ VỀ TỪ ĐIỂM ZERO MÁY

Lệnh chuẩn bị G29 hoàn toàn ngược với các lệnh G28 và G30. Lệnh G28 tự động đưa dụng cụ cắt trở về vị trí zero máy, lệnh G29 sẽ đưa dao về vị trí *ban đầu* (gốc), *thông qua điểm trung gian*.

Trong lập trình thông dụng, lệnh G29 thường tiếp sau lệnh G28 hoặc G30, các nguyên tắc liên quan đến ký hiệu trục số gia hoặc tuyệt đối vẫn đúng với lệnh G29 hoàn toàn như các lệnh G28 và G30. Mọi trục được lập trình đều chuyển động với tốc độ nhanh đến điểm trung gian trước, được xác định từ block lệnh G28 hoặc G30 đứng trước. Ví,dụ về ứng dụng tiện sẽ minh họa khái niệm này:

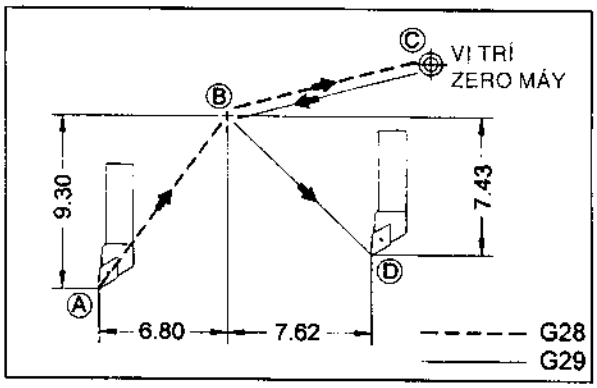
```
...
T0300
...
G28 U5.0 W3.0
G29 U-4.0 W2.375
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lệnh G29 luôn luôn được đưa ra trong chế độ xóa của bù bán kính dao (G40) và các chu kỳ cố định (G80), nếu chúng được dùng trong chương trình. Bạn hãy dùng các mã G xóa tiêu chuẩn, G40 để xóa bù bán kính dao và G80 để xóa chu kỳ cố định, trước khi dùng lệnh G29 trong chương trình.

Phác thảo chuyển động dao được minh họa trên Hình 20.7.



Hình 20.7. Sự trả về tự động từ vị trí zero máy.

Minh họa này cho thấy chuyển động dao từ điểm A đến điểm B, sau đó đến điểm C, trở lại điểm B, và cuối cùng đến điểm D. Điểm A là điểm bắt đầu của chuyển động. B là điểm trung gian, C là điểm quy chiếu zero máy, và D là điểm cuối cần đạt đến, vị trí đích.

Các lệnh chương trình tương đương, bắt đầu ở vị trí dao hiện hành, điểm A, kết quả là quỹ đạo A đến B đến C đến B và đến D là tương đối đơn giản:

```
G28 U18.6 W6.8
...
G29 U-14.86 W7.62
```

Đương nhiên, có thể có hoạt động nào đó được lập trình giữa hai block này, ví dụ, sự thay dao hoặc hoạt động của máy.

Tương tự lệnh G27, nhiều nhà lập trình CNC không ủng hộ lệnh G29. Đây là một trong các lệnh có thể rất hữu ích trong một số trường hợp, nhưng hầu như không cần thiết trong công việc hàng ngày. Tuy nhiên, hiểu về lệnh này là cần thiết, do đôi khi bạn có thể sử dụng chúng trong lập trình.

TRẢ VỀ ZERO MÁY THỨ CẤP

Ngoài lệnh G28 trả về zero máy, các máy CNC còn có lệnh G30. Trong chương này, nhiều ví dụ áp dụng các lệnh G28 và G30 như nhau và đôi khi gộp thành G28/G30 để dễ giải thích. Vậy sự khác biệt giữa chúng là gì và công dụng của lệnh G30 là gì?

Theo định nghĩa, lệnh chuẩn bị G30 là lệnh trả về zero máy đến *vị trí zero máy thứ cấp*. Vị trí này phải có trên máy vào thời điểm đặt mua máy. G30 là đồng nhất với G28 ngoại trừ đây là zero chương trình *thứ cấp*.

Zero chương trình thứ cấp này có thể là điểm quy chiếu thứ hai, ba, thậm chí thứ tư, do nhà chế tạo máy chuyên biệt, không phải mọi máy CNC đều có vị trí quy chiếu zero máy thứ cấp. Vị trí này chỉ dùng cho các mục đích đặc biệt, chủ yếu trên các trung tâm gia công ngang.

Định dạng lập trình đối với lệnh G30 tương tự lệnh G28, có thêm địa chỉ P.

```
G30 P.. X.. Y.. Z..
```

Trong đó:

G30:	Biểu thị chọn vị trí quy chiếu thứ cấp
P:	Có thể là P2, P3, và P4 để xác định vị trí thứ cấp (2-4).
XYZ:	Định nghĩa điểm trung gian (cần chuyên biệt ít nhất một trục).

Công dụng chủ yếu của điểm quy chiếu zero máy thứ cấp trong lập trình CNC là để thay đổi mâm dao. Trong xác lập tham số bộ điều khiển, khoảng cách của điểm quy chiếu thứ cấp được tính từ điểm quy chiếu sơ cấp và thường không thay đổi trong suốt tuổi thọ của máy CNC.

Để phân biệt giữa các vị trí zero máy thứ cấp, địa chỉ P được đưa vào block G30 (không dùng địa chỉ P cho G28). Nếu máy CNC chỉ có một vị trí quy chiếu máy thứ cấp, địa chỉ P thường *không cần* dùng trong chương trình, khi đó, P1 được giả thiết:

```
G30 X.. Y..        là đồng nhất với:
G30 P1 X.. Y..
```

Trong trường hợp này sự xác lập điểm quy chiếu thứ hai là trong phạm vi các tham số của bộ điều khiển. Đối với các khảo sát lập trình khác, lệnh G30 được dùng hoàn toàn như lệnh trả về zero máy G28.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

21 NỘI SUY TUYẾN TÍNH

Nội suy tuyến tính có quan hệ chặt chẽ với chuyển động chạy dao nhanh. Chuyển động dao nhanh nghĩa là chuyển dao từ vị trí này đến vị trí khác mà *không cắt gọt*, còn chế độ nội suy tuyến tính được thiết kế để thực sự *cắt gọt vật liệu*, chẳng hạn phay mặt, gia công lỗ, gia công biên dạng, và nhiều chuyển động cắt gọt khác.

Nội suy tuyến tính được dùng trong lập trình gia công để thực hiện chuyển động cắt gọt thẳng từ điểm bắt đầu cắt đến điểm cuối, hoàn toàn sử dụng *khoảng cách ngắn nhất* đối với quỹ đạo dao cắt. Chuyển động được lập trình trong chế độ nội suy tuyến tính luôn luôn là đường thẳng, nối các điểm bắt đầu và kết thúc biên dạng. Trong chế độ này, dao cắt chuyển động từ vị trí này đến vị trí khác theo khoảng cách ngắn nhất giữa các điểm đó. Đây là tính năng lập trình rất quan trọng, được dùng chủ yếu trong gia công đường bao hoặc biên dạng. Chuyển động góc bất kỳ (vạt góc, côn, góc, nghiêng,..) phải được lập trình trong chế độ này để bảo đảm chính xác. Có thể có ba kiểu chuyển động trong chế độ nội suy tuyến tính.

- Chuyển động ngang ... chỉ trên một trục.
- Chuyển động đứng ... chỉ trên một trục.
- Chuyển động chéo ... nhiều trục.

Thuật ngữ *nội suy tuyến tính* có nghĩa là hệ điều khiển có khả năng tính toán hàng ngàn điểm trung gian giữa điểm đầu và điểm cuối của hành trình cắt. Kết quả tính toán này là quỹ đạo ngắn nhất giữa hai điểm đó. Mọi tính toán đều tự động – hệ điều khiển liên tục xác định các tọa độ và điều chỉnh tốc độ ăn dao cho tất cả các trục cắt gọt, thường là hai hoặc ba trục.

LỆNH TUYẾN TÍNH

G01	Nội suy tuyến tính

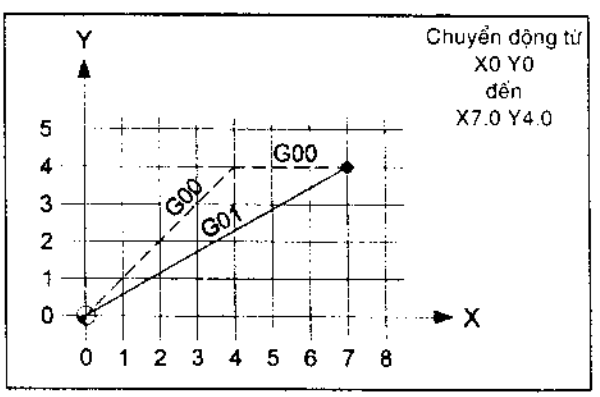
Trong chế độ G01, hàm ăn dao F phải có hiệu lực. Block chương trình thứ nhất khởi đầu chế độ nội suy tuyến tính phải có tốc độ cắt có hiệu lực, nếu không có hiệu lực sự cảnh báo sẽ xuất hiện trong lần chạy thứ nhất, ngay khi cấp điện cho máy. Lệnh G01 và hàm ăn dao F đều có tính chế độ, nghĩa là có thể bỏ qua chúng trong mọi block nội suy tuyến tính kế tiếp, khi đã có G01 và tốc độ cắt không thay đổi. Chỉ sự thay đổi vị trí tọa độ là cần thiết để gán cho trục trong block chương trình. Ngoài chuyển động một trục, chuyển động tuyến tính dọc theo hai hoặc ba trục cũng có thể được lập trình đồng thời.

Khởi đầu và kết thúc chuyển động tuyến tính

Chuyển động tuyến tính, tương tự mọi chuyển động khác trong lập trình CNC, là chuyển động giữa hai điểm đầu và cuối của đường bao. Chuyển động này có vị trí *bắt đầu* và vị trí *kết thúc*. Vị trí *bắt đầu* bất kỳ được gọi là điểm *xuất phát*, vị trí *cuối* được gọi là vị trí *đích*. Khởi đầu chuyển động tuyến tính được xác định từ vị trí dao hiện hành, kết thúc được xác định theo các tọa độ đích của block hiện hành. Có thể dễ dàng nhận thấy vị trí cuối của một chuyển động sẽ trở thành vị trí đầu của chuyển động kế tiếp, khi dao di chuyển dọc theo chi tiết, đi qua mọi điểm thay đổi biên dạng.

Nội suy tuyến tính một trục

Chuyển động dao được lập trình dọc theo một trục luôn luôn là chuyển động song song với trục đó, bất kể chế độ chuyển động. Lập trình trong chế độ G00 hoặc G01 sẽ đưa đến kết quả cùng một vị trí cuối được lập trình, nhưng các tốc độ ăn dao khác nhau và với các kết quả khác nhau. Bạn hãy xem Hình 21.1 để so sánh hai chế độ chuyển động này.



Hình 21.1. So sánh chế độ chạy dao nhanh và chế độ nội suy tuyến tính.

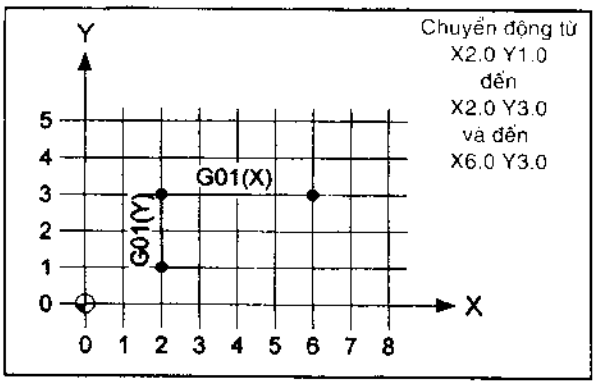
Đối với trung tâm gia công CNC và các máy liên quan, mọi chuyển động dao song song với

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

cạnh bàn máy đều là chuyển động một trục. Trên máy tiện CNC, nhiều nguyên công tiện trong và ngoài, được lập trình theo chuyển động một trục. Trong mọi trường hợp, chuyển động một trục có thể theo trục *đứng* hoặc trục *ngang*, trong phạm vi mặt phẳng (làm việc) hiện hành. Chuyển động một trục không thể là chuyển động chép, do đòi hỏi hai, ba, hoặc nhiều trục.

Hình 21.2 minh họa chuyển động nội suy tuyến tính một trục, một dọc theo trục X và

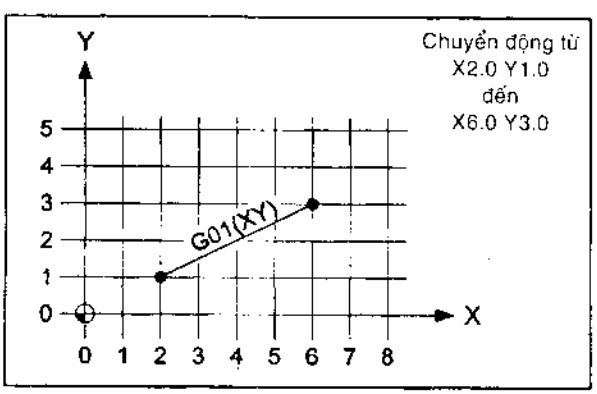


Hình 21.2. *Chuyển động nội suy tuyến tính một trục*

một dọc theo trục Y.

Nội suy tuyến tính hai trục

Chuyển động tuyến tính còn có thể được lập trình đồng thời dọc theo *hai trục*. Đây là tình huống rất phổ biến khi điểm bắt đầu của chuyển động tuyến tính và điểm cuối có ít nhất hai tọa độ khác nhau khi trong chế độ nội suy tuyến tính G01. Kết quả của chuyển động hai trục là chuyển động dao theo đường chéo. Chuyển động này sẽ luôn luôn là khoảng cách ngắn nhất giữa điểm đầu và điểm cuối và kết quả là đường thẳng theo góc do bộ điều khiển tính toán (Hình 21.3).

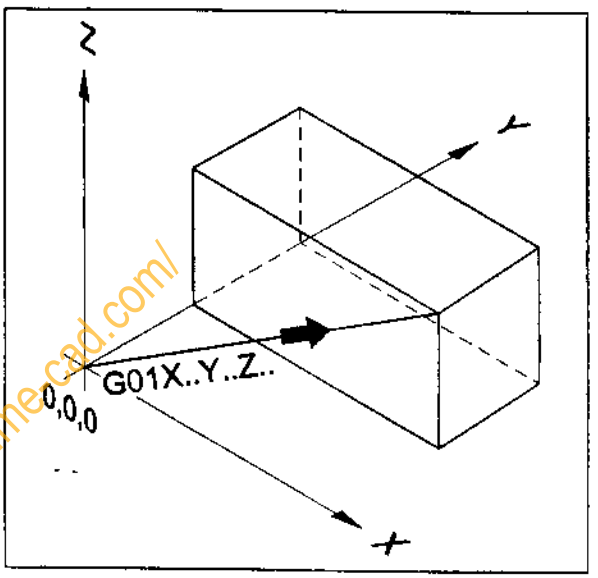


Hình 21.3. *Chuyển động nội suy tuyến tính hai trục.*

Nội suy tuyến tính ba trục

Chuyển động tuyến tính xảy ra đồng thời theo ba trục, được gọi là *nội suy tuyến tính ba trục*. Chuyển động tuyến tính đồng thời theo ba trục là có thể trên hầu như mọi trung tâm gia công CNC. Lập trình chuyển động tuyến tính loại này là không dễ dàng, đặc biệt khi làm việc với các chi tiết phức tạp. Do cần nhiều tính toán phức tạp, lập trình bằng tay là không đủ hiệu quả. Kiểu chương trình loại này cần được thực hiện trên hệ thống lập trình dựa trên máy tính chuyên nghiệp, chẳng hạn *Mastercam*™ (Chương 52).

Chuyển động tuyến tính đồng thời cả ba trục được minh họa trên Hình 21.4.



Hình 21.4. *Chuyển động nội suy tuyến tính đồng thời cả ba trục.*

ĐỊNH DẠNG LẬP TRÌNH

Để lập trình chuyển động dao theo chế độ nội suy tuyến tính, bạn hãy dùng lệnh chuẩn bị G01 cùng với chuyển động dao theo một, hai, hoặc ba trục, và tốc độ cắt (địa chỉ F) thích hợp cho chi tiết gia công:

```
G01 X.. Y.. Z.. F..
```

Mọi mục nhập trong block chuyển động tuyến tính đều có tính chế độ và chỉ cần lập trình nếu chúng là mới hoặc thay đổi. Chỉ lệnh block bị tác động từ sự thay đổi đó mới gộp vào block chương trình.

Tùy theo phương pháp lập trình được chọn, chuyển động nội suy tuyến tính có thể được lập trình trong chế độ tuyệt đối hoặc số gia, sử dụng các lệnh chuẩn bị G90 và G91 cho phay, và các địa chỉ U và W cho nguyên công tiện.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tốc độ cắt thực đối với chuyển động dao xác định có thể được lập trình theo hai chế độ:

- ... theo thời gian — mm/min *hoặc* in/min
- ... theo vòng quay trục chính — mm/rev *hoặc* in /rev

Sự lựa chọn tùy theo kiểu máy và đơn vị kích thước được sử dụng. Nói chung trung tâm gia công CNC, máy khoan, máy phay, máy cắt bằng ngọn lửa, gia công tia lửa điện..., sử dụng tốc độ cắt theo thời gian. Máy tiện CNC thường sử dụng lượng ăn dao theo vòng quay.

Khoảng tốc độ cắt

Từng hệ thống CNC chỉ hỗ trợ tốc độ cắt trong khoảng xác định. Đối với nội suy tuyến tính trong các ứng dụng phay, tốc độ cắt thấp nhất là 0.0001, theo *in/min*, *mm/min* hoặc *deg/min* (độ/phút). Tốc độ cắt thấp nhất đối với nội suy tuyến tính trong nguyên công tiện phụ thuộc vào số gia nhỏ nhất của các trục tọa độ XY. Hai bảng dưới đây trình bày các khoảng tốc độ cắt được hệ thống CNC hỗ trợ. Bảng thứ nhất dùng cho phay, bảng thứ hai dùng cho tiện. Mọi đơn vị dùng trong lập trình gia công đều được giới thiệu.

Bảng 21.1. *Khoảng tốc độ cắt khi phay*

Số gia chuyển động tối thiểu	PHAY
0.001 mm	0.0001-240000.00 mm/min
0.001 độ	0.0001-240000.00 deg/min
.001 inch	.0001-240000.00 in/min

Bảng 21.2. *Khoảng tốc độ cắt khi tiện*

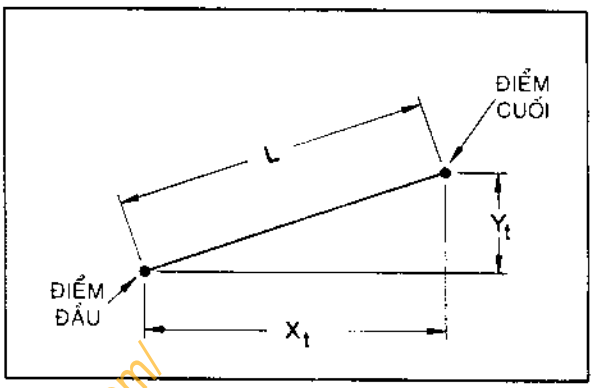
Số gia chuyển động tối thiểu	TIỆN
0.001 mm	0.00001-500.00000 mm/min
0.001 độ	0.00001-500.00000 deg/min
.001 inch	.000001-500.00000 in/min

Có thể tốc độ cắt cực đại là rất cao. Đối với cắt gọt thực tế, điều này là đúng. Tuy nhiên, bạn cần nhớ các khoảng này chỉ liên quan với hệ điều khiển, *không* liên quan với máy. Nhà chế tạo máy luôn luôn giới hạn tốc độ cắt cực đại, phù hợp với các khả năng và thiết kế của máy. Hệ thống điều khiển chỉ cung cấp khoảng lý thuyết, đáp ứng yêu cầu của nhà chế tạo máy hơn là người dùng thực sự. Ý đồ trong trường hợp này là trao cho nhà chế tạo máy tính linh hoạt trong phạm vi các tiến bộ công nghệ hiện đại. Khi công nghệ thay đổi, nhà chế tạo hệ điều khiển sẽ đáp ứng các thay đổi đó bằng cách tăng khoảng tốc độ.

Tốc độ cắt trên trục riêng rẽ

Vấn đề tốc độ cắt thực trên từng trục hầu như là không cần thiết trong lập trình, được trình bày ở đây chỉ nhằm định hướng về toán học. Bạn có thể không cần biết các tính toán này, hệ thống CNC sẽ thực hiện chúng khi cần thiết, một cách chính xác và hoàn toàn tự động. Mặt khác, về lý thuyết bạn cần hiểu về các tính toán đó.

Các tính toán được hệ thống CNC thực hiện, theo các mục nhập được minh họa trên Hình 21.5.



Hình 21.5. *Dữ liệu để tính toán tốc độ ăn dao tuyến tính riêng rẽ.*

Bạn hãy đánh giá ví dụ dưới đây về chuyển động tuyến tính và thử áp dụng các công thức được nêu ở phần dưới:

```
G00 X10.0 Y6.0                (START POINT)
G01 X14.5 Y7.25 F12.0          (END POINT)
```

Chuyển động tuyến tính xảy ra giữa hai điểm, từ điểm đầu ở X10.0Y6.0 đến điểm cuối ở X14.5Y7.25 tốc độ cắt được lập trình 12 in/min là F12.0. Điều đó có nghĩa là hành trình thực dọc theo từng trục là đã biết hoặc có thể tính toán:

$X_t = 14.5 - 10.0 = 4.5$

$Y_t = 7.25 - 6.0 = 1.25$

$Z_t = 0$

Chiều dài L của chuyển động dao toàn phần (Hình 21.5) là chuyển động thực tế, và có thể tính bằng cách dùng *Định lý Pythagore*:

$$L = \sqrt{X_t^2 + Y_t^2 + Z_t^2}$$

Công thức nêu trên hoàn toàn quen thuộc, kết quả sẽ là chiều dài hành trình.

$$L = \sqrt{4.5^2 + 1.25^2} = 4.6703854$$

Hệ điều khiển áp dụng công thức này một cách tự động để tính chuyển động thực theo trục X (4.25) theo trục Y (1.25) cộng với chiều

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

dài của chính chuyển động đó (4.6703854). Từ kết quả này, máy tính sẽ tính các độ dài theo trục X và Y, không có chuyển động theo trục Z.

$$F_x = \frac{X_t}{L \times F}$$

$F_x = 4.5 / 4.6703854 \times 12 = 11.562215$

$$F_y = \frac{Y_t}{L \times F}$$

$F_y = 1.25 / 4.6703854 \times 12 = 3.2117263$

$$F_z = \frac{Z_t}{L \times F}$$

$F_z = 0 / 4.6703854 \times 12 = 0.0$

Trong ví dụ này không có chuyển động trên trục Z. Nếu trục Z là một phần trong chuyển động dao, ví dụ, khi chuyển động đồng thời cả ba trục, quy trình về logic là đồng nhất, do đó phải xét thêm trục Z trong các tính toán.

VÍ DỤ LẬP TRÌNH

Để minh họa ứng dụng thực tiễn của chế độ nội suy tuyến tính trong chương trình CNC, dưới đây là ví dụ đơn giản (Hình 21.6).

➲ **Ví dụ 1**

```
(CLOCKWISE DIRECTION FROM P1)

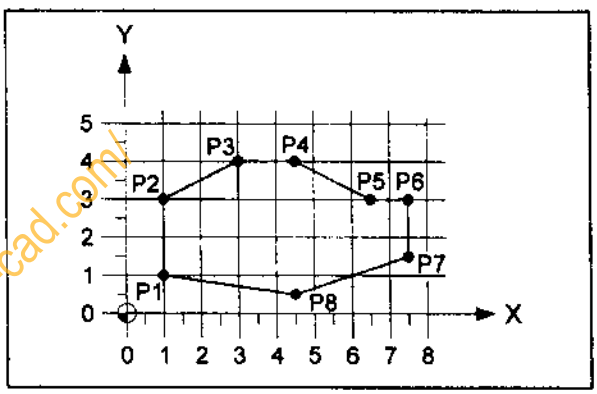
G90 ...                          (ABSOLUTE MODE)
G01 X1.0 Y3.0 F...                    (P1 TO P2)
X3.0 Y4.0                             (P2 TO P3)
X4.5                                  (P3 TO P4)
X6.5 Y3.0                             (P4 TO P5)
X7.5                                  (P5 TO P6)
Y1.5                                  (P6 TO P7)
X4.5 Y0.5                             (P7 TO P8)
X1.0 Y1.0                             (P8 TO P1)
...
```

➲ **Ví dụ 2**

```
(COUNTERCLOCKWISE DIRECTION FROM P1)

G90 ...                          (ABSOLUTE MODE)
G01 X4.5 Y0.5 F...                    (P1 TO P8)
X7.5 Y1.5                             (P8 TO P7)
Y3.0                                  (P7 TO P6)
X6.5                                  (P6 TO P5)
X4.5 Y4.0                             (P5 TO P4)
X3.0                                  (P4 TO P3)
X1.0 Y3.0                             (P3 TO P2)
Y1.0                                  (P2 TO P1)
...
```

Nội suy tuyến tính cung cấp phương tiện lập trình mọi chuyển động trực chuẩn (đứng và ngang) và chuyển động chéo theo khoảng cách tuyến tính ngắn nhất giữa hai điểm. Tốc độ cắt phải được lập trình trong chế độ này, để cắt gọt vật liệu một cách hợp lý. Chú ý, nếu vị trí tọa độ không thay đổi từ điểm này đến điểm kế tiếp, một block đến block kế tiếp sẽ không cần lặp lại trong (các) block tiếp theo.



Hình 21.6. *Ví dụ minh họa nội suy tuyến tính đơn giản*

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

22 HÀM BỎ QUA BLOCK

Hàm *bỏ qua block* là tính năng tiêu chuẩn của hầu như mọi bộ điều khiển CNC. Mục đích chính là cung cấp tính linh hoạt cao hơn cho nhà lập trình khi thiết kế chương trình để không xảy ra quá *hai khả năng va chạm*. Nếu không có hàm bỏ qua block, phương pháp duy nhất là viết hai chương trình gia công riêng rẽ, mỗi chương trình chỉ dành cho một khả năng.

CÁC ỨNG DỤNG CƠ BẢN

Để hiểu bản chất của hai khả năng va chạm, bạn hãy xét ứng dụng lập trình dưới đây. Yêu cầu là viết chương trình cắt gọt bề mặt. Vấn đề là phôi gia công cung cấp cho máy CNC có kích thước không đồng nhất. Một số phôi có kích cỡ hơi nhỏ hơn chỉ cần một lần gia công cắt gọt, số khác lớn hơn đòi hỏi hai lần cắt. Đây là điều bình thường trong xưởng cơ khí, cần được xử lý một cách hiệu quả. Bạn có thể viết hai chương trình nhưng sẽ không hiệu quả bằng một chương trình sử dụng *hàm bỏ qua block*.

Điều này minh họa tình huống hai lựa chọn mâu thuẫn được yêu cầu *đồng thời* trong một chương trình. Giải pháp dễ thấy nhất là chuẩn bị hai chương trình cho từng lựa chọn. Nhiệm vụ này dễ thực hiện nhưng rất nhàm chán, tốn thời gian và không hiệu quả. Giải pháp thứ hai là viết *một chương trình* với các chuyển động dao để gia công cắt cho *cả hai khả năng*. Để tránh chạy dao vô ích cho các chi tiết chỉ yêu cầu một lần cắt gọt, hàm bỏ qua block được đưa vào chương trình và áp dụng cho tất cả các block liên quan với lần cắt gọt *thứ nhất*. Lần cắt gọt "thứ hai" sẽ áp dụng cho mọi chi tiết.

Ứng dụng tiếp theo của hàm bỏ qua block là sự chuyển đổi trạng thái ON/OFF có chọn lọc, chẳng hạn hàm chất làm nguội, dừng chương trình có lựa chọn, xác lập lại chương trình. Hàm này còn hữu dụng khi cần bỏ qua sự thao tác chương trình xác định, áp dụng hoặc không áp dụng dao đã chọn cho gia công chi tiết. Quyết định lập trình bất kỳ yêu cầu lựa chọn từ *hai* tùy chọn được xác định trước thường cho phép sử dụng hàm bỏ qua block.

KÝ HIỆU BỎ QUA BLOCK

Bạn cần dùng ký hiệu lập trình đặc biệt để xác nhận hàm bỏ qua block trong chương trình. Ký hiệu này là dấu nghiêng phải [/]. Hệ thống sẽ công nhận dấu này là mã để bỏ qua block. Đối với hầu hết các ứng dụng lập trình CNC, dấu nghiên này là ký tự *thứ nhất* trong block.

➲ **Ví dụ 1**

```
N1 ...                          (ALWAYS PROCESSED)
N2 ...                          (ALWAYS PROCESSED)
N3 ...                          (ALWAYS PROCESSED)
/ N4 ...      (PROCESSED IF BLOCK SKIP IS OFF)
/ N5 ...      (PROCESSED IF BLOCK SKIP IS OFF)
/ N6 ...      (PROCESSED IF BLOCK SKIP IS OFF)
N7 ...                          (ALWAYS PROCESSED)
N8 ...                          (ALWAYS PROCESSED)
```

Trên *một số* hệ điều khiển, mã bỏ qua block còn có thể được sử dụng một cách chọn lọc cho các địa chỉ *bên trong* block, thay vì ở đầu block. Bạn hãy xem sổ tay hướng dẫn để biết kỹ thuật này có sử dụng được hay không?

➲ **Ví dụ 2**

```
N6 ...
N7 G00 X50.0 / M08
N8 G01 ...
.
```

Nếu hệ điều khiển cho phép bỏ qua block *bên trong* block lập trình, mọi lệnh *đứng trước* dấu nghiêng đều được thực thi, bất kể xác lập chuyển đổi bỏ qua block. Nếu hàm bỏ qua block hoạt động (ON), chỉ các lệnh *tiếp sau* mã dấu nghiêng bị bỏ qua. Trong *Ví dụ 2*, hàm chất làm nguội M08 (block 07) sẽ bị bỏ qua. Nếu hàm bỏ qua block không hoạt động (OFF), toàn bộ block trong *Ví dụ 2* sẽ được thực thi, kể cả hàm chất làm nguội.

XÁC LẬP BỘ ĐIỀU KHIỂN

Bất kể vị trí dấu nghiêng trong block, chương trình sẽ được xử lý theo hai cách, hoặc thực thi toàn bộ, hoặc các lệnh sau dấu nghiêng sẽ bị bỏ qua. Quyết định về sử dụng hay không sử dụng hàm bỏ qua block được thực hiện trong quá trình gia công thực tế, tùy thuộc vào người vận hành và kiểu gia công. Đối với mục đích này, nút nhấn, hoặc công tắc chuyển đổi, hoặc sự lựa chọn khoản mục menu được cung cấp trên khoảng điều khiển của máy CNC. Sự lựa chọn hàm bỏ qua block có thể là *hoạt động* (ON) hoặc *không hoạt động* (OFF).

Hầu hết các chương trình đều không đòi hỏi các mã bỏ qua block. Trong các trường hợp đó, chế độ xác lập đối với hàm bỏ qua block trên

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

bảng điều khiển là không cần thiết, tuy nhiên bạn vẫn nên dùng chế độ OFF. Xác lập công tắc này trở nên *rất* quan trọng, nếu chương trình có ít nhất *một* block chứa ký hiệu dấu nghiêng. Xác lập hoạt động ON cho phép bỏ qua mọi lệnh trong block tiếp sau dấu nghiêng trong khi thực thi chương trình. Xác lập không hoạt động OFF cho phép bộ điều khiển bỏ qua mã dấu nghiêng và xử lý tất cả các lệnh trong chương trình.

Hàm bỏ qua block xác lập theo vị trí ON có nghĩa là "bỏ qua mọi lệnh block đứng sau dấu nghiêng"

Hàm bỏ qua block xác lập theo vị trí OFF có nghĩa là "thực thi tất cả các lệnh block"

Trong ví dụ nêu trên, nội dung của các block N4, N5, và N6 sẽ bị bỏ qua, nếu hàm bỏ qua block là ON. Chúng sẽ được xử lý, nếu xác lập công tắc là OFF. Ví dụ 2, chứa dấu nghiêng trong block N7. Ký hiệu dấu nghiêng đứng trước hàm M08 (chất làm nguội ON). Nếu công tắc hàm bỏ qua block là ON, chất làm nguội sẽ bị bỏ qua; nếu là OFF, hàm làm nguội sẽ hoạt động. Ứng dụng này có thể hữu dụng trong chế độ chạy khô, bỏ qua sự làm nguội trong khi kiểm chứng chương trình.

Một số bộ điều khiển chỉ cho phép đặt mã dấu nghiêng ở vị trí ký tự thứ nhất trong block: /N

BỎ QUA BLOCK VÀ CÁC LỆNH CHẾ ĐỘ

Để hiểu hoạt động của các giá trị chế độ với các block bị bỏ qua, bạn cần nhớ các lệnh chế độ có thể được chuyên biệt chỉ một lần trong chương trình, trong block nơi chúng xuất hiện lần thứ nhất. Các lệnh chế độ không lặp lại trong các block kế tiếp, nếu chúng vẫn không thay đổi.

Trong các chương trình hoàn toàn không sử dụng hàm bỏ qua block, chương trình được thực thi lần lượt qua từng block. Khi sử dụng hàm bỏ qua block, bạn cần đặc biệt chú ý các lệnh *chế độ*. Bạn cần nhớ, lệnh được thiết lập trong block sử dụng mã dấu nghiêng không phải lúc nào cũng có hiệu lực do phụ thuộc vào xác lập của công tắc bỏ qua block. Lệnh chế độ bất kỳ, được chuyển từ phần có dấu nghiêng đến phần không có dấu nghiêng, có thể bị mất, nếu hàm bỏ qua block được sử dụng. Sự tổn thất các lệnh chế độ khi lập trình hàm bỏ qua block có thể đưa đến nhiều hậu quả xấu.

Giải pháp lập trình đơn giản là tránh xảy ra vấn đề này. Bạn hãy *lặp lại* mọi lệnh chế độ trong phần chương trình không bị ảnh hưởng của hàm bỏ qua block.

Bạn hãy so sánh hai ví dụ sau đây:

➲ Ví dụ A - Các lệnh chế độ *không được* lặp lại

```
N5 G00 X10.0 Y5.0 Z2.0
/ N6 G01 Z0.1 F30.0 M08
N7 Z-1.0 F12.0              (G01 AND M08 MISSING)
N8 ...
```

➲ Ví dụ B - Các lệnh chế độ *được* lặp lại

```
N5 G00 X10.0 Y5.0 Z2.0
/ N6 G01 Z0.1 F30.0 M08
N7 G01 Z-1.0 F12.0 M08
N8 ...
```

Trong cả hai ví dụ A và B, block chương trình chứa dấu nghiêng biểu thị vị trí trung gian trên trục Z là X.01. Vị trí này có thể chỉ cần thiết trong một số trường hợp trong khi gia công, người vận hành sẽ quyết định có sử dụng hay không và thời điểm sử dụng.

Block cơ bản, trong cả hai ví dụ là N6, chứa các hàm chế độ. Các lệnh G01, Z01, F30.0 và M08 sẽ luôn luôn có hiệu lực, trừ khi chúng bị xóa hoặc thay đổi trong block bất kỳ phía sau. Từ block N7 dường như vị trí tọa độ Z và giá trị tốc độ cắt thay đổi. Tuy nhiên, các lệnh G01 và M08 không được lặp lại trong ví dụ A, sẽ *không* hoạt động nếu công tắc bỏ qua block được xác lập là ON.

Cả hai ví dụ A và B sẽ có cùng kết quả chỉ khi hàm bỏ qua block không hoạt động (OFF). Hệ điều khiển sẽ thực thi các lệnh trong *mọi* block, theo đúng thứ tự lập trình.

Hai ví dụ nêu trên sẽ có kết quả khác nhau. Nếu hàm bỏ qua block là hoạt động (ON), các lệnh block sau dấu nghiêng sẽ *không* được xử lý. Ví dụ A dưới đây sẽ có kết quả không thể chấp nhận, do có khả năng xảy ra va chạm. Ví dụ B sử dụng phương pháp tinh tế hơn. Dưới đây là các kết quả khi bỏ qua block N6.

➲ Ví dụ A – Các lệnh chế độ *không được* lặp lại

```
N5 G00 X10.0 Y5.0 Z2.0          (RAPID MOTION)
N7 Z-1.0 F12.0                  (RAPID MOTION)
N8 ...
```

➲ Ví dụ B – Các lệnh chế độ *được* lặp lại

```
N5 G00 X10.0 Y5.0 Z2.0          (RAPID MOTION)
N7 G01 Z-1.0 F12.0 M08      (FEEDRATE MOTION)
N8 ...
```

Bạn hãy lưu ý chuyển động tuyến tính G01, tốc độ cắt F30.0 và chất làm nguội M08 bị bỏ qua trong ví dụ A. Các trục X và Y không được cập nhật do đó sẽ không thay đổi. Kết luận là ví dụ A sẽ đưa đến chuyển động nhanh theo trục Z trong hai block kế tiếp, có thể gây ra tình huống nguy hiểm. Trong ví dụ B, lập trình lặp lại tất cả các lệnh G01, F12.0, M08 bảo đảm chương trình chạy đúng theo yêu cầu. Phần kế tiếp sẽ trình bày các nguyên tắc thiết kế chương trình đối với nhiều ứng dụng thực tiễn.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tóm lại, nguyên tắc cơ bản về lập trình CNC với các block sử dụng hàm bỏ qua block, là:

Luôn luôn lập trình *tất cả các lệnh* dù có thể phải lặp lại một số giá trị chương trình và các lệnh cần được bảo toàn.

Ký hiệu dấu nghiêng có thể được đặt vào chương trình *sau khi* chương trình được thiết kế cho *cả hai* tùy chọn. Chỉ đặt dấu nghiêng trong các block xác định sự bỏ qua tùy chọn đối với mọi block chương trình đã chọn. *Luôn luôn kiểm tra chương trình.*

Chương trình CNC bất kỳ chứa hàm bỏ qua block phải được kiểm tra ít nhất hai lần.

Kết quả của các kiểm tra phải *luôn luôn* thỏa mãn, cả hai trường hợp hàm bỏ qua block có và không có hiệu lực. Nếu phát hiện lỗi, *dù là nhỏ*, cũng cần phải xử lý ngay. Sau khi sửa lỗi, bạn hãy kiểm tra lại chương trình ít nhất *hai lần* cho cả hai trường hợp. Lý do để kiểm tra hai lần là sự chỉnh sửa lỗi trong một kiểu xử lý (có hoặc không có hàm bỏ qua block) có thể gây ra lỗi khác trong kiểu xử lý kia.

VÍ DỤ LẬP TRÌNH

Hàm bỏ qua block tuy đơn giản, thường không được chú ý, nhưng là công cụ lập trình rất hữu ích. Nhiều chương trình trở nên linh hoạt hơn khi sử dụng tính năng này một cách sáng tạo. Kiểu công việc và tính sáng tạo là các tiêu chuẩn để áp dụng thành công tính năng bỏ qua block trong lập trình. Trong các ví dụ dưới đây, sẽ minh họa một số ứng dụng thực tiễn về hàm bỏ qua block. Bạn có thể coi các ví dụ này là điểm khởi đầu trong thiết kế chương trình tổng quát hoặc các ứng dụng gia công tương tự.

Gia công cắt gọt thô

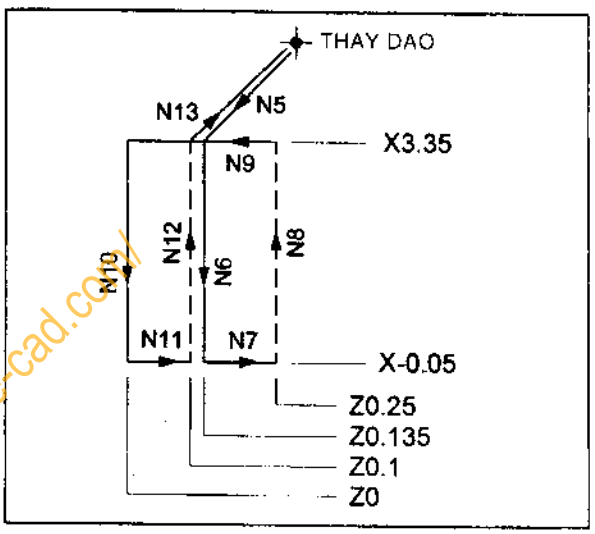
Gia công thô thường được thực hiện với tốc độ cắt lớn, chuẩn bị cho quá trình gia công tinh. Khi gia công các hình dạng không đều (vật đúc, vật rèn,...) hoặc cắt gọt thô trên máy tiện, có thể hơi khó xác định số lần cắt. Ví dụ, một số vật đúc chỉ có lượng dư gia công nhỏ, có lẽ một lần cắt thô là đủ. Các vật đúc khác (cùng loại) có thể lớn hơn, cần đến hai đường cắt thô.

Nếu chương trình được thiết kế chỉ với *một* đường cắt thô, các vấn đề có thể phát sinh khi gia công phôi có lượng dư lớn. Lập trình *hai* đường cắt cho tất cả các chi tiết sẽ làm cho chương trình an toàn hơn, nhưng sẽ không hiệu quả đối với các phôi có lượng dư nhỏ, do có các chuyển động dao hoàn toàn không cắt gọt.

Ví dụ: Gia công mặt các phôi có lượng dư khác nhau

Cắt gọt bề mặt phôi có kích thước khác nhau là vấn đề phổ biến trong xưởng cơ khí. Giải pháp thích hợp là đồng nhất đối với phay và tiện. Chương trình phải có các chuyển động dao cho *hai* lần cắt và hàm bỏ qua block được dùng trong mọi block có liên quan với đường cắt *thứ nhất*.

Dưới đây là ví dụ về chương trình tiện để gia công bề mặt (vạt mặt) khi phôi có lượng dư trong khoảng 0.08 (2 mm) đến 0.275 (7 mm). Sau khi xem xét vài tùy chọn gia công, nhà lập trình quyết định phôi có lượng dư lớn nhất có thể gia công với một lần cắt gọt là 0.135 (3.5 mm) (Hình 22.1).



Hình 22.1. Phôi có lượng dư khác nhau khi cắt gọt trên máy tiện – Chương trình O2201

```
O2201 (TURNING)
(VARIABLE FACE STOCK)
N1 G20 G40 G99
N2 G50 S2000
N3 G00 T0200 M42
N4 G96 S400 M03
N5 G41 X3.35 Z0.135 T0202 M08
/ N6 G01 X-0.05 F0.01
/ N7 G00 Z0.25
/ N8 X3.35
N9 G01 Z0 F0.05
N10 X-0.05 F0.01
N11 G00 Z0.1
N12 X3.5
N13 G40 X12.0 Z2.0 T0200
N14 M30
%
```

Block N5 chứa lệnh khởi đầu đưa dao đến gần phôi. Ba block kế tiếp đều có dấu nghiêng đứng trước. Trong N6, dao sẽ cắt gọt bề mặt (vạt mặt) tại Z0.135, N7 đưa dao ra xa bề mặt, block N8 là chuyển động nhanh trở lại đường kính ban đầu. Sau N8 sẽ không có block cần bỏ qua. Block N9 chứa tốc độ cắt mặt đầu Z0,

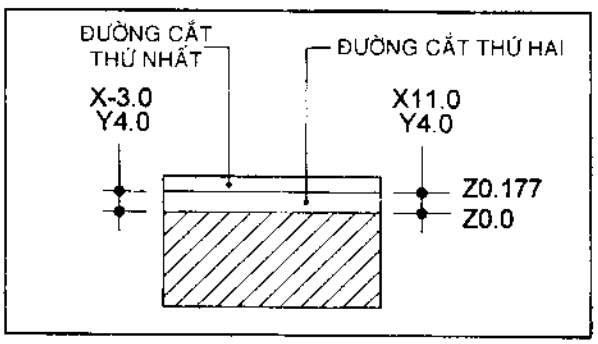
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

block N9 là chuyển động cắt bề mặt. N11 là chuyển động chạy dao ra xa, tiếp theo là các block cuối tiêu chuẩn.

Bạn hãy đánh giá ví dụ này *hai lần* để biết các chuyển động xảy ra. Trong lần đánh giá *thứ nhất*, bạn hãy đọc *tất cả* các block nhưng không xét đến hàm bỏ qua block. Lần đánh giá *thứ hai*, bạn hãy bỏ qua tất cả các block có chứa mã dấu nghiêng. Sẽ có kết quả đồng nhất khi so sánh với lần đánh giá thứ nhất. Khác biệt duy nhất là số lần cắt thực sự chỉ là *một* thay vì hai. Trong nguyên công phay, quy trình cũng tương tự.

Ví dụ về nguyên công phay sử dụng dao phay mặt đầu Ø5. Lượng dư gia công thay đổi trong khoảng 0.125 – 0.315. Chiều sâu cắt hợp lý lớn nhất được chọn là 0.177 (4.5 mm) (Hình 22.2).



Hình 22.2. *Lượng dư khi phay – chương trình O2202*

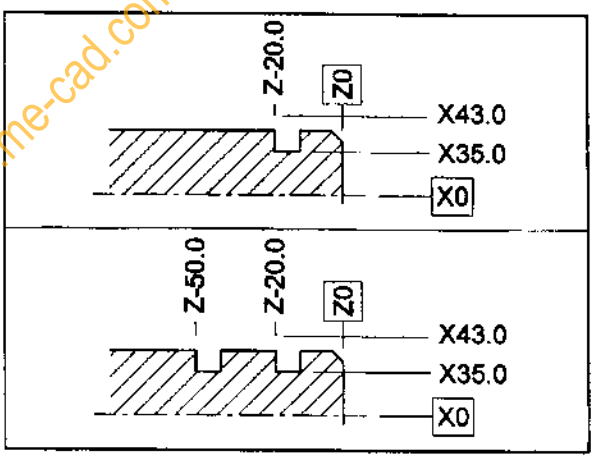
```
O2202 (MILLING)
(VARIABLE FACE STOCK)
N1 G20
N2 G17 G40 G49 G80
N3 G90 G00 G54 X11.0 Y4.0
N4 G43 Z1.0 5550 M03 H01
N5 G01 Z0.177 F15.0 M08
/ N6 X-3.0 F18.0
/ N7 Z0.375
/ N8 G00 X11.0
N9 G01 Z0
N10 X-3.0 F18.0
N11 G00 Z1.0 M09
N12 G28 X-3.0 Y4.0 Z1.0 M13 M30
%
```

Block N5 trong ví dụ này đưa dao đến trục Z cho lần cắt gọt thứ nhất ở mức Z0.177 có thể bỏ qua ba block kế tiếp nếu cần thiết. Trong block N6, dao phay sẽ cắt gọt bề mặt ở vị trí Z0.177, N7 là chuyển động dao ra xa sau khi phay, N8 trả dao về vị trí X ban đầu. Sau N8 sẽ không có các block cần bỏ qua. Block N9 không cần tốc độ cắt, đó có thể là F15.0 *hoặc* F18.0 tùy theo block N6 đến N8 có bị bỏ qua hay không. Tốc độ cắt là rất quan trọng trong N10. Sự lặp lại này bảo đảm tốc độ cắt cần thiết ở đúng block, khi thực sự xảy ra sự cắt gọt.

Cả hai ví dụ tiện và phay đều cung cấp kiến thức cơ bản về *tính logic* được dùng trong thiết kế chương trình sử dụng hàm bỏ qua block. Phương pháp này cũng được áp dụng cho nhiều lần cắt gọt và cho các nguyên công khác.

Thay đổi biên dạng gia công

Ứng dụng thứ hai, trong đó có thể sử dụng hàm bỏ qua block một cách hiệu quả, là lập trình theo nhóm đơn giản. Thuật ngữ *lập trình theo nhóm* có nghĩa là tình huống lập trình trong đó có thể có sự khác biệt nhỏ về thiết kế giữa hai hoặc nhiều sản phẩm. Sự khác biệt nhỏ giữa các chi tiết *tương tự nhau* thường cho phép sử dụng hàm bỏ qua block. Sự khác biệt nhỏ giữa các bản vẽ có thể được giải quyết trong một chương trình sử dụng hàm bỏ qua block. Hai ví dụ dưới đây sẽ minh họa các khả năng lập trình *thay đổi quỹ đạo chạy dao*. Trong một ví dụ, sẽ nhấn mạnh vị trí gia công được bỏ qua. Trong ví dụ thứ hai, sẽ nhấn mạnh sự thay đổi biên dạng gia công. Cả hai ví dụ đều dùng đơn vị hệ mét và minh họa nguyên công cắt rãnh đơn giản. Trong ví dụ về tiện, Hình 22.3 minh họa chi tiết cần gia công với chương trình O2203.



Hình 22.3. *Biên dạng gia công thay đổi – tiện rãnh*

Hình trên minh họa kết quả với hàm bỏ qua block xác lập là ON, hình dưới là kết quả với hàm bỏ qua block là OFF, sử dụng *cùng một* chương trình.

```
O2203 (LATHE EXAMPLE)
N1 G21
...
N12 G50 S1800
N13 G00 T0600 M42
N14 G96 S100 M03
N15 X43.0 Z-20.0 T0606 M08
N16 G01 X35.0 F0.13
N17 G00 X43.0
/ N18 Z-50.0
/ N19 G01 X35.0
/ N20 G00 X43.0
N21 X400.0 Z45.0 T0600 M01
```

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

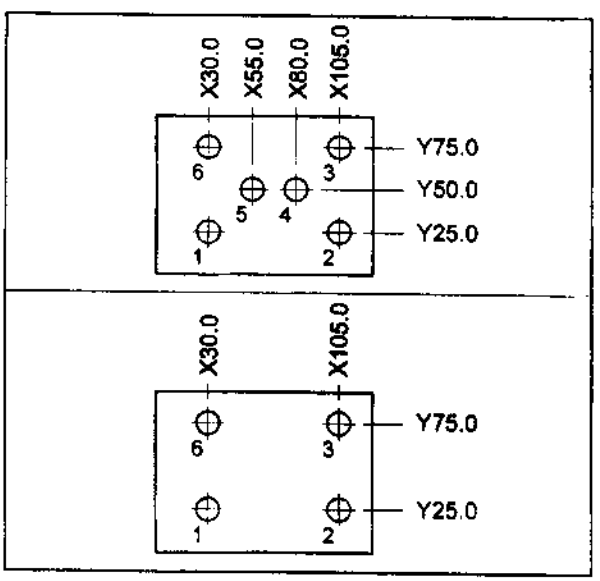
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chương trình O2203 là đơn giản cho hai chi tiết có các đặc tính tương tự nhau: chi tiết thứ nhất có một rãnh, chi tiết thứ hai có hai rãnh trên cùng đường kính. Trong ví dụ này, hai rãnh là đồng nhất, chúng có cùng chiều rộng và chiều sâu, được gia công với cùng một dao cắt. Sự khác biệt duy nhất giữa hai chi tiết là số rãnh và vị trí rãnh thứ hai. Sự gia công chi tiết này đòi hỏi xác lập hàm bỏ qua block là ON hoặc OFF, tùy theo rãnh được gia công.

Bạn hãy đánh giá các block quan trọng trong chương trình này. Block N15 là chuyển động dao ban đầu để đến vị trí khởi đầu của rãnh thứ nhất ở Z-20.0. Trong hai block kế tiếp, N16 và N17, rãnh sẽ được cắt và dao lùi ra ngoài đường kính. Ba block tiếp theo sẽ cắt rãnh thứ hai, nếu có yêu cầu. Đây là lý do để sử dụng mã bỏ qua block. Trong N18, dao di chuyển đến vị trí khởi đầu của rãnh 2 từ Z-50.0, trong block N19 rãnh được gia công. Trong N20, dao lùi ra khỏi rãnh đến vị trí chờ.

Ví dụ về nguyên công phay nêu trên Hình 22.4, cũng theo hệ mét, được biểu thị bằng chương trình O2204. Chương trình này cho phép gia công hai sơ đồ tương tự nhau có 4 lỗ như nhau trên cả hai chi tiết, còn chi tiết thứ hai không có hai lỗ. Đây là ví dụ tốt về chương trình các chi tiết tương tự nhau, sử dụng bỏ qua block.

Cả hai biến thể của chương trình O2204 đều gia công sơ đồ với 6 và 4 lỗ. Hàm bỏ qua block được sử dụng để một chương trình có thể gia công *cả hai* sơ đồ lỗ. Phần trên của Hình 22.4 minh họa sơ đồ lỗ khi hàm bỏ qua block được xác lập OFF, phần dưới minh họa sơ đồ lỗ khi chế độ bỏ qua block có xác lập ON.

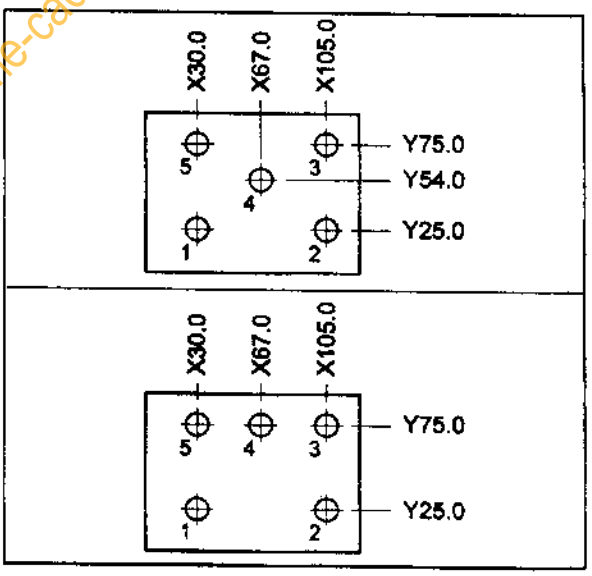


Hình 22.4. *Chương trình O2204 gia công các sơ đồ khác nhau, khoan lỗ trên máy phay – kết quả với sự bỏ qua block OFF (trên) và ON (dưới).*

```
O2204 (MILLING EXAMPLE)
N1 G21
...
N16 G90 G00 G54 X30.0 125.0 M08
N17 G43 Z25.0 S1200 M03 H04
N18 G99 G81 R2.5 Z-4.0 F100.0 (HOLE 1)
N19 X105.0 (HOLE 2)
N20 Y75.0 (HOLE 3)
/ N21 X80.0 Y50.0 (HOLE 4)
/ N22 X55.0 (HOLE 5)
N23 G98 X30.0 Y75.0 (HOLE 6)
N24 G80 G28 X30.0 Y75.0 Z25.0
N25 M01
```

Các block N18 đến N20 sẽ khoan các lỗ 1, 2, và 3. Lỗ 4 trong N21 và lỗ 5 trong N22 sẽ được khoan nếu hàm bỏ qua block là không hoạt động (OFF), nhưng sẽ không được gia công khi hàm bỏ qua block hoạt động (ON). Block N23 luôn luôn khoan lỗ số 6.

Biến thể của ứng dụng này là chương trình O2205. Ở đây có *năm* vị trí lỗ, nhưng hàm bỏ qua block được dùng *bên trong* một block để chỉ điều khiển vị trí Y của lỗi. Phần trên của Hình 22.5 là sơ đồ lỗ khi hàm bỏ qua block là OFF, còn phần dưới là sơ đồ lỗ khi hàm này là ON. Lỗ giữa sẽ có vị trí trục Z khác, tùy theo xác lập của hàm bỏ qua block tại máy công cụ.



Hình 22.5. *Chương trình O2205 gia công các sơ đồ lỗ khác nhau. Kết quả của sự bỏ qua block OFF (trên) và ON (dưới)*

```
O2205 (MILLING EXAMPLE)
N1 G21
...
N16 G90 G00 G54 X30.0 125.0 M08
N17 G43 Z25.0 S1200 M03 H04
N18 G99 G81 R2.5 Z-4.0 F100.0          (HOLE 1)
N19 X105.0                             (HOLE 2)
N20 Y75.0                              (HOLE 3)
N21 X67.0 / Y54.0                      (HOLE 4)
N22 G98 X30.0 Y75.0                    (HOLE 5)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N23 G80 G28 X3.0 Y75.0 Z25.0
N24 M01
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lỗ 4 trong block N21 sẽ được khoan ở vị trí X67.0Y25.0, nếu chế độ bỏ qua block là ON. Địa chỉ Y54.0 trong block N21 sẽ không được xử lý. Nếu chế độ bỏ qua block là OFF, lỗ 4 sẽ được khoan ở tọa độ X67.0Y54.0. Trong trường hợp đó, vị trí Y75.0 từ block N20 sẽ bị bỏ qua. Để bảo đảm khoan đúng vị trí 5, cần viết lại tọa độ Y75.0 trong N22. Nếu không viết tọa độ này, Y54.0 từ block N22 sẽ được thực hiện trong chế độ OFF bỏ qua block.

Sử dụng tính năng bỏ qua block là cách đơn giản nhất để thiết kế nhóm các chi tiết tương tự nhau. Các ứng dụng bị giới hạn với hàm bỏ qua block, nhưng chúng cung cấp cơ sở cho kỹ thuật lập trình mạnh và tính sáng tạo logic.

Cắt gọt thử để đo

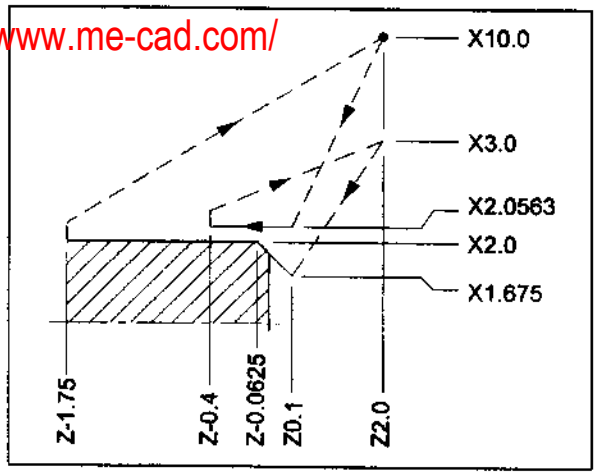
Ứng dụng kế tiếp của hàm bỏ qua block là cung cấp cho người vận hành máy phương tiện để đo chi tiết *trước khi* gia công tinh hoặc gia công hoàn tất. Do sự không hoàn thiện kích thước của dao cắt kết hợp với các yếu tố khác, chi tiết hoàn tất có thể hơi lệch ra ngoài khoảng dung sai cho phép.

Phương pháp dưới đây rất hữu ích để lập trình các chi tiết đòi hỏi dung sai rất cao. Đây còn là phương pháp hữu dụng cho các chi tiết có hình dạng khó đo đạc sau khi hoàn tất gia công, ví dụ các dạng cong. Phương pháp này còn được dùng cho các chi tiết khi thời gian chu kỳ của từng dụng cụ cắt tương đối dài và mọi giá trị bù dao cần được tinh chỉnh *trước khi* gia công sản xuất thực sự.

Phương pháp lập trình này rất hiệu quả, có khả năng loại bỏ sự cắt lại, tăng độ bóng bề mặt và giảm hoặc tránh tạo ra phế phẩm. Trong các trường hợp đó, bạn có thể dùng phương pháp lập trình *cắt gọt thử* sử dụng hàm bỏ qua block. Xác lập chế độ bỏ qua block là OFF, người vận hành máy kiểm tra kích thước gia công thử, điều chỉnh các giá trị bù, nếu cần thiết, và tiếp tục gia công với xác lập bỏ qua block là ON.

Các khái niệm được dùng trong ví dụ O2206 có thể áp dụng cho cả tiện và phay (Hình 22.6).

```
O02206
(TRIAL CUT - LATHE)
N1 G20
...
N10 G50 S1400
N11 G00 T0600 M43
N12 G96 S600 M03
/ N13 G42 X2.0563 Z0.1 T0606 M08
```



Hình 22.6. Áp dụng cắt gọt thử để đo trên máy tiện, chương trình O2206.

```
/ N14 G01 Z-0.4 F0.008
/ N15 X2.3 F0.03
/ N16 G00 G40 X3.0 Z2.0 T0600 M00
/ (TRIAL DIA IS 2.0563 INCHES)
/ N17 G96 S600 M03
N18 G00 G42 X1.675 Z0.1 T0606 M08
N19 G01 X2.0 Z-0.0625 F0.007
N20 Z-1.75
N21 X3.5 F0.01
N22 G00 G40 X10.0 Z2.0 T0600
N23 M01
```

Khi chương trình O2206 được xử lý với xác lập OFF bỏ qua block, mọi block đều được thực thi gồm cả cắt gọt thử và gia công biên dạng hoàn tất. Với xác lập ON bỏ qua block, sẽ *không có* cắt gọt thử. Trong trường hợp đó, mọi lệnh đều được bảo toàn bằng cách lặp lại (N18 và N19). Sự lặp lại này là cốt lõi để xử lý thành công trong cả hai chế độ của hàm bỏ qua block. Hàm M00 trong N16 luôn luôn dừng máy để có thể kiểm tra kích thước.

Sự lựa chọn đường kính cắt thử là 2.0563 trong ví dụ này có thể còn chưa rõ,. Logic ở đây là gì? Đường kính cắt thử có thể là giá trị khác, chẳng hạn 2.05. Điều đó sẽ để lại lượng dư 0.025 trên cả hai phía để gia công hoàn tất. Trong thực tế hoàn toàn *có thể* lựa chọn gia công đường kính khác. Số có 4 chữ số thập phân được chọn chỉ với một lý do, thử thách *tâm lý* người vận hành duy trì các xác lập bù chính xác. Các nhà lập trình hoàn toàn có thể chọn số có ba hoặc hai chữ số thập phân, tùy theo sở thích.

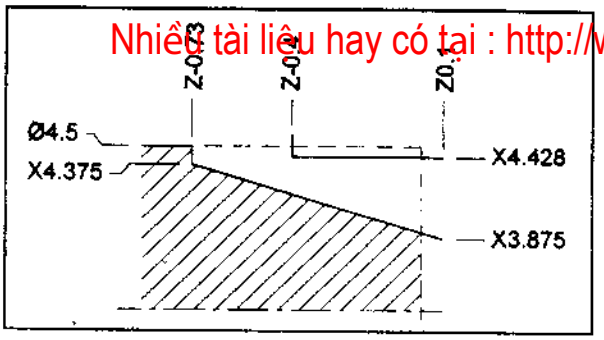
Trong ví dụ kế tiếp, sự cắt gọt thử sẽ được lập trình trước khi cắt gọt thực, nhưng với lý do khác (Hình 22.7).

Trong chương trình O2207, hình dạng hoàn tất của chi tiết gia công là hình côn, hơi khó đo đạc sau khi gia công. Điều chỉnh bù dao theo

http://www.me-cad.com

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 22.7. Cắt gọt thử để tiện côn – chương trình O2207.

phương pháp thử và sai không phải là giải pháp hợp lý. Lập trình cắt gọt thử cho một phần phôi, *dọc theo đường kính thẳng*, cho phép người vận hành kiểm tra kích thước cắt thử một cách thuận tiện và điều chỉnh sự bù dao *trước khi* tiện mặt côn.

```
O2207
(TRIAL CUT FOR TAPER - ONE TOOL)
N1 G20 G99 G40
N2 G50 S1750 T0200 M42
N3 G96 S500 M03
/ N4 G00 G42 X4.428 Z0.1 T0202 M08
/ N5 G01 Z-0.4 F0.008
/ N6 U0.2 F0.03
/ N7 G00 G40 X10.0 Z5.0 T0200 M00
/ (TRIAL CUT DIA IS 4.428 INCHES)

/ N8 G96 S500 M03
N9 G00 G42 X4.6 Z0.1 T0202 M08
N10 G71 P11 Q13 U0.06 W0.005 D1500 F0.01
N11 G00 X3.875
N12 G01 X4.375 Z-0.73 F0.008
N13 X4.6 F0.012
N14 S550 M43
N15 G70 P11 Q13
N16 G00 G40 X10.0 Z5.0 T0200 M01
```

Chương trình O2207 minh họa tình huống chung, một dao được dùng cho cả cắt thô *và* cắt tinh, cho thấy phương pháp logic sử dụng hàm bỏ qua block ở dạng đơn giản. Trong hầu hết các ứng dụng, có thể cần các dao riêng để cắt thô và cắt tinh, tùy theo độ chính xác yêu cầu. Khi sử dụng hai dao, kích thước cắt thử đối với gia công tinh thường quan trọng hơn so với dao cắt thô. Trong chương trình O2208 hàm bỏ qua block được minh họa sử dụng hai dao cắt – T02 để cắt thô, T04 để cắt tinh, sử dụng lại Hình 22.7.

```
O2208
(TRIAL CUT FOR TAPER - TWO TOOLS)
N1 G20 G99 G40
N2 G50 S1750 T0200 M42
N3 G96 S500 M03
/ N4 G00 G42 X4.46 Z0.1 T0202 M08
/ N5 G01 Z-0.4 F0.008
/ N6 U0.2 F0.03
/ N7 G00 G40 X10.0 Z5.0 T0200 M00
/ (T02 TRIAL CUT DIA IS 4.46 INCHES)

/ N8 G50 S1750 T0400 M43
/ N9 G96 S550 M03
/ N10 G00 G42 X4.428 Z0.1 T0404 M08
/ N11 G01 Z-0.4 F0.008
/ N12 U0.2 F0.03
/ N13 G00 G40 X10.0 Z5.0 T0400 M00
/ (T04 TRIAL CUT DIA IS 4.428 INCHES)

/ N14 G50 S1750 T0200 M42
/ N15 G96 S500 M03
N16 G00 G42 X4.6 Z0.1 T0202 M08
N17 G71 P18 Q20 U0.06 W0.005 D1500 F0.01
N18 G00 X3.875
N19 G01 X4.375 Z-0.73 F0.008
N20 X4.6 F0.012
N21 G00 G40 X10.0 Z5.0 T0200 M01

N22 G50 S1750 T0400 M43
N23 G96 S550 M03
N24 G00 G42 X122.0 Z3.0 T0404 M08
N25 G70 P18 Q20
N26 G00 G40 X10.0 Z5.0 T0400 M09
N27 M30
%
```

Ví dụ O2208 có thể tiếp tục cải tiến, ví dụ, bằng cách gộp sự điều khiển mặt côn theo chiều rộng. Lập trình cắt gọt thử là rất hữu ích nhưng là kỹ thuật ít được chú ý, dù có nhiều ứng dụng thực tiễn.

Kiểm chứng chương trình

Hàm bỏ qua block còn có thể hữu ích để kiểm chứng chương trình mới trên máy, tìm các lỗi sai, nếu có. Người vận hành CNC ít kinh nghiệm thường hơi khó chạy chương trình mới. Một trong các vấn đề phổ biến nhất của người vận hành là chuyển động nhanh ban đầu hướng đến chi tiết, đặc biệt khi khoảng trống tương đối nhỏ. Trên hầu hết các bộ điều khiển, người vận hành có thể xác lập tốc độ chạy dao nhanh theo giá trị 100%, 50%, 25% hoặc thấp hơn. Trên các bộ điều khiển cũ, điều này không thực hiện được.

Hai ví dụ kế tiếp O2209 và O2210 sẽ minh họa phương pháp lập trình nhằm loại bỏ vấn đề khi gá lắp và kiểm chứng chương trình, cho phép duy trì tốc độ chạy dao nhanh toàn phần trong các nguyên công lặp lại để tăng năng suất.

Hàm bỏ qua block trong các ví dụ này có vai trò hơi khác với các ví dụ trước, được dùng cho *một phần* của block, thay vì toàn bộ block, nếu bộ điều khiển hỗ trợ phương pháp này.

```
O2209 (TIEN)
N1 G20 G40 G99
N2 G50 S2000
N3 G00 T0200 M42
N4 G96 S400 M03
N5 G41 X2.75 Z0 T0202 M08 /G01 F0.1
N6 G01 X.. F0.004
N7 ...

O2210 (PHAY)
N1 G20 G17 G40 G80
N2 G90 G00 G54 X219.0 Y75.0 M08
N3 G43 Z-1.0 S600 M03 H01 /G01 F30.0
N4 G01 X.. F12.0
N5 ...
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trong cả hai ví dụ, hàm bỏ qua block được dùng *bên trong* một block. Thiết kế của cả hai chương trình đều sử dụng ưu thế *hai lệnh mâu thuẫn trong một block*. Nếu hai lệnh mâu thuẫn nhau cùng hiện diện trong một block, lệnh *đứng sau* trong block sẽ có hiệu lực.

Trong hai ví dụ này, lệnh *thứ nhất* là G00, lệnh *thứ hai* là G01. Thông thường, chuyển động G01 được ưu tiên. Do mã dấu nghiêng, bộ điều khiển sẽ chấp nhận G00, nếu xác lập bỏ qua block là ON, nhưng sẽ dùng G01 nếu xác lập bỏ qua block là OFF. Khi chế độ bỏ qua block là OFF, cả hai lệnh chuyển động đều được đọc và lệnh *thứ hai* trong block đó sẽ có hiệu lực (G01 vượt qua G00). Bạn hãy cảnh giác với một khả năng, đã được nhấn mạnh:

Bỏ qua block bên trong block có thể không có hiệu lực trên một số bộ điều khiển.

Trong lần chạy máy thứ nhất, người vận hành cần xác lập bỏ qua block là OFF, làm cho lệnh G01 có hiệu lực. Chuyển động dao sẽ chậm hơn, nhưng an toàn hơn. Ngoài ra, công tắc vượt qua tốc độ cắt của hệ điều khiển cũng sẽ có hiệu lực, làm tăng thêm tính linh hoạt.

Khi hoàn tất sự kiểm chứng chương trình và sự tiếp cận dao an toàn được khẳng định, sự bỏ qua block có thể được xác lập là ON, do đó không xử lý, chuyển động G01, cả O2209 và O2210 đều là các ví dụ điển hình để đạt được kết quả chuyên biệt..

Ứng dụng cấp phôi thanh

Trên máy tiện CNC, hàm bỏ qua block có thể được dùng trong chế độ cấp phôi thanh (chẳng hạn qua mâm cặp) để gia công liên tục. Nếu bộ cấp phôi thanh cho phép điều đó, kỹ thuật này sẽ rất đơn giản. Chương trình thường có *hai kết thúc* – một sẽ sử dụng hàm M99, kết thúc thứ hai sử dụng hàm M30. Block M99 với ký hiệu bỏ qua block sẽ được đặt *trước* mã M30 trong chương trình gia công. Kỹ thuật đặc biệt này được trình bày trong Chương 43.

Bỏ qua block theo số

Trong khi gia công, hàm bỏ qua block có thể được xác lập theo ON hoặc OFF và giữ nguyên chế độ này trong *toàn bộ* chương trình. Nếu cần xác lập ON cho một phần chương trình nhưng không xác lập cho phần khác, người lập trình được thông báo rõ, thường là bằng các ghi chú trong chương trình. Sự thay đổi chế độ bỏ qua block ở giữa chương trình có thể không an toàn và thường gây ra các vấn đề.

Tính năng tùy chọn trên một số bộ điều khiển là hàm bỏ qua block *theo số* hoặc có *tính chọn lọc*. Tùy chọn này cho phép người vận hành lựa chọn (các) phần của chương trình cần xác lập ON và (các) phần yêu cầu xác lập OFF. Các xác lập này có thể được thực hiện *trước khi* nhấn phím *Cycle Start* để khởi động chương trình. Phương pháp này cũng sử dụng dấu nghiêng, tiếp theo là số nguyên, từ 1 đến 9. Sự lựa chọn thực sự chế độ này được thực hiện trên màn hình điều khiển (*Settings*) theo số công tắc tương ứng.

Ví dụ, chương trình có thể gồm ba nhóm, mỗi nhóm có xác lập riêng về hàm bỏ qua block. Bằng cách dùng số công tắc sau ký hiệu dấu nghiêng, các nhóm được xác định rõ ràng và toàn bộ điều cần làm của người vận hành là *tương hợp* các xác lập điều khiển với hoạt động được yêu cầu.

```
N1 ...
N2 ..
/1 N3 ...                    (BLOCK SKIP GROUP 1)
/1 N4 ...                    (BLOCK SKIP GROUP 1)
...
...
N16...
/2 N17...                    (BLOCK SKIP GROUP 2)
/2 N18 ...                   (BLOCK SKIP GROUP 2)
/2 N19 ...                   (BLOCK SKIP GROUP 2)
...
...
N29 ...
/3 N30 ...                   (BLOCK SKIP GROUP 3)
/3 N31 ...                   (BLOCK SKIP GROUP 3)
...
...
N45 ...
```

Các nguyên tắc áp dụng cho hàm bỏ qua block có chọn lọc hoàn toàn như hàm bỏ qua block thông thường. Sự lựa chọn /1 hoàn toàn như dấu nghiêng không có số, do đó các block N3 và N4 nêu trên có thể viết như sau:

```
/N3..
/N4..
```

Chỉ một số bộ điều khiển hỗ trợ hàm bỏ qua block có chọn lọc

Các chương trình sử dụng hàm bỏ qua block có chọn lọc là rất hiệu quả nhưng tăng thêm trách nhiệm cho người vận hành máy. Đối với nhiều chương trình gia công, chỉ cần sử dụng hàm bỏ qua block tiêu chuẩn cũng đủ để phát huy toàn bộ năng lực lập trình khả dụng.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

23 LỆNH DWELL

Dwell được hiểu là *sự tạm dừng* trong chương trình, đây là sự tạm dừng có chủ ý trong khi xử lý chương trình. Trong khoảng thời gian đó – được chuyên biệt trong chương trình – mọi chuyển động trục đều dừng lại, mọi hàm và lệnh chương trình đều không bị tác động. Sau thời lượng đó, hệ điều khiển khôi phục sự xử lý chương trình với block *kế tiếp* block chứa lệnh dwell (tạm dừng)

ỨNG DỤNG LẬP TRÌNH

Lập trình dwell là rất dễ dàng và có thể rất hữu ích trong hai nhóm ứng dụng chính:

- Trong khi cắt gọt thực sự, khi dao tiếp xúc với phôi.
- Để thao tác các phụ tùng trên máy, khi không cắt gọt.

Mỗi ứng dụng đều rất quan trọng đối với nhà lập trình, nhưng hai ứng dụng này không được sử dụng đồng thời.

Ứng dụng khi cắt gọt

Khi cắt gọt, dao tiếp xúc với chi tiết gia công. Sự tạm dừng (dwell) có thể được áp dụng trong khi cắt gọt vì nhiều lý do. Nếu trục chính quay, sự quay đó là *rất* quan trọng.

Trong thực tiễn, ứng dụng dwell trong khi cắt gọt chủ yếu để làm gãy phôi cắt, chẳng hạn khi khoan lỗ, doa ngược, cắt rãnh, hoặc cắt đứt. Dwell còn có thể được dùng khi tiện hoặc doa để loại bỏ các dấu trên chi tiết do lực đẩy của dụng cụ cắt. Lực đẩy này là do áp suất của dao trong khi cắt gọt. Trong nhiều ứng dụng khác, hàm dwell rất hữu ích để kiểm soát sự giảm tốc độ cắt ở góc trong khi sử dụng tốc độ cắt nhanh, điều này có thể rất có ích trên các hệ điều khiển cũ. Trong cả hai trường hợp, lệnh dwell "buộc" nguyên công cắt gọt *hoàn tất* trong một số block, *trước khi* có thể thực thi block kế tiếp. Nhà lập trình vẫn có thể cung cấp chính xác thời lượng tạm dừng. Thời lượng này phải đủ, không quá ngắn cũng không quá dài.

Lệnh dwell luôn luôn hoàn tất trước khi bắt đầu thao tác kế tiếp

Ứng dụng cho các phụ tùng

Ứng dụng phổ biến thứ hai của lệnh dwell là đứng sau một số hàm M. Các hàm này được dùng để điều khiển nhiều phụ tùng máy CNC, chẳng hạn bộ cấp phôi thanh, ụ động, đồ gá, cữ chặn,... Thời gian tạm dừng được lập trình sẽ cho phép hoàn tất quy trình nào đó, chẳng hạn thao tác ở ụ động. Trục chính của máy có thể quay hoặc tĩnh tại trong các trường hợp đó. Do không có sự tiếp xúc giữa dao và chi tiết gia công, trục quay hay không quay đều *không* quan trọng.

Trên một số máy CNC, có thể cần lệnh dwell khi thay đổi tốc độ trục chính, thường là sau khi đổi số tốc độ. Điều này được dùng chủ yếu trên máy tiện CNC. Trong các trường hợp đó, hướng dẫn tốt nhất về *phương pháp* và *thời điểm* lập trình thời gian dwell là tuân theo hướng dẫn của nhà chế tạo máy CNC. Các ví dụ về sử dụng dwell cho các phụ tùng máy tiện sẽ được trình bày trong Chương 43.

LỆNH DWELL

G04	Lệnh dwell

Lệnh chuẩn bị chung cho dwell là G04. Tương tự các lệnh G khác, lệnh G04 độc lập sẽ không có tác dụng. Lệnh này luôn luôn được dùng với địa chỉ chuyên biệt thời lượng dwell. Các địa chỉ dùng cho lệnh dwell là X, P, hoặc U (địa chỉ U chỉ dùng trên máy tiện CNC). Thời lượng thực được chuyên biệt theo địa chỉ đã chọn là *miligiây (ms)* hoặc *giây (s)* tùy theo địa chỉ đó. Một số hệ điều khiển sử dụng địa chỉ khác cho lệnh dwell, nhưng mục đích và phương pháp lập trình không thay đổi.

Một số chu kỳ cố định trên trung tâm gia công cũng sử dụng lệnh dwell. Lệnh dwell này được lập trình cùng với dữ liệu chương trình không trong block riêng rẽ. Chỉ có các chu kỳ cố định yêu cầu thời lượng dwell mới được dùng trong cùng block đó. Đối với mọi ứng dụng khác, lệnh dwell phải được lập trình theo *block độc lập*, chỉ có hiệu lực đối với block đó và không chuyển sang block kế tiếp. Dwell là hàm chỉ một block và *không* có tính chế độ. Trong khi thực thi dwell, trạng thái xử lý hiện hành của chương trình không thay đổi, nhưng thời gian chung của chu kỳ sẽ kéo dài theo thời lượng dwell.

Cấu trúc của lệnh dwell

Cấu trúc hoặc định dạng của hàm dwell như sau:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
X5.3     ... Mọi máy móc, trừ các chu kỳ cố định.
U5.3     ...Chỉ dùng cho máy tiện.
P53      ... Mọi máy công cụ, kể cả chu kỳ cố định.
```

Trong trường hợp bất kỳ, sự biểu diễn là 5 chữ số *trước* và ba chữ số *sau* dấu thập phân, dù điều đó có thể thay đổi trên các hệ điều khiển khác nhau.

Các ví dụ về ứng dụng thực tiễn của định dạng dwell bao gồm:

```
G04 X2.0      ... thường dùng cho các dwell dài
G04 P2000     ... thường dùng cho các dwell
                  trung bình và ngắn
G04 U2.0      ... chỉ dùng trên máy tiện, đơn vị
                  là giây
```

Trong ví dụ này, dwell là 2 giây hay 2000 mili giây với cả ba định dạng. Ví dụ kế tiếp cũng tương tự

```
G04 X0.5
G04 P500
G04 U0.5
```

Ví dụ này minh họa thời lượng dwell là 500 mili giây hoặc nửa giây với cả ba định dạng.

Trong chương trình CNC, hàm dwell có thể xuất hiện theo cách thức dưới đây, bạn hãy lưu ý dwell trong một block riêng:

```
N21 G01 Z-1.5 F12.0
N22 G04 X0.3                (LỆNH DWELL 0.3 SEC)
N23 Z-2.7 F8.0
```

Các chương trình sử dụng địa chỉ X hoặc U có thể gây ra sự nhầm lẫn, đặc biệt đối với các nhà lập trình mới. Các địa chỉ X hoặc U có thể được diễn dịch không đúng, do có thể hiểu là chuyển động trục. *Điều này không thể xảy ra.* Theo định nghĩa, trục X và ứng dụng tiện của trục này, trục U, là trục *dwell*. X là trục chung duy nhất cho mọi máy CNC.

Không có chuyển động trục xảy ra khi địa chỉ X, P hoặc U được dùng với lệnh dwell G04

Bộ điều khiển diễn dịch lệnh này theo dwell, *không theo chuyển động trục*. Điều này là do sự hiện diện của lệnh chuẩn bị G04, quyết định *ý nghĩa* của địa chỉ đi kèm. Nếu sử dụng địa chỉ X hoặc U đối với dwell không thuận tiện, bạn hãy dùng địa chỉ P. Bạn cần nhớ, địa chỉ P *không* chấp nhận dấu thập phân, do đó dwell được lập trình trực tiếp theo số *miligiây* để điều khiển thời lượng tạm dừng. Một miligiây tương ứng 1/1000 giây, do đó một giây là 1000 miligiây.

Các địa chỉ X và U còn có thể được lập trình theo miligiây, không dùng dấu thập phân, ví dụ:

```
G04 X2.0   tương đương với   G04 X2000
```

Có thể bỏ các chữ số zero đứng trước trong định dạng không có dấu thập phân (nhưng không được bỏ các chữ số zero đứng sau):

```
P1 =  P0001          ...1 mili giây
P10 =  P0010         ...10 mili giây
P100 =  P0100        ...100 mili giây
```

Tùy theo định dạng lập trình đối với dwell, khoảng thời lượng lập trình có thể thay đổi. Đối với định dạng sử dụng 5 chữ số đứng trước và 3 chữ số đứng sau dấu thập phân, khoảng thời lượng là 0.009 giây đến 99999.999 giây, tương ứng khoảng từ 1/1000 giây đến *27 giờ, 46 phút và 39,999 giây*.

Các ứng dụng lập trình dwell là như nhau trên cả trung tâm gia công và máy tiện CNC, nhưng địa chỉ U chỉ có thể được sử dụng trong các chương trình tiện. Sự lựa chọn đơn vị hệ mét hoặc hệ Anh không ảnh hưởng đến hàm dwell.

LỰA CHỌN THỜI GIAN TẠM DỪNG

Rất ít khi thời gian tạm dừng dwell vượt quá vài giây, hầu hết đều dưới một giây. Dwell luôn luôn là *thời gian không gia công*, do đó phải được chọn với thời lượng ngắn nhất cần thiết để hoàn tất hoạt động được yêu cầu. Thời gian để hoàn tất thao tác máy hoặc phụ tùng máy thường do nhà chế tạo máy đề nghị. Sự lựa chọn thời gian dwell đối với các mục đích cắt gọt luôn luôn là trách nhiệm của nhà lập trình. Tuy nhiên, một số nhà lập trình thường lập trình thời gian tạm dừng quá dài. Một giây dường như chỉ là thời gian rất ngắn, nhưng bạn hãy xét ví dụ dưới đây:

Trong một block chương trình, hàm dwell được gán thời lượng là một giây. Tốc độ trục chính được chọn là 4805/min và dwell được áp dụng ở 50 vị trí trên chi tiết có lẽ với nguyên công khoan điểm. Điều đó có nghĩa là chu kỳ thời gian đối với *từng* chi tiết dài thêm 50 giây *với* dwell so với *không có* dwell. Năm mươi giây dường như không quá bất hợp lý. Nhưng có thật sự cần thiết không? Bạn hãy thử suy nghĩ và tính toán. Nếu buộc phải sử dụng dwell, bạn hãy tính thời lượng *tối thiểu* có thể thực hiện công việc đó. Có thể dễ dàng lựa chọn thời lượng dwell một cách tùy ý. Trong ví dụ này, thời lượng dwell tối thiểu chỉ là 0.125 giây:

```
60 / 480 = 0.125
```

Thời lượng dwell tối thiểu này *nhỏ hơn 8 lần* so với dwell được lập trình một giây. Nếu sử dụng dwell tối thiểu thay cho dwell được chọn, thời gian chu kỳ sẽ chỉ tăng 6.25 giây, thay vì 50 giây, sự cải thiệt rõ rệt về hiệu suất chương trình và năng suất trên máy công cụ.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

XÁC LẬP CHẾ ĐỘ VÀ DWELL

Hầu hết các chương trình trên trung tâm gia công đều sử dụng tốc độ cắt theo thời gian (lập trình theo *in/min* hoặc *mm/min*). Các ứng dụng tiện thường lập trình tốc độ cắt theo vòng quay *in/rev* hoặc *mm/rev*. Trên nhiều bộ điều khiển Fanuc, xác lập tham số cho phép lập trình dwell theo thời lượng hoặc *số vòng quay trục chính*. Mỗi ứng dụng đều có ưu điểm và công dụng thực tiễn riêng. Tùy theo xác lập tham số hệ thống, lệnh dwell sẽ biết ý nghĩa với từng xác lập tương ứng:

Xác lập thời gian

Đây là xác lập mặc định phổ biến trên nhiều thiết bị CNC. Đối với *xác lập thời gian*, dwell luôn luôn được lập trình theo giây hoặc mili giây trong khoảng cho phép của bộ điều khiển, chẳng hạn từ 0.001 đến 99999.999 giây, thông dụng trên hầu hết các bộ điều khiển Fanuc, hoặc tương tự Fanuc:

```
G04 P1000
```

... biểu thị dwell một giây, tương đương 1000 mili giây.

Xác lập số vòng quay

Đối với xác lập *số vòng quay trục chính*, dwell được xác lập theo số lượng vòng quay của trục chính trong khoảng 0.001 – 99999.999 vòng, ví dụ:

```
G04 P1000
```

... biểu thị dwell kéo dài trong một vòng quay trục chính

DWELL TỐI THIỂU

Trong khi cắt gọt, dao cắt tiếp xúc với chi tiết gia công, định nghĩa dwell tối thiểu là rất quan trọng, nhưng chế độ xác lập là không quan trọng (thời gian hoặc số vòng quay).

Dwell tối thiểu là thời gian cần thiết để hoàn tất một vòng quay trục chính.

Dwell tối thiểu được lập trình theo giây, có thể tính toán theo công thức

$$\text{Dwell tối thiểu (giây)} = \frac{60}{\text{r/min}}$$

➲ **Ví dụ**

Để tính dwell tối thiểu theo giây, nếu biết số vòng quay trục chính là 420 r/min, bạn hãy chia *r/min* cho 60 (do một phút bằng 60 giây):

```
60 / 420 = 0.143 giây tạm dừng
```

Sự lựa chọn định dạng của block chứa dwell trong chương trình có thể khác nhau, tùy theo kiểu máy và phong cách lập trình. Mọi ví dụ dưới đây đều sử dụng thời gian dwell (tạm dừng) là 0.143 giây:

```
G04 X0.143
G04 P143
G04 U0.143
```

Bất kể định dạng được sử dụng, mọi giá trị dwell trong ví dụ đều chuyên biệt là 0.143 giây, tương đương 143 mili giây. Chương trình có thể chứa các định dạng dwell khác nhau, nhưng ít được dùng trong thực tế.

Đối với các ứng dụng dwell *thực tiễn* trong chương trình CNC, dwell tối thiểu được tính toán chỉ đúng về toán học nhưng có thể *không* phù hợp với ứng dụng thực tế. Bạn nên làm tròn giá trị tính toán theo hướng *tăng*. Ví dụ G04 X0.134 có thể chọn là G04 X0.2 hoặc G04 X0.3. Lý do để thực hiện sự điều chỉnh này là từ các điều kiện gia công thực tế. Chẳng hạn người vận hành CNC có thể tiến hành gia công với tốc độ trục chính theo chế độ override, ví dụ 50%. Do override tốc độ trục chính 50% thường là giá trị tối thiểu trên hầu hết các bộ điều khiển CNC, *sự gấp đôi* dwell tối thiểu sẽ *luôn luôn bảo đảm ít nhất một vòng quay* mà không bị tổn thất năng suất.

SỐ VÒNG QUAY

Trong các chế độ dwell khác nhau (được chọn bằng tham số hệ thống) định dạng lập trình là *như nhau*, nhưng ý nghĩa sẽ rất khác nhau. Trong một số ứng dụng, có thể cần lập trình sự tạm dừng theo *số vòng quay* thay vì thời gian.

Ví dụ, trong nguyên công cắt rãnh trên máy tiện, chuyển động dao được lập trình đến đáy rãnh sẽ đòi hỏi dao cắt rãnh duy trì vị trí đó trong một khoảng thời gian để làm sạch đáy rãnh. Việc tính toán thời gian theo giây sẽ giải quyết vấn đề này, nhưng có một giải pháp khác có thể rất hấp dẫn. Nhiều bộ điều khiển cung cấp sự lập trình số chiều quay trục chính một cách trực tiếp, nhưng phải xác lập tham số hệ thống. Ví dụ, có thể lập trình thời lượng tạm dừng (dwell) đủ lâu để hoàn tất *ba vòng quay trục chính*, bất kể r/min của trục chính.

Xác lập hệ thống

Nếu hệ thống điều khiển được xác lập để chấp nhận dwell theo *số vòng quay trục chính*, thay vì thời gian theo giây hoặc miligiây, sự lập trình sẽ rất đơn giản. Toàn bộ điều cần làm là

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

gọi lệnh G04, tiếp theo là số vòng quay theo yêu cầu:

```
G04 X3.0
G04 P3000
G04 U3.0
```

Các định dạng này đều có cùng kết quả dwell trong ba vòng quay trục chính. Làm thế nào để biết giá trị trong chương trình là thời gian hay số vòng quay? Không thể biết điều đó, bạn *chỉ biết* các xác lập điều khiển. Dấu hiệu duy nhất là đánh giá các giá trị nhập tạm dừng (dwell). Số vòng quay 3.0 thường ngắn hơn nhiều so với 3.0 giây tạm dừng, bạn hãy chú ý, số vòng quay vẫn có thể sử dụng dấu thập phân, chẳng hạn ½ hoặc ¼ vòng quay.

Đương lượng thời gian

Hai chế độ dwell này không thể đặt chung trong một chương trình, thậm chí giữa các chương trình cũng rất khó. Tham số hệ thống CNC mỗi lần chỉ có thể xác lập một chế độ dwell. Do các tham số điều khiển thường được xác lập cho dwell theo giây hoặc miligiây, thay vì theo số vòng quay, bạn cần phải tính đương lượng thời gian, do đó phải biết đường kính trục chính (r/min)

Thời gian dwell theo giây có thể tính theo số vòng quay với công thức:

$$\text{Dwell}_{\text{giây}} = \frac{60 \times n}{\text{r/min}}$$

Trong đó:

60:	số giây trong một phút
n:	số vòng quay trục chính yêu cầu
r/min:	tốc độ trục chính hiện hành (vòng/phút)

➲ Ví dụ

Để tính thời gian dwell theo giây tương đương *ba* vòng quay trục chính với tốc độ trục chính là 420 r/min, có thể áp dụng công thức:

$\text{Dwell}_{\text{giây}} = 60 \times 3 / 420 = 0.429$

Block chương trình biểu thị ba vòng quay trục chính theo thời lượng dwell sẽ là một trong ba dạng dưới đây:

```
G04 X0.429
G04 P429
G04 U0.429
```

Bạn cũng có thể tính *đương lượng thời gian dwell* từ số vòng quay trục chính. Nói chung, kết quả thường không phải là số nguyên, bạn cần làm tròn theo giá trị nguyên lớn hơn gần nhất. Công thức nêu trên được đổi thành:

$$\text{Dwell}_{\text{số vòng quay}} = \frac{\text{r/min} \times \text{Dwell}_{\text{giây}}}{60}$$

➲ Ví dụ

Để kiểm chứng công thức là đúng, bạn hãy dùng giá trị 0.429 giây đã tính từ ví dụ trước và tính số vòng quay với tốc độ trục chính 420 r/min:

$\text{Dwell}_{\text{vòng}} = 420 \times 0.429 / 60 = 3.003$ *vòng quay*

Kết quả này khẳng định công thức nêu trên là đúng. Nói chung, tính toán này bắt đầu với giá trị dwell đã được làm tròn, ví dụ đến 0.5 giây.

$\text{Dwell}_{\text{vòng}} = 420 \times 0.5 / 60 = 3.5$ *vòng quay*

Thời gian dwell dựa trên số vòng quay trục chính được dùng chủ yếu trên các máy tiện CNC đặc biệt khi cắt gọt với tốc độ trục chính rất chậm. Sự quay trục chính chậm không có dung sai rộng như tốc độ nhanh, không cho phép sai số lớn trong tính toán dwell. Bạn cần nhớ, mục đích là nhận được *ít nhất một vòng quay chi tiết* để đạt được kết quả gia công mong muốn. Hơn nữa, tại sao phải lập trình dwell? Bạn hãy xét ví dụ:

Dwell được lập trình có thời lượng là *nửa giây*, với tốc độ trục chính 80 r/min. Số vòng quay trong 0.5 giây sẽ là:

$80 \times 0.5 / 60 = 0.6666667$

nhỏ hơn một vòng quay hoàn chỉnh của trục chính. Thời lượng dwell 0.5 giây là không đủ. Dwell tối thiểu được tính toán với công thức nêu trên sẽ là:

$60 \times 1 / 80 = 0.75$ giây

Nói chung, kiểu tính toán này ít sử dụng trong thực tế, giá trị dwell lập trình thường tính toán *theo thời gian* tiêu chuẩn.

THỜI GIAN DWELL DÀI

Đối với các mục đích gia công trên máy CNC, thời gian dwell dài thường ít được yêu cầu và hầu như không cần thiết. Tuy nhiên, có vài trường hợp cần dùng thời gian dwell dài.

Thời gian dwell dài là thời lượng được lập trình *dài hơn giá trị trung bình được thiết lập* cho hầu hết các ứng dụng bình thường. Hầu như rất ít khi cần lập trình thời gian dwell trong khi gia công chi tiết vượt quá một, hai, ba, hoặc bốn giây. Khoảng thời lượng lớn khả dụng trên hệ thống điều khiển (*hơn 27 giờ*) đối với *sự bảo trì máy* là quan trọng hơn so với nhà lập trình CNC. Ví dụ về ứng dụng dwell dài là chương trình do kỹ thuật viên bảo trì sử dụng để kiểm tra chức năng của trục chính.

Bạn hãy xét tình huống thực tế về bảo

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

dưỡng máy, trục chính của máy CNC được sửa chữa và cần kiểm tra trước khi đưa máy vào sản xuất. Sự kiểm tra chủ yếu là chạy trục chính với các tốc độ khác nhau, mỗi tốc độ sẽ chạy với thời gian xác định.

Trong ví dụ này, bộ phận bảo trì cần một chương trình CNC nhỏ, trong đó trục chính quay 100 r/min trong 10 phút, sau đó quay 500 r/min trong 20 phút, cuối cùng là 1000 r/min trong 30 phút. Chương trình không hẳn là cần thiết, do kỹ thuật viên bảo trì có thể kiểm tra bằng các phương pháp thủ công. Phương pháp thủ công tuy không thực sự hiệu quả nhưng vẫn được dùng cho mục đích kiểm tra bảo trì.

Trong các trường hợp đó, phương pháp tốt hơn có lẽ là *lưu* quy trình kiểm tra theo chương trình nhỏ, trực tiếp trong bộ nhớ CNC. Chương trình bảo trì hơi khác nhau giữa trung tâm gia công và máy tiện nhưng các đối tượng chính là như nhau.

➲ Ví dụ–Trung tâm gia công–kiểm tra trục chính

```
S100 M03              (100 R/MIN INITIAL SPEED)
G04 X600.0          (600 SECONDS IS 10 MINUTES)
S500             (SPEED INCREASED TO 500 R/MIN)
G04 X1200.0        (1200 SECONDS IS 20 MINUTES)
S1500         (SPEED INCREASED TO 1500 R/MIN)
G04 X1800.0        (1800 SECONDS IS 30 MINUTES)
M05                             (SPINDLE STOP)
```

Ví dụ, trung tâm gia công khởi động với tốc độ trục chính ban đầu là 100 r/min. Tiếp theo là dwell 600 giây, bảo đảm 10 phút chạy với tốc độ không đổi. Tốc độ trục chính sau đó tăng đến 500 r/min và thời gian dwell là 1200 giây (20 phút). Cuối cùng là tốc độ trục chính 1500 r/min chạy trong 1800 giây (30 phút), và trục chính dừng lại.

➲ Ví dụ – máy tiện – kiểm tra trục chính

```
M43                             (GEAR RANGE SELECTION)
G97 S100 M03          (100R/MIN INITIAL SPEED)
G04 X600.0          (600 SECONDS IS 10 MINUTES)
S500             (SPEED INCREASED TO 500 R/MIN)
G04 X1200.0        (1200 SECONDS IS 20 MINUTES)
S1500         (SPEED INCREASED TO 1500 R/MIN)
G04 X1800.0        (1800 SECONDS IS 30 MINUTES)
M05                             (SPINDLE STOP)
```

Chương trình kiểm tra trục chính trên máy tiện hầu như tương tự chương trình dùng cho trung tâm gia công nêu trên. Xác lập tốc độ trục chính ban đầu là lựa chọn khoảng tốc độ trên hộp số, ví dụ M43. Tốc độ quay trục chính là 100r/min, thời gian dwell là 600 giây để trục chính quay trong 10 phút. Sau đó tăng tốc độ đến 500 r/min, trục chính quay trong 20 phút (1200 giây). Trước khi dừng, trục chính quay 1500 r/min trong 30 phút (1800 giây).

Cần tuân thủ mọi nguyên tắc an toàn khi sử dụng thời lượng dwell dài

Trục dwell là trục X

Màn hình bộ điều khiển hiển thị thời lượng còn lại trước khi kết thúc thời gian dwell. Bạn có thể quan sát điều đó bằng cách xem hiển thị X của bộ chỉ thị *Distance – To – Go* trên màn hình POS (position – vị trí) của bộ điều khiển Fanuc. Màn hình này luôn luôn hiển thị theo X, do trục X là trục dwell duy nhất, bất kể hệ điều khiển, dù các địa chỉ P hoặc U được lập trình. Tại sao trục X được chọn là trục dwell và tại sao không là trục khác? Lý do rất đơn giản, bởi vì *X là trục duy nhất chung* cho mọi máy công cụ CNC, máy khoan, máy phay, trung tâm gia công, máy cắt bằng ngọn lửa, máy cắt bằng tia nước, máy laser, ... Tất cả đều sử dụng các trục XYZ. Máy tiện sử dụng các trục XZ (không dùng trục Y), máy gia công tia lửa điện (EDM) sử dụng các trục XY (không dùng trục Z). Các máy khác cũng tương tự.

An toàn và dwell

Bạn cần luôn luôn chú ý khi sử dụng chương trình có thời lượng dwell dài, đặc biệt đối với mục đích bảo trì hoặc kiểm tra sau khi sửa chữa. Không được để máy CNC chạy nếu không có sự canh chừng. Nếu cần thời gian dài để kiểm tra cần có các ký hiệu hoặc bảng cảnh báo đặt ở gần máy để tránh tình huống không an toàn, thậm chí phải có người giám sát bên cạnh.

Về nguyên tắc, hàm dwell không được phép lập trình chỉ để người vận hành máy có thời gian thực hiện thao tác bằng tay trong khi xử lý chương trình. Các thao tác bằng tay chẳng hạn đánh bóng, giũa, khử ba via, đảo ngược chi tiết, thay dao, loại bỏ phoi, bôi trơn, kiểm tra, ... phải luôn luôn thực hiện, nếu thực sự cần thiết trong khi thực thi chương trình, trong chương trình vận hành bằng tay, *không được phép thực thi với sự điều khiển chương trình.*

Không được dùng dwell để thực hiện các thao tác bằng tay trên máy CNC

CHU KỲ CỐ ĐỊNH VÀ DWELL

Chương 24 sẽ trình bày chi tiết về chu kỳ cố định trên máy khoan CNC và trung tâm gia công. Trong phần này chỉ nêu các vấn đề về dwell và chu kỳ cố định.

Một số chu kỳ cố định có thể được lập trình với dwell:

- Nói chung, các chu kỳ G76, G82, G88, G89
- Các chu kỳ G74 và G84 phải có xác lập tham số.

Địa chỉ dwell trong chu kỳ cố định luôn

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

luôn là P để tránh trùng với địa chỉ X trong một block. Địa chỉ U và lệnh G04 *không* được lập trình trong chu kỳ cố định, hàm dwell được "thiết lập" trong mọi chu kỳ cố định cho phép tham số dwell (về kỹ thuật, hầu như tất cả các chu kỳ đều cho phép điều đó). Các nguyên tắc tính toán thời lượng dwell cho chu kỳ cố định hoàn toàn như các ứng dụng gia công khác.

⊃ Ví dụ

```
N9 G82 X1.2 Y0.6 R0.2 Z-0.7 P300 F12.0
```

Trong ví dụ này, dwell là 300 mili giây (0.3 giây) được xác định theo địa chỉ P. Dwell này sẽ có hiệu lực khi hoàn tất chuyển động dọc theo trục Z (chuyển động cắt thực sự), nhưng *trước* chuyển động nhanh trả dao trở về.

Nếu G04 được lập trình theo block riêng rẽ trong chu kỳ cố định, chẳng hạn giữa block 82 và block 80, *không có chu kỳ nào* được thực thi trong block đó và giá trị P trong định nghĩa chu kỳ cố định *không được* cập nhật. Trên các bộ điều khiển hiện đại, xác lập tham số hệ thống sẽ cho phép hoặc không cho phép ứng dụng này. Nếu sử dụng phương pháp đó, lệnh G04 P.. sẽ có hiệu lực *trước* chuyển động chạy dao nhanh từ vị trí vừa hoàn tất. Hàm dwell luôn luôn thực thi trong khi dao ở ngoài lỗ gia công, trong khu vực trống. *Tính năng này ít được yêu cầu.*

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

24 CHU KỲ CỐ ĐỊNH

Gia công các lỗ có lẽ là nguyên công thông dụng nhất, chủ yếu được thực hiện trên máy phay CNC và trung tâm gia công. Ngay cả trong các ngành công nghiệp đòi hỏi nhiều chi tiết phức tạp như hàng không, vũ trụ, điện tử, khí cụ đo, khuôn mẫu,... gia công các lỗ cũng là phần rất quan trọng trong quy trình công nghệ.

Khi suy nghĩ về gia công lỗ, bạn nghĩ đến các nguyên công, chẳng hạn khoan tâm, khoan điểm và khoan tiêu chuẩn, sử dụng các loại mũi khoan tương ứng. Tuy nhiên, lĩnh vực này rộng hơn nhiều. Các nguyên công khác có liên quan cũng thuộc lĩnh vực gia công lỗ. Khoan tâm tiêu chuẩn, khoan điểm và khoan được sử dụng cùng với các nguyên công liên quan chẳng hạn chuốt, doa, cắt ren,...

Gia công một lỗ đơn giản có lẽ chỉ cần một dụng cụ cắt nhưng lỗ phức tạp và chính xác có thể cần vài dao để hoàn tất. Số lượng lỗ cần gia công trên một chi tiết cũng là yếu tố quan trọng để lựa chọn phương pháp lập trình.

Ngay cả các lỗ được gia công với cùng một dụng cụ cắt cũng có thể khác nhau. Các lỗ cùng đường kính có thể có chiều sâu khác nhau, thậm chí ở các bậc khác nhau trên chi tiết gia công. Nếu xét mọi góc độ, có thể nhận thấy gia công một lỗ là tương đối đơn giản, nhưng gia công nhiều lỗ khác nhau trong một chương trình sẽ đòi phương pháp được sắp xếp và lập kế hoạch cẩn thận.

Trong đa số các ứng dụng lập trình, gia công lỗ có nhiều đặc tính tương tự nhau giữa các chi tiết khác nhau. Gia công lỗ là nguyên công có thể dự đoán trước một cách hợp lý, loại nguyên công này là lý tưởng để lập trình hiệu quả trên máy tính. Vì lý do đó, hầu như mọi nhà chế tạo bộ điều khiển CNC đều đưa các phương pháp lập trình gia công lỗ vào hệ thống điều khiển của họ. Các phương pháp này sử dụng *chu kỳ cố định*.

GIA CÔNG ĐIỂM – ĐIỂM

Gia công lỗ không phải là quy trình quá phức tạp, không đòi hỏi gia công biên dạng, cũng không có chuyển động dao nhiều trục. Chuyển động duy nhất khi cắt gọt thực sự là theo một trục, hầu như luôn luôn là trục Z. Kiểu gia công này thường được gọi là *gia công điểm – điểm*.

Gia công điểm – điểm đối với các lỗ là phương pháp điều khiển các chuyển động của dụng cụ cắt theo trục X và trục Y với tốc độ *nhanh*, và trục Z với *tốc độ cắt*. Một số chuyển động theo trục Z cũng có thể là chuyển động nhanh. Điều đó có nghĩa là không có chuyển động cắt gọt theo các trục XY khi gia công lỗ. Khi dụng cụ cắt hoàn tất mọi chuyển động trên trục Z và lùi ra khỏi lỗ sẽ chuyển động theo X và Y để tiến đến vị trí mới trên chi tiết. Tại đây sẽ lặp lại các chuyển động trục Z. Nói chung chuỗi thứ tự các chuyển động này xảy ra tại nhiều vị trí. Phương pháp gia công này là điển hình đối với các chu kỳ cố định khi thực hiện các nguyên công khoan, chuốt, ta rô ren, doa,...

Cấu trúc chương trình cơ bản đối với gia công điểm – điển có thể thu gọn thành bốn bước chung (ví dụ, chuỗi thứ tự khoan):

- ❑ Bước 1: Chuyển động nhanh đến vị trí lỗ ...dọc theo trục X và / hoặc trục Y.
- ❑ Bước 2: Chuyển động nhanh đến điểm bắt đầu cắt gọt ... dọc theo trục Z.
- ❑ Bước 3: Chuyển động cắt (ăn dao) đến chiều sâu yêu cầu ...dọc theo trục Z.
- ❑ Bước 4: Lùi dụng cụ cắt ra khỏi lỗ đến vị trí an toàn ... dọc theo trục Z

Bốn bước này là số lượng block *tối thiểu* cần thiết để lập trình gia công một lỗ, sử dụng phương pháp lập trình bằng tay, *không* dùng chu kỳ cố định. Nếu chỉ có một hoặc hai lỗ trên bản vẽ chi tiết và sự gia công chỉ là khoan tâm hoặc khoan lỗ đơn giản, chương trình sẽ không dài. Đây không phải trường hợp phổ biến, trong thực tế thường có nhiều lỗ trên chi tiết và cần nhiều dụng cụ cắt để hoàn tất từng lỗ theo các yêu cầu kỹ thuật. Chương trình có thể rất dài, khó diễn dịch và khó thay đổi, thậm chí vượt quá dung lượng nhớ của bộ điều khiển CNC tiêu chuẩn.

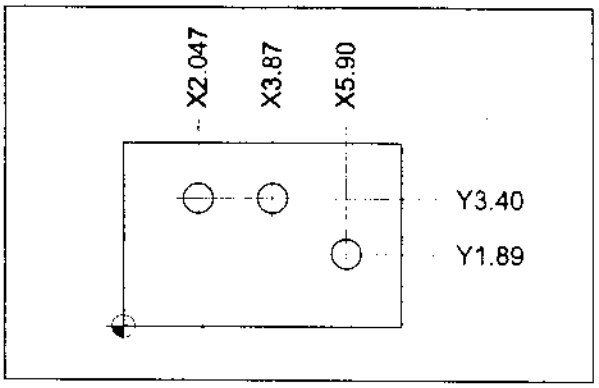
Có lẽ công việc tốn nhiều thời gian nhất trong lập trình gia công điểm – điểm là khối lượng dữ liệu *lặp lại* cần được viết trong chương trình CNC. Vấn đề này đã được giải quyết bằng cách dùng các chu kỳ cố định. Các chu kỳ này còn được gọi là *chu kỳ đóng hộp* do nhiều thông tin lặp lại *đóng hộp* vào không gian tương đối nhỏ trong vi mạch máy tính.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chuyển động dao và chu kỳ cố định

Hai ví dụ dưới đây sẽ so sánh các khác biệt khi lập trình gia công các lỗ trong các block riêng rẽ (O2401), từng bước chuyển động dao được lập trình trong một block chuyển động, và gia công các lỗ đó sử dụng chu kỳ cố định (02402). Phần này chưa đưa ra các giải thích chỉ minh họa giữa hai phương pháp lập trình khác nhau. Dụng cụ cắt là mũi khoan tiêu chuẩn ∅3/16 được dùng để khoan ba lỗ cụt có chiều sâu 0.625 in (Hình 24.1).



***Hình 24.1**. Sơ đồ các lỗ trên chi tiết – các chương trình O2401 và O2402.*

```
O2401 (VI DU 1)
(SU DUNG CAC BLOCK RIENG RE)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X5.9 Y1.89 S900 M03
N4 G43 Z1.0 H01 M08
N5 Z0.1 M08
N6 G01 Z-0.6813 F4.5
N7 G04 P200
N8 G00 Z0.1
N9 X3.87 Y3.4
N10 G01 Z-0.6813
N11 G04 P200
N12 G00 Z0.1
N13 X2.047
N14 G01 Z-0.6813
N15 G04 P200
N16 G00 Z0.1 M09
N17 G28 X2.047 Y3.4 Z1.0
N18 M30
%
```

Chương trình O2402 sử dụng chu kỳ cố định để tăng tính hiệu quả:

```
O2502 (VI DU 2)
(SU DUNG CHU KY CO DINH)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X5.9 Y1.89 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G82 R0.1 Z-0.6813 P200 F4.5
N6 X3.87 Y3.4
N7 X2.047
N8 G80 G28 X2.047 Y3.4  Z1.0 M09
N9 M30
%
```

Chương trình O2401 cần đến 18 block dù chỉ gia công có ba lỗ. Trong chương trình O2402 sử dụng chu kỳ cố định, chỉ cần có 9 block, chương trình này ngắn hơn và dễ đọc hơn, do không có các block lặp lại. Các cải tiến chương trình, cập nhật và các thay đổi có thể được thực hiện dễ dàng hơn, nếu có yêu cầu. Bạn hãy dùng chu kỳ cố định để gia công các lỗ, thậm chí chỉ gia công một lỗ.

LỰA CHỌN CHU KỲ CỐ ĐỊNH

Các chu kỳ cố định do nhà chế tạo bộ điều khiển thiết kế nhằm loại bỏ *sự lặp lại* trong lập trình bằng tay và cho phép dễ dàng *thay đổi dữ liệu chương trình* tại máy CNC.

Ví dụ, một số các lỗ như nhau có thể có chung điểm bắt đầu, chung chiều sâu, chung tốc độ cắt, ... Chỉ vị trí trục X và trục Y là khác nhau giữa các lỗ. Mục đích của chu kỳ cố định là cho phép lập trình các giá trị cần thiết chỉ một lần – *đối với lỗ thứ nhất trên bản vẽ*. Các giá trị chuyên biệt này có tính chế độ trong suốt chu kỳ và không cần lặp lại, trừ khi một vài giá trị thay đổi. Sự thay đổi này thường là vị trí XY của lỗ mới, nhưng các giá trị khác cũng có thể thay đổi đối với lỗ bất kỳ vào thời điểm bất kỳ, đặc biệt đối với các lỗ phức tạp.

Chu kỳ cố định được gọi vào chương trình bằng lệnh G. Fanuc và các hệ điều khiển tương tự hỗ trợ các chu kỳ cố định dưới đây:

G73	Chu kỳ khoan khoét tốc độ cao
G74	Chu kỳ tarô ren trái
G76	Chu kỳ doa chính xác
G80	Xóa chu kỳ cố định (nếu có chu kỳ)
G81	Chu kỳ khoan
G82	Chu kỳ khoan có dwell (tạm dừng)
G83	Chu kỳ khoan khoét
G84	Chu kỳ tarô ren phải
G85	Chu kỳ doa
G86	Chu kỳ doa
G87	Chu kỳ doa ngược
G88	Chu kỳ doa
G89	Chu kỳ doa

Danh sách này chỉ có tính khái quát, nêu ra công dụng *phổ biến* nhất của từng chu kỳ, nhưng có thể không phải là công dụng duy nhất. Ví dụ, một số chu kỳ doa có thể hoàn toàn thích hợp để chuốt, dù không có chu kỳ chuốt lỗ được chuyên biệt một cách trực tiếp. Phần kế tiếp sẽ trình bày định dạng lập trình, các chi tiết của từng chu kỳ và các đề nghị ứng dụng hợp lý của chúng. Bạn hãy coi các chu kỳ cố định là các khả năng được *xây dựng sẵn* bên trong bộ điều khiển.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Định dạng chung của chu kỳ cố định là chuỗi các giá trị tham số được chuyên biệt theo địa chỉ đặc thù (không phải mọi tham số điều khiển đều khả dụng cho từng chu kỳ).

N.. G.. G.. X.. Y.. R.. Z.. P.. Q.. I.. J.. F.. L.. (hoặc K..)

Ý nghĩa của các địa chỉ trong chu kỳ cố định (được trình bày theo thứ tự xuất hiện của chúng).

N = Số block

Trong khoảng N1 đến N9999 hoặc N1 đến N99999, tùy theo hệ điều khiển.

G (lệnh G thứ nhất) = G98 hoặc G99

- G98 trả dao về vị trí Z ban đầu
- G99 trả dao về điểm do địa chỉ R xác định.

G (lệnh G thứ hai) = Số chu kỳ

- Chỉ chọn một trong các lệnh G dưới đây:

G73 G74 G76 G81 G82 G83
G84 G85 G86 G87 G88 G89

X = Vị trí lỗ theo trục X

Giá trị X có thể là tuyệt đối hoặc số gia

Y = Vị trí lỗ theo trục Y

Giá trị Y có thể là tuyệt đối hoặc số gia

R = Vị trí khởi đầu trên trục Z = Mức R

- Vị trí tại đó tốc độ cắt được kích hoạt

Vị trí mức R có thể có giá trị tuyệt đối hoặc số gia

Z = Vị trí kết thúc trên trục Z

- Vị trí kết thúc sự ăn dao

Vị trí chiều sâu Z có thể có giá trị tuyệt đối hoặc số gia

P = Thời lượnG dwell

- Được lập trình theo miligiây (1 giây = 1000 miligiây)

Thời lượng dwell thực tế chỉ áp dụng cho các chu kỳ cố định G76, G82, G88 và G89; cũng có thể áp dụng cho G74 và G84 tùy theo xác lập tham số điều khiển.

- Thời lượng dwell có thể trong khoảng 0.001 – 99999.999 giây, lập trình là P1 đến P99999999.

Q = Địa chỉ Q có hai ý nghĩa

- Khi được dùng với chu kỳ G73 hoặc G83, Q có nghĩa là chiều sâu lỗ khoét
- Khi được dùng với chu kỳ G76 hoặc G87, Q có nghĩa là lượng dịch chuyển dao

Các địa chỉ I và J có thể được sử dụng thay cho địa chỉ Q, tùy theo xác lập tham số điều khiển.

I = Lượng dịch chuyển

- Phải gộp chiều dịch chuyển trục X cho chu kỳ G76 hoặc G87

Dịch chuyển I có thể được dùng thay cho Q

J = Lượng dịch chuyển

- Phải gộp chiều dịch chuyển trục Y cho chu kỳ G70 hoặc G87

Dịch chuyển J có thể được dùng thay cho Q.

F= Tốc độ cắt

- Chỉ áp dụng cho chuyển động cắt

Giá trị này sử dụng đơn vị *in/min* hoặc *mm/min* tùy theo sự lựa chọn hệ đơn vị

L (hoặc K) = Số lần lặp lại chu kỳ

- Phải trong khoảng L0 – L9999 (hoặc K0 – K9999). L1 (K1) là điều kiện mặc định.

CÁC NGUYÊN TẮC CHUNG

Lập trình là công việc có tính kiểm soát cao, nghĩa là có các nguyên tắc, các điều kiện hạn chế, các giới hạn, và các ràng buộc cần tuân thủ. Lập trình CNC không phải là lập trình ngôn ngữ nhưng có nhiều điểm chung. Ví dụ, bạn có thể gặp lập trình *Fanuc* hoặc *Siemens*, lập trình *Cincinati*, lập trình *Mitsubishi* hoặc *Mazatrol*. Các chu kỳ cố định là những chương trình thu nhỏ.

Bạn hãy coi chu kỳ cố định là tập hợp module nhỏ, các module chứa chuỗi các bước hướng dẫn gia công được *lập trình sẵn*. Chu kỳ được gọi là "cố định", do định dạng bên trong của chúng không thể thay đổi. Các lệnh chương trình này liên quan đến kiểu chuyển động dao đặc biệt lặp lại khi gia công. Các nguyên tắc cơ bản và giới hạn liên quan với chu kỳ cố định được tóm tắt như sau:

- Chế độ kích thước tuyệt đối hoặc số gia có thể được thiết lập *trước* khi chu kỳ được lập trình hoặc vào thời điểm bất kỳ *trong* chế độ chu kỳ cố định.
- G90 chọn chế độ tuyệt đối, G91 chọn chế độ số gia.
- Cả G90 và G91 đều có tính chế độ.
- Nếu một trong hai trục X và Y bị bỏ qua trong chế độ chu kỳ cố định, chu kỳ sẽ được thực thi tại vị trí chuyên biệt của một trục và vị trí hiện hành của trục kia.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Nếu bỏ qua cả trục X và Y trong chế độ chu kỳ cố định, chu kỳ sẽ được thực thi ở vị trí dừng sau cắt hiện hành.
- Nếu không có cả lệnh G98 hoặc G99 được lập trình cho chu kỳ cố định, hệ điều khiển sẽ chọn lệnh mặc định theo xác lập của tham số hệ thống (thường là G98).
- Địa chỉ P để gán thời lượng dwell không sử dụng dấu thập phân (không dùng G04), dwell luôn luôn lập trình theo mili giây.
- Nếu LO được lập trình trong block chu kỳ cố định, hệ điều khiển sẽ lưu dữ liệu của block đó để dùng sau này, nhưng không thực thi chúng tại vị trí tọa độ hiện hành.
- Lệnh G80 luôn luôn xóa chu kỳ cố định bất kỳ và tạo ra chuyển động nhanh cho lệnh chuyển động dao kế tiếp bất kỳ. Block chứa G80 sẽ không xử lý chu kỳ cố định.

➲ **Ví dụ:**

```
G80 Z1.125          hoàn toàn như
G80 G00 Z1.125      hoặc
G00 Z1.125
```

Các mã G chuẩn bị thuộc nhóm 01, gồm G00, G01, G02, G03 và G32 là các lệnh chuyển động chính cũng sẽ xóa chu kỳ cố định hoạt động

Chú ý: Trong trường hợp sử dụng lệnh chu kỳ cố định và lệnh chuyển động thuộc *Nhóm 01* trong một block, *thứ tự* lập trình các lệnh này là rất quan trọng:

```
G00 G81 X.. Y.. R.. Z.. P.. Q.. L.. F..
```

chu kỳ cố định sẽ được xử lý, còn trong

```
G81 G00 X.. Y.. R.. Z.. P.. Q.. L.. F..
```

chu kỳ cố định *không* được xử lý, nhưng sẽ thực hiện các chuyển động X và Y, các giá trị khác bị bỏ qua, trừ giá trị tốc độ cắt F được lưu lại. Bằng mọi giá bạn hãy *tránh tình huống này*!

Chương này sẽ trình bày chi tiết các chu kỳ cố định và từng chu kỳ đều có minh họa cấu trúc. Các minh họa sẽ sử dụng ký hiệu đồ họa, mỗi ký hiệu đều có ý nghĩa riêng. Hình 24.2

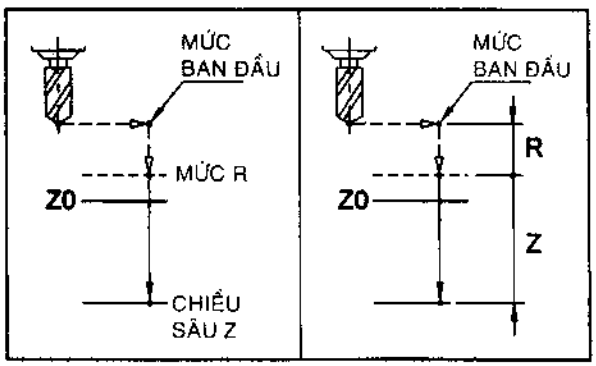
	Chuyển động nhanh và chiều chuyển động
	Chuyển động cắt gọt và chiều chuyển động
	Chuyển động bằng tay và chiều chuyển động
	Dịch chuyển thanh doa và chiều dịch chuyển
	Tọa độ được lập trình
Q	Giá trị dịch chuyển / chiều sâu khoét lỗ
d	Giá trị khoảng hở
CW / CCW	Sự quay trục chính
OSS	Dừng trục chính định hướng
DWELL	Hàm dwell được thực thi

Hình 24.2. *Ký hiệu và viết tắt được dùng trong các minh họa chu kỳ cố định*

nêu rõ ý nghĩa của từng ký hiệu được dùng trong các minh họa.

GIÁ TRỊ TUYỆT ĐỐI VÀ GIÁ TRỊ SỐ GIA

Tương tự mọi quy trình gia công khác, gia công lỗ cũng sử dụng chế độ lập trình tuyệt đối G90 hoặc số gia G91 để lập trình các chu kỳ cố định. Sự lựa chọn sẽ tác động chủ yếu đến vị trí XY của lỗ gia công, mức R, và chiều sâu Z (Hình 24.3).



Hình 24.3. *Các giá trị nhập theo chế độ tuyệt đối và số gia cho chu kỳ cố định.*

Trong phương pháp lập trình tuyệt đối, mọi giá trị đều quan hệ với điểm gốc, *zero chương trình*. Trong phương pháp số gia, vị trí XY của một lỗ là khoảng cách từ vị trí XY của lỗ trước đó. Mức R là khoảng cách từ giá trị Z *cuối cùng*, được thiết lập trước khi gọi chu kỳ, đến vị trí kích hoạt tốc độ cắt. Giá trị chiều sâu Z là khoảng cách giữa mức R và điểm kết thúc chuyển động cắt. Khi khởi đầu chu kỳ cố định bất kỳ, chuyển động dao đến mức R luôn luôn theo chế độ nhanh.

CHỌN MỨC BAN ĐẦU

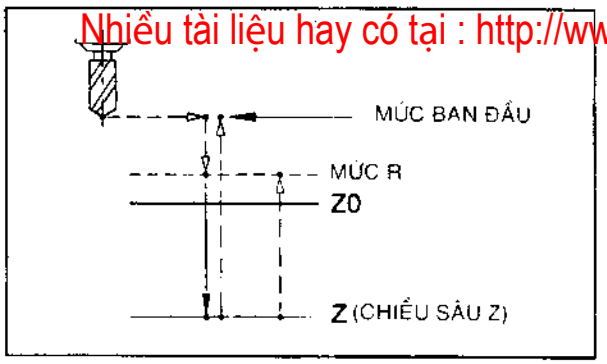
Có hai lệnh chuẩn bị điều khiển dao trở về theo trục Z khi *hoàn tất* chu kỳ cố định.

G98	... làm cho dao trở về vị trí *ban đầu* = *gán địa chỉ Z* .
G99	... làm cho dao trở về vị trí *mức R* = *gán địa chỉ R*.

Các mã G98 và G99 chỉ dùng cho chu kỳ cố định. Chức năng chính của chúng là vòng tránh chướng ngại giữa các lỗ trong sơ đồ gia công. Các chướng ngại có thể gồm kẹp định vị, đồ gá định vị, các bộ phận nhô lên của chi tiết, các vùng không gia công, phụ tùng máy,... Nếu không có các lệnh này, chu kỳ có thể bị xóa và dao dịch chuyển đến vị trí an toàn. Chu kỳ sau đó có thể được khôi phục lại. Với các lệnh G98 và G99, có thể tránh các chướng ngại mà

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 24.4. *Lựa chọn mức ban đầu cho chu kỳ cố định.*

không xóa chu kỳ cố định, lập trình sẽ hiệu quả hơn.

Theo định nghĩa, mức ban đầu là giá trị tuyệt đối của hệ tọa độ *trục Z cuối cùng* trong chương trình, trước khi gọi chu kỳ cố định (Hình 24.4).

Từ quan điểm thực tiễn, bạn hãy luôn luôn chọn vị trí này theo mức *an toàn* có cân nhắc kỹ. Điều quan trọng là mức dao lùi lại khi lệnh G98 có hiệu lực phải *cao hơn mọi chướng ngại*. Bạn hãy sử dụng mức ban đầu một các thận trọng sao cho dụng cụ cắt không bị va chạm trong khi chuyển động nhanh sự va chạm xảy ra khi dao cắt tiếp xúc không mong muốn với chi tiết, đồ gá, hoặc bộ phận máy.

➲ **Ví dụ về lập trình mức ban đầu:**

Đoạn chương trình dưới đây là ví dụ điển hình về lập trình vị trí mức ban đầu:

```
...
N11 G90 G54 G00 X10.0 Y4.5 S1200 M03
N12 G43 Z2.0 H01 M08 (INITIAL LEVEL AT Z2.0)
N13 G98 G81 X10.0 Y4.5 R0.1 Z-0.82 F5.0
N14 ...
...
...
N20 G80
...
```

Chu kỳ cố định (G81 trong ví dụ) được gọi trong block N13. Giá trị trục Z cuối cùng *trước* block này được lập trình trong block N12 là Z 2.0. Đây là xác lập vị trí ban đầu, *hai inch phía trên* mức Z0 của chi tiết. Mức Z này có thể được chọn theo chiều cao tiêu chuẩn chung, nếu các chương trình là ổn định, hoặc có thể khác nhau giữa các chương trình. Ở đây, an toàn là yếu tố quyết định.

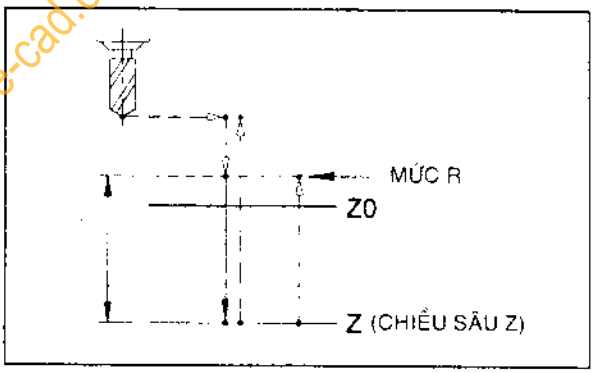
Khi chu kỳ cố định được áp dụng, không thể thay đổi mức Z ban đầu, trừ khi chu kỳ được xóa trước với lệnh G80. Sau đó mức Z ban đầu có thể thay đổi và gọi lại chu kỳ theo yêu cầu.

Mức Z cần được lập trình theo giá trị tuyệt đối, trong chế độ G90.

CHỌN MỨC R

Vị trí dao cắt từ đó bắt đầu cắt gọt cũng được chuyên biệt theo trục Z. Điều đó có nghĩa là block chu kỳ cố định đòi hỏi *hai vị trí* liên quan đến trục Z, một là điểm bắt đầu, tại đó dao bắt đầu cắt gọt, và thứ hai là điểm cuối, biểu thị chiều sâu lỗ. Các nguyên tắc lập trình cơ bản không cho phép một trục được lập trình hơn một lần trong một block. Do đó, cần có sự điều chỉnh trong thiết kế bộ điều khiển để hỗ trợ hai giá trị Z cần thiết cho chu kỳ cố định. Giải pháp đơn giản là một trong hai giá trị Z phải được *thay bằng địa chỉ khác*.

Do trục Z có quan hệ chặt chẽ với chiều sâu, sẽ giữ nguyên ý nghĩa này trong mọi chu kỳ. Địa chỉ thay thế được dùng cho vị trí Z của dao cắt từ đó bắt đầu áp dụng tốc độ cắt. Địa chỉ này sử dụng ký hiệu là chữ R, thường được gọi là *mức R*. Bạn hãy coi mức R là "*chuyển động nhanh đến điểm bắt đầu*" trong đó luôn luôn nhấn mạnh "*chuyển động nhanh đến*" và chữ "*R*" (Hình 24.5).



Hình 24.5. *Chọn mức R cho chu kỳ cố định*

Mức R không chỉ là điểm bắt đầu tốc độ cắt, mà còn là mức Z để dụng cụ cắt lùi lại, nếu lệnh chuẩn bị G99 được lập trình. Nếu G98 được lập trình, sự lùi lại sẽ đến mức ban đầu. Ngoại lệ là chu kỳ doa ngược G87 sẽ được trình bày ở phần sau. Chu kỳ này không sử dụng chế độ lùi dao G99, chỉ dùng G98. Tuy nhiên, đối với mọi chu kỳ, giá trị mức R cần được chọn rất cẩn thận. Các giá trị phổ biến nhất là 0.04–0.20 in (1-5 mm) trên Z0 của chi tiết. Sự gá lắp chi tiết cũng phải được xem xét, nếu cần, có thể điều chỉnh lại.

Mức R thường tăng lên khoảng 3 –4 lần đối với các nguyên công tarô ren sử dụng chu kỳ G74 và G84 để cho phép tăng tốc độ cắt đến giá trị cực đại.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

➲ **Ví dụ về lập trình mức R**

```
...
N29 G90 G00 G54 X6.7 Y8.0 S850 M03
N30 G43 Z1.0 H04 M08 (INITIAL LEVEL IS 1.0)
N31 G99 G85 R0.1 Z-1.6 F9.0   LEVEL IS 0.1)
N32 ...
...
...
N45 G80
```

Mức ban đầu trong ví dụ này là block N30, xác lập theo Z1.0. Mức R được xác lập trong block N31 (block gọi chu kỳ) là 0.100 inch. Trong cùng block này, lệnh G99 được lập trình và không thay đổi trong suốt chu kỳ đó. Điều này có nghĩa là vị trí dao phía trên zero chi tiết là 0.100 in khi bắt đầu và khi kết thúc chu kỳ. Khi dụng cụ cắt chuyển động từ một lỗ đến lỗ kế tiếp, sẽ chỉ chạy theo các trục XY với mức Z *cao* 0.100 in phía trên bề mặt chi tiết gia công.

Vị trí mức R thường *thấp hơn* vị trí mức ban đầu. Nếu như mức này trùng nhau, các điểm bắt đầu và kết thúc là bằng vị trí ban đầu. Mức R thường được lập trình theo giá trị tuyệt đối trong chế độ G90, nhưng có thể đổi sang chế độ số gia G91, nếu thực sự cần thiết.

TÍNH TOÁN CHIỀU SÂU Z

Chu kỳ cố định phải có chiều sâu cắt, tại đó dụng cụ cắt *dừng* ăn dao. Chiều sâu được lập trình theo địa chỉ Z trong block chu kỳ. Điểm kết thúc chiều sâu cắt được lập trình theo giá trị Z, thường *thấp hơn* mức R và mức *ban đầu*, riêng chu kỳ G87 là ngoại lệ.

Để đạt được chương trình chất lượng cao, bạn hãy cố gắng lập trình chiều sâu Z được *tính toán chính xác*, không dự đoán và không làm tròn giá trị đó. Ví dụ, chiều sâu tính toán là 0.6979, bạn *không được làm tròn* đến 0.6980 hoặc 0.70. Đây không phải là vấn đề thói quen hay phong cách lập trình, mà là nguyên tắc để bảo đảm độ chính xác gia công. Với phương pháp này, bạn có thể dễ dàng xác định nguyên nhân của vấn đề phát sinh trong khi gia công.

Tính toán chiều sâu Z dựa trên các chuẩn dưới đây:

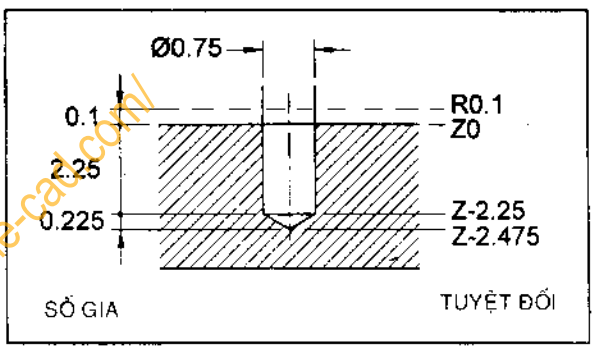
- Kích thước lỗ trên bản vẽ (đường kính và chiều sâu).
- Phương pháp lập trình tuyệt đối hay số gia.
- Kiểu dụng cụ cắt + chiều dài đỉnh cắt.
- Chiều dày của phôi hoặc chiều sâu đường kính toàn phần của lỗ.
- Các khoảng hở được chọn – trên và dưới phôi (khoảng hở dưới phôi dùng cho các lỗ suốt).

Trên trung tâm gia công đứng, Z0 thường được lập trình ở mặt chi tiết gia công hoàn tất. Trong trường hợp đó, giá trị tuyệt đối của địa chỉ Z luôn luôn được lập trình theo giá trị *âm*. Chắc bạn còn nhớ, *không có* dấu trong địa chỉ trục có nghĩa là giá trị dương của địa chỉ đó. Phương pháp này có một ưu điểm rõ rệt. Nếu nhà lập trình quên viết dấu âm (trừ), giá trị chiều sâu sẽ tự động trở thành giá trị dương. Khi đó, dụng cụ cắt sẽ chuyển động ra xa chi tiết, thường đến khu vực an toàn. Chương trình gia công sẽ không chính xác, nhưng có thể dễ dàng chỉnh sửa, chỉ tốn ít thời gian.

➲ **Ví dụ về tính toán chiều sâu Z**

Để minh họa ví dụ thực tiễn về tính toán chiều sâu Z, bạn hãy xét chi tiết lỗ trên Hình 24.6. Bạn sẽ dùng mũi khoan Ø 0.75 inch để gia công lỗ có chiều sâu toàn phần là 2.25 inch. Nếu sử dụng mũi khoan xoắn tiêu chuẩn, bạn cần xét phần đỉnh mũi khoan, góc đỉnh thường là 118-120^0, do đó phải cộng thêm 0.225 inch vào chiều sâu đã chuyên biệt:

```
0.3 x 0.75 = 0.225
2.25 + 0.225 = 2.475
```



Hình 24.6. *Tính toán chiều sâu Z cho chu kỳ khoan cố định.*

Dựa trên kết quả này, tổng chiều sâu Z 2.475 inch có thể được viết trong chương trình:

```
G99 G83 X9.0 Y-4.0 R0.1 Z-2.475 Q1.125 F12.0
```

Chu kỳ khoan khoét G83 được dùng trong ví dụ này để đạt được kết quả gia công theo yêu cầu, dù các giá trị R và Z không thay đổi đối với các chu kỳ G81, G82 hoặc G73. Tính toán chiều dài đỉnh dụng cụ cắt sẽ được trình bày trong Chương 25.

CÁC CHU KỲ CỐ ĐỊNH

Để hiểu cơ chế hoạt động của từng chu kỳ cố định, điều quan trọng là biết cấu trúc bên trong của chu kỳ và các chi tiết về định dạng lập trình. Dưới đây sẽ đánh giá từng chu kỳ cố định một cách chi tiết. Tiêu đề chu kỳ phản ánh định dạng lập trình cơ bản của chu kỳ đó. Phần này còn trình bày các ứng dụng phổ biến của từng chu kỳ gia công.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Mọi chi tiết này đều quan trọng, giúp bạn hiểu bản chất của từng chu kỳ và phương pháp chọn chu kỳ để có kết quả gia công tối ưu. Kiến thức về cấu trúc bên trong chu kỳ còn giúp bạn thiết kế các chu kỳ đặc thù riêng, đặc biệt trong lĩnh vực lập trình macro chuyên biệt.

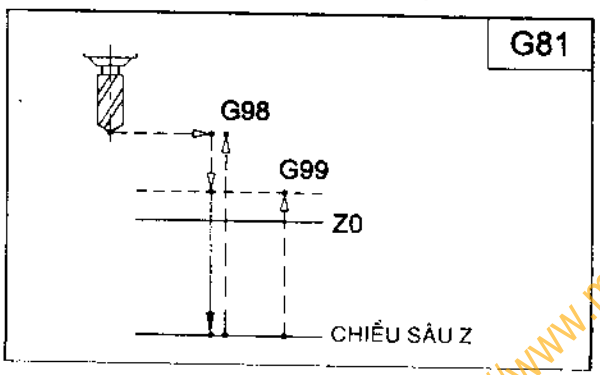
G81 - Chu kỳ khoan

G98 (G99) G81 X.. Y.. R.. Z.. F..

Bước	Chu kỳ G81
1	Chuyển động nhanh đến vị trí XY
2	Chuyển động nhanh đến *mức R*
3	Chuyển động cắt gọt đến *chiều sâu Z*
4	Lùi nhanh đến *mức ban đầu* (với G 98) *hoặc* lùi nhanh đến *mức R* (với G99)

SỬ DỤNG CHU KỲ G81 – Hình 24.7:

Chủ yếu để khoan, khoan tâm, không yêu cầu dwell ở chiều sâu Z. Nếu dùng để doa, chu kỳ G81 sẽ gây ra các vết xước trên vách lỗ khi rút dao ra ngoài.



Hình 24.7. *Chu kỳ cố định G81 – thường dùng để khoan lỗ*

G82 - Chu kỳ khoan điểm

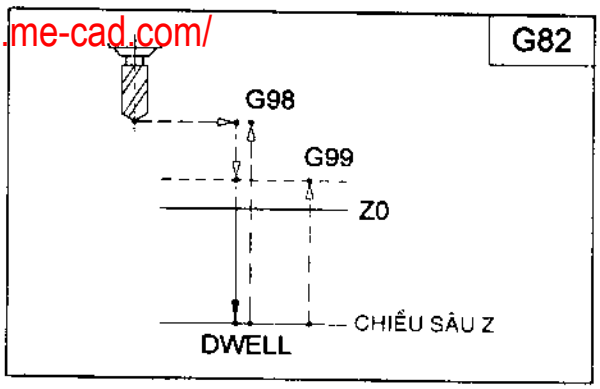
G98 (G99) G82 X.. Y.. R.. Z.. P.. F..

Bước	Chu kỳ G82
1	Chuyển động nhanh đến vị trí XY
2	Chuyển động nhanh đến *mức R*
3	Chuyển động cắt gọt đến *chiều sâu Z*
4	Dwell tại chiều sâu Z, tính theo miligiây (P)
5	Lùi nhanh về *mức ban đầu* (với G98) *hoặc* lùi nhanh đến *mức R* (với G99)

SỬ DỤNG CHU KỲ G82 - Hình 24.8:

Khoan với dwell, dao tạm dừng ở đáy lỗ. Dùng để khoan tâm, khoét, lả miệng,... yêu cầu độ bóng ở đáy lỗ. Thường dùng khi cần lập trình tốc độ trục chính chậm.

Nếu dùng để doa, vách lỗ sẽ có vết xước khi rút dao ra ngoài.



Hình 24.8. *Chu kỳ cố định G82 - chủ yếu được sử dụng đối với khoan điểm*

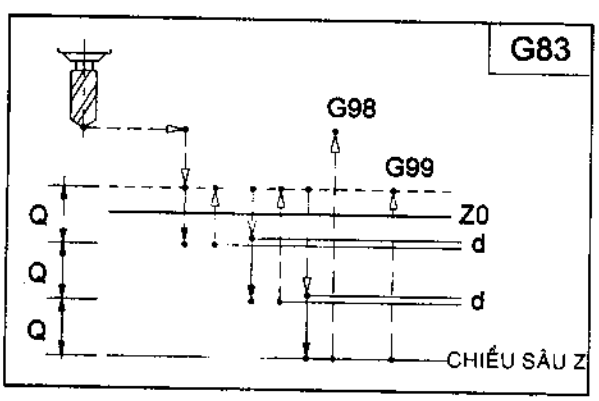
G83 - Chu kỳ khoan lỗ sâu - Tiêu chuẩn

G98 (G99) G83 X.. Y.. R.. Z.. Q.. F..

Bước	Chu kỳ G83
1	Chuyển động nhanh đến vị trí XY.
2	Chuyển động nhanh đến *mức R*.
3	Chuyển động cắt gọt đến *chiều sâu Z* theo *giá trị Q*.
4	Lùi dao nhanh đến mức R.
5	Chuyển động nhanh đến chiều sâu đã khoan với khoảng hở theo tham số hệ thống.
6	Lặp lại các bước 3, 4, 5 cho đến khi đạt được chiều sâu Z lập trình.
7	Lùi nhanh về mức ban đầu (G98) *hoặc* lùi nhanh về mức R (G99)

SỬ DỤNG CHU KỲ G83 Hình 24.9

Để khoan lỗ sâu, còn gọi là khoan khoét, mũi khoan còn được lùi lên trên chi tiết (đến vị trí hở) sau khi khoan đến chiều sâu xác định. Bạn hãy so sánh chu kỳ này với chu kỳ khoan lỗ sâu tốc độ cao G73.



Hình 24.9. *Chu kỳ cố định G83 – dùng để khoan lỗ sâu (chu kỳ này lùi dao đến mức R sau mỗi lần khoét)*

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

G73 - Chu kỳ khoan lỗ sâu - Tốc độ cao

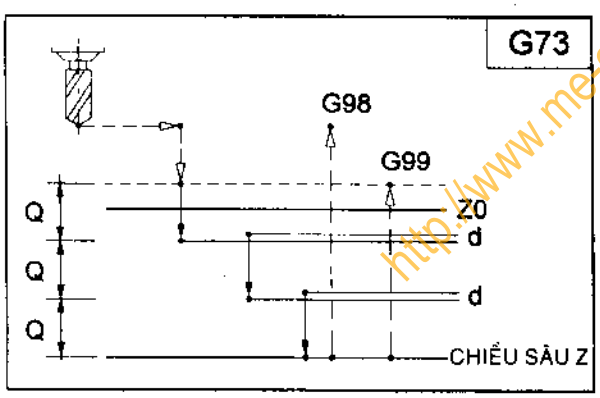
G98 (G99) G73 X.. Y.. R.. Z.. Q.. F..

Bước	Chu kỳ G73
1	Chuyển động nhanh đến vị trí XY.
2	Chuyển động nhanh đến *mức R*.
3	Chuyển động cắt gọt đến *chiều sâu Z* theo *giá trị Q*.
4	Lùi nhanh theo giá trị khoảng hở (khoảng hở được xác lập theo tham số hệ thống).
5	Chuyển động cắt gọt theo trục Z bằng *lượng Q* cộng với khoảng hở.
6	Lặp lại các bước 4 và 5 cho đến khi đạt được *chiều sâu Z* lập trình.
7	Lùi nhanh đến *mức ban đầu* (với G98) *hoặc* lùi nhanh đến *mức R* (với G99)

SỬ DỤNG CHU KỲ G73 – Hình 24.10

Để khoan lỗ sâu, còn gọi là khoan khoét, khi sự bẻ gãy phoi là quan trọng hơn so với lùi hoàn toàn dao cắt ra khỏi lỗ. Chu kỳ G73 thường dùng cho các mũi khoan dài với sự lùi dao hoàn toàn là không quá quan trọng.

Chu kỳ cố định G73 hơi nhanh hơn chu kỳ G83 do đó có tên là "tốc độ cao", thời gian tiết kiệm do không lùi dao về mức R sau mỗi lần khoét. Bạn hãy so sánh chu kỳ này với chu kỳ khoan lỗ sâu tiêu chuẩn G83.



Hình 24.10. *Chu kỳ cố định G73 khoan lỗ sâu (không lùi dao đến mức R sau mỗi lần khoét).*

Tính toán số lần khoét

Khi sử dụng các chu kỳ G83 và G73 trong chương trình bạn phải có ý tưởng về số lần khoét dao sẽ thực hiện trong từng lỗ. Số lần khoét đủ cho hàng trăm hoặc hàng ngàn lỗ sẽ tạo ra sự tổn thất thời gian có thể rất lớn. Bạn hãy tránh quá nhiều lần khoét cho một lỗ. Về nguyên tắc, hoàn toàn có thể *tính toán* số lần khoét.

Tính toán số lần khoét có thể áp dụng cho cả hai chu kỳ G83 và G73. Tính toán này dựa trên giá trị của địa chỉ Q và khoảng cách toàn phần từ mức R đến chiều sâu Z, không tính từ mặt trên của chi tiết gia công. Bạn hãy chia khoảng cách đó cho số lần khoét ở từng vị trí lỗ. Số lần khoét trong chu kỳ phải là số nguyên, phần thập phân cần được làm tròn *tăng* đến số nguyên gần nhất.

➲ Ví dụ 1 - Đơn vị Anh:

```
G90 G98 G83 X.. Y.. R0.1 Z-1.4567 Q0.45 F..
```

Trong ví dụ này, khoảng cách giữa mức R và chiều sâu Z là 1.5567 in, giá trị Q là 0.45 in, do đó số lần khoét sẽ là:

1.5567 / 0.45 = 3.4593333

Kết quả có quá nhiều chữ số thập phân, hoàn toàn không thể sử dụng do hầu hết các bộ điều khiển chỉ chấp nhận *bốn* chữ số thập phân theo hệ Anh và ba chữ số thập phân cho hệ mét, do đó cần *làm tròn tăng* đến số nguyên gần nhất là bốn.

Kết quả, mỗi lỗ cần 4 lần khoét. Chiều sâu lỗ không thể thay đổi, do đó phương pháp khả dụng duy nhất là thay đổi mức R và/ hoặc chiều sâu của từng lần khoét. Mức R thường ở sát mặt trên của chi tiết, do đó hơi khó thay đổi. Như vậy, chỉ còn giá trị Q, chiều sâu mỗi lần khoét. Khi tăng Q, số lần khoét sẽ giảm, và ngược lại, nếu giảm giá trị Q số lần khoét sẽ tăng.

➲ Ví dụ 2 – Đơn vị mét:

```
G90 G99 G73 X.. Y.. R2.5 Z-42.5 Q15.0 F..
```

Trong ví dụ này, khoảng cách giữa *mức R* và *chiều sâu Z* là 45 mm, và giá trị Q là 15mm. Số lần khoét là 45/15 = 3. Không cần làm tròn vì đây là số nguyên, số lần khoét được thực thi cho từng lỗ là 3.

Để *tăng* số lần khoét,
bạn hãy *giảm* giá trị Q hiện hành.

Để *giảm* số lần khoét,
bạn hãy *tăng* giá trị Q hiện hành

Xác lập giá trị Q sẽ chính xác hơn, nếu được tính toán thực sự thay vì phỏng đoán. Để đạt được số lần khoét chính xác, bạn hãy chia tổng khoảng cách giữa mức R và chiều sâu Z cho số lần khoét được yêu cầu. Kết quả sẽ là giá trị Q được lập trình cho số lần khoét được chọn. Nếu cần làm tròn, bạn hãy làm tròn *tăng*.

➲ Ví dụ 3 – Đơn vị mét:

Trong ví dụ này, khoảng cách giữa mức R và chiều sâu Z là 56 mm, yêu cầu chính xác ba lần khoét, tính toán chiều sâu từng lần khoét tương đối đơn giản:

56 / 3 = 18.666667

Kết quả tính toán phải được làm tròn đến 18.667 hoặc 18.666. Dù đây chỉ là *một* micron

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

(0.001mm) nhưng sẽ có khác biệt lớn tùy theo số được làm tròn. Nếu chỉ yêu cầu 3 lần khoét, làm tròn *tăng*, đến Q18.667

Cắt lần 1: 18.667
Cắt lần 2: 18.667
Cắt lần 3: 18.666

Tổng cộng: 56mm

Nếu làm tròn giảm, đến Q 18.666, số lần khoét sẽ là 4 và thực tế hầu như không cắt gọt trong lần khoét cuối cùng:

Cắt lần 1: 18.666
Cắt lần 2: 18.666
Cắt lần 3: 18.666
Cắt lần 4: 0.002

Tổng cộng 56 mm

➲ **Ví dụ 4 - Đơn vị Anh:**

Trong ví dụ này, khoảng cách giữa mức R và chiều sâu Z là 2.5 inch và cần 4 lần khoét.

Q = 2.5 / 4 = 0.625

Trong trường hợp này không cần làm tròn, và Q0.625 sẽ *chính xác* bốn lần khoét với chiều sâu như nhau.

Giá trị khoan khoét của xác lập Q không thể thay đổi cho một lỗ, mọi lần khoét trong một lỗ sẽ có cùng chiều sâu, với ngoại lệ có thể ở lần khoét cuối cùng. Nếu lượng khoét cuối cùng lớn hơn so với khoảng cách còn lại của *chiều sâu Z* được lập trình, sẽ chỉ khoan khoảng cách đó.

Chiều sâu khoét không được vượt quá vị trí tọa độ *chiều sâu Z*.

Giá trị Q lập trình có thể được xử lý theo phương pháp sáng tạo. Bằng cách thay đổi giá trị Q một cách khéo léo, có thể đạt được các kết quả đặc biệt, chẳng hạn vị trí chính xác của đỉnh dụng cụ cắt khi khoan vào chi tiết. Phương pháp này được trình bày chi tiết trong Chương 25.

Để xác định chiều sâu khoét "tốt nhất", bạn hãy xét các điều kiện gia công tổng thể. Độ cứng vững khi gá lắp, đồ gá chi tiết, thiết kế của dụng cụ cắt, tính gia công cắt gọt của vật liệu và các yếu tố khác.

Mục đích của khoan khoét là làm cho chương trình gia công hiệu quả trong các điều kiện an toàn. Điều đó có nghĩa là lập trình giá trị Q sâu nhất một cách hợp lý và có tính thực tiễn trong quy trình công nghệ cụ thể. Bạn cần nhớ có hai chu kỳ khả dụng, G84 tiêu chuẩn và G73 đôi khi ít được chú ý.

G84 - Chu kỳ tarô ren - Tiêu chuẩn

G98 (G99) G84 X.. Y.. R.. Z.. F..

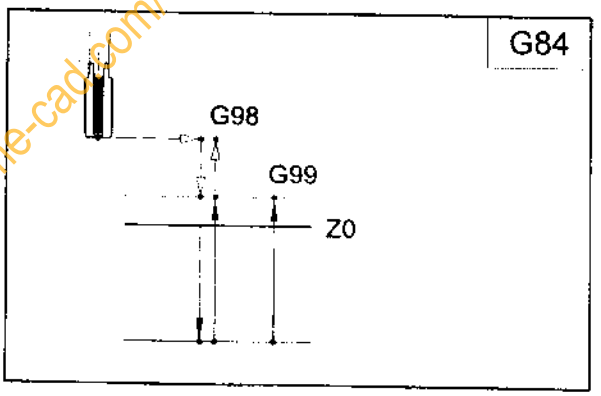
Chuỗi thứ tự của chu kỳ G84 dựa trên sự quay trục chính *bình thường* ban đầu, chuyên biệt theo M03.

Tarô phải là *ren phải* để dùng cho chu kỳ G84 với sự quay trục chính H03 có hiệu lực.

Bước	Chu kỳ G84
1	Chuyển động nhanh đến vị trí XY
2	Chuyển động nhanh đến *mức R*
3	Chuyển động cắt gọt đến *chiều sâu Z*
4	Dừng quay trục chính
5	Trục chính quay ngược (M04) và cắt gọt lùi về *mức R*
6	Dừng quay trục chính
7	Trục chính quay bình thường (M03) và lùi về *mức ban đầu* (với G98) *hoặc* vẫn giữ *mức R* (với G99)

SỬ DỤNG CHU KỲ G84 – Hình 24.11

Chỉ tarô ren phải khi bắt đầu chu kỳ, sự quay trục chính bình thường (quay thuận) với lệnh M03 phải có hiệu lực



Hình 24.11. *Chu kỳ G84 – chỉ dùng để tarô ren phải*

G74 - Chu kỳ tarô ren - Ngược

G98 (G99) G74 X.. Y.. R.. Z.. F..

Thứ tự của chu kỳ G74 dựa trên sự quay *ngược* ban đầu của trục chính – M04.

Tarô là loại *ren trái* để dùng trong chu kỳ G74 và sự quay trục chính M04 phải có hiệu lực.

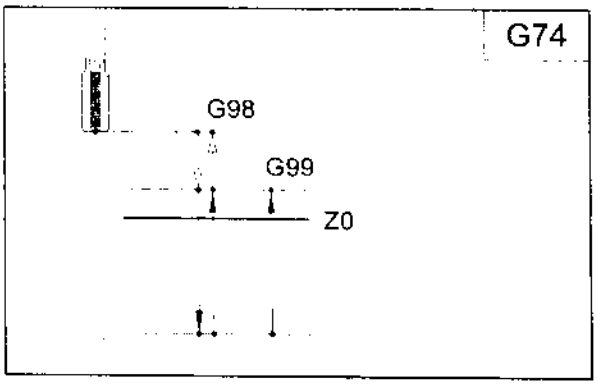
Bước	Chu kỳ G74
1	Chuyển động nhanh đến vị trí XY
2	Chuyển động nhanh đến *mức R*
3	Chuyển động cắt gọt đến chiều sâu Z
4	Dừng quay trục chính
5	Trục chính quay thuận (M03) và cắt gọt lùi về *mức R*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

6	Dừng quay trục chính
7	Trục chính quay ngược (M04) và lùi về *mức ban đầu* (với G98) *hoặc* vẫn giữ *mức R* (với G99)

SỬ DỤNG CHU KỲ G74 – Hình 24.12

Chỉ tarô ren trái. Khi bắt đầu chu kỳ, sự quay trục chính là quay ngược. M04 phải có hiệu lực



***Hình 24.12.** Chu kỳ G74 – chỉ dùng để tarô ren trái*

Chương 25 sẽ trình bày các kỹ thuật gia công lỗ, kể cả tarô ren.

Phần dưới đây chỉ trình bày các vấn đề lập trình và tarô ren quan trọng nhất, áp dụng cho cả chu kỳ G84 và G74.

- Mức R trong chu kỳ tarô ren phải cao hơn so với trong các chu kỳ khác để bảo đảm ổn định sự ăn dao trong khi gia tốc.
- Lựa chọn tốc độ cắt cho tarô là rất quan trọng. Trong tarô ren, có quan hệ trực tiếp giữa tốc độ trục chính và đầu mối ren, quan hệ này phải duy trì trong suốt thời gian gia công.
- Các công tắc override trên bảng điều khiển dùng cho tốc độ trục chính và tốc độ cắt sẽ không có hiệu lực trong khi xử lý chu kỳ G84 hoặc G74.
- Chuyển động cắt ren (vào hoặc ra khỏi lỗ) sẽ hoàn tất kể cả khi nhấn phím duy trì ăn dao trong khi xử lý chu kỳ tarô ren, vì các lý do an toàn.

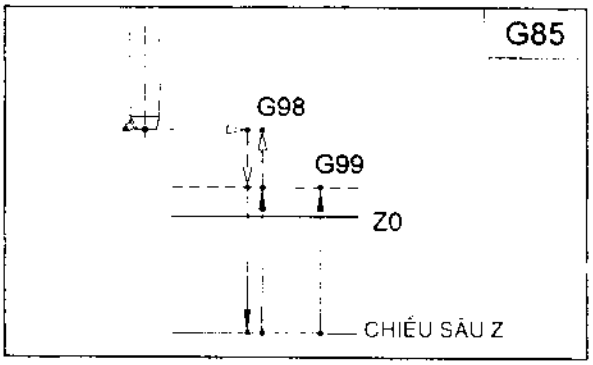
G85 – Chu kỳ doa

G 98 (G99) G85 X.. Y.. R.. Z.. F..

Bước	Chu kỳ G85
1	Chuyển động nhanh đến vị trí XY
2	Chuyển động đến *mức R*
3	Chuyển động cắt gọt đến *chiều sâu Z*
4	Chuyển động cắt gọt lùi về *mức R*
5	Lùi nhanh đến *mức ban đầu* (với G98) *hoặc* giữ nguyên ở *mức R* (với G99)

SỬ DỤNG CHU KỲ G85 – Hình 24.13

Chu kỳ doa G85 thường dùng cho các nguyên công doa và chuốt. Chu kỳ này được dùng trong các trường hợp chuyển động dao vào và ra các lỗ để tăng độ bóng bề mặt lỗ, dung sai kích thước và / hoặc độ đồng tâm... Nếu sử dụng chu kỳ G85 để doa, bạn cần nhớ trên một số chi tiết có thể có một lượng nhỏ vật liệu bị cắt gọt khi lùi dao ra khỏi lỗ. Đặc tính vật lý này là do áp suất của dao cắt trong khi lùi dao. Nếu độ bóng bề mặt bị giảm do lùi dao,



***Hình 24.13.** Chu kỳ G85 – dùng cho chuốt và doa*

bạn hãy thử sử dụng chu kỳ doa khác.

G86 – Chu kỳ doa

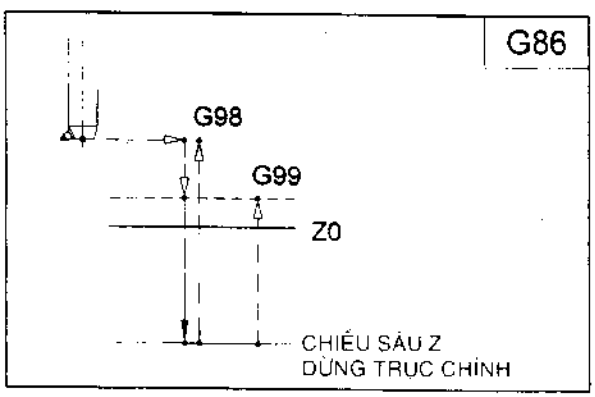
G98 (G99) G86 X.. Y.. R.. Z.. F..

Bước	Chu kỳ G86
1	Chuyển động nhanh đến vị trí XY với trục chính quay.
2	Chuyển động nhanh đến *mức R*.
3	Chuyển động cắt gọt đến *chiều sâu Z*.
4	Dừng quay trục chính.
5	Lùi dao nhanh đến *mức ban đầu* (với G93) *hoặc* lùi dao nhanh đến *mức R* (với G99).

SỬ DỤNG CHU KỲ G86 – Hình 24.14

Để doa thô các lỗ hoặc các lỗ cần tiếp tục gia công. Chu kỳ này rất giống chu kỳ G81. Sự khác biệt là trục chính dừng ở đáy lỗ.

Ghi chú: Mặc dầu chu kỳ này tương tự chu kỳ G81 nhưng vẫn có các đặc tính riêng. Trong chu kỳ khoan tiêu chuẩn G81 dao lùi lại trong khi trục chính của máy công cụ đang quay, nhưng trong chu kỳ G86, trục chính dừng lại. Không được phép dùng chu kỳ G86 để khoan, chẳng hạn



***Hình 24.14.** Chu kỳ G86 dùng cho các nguyên công doa thô và bán tinh.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

nhằm tiết kiệm thời gian – do phoi vụn dính ở cạnh cắt xoắn của mũi khoan có thể làm hư hại bề mặt khoan hoặc chính mũi khoan.

G87 – Chu kỳ doa ngược

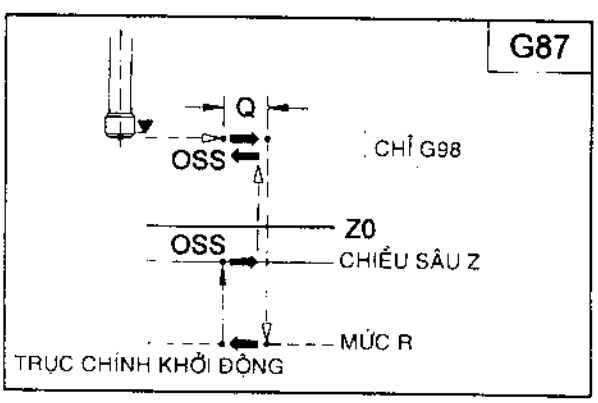
Có hai định dạng lập trình khá dụng cho chu kỳ doa ngược, G87 – định dạng thứ nhất (sử dụng Q) thông dụng hơn so với định dạng thứ hai (sử dụng I và J).

G98 G87 X.. Y.. R.. Z.. Q.. F..
G98 G87 X.. Y.. R.. Z.. I.. J.. F..

Bước	Chu kỳ G87
1	Chuyển động nhanh đến vị trí XY
2	Dừng quay trục chính
3	Định hướng trục chính
4	Dịch chuyển *ra* theo giá trị Q *hoặc* dịch chuyển theo giá trị và chiều của I và J
5	Chuyển động nhanh đến *mức R*
6	Dịch chuyển *vào* theo giá trị Q *hoặc* dịch chuyển theo chiều ngược của I và J
7	Trục chính quay thuận (M03)
8	Chuyển động cắt gọt đến chiều *sâu Z*
9	Trục chính dừng lại
10	Định hướng trục chính
11	Dịch chuyển *ra* theo giá trị Q *hoặc* dịch chuyển theo giá trị và chiều của I và J
12	Lùi dao nhanh đến mức ban đầu
13	Dịch chuyển *vào* theo giá trị Q *hoặc* dịch chuyển theo chiều ngược của I và J
14	Trục chính quay

SỬ DỤNG CHU KỲ G87 – Hình 24.15

Đây là chu kỳ đặc biệt, chỉ sử dụng cho một số (không phải tất cả) nguyên công doa ngược. Công dụng thực tiễn của chu kỳ này rất hạn chế, do các yêu cầu dụng cụ cắt và gá lắp. Bạn chỉ nên dùng chu kỳ G87 nếu tổng chi phí có tính kinh tế. Trong nhiều trường hợp, bạn có thể chọn phương pháp dảo ngược chi tiết trong nguyên công thứ cấp.



Hình 24.15. *Chu kỳ G87 chỉ dùng cho doa ngược*

Chú ý: Dao doa phải được gá lắp rất cẩn thận, cần được xác lập trước để tương hợp với đường kính được yêu cầu doa ngược. Lưỡi cắt phải được xác lập trong chế độ định hướng trục chính, quay mặt ngược chiều với chiều dịch chuyển.

G99 không được dùng với chu kỳ G87

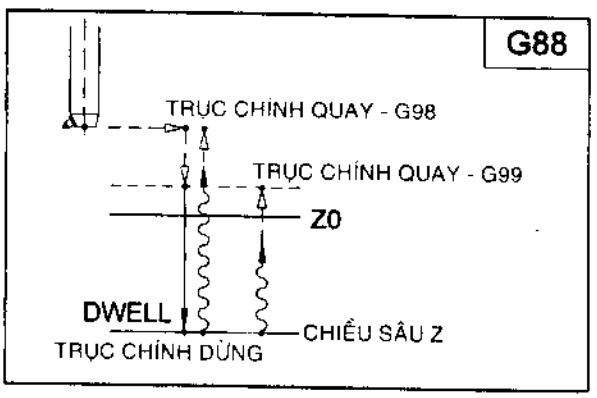
G88 – Chu kỳ doa

G98 (G99) G88 X.. Y.. R.. Z.. P.. F..

Bước	Chu kỳ G88
1	Chuyển động nhanh đến vị trí XY
2	Chuyển động nhanh đến *mức R*
3	Chuyển động cắt gọt đến *chiều sâu Z*
4	Dwell tại chiều sâu này – tính theo mili giây (P..)
5	Dừng quay trục chính (tạo điều kiện feedhold, người vận hành CNC chuyển sang chế độ thao tác bằng tay và thực hiện tác vụ bằng tay, sau đó chuyển trở lại chế độ nhớ). SỰ KHỞI ĐỘNG CHU KỲ sẽ trở về chế độ bình thường
6	Lùi dao nhanh đến *mức ban đầu* (với G98) *hoặc* lùi dao nhanh đến *mức R* (với G99)
7	Trục chính quay

SỬ DỤNG CHU KỲ G88 - Hình 24.16

Chu kỳ G88 ít được sử dụng, chỉ giới hạn cho các nguyên công doa với các dụng cụ cắt đặc biệt đòi hỏi sự can thiệp bằng tay ở đáy lỗ. Khi hoàn tất điều đó, dao lùi ra ngoài lỗ để bảo đảm an toàn. Chu kỳ này có thể được một số nhà chế tạo dụng cụ cắt sử dụng cho một số nguyên công.



Hình 24.16. *Chu kỳ G88 được dùng khi cần thao tác bằng tay*

G89 – Chu kỳ doa

G98 (G99) G89 X.. Y.. R.. Z.. P.. F..

Bước	Chu kỳ G89
1	Chuyển động nhanh đến vị trí XY
2	Chuyển động nhanh đến *mức R*
3	Chuyển động cắt gọt đến *chiều sâu Z*
4	Dwell tại chiều sâu này – tính theo mili giây (P..)
5	Chuyển động cắt gọt đến *mức R*

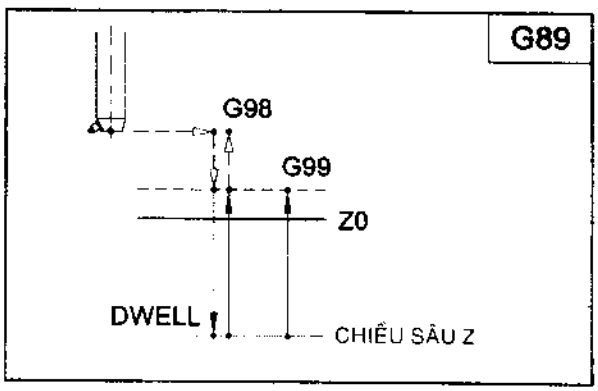
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

6	Lùi dao nhanh đến *mức ban đầu* (với G98) *hoặc* giữ nguyên ở *mức R* (với G99)

SỬ DỤNG CHU KỲ G89 – Hình 24.17

Dùng cho các nguyên công doa, khi sự cắt gọt đòi hỏi các chiều vào và ra khỏi lỗ gia công, với dwell chuyên biệt ở đáy lỗ. Dwell là giá trị duy nhất phân biệt giữa chu kỳ G89 và chu kỳ G85.



Hình 24.17. *Chu kỳ G89 – dùng cho doa và chuốt*

G76 – Chu kỳ doa chính xác

Đây là chu kỳ rất hữu dụng cho các lỗ chất lượng cao. Có hai định dạng lập trình khả dụng cho chu kỳ doa chính xác G76 – định dạng thứ nhất (sử dụng Q) thông dụng hơn định dạng thứ hai (sử dụng I và J)

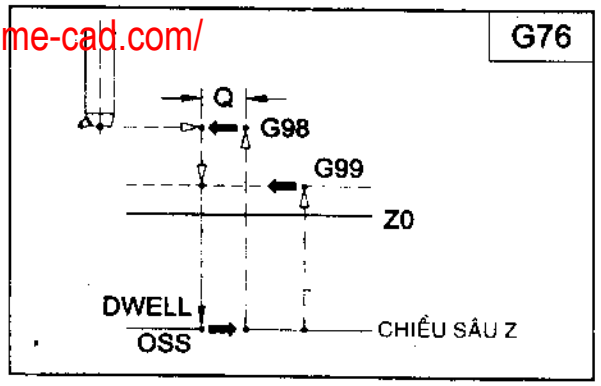
G98 (G99) G76 X.. Y.. R.. Z.. P.. Q.. F..
G98 (G99) G76 X.. Y.. R.. Z.. P.. I.. J.. F..

Bước	Chu kỳ 76
1	Chuyển động nhanh đến vị trí XY
2	Chuyển động nhanh đến *mức R*
3	Chuyển động cắt gọt đến *chiều sâu Z*
4	Dwell tại chiều sâu này, tính theo mili giây (P...) *(nếu được dùng)*
5	Dừng quay trục chính
6	Định hướng trục chính
7	Dịch chuyển *ra* theo giá trị Q *hoặc* dịch chuyển theo giá trị và chiều của I và J
8	Lùi nhanh đến *mức ban đầu* (với G98) *hoặc* giữ nguyên *mức R* (với G99)
9	Dịch chuyển *vào* theo giá trị Q *hoặc* dịch chuyển ngược chiều của I và J
10	Trục chính quay

SỬ DỤNG CHU KỲ G76 – Hình 24.18

Các nguyên công doa – thường để làm bóng lỗ khi chất lượng lỗ hoàn tất là rất quan trọng. Chất lượng này có thể xác định bằng độ chính xác kích thước lỗ, độ bóng bề mặt cao, hoặc cả hai.

Chu kỳ G76 còn được dùng để gia công các lỗ trục và song song với các trục của chúng.



Hình 24.18. *Chu kỳ G76 – doa lỗ chất lượng cao*

XÓA CHU KỲ CỐ ĐỊNH

Chu kỳ cố định bất kỳ đang hoạt động có thể bị xóa với lệnh G80. Chế độ điều khiển được chuyển một cách tự động sang chế độ chuyển động nhanh G00:

```
N34 G80
N35 Y5.0 Y-5.75
```

Block N35 không có lệnh chuyển động nhanh, nhưng *hàm ý* điều đó. Đây là thực hành lập trình bình thường. Nhưng vẫn có thể chuyên biệt G00 tùy theo ý thích cá nhân, dù không cần thiết:

```
N34 G80
N35 G00 Y5.0 Y-5.75
```

Cả hai ví dụ trên đều có cùng kết quả. Phiên bản thứ hai có thể là lựa chọn tốt hơn. Sự phối hợp hai ví dụ này là điều nên làm:

```
N34 G80 G00 X 5.0 Y - 5.75
```

Trong cả ba trường hợp, hầu như không có khác biệt, nhưng rất quan trọng để *hiểu* các chu kỳ. Dù G00 không có G80 vẫn có thể xóa chu kỳ, nhưng đây là điều không nên.

LẶP LẠI CHU KỲ CỐ ĐỊNH

Khi chu kỳ cố định được chọn để lập trình cho nhiều lỗ, chu kỳ này chỉ xử lý mỗi lần một lỗ trên chi tiết gia công. Đây là điều kiện bình thường, dựa trên giả thiết hầu hết các lỗ nhỏ đòi hỏi một dụng cụ cắt trong một chu kỳ. Trong chương trình CNC, không có lệnh đặc biệt *hiển nhiên* nêu rõ số lần xử lý chu kỳ cố định. Điều này là đúng, lệnh đó là không rõ ràng, nhưng vẫn có. Giả thiết là chu kỳ cố định phải thực thi *một lần*, nghĩa là hoàn toàn không lặp lại.

Thông thường, hệ thống điều khiển sẽ thực thi chu kỳ cố định chỉ một lần ở vị trí đã cho, không cần lập trình số lần thực thi, do mặc định hệ thống chỉ một lần. Để lặp lại chu kỳ

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

vài lần, bạn hãy lập trình lệnh đặc biệt *"báo cho"* hệ thống CNC số *lần* bạn muốn thực thi chu kỳ đó.

Địa chỉ L hoặc K

Lệnh chuyên biệt số lần lặp lại (đôi khi được gọi là *vòng lặp*) được lập trình với địa chỉ L hoặc K trên một số bộ điều khiển. Địa chỉ L hoặc K để lặp lại chu kỳ cố định được giả thiết có giá trị là một, tương đương với L1 hoặc K1. Địa chỉ L1 hoặc K1 không cần chuyên biệt trong chương trình.

Ví dụ, gọi chu kỳ cố định trong chuỗi thứ tự khoan dưới đây:

```
N33 G90 G99 ...
N34 G81 X17.0 Y20.0 R0.15 Z-2.4 F12.0
N35 X22.0
N36 X27.0
N37 X32.0
n38 G80 ...
```

tương đương với

```
N33 G90 G99 ...
N34 G81 X17.0 Y20.0 R0.15 Z-2.4 F12.0 L1
(K1)
N35 X22.0 L1 (K1)
N36 X27.0 L1 (K1)
N37 X32.0 L1 (K1)
N38 G80 ...
```

Cả hai ví dụ đều cung cấp cho hệ điều khiển các lệnh để khoan 4 lỗ theo hàng thẳng – một ở vị trí X17.0 Y 20.0, các lỗ khác lần lượt ở các vị trí X22.0 Y20.0, X27.0 Y20.0 và X32.0 Y20.0 tất cả đều sâu 2.4 inch.

Nếu tăng giá trị L hoặc K trong ví dụ thứ hai (hoặc đưa thêm vào ví dụ thứ nhất), ví dụ từ L1 lên L5 (hoặc K1 lên K5). Chu kỳ cố định sẽ lặp lại 5 lần tại *từng* vị trí lỗ. Trong thực tế không cần loại gia công này. Bằng cách thay đổi định dạng này, sự lặp lại chu kỳ có thể được sử dụng để chương trình mạnh hơn và hiệu quả hơn:

```
N33 G90 G99 ...
N34 G81 X17.0 Y20.0 R0.1 Z-2.4 F12.0
N35 G91 X5.0 L3 (K3)
N36 G90 G80 G00 ...
```

Với thay đổi đó, ưu thế của tính năng *"ấn"* trong ví dụ thứ nhất được tăng cường – *số gia bằng nhau* giữa các lỗ chính xác là 5.0 inch. Bằng cách dùng chế độ số gia, trên cơ sở tạm thời trong block N35 và dùng năng lực lặp lại của K hoặc L, chương trình CNC có thể được rút ngắn một cách rõ rệt. Phương pháp lập trình này là rất hiệu quả khi cần gia công sơ đồ nhiều lỗ trong một chương trình. Sự tăng cường tiếp theo là kết hợp L hoặc K với macro hoặc chương trình con.

L0 hoặc K0 trong chu kỳ

Trong phần trước đã đề cập, mặc định đối với sự lặp lại chu kỳ là L1 hoặc K1, không cần chuyên biệt trong chương trình. Giá trị L hoặc K bất kỳ khác L1 hoặc K1 phải được chuyên biệt trong khoảng cho phép của địa chỉ L hoặc K. Khoảng này là L0 – L9999 hoặc K0 – K9999. Giá trị L/K thấp nhất là L0 hoặc L0, không phải là L1 hoặc K1. Tại sao chúng ta muốn lập trình chu kỳ cố định, sau đó yêu cầu *"không thực hiện chu kỳ đó"*? Địa chỉ L0 hoặc K0 có nghĩa là *"không thực thi chu kỳ"*. Lợi ích của L0/K0 được nêu rõ trong các ví dụ ở phần chương trình con (Chương 38).

Bằng cách lập trình L0 hoặc K0 trong chu kỳ cố định, ý nghĩa thực *không phải là "không thực thi chu kỳ đó"* mà là *"không thực thi chu kỳ này, nhưng nhớ các tham số chu kỳ để sử dụng trong tương lai"*.

Đối với hầu hết các nguyên công, chu kỳ cố định tương đối đơn giản, nhưng có một số tính năng phức tạp có thể được áp dụng một cách hiệu quả ngay cả khi gia công một lỗ.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

25 GIA CÔNG CÁC LỖ

Đa số các chương trình trên trung tâm gia công CNC là dùng cho gia công lỗ. Từ khoan điểm đơn giản đến chuốt, tarô ren và doa ngược phức tạp, lĩnh vực gia công lỗ là rất rộng. Trong chương này, bạn sẽ nghiên cứu nhiều phương pháp lập trình để gia công các lỗ và các kỹ thuật tương ứng. Các nguyên công khoan và doa, kể cả chuốt, tarô ren và doa một lưỡi cắt, sẽ được trình bày chi tiết.

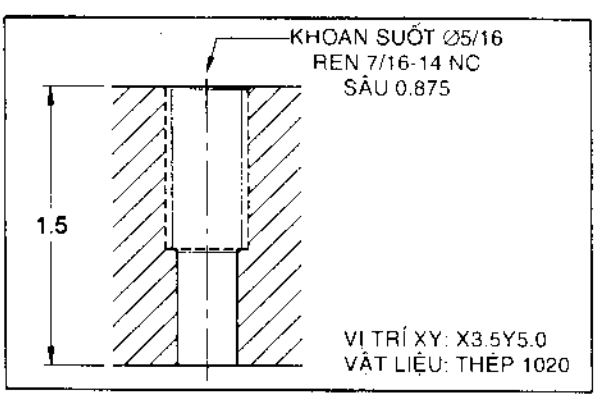
Kiểu gia công lỗ phổ biến nhất trên trung tâm gia công CNC là khoan, tarô ren, chuốt, và doa một lưỡi cắt. Quy trình gia công có thể là khoan tâm hoặc khoan điểm chuỗi các lỗ, sau đó khoan, ta rô ren hoặc doa. Sự gia công một lỗ cũng vẫn có thể có lợi khi sử dụng các chu kỳ, G01 đến G89, G73 và G76 (Chương 24).

ĐÁNH GIÁ MỘT LỖ

Trước khi gia công, dù chỉ một lỗ trên máy CNC, cần lập trình mọi quỹ đạo dao. Trước đó, cần xác định các dụng cụ cắt, chọn tốc độ và lượng ăn dao, gá lắp và nhiều vấn đề khác cần giải quyết. Bất kể phương pháp được chọn, bạn hãy luôn luôn bắt đầu với việc *đánh giá lỗ cho trước*.

Bước thứ nhất, liên quan đến dữ liệu bản vẽ, gồm vật liệu gia công, vị trí và các giá trị kích thước lỗ. Lỗ ít khi được vẽ một cách chi tiết, do đó nhà lập trình có thể thiếu các *kích thước* cụ thể. Hình 25.1 minh họa lỗ hơi phức tạp có thể gia công trên máy CNC.

Mọi thông tin liên quan đều trên bản vẽ nhưng có thể phải xác định thêm một số chi tiết và các yêu cầu khác. Vị trí lỗ X3.5Y5.0 được ghi trong bản vẽ, cùng với vật liệu là thép C thấp (0.2%C). Chương trình trục Z sẽ gán cho mặt trên của chi tiết. Các nguyên công khoan, tarô ren là đương nhiên, nhưng vẫn chưa đủ để lập trình.



Hình 25.1. *Đánh giá lỗ – ví dụ lập trình O2501*

Cần bao nhiêu dao cắt? Khoan tâm có duy trì vị trí chính xác của lỗ? Có nên chọn khoan điểm? Vạt góc cho lỗ khoan để ta rô như thế nào? Dung sai và động bóng bề mặt lỗ?...

Lựa chọn dao và ứng dụng

Chỉ dựa và thông tin trên bản vẽ, dường như chỉ cần hai dao để lập trình gia công lỗ này. Trong thực tế, cần *diễn dịch* phần thông tin ẩn – do bản vẽ không nêu *phương pháp* gia công lỗ, chỉ cung cấp các yêu cầu liên quan đến công dụng và chức năng của lỗ. Thợ CNC sẽ chọn *bốn* dụng cụ cắt để có kết quả gia công tốt nhất. Nếu chọn 4 dụng cụ cắt, thứ nhất sẽ là *mũi khoan điểm* 90^0, tiếp theo là *khoan lỗ ren*, *mũi khoan lỗ suốt*, và cuối cùng là *tarô ren*. Mũi khoan điểm tiêu chuẩn có thể thay cho khoan điểm nhưng phải có thêm dụng cụ cắt để vạt góc đường kính lỗ ở phần đỉnh.

Đối với ví dụ này, cần dùng bốn dụng cụ cắt dưới đây:

- ❑ Dao 1 – T01 – Mũi khoan điểm 90^0 (+ vạt góc)
- ❑ Dao 2 – T02 – Mũi khoan lỗ ren chữ U (∅ .368)
- ❑ Dao 3 – T03 – Mũi khoan điểm ∅5/16 (khoan lỗ suốt)
- ❑ Dao 4 – T04 – Tarô ren 7/16 – 14 UNC

Dao 1 – Mũi khoan điểm 90^0

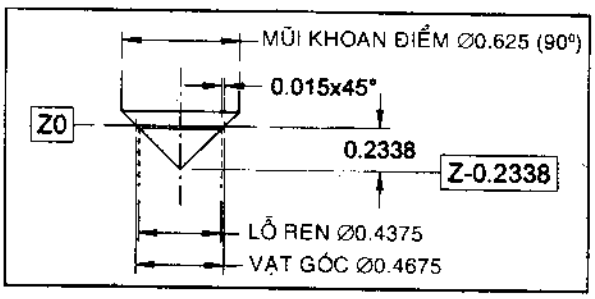
Dao thứ nhất sẽ là mũi khoan điểm 90^0, với hai mục đích. Thứ nhất là khoan tâm và khơi đầu lỗ tại vị trí XY chính xác cao. Mũi khoan tâm hoặc khoan điểm có độ cứng vững cao hơn so với mũi khoan xoắn, do đó sẽ *khởi đầu* lỗ khoan, sao cho mũi khoan tiếp theo không bị lệch khỏi quỹ đạo (bảo đảm các yêu cầu về độ đồng tâm và vị trí lỗ cơ bản). Thứ hai là vạt góc ở phần đỉnh lỗ, do đó đường kính mũi khoan điểm phải *lớn hơn* đường kính vạt góc được yêu cầu. Trong trường hợp này sẽ sử dụng mũi khoan điểm ∅5/8, thích hợp để vạt góc lỗ ∅7/16.

Bản vẽ không nêu rõ phần vạt góc, nhưng thợ cơ khí thường thực hiện vạt góc nhỏ, đôi khi được gọi là góc vát hẹp, trừ khi có yêu cầu khác. Vạt góc thích hợp sẽ là .015 × 45^0.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Sau khi chọn mũi khoan điểm, cần tính chiều sâu cắt, *tính toán* thấy cho phòng đoán. Để đạt được sự vạt góc .015 × 45^0 cho cỡ tarô Ø7/16 (Ø.4375), đường kính tarô được *làm rộng thêm* .015 ở mỗi phía (.03/đường kính) đến đường kính vạt góc .4675. Hình 25.2 minh họa quan hệ giữa lỗ và dụng cụ cắt (đường kính và chiều sâu).



Hình 25.2. *Chi tiết khoan điểm – T01 trong chương trình O2501*

Chú ý, đối với mũi khoan có góc đỉnh là 90^0, chiều sâu cắt bằng đúng một nửa đường kính vạt góc (Ø × 0.5). Chiều dài đỉnh mũi khoan sẽ được trình bày trong phần kế tiếp.

.4675 / 2

hoặc .4675 x.5 =.23375

hoặc Z-0.2338

Dao 2 – Mũi khoan lỗ ren

Theo logic, dụng cụ thứ hai là mũi khoan. Trong ví dụ này, cần *hai* mũi khoan, một cho lỗ suốt (Ø5/16 = Ø.3125), và một cho phần tarô ren (mũi khoan chữ U = Ø.368). Câu hỏi là mũi khoan nào trước? Vấn đề gì sẽ xảy ra?

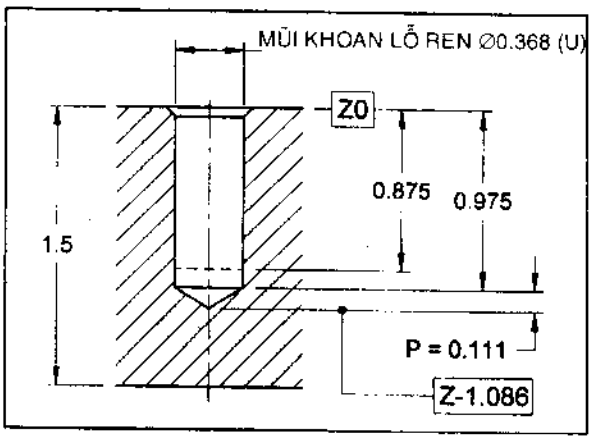
Chắc chắn *sẽ có vấn đề* mũi khoan nào được lập trình trước. Cốt lõi ở đây là sự khác biệt giữa hai đường kính mũi khoan. Hiệu số là rất nhỏ, chỉ là .0555. Từ quan điểm gia công, nên sử dụng *mũi khoan lớn trước*. Mũi khoan lỗ ren lớn hơn mũi khoan lỗ suốt, do đó T02 sẽ là mũi khoan lỗ ren. Nếu mũi khoan nhỏ (khoan lỗ suốt) được lập trình trước, mũi khoan lớn gia công sau có thể làm cho lỗ không chính xác do lượng dư gia công rất nhỏ.

Câu hỏi tiếp theo là kích cỡ mũi khoan thứ nhất, đây là *mũi khoan lỗ ren*. Mũi khoan này sẽ tạo ra lỗ tròn có kích thước thích hợp (*đường kính, chiều sâu*) dùng cho mũi tarô trong thứ tự gia công. Do cần tarô ren, sẽ có sự khác biệt lớn về *công dụng* của mũi tarô được sử dụng. Không phải mọi lỗ ren đều được thực hiện bằng cùng một phương pháp. Một số yêu cầu lắp lỏng, số khác đòi hỏi lắp chặt. Độ dôi đối với mũi tarô được xác định theo kích cỡ mũi khoan lỗ ren. Hầu hết các ứng dụng tarô ren đều trong khoảng 72–77% chiều sâu ren toàn phần. Trong trường hợp này, T02 (mũi khoan chữ U) sẽ cần khoảng 77% chiều sâu ren. Số phần trăm chiều sâu ren được ghi kèm theo mũi tarô. Ví dụ, các lựa chọn cho mũi tarô 7/16 – 14 gồm:

Ø mũi khoan	Giá trị thập phân	% chiều sâu ren toàn phần
T	.3580	86%
23/64	.3594	84%
U	.3680	75%
3/8	.3750	67%
V	.3770	65%

Nói chung, đối với phôi mỏng, cần dùng chiều sâu ren 75 – 80%, phôi rất mỏng có thể đến 100%. Trong hầu hết các trường hợp, ren có 53% chiều sâu sẽ *làm gãy* bulông khi vặn vào lỗ ren. Ren 100% chỉ bền hơn 5% so với ren 75% nhưng đòi hỏi *công suất* máy để tarô ren cao gấp 3 lần.

Chiều sâu Z lập trình của mũi khoan lỗ ren phải đủ sâu để bảo đảm chiều sâu cắt ren toàn phần là .875. Điều đó có nghĩa là đường kính *toàn phần* của mũi khoan này phải xuống sâu hơn, chẳng hạn đến chiều sâu .975. Điều này cho phép phần vát ở đỉnh tarô *dưới* chiều sâu cắt ren toàn phần .875 trên bản vẽ. Hình 25.3 minh họa các giá trị mũi khoan lỗ ren.



Hình 25.3. *Chi tiết nguyên công khoan lỗ ren - T02 trong chương trình O2501.*

Chiều sâu Z lập trình của mũi khoan lỗ ren còn xét đến một yếu tố – *chiều dài đỉnh mũi khoan*. Chiều dài đỉnh dụng cụ cắt hoặc mũi khoan đôi khi được viết tắt là TPL hoặc P. Bảng 25.2 liệt kê các hệ số toán học dùng để tính chiều dài đỉnh mũi khoan. Hệ số thông dụng nhất là đường kính mũi khoan nhân với 0.300, với góc đỉnh mũi khoan 118^0.

P = .368 x .300 = .1104 = .1110

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

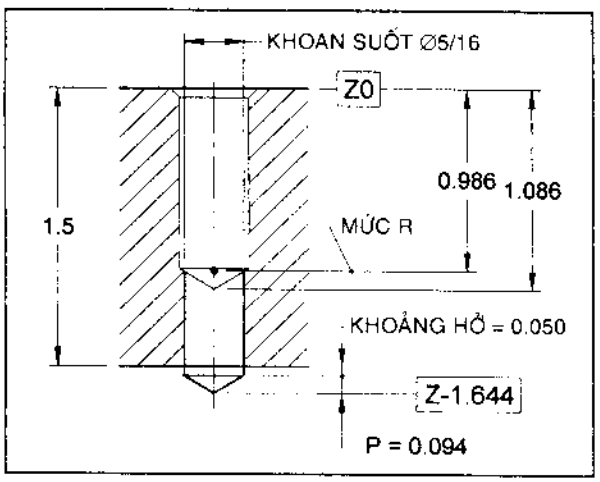
Cộng hai kết quả tính toán nêu trên (.975 +.111) sẽ nhận được chiều sâu Z lập trình là Z-1.080

Dao 3 – Mũi khoan suốt

Dao kế tiếp là mũi khoan lỗ suốt. Trong ví dụ này, T03 là mũi khoan tiêu chuẩn ∅5/16.

Đối với chiều sâu cắt của mũi khoan lỗ suốt, cần vài tính toán đơn giản. Để thực hiện các tính toán đó, bạn cần biết chiều sâu lỗ, trên bản vẽ giá trị này là 1.5 inch, sau đó cộng chiều dài đỉnh mũi khoan tính toán với chiều lỗ, thường có thêm khoảng hở nhỏ.

Các phép tính đối với nguyên công khoan lỗ suốt được nêu trên Hình 25.4.



Hình 25.4. *Chi tiết nguyên công khoan lỗ suốt – T03 trong chương trình O2601*

Trước hết, bạn tính chiều dài đỉnh mũi khoan P khi biết hai giá trị, *đường kính* mũi khoan và *góc đỉnh* mũi khoan. Đối với mũi khoan tiêu chuẩn ∅5/16 (∅ .3125) góc đỉnh là 118^0, do đó sẽ dùng hệ số 0.300. Chiều dài đỉnh mũi khoan P sẽ là:

P = .3125 x .300 = .09375 = .0938 = .094

Đối với lỗ suốt trong ví dụ này, chiều sâu trên bản vẽ 1.5 inch *cộng* với chiều sâu tính toán .094 *đường như* là đủ để khoan lỗ sử dụng mũi khoan đã chọn.

Trong hầu hết các ứng dụng khoan lỗ suốt, giá trị này sẽ *không đủ*, cần có thêm khoảng hở để mũi khoan xuyên qua, chẳng hạn .050 in. Giá trị lập trình đối với chiều sâu khoan toàn phần (giá trị Z tuyệt đối trong chương trình) là *tổng* chiều sâu lỗ danh định, cộng chiều dài góc đỉnh khoan, cộng khoảng hở được chọn. Trong ví dụ chương trình này, giá trị chiều sâu khoan suốt sẽ là:

Chiều sâu = 1.5 +.094 +.05
= 1.644 *hoặc* Z-1.644 *trong chương trình*

Bạn còn nhớ, dao T02 đã khoan lỗ với đường kính và chiều sâu xác định. Điều đó có nghĩa là dao T03 có ∅.3125 được đưa và lỗ ∅.368 hiện hữu. Sự khoan sẽ bắt đầu từ bên trong lỗ hiện hữu thay vì từ phía trên chi tiết. Trong chương trình, giá trị R được sử dụng và được chọn là R-0.986 với khoảng hở .100 phía trên đáy lỗ hiện hữu.

Dao 4 – Mũi tarô

Chỉ còn lại một dao để hoàn tất ví dụ này, dao đó được dùng để tarô ren 7/16 – 14. Cỡ ren được ghi trên bản vẽ có đường kính danh định 7/16 với 14 ren/inch (1/14 = .0714 = bước ren). Thời điểm bất kỳ mũi tarô được dùng trong chương trình, bạn hãy chú ý chiều sâu lập trình theo trục Z, đặc biệt là lỗ cụt hoặc bán cụt. Ví dụ này có lỗ bán cụt, do lỗ suốt nhỏ hơn lỗ tarô ren. Nếu không có lỗ suốt, ở đây sẽ là lỗ cụt, và nếu lỗ suốt có cùng đường kính với mũi khoan lỗ ren, lỗ đó được gọi là lỗ suốt 100%.

Tính chiều sâu Z cho lỗ suốt là đơn giản nhất, sẽ hơi khó hơn với lỗ bán suốt, còn với lỗ cụt cần lập trình chiều sâu Z rất cẩn thận.

Bản vẽ lỗ yêu cầu chiều sâu tarô ren là .875 inch. Đây là chiều sâu *toàn phần* có ren. Chiều sâu toàn phần này là khoảng cách thực để vít hoặc bulông được vặn vào.Chiều sâu lập trình, chiều sâu *kéo dài*, sẽ phải *lớn hơn* chiều sâu lý thuyết, nhằm đạt được mục đích đó. Để tính chiều sâu kéo dài, cần đánh giá phần vạt góc ở đỉnh mũi tarô (kiểu và chiều sâu), sẽ được trình bày trong phần tarô ren.

Chiều sâu Z hợp lý là Z-0.95 (tăng thêm khoảng một bước ren) và có thể tối ưu hóa sau khi thực sự gia công. Đây không phải là tính toán thực sự, chỉ là sự *"ước tính thông minh"* dựa theo kinh nghiệm. Phần ứng dụng gia công lỗ đến đây là hoàn tất, bạn đã có đủ dữ liệu viết chương trình. Một số quy trình trong ví dụ này sẽ được giải thích chi tiết hơn.

Dữ liệu chương trình

Trong ví dụ này chỉ gia công một lỗ. Nếu cần nhiều lỗ, bạn có thể bổ sung bằng cách chỉnh sửa chương trình dưới đây. Đối với một lỗ trong ví dụ, chương trình gộp tất cả các xem xét về bốn dụng cụ cắt nêu trên. Trục chính phải rỗng khi bắt đầu chương trình:

```
O2501 (MỘT LỖ)
(T01 - 5/8 DIA - 90 DEGREE SPOT DRILL)
N1 G20
N2 G17 G40 G80 T01
N3 M06
N4 G90 G54 G00 X3.5 Y5.0 S900 M03 T02
N5 G43 Z0.1 H01 M08
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N6 G99 G82 R0.1 Z-0.2338 P300 F4.0
N7 G80 Z1.0 M09
N8 G28 Z1.0 M05
N9 M01

(T02 - LETTER U DRILL - 0.368 DIA DRILL)
N10 T02
N11 M06
N12 G90 G54 G00 X3.5 Y5.0 S1100 M03 T03
N13 G43 Z0.1 H02 M08
N14 G99 G83 R0.1 Z-1.086 Q0.5 F8.0
N15 G80 Z1.0 M09
N16 G28 Z1.0 M05
N17 M01

(T03 - 5/16 DRILL THROUGH - 0.3125 DIA)
N18 T03
N19 M06
N20 G90 G54 G00 X3.5 Y5.0 S1150 M03
N21 G43 Z0.1 H03 M08
N22 G98 G81 R-0.986 Z-1.644 F8.0
N23 G80 Z1.0 M09
N24 G28 Z1.0 M05
N25 M01

(T04 - 7/16-14 TAP)
N26 T04
N27 M06
N28 G90 G54 G00 X3.5 Y5.0 S750 M03 T01
N29 G43 Z0.4 H04 M08
N30 G99 G84 R0.4 Z-0.9 F53.57    (F = S x LEAD)
N31 G80 G00 Z1.0 M09
N32 G28 Z1.0 M05
N33 G00 X-1.0 Y10.0    (PART CHANGE POSITION)
N34 M30
%
```

Chương trình này cho thấy dù chỉ một lỗ cũng đòi hỏi sự suy nghĩ thấu đáo, lựa chọn kỹ thuật lập trình và kiến thức về công nghệ.

CÁC NGUYÊN CÔNG KHOAN

Ví dụ O2501 minh họa loại lập trình và các điều kiện gia công cần thiết đối với lỗ điển hình. Bạn hãy xem xét chi tiết các nguyên công khoan và dụng cụ cắt liên quan.

Các kiểu nguyên công khoan

Nguyên công khoan được xác định theo *kiểu lỗ khoan* hoặc *kiểu dụng cụ cắt*.

Bảng 25.1: *Phân loại nguyên công khoan*

Kiểu dụng cụ cắt	Kiểu lỗ gia công
Mũi khoan tâm	Lỗ suốt
Mũi khoan điểm	Lỗ vát miệng (lá miệng)
Mũi khoan xoắn	Lỗ bán cụt
Mũi khoan khoét	Lỗ cụt
Mũi khoan chắp	Lỗ khoan sẵn
Mũi khoan đặc biệt	...

Các kiểu mũi khoan

Mũi khoan được phân loại theo *thiết kế* và theo *kích cỡ*. Thiết kế phổ biến nhất là lâu đời nhất là mũi khoan xoắn, thường được chế tạo bằng thép gió (tháp dụng cụ cắt gọi tốc độ cao), hợp kim Fe-Co, hợp kim cứng (carbides),... Các thiết kế khác bao gồm mũi khoét, khoan tâm, khoan điểm (lá miệng), mũi khoan chắp mảnh carbides. Sự khác biệt về kích cỡ không chỉ giữa các mũi khoan hệ Anh và hệ mét, mà còn được phân loại kỹ hơn với các đơn vị Anh. Do kích thước theo hệ Anh dựa trên đơn vị inch (tương đối lớn), cần có sự phân biệt chi tiết hơn. Kích thước inch của các mũi khoan tiêu chuẩn theo hệ Anh được chia làm ba nhóm:

- KÍCH CỠ PHÂN SỐ

  1/64 là nhỏ nhất, số gia đường kính (tăng dần) theo 1/64

- KÍCH CỠ THEO SỐ

  Số cỡ mũi khoan 80 đến số cỡ mũi khoan 1

- KÍCH CỠ THEO CHỮ

  Chữ kích cỡ mũi khoan từ A đến Z.

Kích cỡ hệ mét không cần các phân loại đặc biệt. Đối với kích cỡ Anh, bạn có thể xem nhiều tài liệu để biết mũi khoan tiêu chuẩn và đương lượng thập phân của chúng.

Các xem xét lập trình

Mũi khoan tiêu chuẩn, bất kể kích cỡ, có hai tính năng quan trọng, *đường kính* và *góc đỉnh*. Đường kính được chọn theo các yêu cầu của bản vẽ, góc đỉnh liên quan đến độ cứng của vật liệu gia công. Chúng quan hệ chặt chẽ với nhau, đường kính xác định kích cỡ lỗ khoan, góc đỉnh xác định chiều sâu. Điều thứ hai cần xét là số rãnh thoát phoi, mũi khoan tiêu chuẩn có hai rãnh.

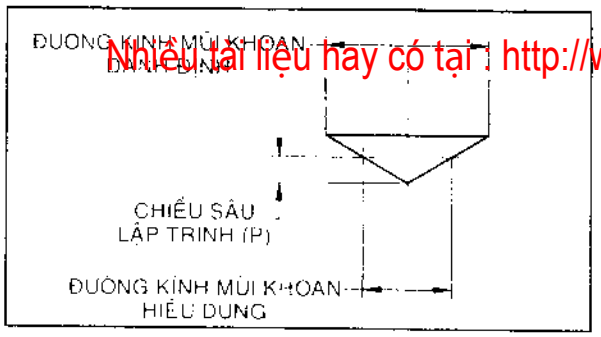
Đường kính mũi khoan danh định

Khảo sát cơ bản là đường kính. Nói chung, đường kính và mũi khoan được chọn dựa vào dữ liệu trên bản vẽ. Nếu bản vẽ cho biết lỗ chỉ khoan và không cần gia công bổ sung, chỉ cần dùng mũi khoan tiêu chuẩn. Đường kính mũi khoan tương đương với kích cỡ lỗ trên bản vẽ. Cỡ mũi khoan này được gọi là đường kính *danh định* (Hình 25.5).

Hầu hết các ứng dụng đều gồm các lỗ đòi hỏi thêm các đặc tính kỹ thuật bổ sung cho đường kính, chẳng hạn dung sai, độ bóng bề mặt, lá miệng, độ đồng tâm ... Trong các trường hợp đó, không thể dùng một mũi khoan tiêu chuẩn để đáp ứng tất cả các yêu cầu. Mũi khoan *danh định*, không bảo đảm lỗ chất lượng cao, do các điều kiện gia công. Sự lựa chọn kỹ thuật *nhiều dụng cụ cắt* để gia công lỗ này là hoàn toàn hợp lý. Ứng dụng chung trong những trường hợp đó là dùng mũi khoan hơi nhỏ hơn đường kính lỗ hoàn tất, sau đó dùng

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

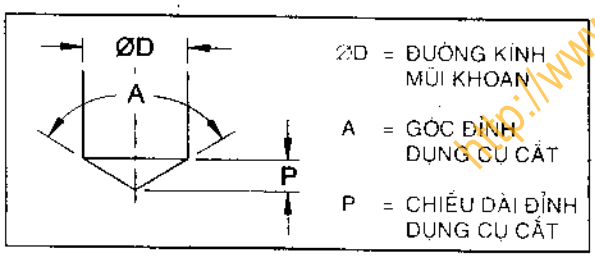


Hình 25.5. Đường kính khoan danh định và đường kính hiệu dụng (mũi khoan xoắn).

thêm các dụng cụ cắt khác, có khả năng hoàn thiện lỗ theo các yêu cầu bản vẽ. Các dụng này gồm thanh doa, dao chuốt, dụng cụ lã miệng, phay, v...v... Sử dụng các dụng cụ đó sẽ tăng khối lượng công việc nhưng cho phép bảo đảm chất lượng gia công.

Chiều dài đỉnh mũi khoan

Vấn đề quan trọng thứ hai là *chiều dài* của đỉnh mũi khoan. Chiều dài này rất quan trọng để thiết lập chiều sâu dao đến đường kính toàn phần. Với ngoại lệ là mũi khoan đáy phẳng, mọi mũi khoan xoắn đều có đỉnh côn, cần biết cả góc đỉnh và chiều dài trong lập trình. Các góc này thường theo tiêu chuẩn và cần tính chiều dài thay vì phỏng đoán, do tầm quan trọng của chúng đối với chiều sâu lỗ chính xác (Hình 25.6).



Hình 25.6. Dữ liệu chiều dài đỉnh của mũi khoan xoắn tiêu chuẩn.

Trên các mũi khoan gắn mảnh hợp kim cứng, chiều dài này hơi khác, do cấu trúc mũi khoan. Loại mũi khoan này không có đáy phẳng do đó cần xét chiều dài đỉnh khi lập trình.

Có thể dễ dàng xác định chiều dài đỉnh mũi khoan nếu biết đường kính (danh định hoặc hiệu dụng) và góc đỉnh. Từ công thức dưới đây và bảng các hệ số (Bảng 25.2), có thể tính chiều dài đỉnh yêu cầu đối với mũi khoan tiêu chuẩn:

$$P = \frac{D}{2 \tan\left(\frac{90}{A/2}\right)}$$

Trong đó: P = Chiều dài đỉnh
A = Góc đỉnh
D = Đường kính mũi khoan.

Công thức này được đơn giản hóa, sử dụng hệ số phân (không đổi đối với từng góc đỉnh)

$$P = D \times K$$

Trong đó: P = Chiều dài đỉnh
D = Đường kính mũi khoan.
K = Hệ số nhân

Bảng 25.2. Hệ số xác định chiều dài đỉnh mũi khoan

Góc đỉnh mũi khoan (Độ)	Hệ số chính xác	Hệ số thực tiễn (K)
60	.866025404	.866
82	575184204	.575
90	.500000000	500
118	300430310	.300
120	.288675135	.289
125	.260283525	.260
130	.233153829	.230
135	.207106781	.207
140	.181985117	.180
145	.157649394	158
150	.133974596	.134

Hệ số trong công thức được làm tròn với giá trị đủ cho mọi ứng dụng lập trình. Giá trị K đối với góc đỉnh mũi khoan 118^0 là 0.300, giá trị thực là 0.300430310. Giá trị làm tròn có ưu điểm là dễ nhớ và dễ sử dụng. Đối với hầu hết các ứng dụng, có lẽ chỉ cần 3 hệ số, góc 90^0 (khoan điểm, khoan vật liệu mềm), 118^0 (vật liệu tiêu chuẩn) và 135^0 (vật liệu cứng). Chúng rất dễ nhớ:

- 0.500 góc đỉnh mũi khoan 90^0
- 0.300 góc đỉnh mũi khoan $118^0 - 120^0$
- 0.200 góc đỉnh mũi khoan 135^0

Khoan tâm

Khoan tâm là nguyên công tạo lỗ nhỏ đồng tâm cho chuôi nhọn ụ động hoặc dẫn hướng để khoan lỗ lớn. Không nên dùng mũi khoan tâm để lã miệng (vạt góc miệng) do góc đỉnh chỉ là 60^0.

Không được khoan tâm lỗ sẽ khoan với mũi khoan gắn mảnh hợp kim!

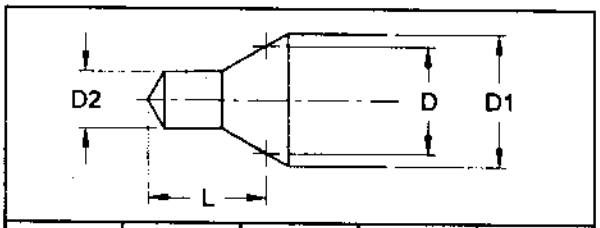
Dụng cụ phổ biến nhất để khoan tâm là mũi khoan tâm tiêu chuẩn (thường được gọi là *mũi khoan kết hợp và khoét miệng*), tạo ra góc 60^0. Tiêu chuẩn công nghiệp Bắc Mỹ sử dụng hệ thống đánh số từ #00 đến #8 và #11 đến #18 cho các mũi khoan tâm. Trong hệ mét, được xác định theo đường kính dẫn hướng, ví dụ mũi

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

khoan tâm 4 mm sẽ có đường kính dẫn hướng 4 mm. Trong đa số trường hợp, sai càng lớn đường kính mũi khoan càng lớn. Đối với một số nguyên công khoan sơ bộ, chẳng hạn lỗ miệng, dụng cụ cắt với góc đỉnh 90^0, được gọi là *mũi khoan điểm*, là loại được dùng phổ biến.

Nhiều nhà lập trình chỉ ước tính chiều sâu khoan tâm, thay vì tính toán.Có lẽ tính toán là không cần thiết cho sự gia công tạm thời. Bạn có thể dùng dữ liệu trên Hình 25.7 để xác định chiều sâu khoan tâm với các mũi khoan có đường kính D khác nhau.



Số	D1	D2	D	L
#1	0.125	0.047	0.100	0.106
#2	0.188	0.078	0.150	0.163
#3	0.250	0.110	0.200	0.219
#4	0.312	0.125	0.250	0.269
#5	0.438	0.188	0.350	0.382
#6	0.500	0.218	0.400	0.438
#7	0.625	0.250	0.500	0.538
#8	0.750	0.312	0.600	0.651

Hình 25.7. *Chiều sâu cắt L với mũi khoan tâm tiêu chuẩn.*

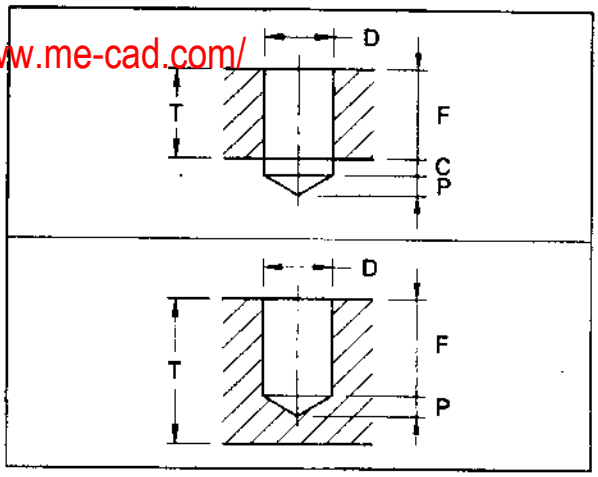
Hình 25.7 nêu rõ mọi kích thước cần thiết của các mũi khoan tâm tiêu chuẩn theo hệ Anh, trong đó quan trọng nhất là chiều sâu cắt L. Tính toán L dựa trên *sự lựa chọn tùy ý* đường kính lỗ miệng D.

Ví dụ, mũi khoan tâm #5 có giá trị chiều sâu L là 0.382, dựa trên đường kính lỗ miệng được chọn tuỳ ý D = 0.350 inch. Các giá trị này có thể chỉnh sửa theo yêu cầu gia công cụ thể, theo đơn vị hệ Anh hoặc hệ mét.

Khoan lỗ suốt

Khoan lỗ suốt qua chi tiết là nguyên công rất phổ biến. Khoan lỗ suốt yêu cầu chiều sâu Z, gồm chiều dày vật liệu, chiều dài đỉnh mũi khoan và khoảng hở *phía sau* điểm xuyên thấu của mũi khoan, được gọi là giá trị *xuyên qua.*

Hình 25.8 minh họa chiều sâu lập trình cho lỗ suốt là *tổng* chiều dày vật liệu T, tương đương với chiều sâu đường kính toàn phần F, cộng khoảng hở xuyên qua C, và chiều dài đỉnh P.



Hình 25.8. *Dữ liệu tính toán chiều sâu khoan. Lỗ suốt (trên) và lỗ cụt (dưới)*

Ví dụ, nếu chiều dày vật liệu là một inch, đường kính mũi khoan Ø5/8 (Ø.625) inch, chiều sâu lập trình, với khoảng hở 0.050 inch sẽ là:

1 + .050 + (5/8 x .300) = 1.2375

Bạn hãy chú ý các chướng ngại (bàn máy, đồ gá, ngàm kẹp,...) khi lập trình khoảng hở xuyên qua. Nói chung, khoảng trống ở mặt dưới chi tiết thường rất nhỏ.

Khoan lỗ cụt

Khác biệt cơ bản giữa khoan lỗ cụt và lỗ suốt là mũi khoan *không* xuyên qua vật liệu. Khoan lỗ cụt hầu như tương tự khoan lỗ suốt, riêng đối với lỗ sâu phải sử dụng phương pháp khoan nhiều đoạn. Ngoài ra, sự lựa chọn dạng hình học thích hợp cho mũi khoan sẽ góp phần cải thiện sự gia công, đồng thời cần làm sạch lỗ khoan.

Trên bản vẽ, chiều sâu lỗ cụt được xác định theo chiều sâu có *đường kính toàn phần*. Chiều dài đỉnh mũi khoan thường *không* được coi là một phần của chiều sâu đó, chỉ *cộng vào* chiều sâu đã chuyên biệt. Trên Hình 25.8, chiều sâu lập trình của lỗ cụt là *tổng* chiều sâu đường kính toàn phần F và chiều dài mũi khoan P.

Ví dụ, mũi khoan tiêu chuẩn Ø3/4 (Ø.750) được dùng để khoan chiều sâu lỗ đường kính toàn phần 1.25 inch, chiều sâu lập trình sẽ là:

1.25 + (.750 x .300) = 1.4750

Trong chương trình gia công, block sẽ là

```
N93 G01 Z- 1.475 F 6.0
```

hoặc, nếu dùng chu kỳ cố định

```
N93 G99 G85 X5.75 Y8.125 R0.1 Z-1.475 F6.0
```

Các lỗ hệ mét cũng được xử lý tương tự. Ví dụ, mũi khoan Ø16 mm được dùng để gia công

Nhiều tài liệu hay có tại : http://www.me-cad.com/

chiều sâu đường kính toàn phần 40 mm. Tính toán sử dụng hệ số Bảng 25.2 là toàn toàn mm với đơn vị Anh:

40 + (16 x 0.300) = 44.8

Chiều sâu (40 mm) ghi trên bản vẽ được *cộng* thêm chiều dài đỉnh mũi khoan (16 × 0.300). Block lập trình có giá trị trục Z bằng tổng chiều sâu 40 mm và chiều dài đỉnh tính toán 4.8 mm.

```
N56 G01 Z-44.8 F150.0
```

Nếu dùng chu kỳ cố định, giá trị trục Z vẫn là 44.8, nhưng có định dạng khác.

```
N56 G99 G87 x 215.0Y175.- R2.5 Z-44.8 F150.0
```

Khi gia công lỗ cụt, phoi cắt có thể kẹt trong lỗ. Điều này sẽ gây ra vấn đề, đặc biệt khi có nguyên công kế tiếp trong lỗ, chẳng hạn chuốt hoặc tarô ren. Bạn cần đưa mã dừng chương trình M00 *hoặc* M01 vào trước nguyên công đó. M00 là lựa chọn tốt hơn, nếu cần làm sạch lỗ mỗi khi thực thi chương trình. Nói chung, mã dừng chương trình tuỳ chọn M01 cũng đủ cho phép làm sạch lỗ.

Khoan lỗ đáy phẳng

Lỗ đáy phẳng là lỗ cụt với đáy 90^0 theo đường tâm khoan. Có hai phương pháp lập trình để gia công kiểu lỗ này. Bạn có thể dùng mũi khoan tiêu chuẩn để khoan lỗ sau đó dùng mũi khoan đáy phẳng có cùng đường kính để hoàn tất lỗ đến chiều sâu toàn phần. Phương pháp thứ hai là dùng *dao phay mặt đầu, không cần* khoan trước. Đây là phương pháp tốt nhất, nhưng có thể không có một số cỡ dao phay.

Lập trình lỗ đáy phẳng với dao phay mặt đầu tương đối đơn giản. Ví dụ, gia công lỗ ∅10 mm đáy phẳng, sâu 25 mm. Sử dụng dao phay mặt đầu ∅10 mm, chương trình sẽ rất ngắn nếu dao đã lắp trong trục chính.

```
O2502 (ĐÁY PHẲNG - 1)
N1 G21
N2 G17 G40 G80
N3 G90 G54 G00 X.. Y.. S850 M03
N4 G43 Z2.5 H01 M08
N5 G01 Z-25.0 F200.0
N6 G04 X0.5
N7 G00 Z2.5 M09
N8 G28 Z3.0 M05
N9 M30
%
```

Bạn còn có thể dùng chu kỳ cố định với vài chỉnh sửa để gia công lỗ đáy phẳng.

Ví dụ kế tiếp là chương trình với hai dụng cụ cắt, mũi khoan tiêu chuẩn ∅1/2 inch và mũi khoan đáy phẳng ∅1/2 inch. Chiều sâu hoàn tất với đáy phẳng là Z-0.95:

```
O2503 (ĐÁY PHẲNG - 2)
T01 (½ INCH STANDARD DRILL)
N1 G20
N2 G17 G40 G80 T01
N3 M06
N4 G90 G54 G00 X.. Y.. S700 M03 T02
N5 G43 Z0.1 H01 M08
N6 G01 Z-0.94 F9.0
N7 G00 Z0.1 M09
N8 G28 Z0.1
N9 M01

(T02 - ½INCH FLAT BOTTOM DRILL / END MILL)
N10 T02
N11 M06
N12 G90 G54 000 X.. Y.. S700 M03 T01
N13 G43 Z0.1 H02 M08
N14 G01 Z-0.74 F15.0
N15 Z-0.95 F7.0
N16 G04 X0.5
N17 G00 Z0.1 M09
N18 G28 Z0.1 M05
N19 M30
%
```

Bạn hãy chú ý ba block trong chương trình O2503. Thứ nhất là block N6, biểu thị chiều sâu mũi khoan tiêu chuẩn. Mũi này dừng lại trước chiều sâu toàn phần khoảng 0.010 inch. Giá trị Z-0.94 được lập trình thay vì Z-0.95. Lý do *không* khoan đến chiều sâu toàn phần với mũi khoan tiêu chuẩn là để tránh gây ra vết đỉnh dao ở tâm lỗ.

Hai block tiếp theo trong phần dao thứ hai của chương trình – block N14 và block N15. Trong N14, mũi khoan đáy phẳng gia công với tốc độ cắt lớn chỉ đến chiều sâu 0.740 inch, do mũi khoan đáy phẳng hầu như không cắt gọt cho đến chiều sâu đó. Bạn hãy xem sự tính toán chiều sâu trung gian 0.740 từ quy trình dưới đây:

Từ chiều sâu toàn phần 0.94 được cắt gọt bằng mũi khoan tiêu chuẩn (T01), bạn hãy trừ chiều dài đỉnh dao P. Đó là 0.15 với góc đỉnh 118^0 của mũi khoan ∅.5. Kết quả là .79. Từ kết quả đó, bạn trừ khoảng hở 0.50 để có giá trị trục Z là Z-.74. Trong block N15, mũi khoan đáy phẳng sẽ cắt lượng còn lại từ T01, với tốc độ cắt thích hợp, thường được lập trình với tốc độ đủ chậm.

Theo quan điểm gia công, lập trình mũi khoan tâm hoặc khoan điểm trước để khởi đầu lỗ gia công sẽ thuận tiện hơn. Nguyên công này sẽ bảo đảm độ đồng tâm cho cả mũi khoan tiêu chuẩn và mũi khoan đáy phẳng. Sự cải thiện tiếp theo là dùng dao phay mặt đầu thay cho mũi khoan đáy phẳng. Dao này thường có độ cứng vững cao hơn và độ chính xác lớn hơn.

Khoan bằng mũi khoan gắn mảnh hợp kim

Một trong các dụng cụ cắt năng suất cao trong gia công hiện đại là mũi khoan gắn mảnh hợp kim (còn gọi là mũi khoan chắp).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 25.9. Phần cắt gọt của mũi khoan chấp

Loại mũi khoan này sử dụng các mảnh carbides (hợp kim cứng) tương tự các dao phay hoặc tiện. Mũi khoan được thiết kế để khoan vật liệu *rắn*, *không* yêu cầu khoan tâm hoặc khoan điểm, tốc độ trục chính cao và tốc độ cắt tương đối thấp. Trong hầu hết các trường hợp, loại mũi khoan này được dùng để khoan các lỗ *suốt*, nhưng cũng có thể khoan lỗ cụt. Kiểu mũi khoan này còn được dùng để doa với lượng dư gia công thấp đến trung bình.

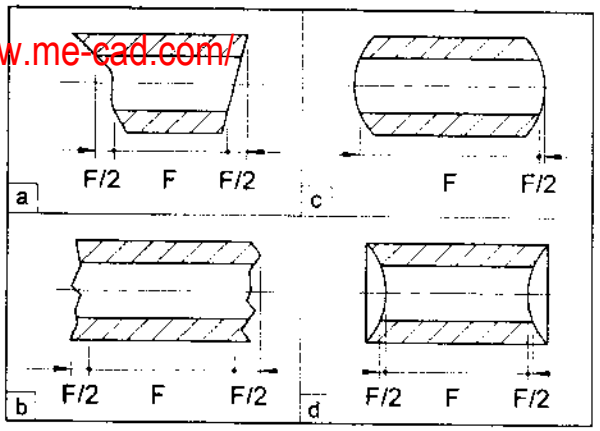
Thiết kế của mũi khoan chấp rất chính xác, bảo đảm chiều dài dụng cụ cắt không đổi, hầu như không cần mài lại lưỡi cắt bị mòn. Hình 25.9 minh họa phần cắt gọt của mũi khoan chấp thông dụng.

Trong minh họa này, đường kính D của mũi khoan là kích cỡ lỗ do mũi khoan tạo ra. Chiều dài đỉnh H do nhà chế tạo mũi khoan xác định. Ví dụ, mũi khoan chấp với đường kính D = 1.25 inch, có chiều dài đỉnh H là .055. Mũi khoan này có thể dùng cho ứng dụng quay hoặc tĩnh tại, ngang hoặc đứng, trên trung tâm gia công hoặc máy tiện. Để đạt hiệu suất cao, cần cung cấp chất làm nguội với áp suất cao *qua suốt* chiều dài mũi khoan đặc biệt khi gia công các vật liệu cứng, lỗ sâu, hoặc nằm ngang. Chất làm nguội không chỉ giải nhiệt mà còn giúp đẩy phoi ra ngoài. Khi sử dụng mũi khoan chấp, cần bảo đảm đủ công suất ở trục chính của máy. Các yêu cầu công suất tại trục chính tăng tỷ lệ với đường kính mũi khoan.

Trên trung tâm gia công, mũi khoan chấp được lắp vào trục máy, do đó quay chung với trục. Theo kiểu lắp này, mũi khoan được dùng trong trục chính phải có độ cứng vững cao. Sự cung cấp chất làm nguội có thể gộp cả chất làm nguội bên trong, bộ ống thích hợp để làm nguội qua lỗ khi mũi khoan được dùng trên trung tâm gia công.

Trên máy tiện, mũi khoan chấp luôn luôn tĩnh tại. Gá lắp chính xác đòi hỏi mũi khoan định vị ở tâm và đồng tâm với đường tâm trục chính.

Cần rất cẩn thận khi khoan trên bề mặt



Hình 25.10. Bề mặt vào hoặc ra không đều đối với tốc độ cắt của mũi khoan chấp: F = tốc độ cắt bình thường, F/2 = giảm tốc độ cắt (một nửa F).

không phẳng. Để có kết quả tốt, bạn hãy dùng mũi khoan chấp trên các bề mặt vuông góc (90^0) với trục mũi khoan (bề mặt phẳng). Trong các giới hạn đó, mũi khoan còn có thể được dùng để vào hoặc ra bề mặt nghiêng lồi, hoặc lõm. Cần giảm tốc độ cắt trong khi cắt gián đoạn. Hình 25.10 minh họa các vị trí cần giảm tốc độ.

Trong minh họa này, chữ F biểu thị vùng được cắt với tốc độ cắt *bình thường* (vào/ra bình thường), F/2 biểu thị vùng yêu cầu *giảm* tốc độ cắt. Đối với sự giảm tốc độ cắt, chỉ cần lập trình một nửa tốc độ cắt bình thường là đủ.

Trên Hình 25.10, hình a là bề mặt nghiêng, b là bề mặt không đều, c và d là các bề mặt lồi và lõm.

Mũi khoan chấp chủ yếu dùng cho khu vực gia công được bảo vệ.

KHOAN NHIỀU ĐOẠN TRONG MỘT LỖ

Phương pháp này còn được gọi là *khoan cắt gián đoạn*. Đây là nguyên công khoan, sử dụng các chu kỳ cố định G83 (tiêu chuẩn) hoặc G73 (tốc độ cao). Khác biệt giữa hai chu kỳ này là phương pháp lùi dao. Trong G83, sự lùi dao sau mỗi đoạn đến mức R (phía trên lỗ), trong G73, chỉ lùi dao rất ngắn (khoảng 02 - .04 inch).

Khoan nhiều đoạn thường dùng cho các lỗ quá sâu đối với một chuyển động dao. Các phương pháp khoan nhiều đoạn còn có vài cơ hội cải tiến kỹ thuật khoan tiêu chuẩn. Dưới đây là một số ứng dụng của phương pháp này khi gia công các lỗ:

- Khoan lỗ sâu.
- Bẻ gãy phoi – có thể dùng cho cả các lỗ ngắn trong vật liệu dẻo.
- Làm sạch phoi tích tụ trong các rãnh thoát phoi của mũi khoan.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Làm nguội và bôi trơn mũi khoan xuyên cho các lưỡi cắt hiệu quả hơn.
- Kiểm soát mũi khoan khi xuyên vào vật liệu

Trong tất cả các trường hợp, các chuyển động khoan của chu kỳ G83 hoặc G73 tạo ra sự cắt gián đoạn có thể được lập trình rất đơn giản bằng cách chuyển biệt giá trị địa chỉ Q trong chu kỳ. Giá trị này là chiều sâu thực của từng đoạn khoan. Giá trị Q càng nhỏ, số đoạn khoan càng lớn và ngược lại. Đối với hầu hết các nguyên công khoan sâu, số lượng *chính xác* các đoạn khoan không quan trọng, nhưng có những trường hợp cần điều khiển chu kỳ khoan nhiều đoạn.

Ứng dụng khoan nhiều đoạn

Đối với hầu hết các ứng dụng khoan nhiều đoạn, chiều sâu Q chỉ cần là chiều sâu *hợp lý*. Ví dụ, lỗ sâu (với chiều sâu Z-2.125 inch ở đỉnh dụng cụ) được khoan với mũi khoan đường kính .250 và chiều sâu từng đoạn là .600. Chu kỳ G83 có thể lập trình như sau:

```
N137 G99 G83 X.. Y.. R0.1 - Z-2.125 Q0.6 F8.0
```

Các giá trị lập trình này là hợp lý. Đối với hầu hết các ứng dụng, số lượng các đoạn khoan thường không quá quan trọng.

Tính toán số đoạn khoan

Nếu số đoạn khoan trong chu kỳ G83/G73 là quan trọng, cần phải *tính toán* giá trị đó. Biết số đoạn và giá trị Q đối với chiều sâu toàn phần cho trước thường không quan trọng. Nếu chương trình chạy hiệu quả, thì không cần chỉnh sửa. Để tìm số đoạn thích hợp cho chu kỳ G83/G73, điều quan trọng là biết *khoảng cách toàn phần* của hành trình mũi khoan giữa mức R và chiều sâu Z (theo giá trị số gia). Điều quan trọng tiếp theo là biết giá trị Q. Khoảng cách hành trình chia cho Q sẽ được số đoạn cần khoan:

$$P_n = \frac{T_d}{Q}$$

Trong đó:

P_n = Số đoạn
T_d = Khoảng cách toàn phần hành trình dao
Q = Chiều sâu đoạn được lập trình

Ví dụ, trong chu kỳ G83

```
NT3 G99 G83 X Y R0.125 Z-1.225 Q.05 F12.0
```

Khoảng cách hành trình toàn phần là 1.350 chia cho .500 sẽ được kết quả là 2.7. Do số đoạn chỉ có thể là số nguyên, số nguyên *lớn hơn* gần nhất sẽ là số đoạn, trong trường hợp này là số 3.

Chọn số đoạn

Trong thực tế thường lập trình với số đoạn *mong muốn*. Nếu chỉ một số lượng xác định các đoạn là đủ để đạt hiệu suất tối ưu, từ đó có thể dễ dàng tính giá trị Q. Do giá trị Q chuyên biệt *chiều sâu của từng đoạn khoan* nhưng *không* xác định số đoạn, cần dùng công thức toán để chọn chiều sâu Q, tương ứng số đoạn mong muốn.

Ví dụ, bạn cần *ba đoạn* khoan trong chu kỳ dưới đây, giá trị chiều sâu Q sẽ là:

```
N14 G99 GR3 X.. Y.. R0.1 Z-1.238 Q.. F12.0
```

Hành trình khoan toàn phần từ mức R đến chiều sâu Z là 1.338 inch. Để tính giá trị Q, công thức có dạng:

$$Q = \frac{T_d}{P_n}$$

Trong đó:

Q = Chiều sâu đoạn được lập trình
T_d = Khoảng cách toàn phần hành trình dao.
P_n = Số đoạn mong muốn.

Sử dụng công thức nêu trên, kết quả 1.338/3 là .446. Do đó, chiều sâu Q trong block G83 là Q0.446:

```
N14 G99 G83 X.. Y.. R0.1 Z-1.239 Q.. F12.0
```

Trong trường hợp này không cần làm tròn số. Bạn hãy quan sát tình huống khác, khoảng cách hành trình thay đổi rất ít:

```
N24 G99 G83 X.. Y.. R0.1 Z-1.239 Q.. F12.0
```

Kết quả của 1.339/3 là .446333333 – số cần được làm tròn đến 4 chữa số thập phân (đơn vị Anh). Về toán học, làm tròn với 4 chữ số thập phân là .4463. Bạn hãy quan sát chiều sâu của từng đoạn.

Đoạn 1	.4463	Chiều sâu tích lũy ...	.4463
Đoạn 2	.4463	Chiều sâu tích lũy ...	.8926
Đoạn 3	.4463	Chiều sâu tích lũy ...	1.3389
Đoạn 4	.0001	Chiều sâu tích lũy ...	1.3390

Ở đây cần đến *bốn đoạn* và đoạn cuối chỉ cắt .0001, hầu như không đáng kể. Trong các trường hợp đó, với lần cắt cuối cùng rất nhỏ và không hiệu quả, bạn cần làm tròn giá trị Q *theo hướng tăng*, trong ví dụ nêu trên, tối thiểu là .4464, có thể làm tròn đến .447.

```
N14 G99 G83 X Y R0.1 Z-1.239 Q0.447 F12.0
```

Bạn cần nhớ, dụng cụ cắt *không* bao giờ vượt quá chiều sâu Z lập trình nhưng có thể đến chiều sâu này với cách thức rất không hiệu quả, do đó cần hiệu chỉnh.

http://www.me-cad.com/

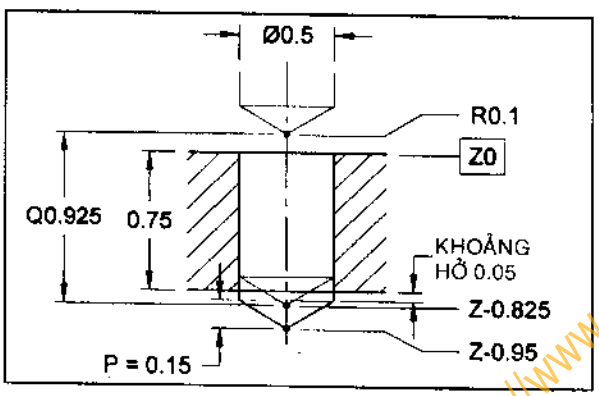
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Điều khiển chiều sâu xuyên qua

Phương pháp lập trình ít được sử dụng, nhưng rất hiệu quả, là sử dụng chu kỳ khoan gián đoạn để điều khiển *sự xuyên qua* của mũi khoan lỗ suốt, bất kể kích cỡ mũi khoan hoặc chiều dày vật liệu. Dưới đây là các cơ sở. Trong nhiều vật liệu có độ dẻo cao, khi mũi khoan bắt đầu xuyên đến đáy chi tiết (khoan lỗ suốt), có thể xuất hiện các điều kiện khó gia công. Mũi khoan có xu hướng *đẩy* vật liệu ra ngoài thay vì cắt gọt. Điều này rất phổ biến khi mũi khoan hơi cùn, vật liệu có độ dẻo cao, hoặc tốc độ cắt hơi cao. Các điều kiện bất lợi này còn làm cho nhiệt phát sinh ở lưỡi cắt, thiếu bôi trơn ở đầu mũi khoan, và các yếu tố khác.

Giải pháp khả dĩ cho vấn đề này là giảm áp suất mũi khoan khi mũi khoan *ở khoảng nửa đường* xuyên qua lỗ, nhưng *chưa* hoàn toàn xuyên qua (Hình 25.11).



Hình 25.11. Điều khiển sự xuyên qua lỗ sử dụng chu kỳ khoan G83.

Chu kỳ khoan gián đoạn G83 là rất thích hợp, nhưng tính toán chiều sâu Q là cực kỳ quan trọng. Tổng số đoạn không quan trọng, chỉ hai đoạn cuối là có tính quyết định đối với mục đích này. Để kiểm soát vấn đề liên quan đến sự xuyên thấu mũi khoan, chỉ cần hai chuyển động. Hình 25.11 minh họa hai vị trí mũi khoan Ø1/2 khoan qua tấm dày 3/4 inch.

Đối với nhiều ứng dụng, loại lỗ này không yêu cầu xử lý đặc biệt chỉ cần một lần cắt xuyên qua (sử dụng chu kỳ G81) và không cần khoan gián đoạn. Bạn hãy đánh giá giải pháp của tình huống này. Mũi khoan Ø.5 có chiều dài đỉnh .300 × 500 = .150. Bạn hãy lấy một nửa chiều dài đỉnh mũi khoan (.075) làm giá trị xuyên qua thứ nhất, sẽ đưa mũi khoan xuống *dưới* tấm dày 3/4 inch một khoảng là .075 inch, đến chiều sâu Z-0.825. *Chiều sâu này sẽ đạt được với giá trị chiều sâu Q*. Bạn cần nhớ chiều sâu Q là giá trị *số gia*, đo từ mức R, trong trường hợp này là R 0.1. Điều đó sẽ chuyên biệt chiều sâu Q là Q 0.925 (.100 trên và .825 dưới Z0). Chiều sâu Z lập trình là chiều sâu khoan hoàn tất. Nếu cộng thêm khoảng hở .05 dưới tấm phôi, chiều sâu Z sẽ là tổng chiều dài phôi (.75), khoảng hở (.05), và chiều dài đỉnh mũi khoan (.150), giá trị lập trình là Z-0.95:

```
G99 G83 X.. Y.. R0.1 Z-0.95 Q0.925 F..
```

Kỹ thuật này không chỉ giải quyết vấn đề liên quan đến chi tiết cụ thể, mà còn cho thấy *tính sáng tạo* trong *lập trình* là không thể thiếu.

CHUỐT

Chuốt rất gần gũi với khoan, ít nhất là về phương pháp lập trình. Mũi khoan được dùng *để tạo ra* lỗ, chuốt được dùng để *làm rộng* lỗ hiện hữu.

Các dao chuốt có hình trụ hoặc côn, thường được thiết kế nhiều hơn hai rãnh với các cấu hình khác nhau, được chế tạo bằng thép gió, hợp kim Fe-Co, Carbides (hợp kim cứng). Mỗi thiết kế dao chuốt đều có các ưu điểm và nhược điểm riêng. Ví dụ, dao chuốt Carbides, có tính chống mài mòn rất cao nhưng có thể không kinh tế đối với một số loại lỗ. Dao chuốt thép gió tuy kinh tế nhưng mòn nhanh hơn hợp kim cứng Carbides. Cần rất cẩn thận khi chọn dao chuốt cho từng ứng dụng gia công cụ thể.

Dao chuốt là dụng cụ cắt *tạo kích thước* chính xác, không dùng để gia công cắt gọt với lượng dư lớn. Trong nguyên công chuốt, lỗ hiện hữu sẽ được *tạo kích cỡ* bảo đảm đúng dung sai cho phép và tăng độ bóng bề mặt. Sự chuốt sẽ *không* bảo đảm độ đồng tâm cho lỗ. Đối với các lỗ yêu cầu cả độ đồng tâm cao *và* dung sai chính xác, cần khoan tâm hoặc khoan điểm trước, sau đó khoan bình thường, doa thô, và cuối cùng là hoàn tất bằng dao chuốt.

Nguyên công chuốt đòi hỏi chất làm nguội để tăng độ bóng bề mặt và loại bỏ phoi trong khi cắt gọt. Các chất làm nguội tiêu chuẩn là hoàn toàn thích hợp, do khi chuốt nhiệt phát sinh không nhiều. Chất làm nguội còn có vai trò rửa sạch phoi và duy trì chất lượng bề mặt cao.

Thiết kế dao chuốt

Về thiết kế, dao chuốt có hai tính năng liên quan trực tiếp với gia công và lập trình CNC thứ nhất là *thiết kế rãnh thoát phoi* và thứ hai là *vạt góc đỉnh dao*.

Hầu hết các dao chuốt đều được thiết kế với hướng rãnh trái, thích hợp để gia công các lỗ *suốt*. Trong khi cắt gọt, rãnh xoắn trái "đẩy"

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

phoi về phía đáy lỗ và ra ngoài. Đối với lỗ cụt, dao chuốt với rãnh thoát phoi xoắn trái sẽ không thích hợp.

Yếu tố thứ hai của thiết kế dao doa là *vạt góc đỉnh dao*. Để tiến vào lỗ hiện hữu chưa lả miệng, cần có khoảng hở dẫn tiến. Đầu dao phải có khoảng hở này. Một số dao chuốt còn có phần côn ở đầu để dễ đi vào lỗ. Đầu dẫn tiến của dao chuốt còn được gọi là "*dẫn tiến vát*" và phần nghiêng được gọi là "*góc tiến*". Cả hai yếu tố này đều được xem xét khi lập trình.

Tốc độ trục chính khi chuốt lỗ

Tương tự các nguyên công khoan, tốc độ trục chính được chọn khi chuốt phải phù hợp với loại vật liệu được gia công. Các yếu tố khác, chẳng hạn gá lắp chi tiết, độ cứng vững, kích cỡ và độ bóng bề mặt của lỗ hoàn tất,... đều góp phần vào sự lựa chọn tốc độ trục chính. Theo nguyên tắc lập trình chung, tốc độ trục chính khi chuốt sẽ hợp lý nếu sử dụng hệ số hiệu chỉnh 0.660 (2/3), dựa trên tốc độ được dùng để khoan vật liệu *cùng loại*. Ví dụ, nếu tốc độ khoan tối ưu là 500 r/min, hai phần ba (2/3) tốc độ này sẽ là *hợp lý* để chuốt:

500 x .660 = 330 r/min

Bạn không được lập trình chuyển động chuốt với trục chính quay ngược, các lưỡi cắt có thể bị gãy hoặc bị cùn rất nhanh.

Tốc độ cắt khi chuốt

Tốc độ cắt khi chuốt được lập trình cao hơn so với khoan, thường *gấp hai* hoặc *gấp ba lần*. Mục đích của tốc độ cắt cao là buộc dao chuốt phải *cắt gọt*, thay vì chà xát lên bề mặt vật liệu. Nếu tốc độ cắt quá thấp, dao chuốt sẽ mòn nhanh. Tốc độ cắt thấp gây ra áp suất lớn khi dao chuốt làm rộng lỗ thay vì cắt gọt lượng dư.

Lượng dư gia công

Lượng dư là phần vật liệu còn lại để gia công tinh. Lỗ để chuốt phải nhỏ hơn (*dưới kích thước*) so với lỗ khoan trước hoặc doa trước, đây là yêu cầu logic. Nhà lập trình sẽ quyết định mức độ nhỏ hơn. Lượng dư nhỏ để chuốt làm cho dao chuốt bị mòn sớm. Lượng dư quá lớn sẽ làm tăng áp suất cắt và lưỡi dao chuốt có thể bị gãy.

Nguyên tắc chung là để lại khoảng 3% đường kính dao chuốt làm lượng dư gia công. Điều này áp dụng cho *đường kính lỗ – thay vì từng phía*. Ví dụ, dao chuốt 3/8 (∅ .375) sẽ làm việc tốt trong hầu hết các điều kiện nếu lỗ được chuốt có đường kính khoảng .364 inch:

.375 – (.375 x 3/100) = 3.6375 = .364

Nói chung, trong thực tế không có mũi khoan có thể khoan đường kính lỗ chính xác theo yêu cầu. Điều đó có nghĩa là sử dụng dao doa để *gia công lỗ* trước khi chuốt. Điều này còn có nghĩa là phải thêm dụng cụ cắt, thời gian gá lắp, chương trình dài hơn và các nhược điểm khác, nhưng chất lượng lỗ tăng rõ rệt. Trong các trường hợp đó, đối với các vật liệu có độ dẻo cao, vật liệu "thời đại không gian", lượng dư gia công còn lại trong lỗ trước khi chuốt, phải giảm đáng kể.

Các vấn đề chuốt

Phương pháp chung để chuốt hầu như tương tự các nguyên công khác. Khi khoan lỗ cụt, sau đó chuốt lỗ, một phần phoi từ quá trình khoan luôn luôn đọng lại trong lỗ, có thể cản trở quá trình chuốt. Sử dụng hàm dừng chương trình M00 *trước* khi chuốt cho phép người vận hành loại bỏ toàn bộ phoi còn lại trước khi chuốt

Kích cỡ dao chuốt luôn luôn quan trọng. Dao chuốt thường được dùng để tạo ra sự *lắp ghép chặt* (ép) hoặc *lắp ghép trượt* với các khoảng dung sai tương ứng cho lỗ chuốt.

Lập trình dao chuốt đòi hỏi chu kỳ cố định. Chu kỳ nào là thích hợp nhất? Không có *chu kỳ chuốt* được xác định một cách trực tiếp. Suy nghĩ về các ứng dụng gia công truyền thống, phương pháp chuốt được chấp nhận là phương pháp *ăn vào và ăn ra*. Phương pháp này đòi hỏi chuyển động ăn dao để cắt gọt vật liệu trong lỗ và yêu cầu chuyển động trở lại vị trí xuất phát, để bảo đảm chất lượng lỗ – kích cỡ và độ bóng bề mặt. Có thể phải lập trình chuyển động nhanh ra khỏi lỗ chuốt để tiết kiệm thời gian, nhưng sẽ ảnh hưởng đến chất lượng. Để gia công tối ưu, *sự ăn ra* lỗ chuốt là cần thiết. Chu kỳ cố định thích hợp trên các bộ điều khiển Fanuc là G85, cho phép các chuyển động *ăn vào* và *ăn ra*. Tốc độ cắt gọt của chu kỳ này là như nhau cho cả hai chuyển động. Sự thay đổi tốc độ cắt sẽ tác động đến cả hai chuyển động - *vào* và *ra*.

DOA MỘT ĐIỂM

Nguyên công gia công lỗ kế tiếp được gọi là *doa*. Doa, với ý nghĩa gia công lỗ, là gia công lỗ – lỗ chỉ dọc theo trục Z, thường áp dụng trên máy phay CNC và trung tâm gia công. Phương pháp này còn được gọi là "*doa một điểm*", do dụng cụ cắt thông dụng nhất là thanh doa chỉ có *một* lưỡi cắt. Doa trên máy tiện CNC được coi là gia công *biên dạng* sẽ được trình bày trong các Chương 33 và 34.

Nhiều ứng dụng đòi hỏi các lỗ chính xác

Nhiều tài liệu hay có tại : http://www.me-cad.com/

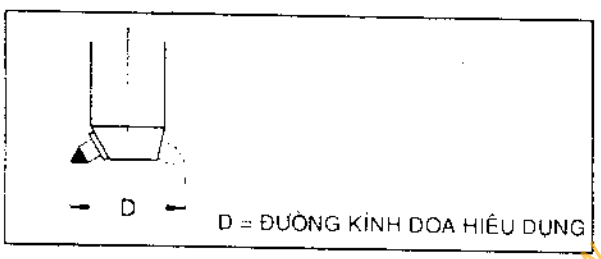
Nhiều tài liệu hay có tại : http://www.me-cad.com/

trước đây được thực hiện trên máy doa đặc biệt, ngày nay có thể áp dụng trên trung tâm gia công CNC sử dụng dao doa một lưỡi cắt (một điểm). Máy công cụ CNC hiện đại được chế tạo với độ chính xác rất cao, đặc biệt độ định vị và lặp lại – dao doa thích hợp có thể tạo ra các lỗ chất lượng cao.

Dao doa một điểm

Đối với mục đích thực tiễn, doa một điểm là gia công *tinh*, hay ít nhất cũng là *bán tinh*. Công việc chính là làm rộng hoặc *tạo kích thước*, cho lỗ đã được khoan, đột hoặc đúc áp lực. Dao doa làm việc trên đường kính lỗ, tạo ra đường kính lỗ mong muốn, trong phạm vi dung sai cho phép với độ bóng bề mặt cao.

Mặc dầu có nhiều thiết kế dao doa, nhưng dao doa một điểm thường được thiết kế theo kiểu *thanh* gắn các mảnh hợp kim cứng. Các mảnh này được gắn ở đầu thanh (hoặc cán dao) có sự điều chỉnh vi mô để tinh chỉnh đường kính doa hiệu dụng (Hình 25.12).



Hình 25.12. *Đường kính hiệu dụng của dao doa một điểm.*

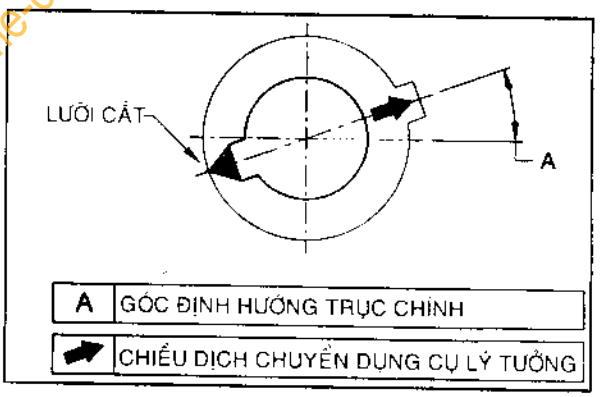
Các kỹ thuật lập trình như nhau áp dụng cho các thanh doa có thiết kế khác nhau, ví dụ *dụng cụ khối*. Dụng cụ khối là thanh doa với hai lưỡi cắt cách nhau 180^0. Nếu *không* có cơ cấu điều chỉnh đường kính trên cán dao, đường kính doa hiệu dụng phải được *xác lập trước*, sử dụng thiết bị đặc biệt hoặc phương pháp *thử và sai*, tuy chậm nhưng đủ chính xác. Thử và sai được coi là một trong các phương pháp khả dụng cho thanh doa một điểm.

Tương tự các dụng cụ cắt khác, thanh doa một điểm đạt được kết quả cắt gọt tốt khi chúng ngắn, cứng vững, và đồng tâm với đường tâm trục chính. Một trong các nguyên nhân chính làm giảm chất lượng lỗ doa là *sự lệch thanh doa* đối với cả phay và tiện. Đỉnh dao (thường là mảnh Carbide) phải được mài chuẩn xác với các khoảng hở và dạng hình học cắt gọt thích hợp. Vị trí thanh doa trong trục chính hoặc hướng lưỡi cắt là rất quan trọng đối với nhiều nguyên công doa trên trung tâm gia công.

Hướng trục chính

Dụng cụ cắt hình trụ bất kỳ, chẳng hạn mũi khoan hoặc dao phay mặt đầu, có thể vào và ra lỗ dọc theo trục Z, với vài chú ý lập trình về chất lượng lỗ gia công. Với doa, tính toàn vẹn bề mặt lỗ là *rất quan trọng*. Nhiều nguyên công doa đòi hỏi dụng cụ cắt không làm hư hại bề mặt lỗ khi ra khỏi lỗ. Có một phương pháp – sử dụng chu kỳ G76 hoặc G87 với tính năng *định hướng trục chính* của máy và *dịch chuyển* dao doa ra xa bề mặt lỗ gia công. Tính năng này đã được trình bày trong Chương 11.

Mục đích duy nhất của định hướng trục chính là thay cán dao vào đúng vị trí một cách chính xác sau mỗi lần thay dao. Không có sự định hướng lại trục chính, đỉnh dao sẽ dừng lại ở vị trí ngẫu nhiên trên chu vi. Định hướng trục chính khi doa chỉ là một phần của giải pháp. Phần thứ hai là xác lập vị trí đỉnh dao doa. Đây thường là trách nhiệm của người vận hành, do cần thực hiện trong khi gá lắp tại máy. Lưỡi cắt của thanh doa phải được xác lập sao cho khi xảy ra sự dịch chuyển trong chu kỳ cố định G76 hoặc G87, đỉnh này sẽ hướng *ra xa* vách lỗ gia công, lý tưởng là theo vector XY liên quan với góc định hướng của trục chính (Hình 25.13).



Hình 25.13. *Thanh doa một điểm và góc định hướng trục chính.*

Định hướng trục chính được thiết kế ở nhà máy và *không thay đổi*. Nhà lập trình phải xét chiều dài và chiều định hướng trục chính.

Khi trục máy được định hướng, trục phải trong chế độ dừng. Trục không thể quay trong khi gia công yêu cầu sự dịch chuyển (ngang của trục). Để biết chi tiết về chu kỳ doa tinh G76 và chu kỳ doa ngược, bạn hãy xem lại Chương 24. Người vận hành máy *phải* biết sự định hướng trục chính và chiều dịch chuyển thực tế của dụng cụ cắt.

Lập trình lỗ doa, sau đó sẽ chuốt, đòi hỏi

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

thanh doa, đảm bảo đường kính và độ bóng bề mặt của lỗ hoàn tất. Nếu doa là nguyên công cuối cùng, độ bóng bề mặt sẽ *rất* quan trọng. Rất khó lui dao doa mà không để lại các vết trên mặt trục của lỗ. Trong trường hợp đó, cần chọn chu kỳ thích hợp, có thể là chu kỳ doa chính xác G76.

Dụng cụ khối

Khi sử dụng thanh doa một điểm để gia công thô hoặc bán tinh, có một tùy chọn hiệu quả hơn. Tùy chọn này cũng sử dụng dao doa nhưng có *hai* lưỡi cắt (cách nhau 180°) thay vì một – được gọi là *dụng cụ khối*. Loại dụng cụ cắt này không dùng cho gia công tinh, do không thể *dịch chuyển* (ngang). Cách duy nhất lập trình dụng cụ khối là trong chuyển động dao "*vào và ra*". Mọi chuyển động "vao" đều theo tốc độ cắt chuyên biệt. Khi chuyển động "*ra*" một số là chuyển động cắt gọt, số khác là chuyển động nhanh, tùy chọn theo lựa chọn chu kỳ. Chu kỳ có thể sử dụng loại dụng cụ khối là G81 và G82 (ăn dao-vào-nhanh-ra), G85 và G89 cắt gọt cả vào và ra trong khi trục chính quay, với G86 dao rút ra trong khi trục chính không quay.

Ưu thế lớn nhất của dụng cụ khối là có thể lập trình với tốc độ cắt cao. Ví dụ, nếu tốc độ cắt của dao một điểm là .007/điểm, dụng cụ khối sẽ gấp đôi, .014/điểm hoặc cao hơn. Các dụng cụ khối thường có đường kính Ø.750 trở lên.

DOA VỚI DỊCH CHUYỂN DAO

Có hai chu kỳ đòi hỏi dụng cụ cắt dịch chuyển ra xa đường tâm lỗ hiện hành, G76 và G87. Chu kỳ G76 hữu dụng hơn và được minh họa trong chương trình O2504.

Chu kỳ doa chính xác G76

Chu kỳ G76 được dùng cho các lỗ đòi hỏi chất lượng cao về kích thước và độ bóng bề mặt. Quá trình doa là bình thường, nhưng sự lùi dao ra khỏi lỗ là đặc biệt. Thanh doa dừng ở đáy lỗ theo vị trí định hướng, dịch chuyển ra xa theo giá trị Q trong chu kỳ và lui về vị trí khởi đầu, dịch chuyển trở lại vị trí bình thường.

Chu kỳ G76 đã được trình bày chi tiết trong Chương 24. Trong chương này là ví dụ lập trình thực tế, gia công lỗ Ø25 mm (Hình 25.14).

Từ bản vẽ, chỉ xét lỗ Ø25 mm, sự nhập chương trình rất đơn giản:

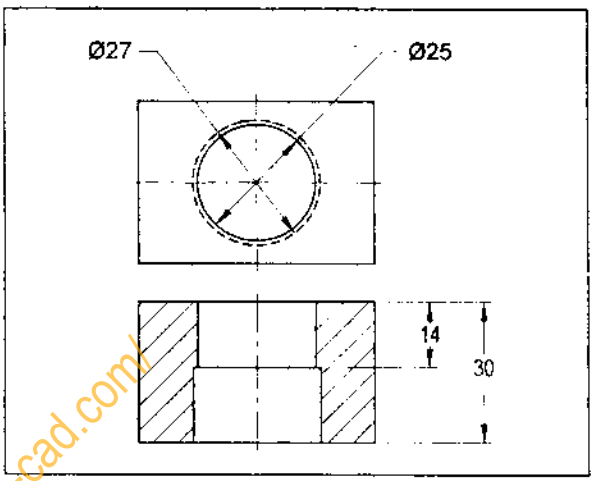
```
N. . G99 G76 X0 Y0 R2.0 Z-31.0 Q0.3 F125.0
```

Lỗ được doa với chu kỳ G76 sẽ có chất lượng cao.

Chu kỳ doa ngược G87

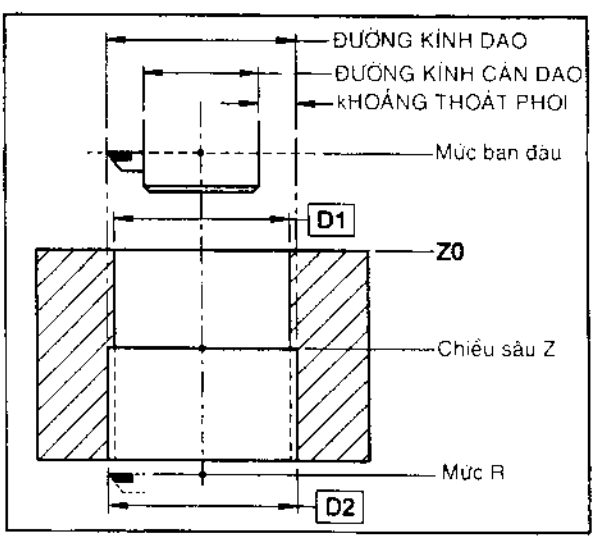
Chu kỳ doa ngược G87 có một số ứng dụng, nhưng không phổ biến. Đây là chu kỳ doa làm việc ngược chiều với các chu kỳ khác – *từ phía sau chi tiết*. Nói chung, doa ngược bắt đầu từ đáy lỗ, "phía sau chi tiết" và doa từ đáy đi lên, theo chiều Z dương.

Chu kỳ G87 đã được trình bày chi tiết trong Chương 24. Hình 25.14 minh họa đường kính 27 mm được doa với sự gá lắp chi tiết lỗ 25 mm. Đường kính lớn là ở "mặt dưới của chi tiết". Do đó sẽ được doa ngược, sử dụng chu kỳ G87.



Hình 25.14. *Bản vẽ dùng cho ví dụ lập trình G76 và G87 – Chương trình O2504*

Hình 25.15 minh họa gá lắp dao để doa lỗ Ø27 từ đáy lỗ đi lên. Trong minh họa này, D1 là đường kính lỗ nhỏ, D2 là đường kính lỗ sẽ doa ngược. D2 luôn luôn lớn hơn D1. Bạn cần bảo đảm có đủ khoảng hở ở cán dao doa bên trong lỗ và ở đáy lỗ.



Hình 25-15. *Gá lắp để doa ngược*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ví dụ lập trình

Để minh họa chương trình hoàn chỉnh, ở đây sẽ dùng 4 dao – *mũi khoan điểm (T01), mũi khoan (T02), thanh doa tiêu chuẩn (T03), và thanh doa ngược (T04).* Chương trình là O2504.

```
O2504 (G76 AND G87 BORING)
(T01 - 15 MM DIA SPOT DRILL - 90 DEG)
N1 G21
N2 G17 G40 G80 T01
N3 M06
N4 G90 G54 G00 X0 Y0 S1200 M03 T02
N5 G43 Z10.0 H01 M08
N6 G99 G82 R2.0 Z-5.0 P100 F100.0
N7 G80 Z10.0 M09
N8 G28 Z10.0 M05
N9 M01

(T02 - 24 MM DIA DRILL)
N10 T02
N11 M06
N12 G90 G54 G00 X0 Y0 S650 M03 T03
N13 G43 Z10.0 H02 M08
N14 G99 G81 R2.0 Z-39.2 F200.0
N15 G80 Z10.0 M09
N16 G28 Z10.0 M05
N17 M01

(T03 - 25 MM DIA STANDARD BORING BAR)
N18 T03
N19 M06
N20 G90 G54 G00 X0 Y0 S900 M03 T04
N21 G43 Z10.0 H03 M08
N22 G99 G76 R2.0 Z-31.0 Q0.3 F125.0  (25 DIA)
N23 G80 Z10.0 M09
N24 G28 Z10.0 M05
N25 M01

(T04 - 27 MM DIA BACK BORING BAR)
N26 T04
N27 M06
N28 G90 G54 G00 X0 Y0 S900 M03 T01
N29 G43 Z10.0 H04 M08
N30 G98 G87 R-32.0 Z-14.0 Q1.3 F125.0 (27 DIA)
N31 G80 Z10.0 M09
N32 G28 Z10.0 M05
N33 G28 X0 Y0
N34 M30
%
```

Bạn hãy tuân thủ tất cả các nguyên tắc khi lập trình hoặc gá lắp chi tiết với chu kỳ cố định G76 hoặc G87 trong chương trình. Một số nguyên tắc đó nhằm bảo đảm tính an toàn.

Các chú ý trong lập trình và gá lắp

Các chú ý về nguyên công doa liên quan đến một số vấn đề đặc biệt là rất cần thiết để áp dụng thành công các chu kỳ G76 và G87. Dưới đây sẽ tóm tắt các chú ý quan trọng.

- Doa lỗ suốt phải được thực hiện trước khi doa ngược
- Chu kỳ doa G76 phải được lập trình qua toàn bộ chiều sâu lỗ, không được phép nửa chừng.
- Đối với chu kỳ G76, chỉ yêu cầu giá trị Q tối thiểu (ví dụ 0.3 mm, 0.012 inch)
- Đối với chu kỳ G87, giá trị Q phải lớn hơn một nửa hiệu số giữa hai đường kính:
  (D2 - D1)/2 = (27 - 25)/2 = 1,
  cộng với giá trị Q tiêu chuẩn tối thiểu (0.3 mm)
- Luôn luôn chú ý thân (cán) thanh doa sao cho không chạm vào bề mặt lỗ trong khi dịch chuyển. Điều này có thể xảy ra với các thanh doa lớn, lỗ nhỏ, hoặc lượng dịch chuyển lớn.
- Luôn luôn chú ý thanh doa, sao cho không chạm vào chướng ngại (vật cản) bên dưới chi tiết. Bạn cần nhớ giá trị bù chiều dài dụng cụ cắt được đo đến lưỡi cắt, không phải đến đáy thực của dụng cụ doa.
- G87 luôn luôn được lập trình trong chế độ G98, không được phép lập trình với G99.
- Bạn phải biết chiều dịch chuyển và gá lắp dao hợp lý.

MỞ RỘNG CÁC LỖ

Lỗ hiện hữu có thể được khoét rộng từ phía trên. Để mở rộng lỗ từ phía trên, bạn có thể sử dụng một trong ba phương pháp làm rộng lỗ hiện hữu. Các phương pháp này được sử dụng phổ biến trong nhiều xưởng cơ khí. Chúng gồm:

- Khoét miệng
- Doa miệng
- Lả miệng

Cả ba phương pháp gia công đó đều mở rộng lỗ hiện hữu với cùng mục đích – cho phép chi tiết lắp ghép tựa chính xác trong lỗ bằng cách tạo ra bề mặt sạch. Ví dụ, đầu bulông tựa trên bề mặt phẳng sẽ yêu cầu nguyên công khoét hoặc lả miệng. Cả ba nguyên công này đều đòi hỏi sự thẳng hàng với lỗ hiện hữu (đồng tâm). Kỹ thuật lập trình về cơ bản là như nhau cho cả ba, mũi khoan có kích cỡ tương đương. Lỗ bất kỳ cần làm rộng phải được khoan trước khi thực hiện một trong ba nguyên công nêu trên.

Khoét miệng

Khoét miệng là nguyên công mở miệng lỗ hiện hữu theo hình côn đến chiều sâu yêu cầu. Khoét miệng được dùng cho các lỗ cần lắp đầu bulông hình côn.Từ ba nguyên công nêu trên, khoét miệng đòi hỏi tính toán chiều sâu chính xác. Khoét miệng thường gồm ba góc:

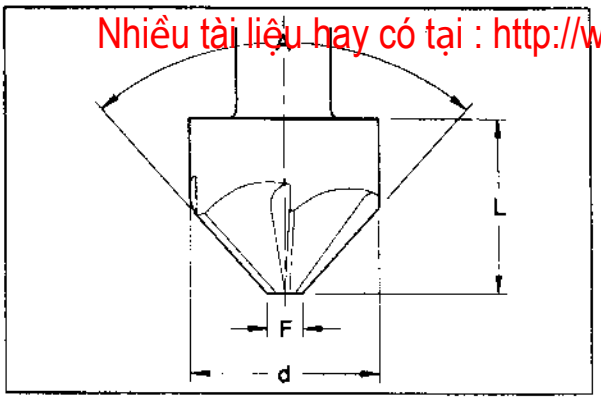
- 60 độ
- 82 độ – thông dụng nhất
- 90 độ

Cũng có thể sử dụng các góc khác, tùy theo yêu cầu cụ thể.

Để minh họa kỹ thuật lập trình và các tính toán, cần biết trước dụng cụ cắt. Hình 25.16 minh họa dụng cụ khoét miệng điển hình.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 25.16. *Các kích thước của dụng cụ khoét miệng*

Trong minh họa này, d là đường kính thân mũi khoét miệng, A là góc khoét, F là đường kính đầu mũi khoét (bằng zero nếu mũi khoét nhọn), l là chiều dài mũi khoét.

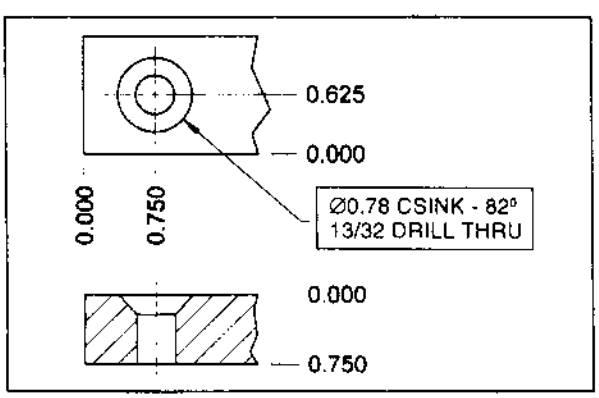
Lập trình khoét miệng đòi hỏi dữ liệu trong bản vẽ. Thông tin này thường được ghi rõ trên bản vẽ, ví dụ:

```
.78 DIA CSINK - 82 DEG
```
(Khoét miệng đường kính 0.78 inch, góc 82°)

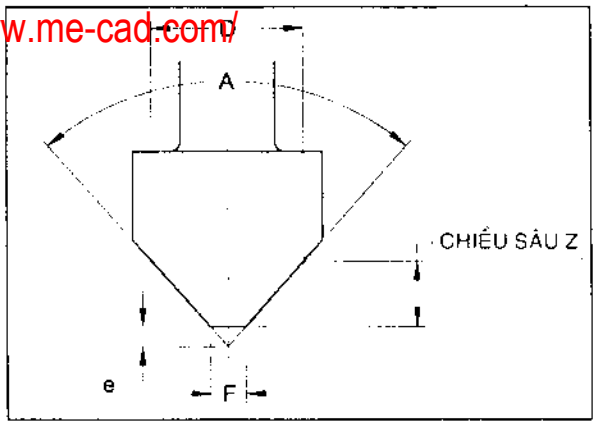
```
13/32 DRILL THRU
```
(Khoan lỗ suốt, mũi khoan Ø13/32)

Có một vấn đề khi lập trình khoét miệng. Đường kính khoét miệng phải chính xác. Đó là Ø.78 trên bản vẽ. Góc khoét là 82°. Đường kính chính xác có thể *được tạo ra* bằng cách tính toán chiều sâu Z một cách cẩn thận. Điều này không quá khó, do có thể sử dụng hệ số K (Bảng 25.2) về chiều dài đỉnh dụng cụ cắt, sau đó tính chiều sâu cắt, tương tự như khi tính cho mũi khoan. Vấn đề ở đây là hệ số K đối với đỉnh mũi khoan luôn luôn *giả thiết là đỉnh nhọn*. Dụng cụ khoét không phải lúc nào cũng có đỉnh nhọn (ngoại trừ loại có kích thước nhỏ). Thay vào đó, chúng có đường kính F ở mũi khoét (Hình 25.16).

Hình 25.17 minh họa ví dụ về yêu cầu khoét miệng, được nêu trên bản vẽ.



Hình 25.17. *Ví dụ lập trình sẽ về khoét miệng.*



Hình 25.18. *Dữ liệu cần thiết để tính toán chiều sâu Z khoét miệng, khi biết góc A, các đường kính D và F.*

Hình 25.18 minh họa các kích thước khoét miệng đã biết và chưa biết, được yêu cầu để lập trình dụng cụ khoét.

Quá trình tính toán tương đối đơn giản. Trước hết, bạn xác định chiều cao e khi biết đường kính F, sử dụng các hệ số tiêu chuẩn (Bảng 25.2) như khi áp dụng cho chiều dài đỉnh mũi khoan:

60^0	K = 0.866
82^0	K = 0.575
90^0	K = 0.500

Trong minh họa này, D là đường kính khoét theo yêu cầu, A là góc khoét, F là đường kính đầu mũi khoan, e là chiều cao phần đỉnh nhọn, và Z-DEPTH là chiều sâu dao khoét được lập trình. Với góc A = 82^0, Đường kính F theo tiêu chuẩn là 3/16 (.1875), chiều cao đỉnh nhọn e được tính như sau:

e = .1875 x .575 (K = .375 ứng với góc 82^0)
e = .1078

Chiều sâu Z của dụng cụ *có đỉnh nhọn* sẽ là

```
Z sâu =.78 x.575 = .4485
```

Do giá trị này *gộp* cả chiều cao phần đỉnh nhọn, để tìm chiều sâu Z thực, cần *trừ* giá trị e từ chiều sâu Z lý thuyết.

```
Z sâu =.4485 -.1078 =.3407
```

Đây là chiều sâu Z *lập trình* và block chương trình để khoét miệng theo yêu cầu bản vẽ sẽ như sau:

```
N3.5 G99 X0.75 Y0.625 R0.1 Z-0.3407 P200 F8.0
```

Chú ý, mức R có thể bị *hạ thấp*, do lỗ suốt đã được khoan trước. Bạn cần cẩn thận, mức R có thể có giá trị âm. Bạn hãy lập trình G98 và mức ban đầu nhỏ, ví dụ Z0.1:

```
N34 G43 Z0.1 H03 M08 (0.1 IS INITIAL LEVEL)
N35 G98 G82 X.75 Y.625 R-.2 Z-.3407 P200 F8.0
```

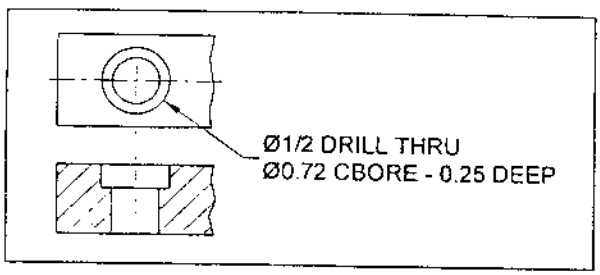
http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Doa miệng

Doa miệng là nguyên công mở rộng lỗ hiện hữu theo hình trụ đến chiều sâu xác định. Doa miệng được dùng cho các lỗ để lắp bulông đầu tròn, thường áp dụng trên các bề mặt thô, không đều, hoặc không vuông góc (90°) với đường tâm bulông. Khi chọn dụng cụ cắt, bạn hãy dùng dao doa miệng được thiết kế đặc biệt cho kiểu gia công này hoặc dùng dao phay mặt đầu thích hợp. Chương trình sử dụng chu kỳ cố định G82. Do chiều sâu doa miệng luôn luôn cho trước, bạn không cần thực hiện các tính toán (Hình 25.19).



Hình 25.19. Ví dụ lập trình nguyên công doa miệng

Trong ví dụ này, lỗ Ø1/2 inch đã được khoan trước. Block chương trình rất đơn giản:

```
N41 G99 G82 X.. Y.. R0.1 Z-0.25 P300 F5.0
```

Trong doa miệng, nếu sử dụng tốc độ trục chính tương đối chậm và tốc độ cắt đủ cao, cần bảo đảm đủ thời gian tạm dừng (dwell) P trong chu kỳ G82. Nguyên tắc là lập trình giá trị gấp đôi hoặc cao hơn dwell tính toán tối thiểu. Dwell tối thiểu là:

$$D_m = \frac{60}{r/min}$$

Ví dụ, nếu tốc độ trục chính được lập trình là 600 r/min, dwell tối thiểu sẽ là = 0.1, và gấp đôi là 0.2 trong chương trình, do đó dwell sẽ là P200. Sự gấp đôi giá trị dwell sẽ bảo đảm ngay cả với 50% tốc độ trục chính vẫn có *ít nhất* một vòng quay trục chính để làm sạch đáy lỗ doa miệng. Nhiều nhà lập trình còn sử dụng thời gian dwell dài hơn, chẳng hạn hơn một hoặc hai vòng quay ở đáy lỗ.

Lả miệng

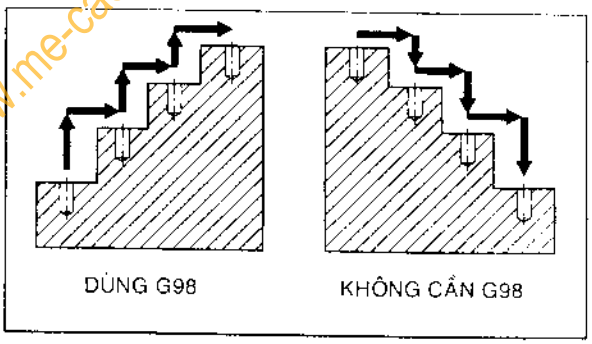
Lả miệng hầu như đồng nhất với doa miệng nhưng chiều sâu cắt rất nhỏ. Nói chung, lả miệng còn được gọi là *doa miệng nông*. Mục đích là cắt gọt đủ để cung cấp bề mặt phẳng cho đầu bulông, vòng đệm, hoặc đai ốc. Kỹ thuật lập trình hoàn toàn như doa miệng.

Trong nhiều trường hợp, cùng một dụng cụ cắt có thể được lập trình để dịch chuyển lên và xuống giữa các độ cao (các bậc hoặc mức trên chi tiết). Ví dụ, mũi khoan sẽ cắt các lỗ có cùng chiều sâu nhưng bắt đầu từ các bậc khác nhau.

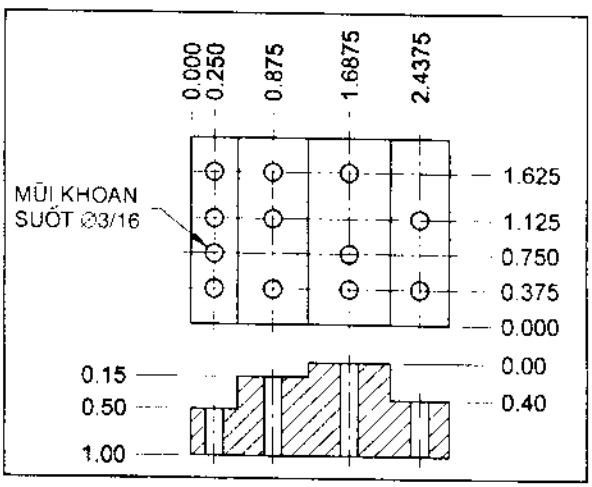
Loại lập trình này đòi hỏi điều kiện chính, dụng cụ cắt *cần được lập trình hiệu quả* (không tốn thất thời gian) và *phải được lập trình an toàn* (không va chạm).

Giải quyết vấn đề lập trình này không khó, chỉ cần đánh giá đúng các tùy chọn khả dụng. Các tùy chọn gồm hai lệnh chuẩn bị G98 và G99, được dùng với các chu kỳ cố định. G98 làm cho dụng cụ cắt trở về *mức ban đầu*, G99 đưa dao cắt về *mức R*. Trong lập trình thực tế, lệnh G98 chỉ được dùng trong các trường hợp khi cần tránh chướng ngại (vật cản) giữa các lỗ.

Hình 25.20 minh họa hai khả năng lập trình. Hình chiếu cạnh của chi tiết bậc nêu rõ chiều chuyển động dao giữa các lỗ. Hình bên trái, chuyển động từ một lỗ đến lỗ kế tiếp có thể gây ra va chạm với bậc, do đó để an toàn *cần dùng* G98. Hình bên phải, không có chướng ngại, *không cần G98*, có thể dùng G99.



Hình 25.20. Chiều chuyển động dao giữa các lỗ tại các độ cao khác nhau.



Hình 25.21. Khoan nhiều mức – ví dụ lập trình O2505

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Xác lập mức ban đầu thường được thực hiện trong block đầu tiên sử dụng chức năng G98 để vị trí an toàn phía trên *tất cả các* chướng ngại.

Ví dụ thực tiễn của kỹ thuật này được nêu trên Hình 25.21 và chương trình O2505.

Ở đây dùng hai dụng cụ-T01 là mũi khoan điểm 90°, cắt đến chiều sâu .108 dưới *từng mặt bậc*, T02 là mũi khoan *suốt* 3/16, được lập trình với chiều sâu tuyệt đối Z-1.106:

```
O2505 (MULTILEVEL EXAMPLE)
(T01 - 0.375 SPOT DRILL - 90 DEG)
N1 G20
N2 G17 G40 G80 T01
N3 M06
N4 G90 G54 G00 X0.25 Y0.375 S900 M03 T02
N5 G43 Z1.0 H01 M08
N6 G99 G82 R-0.4 Z-0.608 P200 F8.0
N7 Y0.75
N8 Y1.125
N9 G98 Y1.625
N10 G99 X0.875 R-0.05 Z-0.258
N11 Y1.125
N12 G98 Y0.375
N13 G99 X1.6875 R0.1 Z-0.108
N14 Y0.75
N15 Y1.625
N16 X2.4375 Y1.125 R-0.3 Z-0.508
N17 Y0.375
N18 G80 Z1.0 M09
N19 G28 Z1.0 M05
N20 M01

(T02 - 3/16 DRILL THRU)
N21 T02
N22 M06
N23 G90 G54 G00 X2.4375 Y0.375 S1000 M03
N24 G43 Z1.0 H02 M08
N25 G99 G83 R-0.3 Z-1.106 Q0.35 F10.0
N26 G98 Y1.125
N27 G99 X1.6875 Y1.625 R0.1
N28 Y0.75
N29 Y0.375
N30 X0.875 R-0.05
N31 Y1.125
N32 Y1.625
N33 X0.25 R-0.4
N34 Y1.125
N35 Y0.75
N36 Y0.375
N37 G80 Z1.0 M09
N38 G28 Z1.0 M05
N39 G00 X-2.0 Y10.0
N40 M30
%
```

Bạn hãy nghiên cứu chương trình này một cách chi tiết. Bạn cần chú ý chiều của các dao-T01 bắt đầu ở lỗ dưới bên trái và kết thúc ở lỗ dưới bên phải theo đường chữ chi (gấp khúc). T02 bắt đầu ở lỗ dưới bên phải và kết thúc tại lỗ dưới bên trái cũng theo đường gấp khúc. Đối với dụng cụ thứ nhất, số lần thay đổi G98 hoặc G99 lớn hơn dụng cụ thứ hai. Trong gia công lỗ nhiều mức bạn cần hiểu ba lĩnh vực điều khiển chương trình, được dùng trong O2505:

- Điều khiển G98 và G99
- Điều khiển mức R
- Điều khiển chiều sâu Z

KHOAN QUA NHIỀU BẬC

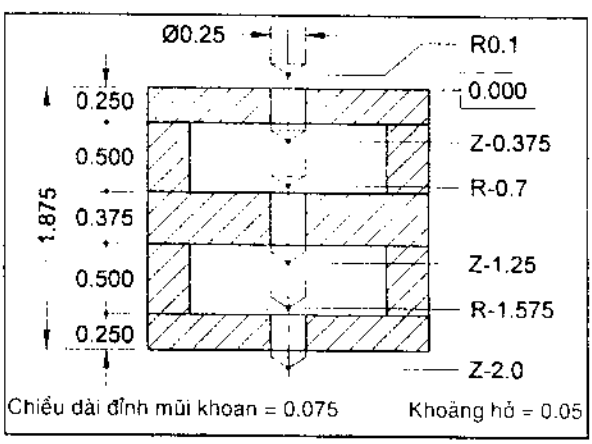
Khoan qua nhiều bậc là nguyên công khoan giữa hai hoặc nhiều chi tiết, cách nhau bằng khoảng không gian trống. Vấn đề lập trình là gia công các lỗ này *một cách hiệu quả*. Có thể dễ dàng lập trình một chuyển động xuyên qua tất cả các chi tiết riêng rẽ và các khoảng trống. Đối với nhiều lỗ, phương pháp này có lẽ không hiệu quả. Bạn hãy khảo sát bản vẽ trên Hình 25.22.

Trong chương trình này, X1.0 Y1.5 là vị trí lỗ. Bản vẽ không nêu rõ mức R và chiều sâu Z, chúng cần được tính toán. Trong ví dụ này, khoảng hở trên và dưới từng tấm chi tiết là.05, ngoại trừ mức R thứ nhất (R0.1). Chiều dài đỉnh mũi khoan Ø1/4 là 0.3 × 0.25 = 0.075 inch.

```
O2506 (WEB DRILLING)
(T01 - 90-DEG SPOT DRILL - 0.5 DIA)
N1 G20
N2 G17 G40 G80 T01
N3 M06
N4 G90 G54 G00 X1.0 Y1.5 S900 M03 T02
N5 G43 Z1.0 H01 M08
N6 G99 G82 R0.1 Z-0.14 P250 F7.0
N7 G80 Z1.0 M09
N8 G28 Z1.0 M05
N9 M01

(T02 - 1/4 DIA DRILL)
N10 T02
N11 M06
N12 G90 G54 G00 X1.0 Y1.5 S1100 M03 T01
N13 G43 Z1.0 H02 M08
N14 G99 G81 R0.1 Z-0.375 F6.0      (TOP PLATE)
N15 R-0.7 Z-1.25                (MIDDLE PLATE)
N16 G98 R-1.575 Z-2.0           (BOTTOM PLATE)
N17 G80 Z1.0 M09
N18 G28 Z1.0 M05
N19 M30
%
```

Bạn hãy lưu ý, một lỗ yêu cầu đến ba block chương trình, thay vì một block. Mỗi block biểu



Hình 25.22. *Ví dụ về khoan qua nhiều bậc – chương trình O2506.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

thị cho một tấm trong chi tiết gia công. Ngoài ra, bạn hãy để ý G98 trong block N16. Chỉ một lỗ được thực hiện trong ví dụ này, do đó G98 là không cần thiết. Lệnh xóa chu kỳ G80 với chuyển động trở về trong block N17 sẽ lùi dao ra khỏi lỗ. Tuy nhiên, nếu gia công nhiều lỗ, cần dịch chuyển dao đến vị trí XY mới *trước* khi lập trình G80. Trong trường hợp này, cần có G98 khi mũi khoan xuyên qua tấm cuối cùng của chi tiết. Ví dụ này chưa phải là giải pháp tối ưu đối với quy trình khoan qua nhiều bậc, do vẫn có chuyển động vô ích. Phương pháp lập trình hiệu quả là sử dụng kỹ thuật lập trình macro.

TARÔ REN

Tarô ren là sự gia công thông dụng thứ hai sau nguyên công khoan lỗ trên trung tâm gia công CNC. Tarô trên máy phay CNC hoặc trung tâm gia công thường sử dụng *hai* chu kỳ cố định khả dụng để lập trình trên hầu hết các hệ điều khiển. Chúng là chu kỳ G84 để tarô ren thuận (R/H) và chu kỳ G74 để tarô ren ngược (L/H):

G84	Tarô thuận – *ren theo chiều tay phải* với chuyển động trục chính M03
G74	Tarô ngược – *ren theo chiều tay trái* với chuyển động trục chính M04

Ví dụ dưới đây minh họa sự lập trình tarô một lỗ, tương tự các chu kỳ cố định khác. Mọi chuyển động dụng cụ cắt, kể cả dừng trục chính và đảo chiều ở đáy lỗ, đều trong chu kỳ cố định.

```
...
N64 G90 G54 G00 X3.5 Y7.125 S600 M03 T06
N65 G43 Z1.0 H05 M08
N66 G99 G84 R0.4 Z-0.84 F30.0
N67 G80 ...
```

Kích cỡ mũi tarô cần dùng là bao nhiêu? Trong ví dụ này, cỡ tarô là tiêu chuẩn, 20TPI (20 ren/inch), và phần cán. Không cần các tọa độ XY trong chu kỳ G84 do vị trí dao hiện hành đã được thiết lập trong block N64. Mức R thông dụng là vị trí bắt đầu ren và chiều sâu Z là chiều sâu tuyệt đối của phần ren. Địa chỉ cuối cùng trong block này là tốc độ cắt tính theo in/min, được lập trình với địa chỉ F.

Bạn hãy lưu ý mức R 0.4 có giá trị *hơi cao* hơn giá trị thường dùng để khoan, chuốt, và các nguyên công tương tự. Ngoài ra, tốc độ cắt được lập trình *có vẻ* quá cao. Thực tế cả hai đều *đúng* và được chọn *một cách cố ý*.

Khoảng hở lớn đối với mức R cho phép *tăng* tốc độ cắt từ 0 đến 30 in/min ở bên ngoài lỗ. Khi mũi tarô tiếp xúc với chi tiết, tốc độ cắt phải đạt giá trị lập trình, *không được thấp hơn*. Nguyên tắc là lập trình khoảng hở để tarô từ *hai đến bốn lần* khoảng hở bình thường. Khoảng hở này sẽ bảo đảm tốc độ cắt đạt hiệu quả hoàn toàn khi thực sự bắt đầu tarô ren. Bạn có thể thử dùng với giá trị hơi nhỏ để tăng hiệu quả cho chương trình. Phương pháp thứ hai là *gấp đôi, gấp ba, hoặc gấp bốn bước của mũi tarô* và dùng giá trị này làm khoảng hở phía trên mũi tarô. Mục đích của khoảng hở này luôn luôn nhằm loại bỏ vấn đề tốc độ cắt liên quan đến sự tăng tốc độ chuyển động.

Vấn đề kế tiếp là tốc độ cắt tương đối cao 30 in/min (F30.0) cũng đã được tính toán cẩn thận. Tốc độ bất kỳ khi tarô đều *phải* đồng bộ với *tốc độ trục chính* – r/min được lập trình theo địa chỉ S. Bạn cần nhớ mũi tarô về cơ bản là dụng cụ *tạo hình* với hình dạng và cỡ ren xác định. Phần kế tiếp sẽ giải thích chi tiết quan hệ giữa tốc độ trục chính và tốc độ cắt. Tốc độ cắt F trong chương trình được tính bằng cách nhân *số đầu mối* ren với *tốc độ* trục chính theo r/min:

F = 1/20 TPI x 600 r/min = 30.0 in/min

Cách tính toán thứ hai là chia *tốc độ trục chính* (r/min) cho *số ren/inch* (TPI):

F = 600 r/m/20 TPI = 30.0 in/min

Chất lượng lỗ được tarô cũng rất quan trọng, ngoài việc lựa chọn tốc độ trục chính và tốc độ cắt, còn có các yếu tố khác. Vật liệu của mũi tarô, lớp tráng phủ, dạng hình học, khoảng hở rãnh thoát phoi, đường cong xoắn, kiểu vạt góc ban đầu, vật liệu được cắt gọt, cán tarô, tất cả đều ảnh hưởng đến chất lượng lỗ ren. Để có kết quả tarô ren tối ưu, cần có cán tarô thích hợp trừ khi máy CNC hỗ trợ *sự tarô chặt*. Thiết kế cán tarô lỏng giúp cho mũi tarô "cảm nhận", tương tự cảm giác của bạn khi tarô bằng tay. Cán tarô lỏng thường được gọi là cán *kéo-nén* và ứng dụng như nhau cho cả trên máy phay và tiện. Kiểu cán này cho phép mũi tarô kéo ra hoặc đẩy vào trong khoảng xác định. Sự khác biệt duy nhất là phương pháp gá lắp dụng cụ cắt (định hướng của tarô) trong máy (đứng hoặc ngang). Các cán tarô lỏng cao cấp còn có *moment xoắn* điều chỉnh được, có thể thay đổi cảm giác của mũi tarô thậm chí cả khoảng kéo và nén.

Các ứng dụng tarô ren trên máy tiện CNC tương tự như trên trung tâm gia công. Không cần *chu kỳ tarô* ren đặc biệt trên bộ điều khiển máy tiện, nhưng mỗi chi tiết chỉ có thể sử dụng một kích cỡ mũi tarô. Từng chuyển động tarô được lập trình với lệnh G32 và phương pháp block – block.

Tarô ren trên máy tiện CNC tuy hơi khác

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

nhưng không khó hơn tarô trên trung tâm gia công. Dù không sử dụng chu kỳ cố định, nhưng lập trình có thể phạm vài sai sót. Chương này sử dụng các ví dụ về tarô trên máy tiện CNC đến chiều sâu thích hợp.

Dạng hình học của mũi tarô

Hiện có nhiều kiểu thiết kế mũi tarô được dùng trong các ứng dụng lập trình CNC khác nhau. Đối với lập trình CNC, chỉ *các cơ sở cốt lõi* của dạng hình học mũi tarô là quan trọng.

Có hai vấn đề trong thiết kế mũi tarô ảnh hưởng trực tiếp đến lập trình và các giá trị nhập:

- Dạng hình học rãnh thoát phoi của mũi tarô.
- Dạng hình học vạt góc đầu mũi tarô.

Dạng hình học rãnh thoát phoi

Dạng hình học *rãnh thoát phoi* của mũi tarô thường dùng các thuật ngữ "xoắn thấp", "xoắn cao", hoặc "rãnh xoắn".Về cơ bản, các thuật ngữ đó xác định *cách thức* mài các lưỡi cắt ren trên thân tarô. Khi lập trình nguyên công tarô ren, tính hiệu quả của dạng hình học rãnh xoắn quan hệ chặt chẽ với tốc độ trục chính. Sự thực nghiệm với tốc độ cắt bị giới hạn theo bước ren, nhưng sự lựa chọn tốc độ trục chính tương đối rộng. Vật liệu của chi tiết *và* dạng hình học rãnh xoắn của mũi tarô *đều* tác động đến tốc độ trục chính của máy. Do hầu như mọi thiết kế dụng cụ cắt (không riêng cho mũi tarô) đều là kết quả của nhiều yếu tố, không thể nói một chiều "hãy dùng dụng cụ này" hoặc "dụng cụ kia" cho lập trình CNC. Bạn cần nhớ mọi mũi tarô đều có chung một số đặc tính.

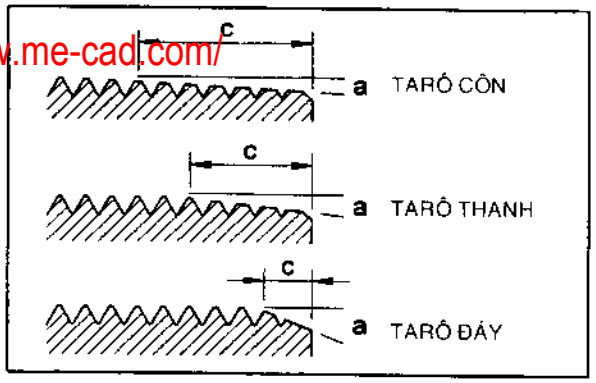
Dạng hình học vạt góc

Dạng hình học vạt góc mũi tarô liên quan đến cấu hình ở đầu mũi tarô. Đối với lập trình CNC, phần quan trọng nhất của dạng hình học ở đầu mũi tarô là *vạt góc đỉnh tarô*.

Để lập trình gia công lỗ một cách chính xác, mũi ta rô phải được chọn theo các yêu cầu kỹ thuật của lỗ được tarô ren. Nếu tarô lỗ cụt cần sử dụng mũi tarô khác với tarô lỗ suốt. Có ba kiểu mũi tarô, phân loại theo dạng hình học ở đầu mũi tarô:

- Ta rô đáy
- Ta rô thanh
- Ta rô côn

Khác biệt cơ bản giữa các mũi tarô là *chiều dài* phần vạt góc. Hình 25.23 minh họa các đặc tính của lỗ khoan ảnh hưởng đến chiều sâu lập trình của mũi tarô được chọn.



Hình 25.23. *Dạng hình học ở đầu mũi tarô.*

Chiều dài c của phần côn (vạt góc) mũi tarô được đo theo số ren. Số ren của mũi tarô côn là 8-10, tarô thanh là 3-5, và tarô đáy là 1-1.5. Góc vát a cũng tùy theo kiểu tarô, 3-5^0 đối với mũi tarô côn, 8-13^0 trên tarô thanh, và 25-35^0 trên tarô đáy.

Lỗ cụt hầu như luôn luôn yêu cầu mũi tarô đáy, lỗ suốt sẽ đòi hỏi mũi tarô thanh trong hầu hết các trường hợp, còn tarô côn chỉ dùng trong một số trường hợp. Chiều dài c càng lớn, yêu cầu khoảng hở đáy của lỗ khoan càng lớn.

Tốc độ mũi tarô và tốc độ cắt

Quan hệ giữa tốc độ trục chính của máy (r/min) và tốc độ cắt lập trình là cực kỳ quan trọng khi lập trình chuyển động cắt theo chế độ *lượng ăn dao/thời gian*. Chế độ *lượng ăn dao/thời gian* được lập trình theo đơn vị Anh là *in/min* và hệ mét là *mm/min*. Chế độ này là điển hình đối với máy phay CNC và trung tâm gia công, hầu như mọi chế độ gia công đều sử dụng in/min hoặc mm/min. Đối với tarô ren, bạn hãy luôn luôn lập trình theo *khoảng cách tuyến tính* mũi tarô phải hành trình trong *một vòng quay trục chính*. Khoảng cách này luôn luôn tương đương với *bước* tarô (chỉ khi tarô ren), do mũi tarô thường chỉ dùng để cắt ren một đầu mối.

Khi sử dụng chế độ *ăn dao theo vòng quay*, chế độ điển hình trên máy tiện CNC, bước tarô luôn luôn tương đương với tốc độ cắt. Ví dụ, bước .050 yêu cầu tốc độ cắt .050 in/rev hoặc F 0.05 trong chương trình.

Trên trung tâm gia công CNC, chế độ cắt luôn luôn là lượng ăn dao/*thời gian*, tính theo *phút*, tốc độ cắt này được xác định bằng một trong các công thức dưới đây:

$$F_t = \frac{r/min}{TPI}$$

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trong đó:

F_i = Tốc độ ăn dao theo thời gian (phút)
r/min = Tốc độ trục chính
TPI = Số ren/inch

Công thức tương tự cũng đưa đến kết quả đồng nhất:

$$F_i = r/min \cdot F_r$$

F_i = Tốc độ ăn dao theo thời gian (phút)
r/min = Tốc độ trục chính
F_r = Tốc độ cắt/ vòng quay

Ví dụ, bước ren 20TPI:

1/20 = .0500 inch

và tốc độ cắt lập trình được xét theo tốc độ trục chính, ví dụ 450 r/min:

F = 450 x .05 = 22.5 = F22.5 (in/min)

Tarô ren hệ mét trên máy tiện cũng tương tự. Ví dụ, tarô bước ren 1.5mm sử dụng 500 r/min được lập trình với tốc độ cắt 750 mm/min

F = 500 x 1.5 = 750.00 = F750.0 (mm/min)

Yếu tố chính để tarô ren thành công là duy trì quan hệ giữa *bước tarô* và *tốc độ trục chính*. Nếu tốc độ trục chính thay đổi, lượng ăn dao/ thời gian (in/min hoặc mm/min) cũng phải thay đổi theo. Đối với nhiều cán tarô kiểu kéo- nén, sự điều chỉnh *giảm* tốc độ cắt khoảng 5% có thể đưa đến kết quả tốt hơn. Điều này là do sự kéo của cán tarô linh hoạt hơn so với lực nén.

Nếu tốc độ trục chính trong ví dụ nêu trên thay đổi từ S450 sang S550 (bước tarô *không thay đổi*, 20TPI), sự thay đổi tốc độ trục chính phải được phản ánh trong tốc độ cắt *mới*:

F = 550 x .05 = 27.50 = F 27.5 (in/min)

Trong chương trình, tốc độ cắt ren mới sẽ là:

F = 27.5 – 5%(27.5) = 26.125

Giá trị vận tốc cắt thực có thể là F26.1 hoặc thậm chí F26.0. Có thể dễ dàng thay đổi tốc độ trục chính của dụng cụ cắt trong chương trình, hoặc trực tiếp trên máy CNC, nhưng sau đó lại quên chỉnh sửa tốc độ cắt cho mũi tarô. Sai lầm này có thể xảy ra trong khi chuẩn bị chương trình ở văn phòng hoặc tối ưu hóa chương trình trên máy. Nếu thay đổi là nhỏ, có thể không xảy ra hư hại. Nếu tốc độ trục chính thay đổi quá lớn, mũi tarô có thể bị gãy trong chi tiết.

Tarô ren ống

Các tarô ren ống có cấu trúc tương tự mũi tarô tiêu chuẩn. Chúng thuộc hai nhóm:

- ❑ Tarô côn — NPT và API
- ❑ Taro thẳng (song song) — NPS

Kích cỡ danh định của chúng, *không* phải là kích cỡ tarô mà là kích cỡ của phần nối ống. Độ côn ống (NPT) theo *Tiêu chuẩn Quốc gia Mỹ* có tỷ số côn 1/16, nghĩa là 3/4 inch/foot (1.78991061° mỗi phía) và vát mũi tarô là 2 đến 3½ ren

Lập trình tarô ống cũng tuân theo các nguyên tắc như tarô tiêu chuẩn. Khó khăn duy nhất là tính toán vị trí chiều sâu Z một cách hợp lý dù có thể không hoàn toàn chính xác. Chiều sâu cuối cùng có thể phải qua thí nghiệm với cán tarô và vật liệu cụ thể.

Cỡ mũi khoan taro thích hợp là rất quan trọng. Luôn luôn có sự khác biệt giữa lỗ khoan sau đó tarô (dùng cho mũi chuốt côn ¾ inch/foot).

Bảng 25.3 liệt kê kích cỡ ren ống côn cho nhóm NPT, mũi khoan, dữ liệu này rất hữu ích cho lập trình CNC. Bảng 25.4 là cỡ ren ống thẳng (NPS) và mũi khoan tương ứng.

Bảng 25.3. *Cỡ ren ống côn*

Nhóm NPT		Chỉ khoan		Chuốt	
Cỡ ống	TPI	Mũi khoan	Cỡ thập phân	Mũi khoan	Cỡ thập phân
1/16	27	D	.2460	15/64	.2344
1/8	27	Q	.3320	21/64	.3281
1/4	18	7/16	.4375	27/64	.4219
3/8	18	37/64	.5781	9/16	.5625
1/2	14	45/64	.7031	11/16	.6875
3/4	14	29/32	.9062	57/64	.8906
1.0	11-1/2	1-9/64	1.1406	1-1/8	1.1250
1-1/4	11-1/2	1-31/64	1.4844	1-15/32	1.4688
1-1/2	11-1/2	1-47/64	1.7344	1-23/32	1.7188
2.0	11-1/2	2-13/64	2.2031	2-3/16	2.1875

Bảng 25.4. *Cỡ ren ống thẳng*

Cỡ ống	TPI	Mũi khoan	Cỡ thập phân
1/16	27	1/4	.2500
1/8	27	11/32	.3438
1/4	18	7/16	.4375
3/8	18	37/64	.5781
1/2	14	23/32	.7188
3/4	14	59/64	.9219
1.0	11-1/2	1-5/32	1.1563
1-1/4	11-1/2	1-1/2	1.5000
1-1/2	11-1/2	1-3/4	1.7500
2.0	11-1/2	2-7/32	2.2188

Tốc độ cắt khi tarô ren ống có cùng các quan hệ như tarô tiêu chuẩn

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Danh mục kiểm tra tarô ren

Khi lập trình nguyên công tarô ren, cần bảo đảm dữ liệu chương trình phản ánh các điều kiện gia công thực tế. Chúng có thể khác nhau giữa các chế độ gá lắp, nhưng đa số đều như nhau đối với nguyên công tarô ren *bất kỳ* trên kiểu máy CNC *bất kỳ*. Dưới đây là danh mục tóm tắt các yếu tố có quan hệ *trực tiếp* với các nguyên công tarô ren trong lập trình CNC.

- ❑ Mép cắt tarô (sắc và được mài chuẩn xác).
- ❑ Thiết kế mũi tarô (phù hợp với lỗ được tarô ren).
- ❑ Sự thẳng hàng mũi tarô (thẳng hàng với lỗ được tarô).
- ❑ Tốc độ trục chính lắp tarô (phù hợp với các điều kiện cắt gọt).
- ❑ Tốc độ tarô ren (theo quan hệ với bước ren tarô và tốc độ trục chính của máy).
- ❑ Gá lắp chi tiết (bảo đảm độ cứng vững cao).
- ❑ Lỗ khoan phải được gia công trước một cách chính xác (cỡ mũi khoan lỗ để tarô ren là rất quan trọng).
- ❑ Khoảng hở từ vị trí bắt đầu tarô (đủ để tăng tốc độ quay mũi tarô).
- ❑ Lựa chọn chất làm nguội.
- ❑ Khoảng hở dưới đáy lỗ (phải bảo đảm chiều sâu phần ren).
- ❑ Điều chỉnh moment quay của cán tarô (để cắt gọt).
- ❑ Tính toàn vẹn của chương trình (không có lỗi).

Nhiều thiết kế cán tarô có các yêu cầu riêng, có thể có hoặc không ảnh hưởng đến phương pháp lập trình.

Với các máy CNC hiện đại, phương pháp *tarô chặt* ngày càng phổ biến. Hầu như không cần cán tarô đặc biệt, chẳng hạn cán dao phay mặt đầu kiểu *kéo-nén* hoặc mâm cặp đặc biệt, do đó tiết kiệm chi phí cán tarô. Tuy nhiên, máy CNC và hệ điều khiển phải hỗ trợ tính năng tarô chặt. Để lập trình tarô chặt, cần sử dụng mã M đặc biệt, ghi rõ trong tài liệu kèm theo máy.

Chế độ ta rô chặt phải được máy CNC hỗ trợ để sử dụng trong chương trình.

GIA CÔNG LỖ TRÊN MÁY TIỆN

Gia công lỗ với dao một lưỡi cắt trên máy tiện CNC nói chung không thông dụng bằng trên trung tâm gia công CNC. Thứ nhất, số lỗ có thể khoan hoặc tarô trong một nguyên công trên máy tiện chỉ là một lỗ/chi tiết, còn số lỗ trên máy phay, có thể hàng chục, trăm, hoặc hàng ngàn. Thứ hai, doa (tiện trong) trên máy tiện là gia công *biên dạng*, khác với doa, trên máy phay, là gia công điểm – điểm.

Mọi nguyên công gia công điểm – điểm trên máy tiện CNC chỉ giới hạn theo các nguyên công có thể được thực hiện với dụng cụ cắt định vị *ở đường tâm* trục chính. Nói chung, các nguyên công này bao gồm khoan tâm, khoan tiêu chuẩn, chuốt và tarô ren. Nhiều kiểu dụng cụ cắt có thể được sử dụng, ví dụ dao phay mặt đầu cắt tâm có thể được dùng để gia công lỗ chính xác. Ở mức độ thấp hơn, các nguyên công khác, chẳng hạn dao là miệng, khoét miệng, có thể được dùng để gia công lỗ hoặc lỗ cụt đáy phẳng. Dụng cụ đánh bóng mặt trong có thể được dùng để gia công lỗ chính xác. Ở mức độ thấp hơn, các nguyên công khac, chẳng hạn khoét lỗ, có thể lập trình theo đường tâm máy tiện, với dụng cụ điểm – điểm, *thay vì dụng cụ gia công biên dạng*. Mọi nguyên công thuộc nhóm này có một điểm chung - chúng đều được dùng ở đường tâm trục chính và được lập trình với vị trí X là X0 trong block chương trình.

Tốc độ trục chính của *mọi* nguyên công theo đường tâm trên máy tiện CNC phải được lập trình theo số vòng quay thực/phút (r/min), *không* theo chế độ tốc độ bề mặt không đổi (CSS). Vì lý do đó, cần dùng lệnh G97, ví dụ:

```
G97 S575 M03
```

sẽ bảo đảm tốc độ 575 r/min với chiều quay trục chính bình thường.

Điều gì sẽ xảy ra nếu chế độ CSS được dùng với lệnh G96 thay vì G97? Hệ thống CNC sẽ sử dụng thông tin đã cho, địa chỉ tốc độ trục chính S, trong chương trình (được cho theo chu vi hoặc bề mặt – tốc độ/phút, ví dụ ft/min). Hệ thống sẽ tính toán tốc độ trục chính yêu cầu theo r/min để máy dụng cụ có thể sử dụng.

Sự tính toán này dựa trên công thức toán học tiêu chuẩn liên hệ với *đường kính* của chi tiết. Nếu đường kính này là zero – *chính xác* là đường tâm trục chính - số vòng quay trục chính sẽ luôn luôn là *r/min cao nhất* khả dụng trong khoảng tốc độ trục chính được chọn hiện hành. Tính toán này là ngoại lệ đối với công thức tính toán r/min tiêu chuẩn, trong đó tốc độ trục chính tại đường tâm (đường kính zero) là zero – 0 r/min

Ví dụ, nếu tốc độ chu vi (bề mặt) đối với vật liệu đã cho là 450 ft/min, giá trị r/min tại ∅3 inch (X 3.0) sẽ xấp xỉ:

$$S = (450 \times 3.82)\ 13 = 573 \text{ r/min}$$

Nếu tốc độ 450 ft/min được áp dụng cho đường kính zero (X0 trong chương trình), công thức không thay đổi nhưng kết quả sẽ là:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

S = (450 x 3.82)/0 = 0 r/min (lỗi)

Tốc độ trục chính không thể là *vô hạn* (∞), do thiết kế của hệ điều khiển, tốc độ này sẽ đạt giá trị *r/min* cực đại trong khoảng tốc độ hiện hành. Ở đây, bạn cần rất cẩn thận, bảo đảm mọi nguyên công theo đường tâm trên máy tiện CNC luôn luôn được thực hiện trong chế độ G97 (r/min) mà *không* trong chế độ G96 (CSS).

Chuyển động tiếp cận của dụng cụ cắt

Xác lập cấu hình bù dạng hình học (các giá trị G50) trên máy tiện CNC thường có giá trị X tương đối lớn và Z tương đối nhỏ. Ví dụ, bù hình học đối với dao có thể là X-11.8 Z-1.0 (hoặc G50 X11.8 Z1.0). Vị trí này biểu thị vị trí thay dao thích hợp có thể áp dụng cho mũi khoan. Điều đó có nghĩa là gì đối với chuyển động dụng cụ cắt trong nguyên công khoan?

Điều này có nghĩa là chuyển động nhanh sẽ hoàn tất chuyển động trục Z trước khi hoàn tất chuyển động trục X. Kết quả là chuyển động dao *rất gần* bề mặt chi tiết:

```
N36 T0200 M42
N37 G97 S700 M03
N38 G00 X0 Z0.1 T0202 M08
N39 ....
```

Để tránh khả năng xảy ra va chạm trong khi dao tiến đến chi tiết, bạn hãy dùng một trong các phương pháp dưới đây:

- Dịch chuyển trục X đến đường tâm trước, tiếp theo là trục Z, trục tiếp đến vị trí bắt đầu khoan.
- Dịch chuyển trục Z trước đến khoảng trống, trục X đến đường tâm, sau đó hoàn tất chuyển động trục X đến vị trí bắt đầu khoan.

Phương pháp thứ nhất có thể chỉ có tính thực tiễn trong những trường hợp khi khu vực chuyển động dao hoàn toàn trống và không có các chướng ngại trên đường đi. Phương pháp thứ hai, có lẽ thông dụng nhất trong lập trình, sẽ dịch chuyển trước trục Z đến gần (nhưng không quá gần) chi tiết, chẳng hạn .50 inch phía trước (Z0.5). Chuyển động kế tiếp là chuyển động trục X – trực tiếp đến đường tâm (X0). Tại điểm này, dụng cụ cắt (chẳng hạn mũi khoan) ở cách xa mặt trục Z. Chuyển động tiếp cận cuối cùng và là vị trí bắt đầu trục Z; gần mặt chi tiết hơn, nơi nguyên công khoan bắt đầu. Phương pháp này loại bỏ (hoặc ít nhất là giảm) khả năng xảy ra va chạm với các chướng ngại (nếu có) trên đường đi. Các chướng ngại có thể là ụ động, đồ gá, tấm mặt... Ví dụ về phương pháp lập trình quỹ đạo dao này là ví dụ đã nêu ở phần trên với vài chỉnh sửa:

```
N36 T0200 M42
N37 G97 S700 M03
N38 G00 X0 Z0.5 T0202 M08
N39 Z0.1
N40 ...
```

Phương pháp lập trình này chia sự tiếp cận dao dọc theo trục Z thành *hai* vị trí dao – một là ở khoảng trống an toàn để tiếp cận, thứ hai là vị trí an toàn để bắt đầu khoan. Có thể có thay đổi nhỏ đối với chuyển động này-sự tiếp cận trục Z cuối cùng sẽ theo tốc độ cắt thay vì tốc độ chuyển động nhanh:

```
N36 T0200 M42
N37 G97 S700 M03
N38 G00 X0 Z0.5 T0202 M08
N39 G01 Z0.1 F0.05
N40 ...
```

Đối với chuyển động tiếp cận cuối cùng, chuyển động trục Z được đổi thành chuyển động tuyến tính với tốc độ cắt tương đối cao .050 in/rev (1.25 mm/rev). Sự vượt qua tốc độ cắt có thể được sử dụng để gá lắp, để điều khiển tốc độ ăn dao. Trong sản xuất thực tiễn, hầu như không bị tổn thất về thời gian chu kỳ.

Chuyển động dao trở về

Các nguyên tắc logic của chuyển động trong không gian áp dụng cho sự tiếp cận dao, cũng đúng cho chuyển động dao trở về. Bạn cần nhớ, chuyển động thứ nhất *rời khỏi* lỗ phải luôn luôn theo trục Z:

```
...
N40 G01 Z-0.8563 F0.007
N41 G00 Z0.1
...
```

Trong block N40, xảy ra chuyển động cắt thực của mũi khoan. Khi cắt gọt hoàn tất, block N41 được thực thi. Mũi khoan lùi nhanh ra khỏi lỗ đến vị trí khởi động ban đầu (Z0.1). Tuy không nhất thiết phải lùi mũi khoan về đúng vị trí đó, nhưng sự lập trình sẽ ổn định hơn.

Sau khi dụng cụ cắt ra khỏi lỗ một cách an toàn, cần trở về vị trí thay dao. Có hai phương pháp:

- Chuyển động đồng thời cả hai trục.
- Lần lượt chuyển động theo từng trục.

Chuyển động đồng thời cả hai trục X và Y không gây ra vấn đề như khi dao tiếp cận chi tiết gia công. Trục Z sẽ hoàn tất chuyển động trước, ra xa bề mặt chi tiết. Ngoài ra cũng không xảy ra va chạm khi chuyển động trở về nếu dao đã đến gần chi tiết một cách thành công và sự lập trình là ổn định.

```
...
N70 G01 Z-0.8563 F0.007
N71 G00 Z0.1
N72 X11.0 Z2.0 T0200 M09
...
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nếu có gì nghi ngờ, hoặc có chướng ngại trên đường chuyển động của dao, ví dụ ụ động, bạn hãy lập trình chuyển động lần lượt theo từng trục. Trong hầu hết các trường hợp, sẽ chuyển động trước theo chiều dương trục X. Do các chướng ngại thường ở bên phải chi tiết gia công:

```
...
N70 G01 Z-0.8563 F0.007
N71 G00 Z0.1
N72 X12.0
N73 Z2.0 T0200 M09
...
```

Ví dụ lập trình này minh họa chuyển động trở về với trục X được lập trình trước. Dao cách mặt trước của chi tiết 0.100 inch là thích hợp, dao bắt đầu cắt từ khoảng cách đó sẽ không gây ra vấn đề.

Khoan và chuốt trên máy tiện

Khoan trên máy tiện CNC là nguyên công phổ biến, chủ yếu để tạo lỗ sẵn cho các nguyên công kế tiếp, chẳng hạn doa. Có ba loại khoan cơ bản được thực hiện trên máy tiện CNC.

- Khoan tâm và khoét miệng.
- Khoan với mũi khoan xoắn.
- Khoan với mũi khoan chắp(gắn Carbides).

Các phương pháp này đều sử dụng những kỹ thuật lập trình tương ứng đã đề cập trong phần phay, nhưng không có chu kỳ cố định phay nào được dùng cho máy tiện. Bạn cần nhớ, trên máy tiện, chi tiết quay do đó dụng cụ cắt là tĩnh tại. Hầu hết các nguyên công tiện đều theo chiều ngang, do đó cần chú ý các vấn đề loại bỏ phoi và chiều phun chất làm nguội.

Chu kỳ khoan gián đoạn - G74

Trên các bộ điều khiển Fanuc, có chu kỳ lặp lại nhiều lần G74, được dùng cho hai nguyên công:

- Gia công thô với sự bẻ gãy phoi.
- Khoan từng đoạn (khoan lỗ sâu).

Phần này sẽ trình bày khoan lỗ sâu theo nhiều đoạn sử dụng chu kỳ G74. Gia công thô với G74 là gia công biên dạng, rất ít sử dụng trong thực tế.

Trong khoan gián đoạn, tương tự các nguyên công khoan khác, trước hết là chọn tốc độ trục chính và tốc độ cắt, sau đó chọn vị trí bắt đầu lỗ, và cuối cùng là vị trí chiều sâu. Ngoài ra, bạn cần thiết lập (hoặc tính toán) chiều sâu của từng đoạn khoan. Chu kỳ tiện G74 chỉ giới hạn cho vài ứng dụng. Định dạng khoan gián đoạn là:

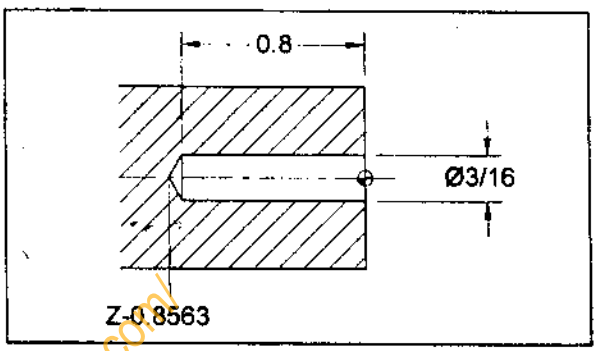
```
G74 X0 Z.. K..
```

Trong đó:

G74	= chọn chu kỳ khoan gián đoạn
X0	= cắt theo đường tâm.
Z	= chuyên biệt điểm kết thúc khoan
K	= chiều sâu từng đoạn khoan (luôn luôn dương)

Chương trình dưới đây sử dụng minh họa trên Hình 25.24, khoan lỗ Ø3/16 (Ø.1875) với chiều sâu từng đoạn khoan là .300 inch:

```
...
N85 T0400 M42
N86 G97 S1200 M03
N87 G00 X0 Z0.2 T0404 M08
N88 G74 X0 Z-0.8563 K0.3 F0.007
N89 G00 X12.0 Z2.0 T0400 M09
N90 M01
```



Hình 25.24. Lỗ khoan gián đoạn trên máy tiện

Chuyển động khoan theo đoạn sẽ bắt đầu từ vị trí Z 0.2 trong block N87 và tiếp tục đến vị trí Z-0.8563 trong block N88. Kết quả là đường cắt dài 1.0563 inch. Tính toán số đoạn tương tự như trên máy phay.

Với chiều dài .300 của từng đoạn, sẽ có đoạn chiều dài *toàn phần* và một đoạn ngắn, tại các vị trí trục Z.

```
Z-0.1
Z-0.4
Z-0.7
Z-0.8563
```

Mặc dầu ba đoạn đầu đều sâu .300, nhưng đoạn thứ nhất bắt đầu ở Z 0.2 và kết thúc ở Z-0.1, nghĩa là 2/3 đoạn này không thực sự cắt gọt. Nhà lập trình có thể chọn cách này hoặc chọn phương pháp khác thích hợp hơn. Ở cuối đoạn chuyển động sử dụng chu kỳ G74, mũi khoan hơi lùi lại với khoảng cách *cố định*. Khoảng cách này được xác định bằng tham số của hệ điều khiển, thường là .020 inch (0.5 mm). Chu kỳ G74 không hỗ trợ sự lùi ra khỏi lỗ sau mỗi đoạn khoan (tương tự chu kỳ G83 trên máy phay).

Bạn hãy lưu ý ở đây không lập trình chuyển động *ra khỏi lỗ* khi hoàn tất chu kỳ khoan gián đoạn. Sự chuyển động này được lập

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

trình sẵn trong chu kỳ G74. Nếu có chuyển động dao chẳng hạn G00Z0.2M05 sau block N88, lùi ra khỏi lỗ, người vận hành sẽ tin tưởng hơn khi chạy chương trình.

Tarô ren trên máy tiện

Tarô ren trên máy tiện CNC là nguyên công phổ biến, tuân theo các nguyên tắc như trên trung tâm gia công. Khác biệt cơ bản đối với máy tiện là không có chu kỳ tarô ren. Thực tế không cần chu kỳ tarô ren trên máy tiện, do hầu hết các nguyên công này chỉ gia công *một lỗ cùng loại*. Có thể xảy ra một số khó khăn khi không có chu kỳ tarô ren, đặc biệt là với các nhà lập trình ít kinh nghiệm. Trước khi đánh giá các khó khăn đó, bạn cần biết dụng cụ giữ mũi tarô (*cán tarô*) và quy trình tarô ren trên máy tiện.

Mũi tarô được chọn phải lắp trong cán tarô đặc biệt; kiểu tốt nhất phải có tính năng kéo và nén, được gọi là *cán tarô lõng*, không được dùng ổ lắp mũi khoan hoặc thiết bị gá lắp tương tự, do có thể làm gãy mũi tarô và hư hại lỗ gia công.

Do không có chu kỳ tarô ren trên máy tiện CNC, từng chuyển động dao được lập trình trong block riêng rẽ. Để thực hiện điều đó, và để tìm phương pháp tarô thích hợp, trước hết bạn hãy đánh giá quy trình tarô ren phải (ren thuận), áp dụng trên máy tiện.

Bước 01 Xác lập vị trí tọa độ XY
Bước 02 Chọn dụng cụ và khoảng tốc độ
Bước 03 Chọn chiều quay và tốc độ trục chính
Bước 04 Chuyển động nhanh đến đường tâm và khoảng hở có bù
Bước 05 Tarô ren đến chiều sâu mong muốn.
Bước 06 Dừng trục chính
Bước 07 Đảo chiều quay trục chính
Bước 08 Lùi mũi tarô ra ngoài lỗ
Bước 09 Dừng trục chính
Bước 10 Chuyển động nhanh đến vị trí ban đầu
Bước 11 Trục chính quay theo chiều thuận hoặc kết thúc chương trình.

Khi được chuyển vào chương trình CNC một cách cẩn thận, quy trình này có thể sử dụng trong lập trình hằng ngày.

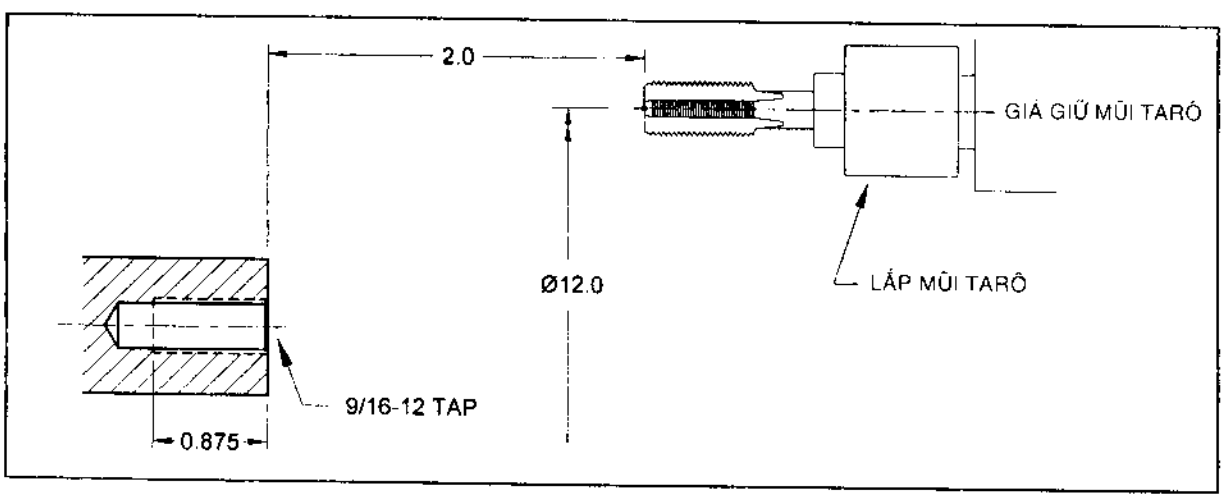
Hình 25.25 minh họa chi tiết và mũi tarô dùng cho ví dụ lập trình O2507. Chương trình ví dụ này tuân theo 11 bước nêu trên. Về kỹ thuật, chương trình này là đúng, nhưng thiếu tính thực tiễn. Có thể có các vấn đề trong chương trình O2507?

```
O2507 (TAPPING ON LATHES)
(ONLY THEORETICALLY CORRECT VERSION)
...
(T02 - TAP DRILL 31/64)
...
...
N42 M01

(T03 - 9/16-12 PLUG TAP)
N43 T0300 M42
N44 G97 S450 M03
N45 G00 X0 Z0.5 M08 T0303
N46 G01 Z-0.875 F0.0833
N47 M05
N48 M04
N49 Z0.5
N50 M05
N51 G00 X12.0 Z2.0 T0300 M09
N52 M30
%
```

Nếu chỉ xem qua chương trình O2507 sẽ khó phát hiện các sai sót. Nói chung, chương trình này bao quát mọi chuyển động cần thiết, do đó đúng về lý thuyết. *Thực tế, chương trình O2507 còn vài sai sót.*

Mọi bước tarô nêu trên đều được tuân thủ một cách cẩn thận, nhưng khi nghiên cứu vẫn hơn sẽ phát hiện hai lĩnh vực có thể có khó khăn hoặc thậm chí nguy hiểm. Vấn đề thứ nhất có thể nảy sinh nếu công tắc xác lập vượt



Hình 25.25. Gá lắp mũi tarô trên máy tiện CNC – Chương trình O2507 và O2508.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

qua tốc độ cắt lập trình, theo chế độ 100%. Bạn cần nhớ, tốc độ cắt khi tarô ren luôn luôn bằng bước ren (F0.0833 là tốc độ cắt đối với bước ren 12TPI). Nếu công tắc này xác lập giá trị khác 100%, ren sẽ bị trượt, dẫn đến gãy mũi tarô hoặc hư hại chi tiết gia công.

Vấn đề thứ hai chỉ trở nên rõ ràng khi chạy chế độ block đơn. Bạn hãy xem block N46 và N47. Trong block N46, mũi tarô đạt đến vị trí cuối trên trục Z – trong khi trục chính vẫn quay. Trục chính sẽ dừng trong block N47, nhưng trong chế độ block đơn, điều đó là quá trễ. Tình huống tương tự sẽ xảy ra trong chuyển động lùi mũi tarô ra ngoài. Trục chính đảo chiều trong block N48, nhưng chỉ chuyển động khi xử lý block N49. Do đó, chương trình O2507 không có tính thực tiễn khi tarô trên máy tiện CNC.

Có một số chi tiết không cần xét đến khi ứng dụng chu kỳ cố định (chẳng hạn chu kỳ tarô ren G84) trong các chương trình phay. Đối với phay, mọi chuyển động dao đều được thiết lập sẵn *bên trong* chu kỳ cố định. Để loại bỏ vấn đề thứ nhất (xác lập vượt qua tốc độ cắt) sự lập trình các hàm M48/M49 sẽ tạm thời xóa công tắc vượt qua tốc độ cắt. Nhưng tốt hơn, bạn nên thay lệnh chuyển động mũi tarô vào và ra từ chế độ G01 hiện hành sang chế độ G32 (hoặc G33 trên một số bộ điều khiển). Lệnh G32 thường được dùng để cắt ren với dao một lưỡi cắt. Với G32 có thể đạt được hai kết quả *trục chính được đồng bộ hóa* và *sự vượt quá tốc độ cắt sẽ không có hiệu lực* theo mặc định. Vấn đề thứ hai sẽ được giải quyết nếu các hàm M trục chính được lập trình trong *cùng block* với chuyển động dao. Điều đó có nghĩa là kết hợp block N46 với N47, và N48 với N49. Chương trình O2508 được cải tiến từ chương trình O2507 sẽ như sau:

```
O2608 (TAPPING ON LATHES)
(PRACTICALLY CORRECT VERSION)
...
(T02 - TAP DRILL 31/64)
---
...
N42 M01

(T03 - 9/16-12 PLUG TAP)
N43 T0300 M42
N44 G97 S450 M03
N45 G00 X0 Z0.5 M08 T0303
N46 G32 Z-0.875 F0.0833 M05
N47 Z0.5 M04
N48 M05
N49 G00 X12.0 Z2.0 T0300 M09
N50 M30
%
```

Block (trong ví dụ là N48) chứa hàm dừng trục chính M05, là không cần thiết nếu mũi tarô là dụng cụ cắt *cuối cùng* trong chương trình, dù không gây ra nguy hại trong chương trình khác. Bạn hãy so sánh chương trình O2508 với chương trình O2507. Chương trình O2508 ổn định hơn và hầu như loại trừ mọi vấn đề có thể xảy ra.

Các nguyên công khác

Có nhiều biến thể lập trình khác liên quan đến gia công lỗ trên trung tâm gia công và máy tiện CNC. Chương này chỉ khái quát một số khả năng quan trọng nhất và phổ biến nhất.

Một số ứng dụng khác, chẳng hạn gia công sử dụng dao doa ngược, dao doa khối, dao với nhiều lưỡi cắt,... có thể ít dùng trong lập trình. Tuy nhiên, lập trình các nguyên công đó không khó hơn lập trình các chuyển động dao thông dụng hàng ngày.

Khả năng thực của nhà lập trình CNC được xác định bằng sự áp dụng kiến thức và kinh nghiệm đối với vấn đề mới, đòi hỏi quá trình suy nghĩ sáng tạo và sức lao động.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

26 SƠ ĐỒ LỖ

Trong các gia công điểm – điểm, gồm khoan, chuốt, doa, tarô ren, ... thường yêu cầu gia công một lỗ hoặc chuỗi các lỗ với cùng dụng cụ cắt, và tiếp theo là các dụng cụ khác. Trong thực tế, gia công nhiều lỗ thường phổ biến hơn một lỗ. Gia công nhiều lỗ với cùng một dụng cụ có nghĩa là gia công *sơ đồ các lỗ*. "Sơ đồ" được hiểu là "*thiết kế hoặc sắp xếp hòa hợp hoặc đặc tính*". Hai hoặc nhiều lỗ được gia công với cùng một dụng cụ sẽ tạo thành sơ đồ. Sơ đồ lỗ mong muốn là sự sắp xếp trên bản vẽ một cách ngẫu nhiên *(thiết kế hoặc sắp xếp đặc tính)* hoặc theo trật tự xác định (*thiết kế hoặc sắp xếp hòa hợp*). Sự lập kích thước sơ đồ lỗ tuân theo chế độ kích thước tiêu chuẩn.

Chương này trình bày một số sơ đồ lỗ điển hình trên chi tiết phẳng và các phương pháp lập trình tương ứng. Để đơn giản hóa, mọi ví dụ lập trình liên quan đến sơ đồ lỗ đều có nguyên công khoan tâm, sử dụng mũi khoan tâm #2, với đường kính vạt miệng.150, đến chiều sâu.163 của chi tiết và dao được lắp sẵn trong trục chính. Để dễ đọc chương trình, trong các ví dụ sẽ không chuyên biệt đường kính lỗ, kích cỡ và chiều dày của vật liệu.

Từ định nghĩa nêu trên, cần xác định sơ đồ lỗ *đặc tính* hoặc *hài hòa*. Đơn giản, đây các lỗ bất kỳ được gia công với cùng một dụng cụ cắt, lần lượt qua từng lỗ, thường theo thứ tự hợp lý. Điều đó có nghĩa là mọi lỗ trong một sơ đồ sẽ có cùng đường kính danh định, gia công bắt đầu ở cùng mức R và kết thúc ở cùng chiều sâu Z. Nói chung, tất cả các lỗ trong sơ đồ đều được gia công như nhau với một dụng cụ cắt.

CÁC KIỂU SƠ ĐỒ LỖ

Các sơ đồ có thể được phân loại theo nhiều nhóm, mỗi nhóm đều có chung đặc tính. Sơ đồ lỗ trong lập trình CNC sẽ thuộc một trong các nhóm sau:

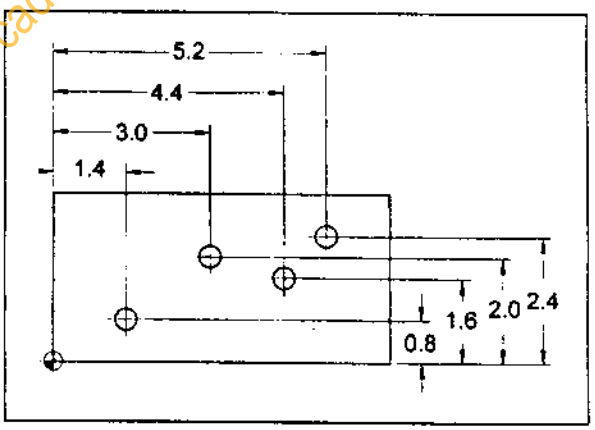
- ❑ Sơ đồ ngẫu nhiên
- ❑ Sơ đồ hàng thẳng
- ❑ Sơ đồ hàng chéo
- ❑ Sơ đồ góc
- ❑ Sơ đồ lưới
- ❑ Sơ đồ cung
- ❑ Sơ đồ đường tròn

Một số nhóm có thể được chia thành các nhóm nhỏ hơn. Hiểu rõ từng nhóm sẽ giúp bạn lập trình các sơ đồ nhóm tương tự.

Hiện có một số hệ điều khiển có sự lập trình sơ đồ lỗ được thiết lập sẵn, ví dụ *sơ đồ đường tròn*. Các module chương trình này sẽ đơn giản hóa rõ rệt sự lập trình sơ đồ lỗ, nhưng cấu trúc của chúng là đặc thù của bộ điều khiển, không thể áp dụng cho bộ điều khiển khác.

SƠ ĐỒ LỖ NGẪU NHIÊN

Sơ đồ thông dụng nhất trong lập trình gia công các lỗ là sơ đồ ngẫu nhiên. Đây là loại sơ đồ, trong đó các lỗ có chung đặc tính gia công, nhưng các khoảng cách X và Y giữa chúng là khác nhau. Nói cách khác, các lỗ trong sơ đồ ngẫu nhiên có cùng dụng cụ cắt, cùng đường kính danh nghĩa, thường là cùng chiều sâu, nhưng khoảng cách giữa chúng là khác nhau (Hình 27-1).



Hình 27-1. Sơ đồ lỗ ngẫu nhiên - ví dụ chương trình O2601.

Không có các kỹ thuật tiết kiệm thời gian được dùng trong lập trình sơ đồ ngẫu nhiên, chỉ chu kỳ cố định đã chọn được dùng ở các vị trí lỗ. Mọi tọa độ XY trong sơ đồ lỗ đều cần lập trình bằng tay; các tính năng của hệ thống điều khiển không thể hỗ trợ điều này:

```
O2601 (SƠ ĐỒ LỖ NGẪU NHIÊN)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X1.4 Y0.8 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0.1 Z-0.163 F3.0
N6 X3.0 Y2.0
N7 X4.4 Y1.6
N8 X5.2 Y2.4
```

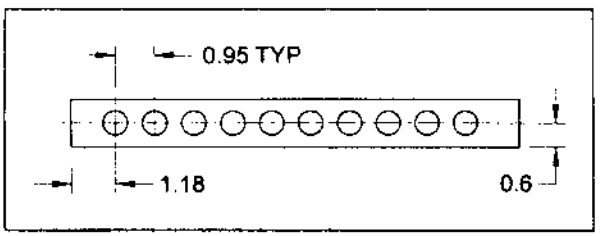

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N9 G80 M09
N10 G28 Z0 M05
N11 G28 X5.2 Y2.4
N12 M30
%
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

SƠ ĐỒ LỖ HÀNG NGANG

Sơ đồ lỗ song song với trục X hoặc Y có bước (khoảng cách) bằng nhau là sơ đồ *hàng ngang*. Hình 26.2 minh họa sơ đồ 10 lỗ theo trục X, với bước là 0.950 inch.



Hình 26.2. *Sơ đồ lỗ hàng ngang – Ví dụ chương trình O2602*

Phương pháp lập trình tận dụng ưu thế của tính năng lặp lại trong chu kỳ cố định, sử dụng địa chỉ L hoặc K. Sẽ không hiệu quả khi lập trình riêng rẽ từng lỗ. Dụng cụ cắt được định vị ở lỗ thứ nhất trong chế độ G90 phải đổi sang chế độ số gia G91, báo cho bộ điều khiển gia công 9 lỗ còn lại theo số gia, chỉ theo trục X. Logic này cũng áp dụng cho sơ đồ lỗ dọc theo trục Y. Trong trường hợp đó, số gia bước cần được lập trình chỉ theo trục Y. Chú ý, số lần lặp lại luôn luôn bằng *số khoảng cách*, không phải số lỗ. Lý do là lỗ thứ nhất đã được gia công trong block gọi chu kỳ.

```
O2602 (SƠ ĐỒ LỖ THẲNG)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X1.18 Y0.6 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0.1 Z-0.163 F3.0
N6 G91 X0.95 L9
N7 G80 M09
N8 G28 Z0 M05
N9 G28 X0 Y0
N10 M30
%
```

Cần nhấn mạnh hai tính năng trong chương trình O2602. Trong block N6, chế độ lập kích thước G90 được đổi sang G91, để tận dụng ưu thế của khoảng cách bước bằng nhau. Khi 10 lỗ đã được gia công, chương trình cần có chuyển động trở về vị trí zero máy, trong ví dụ này, dọc theo cả ba trục. Tuy nhiên, nếu không tính toán, sẽ *không biết* vị trí tuyệt đối ở lỗ thứ 10 theo trục X (trục Y có giá trị vị trí không thay đổi.06 inch = Y 0.6). Để giải quyết vấn đề này, bạn hãy xóa chu kỳ với G80, để chế độ G91 có hiệu lực và dịch chuyển về vị trí zero máy theo trục Z trước (vì các lý do an toàn). Sau đó, vẫn trong chế độ số gia G91 – đồng thời trở về zero máy trên cả hai trục X và Y.

Nói chung, sau dụng cụ cắt thứ nhất trong ví dụ này là các dụng cụ khác để hoàn tất gia công lỗ. Để bảo vệ chương trình và sự gia công không gặp các vấn đề, bạn cần bảo đảm lệnh chế độ tuyệt đối G90 được gọi lại cho từng dụng cụ cắt kế tiếp.

SƠ ĐỒ LỖ HÀNG CHÉO

Sơ đồ các lỗ theo hàng chéo là biến thể của sơ đồ hàng ngang. Sự khác biệt là bước số gia áp dụng cho cả hai trục X và Y. Sơ đồ lỗ kiểu này được thiết lập trên bản vẽ chi tiết sử dụng một trong hai phương pháp lập kích thước:

- Các tọa độ X và Y được cho theo lỗ thứ nhất và lỗ cuối cùng

Trong phương pháp này, vị trí chéo không được chuyên biệt và không nêu rõ bước giữa các lỗ:

- Chỉ ghi các tọa độ X và Y cho lỗ thứ nhất.

Trong phương pháp này, vị trí chéo được chuyên biệt và ghi rõ bước giữa các lỗ.

Trong trường hợp trên, mọi kích thước X và Y cần thiết đều khả dụng để viết chương trình. Tuy nhiên, phương pháp lập trình sẽ khác nhau giữa các kiểu ghi kích thước trên bản vẽ.

Sơ đồ xác định theo tọa độ

Phương pháp lập trình này tương tự như sơ đồ hàng ngang. Do chưa biết bước giữa các lỗ, cần tính số gia dọc theo từng trục X và Y. Khoảng cách trục này thường được gọi là *khoảng cách delta* (delta X được đo theo trục X, delta Y đo theo trục Y). Tính toán đó có thể được thực hiện theo hai cách có độ chính xác như nhau.

Phương pháp tính toán thứ nhất có thể sử dụng các hàm lượng giác, hoặc dùng tỷ số giữa các cạnh. Trên Hình 26.3, chiều dài sơ đồ theo trục X là 10.82 inch và dọc theo trục Y là 2.0:

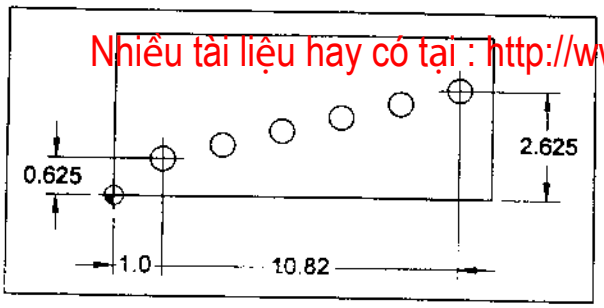
(2.625 – 0.625 = 2.0)

Sơ đồ loại này có tất cả các lỗ cách đều theo các trục X và Y. Do tất cả các lỗ cách đều, tỷ số các cạnh của từng lỗ trên trục X bằng khoảng cách toàn phần 10.82 chia cho số khoảng cách trên trục X; số gia dọc theo trục Y bằng khoảng cách toàn phần 2.0 chia cho số khoảng cách trên trục Y. Số khoảng cách của sơ đồ 6 lỗ là 5, do đó số gia trục X (delta X) là:

10.82 / 5 = 2.1640

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 26.3. Sơ đồ hàng chéo với hai tập hợp tọa độ – chương trình O2603

và số gia trục Y (delta Y) là:

2.0 / 5 = 0.4

Phương pháp thứ hai sử dụng các hàm lượng giác, cũng có thể được dùng để kiểm chứng phương pháp thứ nhất và ngược lại. *Cả hai kết quả phải đồng nhất*. Trước hết cần tính các giá trị trung gian:

A = tg^{-1} (2.0 / 10.82) = 10.4725/349^{0}

C = 2.0 / sinA = 11.00329063

C1 = C / 5 = 2.20065813

Số gia thực tế dọc theo trục có thể được tính toán, sử dụng kích thước C1 làm khoảng cách giữa các lỗ:

Số gia X = C1 x cosA = 2.1640
Số gia Y = C1 x sinA = 0.4000

Các số gia tính toán là như nhau trong cả hai phương pháp, do đó kết quả là đúng, có thể được dùng để viết chương trình (O2603) – block N6 chứa các giá trị đó:

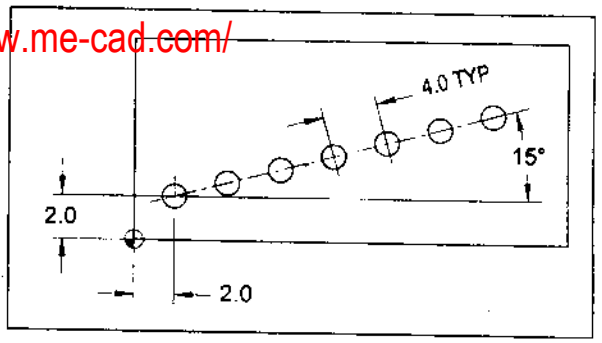
```
O2603 (ANGULAR ROW 1)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X1.0 Y0.625 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0.1 Z-0.163 F3.0
N6 G91 X2.164 Y0.4 L5 (K5)
N7 G80 M09
N8 G28 Z0 M05
N9 G28 X0 Y0
N10 M30
%
```

Chú ý, cấu trúc chương trình là đồng nhất với ví dụ về sơ đồ hàng ngang, ngoại trừ dịch chuyển số gia với địa chỉ L5 (K5) dọc theo hai trục, thay vì một.

Sơ đồ xác định theo góc

Sơ đồ hàng chéo còn có thể được xác định trên bản vẽ bằng các tọa độ X và Y của lỗ thứ nhất, số các lỗ cách đều, khoảng cách giữa các lỗ và góc nghiêng (Hình 26.4).

Để tính toán các giá trị tọa độ X, Y, bạn hãy dùng các hàm lượng giác:



Hình 26.4. Sơ đồ hàng chéo với các tọa độ, bước, và góc nghiêng – chương trình O2604

X = 4.0 x cos15^{0} = 3.863703305
Y = 4.0 x sin15^{0} = 1.03527618

Chương trình có thể được viết sau khi bạn làm tròn các giá trị tính toán – chương trình O2604:

```
O2604 (ANGULAR ROW 2)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X2.0 Y2.0 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0.1 Z-0.163 F3.0
N6 G91 X3.8637 Y1.0353 L6 (K6)
N7 G80 M09
N8 G28 Z0 M05
N9 G28 X0 Y0
N10 M30
%
```

Do các số gia tính toán là các giá trị được làm tròn, sai số tích lũy là không thể tránh. Trong hầu hết các trường hợp, sai số bất kỳ có thể chỉ trong các giới hạn dung sai cho phép. Tuy nhiên đối với các chi tiết đòi hỏi độ chính xác cao, sai số này có thể quan trọng và cần phải xem xét.

Để bảo đảm mọi tính toán đều đúng, có thể sử dụng phương pháp đơn giản để so sánh các giá trị tính toán:

➲ **Bước 1**

Tìm các tọa độ tuyệt đối XY của lỗ *cuối*

X = 2.0 + (4.0 x 6 x cos15^{0}) = X25.1822
Y = 2.0 + (4.0 x 6 x sin15^{0}) = Y8.2117

➲ **Bước 2**

So sánh các tọa độ XY mới với các số gia tính toán trước đó theo quan hệ của chúng với lỗ cuối cùng của sơ đồ (sử dụng các giá trị đã làm tròn):

X = 2.0 + 3.8637 x 6 = 25.1822
Y = 2.0 + 1.0353 x 6 = 8.2118

Chú ý, các giá trị X và Y đều chính xác. Khi làm tròn, đặc biệt khi có nhiều lỗ, sai số tích lũy có thể làm cho sơ đồ vượt ra ngoài dung sai cho phép. Trong trường hợp đó, cách duy nhất để lập trình là tính toán các tọa độ của từng lỗ

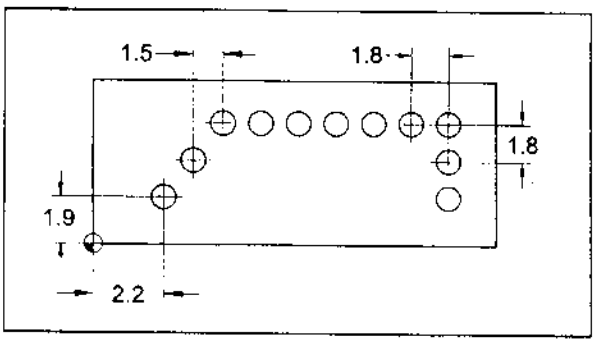
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

theo kích thước tuyệt đối (nghĩa là từ điểm *chung* thay vì điểm *trước đó*). Quy trình lập trình sẽ hơi dài hơn, nhưng chính xác hơn nhiều.

SƠ ĐỒ GÓC

Sơ đồ các lỗ có thể được sắp xếp theo góc - đây là sơ đồ kết hợp các sơ đồ hàng ngang và/hoặc chéo (Hình 26.5).



Hình 26.5. *Sơ đồ góc – ví dụ lập trình O2605*

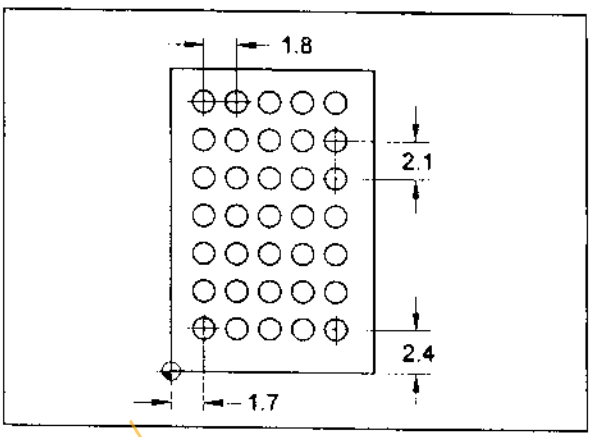
Mọi nguyên tắc đối với sơ đồ hàng ngang và chéo đều áp dụng đúng cho sơ đồ góc. Sự khác biệt quan trọng nhất là lỗ góc, *chung* cho cả hai hàng. Sơ đồ góc có thể được lập trình bằng cách gọi chu kỳ cố định cho từng hàng. Bạn sẽ nhanh chóng nhận thấy mỗi lỗ góc sẽ được gia công *hai lần*. Bạn hãy hình dung toàn bộ quy trình, lỗ *cuối cùng* của một hàng cũng là lỗ *thứ nhất* của hàng kế tiếp. Đối với nhiều sơ đồ góc, bạn có thể thiết lập macro riêng. Giải pháp thông thường là dịch chuyển dao đến vị trí thứ nhất gọi chu kỳ đã chọn, và *giữ nguyên trong* chu kỳ đó:

```
O2605 (CORNER PATTERN)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X2.2 Y1.9 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0.1 Z-0.163 F3.0
N6 G91 X1.5 Y1.8 L2 (K2)
N7 X1.8 L6 (K6)
N8 Y-1.8 L2 (K2)
N9 G80 M09
N10 G28 Z0 M05
N11 G28 X0 Y0
N12 M30
%
```

Chương trình tương đối dễ hiểu. Trong block N6, các lỗ hàng chéo được gia công, bắt đầu từ lỗ dưới bên trái, trong N7 là các lỗ hàng ngang, và N8 là các lỗ hàng dọc được gia công. Thứ tự này là liên tục. Cũng như trong các ví dụ trước, bạn hãy nhớ số lần lặp lại L hoặc K là số lượng dịch chuyển (số khoảng cách) không phải số lỗ.

SƠ ĐỒ LƯỚI

Sơ đồ lưới thẳng cơ bản có thể được định nghĩa là tập hợp các lỗ ngang và dọc cách đều, mỗi hàng (cột) có cùng số lỗ cách đều. Nếu khoảng cách giữa các lỗ dọc bằng khoảng cách giữa các lỗ ngang lưới sẽ là hình vuông. Nếu khoảng cách giữa các lỗ dọc khác với khoảng cách ngang tương ứng, lưới sẽ là hình chữ nhật. Sơ đồ lưới còn được gọi là *sơ đồ lỗ chữ nhật* (Hình 26.6).



Hình 26.6. *Sơ đồ lưới chữ nhật – ví dụ chương trình O2606.*

Sơ đồ lưới tương tự chuỗi sơ đồ góc sử dụng các kỹ thuật lập trình gần như nhau. Vấn đề cơ bản khi lập trình sơ đồ lưới là tính hiệu quả. Từng hàng có thể được lập trình theo sơ đồ hàng đơn, ví dụ, bắt đầu từ bên trái của mỗi hàng. Về kỹ thuật, điều đó là đúng, dù không hiệu quả do tốn thời gian, khi dao chuyển động từ lỗ cuối cùng của một hàng đến lỗ đầu tiên của hàng kế tiếp.

Phương pháp hiệu quả hơn có dạng chuyển động gấp khúc. Để lập trình chuyển động này, cần lập trình hàng hoặc cột thứ nhất bắt đầu từ lỗ góc bất kỳ. Hoàn tất hàng (cột) đó, chuyển sang lỗ gần nhất của hàng (cột) kế tiếp và lặp lại quá trình này cho tất cả các hàng (cột) còn lại. Tổn thất thời gian do chuyển động nhanh sẽ ở mức tối thiểu.

```
O2606 (STRAIGHT GRID PATTERN)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X1.7 Y2.4 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0.1 Z-0.163 F3.0
N6 G91 Y2.1 L6 (K6)
N7 X1.8
N9 Y-2.1 L6 (K6)
N10 X1.8
N11 Y2.1 L6 (K6)
N12 X1.8
N13 Y-2.1 L6 (K6)
N14 X1.8
N15 Y2.1 L6 (K6)
```

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

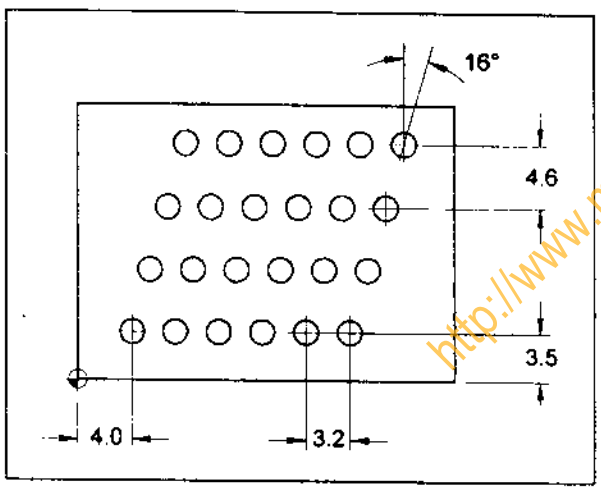
Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N16 G80 M09
N17 G28 Z0 M05
N18 G28 X0 Y0
N19 M30
%
```

Cần chú ý hai tính năng trong chương trình này – thứ nhất là chuyển từ một hàng trong sơ đồ đến hàng kế tiếp – không cần lặp lại địa chỉ L hoặc K, do chỉ *một* lỗ được gia công ở vị trí đó. Tính năng thứ hai có thể khó nhận biết hơn. Để rút ngắn chương trình, bạn hãy bắt đầu từ trục chứa số lượng lỗ *lớn hơn* (trục Y trong chương trình O2606). Ví dụ này là biến thể của các ví dụ trước, áp dụng tất cả các nguyên tắc đã nêu. Chương trình con đặc biệt được viết cho sơ đồ lưới là phương pháp lập trình thông dụng hiện nay.

Sơ đồ lưới nghiêng

Mặc dầu sơ đồ lưới thẳng là dạng phổ biến nhất với sắp xếp các lỗ theo hình vuông hoặc chữ nhật, nhưng sơ đồ lưới còn có thể dạng hình bình hành, được gọi là *sơ đồ lưới nghiêng* (Hình 26.7).



Hình 26.7. Sơ đồ lưới nghiêng, chương trình O2607

Phương pháp lập trình tương tự sơ đồ lưới chữ nhật, nhưng phải tính thêm số gia (bước) theo đường chéo, tương tự các phương pháp đã nêu.

Số gia (bước lỗ) chưa biết trên bản vẽ là khoảng cách đo theo trục X, trừ lỗ thuộc hàng ngang đến lỗ kế tiếp trong hàng ngang tiếp theo:

$$X = 4.6 \times tg16^0 = 1.319028774 \text{ (X1.319)}$$

Chương trình có thể viết theo phương pháp tương tự lưới hàng ngang, nhưng sự "chuyển tiếp" giữa các hàng sẽ xảy ra dọc theo hai trục:

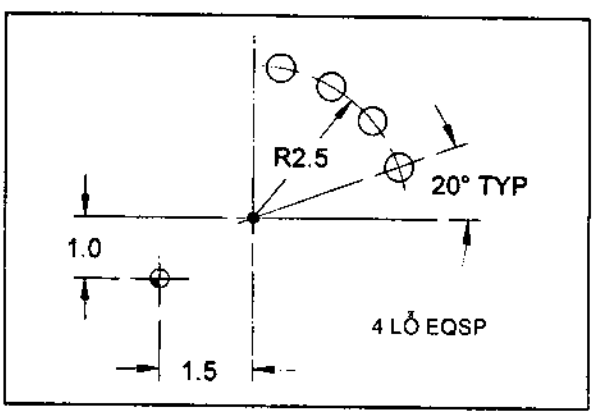
```
O2607 (ANGULAR GRID)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X4.0 Y3.5 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0.1 Z-0.163 F3.0
N6 G91 X3.2 L5 (K5)
N7 X1.319 Y4.6
N8 X-3.2 L5 (K5)
N9 X1.319 Y4.6
N10 X3.2 L5 (K5)
N11 X1.319 Y4.6
N12 X-3.2 L5 (K5)
N13 G80 M09
N14 G28 Z0 M05
N15 G28 X0 Y0
N16 M30
%
```

Nhiều nhà lập trình kinh nghiệm sẽ xem xét phương pháp tăng hiệu suất cho các chương trình gia công lưới nghiêng bằng cách sử dụng chương trình con hoặc *User Macros*. Các chương trình con đặc biệt hữu dụng đối với các sơ đồ lưới gồm nhiều hàng và cột. Nội dung về các chương trình con và ví dụ thực tiễn được trình bày trong Chương 38.

SƠ ĐỒ LỖ HÌNH CUNG TRÒN

Sơ đồ gồm các lỗ cách đều sắp xếp trên cung tròn (một phần đường tròn) được gọi là *sơ đồ cung*.

Phương pháp lập trình sơ đồ cung về cơ bản cũng tương tự các sơ đồ lỗ khác. Bạn chọn lỗ thứ nhất là lỗ thuận tiện nhất. Lỗ nào dễ tìm các tọa độ hơn, lỗ đầu tiên hay lỗ cuối cùng trên cung tròn? Có lẽ bắt đầu từ 0^0 sẽ tốt hơn? (Hình 26.8).



Hình 26.8. Sơ đồ cung – chương trình O2608

Trong sơ đồ này, vị trí tâm cung tròn đã biết, bạn cũng biết bán kính cung, khoảng cách góc giữa các lỗ, và số lỗ cách đều trên chu vi.

Cần tính toán để tìm tọa độ X và Y của từng tâm lỗ. Tính toán sử dụng các hàm lượng giác áp dụng riêng rẽ cho từng lỗ- mọi số liệu đều được ghi trên bản vẽ.

Cần tính tọa độ của *cả hai* trục X và Y cho

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

từng lỗ, sử dụng *hai* công thức lượng giác cơ bản. Cách tốt nhất để minh họa sự lập trình sơ đồ lỗ là dùng ví dụ trên bản vẽ. Nhiệm vụ lập trình được chia thành 4 bước:

➲ **Bước 1**

Bắt đầu từ lỗ gần 0^0 nhất, sau đó tiếp tục cho các lỗ kế tiếp ngược chiều kim đồng hồ trên cung tròn.

➲ **Bước 2**

Sử dụng các hàm lượng giác để tính tọa độ X và Y của lỗ *thứ nhất*:

Lỗ #1

X = 1.5 + 2.5 x cos 20^0 = 3.849231552 (X3.8492)
Y = 1.0 + 2.5 x sin 20^0 = 1.855050358 (Y1.8551)

➲ **Bước 3**

Sử dụng các hàm lượng giác như Bước 2 để tính các tọa độ XY cho 3 lỗ còn lại. Đối với từng lỗ trong sơ đồ, lần lượt cộng thêm góc 20^0, do đó góc lỗ thứ hai là 40^0, thứ ba là 60^0, và thứ tư là 80^0 ...

Lỗ #2

X = 1.5 + 2.5 cos 40^0 = 3.415111108 (X3.415
Y = 1.0 + 2.5 sin 40^0 = 2.606969024 (Y2.607)

Lỗ #3

X = 1.5 + 2.5 cos 60^0 = 2.750000000 (X2.75)
Y = 1.0 + 2.5 sin 60^0 = 3.165063509 (Y3.1651)

Lỗ #4

X = 1.5 + 2.5 cos 80^0 = 1.934120444 (X1.9341)
Y = 1.0 + 2.5 sin 80^0 = 3.462019383 (Y3.462)

➲ **Bước 4**

Nếu các tọa độ XY được tính toán theo đúng thứ tự sẽ sử dụng trong chương trình, danh sách các vị trí lỗ có thể liệt kê theo thứ tự đó:

Lỗ # 1: X3.8492 Y1.8551
Lỗ # 2: X3.4151 Y2.6070
Lỗ # 3: X2.7500 Y3.1651
Lỗ # 4: X1.9341 Y3.4620

Với tọa độ các lỗ đã tính toán, bạn có thể viết sơ đồ cung – chương trình O2608

```
O2608 (ARC PATTERN)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X3.8492 Y1.8551 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0.1 Z-0.163 F3.0
N6 X3.4151 Y2.607
N7 X2.75 Y3.1651
N8 X1.9341 Y3.462
N9 G80 M09
N10 G28 Z0.1 M05
N11 G28 X1.9341 Y3.462
N12 M30
%
```

Còn hai phương pháp khác (có lẽ hiệu quả hơn) để lập trình sơ đồ các lỗ sắp xếp theo cung tròn. Phương pháp thứ nhất, tận dụng ưu thế của hệ tọa độ cục bộ (địa phương) G52, sẽ được trình bày trong Chương 39. Phương pháp thứ hai sẽ dùng hệ tọa độ cực (tùy chọn trên hầu hết các bộ điều khiển), sẽ được đề cập ở cuối chương này – trong chương trình O2610.

SƠ ĐỒ LỖ THEO ĐƯỜNG TRÒN

Sơ đồ các lỗ cách đều trên chu vi đường tròn được gọi là *sơ đồ lỗ theo đường tròn*. Do đường kính hình tròn là đường kính bước thực hiện của sơ đồ, loại sơ đồ này còn được gọi là *sơ đồ bước đường tròn*. Phương pháp lập trình cũng tương tự các sơ đồ khác, đặc biệt là sơ đồ các lỗ theo cung tròn, chủ yếu phụ thuộc vào sự định hướng của sơ đồ và phương pháp lập kích thước bản vẽ.

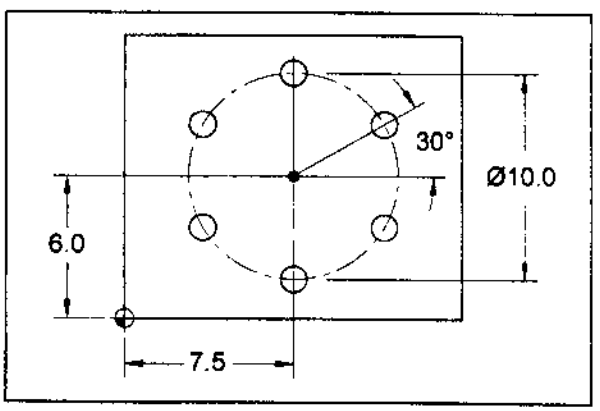
Đường tròn trên bản vẽ được xác định bằng các tọa độ XY của tâm đường tròn, bán kính hoặc đường kính, số lỗ cách đều trên chu vi, định hướng góc của từng lỗ theo trục X, ...

Trên chu vi đường tròn có thể có số lỗ cách đều bất kỳ, tuy nhiên, số lỗ phổ biến là:

4, 5, 6, 8, 10, 12,16, 18, 20, 24

Trong các ví dụ kế tiếp, sơ đồ 6-lỗ và 8-lỗ (và các bội số của chúng) có *hai* quan hệ góc tiêu chuẩn theo trục X là 0^0.

Hình 26.9 minh họa bản vẽ sơ đồ đường tròn điển hình.Phương pháp lập trình cho sơ đồ này tương tự sơ đồ cung tròn.



***Hình 26.9**. Sơ đồ lỗ theo chu vi đường tròn – chương trình O2609*

Trước hết, bạn chọn vị trí bắt đầu gia công lỗ, thường là ở zero chương trình, sau đó tìm các tọa độ XY tuyệt đối của tâm hình tròn. Trong hình minh họa, tọa độ tâm hình tròn là X7.5Y6.0. Sẽ không gia công vị trí này, nhưng tâm hình tròn sẽ là điểm khởi đầu để tính toán

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tất cả các tọa độ trên chiều thứ hai. Để biết các tọa độ tâm hình tròn, bạn cần ghi lại. Từng tọa độ lỗ trên chu vi phải được điều chỉnh theo một trong các giá trị tọa độ đó. Sau khi tính toán cho lỗ thứ nhất (dựa theo tâm hình tròn), bạn hãy tiếp tục tính toán các tọa độ XY của mọi lỗ còn lại trên chu vi.

Trong ví dụ O2609 có 6 lỗ cách đều trên chu vi hình tròn đường kính 10.0 inch. Điều đó có nghĩa là số gia 60^0 giữa các lỗ ($360^0/6 = 60^0$). Vị trí bắt đầu gia công thường ở biên giữa các góc phần tư. Điều đó có nghĩa là nơi bắt đầu sẽ ở vị trí tương ứng 3, 12, 9, hoặc 6 giờ trên mặt đồng hồ. Trong ví dụ này, sự khởi đầu sẽ ở vị trí 3 giờ. Không có lỗ ở vị trí đã chọn, lỗ gần nhất sẽ theo góc 30^0 ngược chiều kim đồng hồ. Bạn có thể ghi lỗ này là lỗ số 1. Các lỗ còn lại được xác định tương tự, theo thứ tự gia công, theo quan hệ với lỗ thứ nhất.

Bạn hãy lưu ý, các tính toán vị trí lỗ đều dùng hàm lượng giác. Tuy có thể dùng phương pháp tính toán khác, nhưng bạn nên chú ý tính thống nhất của mọi phép toán:

Lỗ #1

$X = 7.5 + 5.0 \times \cos 30^0 = 11.830127$ (X11.8301)
$Y = 6.0 + 5.0 \times \sin 30^0 = 8.500000$ (Y8.5)

Lỗ #2

$X = 7.5 + 5.0 \times \cos 90^0 = 7.5000000$ (X7.5)
$Y = 6.0 + 5.0 \times \sin 90^0 = 11.0000000$ (Y11.0)

Lỗ #3

$X = 7.5 + 5.0 \times \cos 150^0 = 3.16987298$ (X3.1699)
$Y = 6.0 + 5.0 \times \sin 150^0 = 8.50000000$ (Y8.5)

Lỗ #4

$X = 7.5 + 5.0 \times \cos 210^0 = 3.16987298$ (X3.1699)
$Y = 6.0 + 5.0 \times \sin 210^0 = 3.50000000$ (Y3.5)

Lỗ #5

$X = 7.5 + 5.0 \times \cos 270^0 = 7.50000000$ (X7.5)
$Y = 6.0 + 5.0 \times \sin 270^0 = 1.00000000$ (Y1.0)

Lỗ #6

$X = 7.5 + 5.0 \times \cos 330^0 = 11.830127$ (X11.8301)
$Y = 6.0 + 5.0 \times \sin 330^0 = 3.500000$ (Y3.5)

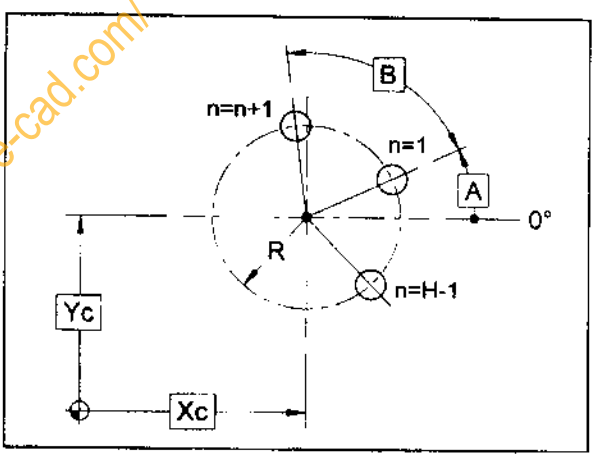
Sau khi tính tất cả các tọa độ, chương trình được viết hoàn toàn như các ví dụ đã nêu:

```
O2609 (BOLT CIRCLE PATTERN)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X11.8301 Y8.5 S900 M03
N4 G43 Z1.0 H01 M08
N5 G99 G81 R0 1 Z-0.163 F3.0
N6 X7.5 Y11.0
N7 X3.1699 Y8.5
N8 Y3.5
N9 X7.5 Y1.0
N10 X11.8301 Y3.5
N11 G80 M09
N12 G91 G28 Z0.0
N13 G91 G28 X0 Y0
N14 M30
%
```

Sẽ logic hơn nếu chọn tâm hình tròn làm zero chương trình, thay vì góc dưới bên trái của chi tiết. Phương pháp này sẽ loại bỏ sự chỉnh sửa theo vị trí tâm hình tròn cho từng giá trị tọa độ và có lẽ sẽ giảm bớt khả năng xảy ra sai sót. Đồng thời, có thể sẽ khó khăn hơn khi xác lập sự bù chi tiết G54 trên công cụ. Giải pháp tốt nhất là sử dụng phương pháp *bù tọa độ cục bộ (địa phương)* G52. Phương pháp này đặc biệt, hữu ích cho các ứng dụng yêu cầu chuyển sơ đồ đường tròn (hoặc sơ đồ bất kỳ) sang các vị trí khác của chi tiết cùng loại. Chương 39 trình bày chi tiết về lệnh G52.

Công thức sơ đồ đường tròn

Trong các công thức nêu trên có nhiều dữ liệu lặp lại. Kiểu tính toán này đưa ra cơ hội thiết lập công thức tổng quát có thể dùng làm cơ sở cho chương trình máy tính, tính toán dữ liệu nhập,... (Hình 26.10).



Hình 26.10. *Cơ sở của công thức tổng quát tính toán tọa độ sơ đồ lỗ theo đường tròn.*

Sử dụng công thức tổng quát có thể tính tọa độ của lỗ bất kỳ trong sơ đồ đường tròn. Công thức là tương tự cho cả hai trục:

$$X = \cos((n-1) \times B + A) + R + X_C$$

$$Y = \sin((n-1) \times B + A) + R + Y_C$$

Trong đó:

X = Tọa độ X của lỗ
Y = Tọa độ Y của lỗ
N = Số lỗ – ngược chiều kim đồng hồ từ 0^0
H = Số lượng các lỗ liên tiếp = $360^0/H$
A = Góc lỗ thứ nhất – Tính từ 0^0
R = Bán kính hình tròn
X_C = Tâm hình tròn từ gốc tọa độ X
Y_C = Tâm hình tròn từ gốc tọa độ Y

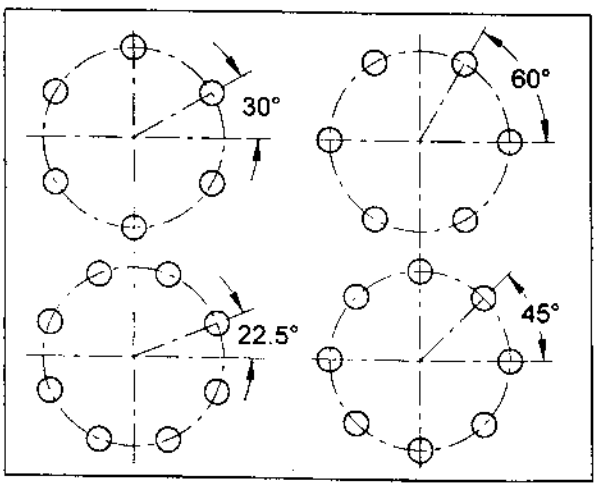
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Định hướng sơ đồ đường tròn

Định hướng sơ đồ đường tròn được xác định theo góc lỗ thứ nhất từ 0^0 của đường tròn. Trong ứng dụng hằng ngày, các sơ đồ đường tròn không chỉ có số lỗ khác nhau mà còn có các định hướng khác nhau. Các đường tròn có số lỗ cách đều thông dụng nhất dựa trên bội số của sáu (6, 12, 18, ...) và bội số của tám (8, 16, 24, ...). Quan hệ này khá quan trọng do định hướng của lỗ thứ nhất sẽ ảnh hưởng đến vị trí của tất cả các lỗ còn lại trong sơ đồ đường tròn.

Hình 26.11 minh họa vị trí lỗ thứ nhất theo vị trí 0^0 trên đường tròn tương ứng vị trí 3 giờ trên mặt đồng hồ.



Hình 26.11. Định hướng lỗ trên chu vi hình tròn 6 – lỗ và 8 – lỗ

HỆ TỌA ĐỘ CỰC

Cho đến phần này, mọi tính toán liên quan đến sơ đồ lỗ theo cung hoặc đường tròn đều dùng hàm lượng giác để tính tọa độ. Điều này dường như hơi chậm đối với hệ thống CNC hiện đại với máy tính mạnh. Hiện có phương pháp lập trình đặc biệt (thường là tùy chọn trên bộ điều khiển) dựa trên *hệ tọa độ cực*. Có hai hàm tọa độ cực khả dụng, thường được viết theo block riêng rẽ:

G15	Xóa hệ tọa độ cực	OFF
G16	Hệ tọa độ cực	ON

Các giá trị nhập chương trình cho sơ đồ cung hoặc đường tròn có thể được lập trình với các lệnh hệ tọa độ cực. Bạn hãy xem lại các tùy chọn của bộ điều khiển trước khi sử dụng phương pháp này. Định dạng là đồng nhất – ví dụ:

```
N.. G9.. G8.. X.. Y.. R.. Z.. F..
```

Có hai yếu tố phân biệt giữa chu kỳ tiêu chuẩn và chu kỳ trong chế độ tọa độ cực.

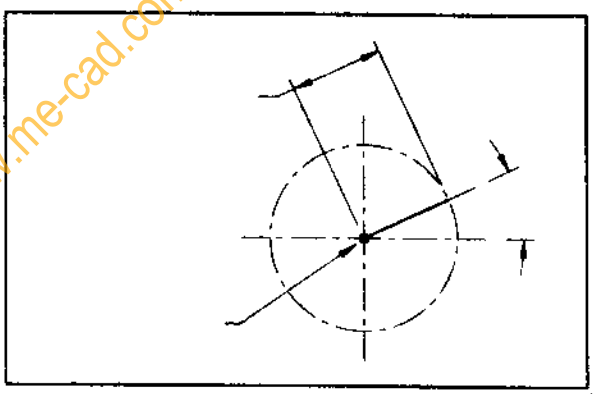
Yếu tố thứ nhất là lệnh G ban đầu đứng trước chu kỳ – không cần mà G đặc biệt cho chu kỳ tiêu chuẩn. Đối với chu kỳ bất kỳ được lập trình trong chế độ hệ tọa độ cực, lệnh chuẩn bị G16 phải được sử dụng để kích hoạt chế độ cực (chế độ ON). Khi hoàn tất chế độ cực và không còn cần thiết trong chương trình, cần kết thúc bằng lệnh G15 (chế độ OFF). Cả hai lệnh này đều phải trong block riêng:

```
N.. G16                    (POLAR COORDINATES ON)
N.. G9.. G8.. X.. Y.. R.. Z.. F..
N.. ...
N.. ...                    (MACHINING HOLES)
N.. ...
N.. G15                    (POLAR COORDINATES OFF)
```

Yếu tố thứ hai là *ý nghĩa* của tọa độ X và Y. Trong chu kỳ tiêu chuẩn, XY xác định vị trí lỗ theo hệ trục tọa độ vuông góc, thường là vị trí tuyệt đối. Trong tọa độ cực và G17 có hiệu lực (mặt phẳng XY), các tọa độ X và Y có ý nghĩa khác, xác định bán kính và góc:

- X xác định bán kính hình tròn của sơ đồ lỗ
- Y xác định góc của lỗ, đo từ 0^0

Hình 26.12 minh họa ba yêu cầu nhập cơ bản trong hệ tọa độ cực.



Hình 26.12. Ba đặc tính cơ bản của tọa độ cực

Ngoài các giá trị X và Y, tọa độ cực còn yêu cầu *tâm* quay. Đây là *điểm cuối cùng* được lập trình trước lệnh G16. Ở phần trước, dữ liệu trong chương trình O2608 và Hình 26.8 được tính toán bằng các hàm lượng giác. Với tùy chọn điều khiển tọa độ cực, chương trình đó sẽ được đơn giản hóa rõ rệt – O2610:

```
O2610 (ARC PATTERN - POLAR)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X1.5 Y1.0 S900 M03 (PIVOT POINT)
N4 G43 Z1.0 H01 M08
N5 G16                     (POLAR COORDINATES ON)
N6 G99 G81 X2.5 Y20.0 R0.1 Z-0.163 F3.0
N7 X2.5 Y40.0
N8 X2.5 Y60.0
N9 X2.5 Y80.0
N10 G15                    (POLAR COORDINATES OFF)
N11 G80 M09
```

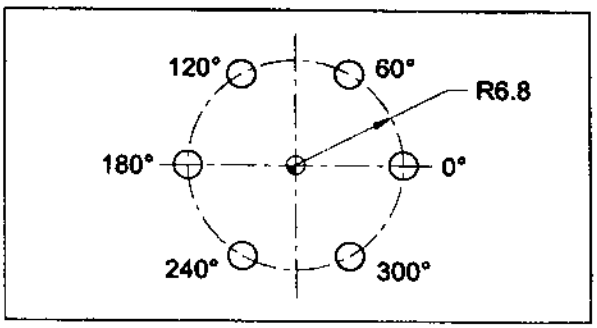
http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N12 G91 G28 Z0 M5
N13 G28 X0 Y0
N14 M30
%
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trong chương trình kế tiếp O2611, các lỗ cách đều trên chu vi hình tròn. Các kích thước trên Hình 26.13 áp dụng cho phương pháp lập trình tọa độ cực.



Hình 26.13. Hệ tọa độ cực áp dụng cho sơ đồ đường tròn – chương trình O2611.

```
O2611 (G15-G16 EXAMPLE)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S900 M03     (PIVOT POINT)
N4 G43 Z1.0 H01 M08
N5 G16                    (POLAR COORDINATES ON)
N6 G99 G81 X6.8 Y0 R0.1 Z-0.163 F3.0
N7 X6.8 Y60.0
N8 X6.8 Y120.0
N9 X6.8 Y180.0
N10 X6.8 Y240.0
N11 X6.8 Y300.0
N12 G15                   (POLAR COORDINATES OFF)
N13 G80 M09
N14 G91 G28 Z0 M05
N15 G28 X0 Y0
N16 M30
%
```

Chú ý, tâm tọa độ cực (còn gọi là tâm *xoay*) được xác định trong block 3 – là *vị trí X và Y cuối cùng* được lập trình *trước* khi gọi lệnh tọa độ cực G16. Trong chương trình O2611, tâm này ở vị trí X0Y0 (block N3) – bạn hãy so sánh với chương trình O2610.

Cả hai, giá trị bán kính và góc, đều có thể được lập trình trong chế độ tuyệt đối G90 hoặc chế độ số gia G91.

Nếu ứng dụng cụ thể yêu cầu nhiều sơ đồ theo cung hoặc đường tròn, bạn có thể bổ sung tùy chọn hệ tọa độ cực vào bộ điều khiển. Nếu lắp đặt tùy chọn User Macros của Fanuc, có thể tạo ra các chương trình macro mà *không cần* có hệ tọa độ cực trong bộ điều khiển, đồng thời bạn sẽ có tính linh hoạt cao hơn khi lập trình.

Lựa chọn mặt phẳng

Chương 28, và đặc biệt là Chương 30, sẽ trình bày về các mặt phẳng. Có ba mặt phẳng toán học được dùng cho nhiều ứng dụng, chẳng hạn tọa độ cực.

G17	Chọn mặt phẳng XY
G18	Chọn mặt phẳng ZY
G19	Chọn mặt phẳng YZ

Lựa chọn đúng mặt phẳng là *cực kỳ quan trọng* để sử dụng chính xác các tọa độ cực. Bạn cần tập thói quen lập trình đúng mặt phẳng cần thiết, kể cả mặt phẳng G17 mặc định.

Mặt phẳng G17 là mặt XY. Nếu làm việc trên mặt phẳng khác, bạn cần tuân thủ các nguyên tắc sau:

Trục thứ nhất của mặt phẳng đã chọn được lập trình với giá trị bán kính cung tròn

Trục thứ hai của mặt phẳng đã chọn được lập trình là vị trí góc của lỗ.

Bảng dưới đây sẽ minh họa cả ba khả năng. Chú ý, nếu không chọn mặt phẳng trong chương trình, hệ điều khiển sẽ sử dụng mặt phẳng mặc định XY theo G17.

Mã G	Mặt phẳng	Trục thứ nhất	Trục thứ hai
G17	XY	X = bán kính	Y = góc
G18	ZX	Z = bán kính	X = góc
G19	YZ	Y = bán kính	Z = góc

Hầu hết các ứng dụng tọa độ cực đều dùng mặt phẳng XY, được lập trình với lệnh G17.

Thứ tự gia công

Thứ tự gia công các lỗ có thể được điều khiển bằng cách đổi *dấu* của giá trị góc trong khi lệnh tọa độ cực có hiệu lực. Nếu giá trị góc được lập trình là số dương, thứ tự gia công sẽ ngược chiều kim đồng hồ, dựa trên vị trí 0^0. Bằng cách đổi sang số âm, thứ tự gia công sẽ thuận chiều kim đồng hồ.

Tính năng này là rất quan trọng để lập trình một cách hiệu quả, đặc biệt đối với nhiều loại sơ đồ các lỗ sắp xếp theo đường tròn. Ví dụ, khoan tâm hoặc khoan điểm có thể được lập trình hiệu quả với các giá trị góc dương (ngược chiều kim đồng hồ). Sự khởi đầu sẽ ở lỗ thứ nhất, và, sau khi thay dao, quá trình khoan có thể tiếp tục theo chiều ngược lại, bắt đầu từ lỗ cuối. Khi đó, mọi giá trị góc đều là số âm, với thứ tự thuận chiều kim đồng hồ cho dụng cụ cắt kế tiếp. Phương pháp này đòi hỏi khối lượng công việc khá lớn trong lập trình tiêu chuẩn, nếu không sử dụng hệ tọa độ cực. Ứng dụng tọa độ cực với lệnh G16 sẽ loại bỏ mọi chuyển động nhanh không cần thiết, do đó rút ngắn thời gian chu kỳ.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

27 PHAY BỀ MẶT

Phay bề mặt là nguyên công điều khiển chiều cao của chi tiết gia công. Đối với hầu hết các ứng dụng, phay bề mặt là nguyên công tương đối đơn giản, ít nhất là không đòi hỏi các chuyển động biên dạng phức tạp. Dụng cụ cắt dùng cho phay bề mặt thường là dao nhiều lưỡi cắt, được gọi là *dao phay mặt*, dù các dao phay mặt đầu cũng được dùng cho một số nguyên công phay bề mặt, thường trong các diện tích nhỏ. Các mặt trên được gia công với dao phay mặt thường vuông góc với trục dụng cụ cắt. Trong lập trình CNC, nguyên công phay bề mặt tương đối đơn giản, chỉ cần xét hai yếu tố quan trọng:

- Chọn đường kính dao
- Vị trí bắt đầu của dao liên quan đến chi tiết gia công

Ngoài ra, bạn cần có kinh nghiệm và kiến thức về các nguyên tắc phay bề mặt, chẳng hạn lựa chọn đúng loại dao, phân phối các đường cắt, công suất máy, và các vấn đề công nghệ liên quan. Một số nguyên lý cơ bản sẽ được đề cập trong chương này nhưng để hiểu rõ vấn đề bạn cần có kiến thức chuyên sâu về công nghệ chế tạo máy.

LỰA CHỌN DAO CẮT

Cũng như các nguyên công phay khác, phay bề mặt sử dụng dao cắt quay trong khi chi tiết đứng yên. Phay bề mặt thực chất là cắt bỏ một phần vật liệu với lượng dư gia công xác định từ bề mặt chi tiết, ở một hoặc vài mức độ sâu, theo một hoặc nhiều lần cắt gọt. Lập trình phay bề mặt tương đối đơn giản, thậm chí nhiều nhà lập trình chưa thật sự chú ý đến sự lựa chọn dao cắt thích hợp, đôi khi còn không xét đến các khả năng và các yêu cầu của máy phay.

Dao phay bề mặt là loại nhiều lưỡi cắt với các mảnh carbides có thể lắp lẫn. Dao phay mặt bằng thép gió ít được dùng trên máy CNC, chỉ thích hợp cho các diện tích nhỏ hoặc vị trí khó tiếp cận bằng phương pháp khác. Trong phay bề mặt, không phải mọi mảnh hợp kim (carbides) trên dụng cụ cắt đều đồng thời làm việc thực sự. Mỗi mảnh hợp kim chỉ cắt gọt trong một phần vòng quay. Đây có thể là yếu tố quan trọng khi thiết lập tuổi bền tối ưu cho dao phay bề mặt. Phay bề mặt đòi hỏi nguồn công suất đáng kể từ máy công cụ. Điều rất quan trọng là gá lắp chính xác các mảnh hợp kim (lưỡi cắt) trên thân dao.

Các tiêu chuẩn lựa chọn cơ bản

Dựa trên chi tiết gia công, lựa chọn dao phay bề mặt đòi hỏi xem xét các yếu tố:

- Điều kiện, tình trạng của máy CNC.
- Vật liệu của chi tiết gia công
- Phương pháp gá lắp
- Phương pháp gá lắp và tính toàn vẹn kẹp chặt
- Cấu tạo của dao cắt.
- Đường kính dao
- Hình dạng của mảnh chắp (hợp kim cứng)

Đường kính dao và dạng hình học của mảnh chắp có ảnh hưởng lớn đến quá trình lập trình thực tế, nhưng các yếu tố khác cũng rất quan trọng.

Đường kính dao phay mặt

Một trong các yếu tố quan trọng nhất đối với nguyên công phay bề mặt là sự lựa chọn kích cỡ dao cắt. Đối với phay một đường cắt, chiều rộng lý tưởng của dao cắt phải khoảng 1.3-1.6 lần chiều rộng chi tiết. Ví dụ, dao phay mặt ∅ 4.0 là thích hợp với đường cắt có chiều rộng 2.5 inch. Tỷ số 1.3-1.6 sẽ bảo đảm sự hình thành và thoát phoi ổn định.

Đối với nhiều đường cắt, cần chọn dao có đường kính tối đa có thể sử dụng, xét đến công suất máy, hình dạng dao và các mảnh hợp kim, độ cứng vững gá lắp, chiều sâu và chiều rộng của từng đường cắt, và các yếu tố khác liên quan đến chế độ gia công.

Mục đích cơ bản của phay bề mặt là gia công mặt chi tiết đến chiều cao xác định. Đối với kiểu gia công này, chọn đường kính dao hợp lý, thường có nghĩa là sử dụng dao đường kính tương đối lớn. Các kích cỡ trong khoảng 2-12 inch (50 – 300 mm) là bình thường, tùy theo máy và chi tiết gia công.

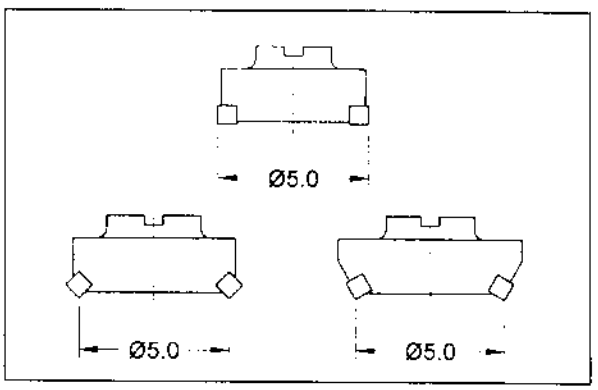
Một yếu tố quan trọng trong phay bề mặt là quan hệ giữa đường kính dao và chiều rộng toàn phần của đường cắt. Ví dụ, bạn chọn dao phay mặt ∅ 5.0 inch, đường kính *danh định* của dao này là 5.0 inch. Đường kính danh định

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

của dao cũng là *chiều rộng toàn phần của đường cắt*. Từ đường kính danh định không thể biết đường kính thân dao. Nói chung, kích cỡ thân dao là không cần thiết, trừ các trường hợp phay bề mặt gần các chướng ngại. Kích cỡ của thân dao có thể cản trở sự tiếp cận một vài phần trên chi tiết. Hình 27.1 minh họa các cấu hình dao phay bề mặt.



Hình 27.1. Đường kính danh định của một số cấu hình dao phay bề mặt

Dạng hình học của mảnh chắp

Kiến thức về thuật ngữ cơ bản của dao phay là cơ sở để hiểu các thuật ngữ được dùng trong lập trình. Các thuật ngữ cơ bản về dạng hình học của mảnh chắp trên thân dao phay được xác định theo thiết kế điều khiển mảnh chắp ăn vào vật liệu trong khi cắt gọt. Có *ba* loại cơ bản, dựa trên góc nghiêng của dao phay mặt (còn gọi là góc *nghiêng*):

- ❑ Dạng hình học dương ... một hoặc hai hàng
- ❑ Dạng hình học âm ... một hoặc hai hàng
- ❑ Dạng kết hợp cả hai ... dương/âm

Hiện có rất nhiều biến thể, ở đây chỉ trình bày các dạng cơ bản.

Dạng hình học dương

Dao cắt có *dạng hình học dương* yêu cầu công suất máy nhỏ hơn so với dao dạng hình học âm, do đó sẽ thích hợp hơn trên các máy CNC có định mức công suất thấp, thường là các máy nhỏ. Chúng có các đặc tính bẻ gãy phoi khá tốt thích hợp để gia công thép khi tải cắt gọt không quá cao. Mảnh chắp dương thường là một hàng do đó tính kinh tế không cao.

Dạng hình học âm

Dao phay có *dạng hình học âm* cung cấp độ bền rất cao của lưỡi cắt thường đòi hỏi công suất máy lớn và gá lắp mạnh. Sự hình thành phoi khi gia công thép hơi khó, chỉ thích hợp với gang. Ưu điểm chính của loại dao này là tính kinh tế do các mảnh ghép âm thường là hai phía, do đó có 8 lưỡi cắt trên mảnh ghép hình vuông, lắp vào các rãnh ở thân dao.

Dạng hình học âm hai hàng

Dạng hình học âm hai hàng chỉ có thể sử dụng với máy có đủ công suất, cả dao cắt và chi tiết gia công phải được gá lắp chắc chắn. Gang và một số vật liệu cứng thường được cắt gọt bằng loại dao này. Phoi có xu hướng tập trung về phía chi tiết gia công và khó thoát ra ngoài, có thể gây ra sự kẹt phoi. Mảnh chắp dương/âm có thể giải quyết vấn đề này.

Dạng hình học dương/âm

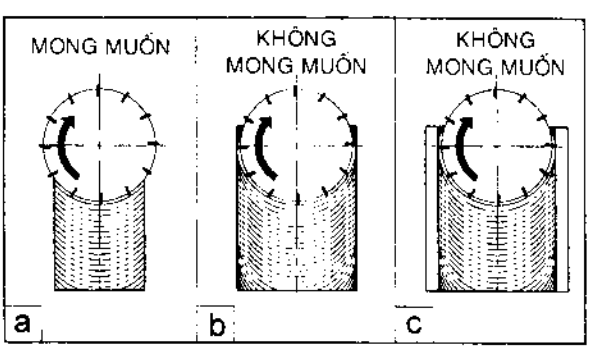
Dạng hình học dương/âm là tối ưu đối với hầu hết các nguyên công phay bề mặt thường dễ bị kẹt phoi. Thiết kế hình học dương/âm cung cấp độ bền của mảnh chắp âm và khả năng "cuộn" phoi thành hình xoắn của mảnh chắp dương. Loại dao này là thích hợp nhất khi phay toàn bộ chiều rộng các bề mặt chi tiết.

CÁC YẾU TỐ CẮT GỌT

Để lập trình chuyển động cắt cho dao phay bề mặt, điều quan trọng là hiểu cơ chế hoạt động của dao trong các điều kiện khác nhau. Ví dụ, trừ các loại dao phay bề mặt có thiết kế đặc biệt và dạng hình học thích hợp của mảnh chắp, hình dạng và chủng loại được sử dụng, bạn không nên phay bề mặt chi tiết có chiều rộng bằng hoặc chỉ hơi lớn hơn đường kính dao. Đường cắt toàn bộ chiều rộng có thể làm cho lưỡi cắt bị mòn nhanh và phoi "dính" vào mảnh chắp. Không chỉ mảnh chắp bị mòn nhanh, độ bóng bề mặt chi tiết cũng bị giảm. Trong trường hợp nghiêm trọng, mảnh chắp có thể bị hư, làm tăng chi phí gia công.

Hình 27.2 minh họa quan hệ mong muốn và không mong muốn giữa đường kính dao và chiều rộng chi tiết khi phay bề mặt.

Minh họa này chỉ cho thấy quan hệ giữa



Hình 27.2. Quan hệ giữa đường kính dao và chiều rộng chi tiết. Kích cỡ dao (a) là thích hợp.

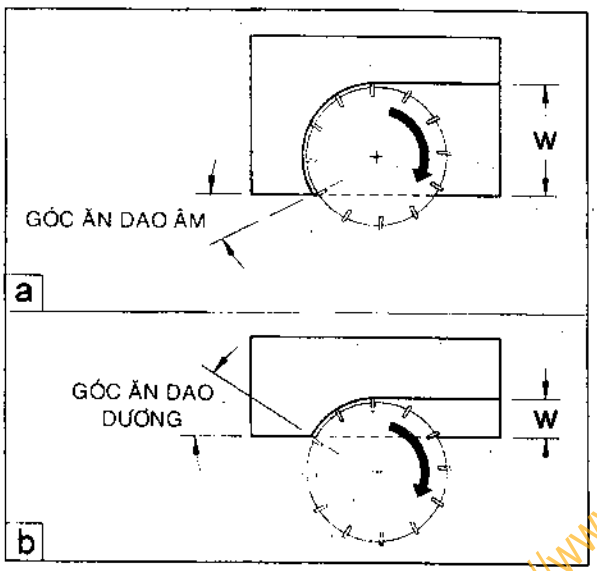
Nhiều tài liệu hay có tại : http://www.me-cad.com/

đường kính dao và chiều rộng chi tiết, không nêu rõ phương pháp dao "ăn" vào tại điểm đầu. Yếu tố quan trọng nhất đối với lập trình CNC cho dao phay bề mặt là *góc ăn dao*.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Góc ăn dao

Góc ăn dao khi phay bề mặt được xác định theo vị trí giữa đường tâm dao và cạnh của chi tiết. Nếu chi tiết có thể được phay với một đường cắt, bạn cần tránh tình huống vị trí đường tâm dao cắt tương hợp với đường tâm chi tiết. Vị trí *trung hòa* này sẽ gây ra sự vấp dao và bề mặt gia công có độ nhám cao. Bạn hãy để dao cắt *lệch* với đường tâm chi tiết, theo góc ăn dao *âm* hoặc góc ăn dao *dương* (Hình 27.3).



Hình 27.3. *Góc ăn dao vào chi tiết. W = chiều rộng đường cắt. (a) Tại đỉnh mảnh chắp bền nhất – góc ăn dao âm. (b) Tại đỉnh mảnh chắp yếu nhất – góc ăn dao dương*

Góc ăn dao trung hòa (không nêu trên Hình) có đường tâm dao trùng với cạnh chi tiết. Khi dao ăn vào chi tiết cần có lực xác định. Đối với góc ăn dao dương, lưỡi cắt yếu sẽ hấp thu hầu hết các lực cắt. Do lưỡi cắt là phần yếu nhất của mảnh chắp, góc dương có thể làm gãy hoặc mẻ lưỡi cắt. Nói chung, phương pháp này *ít* được dùng trong thực tế.

Góc ăn dao âm sẽ hấp thu lực cắt ở giữa, tại điểm bền nhất của mảnh chắp. Đây là phương pháp thông dụng, làm tăng tuổi bền của dao. Bạn cần duy trì đường tâm dao *trong* phạm vi diện tích bề mặt chi tiết, thay vì bên ngoài, sao cho mảnh chắp luôn luôn ăn vào chi tiết theo góc âm.

Mọi ví dụ nêu trên đều giả thuyết phoi là nguyên khối trong khi cắt gọt. Nếu dao chạy qua *không gian trống*, sự cắt gọt sẽ gián đoạn. Vấn đề *vào dao* và *thoát dao* khi sự cắt gọt gián đoạn sẽ làm thay đổi góc ăn dao.

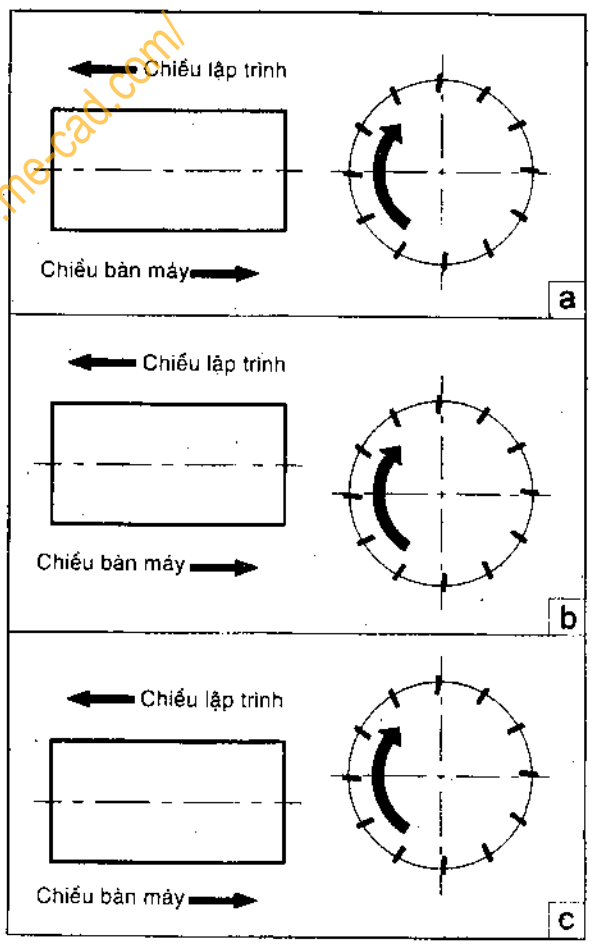
Chế độ phay

Trong nguyên công phay, yếu tố rất quan trọng là quan hệ giữa chiều cắt gọt được lập trình và chiều chuyển động của bàn máy. Yếu tố này gần như có tính quyết định, do đó sẽ được trình bày tương đối chi tiết.

Nói chung có *ba* chế độ khả dụng trong nguyên công phay:

- ❑ Chế độ phay trung hòa.
- ❑ Chế độ phay thuận.
- ❑ Chế độ phay ngược.

Chế độ phay trung hòa là tình huống dao cắt chạy theo đường tâm rãnh hoặc bề mặt, phay ngược theo một phía và phay thuận theo phía bên kia đường tâm đó. Chế độ phay *thuận* còn được gọi là phay *"lên"* và phay *ngược* là chế độ phay *"xuống"*. Các thuật ngữ *phay ngược* và *phay thuận* thường được dùng nhiều với phay



Hình 27.4. *Các chế độ phay bề mặt: (a) Chế độ phay trung hòa. (b) Chế độ phay "xuống" hoặc phay ngược. (c) Chế độ phay "lên" hoặc phay thuận*

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

chu vi hơn là phay bề mặt, nhưng đều áp dụng các nguyên tắc như nhau. Đối với đầu hết các đường cắt khi phay bề mặt, chế độ phay ngược có lẽ là lựa chọn tốt nhất.

Trên Hình 27.4, ví dụ (a) minh họa chế độ *cắt* trung hòa, ví dụ (b) là chế độ cắt *xuống* (chế độ phay *ngược*) và ví dụ (c) là chế độ cắt *lên* (chế độ phay *thuận*).

Số lượng mảnh chắp

Tùy theo kích thước phay bề mặt, dụng cụ cắt thông dụng là dao nhiều răng (nhiều lưỡi cắt). Quan hệ giữa số lượng mảnh chắp trên thân dao và đường kính hiệu dụng của dao được gọi là *mật độ dao cắt* hoặc *bước dao cắt*.

Dao phay bề mặt được chia làm ba nhóm dựa trên mật độ dao cắt:

- ❑ Mật độ thô ...bước thô
- ❑ Mật độ trung bình ...bước trung bình
- ❑ Mật độ mịn ...bước mịn

Nói chung, dao cắt bước thô thường là lựa chọn thích hợp. Càng nhiều lưỡi cắt đồng thời tham gia vào sự cắt gọt, đòi hỏi công suất máy càng cao. Bất kể mật độ dao cắt, điều quan trọng là có đủ các khoảng hở thoát phoi – phoi phải thoát dễ dàng để không bị kẹt dưới lưỡi cắt.

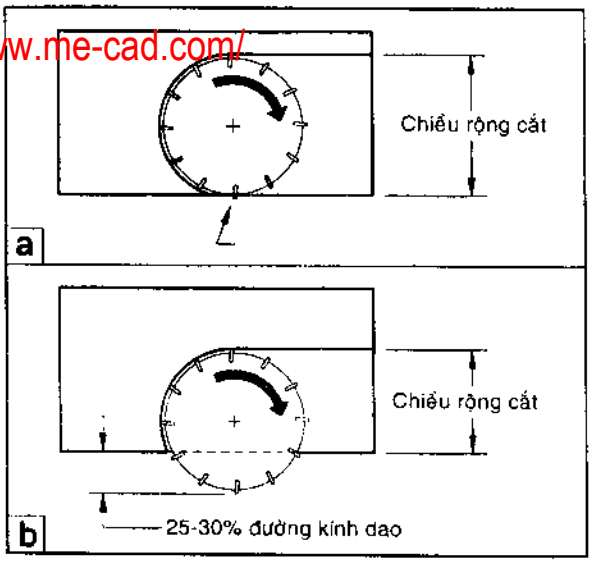
Tại mọi thời điểm cắt gọt, ít nhất một lưỡi cắt phải tiếp xúc với vật liệu, để tránh cắt gián đoạn. Sự cắt bị gián đoạn có thể làm hư hại dao và máy. Tình huống này có thể xảy ra nếu dùng dao đường kính lớn để phay chiều rộng nhỏ.

CÁC KỸ THUẬT LẬP TRÌNH

Tuy lập trình phay bề mặt là tương đối đơn giản, nhưng sẽ tốt hơn nếu chú ý đến một số yếu tố quan trọng. Do phay bề mặt thường bao quát diện tích cắt tương đối lớn, điều quan trọng là xem xét cẩn thận quỹ đạo thực của dao từ điểm đầu đến điểm cuối. Dưới đây là các điểm cần đánh giá đối với nguyên công phay bề mặt:

- ❑ Luôn luôn đưa dao cắt đến chiều sâu yêu cầu ở bên ngoài chi tiết
- ❑ Nếu có yêu cầu cao về độ bóng bề mặt, cần đối chiều dao ở bên ngoài chi tiết.
- ❑ Duy trì tâm dao bên trong diện tích chi tiết để có các điều kiện cắt gọt tốt hơn
- ❑ Chọn đường kính dao khoảng 1.5 lần chiều rộng đường cắt mong muốn.

Hình 27.5 minh họa tấm đơn giản dùng cho các ví dụ

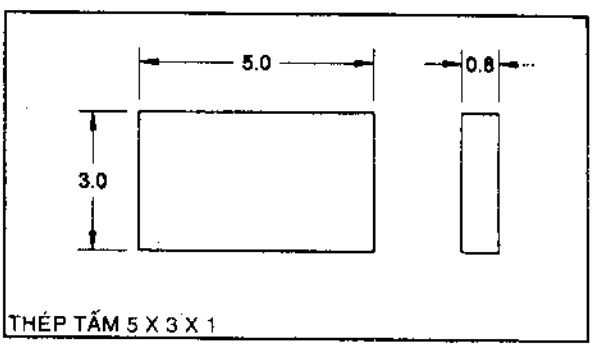


***Hình 27.5.** Chiều rộng đường cắt khi phay bề mặt – (b) phương pháp nên dùng*

Hình 27.5a minh họa chiều rộng *sai* và Hình 27.5 b là chiều rộng *thích hợp* khi phay bề mặt. Trong ví dụ (a), dao cắt ăn vào chi tiết với đường kính toàn phần, gây ra ma sát ở lưỡi cắt và làm giảm tuổi bền của dao. Ví dụ (b) chỉ giữ 2/3 đường kính dao cắt trên bề mặt gia công, cho phép tạo ra chiều rộng phoi cắt thích hợp, tạo góc ăn dao âm khi cắt gọt.

Phay bề mặt với một đường cắt

Đối với ví dụ lập trình thứ nhất về phay bề mặt, bạn sẽ dùng tấm 5 × 3 (dày 1 inch) được phay bề mặt dọc theo toàn bộ mặt trên đến chiều dày hoàn tất 0.800 inch (Hình 27.6).



***Hình 28.6.** Ví dụ về phay bề mặt một đường cắt – chương trình O2701*

Từ bản vẽ có thể thấy phay bề mặt sẽ xảy ra *dọc theo* chi tiết, do đó sẽ chọn chiều ngang theo trục X. Trước khi bắt đầu chương trình, cần chọn hai yếu tố chính:

- ❑ Đường kính dao phay bề mặt.
- ❑ Vị trí bắt đầu và kết thúc đường cắt.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

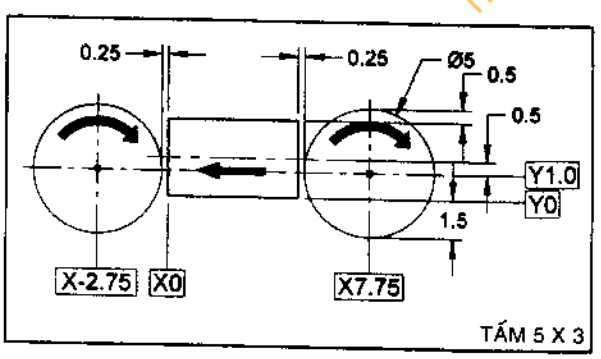
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Còn có các yếu tố khác cần xem xét, nhưng hai yếu tố nêu trên là quan trọng nhất.

Chi tiết chỉ rộng 3-inch, do đó cần chọn dao *rộng hơn* 3 inch. Tuy dao ∅ 4.0 dường như là lựa chọn đương nhiên, nhưng cần xét xem dao có đáp ứng các điều kiện nêu trên không. Đường kính dao phải khoảng 1.3 – 1.6 chiều rộng đường cắt. Trong trường hợp này, 3 × 1.3 = 3.9 và 3 × 1.6 = 4.8. Với dao phay bề mặt ∅ 4.0, điều đó có nghĩa là chỉ lớn hơn 1.33 lần. Khi xét yêu cầu dao cắt phải rộng hơn cả hai cạnh biên của chi tiết, chọn dao phay bề mặt đường kính *5 inch* sẽ tốt hơn.

Sau khi chọn đường kính dao, cần tập trung vào các vị trí *bắt đầu* và *kết thúc*. Vì các lý do an toàn, cần đưa dao đến chiều sâu để bắt đầu *cách xa* chi tiết. Do đã chọn đường cắt theo trục X (nằm ngang), vấn đề là từ trái sang phải hay ngược lại từ phải sang trái. Điều này thực sự không quá quan trọng, ngoại trừ chiều thoát phoi, do đó chọn chiều từ phải sang trái là tùy ý.

Điểm X0Y0 trên chi tiết là góc dưới bên trái. Để thiết lập vị trí X khởi đầu, bạn hãy xét chiều dài chi tiết 5.0 inch, bán kính dao (5/2 = 2.5) và khoảng hở (0.25). Vị trí khởi đầu trên trục X sẽ là tổng các giá trị này, X7.75. Đối với tính toán vị trí bắt đầu trên trục Y, bạn hãy xét phần dư ra trên cả hai cạnh và chọn chế độ phay ngược. Trong thực tế, sự phay ngược sẽ được kết hợp với một phần phay thuận, điều này là bình thường trong các nguyên công phay bề mặt. Hình 27.7 minh họa vị trí bắt đầu của dao cắt X7.5Y1.0, và vị trí kết thúc tại X-2.75Y1.0, cùng với các chi tiết tính toán.



Hình 27.7. *Các vị trí phay bề mặt trong ví dụ phay một đường cắt.*

Vị trí Y1.0 dựa trên yêu cầu có 1/4 - 1/3 đường kính dao dư ra trên cạnh chi tiết, để đạt được góc ăn dao tối ưu. Phần dư 1.5 inch là 30% đường kính dao cắt, vị trí Y lập trình được thiết lập tại Y1.0.

Với các dữ liệu nêu trên, có thể bắt đầu chương trình phay bề mặt một đường cắt, với mặt trên của chi tiết là zero chương trình (Z0). Chương trình O2701 chỉ sử dụng một đường cắt.

```
O2701
(SINGLE FACE MILLING CUT)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X7.75 Y1.0 S344 M03
N4 G43 Z1.0 H01
N5 G01 Z-0.2 F50.0 M08
N6 X-2.75 F21.0
N7 G00 Z1.0 M09
N8 G28 X-2.75 Y1.0 Z1.0
N9 M30
%
```

Tốc độ trục chính và vận tốc dựa trên tốc độ bề mặt 450 ft/min, 0.006"/răng và 8 mảnh chắp, là các giá trị hợp lý. Chú ý sự tiếp cận Z trong block N4. Mặc dầu dao ở trong khoảng trống, nhưng để bảo đảm an toàn, chuyển động nhanh được phân chia giữa các block N4 và N5. Để tăng độ tin cậy, có thể chọn chuyển động nhanh trực tiếp Z-0.2. Ví dụ này sử dụng zero chương trình phía trên chi tiết *chưa gia công*.

Phay nhiều đường cắt

Các nguyên tắc tổng quát áp dụng cho phay bề mặt một đường cắt cũng đúng với phay nhiều đường cắt. Do đường kính dao phay bề mặt thường quá nhỏ để cắt toàn bộ lượng dư gia công trong một lần cắt trên diện tích lớn, cần lập trình nhiều đường cắt có cùng chiều sâu cắt.

Có vài phương pháp phay bề mặt trên diện tích lớn, mỗi phương pháp đều có phạm vi ứng dụng riêng. Các phương pháp phổ biến bao gồm *cắt một chiều nhiều đường* và *cắt hai chiều nhiều đường*, với cùng chiều sâu Z.

Cắt một chiều nhiều đường bắt đầu từ cùng một vị trí trên một trục nhưng thay đổi vị trí trên trục còn lại, phía trên chi tiết. Đây là phương pháp phay bề mặt phổ biến, nhưng không hiệu quả do có nhiều chuyển động nhanh trả dao về vị trí ban đầu.

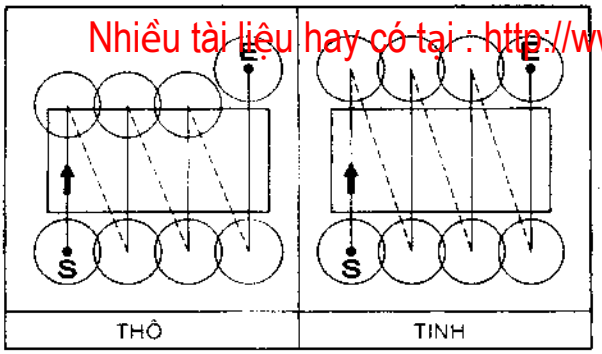
Cắt hai chiều nhiều đường, thường được gọi là cắt theo quỹ đạo gấp khúc, cũng được sử dụng thường xuyên; hiệu quả hơn cắt một chiều, nhưng dao phay phải chuyển từ phay ngược sang phay thuận và ngược lại. Phương pháp này thường chỉ dùng cho một số ứng dụng.

Trong hai minh họa kế tiếp, Hình 27.8 là sơ đồ phay một chiều và Hình 27.9 là sơ đồ phay hai chiều.

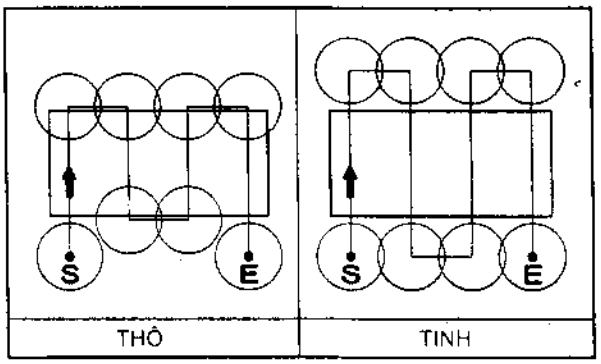
Bạn hãy so sánh các chuyển động XY của hai phương pháp này. Ngoài ra, sự khác biệt về quỹ đạo dao (vị trí dụng cụ cắt) giữa gia công thô và tinh cũng được nêu rõ. Chiều cắt gọt có

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 27.8. *Phương pháp một chiều nhiều đường cắt áp dụng cho gia công thô và gia công tinh.*

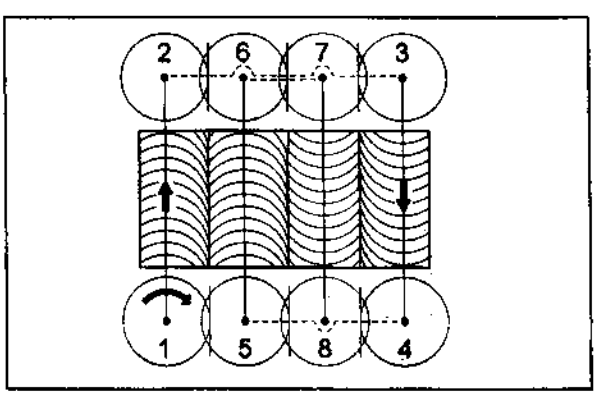


Hình 27.9. *Phương pháp hai chiều nhiều đường cắt áp dụng cho gia công thô và gia công tinh.*

thể theo trục X hoặc trục Y, nhưng các nguyên tắc về chuyển động cắt là như nhau.

Bạn hãy lưu ý vị trí bắt đầu (S) và vị trí kết thúc (E) trong hai minh họa, được biểu thị bằng chấm đen ở tâm dao cắt. Bất kể phương pháp cắt gọt, dao phay bề mặt luôn luôn ở khoảng trống tại các điểm bắt đầu và kết thúc cắt gọt, chủ yếu để an toàn.

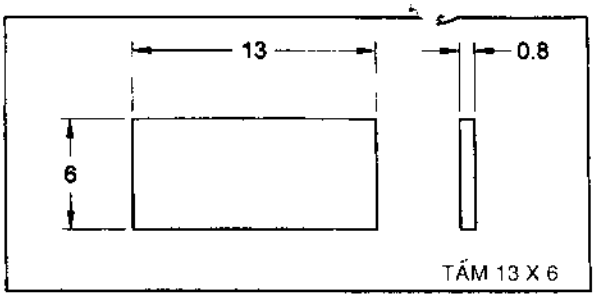
Phương pháp phay tương đối hiệu quả là chỉ cắt trong một chế độ, thường là phay ngược. Phương pháp này có dạng chuyển động tròn hoặc xoắn (dọc theo các trục XY) và là phương pháp được dùng phổ biến, kết hợp hai phương pháp nêu trên (Hình 27.10).



Hình 27.10. *Quỹ đạo dao theo chế độ phay ngược, áp dụng cho phương pháp một chiều.*

Minh họa này nêu rõ *thứ tự* và *chiều* của mọi chuyển động dao riêng rẽ. Ý tưởng là thực hiện từng đường cắt có chiều rộng gần như nhau, với khoảng 2/3 đường kính dao vào thời điểm bất kỳ, luôn luôn trong chế độ phay ngược.

Ví dụ lập trình nhiều đường cắt dựa theo bản vẽ trên Hình 27.11.

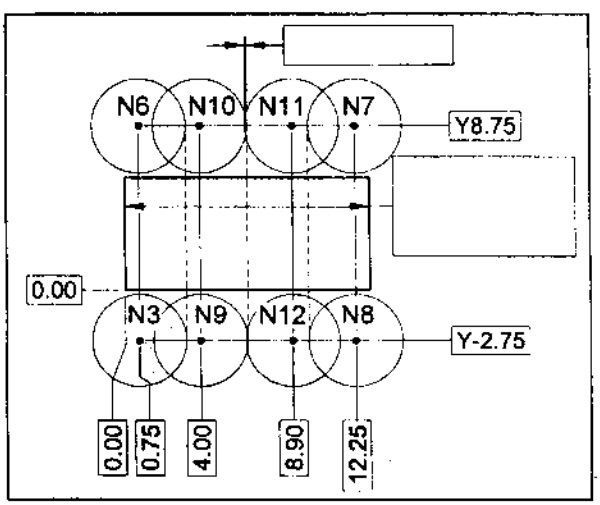


Hình 27.11. *Ví dụ phay nhiều đường cắt – chương trình O2702.*

```
O2702
(MULTIPLE FACE MILLING CUTS)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0.75 Y-2.75 S344 M03   (POS 1)
N4 G43 Z1.0 H01
N5 G01 Z-0.2 F50.0 M08
N6 Y8.75 F21.0                          (POS 2)
N7 G00 X12.25                           (POS 3)
N8 G01 Y-2.75                           (POS 4)
N9 G00 X4.0                             (POS 5)
N10 G01 Y8.75                           (POS 6)
N11 G00 X8.9                (POS 7 - 0.1 OVERLAP)
N12 G01 Y-2.75                          (POS 8 - END)
N13 G00 Z1.0 M09
N14 G28 X8.75 Y-2.75 Z1.0
N15 M30
%
```

Trong chương trình O2702, mọi block liên quan đều ghi vị trí dao tương ứng các số trên Hình 27.10.

Chiều rộng chi tiết 13 inch được chia thành bốn đường cắt bằng nhau, rộng 3.25 inch, hơi



Hình 27.12. *Chi tiết phay nhiều đường trong chương trình O2702*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

nhỏ hơn 2) dao cắt Ø 5.0. Các khoảng hở 0.25 inch tương tự ví dụ phay một đường cắt. Sự sai lệch so với chuẩn là chuyển động đến vị trí 7 trên Hình 27.10 và block N11 trong chương trình. Đường cắt cuối cùng là từ vị trí 7 đến vị trí 8. Để tăng độ bóng bề mặt, đường cắt này sẽ chồng chập ở X9.0 với 0.100 giá trị lập trình X8.9. Hình 27.12 nêu rõ sơ đồ của chương trình O2702, gồm cả các số block.

Một số ví dụ này có thể được thực hiện theo cách ngắn hơn, dọc theo trục X, chương trình sẽ gọn hơn. Tuy nhiên, với mục đích minh họa, sử dụng trục Y sẽ thuận tiện hơn.

SỬ DỤNG BÙ VỊ TRÍ

Trong cả hai ví dụ nêu trên đã tính toán vị trí XY khởi đầu của dao phay, xét theo đường kính và khoảng hở thích hợp. Để sử dụng chương trình O2701 làm ví dụ, điểm khởi đầu là X7.75 Y1.0. Chi tiết là 5.0 inch, cộng khoảng hở 0.25, cộng bán kính dao 2.5 inch, tổng số là X7.75 giá trị tuyệt đối của tâm dao cắt. Nhược điểm lớn của phương pháp này sẽ xuất hiện khi sử dụng dao phay bề mặt có *đường kính khác* với đường kính trong chương trình. Sự thay đổi dao vào phút cuối trên máy phay có thể gây ra các vấn đề. Hoặc khoảng hở *quá lớn* (nếu dao mới nhỏ hơn) hoặc *không đủ* khoảng hở nếu dao mới có đường kính lớn hơn dao cũ. Dưới đây là biện pháp giải quyết vấn đề này.

Giải pháp là sử dụng tính năng *bù vị trí* "lạc hậu" trên hệ thống điều khiển, đã trình bày trong Chương 16. Có lẽ đây là ứng dụng thực tiễn duy nhất của tính năng bù vị trí trên các trung tâm gia công CNC hiện đại.

Bạn hãy xem lại ví dụ O2701, các Hình 27.6 và 28.7. Các minh họa này nêu rõ cần phay bề mặt (một đường cắt) tấm 5 × 3, sử dụng dao phay bề mặt ∅5 inch. Để tuân thủ các nguyên tắc an toàn trong gia công, dao phay phải được định vị ở khoảng trống, cách xa chi tiết. Để các lưỡi cắt cách chi tiết 1/4 inch, khoảng hở 0.25 inch được cộng với *bán kính dao phay bề mặt*, là 2.5 inch, để đạt đến vị trí khởi đầu thực sự của dao cắt.

Trong chương trình phay bề mặt, tình huống này sẽ xảy ra theo một trong hai dạng dưới đây:

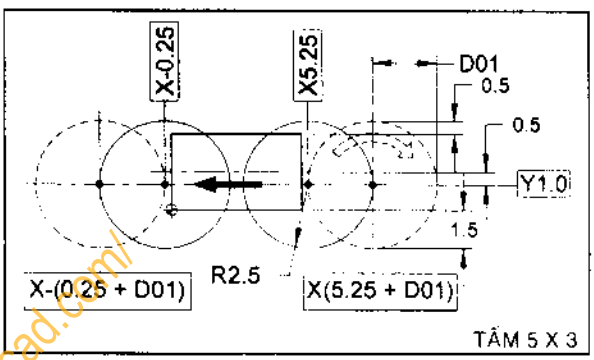
- ❑ Bán kính dao phay được lập trình sử dụng các giá trị thực
- ❑ Sử dụng phương pháp vị trí

Trong trường hợp thứ nhất, chương trình O2701 sẽ có nội dung như sau:

```
O2701
(SINGLE FACE MILLING CUT - NO COMPENSATION)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X7.75 Y1.0 S344 M03
N4 G43 Z1.0 H01
N5 G01 Z-0.2 F50.0 M08
N6 X-2.75 F21.0
N7 G00 Z1.0 M09
N8 G28 X-2.75 Y1.0 Z1.0
N9 M30
%
```

Block N3 dịch chuyển dao phay đến vị trí khởi đầu thực sự đã tính toán của đường cắt. Trong block N6, đường cắt hoàn tất – tại vị trí kết thúc đã tính toán trước. Chương trình O2703 sử dụng bù vị trí cũng tương tự, nhưng có một số khác biệt rõ rệt.

Bạn hãy so sánh chương trình gốc O2701 với chương trình mới O2703, sử dụng tính năng bù vị trí (Hình 27.13).



Hình 27.13. *Ví dụ về bù vị trí áp dụng cho phay bề mặt – chương trình O2703*

```
O2703
(SINGLE FACE MILLING CUT)
(USING POSITION COMPENSATION)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X8.0 Y1.0 S344 M03
N4 G43 Z1.0 H01
N5 G46 X5.25 D01
N6 G01 Z-0.2 F50.0 M08
N7 G47 X-0.25 F21.0
N8 G00 Z1.0 M09
N9 G91 G28 X0 Y0 Z0
N10 M30
%
```

Khi so sánh, bạn hãy chú ý các khác biệt chính trong block N3 (giá trị X mới), trong block N5 (bù G46), và trong block N7 (bù vị trí G47). Tình huống này sẽ trở nên rõ ràng khi được đánh giá chi tiết hơn.

Block N3 chứa vị trí X với giá trị X 8.0. Đây là vị trí khởi đầu. Do kế hoạch là áp dụng bù vị trí G46, dao cần ở vị trí có giá trị *lớn hơn* giá trị mong đợi khi sự bù vị trí hoàn tất. Do đó, X8.0 là giá trị tùy ý. Chú ý, nếu lệnh bù G54 được chọn, vị trí khởi đầu phải có giá trị *nhỏ hơn* giá trị mong đợi khi hoàn tất sự bù. Điều này là do bù vị trí luôn luôn liên quan đến *chiều lập trình*.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Block N5 được đưa thêm vào chương trình O2703, chứa lệnh bù vị trí G46, là sự giảm theo chiều lập trình với giá trị bù được ghi trong thanh ghi của bộ bù D01. Chú ý, giá trị tọa độ lập trình X5.25, là tổng chiều dài chi tiết (5.0 inch) và khoảng hở được chọn (0.250). Bán kính dao phay là hoàn toàn không cần thiết trong chương trình. Ưu điểm chính của phương pháp này là, trong phạm vị hợp lý, các tọa độ lập trình sẽ không thay đổi, dù có thay đổi về đường kính dao. Ví dụ, nếu sử dụng dao phay bề mặt Ø3.5 inch, sự gia công có thể rất tốt, nhưng cần thay đổi vị trí khởi đầu. Trong trường hợp này, giá trị đã lưu của D01 sẽ là 1.75 và khối N5 vẫn chứa X5.25. Hệ thống CNC sẽ hoạt động bình thường. Block cuối cùng cần xem xét là N7, block này chứa lệnh bù vị trí G47. Giá trị X là tương đương với khoảng hở đã chọn X-0.25. Lệnh G47 có nghĩa là kéo dài gấp đôi giá trị bù dọc theo chiều dài lập trình. Điều này là cần thiết do nhu cầu ở vị trí khởi đầu và kết thúc đường cắt. Ngoài ra, bạn cần lưu ý vị trí ban đầu và vị trí khởi đầu được bù *không thể như nhau*, nếu chúng trùng nhau sẽ không có sự bù vị trí. Với tính sáng tạo, phay bề mặt có thể được lập trình rất hiệu quả sử dụng tính năng bù vị trí.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

28 NỘI SUY ĐƯỜNG TRÒN

Trong đa số các ứng dụng lập trình CNC, chỉ hai kiểu chuyển động dụng cụ cắt liên quan đến biên dạng. Một là *Nội suy Tuyến tính*, đã được đề cập ở các chương trước và thứ hai là *Nội suy Đường tròn*, được trình bày trong chương này, điều khiển dao theo cung tròn tương tự với phương pháp lập trình quỹ đạo dao tuyến tính. Chúng thường được sử dụng trong gia công biên dạng trên các trung tâm gia công CNC đứng và ngang, trên các máy tiện, và nhiều máy CNC khác, gồm máy phay, máy cắt bằng laser, gia công tia lửa điện EDM, ...

Nội suy đường tròn được dùng để lập trình các cung hoặc đường tròn trong các ứng dụng gia công bán kính trong và ngoài, các hốc tròn, hình cầu hoặc conic, rãnh, cắt góc, cắt gọt xoắn, doa ngược ... Thiết bị CNC sẽ nội suy cung tròn với độ chính xác rất cao nếu được cung cấp đủ thông tin cần thiết trong chương trình.

ĐỊNH DẠNG LẬP TRÌNH

Định dạng lập trình đối với quỹ đạo dao nội suy đường tròn phải chứa nhiều tham số, nếu không có chúng, nhiệm vụ cắt gọt theo cung tròn sẽ không khả thi. Các tham số quan trọng bao gồm:

- Chiều cắt gọt cung (thuận – CW, hoặc ngược – CCW, chiều kim đồng hồ)
- Các điểm bắt đầu và kết thúc cung
- Tâm cung và giá trị bán kính.

Tốc độ cắt cũng phải có hiệu lực, sẽ được trình bày chi tiết trong chương này. Các mã G chế độ đặc biệt được dùng cho lập trình chuyển động tròn và các tham số liên quan với bán kính cũng cần được chuyên biệt.

Chiều cắt gọt theo cung

Dao cắt chuyển động trên cung tròn theo một trong hai hướng – *thuận chiều (CW)* hoặc *ngược chiều (CCW)* kim đồng hồ theo quy ước. Trên hầu hết các máy, chiều chuyển động được xác định bằng cách chiếu vuông góc đến mặt phẳng chứa chuyển động tròn được lập trình. Chuyển động *từ* trục đứng của mặt phẳng *hướng đến* trục ngang là thuận chiều, chuyển động đảo lại là ngược chiều kim đồng hồ. Quy ước này là từ *toán học*, đôi khi có thể không tương hợp với định hướng của các trục máy. Chương 30 sẽ trình bày chi tiết sự gia công trên các mặt phẳng, chương này chỉ đề cập khái quát.

Trong định dạng chương trình thông dụng, lệnh thứ nhất của block chuyển động cắt gọt theo đường tròn là *chiều cắt gọt*. Chiều này xác định chuyển động dao cắt dọc theo cung được lập trình, thuận hoặc ngược chiều đồng đồ. Chiều chuyển động dọc theo cung được lập trình sử dụng các lệnh chuẩn bị trong block đó.

Block nội suy đường tròn

Có hai lệnh chuẩn bị liên quan với lập trình chiều cung tròn

G02	Chuyển động thuận chiều CW
G03	Chuyển động ngược chiều CCW

Cả hai lệch G02 và G03 đều có tính chế độ, do đó chúng giữ nguyên hiệu lực cho đến cuối chương trình hoặc bị xóa bằng lệnh khác từ cùng nhóm mã G, thường là lệnh chuyển động.

Các lệnh chuẩn bị G02 và G03 là từ khóa được dùng trong lập trình để thiết lập chế độ nội suy đường tròn. Các lệnh tọa độ tiếp sau G02 hoặc G03 luôn luôn được gán trong mặt phẳng đã chọn. Mặt phẳng này dựa trên các phối hợp trục khả dụng XY, ZX, và ZY cho các ứng dụng phay hoặc tương tự. Nói chung, không cần chọn mặt phẳng trên máy tiện, dù một số bộ điều khiển biểu thị mặt phẳng ZX là G18.

Sự lựa chọn mặt phẳng, phối hợp các tham số chuyển động tròn và chiều cung cắt gọt xác định *điểm cuối của cung*, giá trị R là *bán kính của cung tròn*. Các vector tâm cung cũng khả dụng, nếu nhà lập trình yêu cầu.

Khi lệnh G02 hoặc G03 được kích hoạt từ chương trình CNC, lệch chuyển động dao hiện hành bất kỳ sẽ bị xóa một cách tự động. Chuyển động bị xóa này thường là G01, G00, hoặc lệnh chu kỳ. Mọi chuyển động quỹ đạo dao theo đường tròn đều phải được lập trình với tốc độ cắt có hiệu lực, áp dụng đúng các nguyên tắc cơ bản nhu đối với nội suy tuyến tính. Điều đó có nghĩa là tốc độ cắt F phải được lập trình *trước* hoặc *trong* block chuyển động cắt. Nếu tốc độ cắt không được chuyên biệt trong block chuyển động tròn, hệ điều khiển sẽ

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tự động tìm tốc độ cắt được lập trình *gần nhất*. Nếu hoàn toàn không có tốc độ cắt hoạt động, nhiều bộ điều khiển thường đưa ra thông báo lỗi. Tốc độ cắt có thể được chuyên biệt theo một trong hai cách. Trực tiếp, bên trong block cắt gọt theo cung tròn, hoặc gián tiếp, bằng cách giả thiết tốc độ cắt hoạt động gần nhất. Chuyển động tròn trong chế độ nhanh là không thể. Ngoài ra, cũng không thể chuyển động tròn đồng thời theo ba trục. Chi tiết về vấn đề này, bạn có thể xem Chương 44 trình bày về phay theo đường cong xoắn.

Trên đa số các bộ điều khiển cũ, địa chỉ bán kính trực tiếp R không thể chuyên biệt, thay vào đó phải sử dụng các vector cung I, J, và K:

```
G02 X.. Y.. I.. J..     Chương trình phay - CW
G02 X.. Y.. I.. K..     Chương trình tiện - CW
G03 X.. Y.. I.. J..     Chương trình phay - CCW
G03 X.. Y.. I.. K..     Chương trình tiện - CW
```

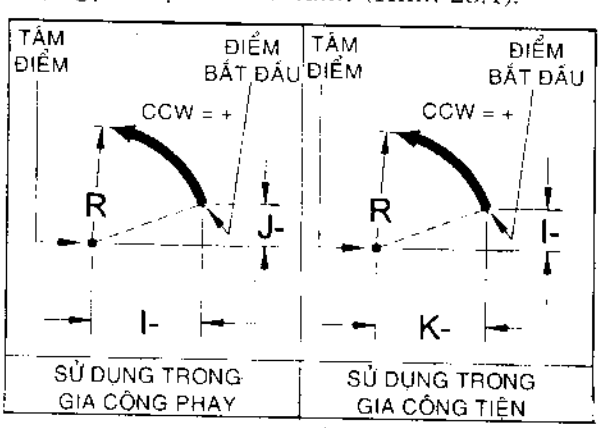
Các hệ điều khiển hỗ trợ sự gán bán kính cung theo địa chỉ R *cũng* sẽ chấp nhận các vector I, J, K, nhưng điều ngược lại là *không* đúng. Nếu *cả* vector cung IJK và bán kính R đều được lập trình trong một block, giá trị bán kính sẽ chiếm ưu tiên, bất kể *thứ tự của chúng*:

```
G02 (G03) X.. Y.. R.. I.. J..
G02 (G03) X.. Y.. I.. J.. R..
```

Các bộ điều khiển *chỉ* chấp nhận các vector I J K sẽ đưa ra thông báo lỗ nếu block nội suy đường tròn chứa địa chỉ R (địa chỉ lạ).

Các điểm bắt đầu và kết thúc cung tròn

Điểm bắt đầu của một cung là nơi *bắt đầu* nội suy đường tròn, xác định theo chiều cắt gọt. Điểm này *phải* ở trên cung tròn và có thể là điểm tiếp tuyến hoặc giao điểm, kết quả của bán kính hòa nhịp hoặc bán kính riêng phần. Lệnh trong block chứa điểm bắt đầu đôi khi được gọi là *lệnh khởi hành* (Hình 28.1).



Hình 28.1. Tâm điểm và điểm bắt đầu của cung tròn.

Điểm bắt đầu cung tròn luôn luôn liên quan với chiều chuyển động cắt và được biểu thị *trước* chuyển động tròn.

Điểm bắt đầu của một cung là vị trí cuối cùng của dao cắt trước lệnh nội suy đường tròn.

Ví dụ:

```
N66 G01 X5.75 Y7.5
N67 G03 X4.625 Y8.625 R1.125
N68 G01 X.. Y..
```

Trong ví dụ này, block N66 biểu thị *kết thúc* biên dạng, chẳng hạn chuyển động tuyến tính, đồng thời là *sự bắt đầu* của cung kế tiếp. Trong block N67, cung được gia công, các tọa độ biểu thị kết thúc cung và điểm bắt đầu của phần tử kế tiếp. Block cuối trong ví dụ là N68, biểu thị điểm cuối của phần từ đã bắt đầu từ cung trước đó. *Điểm cuối* của cung này là điểm tọa độ của hai trục bất kỳ, nơi *kết thúc* chuyển động tròn. Điểm đó đôi khi được gọi là vị trí *đích*.

Bán kính và tâm cung

Bán kính cung có thể được gán với địa chỉ R hoặc với các vector tâm cung I, J và K. Địa chỉ R cho phép lập trình bán kính cung trực tiếp, các vector tâm cung I, J, K được dùng để xác định ví trí thực của tâm cung. Hầu hết các hệ điều khiển hiện đại đều hỗ trợ nhập địa chỉ R, các bộ điều khiển cũ chỉ yêu cầu vector tâm cung. Định dạng lập trình cơ bản chỉ khác biệt đôi chút giữa các hệ thống phay và tiện, đặc biệt là đối với địa chỉ R:

```
G02 X.. Y.. R...        Chương trình phay - CW
G02 X.. Z.. R...        Chương trình tiện - CW
G03 X.. Y.. R...        Chương trình phay - CCW
G03 X.. Z.. R...        Chương trình tiện - CCW
```

Khái niệm quan trọng cần hiểu là chiều cắt gọt CW hoặc CCW không lệ thuộc vào bán kính hoặc tâm cung. Hệ điều khiển cần thông tin nhiều hơn là chiều và điểm đích để cắt gọt theo cung mong muốn. Thông tin bổ sung này phải có định nghĩa xác định cung được lập trình với bán kính *duy nhất*.

Bán kính duy nhất này đạt được bằng cách lập trình địa chỉ R khi nhập bán kính trực tiếp, hoặc sử dụng các vector tâm cung I J K. Địa chỉ R là bán kính thực của quỹ đạo dao, thường được lấy từ bản vẽ chi tiết.

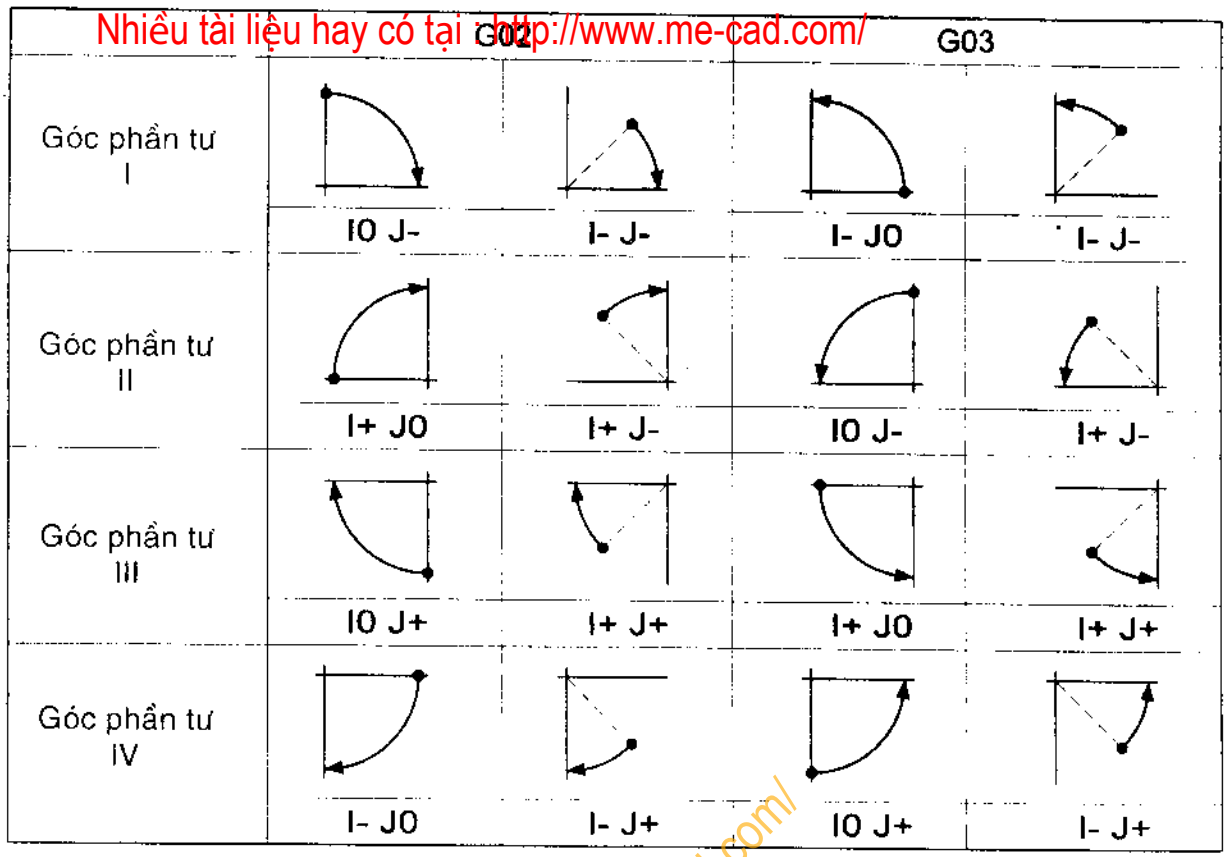
Vector tâm cung

Hình 28.2 minh họa dấu của các vector I và J trong mọi định hướng khả dĩ. Trong các mặt phẳng khác nhau, sẽ sử dụng các cặp vector tương ứng, nhưng logic sử dụng chúng không thay đổi.

Các vecotr I, J, và K được sử dụng theo các định nghĩa dưới đây (chỉ minh họa I và J).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 28.2. *Các vector cung I và J và dấu của chúng trong các góc phần khi (mặt phẳng XY)*

Vector tâm cung *I* là khoảng cách, với chiều chuyên biệt, đo từ điểm bắt đầu của cung đến tâm cung, song song với trục X.

Vector tâm cung *J* là khoảng cách, với chiều chuyên biệt, đo từ điểm bắt đầu của cung đến tâm cung, song song với trục Y.

Vector tâm cung *K* là khoảng cách, với chiều dài chuyên biệt, đo từ điểm bắt đầu của cung đến tâm cung, song song với trục Z.

Khoảng cách giữa điểm bắt đầu và tâm điểm của cung (các vector IJK) hầu như luôn luôn đo theo khoảng cách *số gia* giữa hai điểm đó. Một số hệ điều khiển, chẳng hạn Cincinnati, sử dụng chế độ tuyệt đối để xác định tâm cung. Trong các trường hợp đó, tâm cung được lập trình theo giá trị *tuyệt đối* từ *zero chương trình*. Bạn cần biết các hệ điều khiển trong xưởng cơ khí để xử lý những tình huống đó.

Sự thiếu tiêu chuẩn trong lĩnh vực này gây ra sự khác biệt lớn trong định dạng lập trình, do đó cần cẩn thận để tránh khả năng xảy ra lỗi. Lỗi dễ xảy ra trong các trường hợp cả hai kiểu bộ điều khiển đều được dùng trong xưởng cơ khí. Không có *tính tương thích* giữa các chương trình sử dụng kích thước tuyệt đối và kích thước số gia đối với tâm cung.

Chiều chuyên biệt chỉ áp dụng cho chế độ số gia của tâm cung. Đây là định nghĩa vị trí tương đối của tâm cung từ điểm bắt đầu, được lập trình với dấu chiều – nếu không có dấu, sẽ giả thiết là chiều dương, dấu trừ (-) là chiều âm và phải được ghi rõ. Các cung sử dụng định nghĩa tâm tuyệt đối sẽ tuân theo các nguyên tắc tiêu chuẩn và lập kích thước tuyệt đối.

Cung trong mặt phẳng

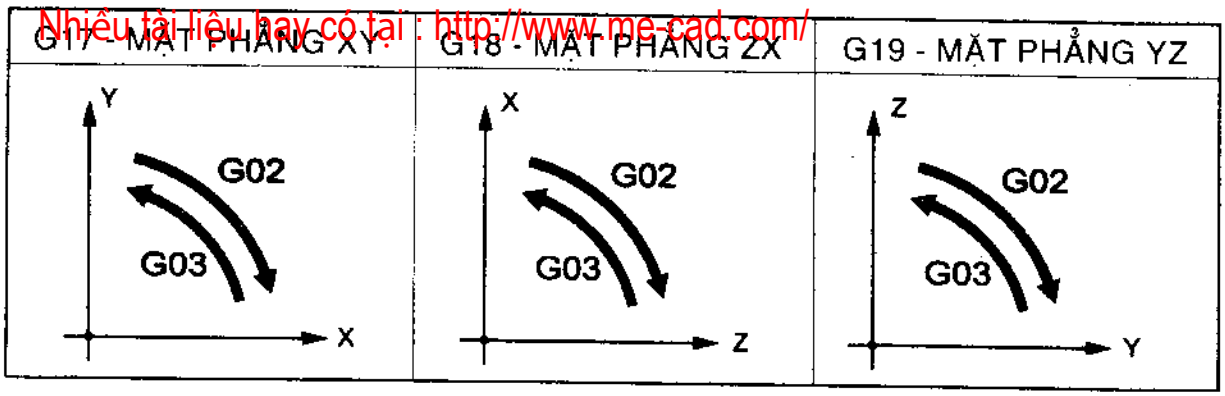
Đối với trung tâm gia công, có thể lập trình cung trên một trong ba mặt phẳng (Hình 28.3). Cần sử dụng đúng các vector cung tương ứng từng mặt phẳng:

```
G17 G02 (G03) X.. Y.. R..        (Hoặc I.. J..)
G18 G02 (G03) X.. Y.. R..        (Hoặc I.. K..)
G19 G02 (G03) Y.. Z.. R..        (Hoặc J.. K..)
```

Nếu mặt phẳng lập trình không thẳng hàng với các trục máy, hoặc nếu các trục trong chương trình được chọn không có sự gán mặt phẳng, chuyển động tròn sẽ xảy ra theo *sự lựa chọn trục trong chương trình*. Bạn hãy luôn luôn cảnh giác khả năng bỏ sót (hoặc quên) chuyển động trục theo chế độ. Phương pháp an toàn nhất để tránh vấn đề nguy hiểm này là tuân theo quy tắc – *không dựa vào các giá trị chế độ*.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 28.3. Chiều cắt gọt cung trong ba mặt phẳng – hướng của các trục dựa trên mặt phẳng toán học, không phải mặt phẳng máy.

Trong các mặt phẳng không tiêu chuẩn, block chương trình nội suy đường tròn phải có các chuyên biệt về *cả hai trục*, cả hai vector cung hoặc giá trị R. Block đó là *đầy đủ* và luôn luôn được thực thi trên cơ sở ưu tiên gán các trục. Phương pháp này thường được dùng để chọn mặt phẳng đã xác định sẵn. Kể cả nếu sự gán mặt phẳng này không chính xác, chuyển động dao vẫn *chính xác* theo mặt phẳng đã xác định trước đó.

LẬP TRÌNH BÁN KÍNH

Lập trình quỹ đạo cung là rất phổ biến. Theo định nghĩa, cung chỉ là *một phần* đường tròn do đó có nhiều cách lập trình cung. Nếu cung là 360^0, cần lập trình vị trí bắt đầu cắt gọt trùng với vị trí kết thúc. Trong trường hợp đó, kết quả sẽ là đường tròn. Nếu chỉ một phần đường tròn được lập trình, chỉ cần lập trình bán kính. Có hai kiểu bán kính được dùng trong lập trình CNC:

- Bán kính hòa hợp
- Bán kính riêng

Từng bán kính này có thể được lập trình theo chiều CW hoặc CCW, có thể là bán kính ngoài hoặc trong, và theo chiều bất kỳ của dụng cụ cắt.

Bán kính hòa hợp

Điểm tiếp tuyến giữa cung và phần tử kế cận tạo thành bán kính *hòa hợp*. Đây là bán kính tiếp xúc giữa *đường và cung*, *cung và đường*, hoặc giữa *hai cung*. Bán kính hòa hợp tạo ra sự chuyển tiếp đồng đều giữa hai phần tử biên dạng. Điểm tiếp tuyến là điểm tiếp xúc duy nhất giữa hai phần tử đó.

Dạng đơn giản nhất của bán kính hòa hợp là giữa hai đường vuông góc và song song với các trục máy. Tính toán các điểm bắt đầu và kết thúc chỉ cần vài phép cộng và trừ. Tính toán sẽ phức tạp hơn nếu hai đường này không vuông góc với nhau. Khi đó, cần dùng các hàm lượng giác để tính điểm đầu, điểm cuối, hoặc cả hai điểm. Các tính toán tương tự cũng được dùng để hòa nhập giữa các phần tử khác. Cung hòa nhập còn được gọi là cung *gờ* hoặc *bán kính gờ*.

Bán kính riêng

Ngược với bán kính hòa hợp là bán kính riêng - không có sự hòa hợp đều giữa hai phần tử biên dạng, thay vào đó là giao điểm. Về toán học, luôn luôn có hai lựa chọn khả dĩ, tuy nhiên bản vẽ chi tiết phải hoàn toàn rõ ràng về hình dạng của bán kính riêng bất kỳ. Bán kính riêng cũng có thể tồn tại giữa *hai đường*, *một đường* và *cung*, hoặc giữa *hai cung*. Bán kính riêng có thể được xác định nơi điểm đầu hoặc điểm cuối *không* tiếp xúc với phần tử kế cận, nhưng giao cắt ở hai điểm. Tính toán thực tế các tọa độ điểm đối với điểm đầu hoặc điểm cuối hoàn toàn tương tự như cung hòa hợp, tùy theo phương pháp lập kích thước được sử dụng trên bản vẽ.

LẬP TRÌNH ĐƯỜNG TRÒN

Mọi hệ thống Fanuc và nhiều bộ điều khiển khác hỗ trợ lập trình đường tròn toàn phần. Đường tròn là cung được gia công theo 360^0. Cắt gọt theo đường tròn về lý thuyết có thể thực hiện trên máy tiện, nhưng thực tế hầu như không thể. Đối với phay, lập trình đường tròn tương đối đơn giản và được yêu cầu cho một số nguyên công, chẳng hạn:

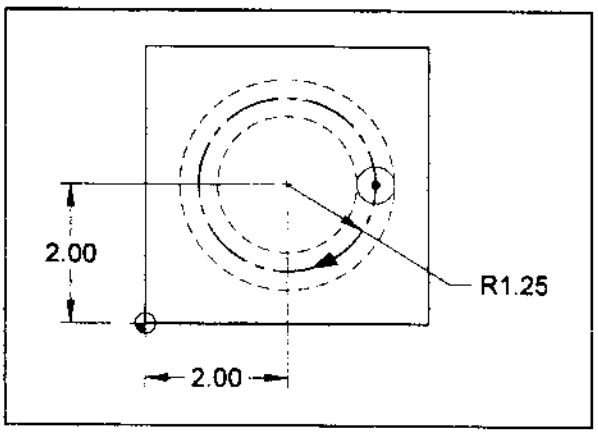
- Phay hốc tròn.
- Phay lõm.
- Phay xoắn (với trục tuyến tính).
- Phay hình trụ, cầu, hoặc côn.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Cắt gọt tốc độ cao 235

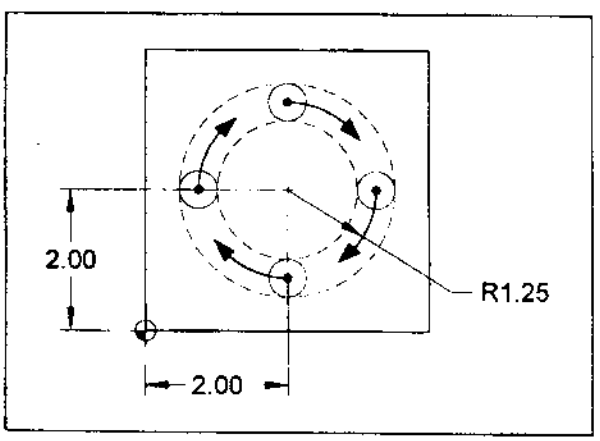
Cắt gọt tốc độ cao thường được định nghĩa là chuyển động dao hoàn tất 360^0 giữa điểm đầu và cuối, với kết quả là các tọa độ đồng nhất cho các vị trí dao đầu và cuối. Dưới đây là ví dụ về lập trình một block cho đường tròn (Hình 28.4).



Hình 28.4. *Lập trình đường tròn toàn phần sử dụng nhập chương trình một block.*

```
...
G90 G54 G00 X3.25 Y2.0 S800 M03
G01 Z-0.25 F10.0
G02 X3.25 Y2.0 I-1.25 J0 F12.0 (FULL CIRCLE)
G00 Z0.1
...
```

Các bộ điều khiển cũ chỉ cho phép nội suy đường tròn không quá một góc phần tư trong block. Trong trường hợp đó, chuyển động tròn phải được phân chia vào 4 hoặc 5 block, tùy theo vị trí khởi đầu của dao. Sử dụng lại bản vẽ trên Hình 28.4, chương trình sẽ hơi dài hơn nhưng có cùng kết quả (Hình 28.5).

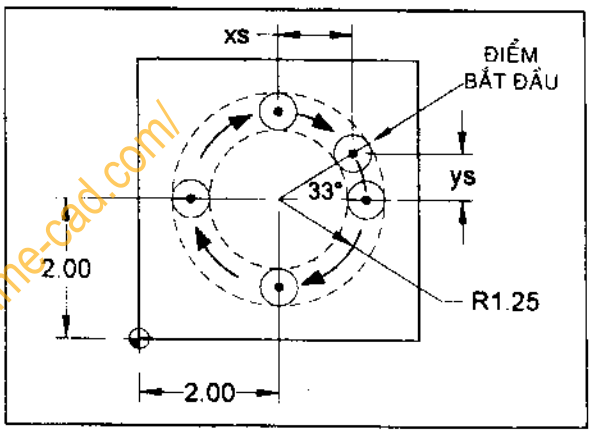


Hình 28.5. *Lập trình đường tròn toàn phần sử dụng nhập chương trình bốn block.*

```
...
G90 G54 G00 X3.25 Y2.0 S800 M03
G01 Z-0.25 F10.0
G02 X2.0 Y0.75 I-1.25 J0 F12.0 (BLOCK 1 OF 4)
G02 X0.75 I2.0 I0 J1.25        (BLOCK 2 OF 4)
G02 X2.0 Y3.25 I1.25 J0        (BLOCK 3 OF 4)
G02 X3.25 Y2.0 I0 J-1.25       (BLOCK 4 OF 4)
G00 Z0.1
...
```

Đây là ví dụ lập trình bốn block bao quát sự cắt gọt theo toàn bộ đường tròn. Các điểm bắt đầu và kết thúc cung đều ở điểm *phần tư* của đường trục, đây là yếu tố rất quan trọng trong lập trình. Điểm *phần tư* trong ví dụ này tương đương vị trí 3 giờ trên đồng hồ (0^0). Chú ý, G02 được lặp lại trong từng block chỉ nhằm nhấn mạnh, không cần lặp lại trong chương trình thực. Điều này cũng áp dụng cho J0 và I0 – không cần viết lại chúng, trừ khi có sự thay đổi.

Lập trình sẽ hơi khó hơn nếu thiết lập điểm bắt đầu cắt gọt không trùng vào điểm bất kỳ trong 4 điểm *phần tư*, 0^0, 90^0, 180^0, và 270^0. Ví dụ, nếu bắt đầu cắt gọt ở 33^0, sẽ có *năm* block, thay vì *bốn*, các điểm tọa độ XY của điểm bắt đầu (được ghi theo X_S và Y_S) phải được tính toán bằng hàm lượng giác (Hình 28.6).



Hình 28.6. *Lập trình đường tròn toàn phần sử dụng năm block.*

```
...
G90 G54 G00 X3.0483 Y2.6808 S800 M03
G01 Z-0.25 F10.0
G02 X3.25 Y2.0 I-1.0483 J-0.6808(BLOCK 1 OF 5)
G02 X2.0 Y0.75 I-1.25 J0        (BLOCK 2 OF 5)
G02 X0.75 Y2.0 I0 J1.25         (BLOCK 3 OF 5)
G02 X2.0 Y3.25 I1.25 J0         (BLOCK 4 OF 5)
G02 X3.0483 Y2.6808 I0 J-1.25   (BLOCK 5 OF 5)
G00 Z0.1
...
```

Các giá trị X_S và Y_S được tính toán theo hàm lượng giác:

$X_S = 1.25 \times \cos 33^0 = 1.0483382$
$Y_S = 1.25 \times \sin 33^0 = 0.6807988$

Từ kết quả này, điểm bắt đầu đường cắt sẽ là:

$X = 2 + XS = 3.0483382 = X3.0483$
$Y = 2 + YS = 2.6807988 = Y2.6808$

Nếu hệ điều khiển hỗ trợ nhập chương trình đường tròn trong *một block*, chương trình

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

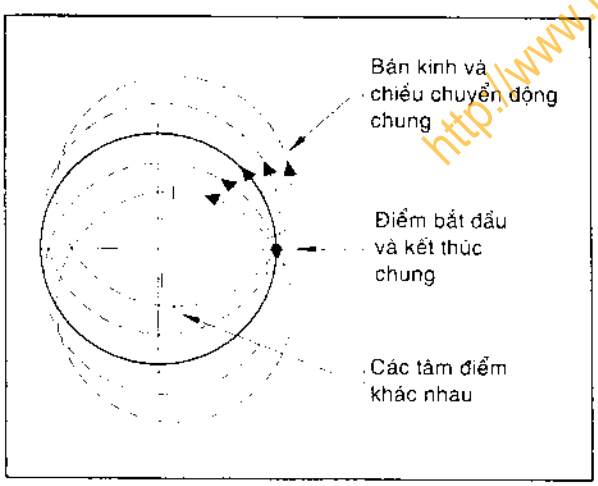
xuất sẽ ngắt tất cả chương trình. Chú ý đến các vector cung I và J – giá trị bán kính R *không thể sử dụng* trong trường hợp này. Lý do là các vector I và J *luôn luôn* có ý nghĩa đặc thù, còn R có thể không rõ nghĩa. Ví dụ dưới đây là đúng, sử dụng các vector I và J:

```
...
G90 G54 G00 X3.0483 Y2.6808 S800 M03
G01 Z-0.25 F9.0
G02 X3.0483 Y2.6808 I-1.0483 J-0.6808
G00 Z0.1
...
```

Các vector I và J không thể tùy tiện thay thế bằng địa chỉ R. Ví dụ này sẽ *sai*.

```
...
G90 G54 G00 X3.0483 Y2.6808 S800 M03
G01 Z-0.25 F9.0
G02 X3.0483 Y2.6808 R1.25 F12.0      (* SAI *)
G00 Z0.1
...
```

Lý do? Về toán học, có *nhiều* tùy chọn để lập trình đường tròn. Nếu giá trị R được lập trình cho cung 360^0, sẽ không xảy ra chuyển động tròn và block đó sẽ bị bộ điều khiển bỏ qua. Đây là biện pháp an toàn được thiết lập trong phần mềm điều khiển, để tránh cắt gọt cung không chuẩn do có quá nhiều khả năng có thể. Trên Hình 28.7, chỉ nêu vài cung khả dĩ. Các đường tròn này có cùng chiều cắt gọt, cùng điểm, bắt đầu, điểm kết thúc và bán kính, *nhưng có các tâm khác nhau*.

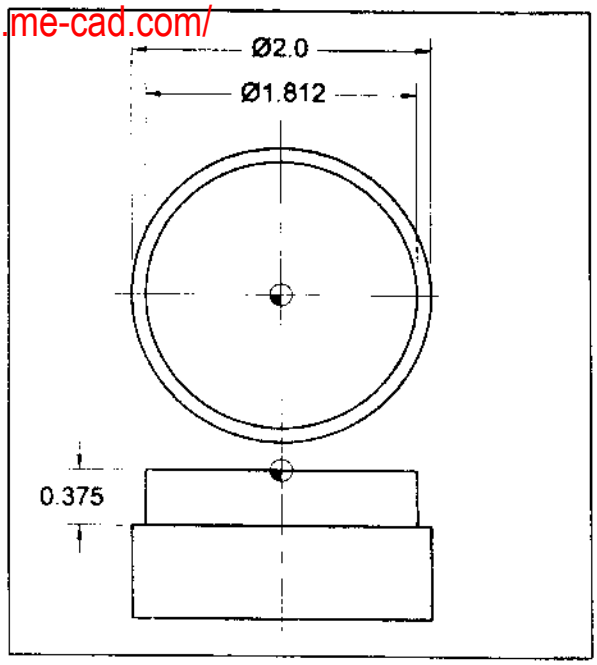


Hình 28.7. Các đường tròn có cùng bán kính và điểm bắt đầu.

Phay mặt lồi

Bản vẽ ví dụ phay mặt lồi đơn giản được nêu trên Hình 28.8.

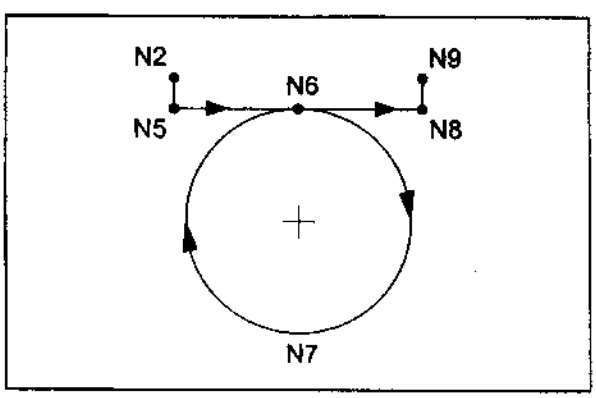
Phay mặt lồi là thuật ngữ được dùng để phay *ngoài* dọc theo chu vi đường tròn. Ngược lại là phay *trong*, chẳng hạn *hốc tròn*. Dao được chọn là dao phay mặt đầu Ø.75 được lập trình theo chiều sâu .375 inch:



Hình 28.8. Ví dụ phay mặt lồi – chương trình O2801

```
O2801
(0.75 DIA END MILL)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X-1.0 Y1.5 S750 M03
N4 G43 Z0.1 H01
N5 G01 Z-0.375 F40.0 M08
N6 G41 Y0.906 D01 F20.0
N7 X0 F14.0
N8 G02 J-0.906
N9 G01 X1.0 F20.0 M09
N10 G40 Y1.5 F40.0 M05
N11 G91 G28 X0 Y0 Z2.0
N12 M30
%
```

Trong chương trình O2801, dao trước hết chuyển động đến vị trí XY và chiều sâu, sau đó bắt đầu bù bán kính dao. Khi đạt đến chiều sâu cắt, dao thực hiện chuyển động phay ngược thẳng đến đỉnh, sau đó quét xung quanh đường tròn đến cùng điểm ban đầu, chuyển động thẳng ra xa và đảo chiều chuyển động, đi đến



Hình 28.9. Ví dụ phay mặt lồi – chuyển động dao trong chương trình O2801

Nhiều tài liệu hay có tại : http://www.me-cad.com/

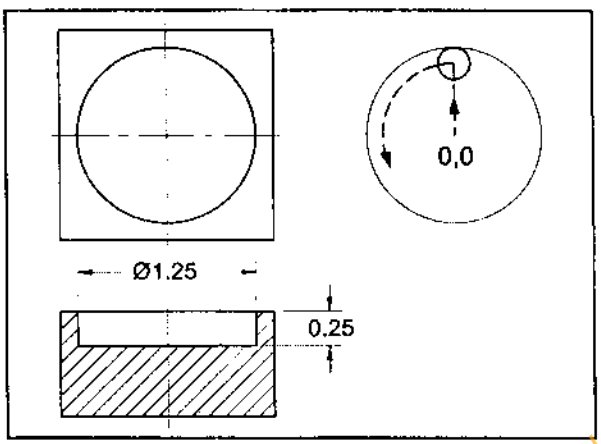
điểm bắt đầu trên trục Y. Hình 28.9 minh họa các số block.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Các ứng dụng kế tiếp có thể gồm phay thô nhiều đường, phay bán kính, hai dao cắt, và các lựa chọn gia công khác.

Cắt đường tròn trong – Khởi đầu tuyến tính

Cắt đường tròn trong tương đối phổ biến và có nhiều ứng dụng, chẳng hạn các hốc tròn hoặc doa ngược. Trong ví dụ, hốc tròn Ø1.25 được gia công đến chiều sâu .250 inch, trong chương trình O2802. Chuyển động tuyến tính đơn giản được dùng để bắt đầu gia công, vị trí hòa hợp điểm ăn dao là không quan trọng. Dụng cụ cắt là dao phay mặt đầu cắt gọt từ tâm hình tròn (Hình 28.10).



Hình 28.10. Cắt gọt đường tròn trong – khởi đầu tuyến tính

```
O2802
(0.5 DIA CENTER END MILL)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 YO S900 M03
N4 G43 Z0.1 H01
N5 G01 Z-0.25 F10.0 M08
N6 G41 Y0.625 D01 F12.0
N7 G03 J-0.625
N8 G01 G40 X0 F20.0 M09
N9 G91 G28 X0 Y0 Z2.0 M05
N10 M30
%
```

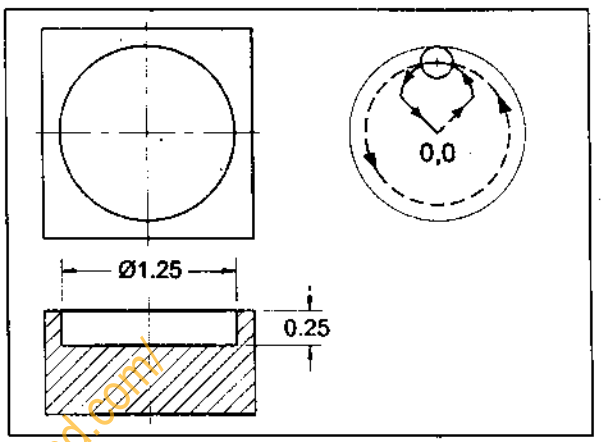
Chương trình O2802 cho thấy cả hai điểm bắt đầu và kết thúc cung đều ở 90^0. Sự bù bán kính dao khởi đầu khi chuyển động từ tâm hình tròn.

Bù bán kính dao không thể bắt đầu hoặc kết thúc trong chế độ chuyển động tròn.

Điều này là đúng với hầu hết các ứng dụng gia công theo đường tròn, trừ vài ngoại lệ sử dụng chu kỳ đặc biệt.

Cắt đường tròn trong – Khởi đầu theo đường tròn

Phương pháp lập trình tiếp cận dao tuyến tính nêu trên sẽ không thực tế khi cần có sự hòa hợp đều giữa sự tiếp cận và đường cắt tròn. Để cải thiện độ bóng bề mặt, điểm bắt đầu chuyển động tròn có thể đạt tới trên cung tròn. Sự khởi đầu thông dụng là từ tâm điểm, ban đầu là tuyến tính 45^0, để áp dụng bù bán kính dao, sau đó trên cung hòa hợp với đường tròn toàn phần. Hình 28.11 minh họa nguyên tắc này và chương trình O2803 là chương trình hoàn chỉnh.



Hình 28.11. Cắt gọt đường tròn trong – khởi đầu tuyến tính và đường tròn.

http://www.me-cad.com/

```
O2803
(0.5 DIA CENTER END MILL)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S900 M03
N4 G43 Z0.1 H01
N5 G01 Z-0.25 F10.0 M08
N6 G41 X0.3125 Y0.3125 D01 F12.0
N7 G03 X0 Y0.625 R0.3125
N8 J-0.625
N9 X-0.3125 Y0.3125 R0.3125
N10 G01 G40 X0 Y0 F20.0 M09
N11 G91 G28 X0 YO Z2.0 M05 N12 M30
%
```

Phương pháp lập trình này hơi dài hơn, nhưng chất lượng bề mặt và sự tiếp cận dao tốt hơn so với khởi đầu tuyến tính.

Nếu hệ điều khiển có tùy chọn *User Macro* và cần gia công nhiều hốc tròn, ví dụ O2803 có thể chuyển thành chương trình con macro. Một số bộ điều khiển có sẵn chu kỳ phay hốc tròn.

Chu kỳ phay đường tròn trong

Một số bộ điều khiển, ví dụ Yasnac hoặc Mitsubishi, (nhưng Fanuc không có), có chu kỳ (chương trình con) cắt gọt đường tròn trong sử dụng các lệnh chuẩn bị đặc biệt, thường là G12

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

và G13. Các chu kỳ này là hỗ trợ lập trình rất thuận tiện. Fanuc cũ nhiều loại trước đã bỏ qua tính năng này.

Có quan hệ logic giữa G02 và G12, giữa G03 và G13.

G12	Chu kỳ cắt gọt toàn phần đường tròn *CW thuận chiều đồng hồ*
G13	Chu kỳ cắt gọt toàn phần đường tròn *CCW ngược chiều đồng hồ*

Định dạng lập trình sử dụng hai lệnh lập trình này tương đối đơn giản:

```
G12 I.. D.. F..      Đường tròn toàn phần CW
G13 I.. D.. F..      Đường tròn toàn phần CCW
```

Trong định dạng này, địa chỉ I là *bán kính* đường tròn hoàn tất được lập trình theo giá trị số gia và dấu. Nếu dấu là dương (+), điểm bắt đầu cắt sẽ ở 0^0. Nếu dấu âm (-), điểm bắt đầu cắt là 180^0. Lệnh này không thể bắt đầu theo chiều trục Y.

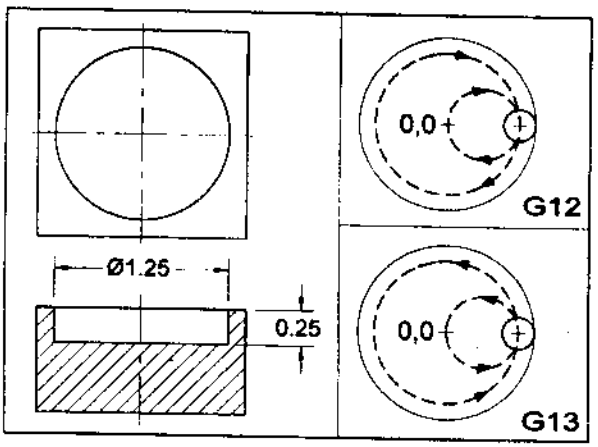
Địa chỉ D được lập trình là số đăng ký điều khiển đối với bù bán kính dao và G là địa chỉ tốc độ cắt. Chu kỳ này có thể hơi khác giữa các bộ điều khiển, nhưng có cùng bản chất.

Các điều kiện khác cũng phải được chấp nhận để sử dụng thành công chu kỳ này. Dụng cụ cắt phải *luôn luôn* khởi đầu ở tâm hốc tròn, mặt phẳng cắt gọt phải xác lập theo mặt phẳng XY và vị trí bắt đầu cung tròn thường được thiết lập sẵn, 0^0 hoặc 180^0 (không bắt đầu từ trục Y). Ngoài ra còn có bù bán kính dao cắt *được thiết lập sẵn* (G12 sang phải, G13 sang trái). Bạn không được lập trình G41 và G42 khi sử dụng G12 hoặc G13. Nếu bán kính dao có hiệu lực, sẽ bị vượt qua khi chọn G12 hoặc G13. Phương pháp an toàn nhất là lập trình hai chu kỳ này trong chế độ G40 (bù bán kính dao bị xóa).

Điều không đúng trong ứng dụng chuyển động tròn khác, sẽ đúng trong trường hợp này. Trong lập trình bình thường các cung và đường tròn, sự bù bán kính dao không thể bắt đầu bằng chuyển động dao theo cung. Trong chế độ lập trình G12/G13, chuyển động ban đầu từ vị trí tâm hình tròn không bù là *chuyển động tròn* đến điểm khởi đầu bù trên chu vi cung tròn. Toàn bộ điều này được thiết lập trong bộ điều khiển và không có các lựa chọn khác. Bạn hãy coi tình huống này là trường hợp đặc biệt, không phải là nguyên tắc.

Trên một số máy CNC của Yasnac, có thêm một số tham số trong định dạng G12/G13 – tham số bán kính, còn gọi là tham số R, biểu thị phần chuyển động nhanh đặc biệt, được thiết kế để giảm thời gian chạy dao không cắt gọt.

Bản vẽ trên Hình 28.12 được dùng làm ví dụ lập trình G12/G13.



Hình 28.12. Cắt gọt theo đường tròn sử dụng các chu kỳ G12/G13 – Chương trình O2804.

```
O2804
(0.5 DIA CENTER CUTTING END MILL)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S900 M03
N4 G43 Z0.1 H01
N5 G01 Z-0.25 F10.0 M08
N6 G13 I0.625 D01 F12.0 M09     (IF AVAILABLE)
N7 G91 G28 X0 Y0 Z2.0 M05
N8 M30
%
```

Chương trình này chỉ giảm bớt hai block nhưng đơn giản hơn nhiều. Bù bán kính dao là tự động (được thiết lập sẵn) và sự chỉnh sửa tại máy sẽ dễ dàng hơn. Ở đây còn có một ưu điểm do điểm bắt đầu trên đường tròn không phải là từ đường thẳng mà là từ cung dẫn hướng vào, chất lượng bề mặt sẽ tốt hơn so với các kiểu tiếp cận dao khác. Đây là phương pháp thông dụng khi yêu cầu độ bóng bề mặt cao. Ngoài ra còn có cung dẫn hướng ra, tương tự cung dẫn hướng vào, khi hoàn tất sự cắt gọt cung tròn.

LẬP TRÌNH CẮT GỌT CUNG TRÒN

Với sự cắt gọt cung toàn phần, chuyển động đủ 360^0, địa chỉ R hoàn toàn không thể sử dụng. Các vector tâm cung I và J được áp dụng, kể cả trên các bộ điều khiển hiện đại.

Điều gì sẽ xảy ra nếu đường tròn là 359.999^0? Thứ nhất, đường tròn *phải* là 360^0, do đó thuật ngữ "đường tròn" sẽ không đúng. Dù hiệu số rất nhỏ (0.001^0), nhưng vẫn có sự khác biệt giữa cung và đường tròn. Tuy khác biệt này về toán học là quan trọng hơn so với lập trình thực tiễn, nhưng vẫn có vai trò rất quan trọng. Trong nội suy đường tròn, *đường*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tròn chưa khép kín vẫn được coi là *cung*. Bạn hãy xét những thí dụ sau đây. Nếu giá trị R là công cung 90^0, có thể lập trình địa chỉ R, ví dụ:

```
G01 X2.0 Y5.25 F12.0
G02 X3.75 Y7.0 R1.75
```

Nếu lập trình cung *chính xác* 180^0, chương trình cũng không quá khác biệt:

```
G01 X2.0 Y5.25 F12.0
G02 X5.5 Y5.25 R1.75
```

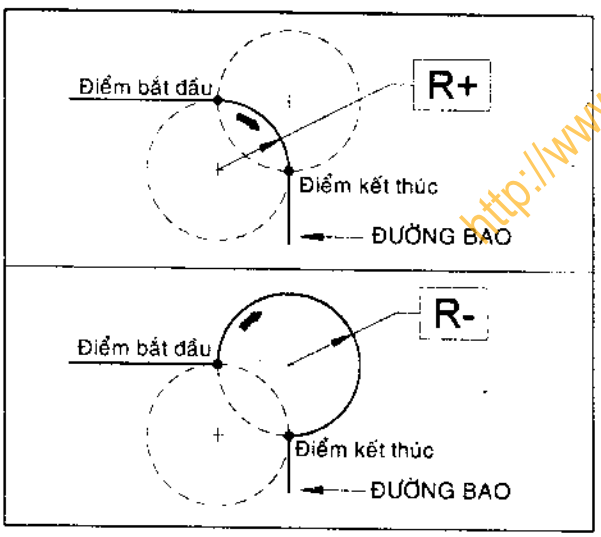
Chú ý, tọa độ Y không thay đổi đối với vị trí đầu và cuối của cung này. Giá trị Y trong block chuyển động tròn không cần lặp lại, sử dụng ở đây chỉ để minh họa.

Ví dụ kế tiếp là lập trình cung 270^0, vẫn sử dụng địa chỉ R. Các block sau đây là đúng?

```
G01 X10.5 Y8.625 F17.0
G02 X13.125 Y6.0 R2.625
```

Các block này có vẻ đúng, các tính toán, định dạng, các lệnh, tất cả đều có vẻ đúng. Nhưng chương trình là *sai*. Kết quả là cung 90^0, thay vì cung 270^0.

Bạn hãy nghiên cứu minh họa trên Hình 28.13. Ở đây không chỉ có một, mà là *hai* khả năng có thể xảy ra khi dùng địa chỉ R cho cung tròn. Đường nét liền là quỹ đạo dao, đường nét đứt biểu thị hai bán kính khả dĩ.



Hình 28.13. *Dấu của địa chỉ R khi cắt theo đường tròn khác dấu do đó khác tâm.*

Các nhà lập trình thường không cần suy nghĩ về các khả năng toán học này khi lập trình các cung *nhỏ hơn* 180^0. Đây là tình huống tương tự với đường tròn đã được đề cập ở phần trước. Tuy có thể dùng các vector I và J để giải quyết vấn đề, nhưng bạn còn có một lựa chọn khác. Địa chỉ R vẫn có thể được dùng trong chương trình, nhưng với dấu *âm* cho cung bất kỳ lớn hơn 180^0. Đối với các cung nhỏ hơn 180^0, bán kính R dương vẫn giữ nguyên tác dụng. Nếu giá trị R không có dấu, sẽ được coi là *dương*. Bạn hãy so sánh ví dụ lập trình:

```
G01 X10.5 Y8.625 F17.0
G02 X13.125 Y6.0 R2.625          (90 ĐỘ)
```

Với ví dụ thứ hai, hoàn toàn đồng nhất, chỉ khác dấu giá trị R

```
G01 X10.5 Y8.625 F17.0
G02 X13.125 Y6.0 R-2.625         (270 ĐỘ)
```

Nếu thường xuyên lập trình các cung lớn hơn 180^0, bạn hãy thiết lập phong cách lập trình riêng, nhằm tránh các nhầm lẫn liên quan đến dấu của địa chỉ R.

TỐC ĐỘ CẮT ĐỐI VỚI CHUYỂN ĐỘNG TRÒN

Trong hầu hết các chương trình, tốc độ cắt đối với nội suy đường tròn được xác định tương tự nội suy tuyến tính. Tốc độ cắt gọt các cung dựa trên các quy ước gia công tiêu chuẩn. Chúng bao gồm gá lắp, tính gia công cắt gọt của vật liệu, đường kính dao và độ cứng vững, kinh nghiệm của nhà lập trình, và các yếu tố khác.

Nhiều nhà lập trình không xét bán kính gia công khi chọn tốc độ cắt cho dao. Tuy nhiên, nếu độ bóng bề mặt là yêu cầu quan trọng, bạn hãy xét kích cỡ của từng bán kính được ghi trên bản vẽ chi tiết. Có lẽ bạn nên lập trình cùng tốc độ cắt chuyển động tuyến tính và chuyển động tròn, sau đó có thể sẽ điều chỉnh, tăng hoặc giảm, theo điều kiện thực tế.

Trong lập trình tiện, không có lý do để phân biệt giữa chuyển động dao tuyến tính và theo đường tròn, bất kể giá trị bán kính. Bán kính mũi dao thường khá nhỏ, chỉ khoảng 0.0313 inch (0.8 mm), và quỹ đạo dao cân bằng là sát với quỹ đạo dao lập trình, lấy từ bản vẽ. Đây không phải là trường hợp lập trình phay biên dạng, trong đó bán kính dao lớn là bình thường và phổ biến.

Tốc độ cắt theo cung được điều chỉnh là không cần thiết trong từng chương trình.Nếu quỹ đạo tâm dao sát với biên dạng chi tiết trên bản vẽ, hoàn toàn không cần điều chỉnh. Mặt khác, khi dao đường kính lớn được dùng để cắt biên dạng với bán kính ngoài nhỏ, có thể phát sinh vấn đề về chất lượng bề mặt. Trong trường hợp đó, quỹ đạo tâm dao tạo ra cung dài hơn so với cung trên bản vẽ. Trong tình huống tương tự, nếu dao đường kính lớn được dùng cho cung trong, quỹ đạo khoảng cách tương đương sẽ ngắn hơn chiều dài cung theo bản vẽ.

Trong lập trình bình thường, tốc độ cắt tuyến tính cũng được dùng cho các cung, được

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

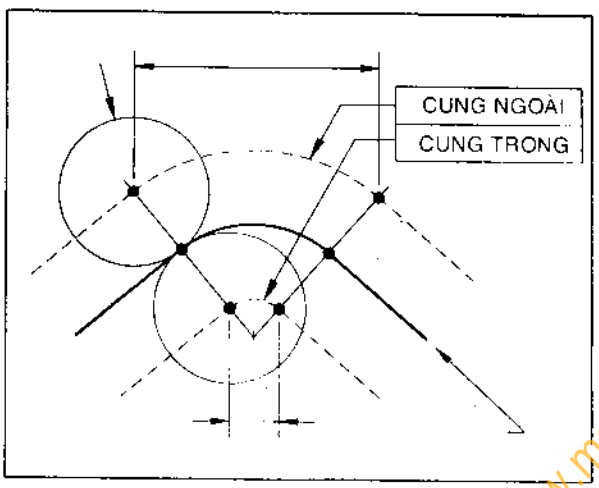
xác định theo tính gia công cắt gọt của vật liệu đã chọ[illegible] công thức tính tốc độ cắt tuyến tính:

$$F_I = r/min \times F_t \times n$$

Trong đó:

F_I = tốc độ cắt tuyến tính (in/min hoặc mm/min)
r/min = Tốc độ trục chính
F_t = Tốc độ cắt/răng
n = Số lưỡi cắt

Tốc độ cắt tuyến tính với 1000 r/min, 0.0045 in/răng và hai lưỡi cắt, sẽ là 9 in/min. Sử dụng đường kính dao tương đối lớn, chẳng hạn Ø.625 (15.875 mm) hoặc lớn hơn, điều chỉnh tốc độ cắt tuyến tính tăng hoặc giảm đối với chuyển động tròn có thể là cần thiết để bảo đảm chất lượng bề mặt gia công.



Hình 28.14. Điều chỉnh tốc độ cắt đối với chuyển động dao theo đường tròn.

Nguyên tắc cơ bản về điều chỉnh tốc độ cắt đối với các cung khi lập trình tốc độ cắt tuyến tính là *tăng đối với cung ngoài* và *giảm đối với cung trong* (Hình 28.14).

Cung ngoài của quỹ đạo dao dài hơn cung trên bản vẽ.

Cung trong của quỹ đạo ngắn hơn cung trên bản vẽ.

Hai công thức cho phép tìm tốc độ cắt theo cung được điều chỉnh, về toán học là tương đương với tốc độ cắt tuyến tính. Cả hai công thức chỉ nên dùng cho các biên dạng trong hoặc ngoài, *không* dùng cho gia công thô vật liệu nguyên khối.

Tốc độ cắt đối với cung ngoài

Đối với các cung ngoài, tốc độ cắt điều chỉnh sẽ *lớn hơn* tốc độ cắt tuyến tính, được tính theo công thức:

$$F_O = \frac{F_I \times (R + r)}{R}$$

Trong đó: F_O = Tốc độ cắt cung ngoài
F_I = Tốc độ cắt tuyến tính
R = Bán kính ngoài của chi tiết
r = Bán kính dao cắt

Dựa trên tốc độ cắt tuyến tính 14 in/min, bán kính cung ngoài .375 cần điều chỉnh *tăng* đối với dao cắt Ø.50:

$F_O = 14 \times (.375 + .25) / .375 = 23.333333$

Kết quả là sự tăng rõ rệt, đến F23.3 trong chương trình. Bạn hãy xét ví dụ này với bán kính dao đến 0.75 inch (Ø1.5):

$F_O = 14 \times (0.375 + 0.75) / 0.375 = 42.0$

Tốc độ cắt tăng từ 14 in/min đến 42 in/min – gấp ba lần.

Tốc độ cắt đối với cung trong

Đối với các cung trong, tốc độ cắt điều chỉnh sẽ nhỏ hơn tốc độ cắt tuyến tính, được tính theo công thức:

$$F_I = \frac{F_I \times (R - r)}{R}$$

Trong đó: F_I = Tốc độ cắt cung trong
F_I = Tốc độ cắt tuyến tính
R = Bán kính trong của chi tiết
r = Bán kính dao cắt

Dựa trên tốc độ cắt tuyến tính 14 in/min, tốc độ cắt đối với bán kính trong .8243 sử dụng dao Ø 1.25, phải được điều chỉnh giảm:

$F_I = 14 \times (.8243 - .625) / .8243 = 3.384932$

Kết quả là tốc độ cắt 3.38 in/min. Trong chương trình này, giá trị 3.38 sẽ được áp dụng cho địa chỉ F.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

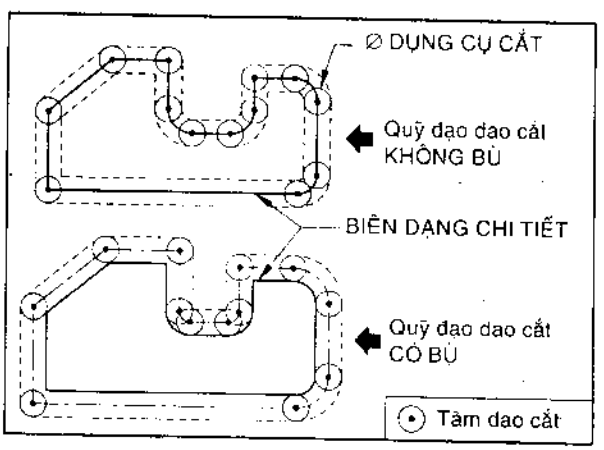
Nhiều tài liệu hay có tại : http://www.me-cad.com/

29 BÙ BÁN KÍNH DAO CẮT

Biên dạng chi tiết – còn gọi là profile – thường được lập trình cho các ứng dụng phay bằng cách thiết lập chiều sâu theo trục Z trước, sau đó dịch chuyển dao lần lượt theo trục X, trục Y, hoặc đồng thời theo cả hai trục đó. Đối với các ứng dụng tiện, trục X hoặc trục Z, hoặc cả hai trục được dùng để gia công mặt, tiện, hoặc doa biên dạng. Đối với cả hai kiểu gia công, từng phần tử biên dạng đòi hỏi một block chuyển động cắt. Các chuyển động giữa những điểm thay đổi biên dạng có thể được lập trình theo inch, hoặc milimet, sử dụng vị trí giá trị tuyệt đối hoặc khoảng cách số gia. Trong khi gia công, bạn cần nhớ kiểu lập trình này sử dụng *đường tâm trục chính* là các chuyển động dao X và Y hoặc X và Z. Dù lập trình đường tâm là phương pháp rất thuận tiện để viết chương trình, nhưng cũng là phương pháp không thể chấp nhận đối với gia công. Trong khi tiếp xúc với phôi, *lưỡi* dụng cụ cắt phải chạm vào biên dạng chi tiết lập trình, thay vì *đường tâm*.

Quỹ đạo dao đối với mọi gia công biên dạng luôn luôn tương đương với chuyển động dao cắt. Dù sử dụng trên trung tâm gia công CNC hay trên máy tiện CNC, *lưỡi dao cắt phải luôn luôn tiếp tuyến với biên dạng*, nghĩa là chuyển động dao phải tạo ra quỹ đạo sao cho tâm dao cắt *luôn luôn ở cùng khoảng cách* tính từ biên dạng của chi tiết. Điều này được gọi là quỹ đạo dao *khoảng cách tương đương*.

Minh họa trên Hình 29.1 nêu rõ hai kiểu quỹ đạo dao, *bù* và *không bù*. Cả hai được áp dụng cho cùng một biên dạng, đường kính dao được biểu thị rõ, kể cả các vị trí dao.



Hình 29.1. *Quỹ đạo dao không bù (trên) và có bù (dưới) bán kính dao.*

CÁC TÍNH TOÁN BẰNG TAY

Bạn có thể nhận thấy một số vấn đề thực tiễn trên Hình 29.1. Điều dễ thấy nhất là biên dạng gia công luôn luôn xảy ra với quỹ đạo dao được bù bán kính dao, nghĩa là tâm dao phải ở vị trí được nêu trong hình dưới. Yêu cầu gia công này không tương hợp với thực tế trên bản vẽ. Trong bản vẽ, mọi kích thước đều theo biên dạng chi tiết, *không* theo biên dạng của tâm dụng cụ cắt. Bản vẽ được lập kích thước theo vị trí dao được minh họa trong hình trên. Vấn đề là tìm vị trí tâm dao cắt từ bản vẽ biên dạng chi tiết.

Giải pháp là *tính toán* quỹ đạo tâm dao. Trong thực tế, có thể không cần tính toán, nếu hệ thống CNC được trang bị tính năng cao cấp *bù bán kính dao cắt*. Trên hệ thống tiện CNC, tính năng này được gọi là *bù bán kính mũi dụng cụ cắt*. Tính năng điều khiển cao cấp đó cho phép nhà lập trình áp dụng lệnh bù, lập trình biên dạng chi tiết theo các kích thước *trên bản vẽ* và để bộc lộ điều khiển thực hiện mọi tính toán cần thiết và các điều chỉnh một cách tự động.

Chương này tập trung vào phương pháp lập trình tự động, sử dụng tính năng bù bán kính (mũi) dao. Mọi máy CNC hiện đại đều có tính năng bù bán kính dao cắt. Nếu tuân theo các nguyên tắc cơ bản, tính năng đó rất dễ áp dụng.

Để tự động hóa hoạt động nào đó, đều đầu tiên là hiểu cơ chế hoạt động đó. Nếu hiểu rõ về cơ chế tự động của hệ thống, bạn sẽ thực hiện công việc dễ dàng hơn, đặc biệt khi gặp khó khăn, bạn có thể nhanh chóng giải quyết vấn đề. Để hiểu cơ chế bù bán kính dao, điều quan trọng là hiểu các nguyên lý trong hệ thống, dựa trên các hệ phương trình toán, đặc biệt là các hàm lượng giác (Hình 29.2).

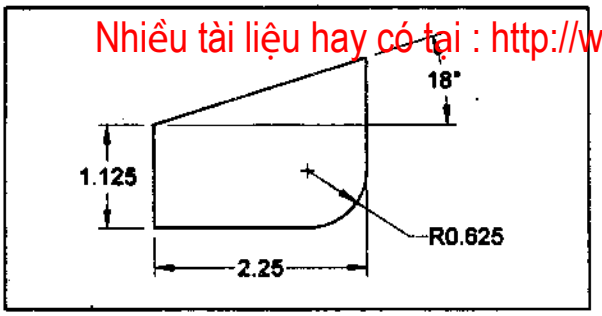
Zero chương trình sẽ được chọn ở góc dưới bên trái chi tiết. Do đây là cắt gọt ngoài, theo chế độ phay ngược, dao sẽ bắt đầu trước theo chiều trục Y. Tại thời điểm đó, các vị trí bắt đầu và kết thúc là không quan trọng, chỉ tính toán các điểm biên dạng riêng rẽ tại các vị trí giao cắt và các điểm tiếp tuyến.

Chú ý có năm điểm trên bản vẽ xác định sự thay đổi biên dạng. Các điểm này có thể là *giao điểm* hoặc *tiếp điểm*. Do mỗi điểm có hai giá trị tọa độ, tổng cộng cần có *mười* giá trị.

http://www.me-cad.com/

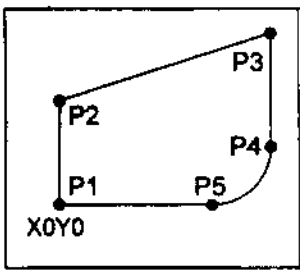
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 29.2. *Bản vẽ đơn giản để tính toán bằng tay (các ví dụ).*

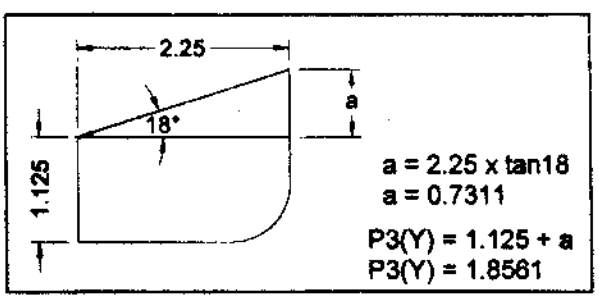
Bản vẽ luôn luôn cung cấp vài điểm không cần tính toán. Bạn nên sắp xếp và đánh dấu trước các điểm đó từ bản vẽ, sau đó vẽ lưu đồ theo thứ tự trên quỹ đạo dao. Bạn hãy quan sát kỹ Hình 29.3, nêu rõ cả năm điểm và các giá trị không cần tính toán, có lẽ chỉ cần cộng hoặc trừ.



Pt	X-AXIS	Y-AXIS
P1	X0.0000	Y0.0000
P2	X0.0000	Y1.1250
P3	X2.2500	?
P4	X2.2500	Y0.6250
P5	X1.6250	Y0.0000

Hình 29.3. *Các điểm thay đổi biên dạng trên quỹ đạo dao cắt*

Trong mười giá trị, đã biết chín. Chỉ có giá trị Y của P3 là chưa có trên bản vẽ. Bất kể có bù bán kính dao hay không, luôn luôn cần vài tính toán và đây là một trong các tính toán đó. Nói chung, lập trình *bằng tay* được thực hiện với *giấy và bút*. Hình 29.4 minh họa phương pháp tính toán P3 sử dụng hàm lượng giác.



Hình 29.4. *Tính toán để tìm tọa độ Y chưa biết*

Các giá trị tọa độ của cả năm điểm được ghi trong bảng dưới đây:

Điểm	Tọa độ X	Tọa độ Y
P1	X0	Y0
P2	X0	Y1.125
P3	X2.25	Y1.8561
P4	X2.25	Y0.625
P5	X1.625	Y0

Sau khi biết tất cả các tọa độ, bạn có đủ dữ liệu để bắt đầu quỹ đạo dao, nhưng *chỉ nếu* sử dụng tính năng bù bán kính dao. Tuy nhiên, vào thời điểm này điều cần nhấn mạnh là *xác định toàn bộ tập hợp các điểm mới – các tọa độ tâm dao cắt*.

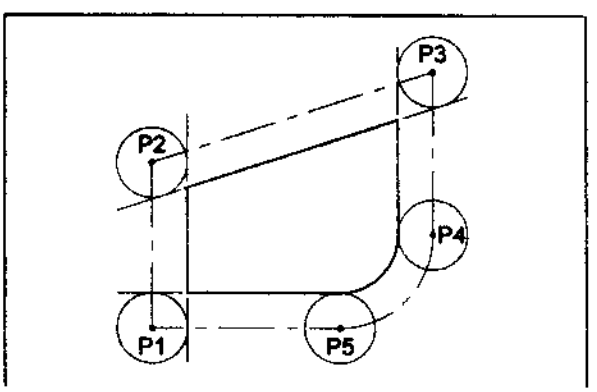
Quỹ đạo các tâm điểm dụng cụ cắt

Hầu như mọi dao phay đều có hình tròn (hoặc trụ). Ví dụ, dao phay mặt đầu, có đường kính với kích cỡ xác định. Ngay cả các dao tiện và doa cũng có đầu tròn (được gọi là *bán kính mũi dụng cụ cắt*) dù tương đối nhỏ. Dao phay và đỉnh dao tiện là các vật thể tròn, do đó luôn luôn có điểm tâm. Điều này có vẻ quá đơn giản, nhưng rất cơ bản, đây là *phần tử chính* trong toàn bộ khái niệm về bù bán kính dao. Mọi hệ thống điều khiển đều phải xét phần tử này.

Ví dụ, nếu dùng máy cưa biên dạng để cắt gỗ – bạn cắt như thế nào? Sử dụng bút chì bạn phác thảo biên dạng trên tấm gỗ, lắp lưỡi cưa và bắt đầu cắt. Ở đâu? *Sự cắt bắt đầu từ* phía ngoài *nét phát thảo biên dạng, nếu ngược lại, sản phẩm cắt sẽ quá lớn hoặc quá nhỏ*, cần phải bù cho chiều rộng lưỡi cưa.

Thao tác này rất đơn giản, có thể thực hiện dễ dàng, hầu như không cần suy nghĩ. *Bán kính* lưỡi cắt (hoặc chiều rộng lưỡi cưa) *được bù* trước và trong khi cắt. Tương tự sự phác thảo hình dạng trên tấm gỗ, biên dạng của chi tiết gia công, *là biên dạng được bù với bán kính dao cắt*.

Quỹ đạo dao do tâm dao cắt tạo ra *luôn luôn* duy trì khoảng cách không đổi từ biên dạng (đường bao) chi tiết. Kiểu quỹ đạo này được gọi là quỹ đạo dao *khoảng cách cân bằng*, nghĩa là "khoảng cách không đổi". Hình 29.5 minh họa bản vẽ với quỹ đạo dao khoảng cách cân bằng.



Hình 29.5. *Quỹ đạo dao cân bằng. Cần biết các tọa độ tâm dao*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Câu hỏi lúc này là – các tọa độ đã tính toán và lưu trong bảng nêu trên sẽ có tác dụng như thế nào? Chúng có hữu ích không? *Có*. Có thể sử dụng chúng trong chương trình không? *Không*. Cần xem xét vài điều kiện bổ sung.

Trong chương trình, tập hợp các điểm *cũ* sẽ được dùng để tính toán tập hợp các điểm *mới*. Bạn hãy tìm các điểm dễ tính toán và thiết lập chúng trước.

Ví dụ, các tọa độ XY của điểm P1. Dễ dàng nhận thấy điểm P1 có giá trị của bán kính dao cắt theo chiều X âm và giá trị bán kính dao theo chiều Y âm so với P1 cũ. Giá trị thực của các điểm *bất kỳ* đều không thể tính toán, *nếu không biết trước bán kính dao cắt.*

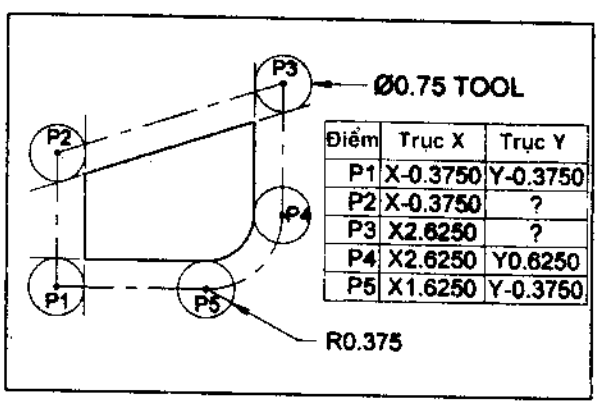
Bán kính dao cắt

Bán kính danh định của dao cắt luôn luôn được biết rõ hoặc đo đạc. Đối với chi tiết đòi hỏi độ chính xác cao, cần biết bán kính dao hầu như 100%, chẳng hạng với dung sai 0.0001" (0.0025 mm = 2.5 μm). Điều này là rất khó đối với các dao được mài lại, đã sử dụng vài lần, dao có kích thước nhỏ hơn hoặc lớn hơn vì lý do nào đó. Trong mọi trường hợp, điều này có nghĩa là đường tâm dao yêu cầu biết chính xác bán kính dao vào thời điểm lập trình.

Tính toán các điểm tâm

Các điểm tọa độ minh họa trên Hình 29.5, biểu diễn tâm bán kính dao tại từng điểm thay đổi biên dạng. Yêu cầu kế tiếp là kích cỡ bán kính dao. Tập hợp tọa độ mới gồm năm điểm có thể được thiết lập dễ dàng. Ví dụ, dao cắt mới Ø.750 có bán kính .375.

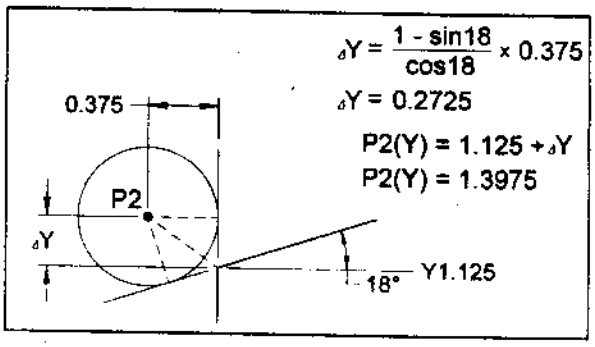
Có thể "đọc" được ưu điểm nào từ minh họa một cách trực tiếp mà không cần tính toán lượng giác? Bạn hãy quan sát và đánh giá Hình 29.6, có thể dễ dàng nhận biết 8 trong 10 giá trị cần thiết.



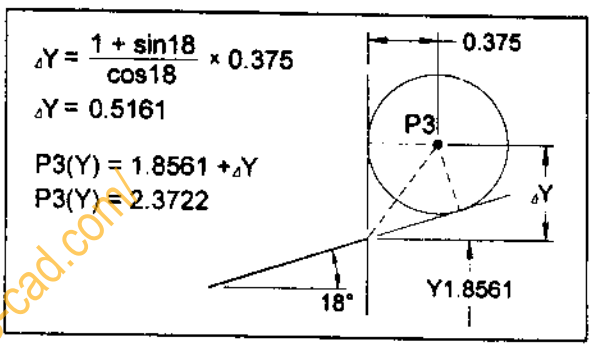
Hình 29.6. Các điểm thay đổi biên dạng trên quỹ đạo tâm dao.

Để biết đầy đủ các giá trị tọa độ tâm dao, cần tính toán các giá trị Y tại P2 và P3. Bạn có thể bắt đầu với điểm P2.

Hình 29.7 minh họa chi tiết tính toán điểm P2. Tính toán tương tự cho điểm P3 được nêu trên Hình 29.8.



Hình 29.7. Tính toán điểm tâm dao cắt P2



Hình 29.8. Tính toán điểm tâm dao cắt P3

Với kết quả này bạn đã biết mọi giá trị tọa độ XY của các điểm tâm dao xung quanh biên dạng chi tiết. Các điểm đó được sắp xếp theo thứ tự gia công và là thứ tự trong chương trình. Không chỉ các vị trí điểm mà còn các mã G, mã M, tốc độ cắt, và các dữ liệu khác.

Thời điểm này vẫn còn quá sớm để viết chương trình. Bảng dưới đây *chỉ* liệt kê các điểm tọa độ mới, biểu thị tâm dao cắt Ø.750.

Số điểm	Tọa độ X	Tọa độ Y
P1	X-0.375	Y-0.375
P2	X-0.375	Y1.3975
P3	X2.625	Y2.3722
P4	X2.625	Y0.625
P5	X1.625	Y-0.375

Trên các công thức tính toán (Hình 29.7, 30.8) có số 1, đây là giá trị của *sin* 90^0, trước Y còn có ký hiệu tam giác nhỏ, chữ Hy Lạp đọc là "delta", có nghĩa là số gia, vector, hoặc khoảng cách.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Các ví dụ nêu trên là điển hình đối với phương pháp lập trình được dùng trên các bộ điều khiển số kiểu cũ. Các bộ điều khiển này (NC, thay vì CNC), hoàn toàn không có tính năng bù bán kính dao. Quỹ đạo dao được xác định theo phương pháp tính toán các điểm thay đổi biên dạng *với bán kính dao cắt có hiệu lực*. Phương pháp lập trình này rất tốn thời gian, tăng khả năng sai sót lập trình, giảm tính linh hoạt trong khi gia công. Ngay cả sự khác biệt giữa bán kính dao lập trình và bán kính dao thực cũng đòi hỏi, tính toán lại, hiệu chỉnh chương trình và tạo ra băng đục lỗ mới (khi đó chưa có bộ nhớ CNC). Với sự phát triển công nghệ điều khiển số và sự bổ sung máy tính cho hệ thống điều khiển (hệ thống CNC hiện đại), phương pháp bù dao cắt cho phép đơn giản hóa rõ rệt quá trình lập trình gia công chi tiết.

Các kiểu bù bán kính dao

Khi công nghệ CNC phát triển, phương pháp bù bán kính dao cũng phát triển theo. Sự phát triển này đã trải qua ba giai đoạn. Ngày nay, chúng được gọi là ba kiểu bù bán kính dao *Kiểu A*, *Kiểu B*, và *Kiểu C*:

- Bù kiểu A – kiểu cũ nhất – sử dụng các vector đặc biệt trong chương trình để thiết lập chiều cắt gọt (G39, G40, G41, G42).
- Bù kiểu B – kiểu cũ – chỉ sử dụng G40, G41, và G42 trong chương trình không có đặc tính dự đoán trước. Bù kiểu B có thể bị cắt gọt quá mức.
- Bù kiểu C – hiện đại- chỉ sử dụng G40, G41, và G42 trong chương trình, nhưng có tính năng dự đoán trước. Bù kiểu C tránh được sự cắt gọt quá mức.

Bù bán kính dao *Kiểu C* – kiểu *dự đoán trước* (còn gọi là kiểu *giao cắt*), được dùng trên mọi hệ thống CNC hiện đại, không cần gọi rõ là *Bù Kiểu C*, do hiện chưa có các kiểu khác.

Định nghĩa và ứng dụng

Bù bán kính dao là tính năng của hệ điều khiển cho phép lập trình biên dạng không cần biết chính xác đường kính (bán kính) dao. Tính năng đặc biệt này thực hiện mọi tính toán cần thiết về các điểm thay đổi biên dạng, dựa trên ba nhóm dữ liệu:

- Các điểm biên dạng trên bản vẽ
- Chiều chuyển biệt của chuyển động dao cắt
- Bán kính dao được lưu trong hệ thống điều khiển

Trong lập trình thực tiễn – và gia công – tính năng này cho phép nhà lập trình CNC viết chương trình không cần biết đường kính chính xác của dao vào thời điểm lập trình, cho phép người vận hành CNC điều chỉnh, tinh chỉnh, kích cỡ dao trong hệ thống điều khiển (danh định, quá cỡ hoặc nhỏ hơn cỡ chuẩn), trong khi gia công thực tế. Trong thực tế, sử dụng bù bán kính dao (bù bán kính mũi dao trên máy tiện) cần được xem xét theo các yếu tố:

- Chưa biết kích cỡ chính xác của bán kính dao
- Điều chỉnh do mòn dao
- Điều chỉnh độ lệch dao
- Gia công thô và gia công tinh
- Duy trì dung sai gia công

...

Mọi gia công biên dạng đều yêu cầu xem xét bán kính dao.

Vào thời điểm này, một số ứng dụng có thể chưa thật rõ ràng, nhưng dần dần bạn sẽ hiểu vấn đề sâu sắc hơn. Các gợi ý dưới đây chỉ là một số khả năng của tính năng bù bán kính dao tự động.

KỸ THUẬT LẬP TRÌNH

Để lập trình bán kính dao trong chế độ bù, bạn cần biết ba nhóm dữ liệu đã nêu ở phần trên:

- Các điểm biên dạng trên bản vẽ.
- Chiều chuyển biệt của chuyển động dao cắt.
- Bán kính dao được lưu trong hệ điều khiển.

Các dữ liệu này là *nguồn dữ liệu* thực. Máy tính chỉ làm việc với dữ liệu do người dùng cung cấp. Chương này giả thiết mọi dữ liệu về các điểm thay đổi biên dạng dựa trên bản vẽ – Các tọa độ XY.

Chiều chuyển động cắt

Mỗi khi lập trình quỹ đạo dao trong hoặc ngoài, đều cần chọn một trong *hai* chiều. Cho đến phần này, các chiều được gọi là *thuận* (CW) hoặc *ngược chiều kim đồng hồ* (CCW). Chiều chuyển động tổng quát dựa trên chuyển động của bàn máy (phay) hoặc dụng cụ cắt (tiện). Cần làm rõ hai nhóm rất khác nhau – nhóm cần khảo sát – chuyển động của bàn máy hay chuyển động của dao? Bạn cần nhớ, *bất kể kiểu máy CNC*, luôn luôn tuân thủ một quy tắc trong lập trình CNC:

Trong lập trình CNC, luôn luôn xét dụng cụ cắt chuyển động xung quanh chi tiết, không xét chuyển động khác.

Câu này hoàn toàn đúng với máy tiện, *nhưng cũng đúng với trung tâm gia công CNC*,

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

dù không thật lễ phép khi nói như vậy, cũng đúng với các máy CNC khác, chẳng hạn máy gia công tia lửa điện EDM, máy cắt laser, máy cắt bằng tia nước, ...Để hiểu rõ vấn đề, cần xem xét kỹ hơn về chiều *thuận* (CW) và *ngược chiều kim đồng hồ* (CCW)

Trái hoặc phải - không phải CW hoặc CCW

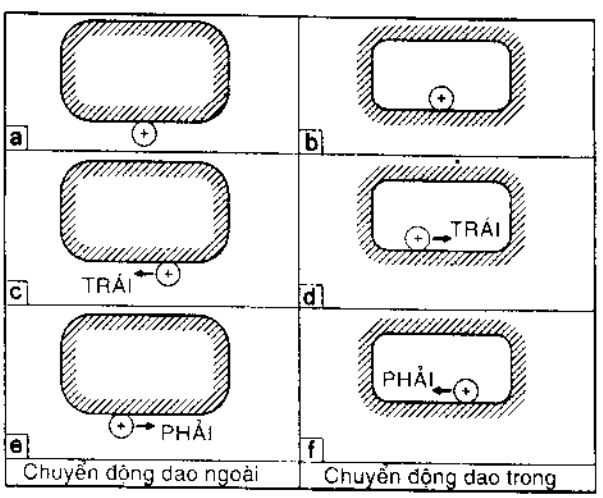
Điều đầu tiên cần chú ý là loại bỏ các thuật ngữ *thuận* và *ngược chiều kim đồng hồ* dễ đưa đến nhầm lẫn. Các thuật ngữ này chỉ sử dụng trong nội suy vòng tròn, không dùng trong bù bán kính dao. Để rõ ràng, bù bán kính dao chỉ dùng các thuật ngữ *trái* và *phải*.

Cũng như trong cuộc sống hàng ngày, khi gặp các thuật ngữ chỉ hướng trái và phải, bạn xác định vị trí đúng của vật thể theo *chiều quan sát* được thiết lập trước.Vật thể chuyển động được coi là bên trái hoặc bên phải vật thể cố định, tùy theo *chiều chuyển động của vật thể*.

Trong lập trình CNC cũng không có sự khác biệt. Quỹ đạo bù dụng cụ cắt được định vị theo bên *trái* hoặc bên *phải* biên dạng tĩnh tại, khi quan sát theo chiều quỹ đạo dao (Hình 29.9).

Minh họa này nêu rõ ba tùy chọn, dao chưa có chiều chuyển động, dao có chiều được chuyên biệt và định vị theo *bên trái biên dạng*, và dao theo chiều *bên phải biên dạng*.

Nói chung, trung tâm gia công CNC thường dùng *bù bên trái* do sự bù này tạo ra chế độ *phay ngược*, giả thiết dao tay phải tiêu chuẩn được dùng với lệnh quay M03. Có thể có trường hợp bù bên phải, tạo ra chế độ *phay thuận*. Chế độ này chỉ được áp dụng trong một số trường hợp đặc biệt. Điều nêu trên chỉ áp dụng cho các hệ thống phay, không áp dụng cho tiện.



Hình 29.9. *Chiều quỹ đạo dao theo biên dạng chi tiết tĩnh tại. (a-b) chưa có chiều chuyển động – chưa biết trái hay phải. (c-d) Dao định vị theo BÊN TRÁI biên dạng. (e-f) Dao định vị theo BÊN PHẢI biên dạng*

Các lệnh bù

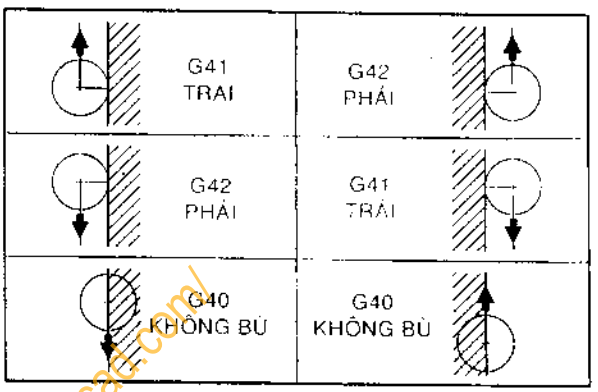
Để lập trình chế độ cắt (chiều cắt), có hai lệnh chuẩn bị cho phép chọn chiều bù bán kính dao cắt.

G41	Bù bán kính dao theo bên TRÁI chiều biên dạng
G42	Bù bán kính dao theo bên PHẢI chiều biên dạng

Lệnh G40 xóa chế độ G41 hoặc G42

G40	XÓA chế độ bù bán kính dao

Hình 29.10 minh họa ba lệnh bù bán kính.

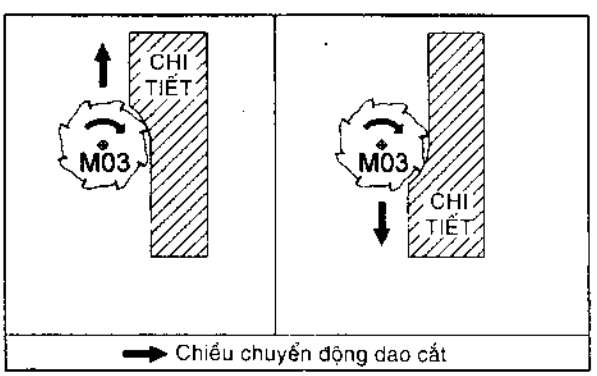


Hình 29.10. *Ứng dụng G40, G41, G42 cho quỹ đạo dao.*

http://www.me-cad.com/

Đối với nguyên công phay, lệnh G41 áp dụng cho chế độ *phay ngược*, G42 áp dụng cho *phay thuận*. Điều này *chỉ đúng* nếu trục chính quay với hàm M03 hoạt động (CW) và dao cắt là tay phải. Nếu dao cắt là tay trái, trục chính phải quay với hàm M04 hoạt động (CCW) và mọi nguyên tắc áp dụng cho bù bán kính dao *hoàn toàn ngược lại* với những điều nêu trên. Khi lệnh G40 có hiệu lực sẽ không có chế độ bù bán kính dao.

Hình 29.11 minh họa lệnh G41 trong chế



Hình 29.11. *Chế độ phay ngược và phay thuận đối với dao phải và lệnh quay trục chính M03.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

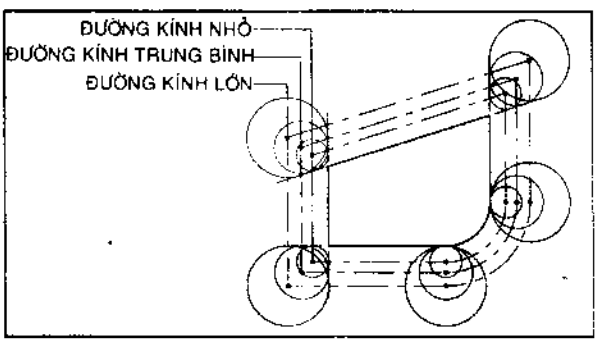
Nhiều tài liệu hay có tại : http://www.me-cad.com/

độ phay ngược và G42 với chế độ phay thuận. Chế độ phay ngược rất thông dụng trong phay CNC, đặc biệt là phay biên dạng.

Bán kính dao

Lợi ích của bù bán kính dao là cho phép lập trình quỹ đạo dụng cụ cắt tựa như biên dạng chi tiết là quỹ đạo dao cắt, không có nghĩa là bỏ qua bán kính dao trong khi lập trình. Câu hỏi logic vào giai đoạn này là nếu bán kính thực của dao cắt không được chuyên biệt trong chương trình, *vậy được chuyên biệt ở đâu?*

Trước hết, bạn hãy xem Hình 29.12, minh họa tác dụng của các bán kính dao cắt khác nhau áp dụng cho một biên dạng chi tiết.



Hình 29.12. *Tác dụng của bán kính dao cắt đối với quỹ đạo dao thực.*

Câu trả lời là *trong khu vực hệ thống điều khiển có tên là các xác lập bù*. Bạn đã biết các khu vực bù (*màn hình bù* trên bộ điều khiển). Các thuật ngữ này đã được sử dụng trong *Bù Vị Trí*, *Bù Chi Tiết*, và *Bù Chiều Dài Dao* (các Chương 16, 17, 18). Hiện tại là thời điểm nghiên cứu các chế độ bù điều khiển chi tiết hơn và nhấn mạnh quan hệ của chúng với chế độ bù bán kính dao. Dù chủ đề này phục vụ cho người vận hành CNC, nhưng cũng rất quan trọng đối với nhà lập trình.

Lịch sử các kiểu bù

Các bộ điều khiển Fanuc đã phát triển qua nhiều năm, do tính thông dụng và độ tin cậy của chúng, nhiều bộ điều khiển kiểu cũ vẫn còn được sử dụng trong các xưởng cơ khí. Để hiểu các chế độ bù và ứng dụng của chúng, điều quan trọng là biết bộ điều khiển Fanuc đang có *kiểu bù nào*. Nguyên tắc có lẽ là bộ điều khiển càng cũ, tính linh hoạt càng thấp và ngược lại. Bạn hãy chú ý thuật ngữ *tính linh hoạt*, không phải là *chất lượng* thấp hay cao. Các khác biệt được phân loại theo *Kiểu Nhớ Bù*. Hiện có ba kiểu nhớ trên các hệ thống Fanuc.

- ❑ Kiểu A – tính linh hoạt thấp
- ❑ Kiểu B – tính linh hoạt trung bình
- ❑ Kiểu C – tính linh hoạt cao

Bạn không nên nhầm lẫn các kiểu nhớ bù dao này với các kiểu bù bán kính dao. Các kiểu bù này xác định cách thức *bù chiều dài dao* và *bù bán kính dao* nhập vào hệ thống điều khiển. Các lệnh bù chi tiết G54 và G59 không bị ảnh hưởng.

Nhớ bù dao kiểu A

Bù dao *Kiểu A* là mức thấp nhất hiện có. Tính linh họa của kiểu này rất hạn chế, do xếp chung các giá trị chiều dài dao với các giá trị bán kính dao trong *một cột*. Do có chung dữ liệu cho hai chế độ bù khác nhau, Kiểu A thường được gọi là chế độ bù *chung*. Trong thực tiễn, điều này có nghĩa là giá trị bù chiều dài dao được lưu trong *cùng một* vùng đăng ký điều khiển với giá trị bán kính dao. Có thể sử dụng địa chỉ H *và/hoặc* D, điều này sẽ được trình bày ở phần sau. Các bộ điều khiển được trang bị kiểu nhớ bù dao này là kiểu kinh tế nhất trong nhóm của chúng.

Nhớ bù dao kiểu B

Bù *Kiểu A* chỉ có một cột màn hình, *Kiểu B* có *hai cột*. Hai cột này hoàn toàn *không* phải là hai cột riêng cho các giá trị chiều dài dao và các giá trị bán kính dao. Cột thứ nhất là *Bù Dạng Hình Học* và cột thứ hai là *Bù Mòn Dao*. Ngoài sự khác biệt này, *Kiểu B* vẫn là chế độ bù chung cho cả hai, giá trị bù chiều dài và giá trị bán kính dao. Chương trình sử dụng địa chỉ H *và/hoặc* D.

Nhớ bù dao Kiểu C

Nhóm bù *Kiểu C* có tính linh hoạt cao nhất. Đây là kiểu duy nhất tách biệt giữa chiều dài dao và bán kính dao, nhưng vẫn giữ sự phân biệt giữa *Bù Hình Học* và *Bù Mòn Dao* như Kiểu B. Điều đó có nghĩa là màn hình điều khiển có 2 + 2 cột, tổng cộng là *bốn* cột. Trong kiểu này các địa chỉ H và D được sử dụng với mục đích đặc thù của từng địa chỉ đó.

Bạn dễ dàng nhận biết kiểu bù khả dụng – chỉ cần quan sát màn hình bộ điều khiển. Hình 29.13 minh họa hình thức của từng *Kiểu Nhớ Bù* (tất cả đều ghi các giá trị zero). Hình thức thực tế có thể hơi khác, tùy theo kiểu loại bộ điều khiển.

Định dạng lập trình

Thông tin tối thiểu cung cấp cho hệ thống điều khiển trong chương trình CNC là lệnh bù G41 hoặc G42 luôn luôn kết hợp với địa chỉ H

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Offset No.	Offset
01	0.0000
02	0.0000
03	0.0000
...	...

Offset No.	Geometry	Wear
01	0.0000	0.0000
02	0.0000	0.0000
03	0.0000	0.0000
...	...	...

Offset No.	H-offset		D-offset	
	Geometry	Wear	Geometry	Wear
01	0.0000	0.0000	0.0000	0.0000
02	0.0000	0.0000	0.0000	0.0000
03	0.0000	0.0000	0.0000	0.0000
...	...	...	...	...

Hình 29.13. *Các kiểu nhớ bù dao Fanuc, A, B, C từ trên xuống.*

hoặc D, thường áp dụng trong khi chuyển động đơn trục (có thể chuyển động đa trục, nếu lập trình cẩn thận):

```
G41 X.. D..    hoặc...
G42 X.. D..    hoặc...
G41 Y.. D..    hoặc...
G42 Y.. D..
```

Phần sau trong chương này sẽ trình bày sự kết hợp hoặc loại bỏ chuyển động dao và số trục có thể sử dụng vào thời điểm bất kỳ. Trước hết cần giải quyết vấn đề sử dụng địa chỉ này và khi nào. Địa chỉ H hay địa chỉ D?

Địa chỉ H hoặc D

Với ba kiểu *Bù Nhớ Dao*, sẽ hợp lý nếu dự đoán các kiểu này có phương pháp lập trình hơi khác nhau, cho đến lúc này, điều đó là đúng.

Cả hai Kiểu A và B đều là bù chung, chỉ với một bộ đăng ký, các giá trị chiều dài dao được lưu chung với bài bán kính dao. Nói chung, Kiểu A và Kiểu B chỉ quan hệ với địa chỉ H. Điều đó có nghĩa là địa chỉ H được dùng với lệnh G43, cũng như các lệnh G41 hoặc G42. Nhiều dụng cụ cắt không yêu cầu bù bán kính dao trong chương trình, nhưng *mọi* dụng cụ cắt đều yêu cầu bù chiều dài dao. Nếu dao yêu cầu *cả* giá trị bù chiều dài dao *và* giá trị bù bán kính dao, cần phải sử dụng *hai giá trị bù khác nhau* từ cùng một khoảng bù trong chương trình và được lưu trong bộ đăng ký điều khiển. Đó là lý do các giá trị bù này được gọi là bù *chung*.

Ví dụ, dao lập trình T05 đòi hỏi cả hai giá trị bù, với các số khác nhau. Giải pháp là sử dụng số dao làm số bù chiều dài dao và tăng số đó theo các giá trị 20, 30, 40 ... để bù bán kính dao. Mục nhập đối với Kiểu A trên màn hình bù sẽ tương tự màn hình trên Hình 29.14:

Offset No.	Offset
...	...
05	-8.6640
...	...
35	0.3750

Hình 29.14. *Màn hình đăng ký bù chung đối với nhớ bù dao Kiểu A*

Đối với bù Kiểu B, sẽ có hai cột, nhưng vẫn là bù chung. Mục nhập trên màn Hình sẽ tương tự Kiểu A, được nêu trên Hình 29.15.

Offset No.	Geometry	Wear
...	...	...
05	-8.6640	0.0000
...	...	...
35	0.3750	0.0000

Hình 29.15. *Màn hình đăng ký bù chung đối với nhớ bù dao Kiểu B.*

Bù Kiểu C sẽ có hai cặp cột. Do chiều dài dao và bán kính dao có các cột riêng, *cùng một giá trị bù* có thể dùng cho cả hai – không cần số gia 20, 30, 40 ... Trong trường hợp này, địa chỉ H dùng cho bù chiều dài dao và địa chỉ D dùng cho bù bù bán kính dao. Hình 29.16 minh họa mục nhập về logic là tương ứng với Kiểu A và Kiểu B.

Offset No.	H-offset		D-offset	
	Geometry	Wear	Geometry	Wear
...	...	...	...	...
...	...	...	...	...
05	-8.6640	0.0000	0.3750	0.0000
...	...	...	...	...

Hình 29.16. *Màn hình đăng ký bù đặc thù đối với nhớ bù dao Kiểu C.*

Bù hình học và bù mòn dao

Tương tự ứng dụng của bù hình học và bù mòn dao đối với *bù chiều dài dao* (Chương 18), các nguyên tắc tổng quát đó cũng có thể được áp dụng cho bù bán kính dao.

Các giá trị bù nhập vào cột bù *Geometry* (Hình học) phải là bán kính dao danh định. Trong các ví dụ dưới đây, bạn sẽ dùng dao Ø.750, với bán kính 0.375 in. Đây là giá trị danh định và là giá trị nhập vào cột bù *Geometry* (Hình học). Cột bù *Wear* (mòn dao) chỉ dùng cho các điều chỉnh, hoặc tinh chỉnh,

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

liên quan với kích thước danh định được yêu cầu khi gá lắp hoặc trong khi gia công. Không có cột riêng cho điều chỉnh hoặc tinh chỉnh đối với bù Kiểu A. Vẫn có thể thực hiện các điều chỉnh, nhưng sự khác hệt duy nhất là giá trị trong cột sẽ luôn luôn thay đổi với từng điều chỉnh, kể cả khi cột đó biểu thị bán kính dao.

ÁP DỤNG BÙ BÁN KÍNH DAO

Cho đến phần này bạn đã biết mọi dữ liệu lập trình cần thiết để áp dụng bù bán kính dao trong chương trình CNC. Ứng dụng thực tế, phương pháp sử dụng chế độ bù trong chương trình và các kỹ thuật ứng dụng, sẽ được trình bày trong phần kế tiếp. Có *bốn yếu tố chính* để sử dụng thành công tính năng bù bán kính dao:

1. Phương pháp khởi động chế độ bù
2. Phương pháp thay đổi chế độ bù
3. Phương pháp kết thúc chế độ bù
4. Điều cần quan sát giữa bắt đầu và kết thúc chế độ bù

Các yếu tố trên đều rất quan trọng, sẽ được trình bày theo thứ tự.

Phương pháp khởi động

Sự khởi động bù bán kính dao cắt đòi hỏi nhiều hơn là chỉ sử dụng G41 X.. D.. trong chương trình. Khởi động chế độ bù có nghĩa là tuân thủ hai nguyên tắc cơ bản, một số khảo sát và quyết định quan trọng. Nguyên tắc số một khá đơn giản – liên quan với vị trí khởi đầu của dao cắt:

Luôn luôn chọn vị trí khởi đầu của dao cắt cách xa biên dạng, trong không gian trống

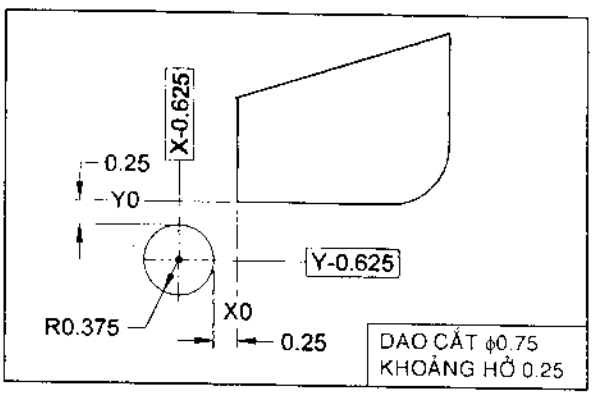
Nguyên tắc số hai cũng đơn giản, dựa trên sự tuân thủ nguyên tắc thứ nhất

Luôn luôn áp dụng bù bán kính dao cùng với chuyển động dao

Hai nguyên tắc này không phải là bắt buộc, chúng có thể bị phá vỡ. Đề nghị ở đây là hãy tuân theo các nguyên tắc đó cho đến khi tìm được phương pháp tốt hơn. Khi chọn vị trí khởi đầu dao cắt, bạn cần giải quyết vài vấn đề :

- ❑ Đường kính dao cắt cần dùng là bao nhiêu?
- ❑ Các khoảng hở cần thiết là bao nhiêu?
- ❑ Chiều chuyển động của dao cắt?
- ❑ Có khả năng xảy ra va chạm không?
- ❑ Nếu cần thiết, có thể sử dụng bao nhiêu đối với đường kính khác?
- ❑ Lượng dư gia công (lượng cắt gọt) là bao nhiêu?

Bản vẽ trên Hình 29.2 sẽ được dùng cho ví dụ kế tiếp và bù bán kính dao đã tính toán ở phần trên sẽ được áp dụng. Để kích hoạt chế độ bù, dao cắt phải *cách xa* khu vực cắt gọt thực, ở trong không gian trống. Dao có đường kính Ø.750, chế độ cắt là phay ngược, và khoảng hở là .250 cách biên dạng. Với các số liệu này, vị trí khởi đầu được tính toán là X-0.625Y-0.625. Hình 29.17 minh họa vị trí khởi đầu này, thỏa mọi nguyên tắc và các vấn đề nêu trên.



Hình 29.17. *Vị trí khởi đầu của dao trước khi áp dụng bù bán kính.*

Vị trí nêu trên không phải là vị trí thích hợp duy nhất, có thể còn có các khả năng khác. Chú ý, dao ở vị trí X-0.625Y-0.625 là *không* bù, các tọa độ là điểm *tâm* của dao. Sau khi thiết lập vị trí khởi đầu, có thể viết vài block cho chương trình :

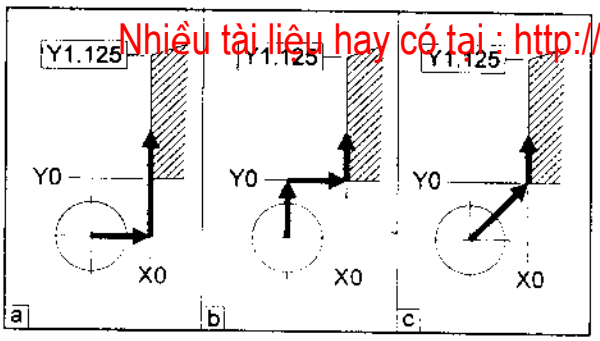
```
O2901 (BẢN VẼ Hình 29.2)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X-0.625 Y-0.625 S920 M03
N4 G43 Z1.0 H01
N5 G01 Z-0.55 F25.0 M08 (FOR 0.5 PLATE THICK)
N6 ....
```

Để tăng tính an toàn, sự tiến đến chiều sâu Z-0.55 (dựa trên tấm dày 1/2 inch) được chia thành hai chuyển động, dù dao đang an toàn trong không gian trống. Sau khi đạt được chiều sâu này, có thể lập trình chuyển động thứ nhất. Chiều dao cắt là bên trái chi tiết (phay ngược) và sử dụng lệnh G41. Chuyển động dao xung quanh chi tiết từ bên trái, có nghĩa là điểm đích thứ nhất trên chi tiết là vị trí X0Y1.125. Tuy nhiên, không thể đến vị trí này một cách trực tiếp, do bên trái của chi tiết cũng cần được gia công. Điều đó có nghĩa là dao phải đến vị trí X0 trước. Quyết định kế tiếp là chọn vị trí Y để đến điểm đích. Nói chung, điều này được thực hiện bằng cách lập trình chuyển động *hướng vào* hoặc chuyển động *tiến vào*.

Hình 29.18 minh họa một số khả năng, tất cả đều đúng và đều đến vị trí X0Y1.125. Khả

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 29.18. Các chuyển động hướng vào có thể áp dụng cho chế độ bù bán kính dao.

năng nào là tốt nhất? Dưới đây là một số chọn lựa:

Lựa chọn (a) khá đơn giản – dao đến X0 trước và lệnh bù bán kính dao được kích hoạt *trong khi* thực hiện chuyển động này. Sau đó, dao tiếp tục đến điểm đích thứ nhất (Y1.125), sẵn sàng trong chế độ bù.

Hai chuyển động này xuất hiện trong chương trình như sau :

```
N.. G01 G41 X0 D01 F15.0
N.. Y1.125
N.. ...                          (P2)
```

Lựa chọn (b) về kỹ thuật là đúng, nhưng đòi hỏi ba chuyển động, trong khi chỉ hai chuyển động là đủ. Điều đó sẽ không được chọn trong chương trình, dù chương trình sẽ vẫn đúng :

```
N.. G01 G41 Y0 D01 F15.0
N.. X0
N.. Y1.125                       (P2)
N.. ....
```

Khả năng cuối cùng, lựa chọn (c), cũng đơn giản và chỉ cần hai chuyển động:

```
N.. G01 G41 X0 Y0 D01 F15.0
N.. Y1.125                       (P2)
N.. ...
```

Trong cả ba khả năng, bù bán kính dao được khởi động *cùng với chuyển động thứ nhất*, trong khi vẫn cách xa biên dạng chi tiết thực. Do lựa chọn (c) kết thúc *trên* chi tiết, chọn lựa chọn (a) là phương pháp lập trình thích hợp hơn. Sự kết hợp (a) và (c) cũng là lựa chọn tốt, với đích trục Y ở phần âm.

Sau khi kích hoạt chế độ bù, các điểm thay đổi biên dạng có thể được lập trình dọc theo chi tiết và máy tính điều khiển sẽ hoạt động bằng cách liên tục duy trì sự bù dao một cách hợp lý. Chương trình O2901 có thể mở rộng đến điểm P5 (Hình 29.3).

```
O2901 (BẢN VẼ Hình 29.2)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X-0.625 Y-0.625 S920 M03
N4 G43 Z1.0 H01
N5 G01 Z-0.55 F25.0 M08  (FOR 0.5 PLATE THICK)
N6 G41 X0 D01 F15.0              (START OFFSET)
N7 Y1.125                                (P2)
N8 X2.25 Y1.8561                         (P3)
N9 Y0.625                                (P4)
N10 G02 X1.6225 Y0 R0.625                (P5)
N11 G01 X...
```

Tại block N10, dao đạt đến điểm kết thúc bán kính 0.625. Biên dạng vẫn chưa hoàn tất, cần gia công cạnh đáy dọc theo trục X. Câu hỏi là cắt đến đâu và thời điểm xóa bù bán kính dao?

Đây là phần cắt gọt cuối cùng trên chi tiết, do đó cần được gia công *trong khi chế độ bù vẫn có hiệu lực*. Dao có thể kết thúc ở X0, nhưng đây là vị trí không thực tế, dao cần dịch chuyển hơi xa hơn, vẫn chỉ theo trục X. Xa hơn bao nhiêu? Tại sao không đến X-0.625, vị trí khởi động ban đầu? Đây không chỉ là vị trí khoảng hở khả dụng, mà còn là vị trí an toàn nhất, tin cậy và ổn định. Block N11 sẽ là :

```
N11 G01 X-0.625
```

Khi đó dao cắt rời xa biên dạng chi tiết và không cần bù bán kính dao. Lệnh bù sẽ được xóa, sự tóm tắt quá trình khởi động có lẽ sẽ giúp bạn hiểu vấn đề rõ hơn.

Ví dụ nêu trên là trường hợp phổ biến nhưng chưa hẳn là điển hình. Nhà lập trình cần có dao cắt. Trong phạm vi hợp lý, dao ∅.750 và dao ∅.875 không quá cách xa nhau, *trừ các khoảng hở*. Khoảng hở 0.250 được chọn cho bán kính .375. Điều đó có nghĩa là chương trình vẫn thích hợp đối với các dao cắt có đường kính đến ∅1.25. Người vận hành CNC có sự linh hoạt, do sự thay đổi duy nhất là giá trị bù D01 trong bộ đăng ký bù của hệ điều khiển. Tốc độ trục chính và tốc độ cắt có thể được điều chỉnh, nếu cần. Phần sau sẽ trình bày điều xảy ra khi áp dụng bù bán kính dao.

Nguyên tắc chung để thiết lập điểm khởi đầu là chọn với khoảng hở lớn hơn bán kính *lớn nhất* của dao có thể sử dụng. Có thể tăng khoảng hở này để dễ thoát phoi hoặc đối với dao lớn hơn đường kính trung bình. Để hoàn tất chương trình, bạn hãy xem xét các phương pháp xóa bù bán kính dao.

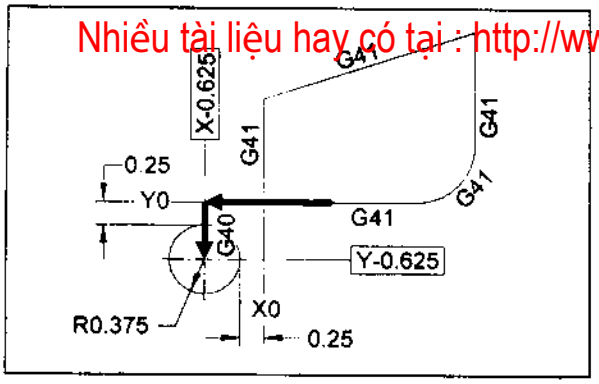
Xóa lệnh bù

Chuyển động *hướng vào* được sử dụng khi khởi động chế độ bù bán kính dao. Để xóa chế độ bù cần dùng chuyển động *hướng ra*. Chiều dài hướng ra (bằng chiều dài hướng vào) hơi lớn hơn hoặc ít nhất là bằng bán kính dao. Các chuyển động hướng vào và hướng ra còn được gọi là chuyển động *tiến vào* và *lùi ra*.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 29.19. Xóa bù bán kính dao – chương trình O2901

Vị trí an toàn nhất để xóa bù bán kính dao, đối với máy bất kỳ, là cách xa biên dạng mới hoàn tất. Vị trí này luôn luôn ở khoảng không gian trống. Vị trí khởi đầu cũng có thể là vị trí kết thúc. Hình 29.19 minh họa sự xóa bù trong ví dụ nêu trên.Chương trình O2901 sẽ như sau:

```
O2901 (BẢN VẼ Hình 29.2)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X-0.625 Y-0.625 S920 M03
N4 G43 Z1.0 H01
N5 G01 Z-0.55 F25.0 M08  (FOR 0.5 PLATE THICK)
N6 G41 X0 D01 F15.0              (START OFFSET)
N7 Y1.125                                (P2)
N8 X2.25 Y1.8561                         (P3)
N9 Y0.625                                (P4)
N10 G02 X1.625 Y0 R0.625                 (P5)
N11 G01 X-0.625
N12 G00 G40 Y-0.625              (CANCEL OFFSET)
N13 Z1.0 M09
N14 G28 X-0.625 Y-0.625 Z1.0
N15 M30
%
```

Cuối cùng, chương trình O2901 cũng hoàn tất. Không cần thay đổi chiều chuyển động dao – sự thay đổi này rất ít khi xảy ra, ít nhất là khi gia công biên dạng trên máy phay. Do đôi khi có thể cần thay đổi chiều, bạn nên biết phương pháp lập trình thay đổi chiều dao.

Thay đổi chiều dao cắt

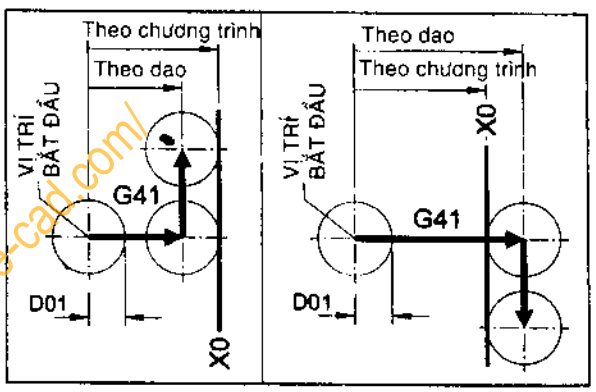
Trong khi phay bình thường, rất ít khi cần thay đổi chiều bù dao cắt từ trái sang phải hoặc ngược lại. Nếu điều này trở nên cần thiết, phương pháp thông thường là đổi từ một chế độ sang chế độ khác *nhưng không* xóa lệnh G40. Phương pháp này ít được dùng trong nguyên công phay, do chuyển từ G41 sang G42 cũng sẽ đổi từ phay ngược sang phay thuận. Tuy nhiên, phương pháp đó rất thông dụng trong lập trình tiện CNC.

CƠ CHẾ HOẠT ĐỘNG CỦA BÙ BÁN KÍNH

Có khả năng lập trình từ các ví dụ cho trước là phương pháp học tập khá tốt. Học theo các ví dụ hoặc phương trình mẫu có thể giải quyết được nhiều trường hợp, tuy nhiên đó mới chỉ là bước đầu, điều cơ bản là *hiểu rõ* mọi nguyên lý lập trình, chẳng hạn các nguyên lý bán kính dao. Phương pháp khởi động là sự bắt đầu tốt. Vấn đề kế tiếp là điều gì xảy ra trong quá trình chuyển động dao trong block N6?

```
N6 G41 X0 D01 F15.0
```

Điều này không đơn giản như mới thoạt nhìn. Bạn không thể chỉ đánh giá một block, chẳng hạn N6, để biết chính xác những điều xảy ra. Nhà lập trình cần hiểu rõ điều bộ điều khiển thực hiện. Máy tính không biết suy nghĩ chỉ thực thi các lệnh lập trình và tuân theo các lệnh đó một cách máy móc. Block N6 là lệnh: *Dịch chuyển đến X0, áp dụng giá trị bán kính lưu trong D01 theo bên trái, trong khi chuyển động tuyến tính 15 in/min*. Đây là lệnh chương trình đối với hệ thống điều khiển. Dao sẽ dừng lại ở đâu? Bạn hãy quan sát Hình 29.20.



Hình 29.20. Sự khởi động chuyển động của dao cắt trong chế độ bù bán kính.

Có *hai* khả năng, và chúng đều đúng. Cả hai đều bù dao cắt theo bên trái của vị trí đích X0. Các điều kiện chuyên biệt trong block N6 đều thỏa – dụng cụ cắt chuyển động đến X0 như mong muốn, chế độ bù được kích hoạt theo *bên trái* biên dạng chi tiết, *trong* quá trình chuyển động, sử dụng giá trị bán kính đã lưu trong D01. Vậy vấn đề là gì?

Tình huống này là *không rõ ràng*. Có *hai khả năng*, nhưng chỉ cần một. Khả năng nào? Đối với ví dụ này, đó là phần bên trái của Hình 29.20, dao chuyển động theo chiều Y+, khi áp dụng bù bán kính. *Đây là yếu tố cốt lõi*. Bộ điều khiển *phải* biết chiều chuyển động tiếp sau block G41 hoặc G42. Bạn hãy quan sát hai cách viết chương trình :

➲ Ví dụ 1 – Hình 29.21 bên trái

Vị trí đích kế tiếp sau N6 là chiều *dương* trục Y:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

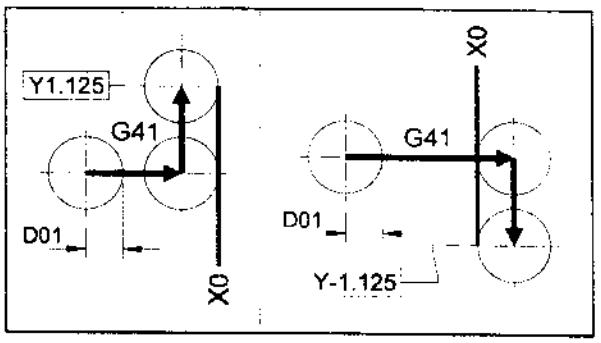
```
N3 G90 G54 G00 X-0.625 Y-0.625 S920 M03
...
N6 G41 X0 D01 F15.0                (START OFFSET)
N7 Y1.125          (POSITIVE Y-MOTION FOLLOWS)
```

➲ **Ví dụ 2 – Hình 29.21 bên phải**

Vị trí đích kế tiếp sau N6 là chiều *âm* trục X:

```
N3 G90 G54 G00 X-0.625 Y-0.625 S920 M03
...
N6 G41 X0 D01 F15.0                (START OFFSET)
N7 Y-1.125         (NEGATIVE Y-MOTION FOLLOWS)
```

Trong cả hai trường hợp, nội dung của block N6 là như nhau, nhưng khác nhau về chuyển động sau block N6 (Hình 29.21).



Hình 29.21. *Tầm quan trọng của chuyển động dao kế tiếp đối với bù bán kính dao. Chiều Y+ kế tiếp sang bên trái, chiều Y- kế tiếp sang bên phải.*

Kiểu bù dự đoán trước

Một mình block N6 không đủ dữ liệu để áp dụng thành công chế độ bù bán kính. Hệ thống điều khiển phải luôn luôn biết rõ *chuyển động kế tiếp*, thực chất là *chiều của chuyển động kế tiếp*. Hệ điều khiển xử lý yêu cầu đó như thế nào? Các bộ điều khiển sử dụng bù bán kính *Kiểu C* có tính năng được gọi là kiểu "*dự đoán trước*" về bù bán kính dao.

Tính năng *dự đoán trước* dựa trên nguyên tắc *đệm* hoặc *đọc trước*. Nói chung, bộ xử lý điều khiển mỗi lần thực thi một block. Sẽ không có chuyển động do block đệm (block kế tiếp) gây ra.

Dưới đây là tóm tắt chuỗi thứ tự các sự kiện:

- Bộ điều khiển trước hết sẽ đọc block chứa sự khởi động bù bán kính dao (block N6)
- Bộ điều khiển tìm tình huống chưa rõ ràng, và không xử lý block đó.
- Bộ điều khiển xử lý block kế tiếp (block N7) để tìm chiều chuyển động dao được lập trình kế tiếp.
- Trong khi "đọc block kế tiếp", hoàn toàn không có chuyển động, bộ điều khiển sẽ chỉ *đăng ký* chiều hướng đến điểm chính và áp dụng bù bán kính theo phía đúng của biên dạng chi tiết, trong block khởi động (N6 trong ví dụ)

Kiểu dự đoán trước bù chi tiết là rất hiện đại trong phần mềm, giúp cho sự lập trình trở nên dễ dàng hơn. Như có thể dự đoán, bạn cần chú ý một số tình huống.

Nguyên tắc đối với bù bán kính kiểu dự đoán trước

Bạn hãy quan sát chương trình mẫu dưới đây, không liên quan với các ví dụ đã nêu ở phần trước.

➲ **Ví dụ – block đơn KHÔNG CÓ CHUYỂN ĐỘNG**

```
N17 G90 G54 G00 X-0.75 Y-0.75 S800 M03
...
N20 G01 X0 D01 F17.0               (START OFFSET)
N21 M08                        (NO MOTION BLOCK)
N22 Y2.5                         (MOTION BLOCK)
...
```

Điều khác biệt trong cấu trúc chương trình này là gì? Bạn hãy bỏ qua lý do về hàm chất làm nguội ON (hoạt động) trong block N21. Thực tế là *không có chuyển động trục* trong block N21, đây là block để hệ thống điều khiển *dự đoán trước* chiều của chuyển động dao kế tiếp. Bạn hãy xem thêm một chương trình mẫu:

➲ **Ví dụ – hai block KHÔNG CÓ CHUYỂN ĐỘNG**

```
N17 G90 G54 G00 X-0.75 Y-0.75 S800 M03
...
N20 G01 X0 D01 F17.0               (START OFFSET)
N21 M08                        (NO MOTION BLOCK)
N22 G04 P1000                  (NO MOTION BLOCK)
N23 Y2.5                         (MOTION BLOCK)
...
```

Điều bất tiện, nhưng không sai, có lẽ là ở đây có *hai block liên tiếp* ngay sau lệnh bù bán kính dao – hai block liên tiếp *không chứa chuyển động bất kỳ*.

Cả hai ví dụ đều biểu thị chương trình có thể khá tốt nếu không áp dụng bù bán kính. Với lệnh bù có hiệu lực, cấu trúc chương trình đó có thể gây ra các vấn đề. Những bộ điều khiển với tính năng "dự đoán trước" có thể chỉ xem trước một hoặc hai block. Nếu bộ điều khiển có tính năng đó sẽ *luôn luôn xem trước một block*. Có thể có hai hoặc nhiều block cần xem trước.

Điều này hoàn toàn phụ thuộc vào các tính năng của bộ điều khiển, và không phải mọi bộ điều khiển đều như nhau. Dưới đây là vài đề nghị cơ bản :

- Nếu bộ điều khiển có tính năng bù bán kính dao kiểu xem trước, nhưng chưa biết số lượng block có thể được xử lý trước, bạn hãy giả thiết chỉ dự đoán trước một block.
- Viết một chương trình kiểm tra để tìm số lượng block bộ điều khiển có thể đọc trước.
- Sau khi khởi động lệnh bù bán kính dao trong chương trình, bạn hãy cố gắng không đưa các block không có

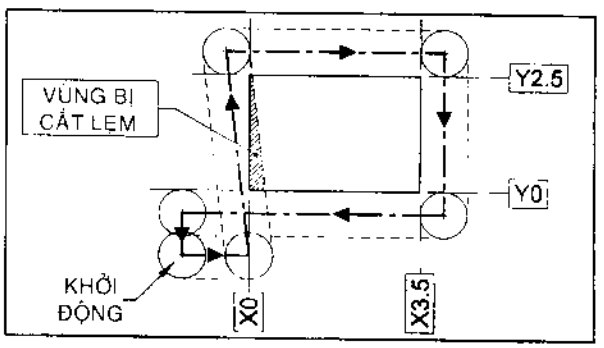
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

lệnh chỉ huy động vào chương trình mới. Bạn có thể sắp xếp lại chương trình đó.

Bạn cần nhớ, bộ điều khiển xử lý dữ liệu nhập chương trình theo các nguyên tắc trong phần mềm. Trước hết phải cung cấp dữ liệu nhập đúng, ở dạng chương trình chính xác.

Sẽ xảy ra dạng đáp ứng nào nếu lập trình sai đối với lệnh bù bán kính? Có lẽ kết quả sẽ là phế phẩm. Nếu hệ điều khiển không thể tính toán vị trí bù dao cắt, hệ đó sẽ hoạt động tựa như bù bán kính hoàn toàn không được lập trình. Điều đó có nghĩa là, chuyển động ban đầu của dao sẽ hướng đến X0 với *tâm dao cắt*. Khi thông tin cần thiết này đi vào bộ điều khiển, sự bù sẽ được áp dụng, thường là quá trễ, sau khi dao cắt đã ăn vào chi tiết. Kết quả sai hầu như sẽ là tất yếu. Chương trình *sai* theo kiểu đó được minh họa trên Hình 29.22:



Hình 29.22. *Quỹ đạo dao bị sai lệch do cấu trúc chương trình sai – Chương trình O2902*

Trong chương trình O2902 có hai block *không chuyển động* gây ra sự sai lệch quỹ đạo dao. Chúng xuất hiện sau khi áp dụng bù bán kính dao - trong các block N7 và N8. Nếu hệ điều khiển chỉ có thể xem trước một block, chương trình sẽ sai và quỹ đạo dao sẽ bị lệch (Hình 29.22).

```
O2902 (CHƯƠNG TRÌNH BÙ BÁN KÍNH SAI)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X-0.5 Y-0.5 S1100 M03
N4 G43 Z1.0 H01
N5 G01 Z-0.55 F20.0          (FOR 0.5 PLATE THICK)
N6 G41 X0 D01 F12.0                (START OFFSET)
N7 M08                          (NO MOTION BLOCK)
N8 G04 P1000                    (NO MOTION BLOCK)
N9 Y2.5                            (MOTION BLOCK)
N10 X3.5                           (MOTION BLOCK)
N11 Y0                             (MOTION BLOCK)
N12 G01 X-0.5                      (MOTION BLOCK)
N13 G00 G40 Y-0.5                (CANCEL OFFSET)
N14 Z1.0 M09
N15 G28 X-0.5 Y-0.5 Z1.0
N16 M30
%
```

Bộ điều khiển chỉ có thể đọc trước *một* hoặc *hai* block *sẽ không* xử lý chương trình O2902 một cách chính xác – chuyển động kế tiếp là trong block *thứ ba* khi lệnh bù có hiệu lực. Để tránh chuyển động dao không chính xác, bạn cần tránh cấu trúc chương trình chứa hơn một block không có chuyển động.

Bán kính dao cắt

Mọi dao phay đều có đường kính, và một nửa đường kính đó là bán kính dao. Với các dao mới, bạn luôn luôn biết bán kính đủ chính xác. Độ chính xác của bán kính phụ thuộc vào chất lượng dao, phương pháp gá lắp dao trong trục máy ... Độ lệch 0.001 – 0.002 inch có thể không gây ra vấn đề khi gia công thô, nhưng khi gia công tinh với độ chính xác cao, cần có dao chính xác hơn. Ngoài ra, cần có phương pháp *hiệu chỉnh* sự mòn dao, hoặc độ lệch dao. Toàn bộ điều này được thực hiện thông qua chỉ số bù D, được dùng làm chuẩn đối với giá trị bán kính thực, lưu trong bộ đăng ký điều khiển.

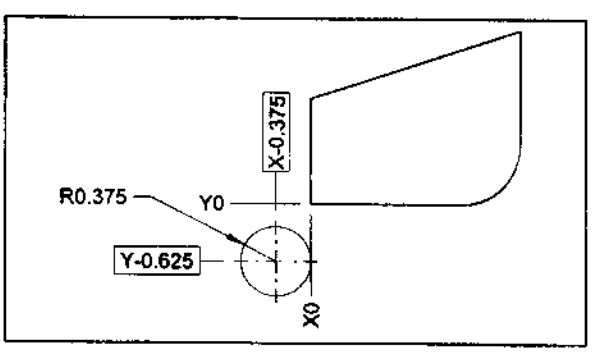
Một nguyên tắc đơn giản, nhưng rất cơ bản, giúp bảo đảm lượng bù bán kính dao chính xác:

Bán kính dao cắt luôn luôn nhỏ hơn chiều dài lập trình của hành trình dụng cụ cắt.

Ví dụ, trong chương trình O2902, vị trí khởi động dao là X-0.625, vị trí đích là X0. Điều đó có nghĩa là chiều dài lập trình của hành trình dao là .625. Bán kính đã chọn là .375, nhỏ hơn và tuân thủ đúng nguyên tắc nêu trên.

Còn có hai khả năng khác – một, bán kính dao *bằng* chiều dài lập trình của hành trình dao, và hai, bán kính dao *lớn hơn* chiều dài lập trình của hành trình dao.

Hình 29.23 minh họa vị trí khởi đầu của dao cắt có *cùng* chiều dài lập trình hành trình dao với bán kính dao. Điều này là được phép trong một số trường hợp nhưng không nên sử dụng. Lý do là sẽ giới hạn khoảng điều chỉnh có thể cần thực hiện đối với bán kính dao thực trong khi gia công.



Hình 29.23. *Vị trí khởi đầu của dao cắt bằng bán kính dao.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

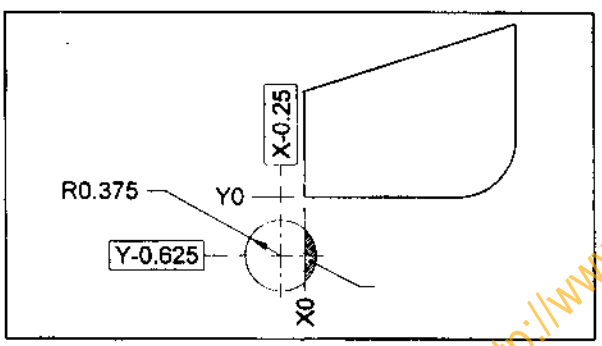
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ví dụ dưới đây minh họa kết quả chiều dài hành trình nhỏ hơn bán kính dao tạo theo trục X. Nếu giá trị D01 nhỏ hơn .375, sẽ có chuyển động đến X0. Nếu lượng D01 bằng .375, hiệu giữa chiều dài lập trình và chiều dài thực bằng không, sẽ không có chuyển động theo trục X. Trong trường đó, sự bù bán kính xảy ra không có chuyển động và sự chuyển động đến vị trí đích Y1.125 sẽ tiếp tục.

```
N3 G90 G00 G54 X-0.375 Y-0.625 S920 M03
...
N6 G41 X0 D01 F15.0                (START OFFSET)
N7 Y1.125                          (P2)
```

Bạn hãy cố gắng tránh các tình huống tương tự, dù về logic là đúng, do chúng hoàn toàn không có tính linh hoạt và có thể gây ra các khó khăn trong tương lai.

Hình 29.24 minh họa vị trí khởi động, nơi dao cắt hơi lấn vào một phía của vị trí đích. Điều này hoàn toàn không được phép và hệ điều khiển sẽ cảnh báo bằng một trong các thông báo "*Cutter radius interference*", "*CRC interference*", cảnh báo *Alarm #041*.



Hình 29.24. *Vị trí khởi động dao cắt nhỏ hơn bán kính dao.*

Chương trình tiếp theo tương tự các ví dụ trước nhưng vị trí khởi động trên trục X quá gần vị trí đích, nếu dao được lưu trong D01 có giá trị .3750:

```
N3 G90 G54 G00 X-0.25 Y-0.625 S920 M03
...
N6 G41 X0 D01 F15.0                (START OFFSET)
N7 Y1.125                          (P2)
```

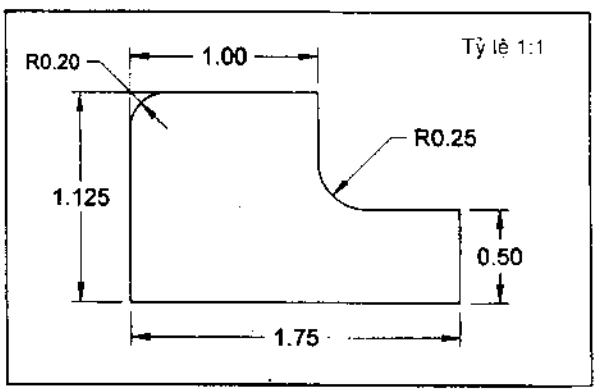
Điều gì sẽ xảy ra? Như thường lệ, bộ điều khiển tính toán hiệu số giữa chiều dài hành trình được lập trình .25 và bán kính dao .375, kiểm tra chiều của chuyển động kế tiếp theo chiều dương trục Y và xác định rằng do dao được định vị bên trái của chuyển động mong muốn, sẽ chuyển động đến .125 theo chiều âm trên trục X. Điều này dường như không thành vấn đề, do có đủ không gian trống. *Nhưng đây là vấn đề* – bộ điều khiển *không nhận biết* thực tế có không gian trống. Nhà lập trình biết, nhưng bộ điều khiển không thể biết. Các kỹ sư ứng dụng cần có thể xem xét nhiều hoạt động, mong muốn sự an toàn cao nhất. Họ quyết định để bộ điều khiển loại bỏ khả năng đó và đưa ra sự cảnh báo.Tùy theo hệ điều khiển, cảnh báo có thể là *"Overcutting will occur in cutter radius compensation C"* hoặc *"CRC interference"*... chỉ số cảnh báo này trên hệ điều khiển Fanuc là *No. 041*. Nhiều nhà lập trình, dù có kinh nghiệm, vẫn có thể gặp loại cảnh báo đó. Nếu không, họ hoặc là rất may mắn hoặc chưa từng sử dụng bù bán kính dao trong chương trình.

Thời điểm bất kỳ xảy ra cảnh báo về bù bán kính dao, bạn hãy kiểm tra kỹ cả *các block xung quanh*, không chỉ block xảy ra sự dừng xử lý.

Trong phần kế tiếp, bạn sẽ xem xét sự cản trở bán kính dao xảy ra trong khi chuyển động dao, không chỉ khi khởi đầu hoặc kết thúc chế độ bù bán kính dao.

Sự cản trở bù bán kính

Ví dụ cuối cùng được minh họa là một trong nhiều khả năng, có thể xảy ra cảnh báo bù bán kính dao. Nguyên nhân khác của cảnh báo này là khi bán kính dao đi vào vùng nhỏ hơn bán kính dao được lưu theo giá trị bù D. Để minh họa, bạn hãy đánh giá chương trình O2903 (Hình 29.25).



Hình 29.25. *Bản vẽ đơn giản dùng cho chương trình O2903.*

```
O2903 (BẢN VẼ Hình 29.25)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X-0.625 Y-0.625 S920 M03
N4 G43 Z1.0 H01
N5 G01 Z-0.55 F25.0 M08  (FOR 0.5 PLATE THICK)
N6 G41 X0 D01 F15.0                (START OFFSET)
N7 Y0.925
N8 G02 X0.2 Y1:125 R0.2
N9 G01 X1.0
N10 Y0.75
N11 G03 X1.25 Y0.5 R0.25
N12 G01 X1.75
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N13 Y0
N14 X-0.625
N15 G00 G40 Y-0.625          (CANCEL OFFSET)
N16 Z1.0 M09
N17 G28 X-0.625 Y-0.625 Z1.0
N18 M30
%
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chương trình này hoàn toàn đơn giản; đây là chương trình đúng và tuân theo mọi nguyên tắc đã nêu ở phần trên. Cốt lõi để thành công là lựa chọn đường kính dao và giá trị nhập của địa chỉ D trong bộ điều khiển. Bạn hãy quan sát điều sẽ xảy ra cùng loại dao cắt được dùng như trong các ví dụ trước, dao phay mặt đầu Ø.750. Lượng của D01 lưu trong bộ điều khiển là .3750.

Bộ điều khiển sẽ xử lý thông tin từ chương trình kết hợp với các giá trị bù để xác định chuyển động dao. Sau đó, bộ điều khiển thực thi các block khi dịch chuyển dao theo biên dạng chi tiết. Nhưng, tại block N7 xuất hiện cảnh báo *No. 041 – vấn đề cản trở bán kính dao.*

Điều gì đã xảy ra? Hoàn toàn không có điều gì sai trong chương trình. Hầu hết mọi người vận hành CNC đều kiểm tra lại chương trình. Sau khi nghiên cứu cẩn thận, nếu họ nhận thấy chương trình là đúng, nguyên nhân của vấn đề phải ở đâu đó bên ngoài chương trình. Bạn không nên đổ lỗi cho máy tính và đừng lãng phí thời gian khi bạn biết chương trình là chuẩn. Hãy kiểm tra giá trị nhập bù bán kính dao trong D01. Giá trị .375 được lưu ở đó. Điều này cũng đúng với dao trong trục chính. Kế tiếp, bạn hãy kiểm tra bản vẽ. Ở đây cũng không có gì sai. Mọi thứ dường như đều đúng nhưng vẫn có cảnh báo bù bán kính dao trên màn hình, bạn hãy thực hiện bước logic kế tiếp.

http://www.me-cad.com/

Luôn luôn đánh giá các quan hệ giữa:

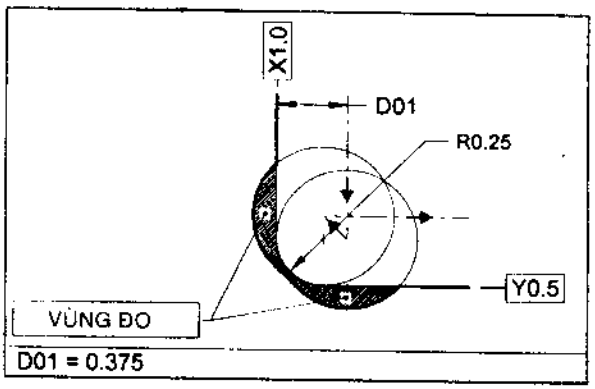
- Các kích thước trên bản vẽ ...và... Giá trị nhập chương trình
- Giá trị nhập chương trình ...*và*... Lượng bù
- Lượng bù ... và... Kích thước trên bản vẽ

Để đánh giá đúng nguyên nhân có lẽ cần đôi chút kinh nghiệm. Trong chương trình O2903, vấn đề là quan hệ giữa lượng bù đã lưu và kích thước bản vẽ.

Bạn hãy nghiên cứu kỹ bản vẽ – ở đây có bán kính góc *trong* .250 trong khi giá trị bù xác lập theo bán kính dao .375. Bán kính dao lớn không thể khớp với bán kính chi tiết .250, do đó xuất hiện sự cảnh báo.

Do không thể thay đổi kích thước bản vẽ, cần thay đường kính dao *nhỏ hơn* .500 inch. Bán kính 0.200 trên bản vẽ không gây ra vấn đề, do đây là bán kính ngoài.

Bộ điều khiển Fanuc *không* cho phép *can thiệp* vào sự bù bán kính dao *Kiểu C*. Tính năng này được thiết lập bên trong do đó không có cơ hội xem xét điều thực sự xảy ra. Hình 29.26 minh họa sự sai lệch dẫn đến cảnh báo nêu trên. Đây là lỗi sai thực tế của các dạng bù bán kính dao cũ *Kiểu A* và *Kiểu B*.

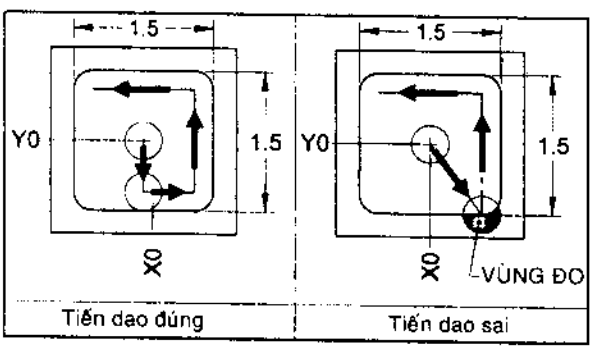


Hình 29.26. *Ảnh hưởng của sự cắt quá mức trong chế độ bù bán kính dao. Bù bán kính Kiểu C (kiểu dự đoán trước) không cho phép xảy ra sự cắt quá mức.*

Khởi động đơn trục và đa trục

Có một vấn đề khác có thể phát sinh trong khi khởi động bù bán kính dao cắt, đặc biệt nếu lập trình chuyển động khởi đầu theo hai trục, thay vì một trục. Bạn hãy xem xét khả năng này khi gia công mặt ngoài mà không xảy ra vấn đề. Phần này sẽ xem xét sự gia công mặt trong.

Bạn hãy đánh giá hai phương pháp tiếp cận trên Hình 29.27, sử dụng sự khởi đầu bù bán kính dao hướng đến biên dạng trong, ví dụ vách hốc lớn hoặc biên dạng trong khác.



Hình 29.27. *Vấn đề có thể xảy ra trong chế độ bù bán kính dao khi khởi động với hai trục đồng thời (gia công biên dạng trong).*

➲ Sự tiếp cận đúng – chuyển động một trục

Sự tiếp cận lập trình đúng được nêu trên hình trái (Hình 29.27) gồm các blocks sau, chỉ liệt kê các block khởi động:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N1 G20 (TIẾP CẬN ĐÚNG VỚI MỘT TRỤC)
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S1200 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.25 F6.0    (FOR 0.25 POCKET DEPTH)
N6 G41 Y-0.75 D01 F10.0          (START OFFSET)
N7 X0.75
N8 Y0.75
...
```

Trong chương trình này bạn không cần quan tâm đến bán kính trong, giá trị lưu trong D01 không xét đến bán kính trong, do đó chỉ biểu thị bán kính dao cắt.

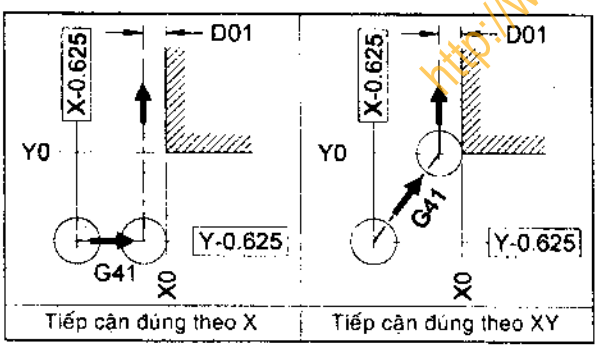
➲ Sự tiếp cận sai – chuyển động nhiều trục:

Sự tiếp cận chuyển động sai được nêu trên hình phải (Hình 29.27) gồm các block sau:

```
N1 G20 (TIẾP CẬN SAI VỚI HAI TRỤC)
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S1200 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.25 F6.0    (FOR 0.25 POCKET DEPTH)
N6 G41 X0.75 Y-0.75 D01 F10.0 (START OFFSET)
N7 Y0.75
...
```

Không có phương pháp để hệ điều khiển tìm mặt đáy của hốc ở Y-0.75. Sự khởi động chế độ bù hoàn toàn như gia công mặt ngoài nhưng rất dễ xảy ra vấn đề.

Bạn hãy so sánh hai sự khởi động khả dĩ đó với bản vẽ trên Hình 29.2. Nếu bù bán kính khởi đầu với chuyển động một trục, kết quả nêu trên hình trái (Hình 29.28). Nếu bù bán kính khởi đầu với chuyển động hai trục, kết quả được nêu trên hình phải.



Hình 29.28. *Khởi động bù bán kính dao khi gia công mặt ngoài: Bên trái – tiếp cận một trục. Bên phải – tiếp cận hai trục*

Dưới đây là vài block đúng của từng phương pháp:

➲ Tiếp cận đúng – chuyển động một trục:

```
N1 G20 (TIẾP CẬN ĐÚNG VỚI MỘT TRỤC)
N2 G17 G40 G80
N3 G90 G54 G00 X-0.625 Y-0.625 S920 M03
...
N6 G41 X0 D01 F15.0              (START OFFSET)
N7 Y1.125                                (P2)
...
```

➲ Tiếp cận đúng – chuyển động nhiều trục:

```
N1 G20 (TIẾP CẬN ĐÚNG VỚI HAI TRỤC)
N2 G17 G40 G80
N3 G90 G54 G00 X-0.625 Y-0.625 S920 M03
...
N6 G41 X0 Y0 D01 F15.0           (START OFFSET)
N7 Y1.125                                (P2)
...
```

Chú ý, trong các trường hợp bù bán kính dao đối với biên dạng ngoài, *cả hai* chương trình nêu trên đều đúng, do không có *sự cản trở* với phần bất kỳ của chi tiết gia công. Tuy nhiên, có *cùng* sự cản trở như khi phay biên dạng trong – sự khác biệt duy nhất là kiểu "cản trở" đó không gây ra hậu quả – chỉ xảy ra trong không gian trống.

TÓM TẮT CÁC NGUYÊN TẮC CHUNG

Các nguyên tắc chỉ quan trọng khi bạn hiểu rõ vấn đề. Cho đến khi đó, sự tóm tắt khái quát sẽ giúp bạn ghi nhớ nội dung dễ dàng hơn. Lập trình bù bán kính dao cũng không ngoại lệ. Các mục dưới đây có ký hiệu [M] dùng cho nguyên công phay, [T] nguyên công tiện, và [M-T] cả phay và tiện :

- ❑ [M-T] Không bao giờ khởi động hoặc xóa bù bán kính trong chế độ gia công cung (với G02 hoặc G03 có hiệu lực). Giữa block khởi động và block xóa, các lệnh gia công cung là được phép và hoàn toàn bình thường, nếu có yêu cầu.
- ❑ [M-T] Bảo đảm bán kính dao luôn luôn nhỏ hơn bán kính trong nhỏ nhất của biên dạng chi tiết.
- ❑ [M-T] Trong chế độ xóa G40, hãy đưa dao đến không gian trống. Hãy luôn luôn xem xét bán kính dao và các khoảng hở hợp lý.
- ❑ [M-T] Áp dụng bù bán kính dao với lệnh G41 hoặc G42, cùng với chuyển động nhanh hoặc tuyến tính đến phần tử biên dạng thứ nhất (G00 hoặc G01 có hiệu lực).
- ❑ [M] Đạt đến chiều sâu phay trục Z trong chế độ G40 (chế độ xóa bù bán kính dao).
- ❑ [M-T] Nếu chọn sự tiếp cận một trục từ vị trí khởi động
- ❑ [M] Đừng quên chỉ số bù D... trong chương trình, đây là lỗi nhỏ nhưng sẽ có giá rất đắT.
- ❑ [M-T] Biết chính xác vị trí lệnh dụng cụ cắt khi áp dụng bù bán kính dọc theo hai trục.
- ❑ [M-T] Trong chế độ bù (G41 hoặc G42 có hiệu lực), hãy chú ý các block không chứa chuyển động trục. Tránh các block không–chuyển động nếu có thể (không có X, Y, Z).
- ❑ [M-T] Xóa bù bán kính dao với lệnh G40, chỉ với chuyển động nhanh hoặc tuyến tính (G00/G01), nên theo chuyển động một trục.
- ❑ [M] Rút dao ra khỏi lỗ (chỉ theo trục Z) sau khi xóa bù bán kính.
- ❑ [M] Bảo đảm bù bán kính dao tương ứng mặt phẳng gia công đã chọn (Chương 30).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- [M-T] Các lệnh trả về zero máy G28 hoặc G30 sẽ không xóa bù bán kính (nhưng xóa bù chiều dài dao).
- [M-T] Có thể nhập lệnh G40 thông qua MDI để xóa bù bán kính dao (thường là biện pháp tạm thời hoặc khẩn cấp).

Nhiều tài liệu hay có tại : http://www.me-cad.com/

VÍ DỤ THỰC TIỄN – PHAY

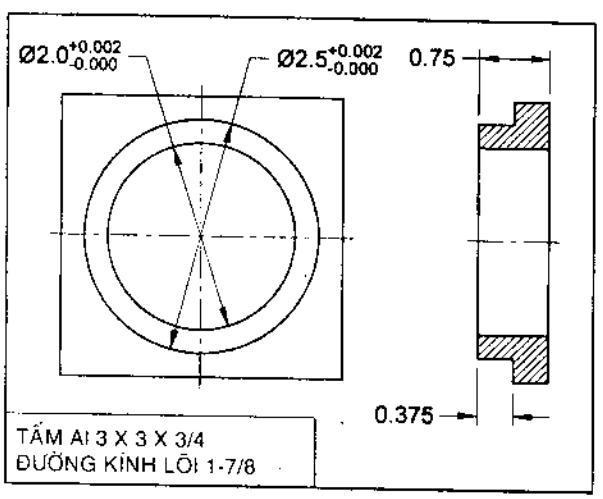
Ví dụ dưới đây giới thiệu ứng dụng thực tiễn về bù bán kính dao cho cả nhà lập trình và người vận hành CNC, bao quát hầu như mọi tình huống có thể xảy ra trong quá trình gia công và đưa ra giải pháp để duy trì các kích thước theo yêu cầu. Điều đầu tiên cần nắm vững là sự khác biệt giữa kích thước chi tiết *được đo* và *được lập trình*.

Dung sai của chi tiết

Khi hoàn tất gia công trên máy CNC (đôi khi có thể trước đó), chi tiết phải trải qua sự kiểm tra. Điều đó có nghĩa là mọi yêu cầu bản vẽ phải đáp ứng để chi tiết đạt chất lượng. Một trong các yêu cầu đó là duy trì dung sai kích thước, được ghi trên bản vẽ, hoặc được ngầm hiểu. Dung sai ngầm hiểu thường là tiêu chuẩn do công ty thiết lập dựa trên số lượng chữ số thập phân được dùng để ghi kích thước.

Ví dụ dưới đây tập trung vào tác dụng của bù bán kính dao đối với kích cỡ chi tiết trong mặt phẳng XY. Vì lý do đó, chỉ giới thiệu ứng dụng đơn giản, với quỹ đạo dao đơn giản nhất, chưa hẳn là phương pháp gia công tối ưu (Hình 29.29).

Tiêu điểm sẽ là dung sai *chuyên biệt* trên bản vẽ +.002/-.000 đối với kích thước của hai đường kính, ∅ 2.5 đường kính ngoài, và ∅2.0 đường kính trong. Chú ý khoảng dung sai là *như nhau* đối với cả hai đường kính. Điều này là rất quan trọng.



Hình 29.29. Bản vẽ minh họa ứng dụng thực tiễn của bù bán kính dao cắt.

Kích thước chi tiết theo đo đạc

Thợ cơ khí có kinh nghiệm biết rõ kích thước đo đạc thực tế của chi tiết phụ thuộc vào nhiều yếu tố chẳng hạn độ cứng vững gá lắp, chiều sâu cắt, vật liệu gia công, chiều cắt gọt, lựa chọn dao, kích thước chính xác, ...

Khi kiểm tra chi tiết, kích thước đo chỉ có thể là một trong ba trường hợp :

- Đúng kích thước ...trong khoảng dung sai đã cho
- Quá kích thước ...phế phẩm nếu gia công mặt trong
- Thiếu kích thước ...phế phẩm nếu gia công mặt ngoài

Trường hợp thứ nhất là lý tưởng, bất kể gia công mặt ngoài hay mặt trong. Nếu kích thước đo *là đúng*, trong *khoảng* dung sai cho phép, chi tiết đạt yêu cầu. Trường hợp thứ hai (dư kích thước) và thứ ba (thiếu kích thước) sẽ được xem xét chung với nhau.

Trong cả hai trường hợp, kích thước đo là *ngoài khoảng dung sai cho phép*. Tình huống này đòi hỏi xét thêm hai vấn đề :

- Phương pháp cắt gọt ngoài ...được gọi là Mặt ngoài OD
- Phương pháp cắt gọt trong ...được gọi là Mặt trong ID

Do dụng cụ cắt tiếp cận biên dạng gia công từ các hướng khác nhau, thuật ngữ *quá kích thước* và *thiếu kích thước* luôn luôn liên quan với *kiểu cắt gọt*. Bảng dưới đây liệt kê các kết quả thường xảy ra:

Điều kiện	Gia công mặt ngoài	Gia công mặt trong
Quá kích thước	Có thể gia công lại	Phế phẩm
Đúng kích thước	Đạt yêu cầu	Đạt yêu cầu
Thiếu kích thước	Phế phẩm	Có thể gia công lại

Quan sát bảng này, rõ ràng là không cần chỉnh sửa nếu kích thước đo trong phạm vi dung sai bất kể gia công mặt trong hay mặt ngoài. Đối với dư hoặc thiếu kích thước, kết quả có thể là phế phẩm hoặc phải gia công lại.

Chi tiết được gia công *mặt ngoài* (∅ 2.500 inch OD trong ví dụ) được đo *lớn hơn* dung sai cho phép có thể gia công lại, nhưng kích thước *nhỏ hơn* dung sai cho phép sẽ trở thành phế phẩm.

Chi tiết được gia công *mặt trong* (∅ 2.000 inch ID trong ví dụ) được đo *nhỏ hơn* dung sai cho phép có thể gia công lại, nhưng kích thước *lớn hơn* dung sai cho phép sẽ trở thành phế phẩm.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

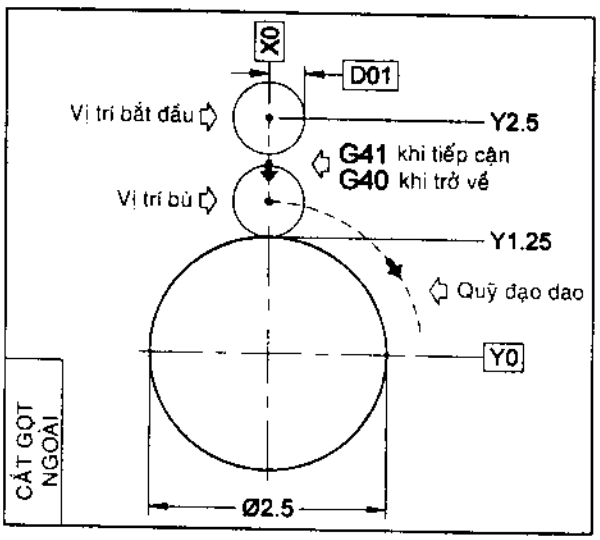
Lượng bù được lập trình

Đặc tính hấp dẫn nhất của bù bán kính dao là cho phép thay kích cỡ dao thực ngay trên máy, bằng *hàm D đăng ký bù*. Trong chương trình ví dụ, chỉ dùng một dao – dao phay mặt đầu *đường kính .750 inch*, và một đường cắt cho từng biên dạng (ngoài và trong). Tọa độ X0Y0Z0 của chương trình là ở tâm các đường tròn và mặt đỉnh của chi tiết:

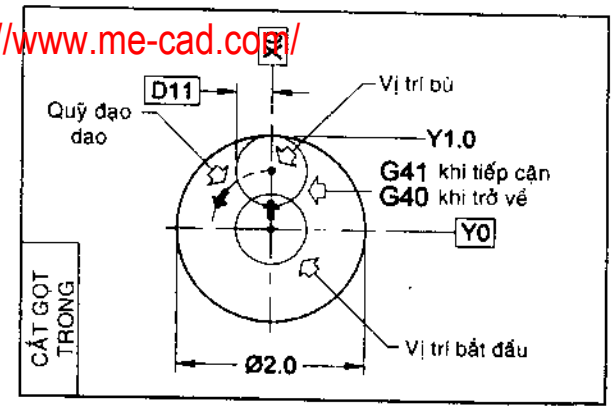
```
O2904      (T01 - 0.75 DIA END FINISHING MILL)
(**** PART 1 - 2.5 DIA CẮT GỌT NGOÀI ****)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y2.5 S600 M03   (START POS.)
N4 G43 Z0.1 H01 M08            (CLEAR+TOOL LG.)
N5 G01 Z-0.375 F20.0        (DEPTH FOR 2.5 DIA)
N6 G41 Y1.25 D01 F10.0       (APPROACH MOTION)
N7 G02 J-1.25              (EXT. CIRCLE CUTTING)
N8 G01 G40 Y2.5                (RETURN MOTION)
N9 G00 Z0.1                      (CLEAR ABOVE)
(**** PART 2 - 2.0 DIA CẮT GỌT TRONG ****)
N10 Y0                        (START POS. AT X0Y0)
N11 G01 Z-0.8 F20.0         (DEPTH FOR 2.0 DIA)
N12 G41 Y1.0 D11 F8.0        (APPROACH MOTION)
N13 G03 J-1.0              (INT. CIRCLE CUTTING)
N14 G01 G40 Y0                 (RETURN MOTION)
N15 G00 Z0.1 M09                 (CLEAR ABOVE)
N16 G28 Z0.1 M05          (Z AXIS MACHINE ZERO)
N17 M01                        (OPTIONAL STOP)
...
```

Hình 29.30 minh họa quỹ đạo ở nửa đầu của chương trình – đường kính *ngoài* 2.500 inch. Hình 29.31 minh họa quỹ đạo ở nửa cuối chương trình – đường kính *trong* 2.000 inch.

Là thói quen trong lập trình CNC và cũng được dùng trong chương trình O2904, quỹ đạo dao sử dụng các kích thước *bản vẽ* và các vị trí do nhà lập trình xác định. Đây không chỉ là phương pháp tiêu chuẩn mà còn rất thuận tiện để xây dựng chương trình. Người vận hành dễ



Hình 29.30. *Chi tiết về quỹ đạo dao ngoài (Chương trình O2904)*



Hình 29.31. *Chi tiết về quỹ đạo dao trong (Chương trình O2904).*

dàng hiểu rõ chương trình, các kích thước bản vẽ dễ theo dõi và có thể thay đổi nếu có yêu cầu. Nói một cách đơn giản, nhà lập trình *bỏ qua bán kính dao* và viết chương trình *tựa như dao cắt là một điểm*, dao cắt có đường kính bằng không.

Lượng bù D – Xác lập tổng quát

Trong thực tế gia công, rất ít khi sử dụng dao đường kính zero, trừ một vài nguyên công chạm khắc. Hầu hết các dao cắt đều có đường kính chuyên biệt và luôn luôn xét đến đường kính thực, nếu không trong chương trình, thì trên máy.

Một trong các yếu tố cơ bản cần thiết lập trước là hệ thống CNC luôn luôn tính toán sự bù chuyên biệt theo *bán kính* dao cắt, *không* theo đường kính. Điều đó có nghĩa là nhà lập trình cung cấp *bù bán kính dao* theo dạng *địa chỉ D*. Trên máy, giá trị bù lập trình D01 sẽ áp dụng cho bán kính dao được đăng ký trong chế độ bù 1, D02 cho bán kính đăng ký trong bù 2,... Trong các bộ đăng ký đó chứa các giá trị thực nào?

Do không có bán kính dao trong chương trình, bộ đăng ký bù D thường chứa giá trị *thực* của bán kính dao. Cần rất cẩn thận, một số tham số máy có thể được xác lập để chấp nhận *đường kính* dao, dù mọi tính toán bên trong đều sử dụng bán kính.

Bạn hãy đánh giá chương trình O2904; giá trị lưu của D01 là bao nhiêu? Sử dụng dao phay mặt đầu Ø .750, do đó D01 được xác lập là .375. Điều này là đúng về lý thuyết, nhưng các yếu tố, chẳng hạn áp lực dao cắt, vật liệu gia công, độ lệch dao, dung sai dao, và nhiều yếu tố khác, đều ảnh hưởng đến kích cỡ chi tiết hoàn tất. Kết luận là giá trị đăng ký D01 có thể là .375, nhưng chỉ trong các điều kiện *lý tưởng*.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Các điều kiện này không xảy ra khi gia công trong thực tế. Những yếu tố ảnh hưởng đến gia công cũng có tác động rõ rệt đối với các kích thước chi tiết. Dễ dàng nhận thấy kích cỡ đo bất kỳ không trong khoảng dung sai chỉ có thể là *dư* hoặc *thiếu* kích thước và phương pháp gia công *mặt trong* hay *mặt ngoài* sẽ *tạo ra* sự khác biệt khi điều chỉnh giá trị bù.

Bất kể phương pháp cắt gọt, có một nguyên tắc áp dụng cho sự điều chỉnh bù bán kính dao trong hệ thống điều khiển bất kỳ – nguyên tắc có hai phần bằng nhau:

Số gia DƯƠNG đối với bù bán kính dao sẽ làm dụng cụ cắt rời RA XA biên dạng gia công.

Số gia ÂM đối với bù bán kính dao sẽ làm cho dụng cụ cắt ĐẾN GẦN biên dạng gia công.

Bạn hãy chú ý thuật ngữ "*số gia*" – với ý nghĩa lượng bù bán kính hiện hành sẽ *thay đổi* hoặc *cập nhật*, nhưng *không* thay thế bằng giá trị mới. Khái niệm "*rời ra xa*" và "*đến gần*" chi tiết được hiểu là chuyển động dao do người vận hành CNC quan sát. Kích cỡ đo của chi tiết có thể được kiểm soát bằng cách *điều chỉnh* giá trị bù bán kính dao trong bộ điều khiển, được lập trình theo địa chỉ D, tuân theo nguyên tắc nêu trên. Nguyên tắc thông dụng nhất, áp dụng *như nhau* cho các điều chỉnh gia công *ngoài và trong*, như sau:

Để TĂNG kích thước CHO giá trị đo, cần dùng lượng xác lập bù D LỚN HƠN

Để GIẢM kích thước CHO giá trị đo, cần dùng lượng xác lập bù D NHỎ HƠN

Những người vận hành CNC giàu kinh nghiệm có thể thay đổi các xác lập bù ngay tại máy, nếu chương trình có lệnh bù bán kính G41 hoặc G42 và chỉ số bù địa chỉ D, với sự xóa thích hợp bằng G40.

Đánh giá những điều *thực sự* xảy ra trong khi chuyển động đối với từng phương pháp cắt gọt (*ngoài* hoặc *trong*) đưa đến một số tùy chọn. Trong cả hai trường hợp, dao cắt đều chuyển động từ vị trí *khởi đầu*, trong không gian trống, đến vị trí *đích* của biên dạng gia công. Đây là chuyển động *áp dụng* chế độ bù bán kính dao, do đó rất quan trọng. Thực vậy, đó là chuyển động *xác định kích cỡ đo hoàn tất* của chi tiết. Phần kế tiếp sẽ trình bày từng phương pháp cắt gọt.

Điều chỉnh lượng bù

Trước khi xem xét các chi tiết cụ thể, bạn hãy suy nghĩ về khả năng thay đổi lượng bù bán kính. Trong các trường hợp cần điều chỉnh kích cỡ, sự thay đổi giá trị bù theo *số gia* có lẽ là cách thích hợp nhất. Thay đổi giá trị bù theo số gia nghĩa là *cộng* hoặc *trừ* bớt giá trị bù hiện hành (Sử dụng phím +INPUT trên màn hình Fanuc) hoặc lưu giá trị điều chỉnh trong cột màn hình bù *Wear* (độ mòn dao). Về nguyên tắc, bạn *không* nên thay đổi dữ liệu chương trình.

Bù đối với cắt gọt ngoài

Bạn hãy đánh giá khoảng dung sai với đường tròn *ngoài* ∅ 2.5. Dung sai của đường kính này là +.002/-0.0, do đó mọi kích thước trong khoảng 2.500 - 2.502 đều đúng. Kích thước *nhỏ hơn* 2.5 là *thiếu* và kích thước *lớn hơn* 2.502 là *dư*.

Đối với cắt gọt *ngoài*, kích thước đo sẽ thuộc một trong ba trường hợp. Mọi ví dụ đều dựa trên kích thước trung bình 2.501 và giá trị D01 là .375, bán kính của dao phay ∅ .750.

➲ **Kích thước đo ngoài – Ví dụ 1**

2.5010 *với* D01 = 0.3750

Đây là kết quả lý tưởng, không cần điều chỉnh lượng bù. Lưỡi dao cắt tiếp xúc với bề mặt gia công một cách chính xác. Tất cả đều hoạt động tốt và xác lập bù là chính xác, chỉ cần giám sát theo tiêu chuẩn. Điều này thường xảy ra với dao cắt mới, gá lắp chắc chắn, và dung sai thông thường.

http://www.me-cad.com/

➲ **Kích thước đo ngoài – Ví dụ 2**

2.5060 *với* D01 = 0.3750

Đường kính đo bị *dư kích thước* .005. Lưỡi cắt *chưa đạt đến* biên dạng, cần đưa lưỡi cắt *đến gần hơn*. Lượng bù bán kính cần giảm theo một nửa giá trị kích thước dư, trên đường kính hoặc chiều rộng, do lượng bù được nhập theo bán kính. Giá trị D01 được điều chỉnh theo số gia .0025, đến D01 = 0.3725

➲ **Kích thước đo ngoài – Ví dụ 3**

2.4930 *với* D01 = 0.3750

Đường kính đo bị *thiếu kích thước*. Lưỡi cắt lấn *sâu* vào bề mặt gia công được lập trình, cần đưa lưỡi cắt *ra xa*. Lượng bù bán kính cần tăng theo một nửa giá trị kích thước thiếu, do lượng bù được nhập theo bán kính. Giá trị D01 được điều chỉnh theo số gia .004, đến D01 = 0.3790.

Bù đối với cắt gọt trong

Khoảng dung sai của đường kính *trong* 2.0 inch là +.002/-0.000, do đó mọi kích thước chi tiết trong khoảng 2.000 – 2.002 đều đúng. Kích thước *nhỏ hơn* 2.000 là *thiếu* và kích thước *lớn hơn* 2.002 là *dư*.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Kích thước đo khi cắt gọt trong sẽ chọn một trong ba trường hợp. Mọi ví dụ đều dựa trên kích thước trung bình 2.001 và D11 có giá trị .375, bán kính của dao ∅ .750.

➲ **Kích thước đo trong – Ví dụ 4**

2.0010 *với* D11 = 0.3750

Đây là kết quả lý tưởng – không cần điều chỉnh giá trị bù. Lưỡi cắt tiếp xúc bề mặt gia công một cách chính xác. Mọi thứ đều hoạt động tốt và xác lập bù là chính xác, chỉ cần giám sát bình thường.

➲ **Kích thước đo trong – Ví dụ 5**

2.0060 *với* D11 = 0.3750

Đường kính đo bị *dư kích thước* .005. Lưỡi cắt lấn *sâu* vào bề mặt gia công, cần đưa lưỡi cắt *ra xa*. Giá trị bù bán kính cần *tăng* theo một nửa kích thước dư. Giá trị D11 phải tăng theo số gia .0025, đến D11 = 0.3775.

➲ **Kích thước đo trong – Ví dụ 6**

1.9930 *với* D11 = 0.3750

Đường kính đo bị *thiếu kích thước* .008. Lưỡi cắt *chưa đạt đến* biên dạng gia công, cần đưa lưỡi cắt *đến gần hơn*. Giá trị bù bán kính cần *giảm* theo một nửa kích thước thiếu. Giá trị D11 giảm theo số gia .004, đến D11 = 0.3710.

Một hay nhiều giá trị bù?

Chương trình O2904 sử dụng D01 cho đường kính ngoài và D11 cho đường kính trong. Chỉ dùng một dao và mục đích là kích thước dung sai trung bình 2.501 cho đường kính ngoài và 2.001 cho đường kính trong. Cần hai chế độ bù hay chỉ cần một chế độ bù trong chương trình?

Bạn cần nhớ những ví dụ nêu trên chỉ đánh giá các khả năng độc lập với nhau. Chương trình O2904 giới thiệu sự liên kết chung giữa hai đường kính, một dao phay mặt đầu ∅.750 dùng để cắt gọt *cả hai* đường kính.

Giả sử chỉ dùng một địa chỉ bù, chẳng hạn D01, với giá trị lưu là .375. Khi đo, đường kính ngoài là 2.501. Bạn khi tiếp tục cắt đường kính trong ∅ 2.000, kết quả đo không phải là 2.001 như mong đợi mà là 1.999, thiếu kích thước đến .002. Lý do là *cả hai đường kính* đều có dung sai +.002/-0.000. Các kết quả là khác nhau – đối với đường kính ngoài, +0.002 có nghĩa là dư kích thước, có thể cắt gọt lại, đối với đường kính trong, +.002 nghĩa là dư kích thước, phải loại bỏ. Do chỉ một giá trị bù không thể điều chỉnh để đáp ứng dung sai trung bình trên *cả hai* đường kính, cần sử dụng *hai chế độ bù*. Nếu D01 = .3750 bảo đảm đường kính ngoài là hoàn hảo, D11 phải có xác lập nhỏ hơn, giá trị sẽ là .3730.

Người vận hành CNC phải luôn luôn chú ý các giá trị bù trong chương trình và hiểu các lượng bù đã lưu, đặc biệt khi sử dụng nhiều giá trị cho một dao cắt.

Trong gá lắp hoặc bảng dụng cụ cắt, nhà lập trình phải liệt kê các giá trị bù được dùng trong chương trình và đề nghị các giá trị khởi đầu cho từng chế độ bù.

Tránh phế phẩm

Khi xét các lượng bù ban đầu, có thể áp dụng một số kỹ thuật sáng tạo. Mục đích là sử dụng các xác lập bù sao cho chi tiết gia công không trở thành phế phẩm, kể cả với dụng cụ cắt chưa được kiểm nghiệm. Người vận hành giàu kinh nghiệm có thể tránh các phế phẩm do các giá trị bù không đúng, ít nhất là ở mức độ nào đó. Chìa khóa là tạo ra vài xác lập bù *tạm thời*. Mục đích là buộc cắt gọt *dư kích thước ngoài* hoặc *thiếu kích thước trong*, đo đạc, điều chỉnh sau đó cắt gọt lại đúng kích thước.

Dù gia công theo quỹ đạo dao trong hay ngoài, kể cả sự gá lắp tốt nhất cũng chưa bảo đảm các kích thước chi tiết trong khoảng dung sai cho phép. Khi gia công biên dạng *ngoài*, có thể *cố ý* cắt đường kính *lớn hơn* yêu cầu, theo cách thức có kiểm soát. Trường hợp đó, nguy cơ có thể là đường kính *quá nhỏ*.

Khi gia công biên dạng *trong*, có thể *cố ý* cắt đường kính *nhỏ hơn* yêu cầu, khi đó sẽ có nguy cơ đường kính *quá lớn*. Những trường hợp này đều có ưu điểm và nhược điểm...

Trong các ví dụ nêu trên, giải pháp là dịch chuyển dao *ra xa* bề mặt ngoài cần gia công theo *số gia bù dương*. Số gia này phải lớn hơn sai số tính toán của bán kính dao, và thích hợp để cắt gọt lại.

Trong cả hai trường hợp, sau khi thực hiện cắt thử nghiệm, bạn hãy đo đường kính và điều chỉnh giá trị bù theo một nửa hiệu số giữa đường kính đo được và đường kính mong muốn. Nếu chỉ cắt một phía, không cần chia đôi hiệu số đó.

Dữ liệu chương trình – Danh định hay trung bình?

Nhiều vị trí tọa độ trong chương trình phản ảnh các kích thước thực được lấy từ bản vẽ. Câu hỏi là – điều gì xảy ra nếu kích thước bản vẽ chuyên biệt khoảng dung sai? Các nhà lập

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

trình CNC có hai quan điểm. Một quan điểm ủng hộ sử dụng giá trị trung bình trong khoảng dung sai, nhóm ý kiến khác ủng hộ sử dụng kích thước danh định, không xét các giá trị dung sai. Cả hai quan điểm đều có các ưu điểm riêng, không loại trừ lẫn nhau. Ở đây sẽ sử dụng kích thước *danh định* và xử lý dung sai bằng cách áp dụng hợp lý các giá trị bù – trên máy công cụ. Có hai lý do chính. Một là, chương trình sử dụng các kích thước danh định sẽ dễ đọc hơn. Hai là, trong trường hợp có các thay đổi trên bản vẽ, chúng sẽ ảnh hưởng dung sai nhiều hơn so với kích thước danh định.

BÙ BÁN KÍNH MŨI DAO

Mọi nguyên tắc đã nêu ở phần trên đều có thể áp dụng cho bù bán kính đối với dao tiện gia công biên dạng. Ở đây có vài khác biệt, chủ yếu do hình dạng dụng cụ cắt.

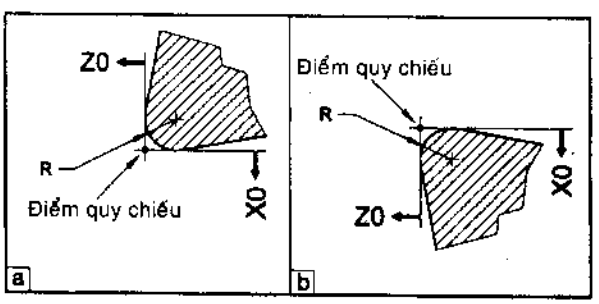
Trong nguyên công phay, dao cắt luôn luôn có hình tròn. Chu vi dao phay là lưỡi cắt và bán kính là giá trị bù. Dao tiện có thiết kế khác. Phổ biến nhất là mảnh Carbide nhiều cạnh. Mảnh chắp có thể có một hoặc nhiều lưỡi cắt. Để tăng tuổi bền của mảnh chắp, lưỡi cắt có bán kính góc tương đối nhỏ. Các bán kính thông dụng của dao tiện và dao doa là:

1/64 = .0156 (Anh) *hoặc* 0.40 mm (hệ mét)
2/32 = .0313 (Anh) *hoặc* 0.80 mm (hệ mét)
3/64 = .0469 (Anh) *hoặc* 1.20 mm (hệ mét)

Do lưỡi cắt thường được gọi là *mũi dao*, thuật ngữ *bù bán kính mũi dao* trở nên thông dụng.

Đỉnh mũi dao

Mũi dao thường là góc của dụng cụ cắt, nơi hai lưỡi cắt hòa nhập thành bán kính mũi. Hình 29.32 minh họa các góc nhỏ phổ biến của dao tiện và dao doa.

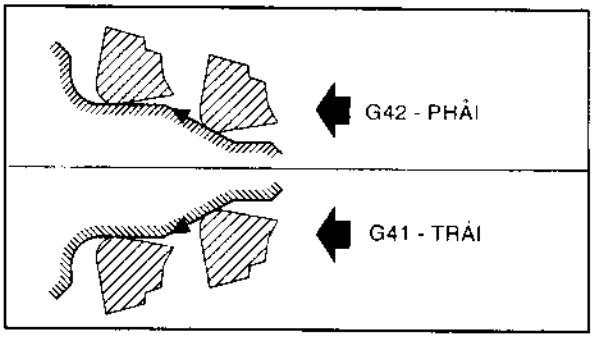


Hình 29.32. *Điểm quy chiếu dao – (a) tiện, (b) doa.*

Điểm quy chiếu mũi dao tiện thường được gọi là *điểm lệnh*, *điểm tưởng tượng*, hoặc *điểm ảo*. Đây là điểm dịch chuyển dọc theo biên dạng, do liên hệ trực tiếp với X0Z0 của chi tiết.

Lệnh bù bán kính

Các lệnh chuẩn bị trong nguyên công phay cũng được dùng để gia công biên dạng trên máy tiện (Hình 29.33):



Hình 29.33. *Ứng dụng tiện của bù bán kính dao*

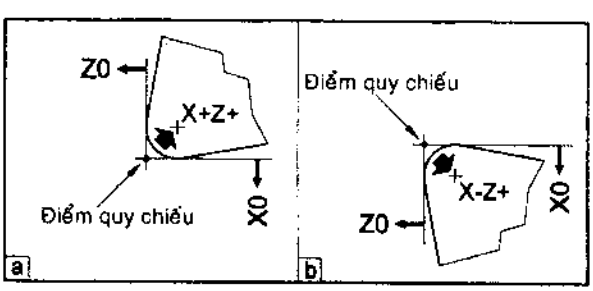
G41	Bù bán kính mũi dao theo BÊN TRÁI của chiều biên dạng
G42	Bù bán kính mũi dao theo BÊN PHẢI của chiều biên dạng
G40	XÓA bù bán kính mũi dao.

Đối với máy tiện, các mã G không sử dụng địa chỉ D-giá trị bù được lưu trong bù *Geometry/Wear* (Hình học/Mòn dao). *Dao tiện có các lưỡi cắt khác*, phần còn lại tương tự dao phay.

Định hướng đỉnh dao

Tâm của đường tròn ký hiệu dao phay mặt đầu phải là khoảng cách cân bằng (cách đều) biên dạng bằng giá trị bán kính. Trong nguyên công phay, lưỡi cắt là một phần của bán kính dao, trong gia công tiện, lưỡi cắt là độc lập. Dao tiện có bán kính nhưng các lưỡi cắt riêng rẽ. Tâm bán kính mũi là khoảng cách cân bằng (cách đều) biên dạng, và các lưỡi cắt thay đổi chiều của chúng, ngay cả đối với một mảnh chắp. Cần có thêm các định nghĩa dưới dạng vector hướng đến tâm bán kính đó. Vector này là *định hướng đỉnh dao*, được đánh số tùy ý. Bộ điều khiển sử dụng chỉ số này để thiết lập tâm bán kính mũi và hướng (Hình 29.34).

Định hướng đỉnh được nhập vào trong quá

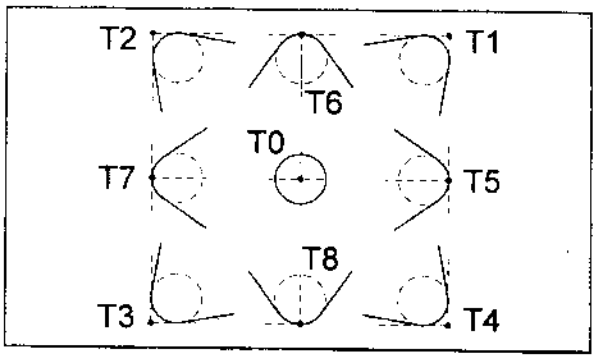


Hình 29.34. *Quan hệ giữa điểm quy chiếu dao và tâm bán kính mũi*

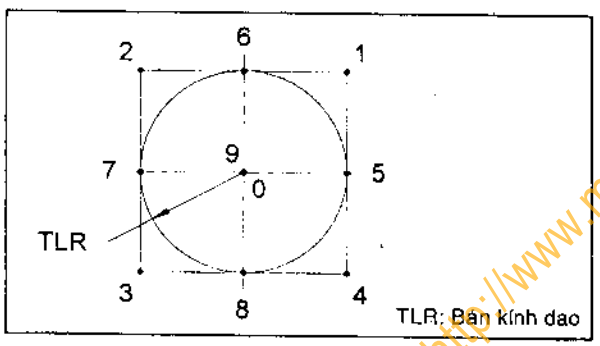
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

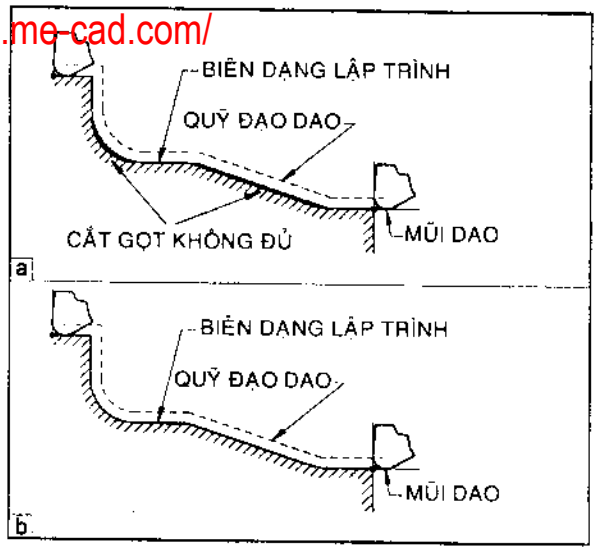
trình xác lập tuân theo các nguyên tắc chung chế. Bộ điều khiển Fanuc yêu cầu chỉ số cố định cho từng đỉnh dao. Số này được nhập vào màn hình bù trên bộ điều khiển, dưới tiêu đề T. Giá trị R của bán kính dao cũng phải được nhập. Nếu đỉnh dao là 0 hoặc 9, bộ điều khiển sẽ bù cho điểm tâm. Hình 29.35 và 30.36 trình bày sự đánh số đỉnh dao tiêu chuẩn trên máy tiện CNC với X+ hướng lên và Z+ theo bên phải điểm gốc tọa độ.



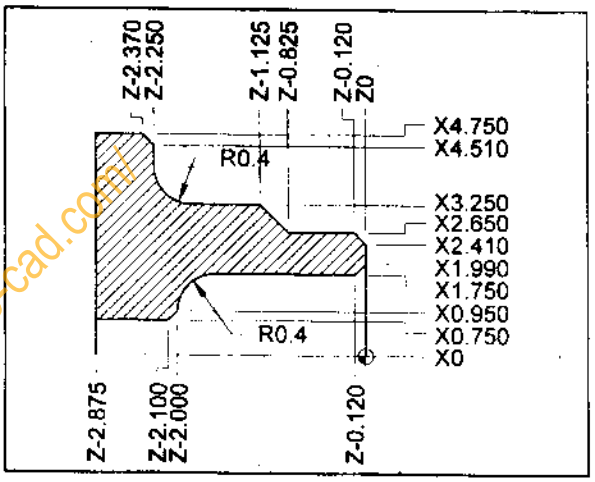
Hình 29.35. Các số đỉnh dao đối với bù bán kính mũi.



Hình 30.36. Sơ đồ minh họa đánh số đỉnh dao (bộ điều khiển Fanuc)



Hình 29.37. Tác dụng của bù bán kính mũi dao – (a) không sử dụng bù (b) có sử dụng bù



Hình 29.38. Bản vẽ đơn giản cho chương trình O2905

http://www.me-cad.com/

Tác dụng của bù bán kính mũi dao

Một số nhà lập trình không thích sử dụng bù bán kính mũi dao. *Điều đó là sai!* Về lý thuyết, không cần chế độ bù, nếu chỉ lập trình một trục. Tuy nhiên, các chuyển động đơn trục là một phần của biên dạng cũng gộp các bán kính, vạt góc, và độ côn. Trong trường hợp đó, bù bán kính mũi dao *là* cần thiết, nếu không bù, tất cả các bán kính, vạt góc, và độ côn sẽ không chính xác. Hình 29.37 minh họa các phần của chi tiết dễ bị *thiếu* hoặc *dư* kích thước, nếu *không* sử dụng bù bán kính mũi dao trong khi gia công.

Chương trình mẫu

Chương trình O2905 minh họa ứng dụng cơ bản của bù bán kính mũi dao trên biên dạng ngoài và trong, theo bản vẽ trên Hình 29.38. Ở đây chỉ nêu sự gia công tinh, cũng cần gia công thô, nhưng sẽ sử dụng chu kỳ lặp lại đặc biệt G71 (Chương 34).

```
O2905
...
N31 T0300                (GIA CÔNG TINH NGOÀI)
N32 G96 S450 M03
N33 G00 G42 X2.21 Z0.1 T0303 M08
N34 G01 X2.65 Z-0.12 F0.007
N35 Z-0.825 F0.01
N36 X3.25 Z-1.125
N37 Z-1.85
N38 G02 X4.05 Z-2.25 R0.4
N39 G01 X4.51
N40 X4.8 Z-2.395
N41 U0.2
N42 G00 G40 X8.0 Z5.0 T0300
N43 M01

N44 T0400                (GIA CÔNG TINH TRONG)
N45 G96 S400 M03
N46 G00 G41 X2.19 Z0.1 T0404 M08
N47 G01 X1.75 Z-0.12 F0.006
N48 Z-1.6 F0.009
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

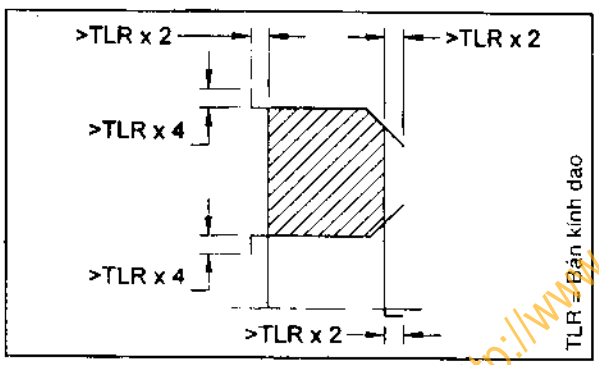
```
N49 G03 X0.95 Z-2.0 R0.4
N50 G01 X0.95 Z2.1
N51 Z-2.925
N52 U-0.2
N53 G00 G40 X8.0 Z2.0 T0400
N54 M01
...
```

Chú ý, các vị trí bắt đầu và kết thúc biên dạng là trong khoảng không gian trống, cách xa chi tiết. Bạn cần bảo đảm có đủ khoảng hở. Nếu khoảng hở không đủ, sẽ xuất hiện *cảnh báo cần trở bù bán kính dao* (cảnh báo #41).

Khoảng hở cần thiết tối thiểu

Về nguyên tắc, từng khoảng hở trong chương trình phải đủ lớn để bảo đảm gấp đôi bán kính mũi dao.

Hình 29.39 minh họa các khoảng hở tối thiểu xác lập ở đầu và cuối đường cắt. Bạn cần bảo đảm khoảng hở không nhỏ hơn hai lần bán kính mũi dao. Ký hiệu *>TLR × 2* và *× 4* nghĩa là khoảng hở lớn hơn hai hoặc bốn lần bán kính mũi. Bán kính gấp đôi theo một phía sẽ là gấp bốn theo đường kính (hai phía).

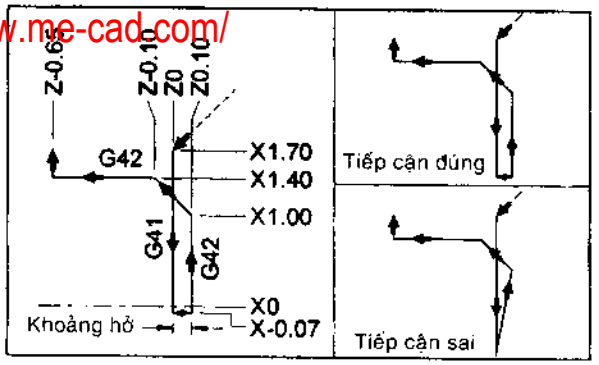


Hình 29.39. *Khoảng hở tối thiểu đối với bù bán kính mũi dao*

Trong bù bán kính mũi dao, lập trình khoảng hở tối thiểu .100 inch (2.5 mm) theo một phía, sẽ cung cấp đủ khoảng hở cho cả ba bán kính mũi dao tiêu chuẩn – 1/64, 1/32, và 3/64 (0.40, 0.80, và 1.20 mm)

Thay đổi chiều chuyển động

Trên máy tiện CNC, sự thay đổi chiều cắt gọt xảy ra thường xuyên hơn so với trung tâm gia công. Ví dụ dưới đây minh họa sự cắt gọt



Hình 29.40. *Thay đổi bù bán kính mũi dao đối với cùng một dụng cụ cắt*

mặt (vạt mặt) trên bề mặt khối với lệnh G41 có hiệu lực, chuyển sang tiện với G42 có hiệu lực (Hình 29.40).

```
...
N21 T0100                          (TIẾP CẬN ĐÚNG)
N22 G96 S400 M03
N23 G00 G41 X1.7 Z0 T0101 M08              (START)
N24 G01 X-0.07 F0.007                  (FACE OFF)
N25 G00 Z0.1                       (ONE AXIS ONLY)
N26 G42 X1.0                  (THEN COMPENSATION)
N27 G01 X1.4 Z-0.1 F0.012          (CONTOURING)
N28 Z-0.65
N29 X...
```

Vạt mặt đầu là chuyển động cắt một trục và chế độ bù được sử dụng không thay đổi. Đối với các chi tiết nguyên khối, vạt mặt phải kết thúc *dưới* đường tâm, X-0.07 trong block N24 tại đường kính hơi lớn hơn gấp đôi bán kính mũi dao. Nếu sự cắt gọt kết thúc tại X0, dao để lại một phần nhỏ chưa hoàn tất ở đường tâm và mặt gia công sẽ không phẳng. Ngoài ra, bạn hãy so sánh các chuyển động dao *đúng* và *không đúng* trên hình bên phải (Hình 29.4). Nếu chương trình nêu trên được sửa lại như sau:

```
...
N21 70100 (TIẾP CẬN SAI)
N22 G96 S400 M03
N23 G00 G41 X1.7 Z0 T0101 M08              (START)
N24 G01 X-0.07 F0.007                  (FACE OFF)
N25 G00 G42 X1.0 Z0.1              (*** WRONG ***)
N26 G01 X1.4 Z-0.1 F0.012          (CONTOURING)
N27 Z-0.65
N28 X..
```

... mặt gia công sẽ không hoàn tất ! Bạn hãy suy nghĩ về điều đó.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

30 LỰA CHỌN MẶT PHẲNG

Trong tất cả các nguyên công cắt gọt, *gia công biên dạng* là ứng dụng CNC phổ biến nhất, có lẽ ngang với gia công lỗ. Trong khi gia công biên dạng, chuyển động dao được lập trình theo ít nhất ba cách thức:

- Chuyển động dao theo một trục.
- Chuyển động dao đồng thời theo hai trục.
- Chuyển động dao đồng thời theo ba trục.

Có thể áp dụng thêm các chuyển động trục khác (ví dụ *trục thứ tư, thứ năm* ...) nhưng trên trung tâm CNC, bạn luôn luôn làm việc với ít nhất *ba* trục, dù *không phải lúc nào cũng* đồng thời. Điều đó phản ánh thực tế không gian ba chiều.

Chương này chỉ áp dụng cho các hệ thống phay CNC, do hệ thống tiện thường chỉ sử dụng hai trục, do đó không yêu cầu sử dụng mặt phẳng. Dao trên máy tiện CNC không thuộc chủ đề này.

Điểm tuyệt đối bất kỳ trong chương trình được xác lập bằng ba tọa độ, theo các trục X, Y và Z. Chuyển động nhanh G00 hoặc chuyển động tuyến tính G01 có thể sử dụng đồng thời *số trục bất kỳ*, nếu chuyển động dao là an toàn trong vùng làm việc, không yêu cầu các xem xét đặc biệt, cũng không cần sự lập trình đặc biệt.

Điều đó không phải là trường hợp của ba kiểu lập trình dưới đây, các xem xét lập trình thay đổi rõ rệt:

- Chuyển động tròn sử dụng lệnh G02 hoặc G03
- Bù bán kính dao sử dụng lệnh G41 hoặc G42
- Chu kỳ cố định sử dụng các lệnh G81 đến G89, G73, G74, và G76

Trong cả ba trường hợp nêu trên – *và chỉ ba trường hợp này* – nhà lập trình phải xét đến sự xác lập đặc biệt trong hệ điều khiển – *lựa chọn mặt phẳng gia công*.

GIA CÔNG TRONG CÁC MẶT PHẲNG

Quỹ đạo dao là sự kết hợp giữa đường thẳng và cung. Chuyển động dao theo một hoặc hai trục luôn luôn xảy ra trong mặt phẳng được ký hiệu bằng hai trục. Kiểu chuyển động này là *hai chiều*. Chuyển động dao bất kỳ theo ba trục đồng thời là chuyển động *ba chiều*.

Mặt phẳng toán học

Trong gia công CNC, những mặt phẳng có thể được xác định và sử dụng là các mặt phẳng tổ hợp của *hai* trong ba trục XYZ. Do đó, *chuyển động cắt gọt tròn, bù bán kính dao*, và *chu kỳ cố định* chỉ có thể xảy ra với một trong ba mặt phẳng khả dụng:

Mặt phẳng XY	Mặt phẳng ZX	Mặt phẳng YZ

Thứ tự gán trục trong định nghĩa mặt phẳng là rất quan trọng. Ví dụ, mặt phẳng XY và YX về *hình học* là đồng nhất. Tuy nhiên, đối với mục đích xác định chiều chuyển động dao (*thuận* và *ngược chiều kim đồng hồ*, *trái* và *phải*), cần thiết lập tiêu chuẩn rõ ràng.

Tiêu chuẩn quốc tế dựa trên quy tắc toán học chữ *thứ nhất* trong ký hiệu mặt phẳng luôn luôn là trục *hoành* (ngang) và chữ *thứ hai* là trục *tung* (dọc) khi quan sát mặt phẳng cả hai trục này luôn luôn *chuẩn trực* (ngang và dọc) và *vuông góc* (90^0) với nhau. Trong CAD/CAM, tiêu chuẩn này xác định sự khác nhau giữa trên và dưới, trước và sau, ...

Cách đơn giản để nhớ ký hiệu *toán học* các trục của cả ba trục mặt phẳng là viết thứ tự chữ cái của ba trục *hai lần* và tách từng cặp với khoảng trắng:

XYZXYZ	...trở thành...	XY ZX YZ

Về *hình học*, các mặt phẳng này được định nghĩa như sau:

Mặt phẳng	Trục hoành	Trục tung
XY	X	Y
ZX	Z	X
YZ	Y	Z

Bạn hãy lưu ý sự nhấn mạnh từ "*hình học*". Sự nhấn mạnh này là cố ý và có lý do. Bạn sẽ nhận thấy có sự khác biệt lớn giữa mặt phẳng hình học và mặt phẳng gia công, được xác định theo chiều quan sát trên máy công cụ:

Mặt phẳng máy công cụ

Trung tâm gia công CNC thông dụng có ba trục. Hai trục bất kỳ tạo thành một mặt phẳng. Mặt phẳng máy có thể được xác định bằng cách quan sát máy từ vị trí vận hành tiêu chuẩn. Đối với trung tâm gia công đứng, có ba

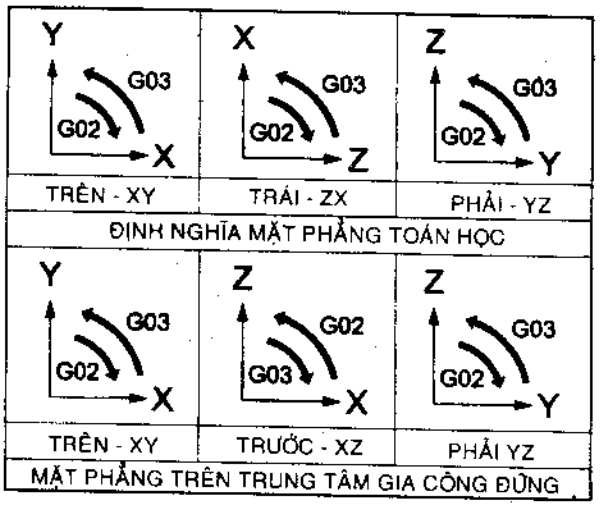
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

điểm chiếu tiêu chuẩn, được quan sát một cách vuông góc (hình thẳng):

- Hình chiếu trên ...mặt phẳng XY
- Hình chiếu trước ...mặt phẳng XZ
- Hình chiếu phải ...mặt phẳng YZ

Hình 30.1 minh họa sự khác biệt giữa hai định nghĩa, do các điểm chiếu không tương thích.



Hình 30.1. *So sánh giữa các mặt phẳng toán học (trên) và mặt phẳng trên trung tâm gia công CNC (dưới).*

Rõ ràng là mặt phẳng XY là hình chiếu từ trên xuống là như nhau trong cả hai định nghĩa, và do đó mặt phẳng YZ là hình chiếu từ bên phải. Mặt phẳng hình học ZX *khác với* hình chiếu từ phía trước đến trên máy công cụ, đó là mặt phẳng XZ.

Mặt phẳng hình học ZX có Z là trục hoành, được *đảo lại* trên mặt phẳng máy của trung tâm gia công. Trên máy này, mặt phẳng đó trở thành XZ với X là *trục hoành*, đây là sự phân biệt rất quan trọng.

Trong lập trình, sự lựa chọn mặt phẳng cực kỳ quan trọng, nhưng thường bị bỏ qua hoặc hiểu sau. Lý do chính là đa số các chuyển động dao (đặc biệt là gia công biên dạng) được lập trình và gia công trên mặt phẳng XY tiêu chuẩn. Trên mọi trung tâm gia công CNC tiêu chuẩn, trục chính luôn luôn vuông góc với mặt phẳng XY.

Lệnh lập trình xác định mặt phẳng

Sự lựa chọn mặt phẳng đối với bộ điều khiển Fanuc tuân thủ định nghĩa *hình học* của mặt phẳng, *không* phải là mặt phẳng máy CNC thực. Trong chương trình gia công, có thể chọn từng mặt phẳng *hình học* bằng lệnh chuẩn bị đặc biệt – mã G đặc thù:

G17	Chọn mặt phẳng XY
G18	Chọn mặt phẳng ZX
G19	Chọn mặt phẳng YZ

Đối với *mọi* chuyển động nhanh (lập trình với G00) *và* mọi chuyển động tuyến tính (lập trình với G01) lệnh lựa chọn mặt phẳng là không cần thiết, thậm chí dư thừa, nhưng điều đó không đúng với các chế độ chuyển động khác, trong đó sự lựa chọn mặt phẳng trong chương trình là *cực kỳ* quan trọng và phải được xem xét rất cẩn thận.

Đối với các ứng dụng gia công sử dụng chế độ nội suy tròn, với lệnh G02 hoặc G03, chế độ bù bán kính dao với lệnh G41 hoặc G42, và chế độ chu kỳ cố định (các lệnh G81 đến G89, G73, G74, G76), sự lựa chọn mặt phẳng có tính quyết định.

Trạng thái điều khiển mặc định

Nếu chương trình không chọn mặt phẳng, bộ điều khiển sẽ tự động chọn mặt phẳng XY (G17) khi phay và ZX (G18) khi tiện. Nếu lập trình mã G chọn mặt phẳng, mã đó sẽ được xếp ở đầu chương trình. Do ba lệnh mặt phẳng chỉ có hiệu lực đối với *chuyển động tròn, bù bán kính dao*, và các *chu kỳ cố định*, lệnh chọn mặt phẳng G17, G18, hoặc G19 có thể được lập trình *trước* khi xảy ra một trong ba chuyển động đó.

Luôn luôn lập trình lệnh chọn mặt phẳng thích hợp. Không nên dựa vào các xác lập điều khiển.

Sự thay đổi chọn mặt phẳng bất kỳ đều có thể lập trình theo yêu cầu, trước khi thực sự thay đổi quỹ đạo dao. Mặt phẳng có thể được thay đổi tùy theo nhu cầu trong chương trình, nhưng vào thời điểm bất kỳ chỉ có một mặt phẳng hoạt động. Sự lựa chọn một mặt phẳng sẽ xóa mặt phẳng khác, do đó các lệnh G17/G18/G19 sẽ xóa lẫn nhau. Dù chỉ có tính thông tin, nhưng khả năng phối hợp cả ba lệnh này trong một chương trình hầu như rất ít khi xảy ra. Trong ba chuyển động nêu trên, chỉ có chuyển động tròn là bị tác động từ sự lựa chọn mặt phẳng, nhưng trước hết bạn hãy xem xét sự lập trình chuyển động nhanh và chuyển động tuyến tính, ít nhất là để so sánh.

CHUYỂN ĐỘNG THẲNG TRONG MẶT PHẲNG

Cả chuyển động nhanh G00 và chuyển động tuyến tính G01 đều được coi là chuyển động thẳng khi so sánh với chuyển động tròn. Có thể lập trình chuyển động thẳng theo một trục, hai trục, hoặc ba trục đồng thời. Các ví dụ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

dưới đây chỉ minh họa các block không liên hệ với nhau.

➲ Ví dụ – Định vị nhanh – G00

```
G00 X5.0 Y3.0
                    mặt phẳng XY – chuyển động nhanh 2D
G00 X7.5 Z-1.5  mặt phẳng XZ – chuyển động nhanh 2D
G00 Y10.0 Z-0.25
                    mặt phẳng YZ – chuyển động nhanh 2D
G00 X2.0 Y4.0 Z-0.75
                    mặt phẳng XYZ  chuyển động nhanh 3D
```

➲ Ví dụ – Nội suy tuyến tính – G01

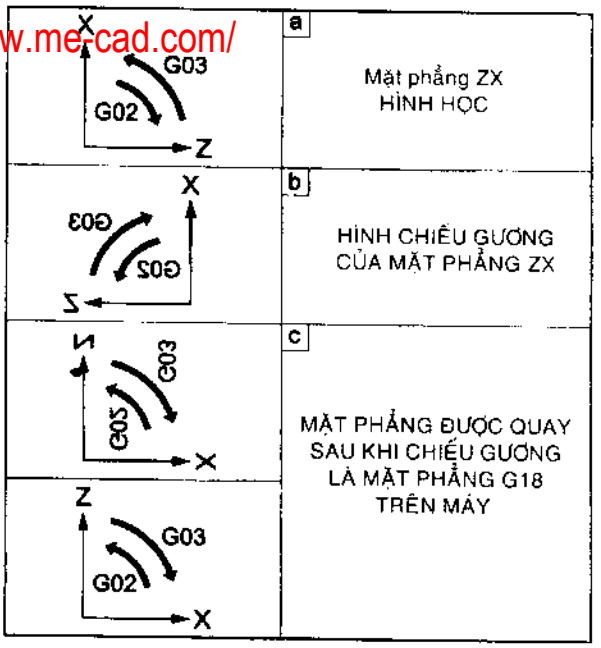
```
G01 X-1.5 Y4.46 F15.0
                    mặt phẳng XY – chuyển động nhanh 2D
G01 X8.875 Z-0.84 F10.0
                    mặt phẳng XZ – chuyển động nhanh 2D
G01 Y12.34 Z0.1 F12.5
                    mặt phẳng YZ – chuyển động nhanh 2D
G01 X6.0 Y13.0 Z-1.24 F12.0
                    mặt phẳng XYZ – chuyển động nhanh 3D
```

Các ví dụ nêu trên biểu thị chuyển động dao theo các trục lập trình. Lệnh chọn mặt phẳng nói chung không cần sử dụng đối với chuyển động thẳng bất kỳ (theo một trục) trừ khi bù bán kính dao hoặc chu kỳ cố định có hiệu lực. Bộ điều khiển sẽ diễn dịch chính xác mọi chuyển động dao, bất kể mặt phẳng nào đang có hiệu lực. Các nguyên tắc áp dụng cho chuyển động tuyến tính khác với các chuyển động tròn.

NỘI SUY TRÒN TRONG MẶT PHẲNG

Để hoàn tất chuyển động tròn một cách chính xác, hệ điều khiển cần nhận đủ thông tin từ chương trình gia công. Khác với chuyển động nhanh khi G00 có hiệu lực hoặc nội suy tuyến tính khi G01 có hiệu lực, sự nội suy chuyển động tròn đòi hỏi *chiều* chuyển động được lập trình. G02 là lệnh áp dụng chiều CW và G03 là chiều CCW. Theo các quy tắc toán học *thuận chiều kim đồng hồ* (CW) luôn luôn được chiếu từ trục tung hướng đến trục hoành trong mặt phẳng bất kỳ được chọn. *Ngược chiều kim đồng hồ* (CCW) luôn luôn chiếu từ trục hoành đến trục tung.

Khi so sánh giữa ký hiệu các trục theo toán học và định hướng thực của các trục máy (dựa trên trung tâm gia công đứng), mặt phẳng XY (G17) và YZ (G19) là tương ứng với nhau. Hai mặt phẳng này thường không gây ra các vấn đề cho nhà lập trình CNC. Mặt phẳng ZX (G18) có thể gây ra vấn đề nghiêm trọng nếu không được hiểu đúng. Về toán học, trục hoành trong mặt phẳng G18 là trục Z và trục X là trục tung. Trên trung tâm gia công đứng, thứ tự định hướng các trục máy là *ngược lại*. Điều quan trọng cần hiểu là chiều CW và CCW *chỉ dường như* bị đảo ngược, nhưng trong thực tế, chúng là như nhau. Nếu định hướng các trục

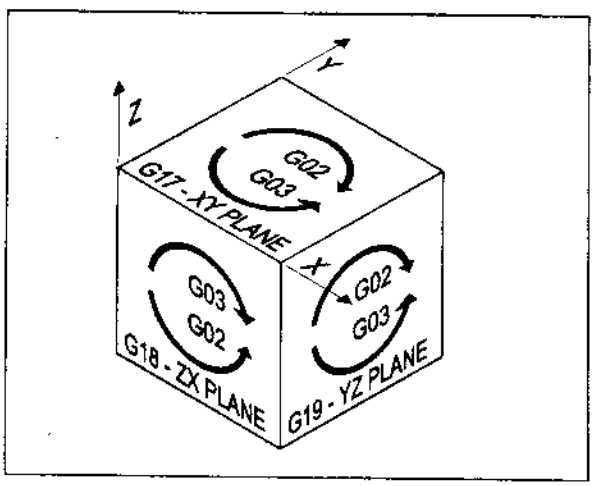


Hình 30.2. *Các bước xếp thẳng hàng mặt phẳng hình học ZX với mặt phẳng XZ, sử dụng sự lựa chọn mặt phẳng G18.*

toán học thẳng hàng với các trục máy, chúng *sẽ* tương hợp với nhau. Hình 30.2 minh họa các bước xếp thẳng hàng các mặt phẳng hình học với mặt phẳng máy.

Bạn hãy chú ý chiều mã G đối với các cung *không* thay đổi trong mặt phẳng hình học (a), mặt phẳng hình học chiếu gương (b), kể cả mặt phẳng chiếu gương quay 90^0 âm (c), dù nếu chính mặt phẳng đó thay đổi. Điều xảy ra ở đây *không* phải là tạo ra mặt phẳng mới bất kỳ. Hình chiếu vẫn biểu thị vật thể ba chiều, chiếu từ điểm chiếu khác.

Trên trung tâm gia công ngang, tình huống cũng tương tự. Mặt phẳng XY (G17) và mặt



Hình 30.3. *Chiều quỹ đạo tròn của dao trong cả ba mặt phẳng máy. Chú ý sự không đồng nhất biểu kiến của mặt phẳng G18.*

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

phẳng ZX (G18) tương hợp giữa ký hiệu toán học và định hướng thực của các trục. Mặt phẳng G19 (YZ) *dường như* đảo ngược, có thể gây ra một số vấn đề nếu không hiểu rõ cấu trúc logic của các mặt phẳng đó.

Sự lựa chọn hợp lý mặt phẳng gia công sẽ cho phép lập trình các gia công biên dạng khác nhau sử dụng nội suy tròn và xoắn, bù bán kính dao, và chu kỳ cố định. Các ứng dụng phổ biến của kiểu gia công này là hòa nhập các bán kính, giao cắt các bán kính, hốc tròn, doa ngược biên dạng, mặt trụ, khối cầu và nón đơn giản, và nhiều hình dạng tương tự khác.

Để hiểu ứng dụng CNC của các lệnh G02 và G03 trong mặt phẳng, bạn hãy xem Hình 30.3.

G17-G18-G19 các lệnh chế độ

Các lệnh chuẩn bị chọn mặt phẳng G17, G18, và G9 đều là lệnh *chế độ* – lập trình một trong các lệnh đó sẽ chỉ kích hoạt mặt phẳng được chọn. Sự lựa chọn mặt phẳng trong chương trình sẽ có hiệu lực cho đến khi bị xóa bằng sự chọn mặt phẳng khác. Ba mã G liên quan đến mặt phẳng hoàn toàn thuộc nhóm mã G số 02.

Các ví dụ định dạng dưới đây minh họa ứng dụng lập trình nội suy chuyển động tròn:

```
G17 G02 X14.4 Y6.8 R1.4
G18 G03 X11.575 Z-1.22 R1.0
G19 G03 Y4.5 Z0 R0.85
```

Một số hệ điều khiển cũ không chấp nhận sự gán bán kính trực tiếp chuyên biệt theo địa chỉ R. Thay vào đó, các vector cung I, J, và K được sử dụng. Đối với lập trình chuyển động tròn trong mặt phẳng đã chọn, cần chọn đúng cặp vector cung:

```
G17 G02 (G03) X.. Y.. I.. J..
G18 G02 (G03) X.. Z.. I.. K..
G19 G02 (G03) X.. Z.. J.. K..
```

Từ đó, bạn cần nhớ:

- Các trục XY – mặt phẳng G17 – các vector I và J
- Các trục XZ – mặt phẳng G18 – các vector I và K
- Các trục YY – mặt phẳng G19 – các vector J và K

Không có dữ liệu trục trong block

Định dạng lập trình nêu trên chứa *đầy đủ dữ liệu* đối với điểm cuối của chuyển động tròn.Trong thực tiễn, nhà lập trình giàu kinh nghiệm không cần lặp lại các giá trị chế độ từ block này sang block khác. Lý do chính của phương pháp này là tiết kiệm thời gian lập trình, rút ngắn chương trình và tăng không gian nhớ khả dụng trong hệ thống điều khiển.

Phần của chương trình ví dụ dưới đây minh họa ứng dụng điển hình trong chương trình với các giá trị trục chế độ *không lặp lại* trong các block kế tiếp.

```
N.. G20                          Đơn vị Anh
...
N40 G17                          Chọn mặt phẳng XY
N41 G00 X20.0 Y7.5 Z-3.0         Điểm khởi động dao
N42 G01 X13.0 F10.1              Không cần chọn lại mặt
                                 phẳng
N43 G18 G02 X7.0 R3.0            Giả thiết không có trục Z
N44 G17 G01 X0                   Không cần chọn lại mặt
                                 phẳng
```

Block N43 đưa ra biên dạng là cung 180^0 trong mặt phẳng ZX. Do lệnh G18 trong N43, bộ điều khiển sẽ diễn dịch chính xác trục "bị thiếu" là Z, có giá trị bằng giá trị trục Z cuối cùng được lập trình (Z-3.0). Ngoài ra, bạn hãy xét lệnh G17 trong block N44. Bạn nên chuyển trạng thái điều khiển trở về sự chọn mặt phẳng gốc ngay khi mặt phẳng này thay đổi, dù không thực sự cần thiết trong ví dụ nêu trên.

Sự bỏ qua lệnh G18 trong block N43 sẽ gây ra lỗi chương trình nghiêm trọng. Nếu G18 bị sót, lệnh đã chọn trước đó (G17) sẽ vẫn có hiệu lực và nội suy tròn sẽ xảy ra trong mặt phẳng XY thay vì mặt phẳng ZX như mong muốn.

Trong trường hợp đó, trục được giả thiết là "bị thiếu" trong mặt phẳng G17 sẽ là Y và giá trị lập trình là Y7.5. Hệ điều khiển sẽ xử lý block kiểu đó tựa như được chuyên biệt trong block hoàn chỉnh:

```
N43 G17 G02 X7.0 Y7.5 R3.0
```

Một tình huống thú vị sẽ xuất hiện nếu thiếu lệnh chọn mặt phẳng G18 trong block N43, block nội suy tròn chứa *hai tọa độ trục* đối với điểm cuối của chuyển động tròn :

```
N43 G02 X7.0 Z-3.0 R3.0          G17 vẫn còn hiệu lực
```

Tuy G17 là mặt phẳng hoạt động, nhưng cung này vẫn được gia công một cách chính xác trong mặt phẳng G18, dù G18 không được lập trình. Điều này là do tính năng đặc biệt của bộ điều khiển, có tên là *lệnh hướng dẫn hoàn chỉnh* hoặc *ưu thế dữ liệu hoàn chỉnh*, được cung cấp trong block N43 nêu trên. Sự gộp hai trục cho điểm cuối của chuyển động tròn có độ ưu tiên cao hơn so với lệnh chọn mặt phẳng. Block hoàn chỉnh là block chứa mọi địa chỉ cần thiết mà không lấy các giá trị chế độ.

Hai lệnh được lập trình trong một block vượt qua lệnh chọn mặt phẳng đang có hiệu lực.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Bù bán kính dao trong mặt phẳng

Chọn mặt phẳng đối với chuyển động nhanh hoặc tuyến tính là không cần thiết, nếu lệnh bù bán kính dao G41 hoặc G42 không có hiệu lực. Về lý thuyết, điều đó có nghĩa là bất kể sự chọn mặt phẳng, mọi chuyển động G00 và G01 đều đúng. Điều này là chính xác, nhưng ít có tính thực tiễn, do hầu hết các chương trình CNC *đều sử dụng* chuyển động biên dạng và chế độ bù bán kính dao. Ví dụ, bạn hãy đánh giá các block dưới đây:

```
N1 G21
...
N120 G90 G00 X50.0 Y100.0 Z20.0
N121 G01 X90.0 Y140.0 Z0 F180.0
```

Khi chuyển động nhanh được lập trình trong block N120, dao sẽ được định vị ở vị trí tuyệt đối X50.0 Y100.0 Z20.0. Vị trí tuyệt đối của chuyển động dao sẽ là X90.0 Y140.0 Z0, sau khi hoàn tất block N121.

Bổ sung lệnh bù bán kính dao G41 hoặc G42 vào block chuyển động nhanh, sự chọn mặt phẳng sẽ trở nên rất quan trọng. Bù bán kính sẽ chỉ có hiệu lực đối với hai trục được chọn bằng lệnh chọn mặt phẳng. Hiện nay *chưa* có sự bù bán kính 3-trục! Trong ví dụ kế tiếp, bạn hãy so sánh các vị trí tuyệt đối của dao trong từng mặt phẳng khi hoàn tất chuyển động nhanh và bù bán kính dao được kích hoạt trong chương trình. Vị trí tuyệt đối của dao khi hoàn tất chuyển động cắt phụ thuộc vào chuyển động tiếp sau block N121.

Giá trị bù bán kính D25 = 100.000 mm, được lưu trong bộ đăng ký bù của hệ điều khiển, được dùng trong ví dụ dưới đây:

➲ Ví dụ

```
N120 G90 G00 G41 X50.0 Y100.0 Z20.0 D25
N121 G01 X90.0 Y140.0 Z0 F180.0
```

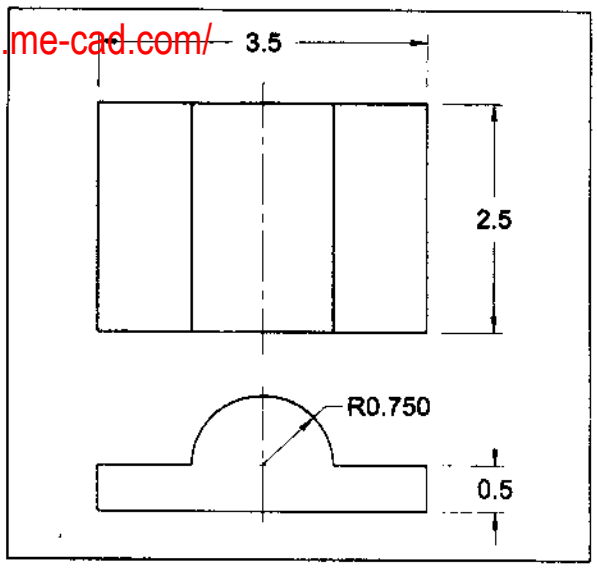
Vị trí bù dao khi hoàn tất block N120 phụ thuộc vào mặt phẳng G17, G18, hoặc G19 đang có hiệu lực :

- Nếu G17 được lập trình với ba trục:
  G17 X.. Y.. Z.. chuyển động XY sẽ được bù
- Nếu G18 được lập trình với ba trục:
  G18 X.. Y.. Z.. chuyển động ZX sẽ được bù
- Nếu G19 được lập trình với ba trục:
  G19 X.. Y.. Z.. chuyển động YZ sẽ được bù

Ví dụ lập trình thực tiễn dưới đây sẽ minh họa cả nội suy tròn và bù bán kính khi áp dụng trong các mặt phẳng khác nhau.

VÍ DỤ THỰC TIỄN

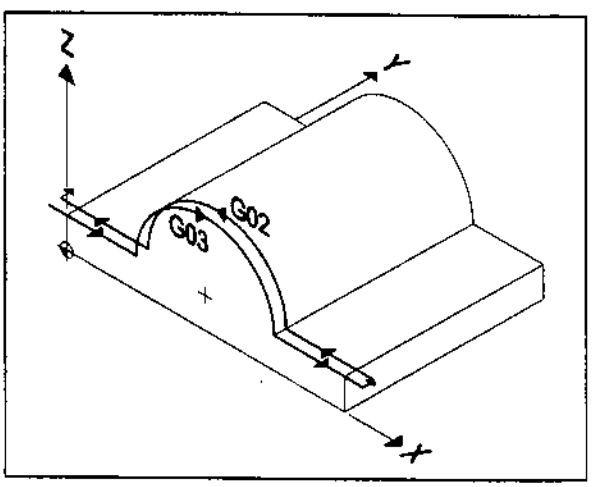
Ví dụ trên Hình 30.4 là chi tiết đơn giản,



Hình 30.4. *Bản vẽ chi tiết cho chương trình O3001*

yêu cầu cắt gọt cung R0.75 trong mặt phẳng XZ. Nói chung, dao phay mặt đầu mũi tròn sẽ được dùng để cắt gọt biên dạng cung. Trong ví dụ này, chỉ lập trình hai đường cắt gọt. Một đường là chuyển động trái sang phải - từ mặt phẳng trái, qua phần trong, và qua mặt phẳng bên phải. Đường thứ hai là phải sang trái - từ mặt phẳng phải, đi qua phần trụ, và qua mặt phẳng bên trái. Giữa các đường cắt còn lập trình sự vượt qua bậc cho dao cắt. Chương trình kiểu này đối với toàn bộ chi tiết có thể được thực hiện trong chế độ số gia và sẽ rất thuận lợi nếu dùng các chương trình con.

Hình 30.5 minh họa chuyển động dao đối với hai hành trình cắt gọt trong chương trình. Để diễn dịch dữ liệu chương trình một cách chính xác, bạn hãy lưu ý zero chương trình ở *góc dưới bên trái* của chi tiết. Cả hai khoảng hở đối với chi tiết là .100 và sự cách bậc là .050:



Hình 30.5. *Quỹ đạo dao trong chương trình O3001*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
O3001
N1 G20
N2 G18                          (CHỌN MẶT PHẲNG ZX)
N3 G90 G54 G00 X-0.1 Y0 S600 M03
N4 G43 Z2.0 H01 M08
N5 G01 G42 Z0.5 D01 F8.0
N6 X1.0
N7 G03 X2.5 I0.75   (= G03 X2.5 Z0.5 I0.75 K0)
N8 G01 X3.6
N9 G91 G41 Y0.05
N10 G90 X2.5
N11 G02 X1.0 I-0.75(= G02 X1.0 Z0.5 I-0.75 K0)
N12 G01 X-0.1
N13 G91 G42 Y0.05
N14 G90 ...
```

Khi làm việc với kiểu chương trình này lần đầu, bạn nên kiểm tra quỹ đạo dao mà không có chi tiết, điều đó sẽ giúp bạn phát hiện các lỗi, nếu có.

Chuyển động cắt gọt ba trục được lập trình bằng tay chỉ đối với các chi tiết không đòi hỏi tính toán nhiều. Đối với các chi tiết yêu cầu tính toán chuyển động phức tạp, bạn nên sử dụng phần mềm lập trình trên máy tính.

CHU KỲ CỐ ĐỊNH TRONG MẶT PHẲNG

Vấn đề lập trình cuối cùng liên quan đến lựa chọn mặt phẳng là ứng dụng mặt phẳng trong chu kỳ cố định. Đối với các chu kỳ trong mặt phẳng G17 (Các vị trí lỗ XY), G17 chỉ quan trọng nếu có sự chuyển từ mặt phẳng này sang mặt phẳng khác trong một chương trình. Với các đồ gá gia công, chẳng hạn *đầu góc vuông*, mũi khoan hoặc dụng cụ cắt được định vị *vuông góc* với đường tâm trục chính, trong mặt phẳng G18 hoặc G19.

Dù các đầu góc vuông không thực sự phổ biến, nhưng vẫn được ưa chuộng trong gia công. Khi lập trình loại đồ gá này, bạn phải xét chiều dụng cụ cắt ăn vào chi tiết (chiều *sâu*). Trong các ứng dụng của chu kỳ cố định, mặt phẳng G17 sử dụng các trục XY cho vị trí tâm lỗ và trục Z cho chiều sâu. Nếu đầu góc vuông được gá lắp để dùng trục Y làm chiều sâu, bạn hãy sử dụng mặt phẳng G18 và các trục XZ sẽ là vị trí tâm lỗ. Nếu gá lắp đầu góc vuông để dùng trục X làm chiều sâu, bạn phải sử dụng mặt phẳng G19 và các trục YZ sẽ là vị trí tâm lỗ. Trong mọi trường hợp, mức R luôn luôn áp dụng cho trục chuyển động theo chiều sâu.

Sự chênh lệch giữa đỉnh dao và đường tâm trục chính là phần treo thực. Bạn cần biết rõ phần treo này và đưa vào mọi chuyển động của trục tương ứng không chỉ để bảo đảm độ sâu chính xác mà còn tăng tính an toàn.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

31 PHAY CHU VI

Dù dao cắt carbide được sử dụng ngày càng rộng rãi để cắt gọt kim loại, dao phay mặt đầu HSS (thép dụng cụ cắt gọt tốc độ cao - thép gió) vẫn được ưa chuộng cho nhiều nguyên công phay. Loại dao cắt này có nhiều ưu điểm, tương đối rẻ tiền, dễ kiếm, và có nhiều công dụng.

Dao thép gió có giá tương đối thấp và khả năng cắt gọt chi tiết với độ chính xác cao, do đó là lựa chọn chủ yếu cho nhiều nguyên công phay. Dao phay mặt đầu có lẽ là dụng cụ cắt đa năng nhất được dùng trên máy CNC.

Dao phay mặt đầu với các mảnh chắp bằng carbide được sử dụng thường xuyên cho nhiều nguyên công. Chủ yếu là khi gia công đòi hỏi lượng ăn dao lớn và các vật liệu cứng. Dao phay thép gió vẫn thông dụng trong gia công hàng ngày.

Chương này xem xét một vấn đề công nghệ khi chương trình CNC yêu cầu dao phay mặt đầu kiểu loại bất kỳ hoặc dụng cụ cắt tương tự được dùng để cắt gọt chu vi hoặc biên dạng. Đây là nguyên công chủ yếu sử dụng cạnh dao để cắt gọt.

DAO PHAY MẶT ĐẦU

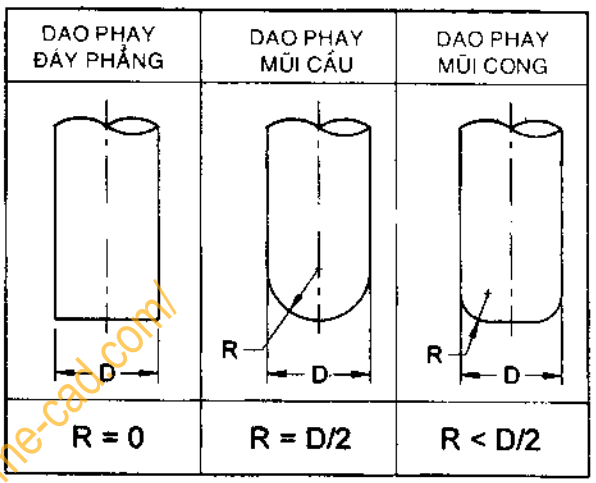
Dao phay mặt đầu là dụng cụ cắt thông dụng nhất khi phay chu vi. Hiện có sự lựa chọn rộng rãi các dao phay mặt đầu, khả dụng cho hầu như mọi nguyên công cắt gọt. Dao phay mặt đầu truyền thống có kích cỡ theo hệ Anh và hệ mét, nhiều cỡ đường kính, kiểu dáng, số lượng rãnh cắt (thoát phoi), các thiết kế góc đặc biệt, cán, và thành phần vật liệu.

Dưới đây là một số nguyên công cắt gọt phổ biến có thể thực hiện với dao phay mặt đầu:

- Phay chu vi và biên dạng.
- Phay rãnh và rãnh then.
- Cắt rãnh, rãnh xoi bề mặt, hốc lõm.
- Hốc hở hoặc kín.
- Phay bề mặt các diện tích nhỏ.
- Phay bề mặt thành (vách) mỏng.
- Doa ngược.
- Vạt mặt, vạt nghiêng miệng lỗ.
- Loại bỏ bavia, ...

Có thể tạo hình dao phay mặt đầu bằng cách mài theo hình dạng mong muốn. Các hình dạng thông dụng bao gồm dao phay mặt đầu đáy *phẳng*, dao phay mặt đầu với bán kính toàn phần (thường được gọi là dao phay mặt đầu *mũi cầu*), và dao phay mặt đầu với bán kính góc (dao phay mặt đầu *mũi cong*).

Hình 31.1 minh họa ba kiểu dao phay mặt đầu thông dụng, cùng với quan hệ giữa bán kính và đường kính dao cắt.



Hình 31.1. Cấu hình cơ bản của ba kiểu dao phay mặt đầu.

http://www.me-cad.com/

Góc thoát phoi

Điều quan trọng là chọn góc thoát phoi thích hợp khi cắt gọt các vật liệu khác nhau. Góc thoát còn được gọi là góc *hở*. Đối với dao phay thép gió, góc thoát phoi tăng khi độ cứng của vật liệu giảm. Ví dụ, góc thoát chính khi cắt gọt thép là $3^0 - 5^0$, nhôm là $10^0 - 12^0$.

Kích cỡ cho dao phay mặt đầu

Ba tiêu chuẩn rất quan trọng liên quan đến với kích thước dao phay mặt đầu cần được xem xét khi gia công CNC:

- Đường kính dao phay mặt đầu.
- Chiều dài dao phay mặt đầu.
- Chiều dài rãnh thoát phoi.

Đối với gia công CNC, đường kính dao phay mặt đầu phải rất chính xác, thường được cung cấp với giá trị danh định. Kích cỡ phi tiêu chuẩn, chẳng hạn các dao cắt được mài lại, phải được xử lý riêng khi gia công CNC. Dù có

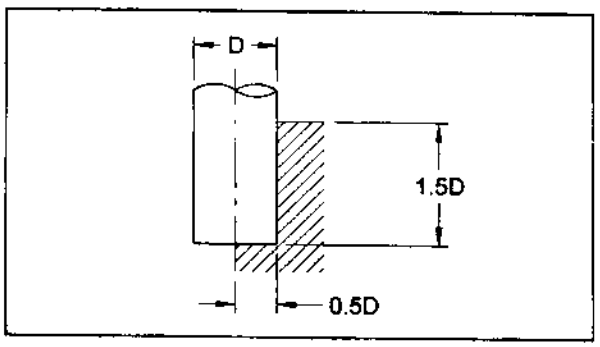
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

sự bù bán kính dao, cũng không nên sử dụng dao phay mặt đầu được mài lại để gia công chính xác, chỉ nên dùng để gia công thô hoặc trong trường hợp khẩn cấp. Điều này không có nghĩa là không thể sử dụng dao cắt mài lại trên máy phay thường.

Chiều dài dao phay mặt đầu chìa ra từ ổ dao là rất quan trọng. Sự chìa ra dài có thể gây ra sự vấp dao góp phần làm mòn lưỡi cắt. Nhược điểm thứ hai là sự lệch dao. Sự lệch này có ảnh hưởng xấu đến kích cỡ và chất lượng bề mặt chi tiết sau khi gia công. Chiều dài rãnh dao là rất quan trọng để xác định chiều sâu cắt.

Bất kể chiều dài toàn phần của dao (chiều dài nhô ra từ trục chính), chiều dài các rãnh thoát phoi xác định chiều sâu cắt. Hình 31.2 minh họa tỷ số thông dụng giữa chiều rộng và chiều sâu khi phay thô:



Hình 31.2. Quan hệ giữa đường kính dao phay mặt đầu và chiều sâu cắt khi phay thô.

Số lượng rãnh thoát phoi

Khi chọn dao phay mặt đầu, đặc biệt là để cắt gọt vật liệu có độ cứng trung bình, số lượng rãnh thoát phoi phải là xem xét chính. Đối với gia công biên dạng, nhiều nhà lập trình chọn (gần như tự động) dao phay mặt đầu bốn – rãnh đối với dao lớn hơn ∅.625 hoặc ∅.750. Dao phay cắt gọt *sâu vào* chi tiết liền khối dọc theo trục Z – thường chỉ có hai rãnh, bất kể đường kính. Kiểu dao phay ăn sâu này còn được gọi là *dao phay mặt đầu cắt gọt tâm*, hoặc *khoan rãnh* (dao phay ngón). Chú ý, thuật ngữ khoan rãnh hoàn toàn không liên quan với mũi khoan, chỉ liên quan đến tác động gia công, ăn sâu vào vật liệu nguyên khối, song song với trục Z.

Cần đặc biệt chú ý các dao phay có đường kính nhỏ đến trung bình. Trong khoảng kích thước này, dao phay mặt đầu có cấu hình hai -, ba -, và bốn – rãnh thoát phoi, được sử dụng tùy theo *loại vật liệu gia công*.

Dao phay mặt đầu càng ít rãnh thoát phoi, sự tích tụ phoi càng giảm khi cắt gọt với lượng ăn dao lớn, đơn giản do rãnh lớn hơn, dễ thoát phoi hơn. Tuy nhiên, khi giảm số lượng rãnh thoát, tốc độ cắt lập trình càng giảm do lưỡi cắt làm việc nặng hơn. Khi cắt gọt vật liệu mềm, chẳng hạn hợp kim Al, Mg, Cu, điều quan trọng là giảm sự tích tụ phoi, do đó thường dùng dao hai rãnh dù phải giảm tốc độ cắt.

Đối với các vật liệu cứng hơn, cần xét thêm hai yếu tố – *sự vấp dao* và *sự lệch dao*. Khi cắt gọt thép, dao phay mặt đầu nhiều rãnh sẽ bị lệch ít hơn, ít vấp dao hơn so với loại chỉ có hai rãnh.

Dao phay mặt đầu ba – rãnh là trung gian giữa loại hai - rãnh và bốn – rãnh. Loại dao này không phải là lựa chọn tiêu chuẩn, dù khả năng cắt gọt khá tốt. Nói chung, dao ba – rãnh rất khó đo đường kính một cách chính xác trong xưởng cơ khí chỉ có thước cặp hoặc panme.

Bất kể số lượng rãnh thoát phoi, dao phay mặt đầu với đường kính lớn sẽ bị lệch ít hơn so với dao cùng loại có đường kính nhỏ. Ngoài ra, chiều dài hiệu dụng của dao phay mặt đầu (đo từ mặt ổ dao) cũng rất quan trọng. Dao càng dài, độ lệch càng lớn. Độ lệch này là độ nghiêng giữa trục dao và đường tâm trục chính.

TỐC ĐỘ TRỤC CHÍNH VÀ TỐC ĐỘ CẮT

Trong các chương trước đã đề cập đến tốc độ trục chính và tốc độ cắt gọt. Công thức tiêu chuẩn (hệ Anh) được dùng để tính toán tốc độ trục chính r/min (vòng/phút) như sau:

$$r/min = \frac{12 \times ft/min}{\pi \times D}$$

Trong đó:

r/min	= tốc độ trục chính (số vòng quay/phút)
12	= hệ số chuyển đổi foot sang inch.
ft/min	= tốc độ bề mặt tính theo feet/phút
π	= 3.1416
D	= Đường kính dao (inch)

Đối với hệ mét, công thức như sau :

$$r/min = \frac{1000 \times m/min}{\pi \times D}$$

Trong đó:

r/min	= tốc độ trục chính (số vòng quay/phút)
1000	= hệ số chuyển đổi milimét sang mét
m/min	= tốc độ bề mặt tính theo mét/phút
π	= 3.1416
D	= Đường kính dao (mm)

Công thức tính tốc độ cắt theo hệ Anh:

$$ft/min = \frac{\pi \times D \times r/min}{12}$$

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Công thức tính tốc độ cắt có hệ mét:

$$m/min = \frac{\pi \times D \times r/min}{1000}$$

Để tính tốc độ cắt đối với nguyên công phay bất kỳ, cần biết trước tốc độ trục chính r/min. Ngoài ra, cũng cần biết số lượng lưỡi cắt/rãnh thoát phoi và tải phoi trên từng rãnh. Đối với đơn vị Anh, tải phoi được đo theo *inch/răng* (răng có thể coi là lưỡi cắt), viết tắt là *in/tooth* (răng). Kết quả là tốc độ cắt tính theo in/min.

Đối với tốc độ cắt trên máy tiện sử dụng dao tiêu chuẩn, số rãnh thoát phoi là không áp dụng được, do đó phải dùng inch/vòng quay (in/rev) hoặc milimét/vòng quay (mm/rev).

$$in/min = r/min \times f_t \times N$$

Trong đó:

in/min	=	tốc độ cắt tính theo inch/phút
r/min	=	tốc độ trục chính theo số vòng quay/phút
f_t	=	tải phoi tính theo inch/răng (in/tooth)
N	=	số răng (số lưỡi cắt)

Đối với hệ mét, tải phoi được tính theo milimét/răng (lưỡi cắt), viết tắt là mm/tooth. Công thức theo hệ mét như sau:

$$mm/min = r/min \times f_t \times N$$

Trong đó:

mm/min	=	tốc độ cắt tính theo milimét/phút
r/min	=	số vòng quay/phút
f_t	=	tải phoi tính theo mm/răng (mm/tooth)
N	=	số răng (số lưỡi cắt)

Ví dụ, dao phay mặt đầu ∅.750 bốn rãnh thoát phoi cắt gọt gang với tốc độ 100ft/min, tải phoi là .004 in/tooth. Các tính toán như sau:

Tốc độ trục chính

r/min = (12 x 100) / (3.14 x .750)
r/min = 509

Tốc độ cắt

in/min = 509 x .004 x 4
in/min = 8.1

Vì các lý do an toàn, bạn cần xét sự gá lắp chi tiết và máy, độ cứng vững, chiều sâu và/hoặc chiều rộng cắt, và các điều kiện khác.

Đại lượng f_t (tải phoi, in/tooth hoặc mm/tooth) có thể tính từ công thức trên. Đối với hệ Anh:

$$f_t = \frac{in/min}{r/min \times N}$$

Đối với hệ mét:

$$f_t = \frac{mm/min}{r/min \times N}$$

Khi cắt gọt dao phay mặt đầu chắp mảnh carbide để cắt gọt thép, nên dùng tốc độ trục chính cao hơn. Tại các tốc độ thấp, dao carbide tiếp xúc với thép nguội. Khi tốc độ trục chính tăng, nhiệt độ thép ở lưỡi cắt tăng, làm giảm độ bền, do đó dễ cắt gọt hơn. Dao carbides có tốc độ cắt cao hơn ba đến năm lần so với dao thép gió tiêu chuẩn. Hai nguyên tắc cơ bản về quan hệ giữa vật liệu dụng cụ cắt và tốc độ trục chính có thể tóm tắt như sau:

Dao thép gió (HSS) sẽ mòn rất nhanh nếu được dùng với tốc độ trục chính cao.

Dao carbides có thể bị mẻ hoặc gãy nếu tốc độ trục chính quá chậm.

Chất làm nguội và chất bôi trơn

Sử dụng chất làm nguội khi dùng dao thép gió để cắt gọt hầu hết các vật liệu kim loại gần như là yêu cầu bắt buộc. Chất làm nguội kéo dài tuổi bền dụng cụ cắt và các thuộc tính bôi trơn của chúng làm tăng độ bóng bề mặt gia công. Mặt khác, đối với dao chắp carbide có thể *không cần* dùng chất làm nguội, đặc biệt khi cắt gọt thô cho thép.

Không phun chất làm nguội trên lưỡi cắt đã ăn vào vật liệu.

CẮT GỌT PHÔI

Mặt dầu phay chu vi chủ yếu là gia công bán tinh và gia công tinh, nhưng dao phay mặt đầu cũng được dùng để gia công thô. Cấu hình rãnh thoát phoi và lưỡi cắt là khác nhau giữa gia công thô và gia công tinh. Dao phay mặt đầu gia công thô thông dụng có các lưỡi cắt dợn sóng – ví dụ *dao phay mặt đầu Strasmann*, tên của người đầu tiên thiết kế loại dao này.

Khi cắt gọt, thường dùng dao phay mặt đầu đường kính lớn với chiều dài hiệu dụng nhỏ để loại bỏ hoặc giảm thiểu sự vấp dao và lệch dao khi cắt gọt với lượng ăn dao lớn.

Đối với các hốc sâu, nên khoan trước đến chiều sâu toàn phần, sau đó dùng lỗ này cho dao phay có đường kính hơi nhỏ hơn lỗ khoan. Do dao phay mặt đầu ăn vào chiều sâu trong khoảng không gian mở, các lần cắt kế tiếp chủ yếu là phay cạnh, mở rộng hốc đến chiều sâu, hình dạng và kích thước mong muốn.

Ăn dao thẳng xuống

Dao phay mặt đầu ăn vào vật liệu gia công chỉ theo trục Z được gọi là *cắt tâm* hoặc *phay khoan*. Đây là nguyên công cắt gọt thông dụng và các bước lập trình để tiến vào khu vực khó tiếp cận, chẳng hạn hốc sâu, rãnh kín,... Kiểu

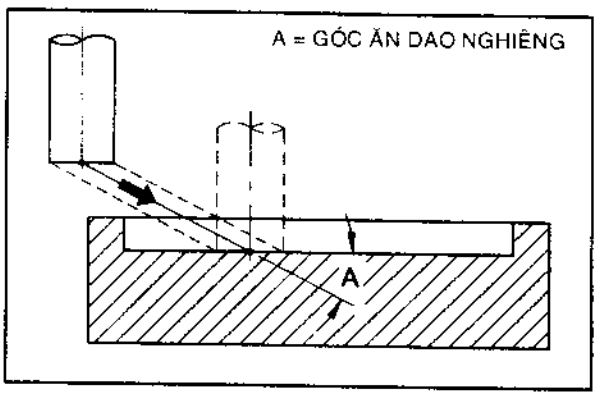
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

gia công này đòi hỏi loại dao chuyên dùng, thường là dao phay ngón.

Ăn dao nghiêng vào và ra

Ăn dao nghiêng là phương pháp sử dụng trục Z để xuyên vào chi tiết gia công, được lập trình đồng thời với trục X *hoặc* trục Y. Tùy theo đường kính dao phay, góc nghiêng khoảng 25^0 với dao ∅1.0, 8^0 với dao ∅2.0, và 3^0 với dao ∅4.0 inch. Tiếp cận nghiêng hướng đến chi tiết có thể áp dụng cho dao phay đáy phẳng, mũi cầu, hoặc mũi cong. Dao phay có đường kính càng nhỏ, sử dụng góc nghiêng càng nhỏ (3^0 – 10^0) (Hình 31.3).



Hình 31.3. Góc ăn dao khi ăn dao nghiêng.

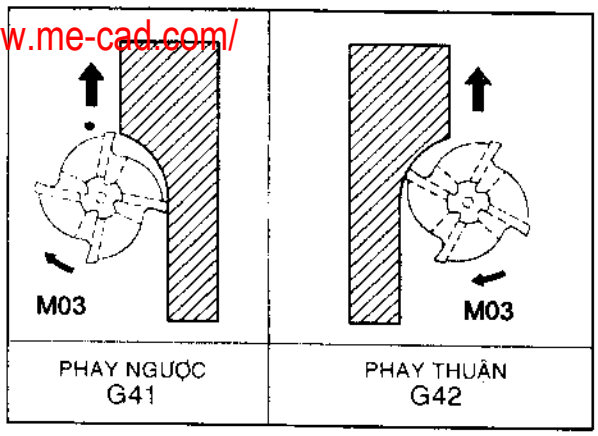
Bạn hãy luôn luôn chú ý vị trí dao XYZ, từ đó dao sẽ bắt đầu cắt gọt ở mặt trên của chi tiết. Nếu chỉ xét điểm bắt đầu và điểm kết thúc, có thể sẽ không đủ. Dễ dàng nhận được vị trí dao bắt đầu và kết thúc, nhưng trong khi cắt gọt có thể vô tình cắt phạm vào phần không mong muốn. Hệ thống CAD và các tính toán sẽ giúp bạn tránh được vấn đề này.

Chiều cắt gọt

Chiều cắt gọt khi gia công biên dạng do nhà lập trình quyết định. Chiều cắt gọt khi phay chu vi sẽ khác nhau tùy theo vật liệu gia công, chủ yếu là chế độ cắt và độ bóng bề mặt. Từ các khái niệm gia công cơ bản, chiều cắt gọt có thể theo một trong hai chế độ:

- Phay ngược – còn gọi là phay ***XUỐNG***
- Phay thuận – còn gọi là phay ***LÊN***

Mỗi khi lệnh G41 được lập trình, bán kính được bù theo bên trái chi tiết và dao là phay *ngược* nếu chiều quay trục chính là bình thường, lập trình với hàm M03, và dao cắt là tay phải. Ngược lại, bù G42, bên phải chi tiết, với kết quả là phay *thuận*. Trong hầu hết các trường hợp, chế độ phay ngược được dùng



Hình 31.4. Chiều cắt gọt liên quan với chi tiết, khi M03 có hiệu lực.

nhiều trong phay biên dạng, đặc biệt là gia công tinh (Hình 31.4).

Phay ngược

Phay ngược – đôi khi còn gọi là phay *xuống* – sử dụng chiều quay của dao theo chiều ăn dao, có xu hướng đẩy chi tiết *vào* bàn máy (hoặc đồ gá). Chiều dày phoi tối đa xảy ra ở đầu cắt, và khi thoát, phoi rất mỏng. Kết quả là hầu hết nhiệt phát sinh được phoi hấp thụ, chi tiết hầu như không bị biến cứng.

> Không nên hiểu sai các từ phay *ngược* và phay *xuống*, chúng biểu thị cùng một chiều cắt gọt.

Cả hai từ này đều đúng, nếu trong ngữ cảnh thích hợp.

Phay thuận

Phay thuận – đôi khi còn gọi là phay *lên* – sử dụng chiều quay của dao ngược chiều ăn dao, có xu hướng kéo chi tiết *ra xa* bàn máy (hoặc đồ gá). Chiều dày phoi tối đa xảy ra ở cuối đường cắt, và khi thoát, phoi rất dày. Kết quả có thể làm biến cứng chi tiết và giảm độ bóng bề mặt.

Chiều rộng và chiều sâu cắt

Để bảo đảm chất lượng, chiều rộng và chiều sâu cắt phải tương ứng các điều kiện gia công, gá lắp, vật liệu cắt gọt, và dao. Chiều rộng cắt còn phụ thuộc vào số rãnh và lưỡi cắt.

Đối với dao phay nhỏ chiều sâu cắt bằng khoảng 1/3 đường kính dao, và hơi tăng đối với dao lớn.

Phay chu vi đòi hỏi kiến thức về gia công cơ khí, tính sáng tạo và kinh nghiệm. Sau mỗi đợt gia công cần ghi chép cẩn thận, có thể rất hữu ích cho nghề nghiệp.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

32 GIA CÔNG RÃNH VÀ HỐC

Trong nhiều ứng dụng trên trung tâm gia công CNC, vật liệu cần được cắt gọt từ bên trong khu vực nào đó, bao quanh bằng vách và đáy phẳng. Quá trình này thường được gọi là gia công *hốc*. Để có hốc thực, vách xác định biên hốc phải khép kín. Tuy nhiên còn có nhiều ứng dụng khác, vật liệu cần được cắt gọt trong vùng hở, chỉ có một phần vách không khép kín. Chương này trình bày các ứng dụng gia công hốc kín, hốc hở, rãnh, và các kỹ thuật lập trình để gia công hốc.

BIÊN HỞ VÀ KÍN

Biên liên tục, trên đó điểm bắt đầu và điểm kết thúc ở các vị trí khác nhau, được gọi là *biên dạng hở*. Biên liên tục được xác định trong chương trình, bắt đầu và kết thúc tại cùng một điểm được gọi là *biên dạng kín*. Theo quan điểm gia công, sự khác biệt cơ bản giữa biên hở và biên kín là *cách thức dao cắt đạt đến chiều sâu biên dạng*.

Biên hở

Biên hở không phải là hốc thực, gia công loại biên dạng này khá linh hoạt, do dao có thể đạt đến chiều sâu mong muốn trong không gian mở. Dao phay chất lượng tốt có thể được dùng để gia công biên hở.

Biên kín

Lượng dư trong biên kín có thể được cắt gọt bằng hai phương pháp, tùy theo nguyên công. Một cách là sử dụng dao gia công ngoài và dịch chuyển dao hướng về vách ngoài của biên, thứ hai là dùng dao gia công trong và dịch chuyển dao hướng vào trong biên Trong cả hai trường hợp, đều là sự gia công thực. Cắt gọt theo phía ngoài chi tiết không phải là gia công hốc mà là phay chu vi (Chương 31). Cắt từ phía trong biên kín là phương pháp phổ biến để gia công hốc với nhiều hình dạng khác nhau. Các hốc có hình dạng đối xứng thường là các rãnh kín, hốc chữ nhật, hốc tròn, ... Hốc có hình dạng không đều phải là loại có dạng hình học gia công được, áp dụng các kỹ thuật gia công và lập trình như hốc đối xứng.

Một trong các hình dạng biên được gia công phổ biến nhất là phay hốc đơn giản, có kích cỡ nhỏ, được gọi là *rãnh*.

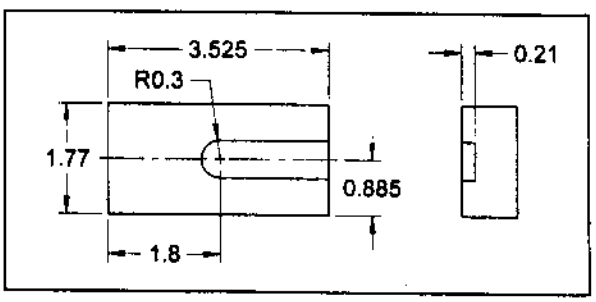
LẬP TRÌNH CÁC RÃNH

Các rãnh thường có một hoặc hai đầu bán nguyệt (có bán kính). Nếu là loại hai đầu, chúng nối với nhau bằng rãnh thẳng. Rãnh có thể hở hoặc kín, với cùng giá trị bán kính ở cả hai đầu, các bán kính khác nhau, hoặc chỉ một bán kính. Rãnh có một bán kính thường là rãnh khóa hoặc rãnh then.

Các rãnh có thể là hở hoặc kín, thẳng, nghiêng, tròn, với vách phẳng hoặc vát (sử dụng dao phay côn). Lập trình gia công rãnh với độ chính xác cao thường đòi hỏi gia công thô và gia công tinh. Cả hai nguyên công có thể thực hiện với một, hai, hoặc nhiều dao, tùy theo vật liệu chi tiết, dung sai, độ bóng bề mặt, và các điều kiện khác.

Một số rãnh, chẳng hạn rãnh then, có thể được thực hiện với dao chuyên dùng, được gọi là dao phay rãnh. Lập trình dao phay rãnh thường là quy trình đơn giản với chuyển động tuyến tính vào và ra. Phức tạp hơn, và chính xác hơn, các rãnh được gia công bằng dao phay ngón, các vách rãnh được gia công biên dạng với sự điều khiển của chương trình.

Hình 32.1 minh họa bản vẽ rãnh hở thông dụng. Bản vẽ này được dùng để minh họa các kỹ thuật lập trình gia công rãnh hở.



Hình 32.1. *Ví dụ lập trình hở O3201*

Ví dụ về rãnh hở

Trước khi lập trình chuyển động dao bất kỳ, bạn hãy nghiên cứu bản vẽ. Từ đó có thể thiết lập các điều kiện gia công, gá lắp, và các yêu cầu khác. Có thể nhanh chóng xác định zero chương trình – các kích thước từ góc dưới bên trái(XY) và mặt trên (Z) của chi tiết. Vị trí đó sẽ trở thành zero chương trình.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Các vấn đề kế tiếp liên quan đến gia công bao gồm:

- Số lượng dao.
- Kích cỡ dao.
- Tốc độ trục chính và tốc độ cắt.
- Chiều sâu cắt cực đại.
- Phương pháp cắt gọt.

Số lượng dao cắt

Có thể dùng một hoặc hai dao để cắt rãnh. Nếu có yêu cầu cao về dung sai kích thước hoặc vật liệu khó cắt gọt, cần dùng hai dao – một để cắt thô, và một để gia công tinh. Các dao đó có thể đường kính bằng hoặc khác nhau. Đối với ví dụ này chỉ dùng một dao để cắt thô và tinh.

Kích cỡ dao

Kích cỡ dao cắt chủ yếu xác định theo chiều rộng rãnh. Trên bản vẽ, rãnh có bán kính .300, vậy chiều rộng rãnh là .600 inch, cần chọn dao hơi nhỏ hơn chiều rộng rãnh. Đối với rãnh trong ví dụ này, dao ∅.500 có lẽ là lựa chọn thích hợp. Khi chọn kích cỡ dao, bạn phải tính toán lượng dư do dao để lại trên vách rãnh để gia công tinh. Lượng dư quá lớn có thể phải gia công bán tinh. Với dao ∅.500 và chiều rộng rãnh .600, lượng dư để lại cho gia công tinh có thể tính như sau:

$$S = \frac{(W - D)}{2}$$

Trong đó:

S = lượng dư còn lại
W = chiều rộng rãnh (= bán kính rãnh nhân hai)
D = bán kính dao

Lượng dư còn lại trên vách rãnh trong ví dụ nêu trên sẽ là:

S = (.600 - .500)/2 = .050

Đây là lượng dư thích hợp để gia công tinh với một đường cắt

Tốc độ trục chính và tốc độ cắt

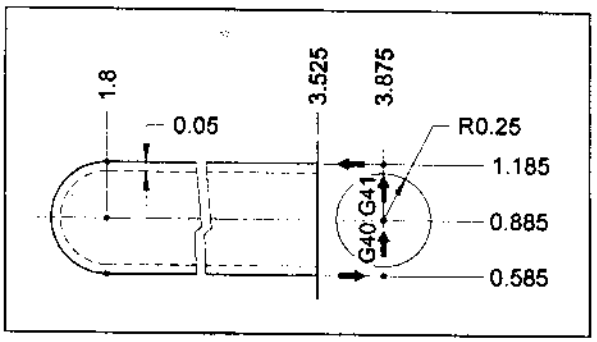
Tốc độ trục chính và tốc độ cắt phụ thuộc vào tình huống cụ thể trên máy CNC, ví dụ này chỉ sử dụng tốc độ hợp lý 950 r/min và tốc độ cắt 8 in/min.

Chiều sâu cắt cực đại

Trên bản vẽ, chiều sâu rãnh là .210 in. Bạn cần luôn luôn xem xét chiều sâu, có thể quá sâu với một lần cắt, đặc biệt khi dao nhỏ hoặc vật liệu cứng. Tuy có thể dùng một lần cắt đến chiều sâu toàn phần, nhưng vẫn cần để lại lượng dư nhỏ ở đáy rãnh để gia công tinh.

Phương pháp cắt gọt

Sau khi thiết lập tất cả các điều kiện gia công, hầu như phương pháp cắt gọt đã rõ ràng. Dao sẽ được định vị ở phía trên, không gian trống, theo đường tâm rãnh. Sau đó dao sẽ ăn vào chiều sâu rãnh theo chế độ phay ngược. Hình 32.2, minh họa các chuyển dao XY và các vị trí lập trình của chúng.



Hình 32.2. Các chi tiết gia công biên dạng đối với ví dụ rãnh hở O3201.

Việc viết chương trình này tương đối dễ dàng. Dao đã lắp trong trục chính, bạn chỉ cần áp dụng các kỹ thuật đã nêu trong phần trước.

```
O3201 (RÃNH HỞ)
N1 G20                                    (INCH MODE)
N2 G17 G40 G80                 (START UP SETTINGS)
N3 G90 G54 G00 X3.875 Y0.885 S950 M03  (START)
N4 G43 Z0.1 H01 M08      (START POSITION ABOVE)
N5 G01 Z-0.2 F50.0        (0.01 LEFT ON BOTTOM)
N6 X1.8 F8.0           (CUT TO SLOT RADIUS CNTR)
N7 G00 Z0.1                 (RETRACT ABOVE WORK)
N8 X3.875                     (RETURN TO START)
N9 G01 Z-0.21 F50.0         (FEED TO FULL DEPTH)
N10 G41 Y1.185 D01 F8.0      (APPROACH CONTOUR)
N11 X1.8                          (CUT TOP WALL)
N12 G03 Y0.585 R0.3            (CUT SLOT RADIUS)
N13 G01 X3.875                 (CUT BOTTOM WALL)
N14 G00 G40 Y0.885      (RETURN TO START POINT)
N15 Z1.0 M09               (RETRACT ABOVE WORK)
N16 G28 X3.875 Y0.885 Z1.0 M05       (M/C ZERO)
N17 M30                            (END OF PROGRAM)
%
```

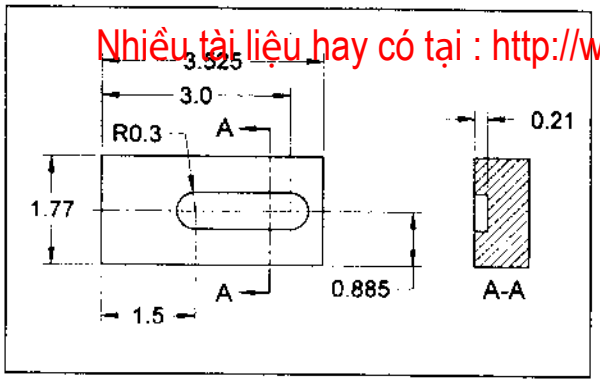
Trong ví dụ này có thêm các ghi chú block để hiểu rõ thứ tự và các bước lập trình. Ở đây chỉ dùng một dao. Để tăng độ chính xác gia công, có thể dùng hai dao nhưng chương trình sẽ dài hơn.

Ví dụ về rãnh kín

Rãnh kín chỉ khác đôi chút so với rãnh hở. Khác biệt lớn nhất là cách thức dao ăn vào vật liệu gia công. Không có vị trí phía ngoài – dao phải ăn vào vật liệu dọc theo trục Z, trừ khi có lỗ khoan sẵn. Một phương là sử dụng *dao phay cắt tâm* (còn gọi là *dao phay ngón*). Nếu không có kiểu dao này, hoặc các điều kiện gia công không thích hợp, dao có thể ăn *nghiêng* vào

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

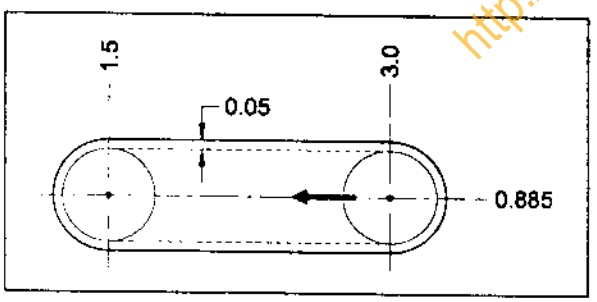


Hình 32.3. *Ví dụ lập trình rãnh kín O3202*

vật liệu. Ăn nghiêng là chuyển động cắt tuyến tính, thường theo các trục XZ, YZ hoặc XYZ.

Ví dụ thứ hai dựa vào bản vẽ trên Hình 32.3. Bản vẽ này là sự chỉnh sửa bản vẽ rãnh hở (Hình 32.2). Các xem xét đã thiết lập ở ví dụ trước sẽ được áp dụng cho rãnh kín. Dao phay Ø.500 sẽ được sử dụng, với dạng hình học cắt gọt tâm cho phép ăn thẳng vào vật liệu.

Ngoài dạng hình học dao cần thiết để gia công kiểu ăn dao thẳng xuống (theo trục Z), phương pháp cắt gọt cũng thay đổi. Đối với rãnh hoặc hốc kín, dao phải chuyển động lên trên chi tiết, đến vị trí XY khởi đầu. Trong ví dụ này, đó sẽ là tâm của một trong các bán kính rãnh. Phần rãnh bên phải được chọn một cách tùy ý. Sau đó, sự ăn sâu với tốc độ cắt nhỏ, tiến đến chiều sâu yêu cầu (để lại lượng dư .01 inch ở đáy), và trong chuyển động tuyến tính sẽ gia công thô phần rãnh giữa hai tâm (Hình 32.4).

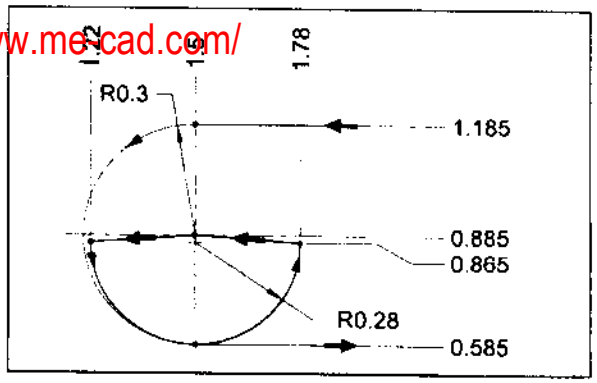


Hình 32.4. *Chi tiết gia công thô đối với ví dụ rãnh kín O3202.*

Sự lùi dao là không cần thiết, dao có thể ăn vào chiều sâu yêu cầu ở cùng một vị trí dao. Lượng dư là .050 trên toàn bộ chu vi vách rãnh. Tại chiều sâu này, và từ vị trí tâm, sau khi gia công cũng thô, sẽ bắt đầu gia công tinh. Lần này, gia công biên dạng sẽ phức tạp hơn, do dao ở vị trí hơi chật.

Tiếp cận biên dạng trong

Trong chương trình này, dao ở tâm bên trái rãnh, sẵn sàng bắt đầu cắt tinh. Chọn chế độ



Hình 32.5. *Chi tiết về tiếp cận tiếp tuyến hướng đến biên dạng trong.*

phay ngược và biên dạng được tiếp cận theo cách thức chuyển động dao liên tục sang trái. Một phương pháp là cắt gọt tuyến tính thẳng từ vị trí dao hiện hành ở tâm tiến đến vị trí cung bên trái (trong khi áp dụng bù bán kính dao).

Phương pháp này có thể dùng được, nhưng khi tiếp cận biên dạng trong, nên sử dụng sự tiếp cận tiếp tuyến. Biên dạng trong được tiếp cận theo tiếp tuyến đòi hỏi cung tiếp cận phụ(được gọi là cung *tiến - vào*) do tiếp cận tuyến tính đến biên dạng sẽ không hiệu quả.

Mặc dầu sự tiếp cận tiếp tuyến sử dụng cung tiến – vào cho phép tăng độ bóng bề mặt, nhưng có thể gây ra vấn đề khác. *Bù dao cắt không thể khởi động trong chế độ nội suy tròn.* Do đó cần *bổ sung* chuyển động không – tròn, sẽ có *hai* chuyển động từ vị trí dao hiện hành tại tâm bán kính rãnh đến điểm bắt đầu biên dạng (Hình 32.5):

- Thứ nhất, chuyển động tuyến tính với bù bán kính dao.
- Thứ hai, chuyển động cung tiếp cận tiếp tuyến.

Bạn hãy xem xét kỹ hơn về phương pháp hình thành cung tiếp cận. Mục đích là chọn vị trí và bán kính cung tiếp cận. Sự chọn vị trí tương đối dễ – cung *phải tiếp tuyến* với biên dạng. Kích cỡ bán kính cần được chọn có cân nhắc. Khi gặp kích thước chưa biết, trước hết bạn hãy suy nghĩ về mục đích của kích thước đó. Mục đích của cung tiếp cận là đưa dao cắt tiến vào theo đường cong hướng đến biên dạng. Điều đó có nghĩa là bán kính cung tiếp cận phải lớn hơn bán kính dao. Ở đây còn có bán kính rãnh, được xác định trên bản vẽ. Quan hệ giữa ba bán kính này như sau:

$$R_t < R_a < R_c$$

Trong đó:

R_t = Bán kính dao cắt
R_a = Bán kính cung tiếp cận (cung tiến – vào)
R_c = Bán kính biên dạng (bán kính rãnh)

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Bán kính biên dạng rãnh (R_t) được xác định từ bản vẽ. Sau khi chọn kích cỡ dụng cụ cắt, bạn sẽ biết bán kính dao (R_t). Còn lại là bán kính cung tiếp cận (R_a). Đây là bán kính cần tính toán và được tính *một cách chính xác*.

Từ công thức, rõ ràng là bán kính cung tiếp cận lớn hơn bán kính dao (.250), cả hai phải nhỏ hơn bán kính biên dạng (.300). Điều đó có nghĩa là bán kính cung tiếp cận sẽ trong khoảng .251 đến .299. Nếu xét số gia .010, đối với chương trình O3202, bán kính .280 được chọn, thỏa công thức nêu trên

$$.250\ (R_t) < .280\ (R_a) < .300\ (R_c)$$

Bạn đã có đủ thông tin cần thiết trước khi viết chương trình. Bạn hãy lưu ý sự tương tự với chương trình gia công rãnh hở O3201.

```
O3202 (RÃNH KÍN)
N1 G20                                    (INCH MODE)
N2 G17 G40 G80                  (STARTUP SETTINGS)
N3 G90 G54 G00 X3.0 Y0.885 S950 M03    (START)
N4 G43 Z0.1 H01 M08     (START POSITION ABOVE)
N5 G01 Z-0.2 F4.0         (0.01 LEFT ON BOTTOM)
N6 X1.5 F8.0        (CUT TO SLOT RADIUS CENTER)
N7 Z-0.21 F2.0               (FEED TO FULL DEPTH)
N8 G41 X1.22 Y0.865 D01 F8.0 (LINEAR APPROACH)
N9 G03 X1.5 Y0.585 R0.28 (CIRCULAR APPROACH)
N10 G01 X3.0                    (CUT BOTTOM WALL)
N11 G03 Y1.185 R0.3   (CUT RIGHT SLOT RADIUS)
N12 G01 X1.5                       (CUT WALL TOP)
N13 G03 Y0.585 R0.3             (CUT RADIUS LEFT)
N14 X1.78 Y0.865 R0.28 (CIRCULAR DEPARTURE)
N15 G01 G40 X1.5 Y0.885    (LINEAR DEPARTURE)
N16 G00 Z1.0 M09          (RETRACT ABOVE WORK)
N17 G28 X1.5 Y0.885 Z1.0 M05        (M/C ZERO)
N18 M30                           (END OF PROGRAM)
%
```

Chương trình O3202 minh hoạ phương pháp tiếp cận biên dạng trong để gia công tinh. Các rãnh khác (chữ nhật, tròn,...), cũng sử dụng các nguyên tắc như trong hai ví dụ nêu trên.

PHAY HỐC

Phay hốc cũng là nguyên công phổ biến trên trung tâm gia công CNC. Phay hốc có nghĩa là cắt gọt vật liệu trong khu vực kín, được xác định bằng biên bao quanh. Khu vực khép kín được xác định bằng các vách và đáy, trong đó vách và đáy có thể là vát, lồi, lõm, tròn, hoặc có hình dạng khác. Các vách hốc tạo ra biên dạng xung quanh. Hốc có thể là hình vuông, chữ nhật, tròn, ..., có thể rỗng bên trong hoặc có các đảo.

Lập trình hốc bằng tay thường chỉ hiệu quả với các hốc đơn giản, hình dạng đều, chẳng hạn chữ nhật hoặc tròn. Các hốc có hình dạng phức tạp cần được lập trình với máy tính.

Nguyên tắc công

Khi lập trình phay hốc cần xét hai vấn đề chính:

- Phương pháp ăn dao
- Phương pháp gia công thô

Để mở khoảng trống để bắt đầu phay hốc (vào vật liệu nguyên khối), chuyển động dao cần được lập trình để đi vào theo chiều trục chính (trục Z), nghĩa là dao phải *cắt tâm*. Nếu khó hoặc không thể cắt tâm, có thể sử dụng phương pháp ăn dao *nghiêng*. Sự ăn dao nghiêng đòi hỏi sử dụng trục Z với trục X, trục Y, hoặc cả hai. Đây sẽ là chuyển động tuyến tính 2-trục hoặc 3-trục. Mọi trung tâm gia công CNC hiện đại đều hỗ trợ phương pháp này.

Phương pháp cắt gọt phần lớn vật liệu trong hốc được gọi là gia công thô. Lựa chọn gia công thô có thể hơi phức tạp. Vị trí, *nơi* dao bắt đầu ăn xuống (trục Z) hoặc ăn nghiêng là rất quan trọng, đó là *chiều rộng* cắt. Có thể khó thực hiện toàn bộ gia công thô chỉ với chế độ phay ngược. Có thể khó để lại lượng dư chính xác để gia công tinh toàn bộ hốc. Nhiều đường cắt sẽ không đều và lượng dư không đồng nhất. Vì lý do đó, cần lập trình gia công bán tinh cho biên dạng hốc trước khi gia công tinh. Một hoặc nhiều dao được dùng cho tình huống đó, tùy theo các yêu cầu chính xác.

Một số phương pháp gia công thô các hốc:

- Đảo chiều cắt vài lần.
- Một chiều – từ trong hốc ra ngoài.
- Một chiều – từ ngoài hốc vào trong.

Trong các ứng dụng máy tính còn có các lựa chọn khác, tùy theo phần mềm và độ phức tạp về hình dạng hốc gia công. Cũng có thể lập trình bằng tay nhưng đòi hỏi tính kiên nhẫn, thời gian, và công sức.

Các kiểu hốc

Các kiểu hốc phổ biến nhất cũng là loại dễ lập trình nhất. Tất cả đều có hình dạng đều đặn, không có các "đảo" bên trong :

- Hốc vuông
- Hốc chữ nhật
- Hốc tròn

Hốc vuông và hốc chữ nhật về cơ bản là như nhau, chỉ chiều dài các cạnh khác nhau, hầu như không có khác biệt trong lập trình.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

HỐC CHỮ NHẬT

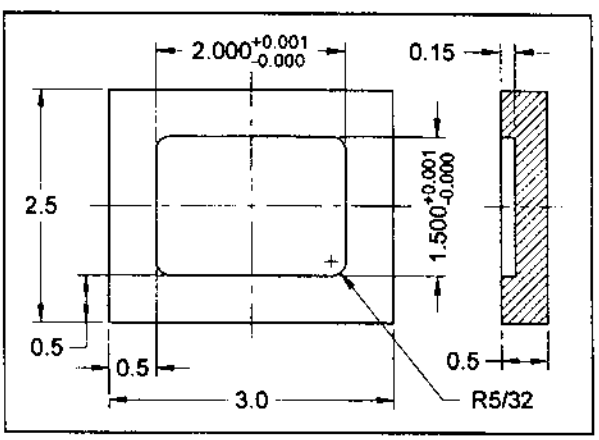
Các hốc vuông và hốc chữ nhật rất dễ lập trình, đặc biệt nếu chúng song song với trục X hoặc Y. Bản vẽ trên Hình 32.6 được dùng cho ví dụ lập trình hốc chữ nhật.

Để minh họa lập trình hốc hoàn chỉnh, điều quan trọng đầu tiên là lựa chọn dao. Vật liệu cũng quan trọng, ngoài ra còn có các điều kiện gia công khác. Mặc dù hốc chữ nhật thường được vẽ với các góc sắc, chúng luôn luôn có các góc bằng hoặc lớn hơn dao khi gia công xong. Các góc trên bản vẽ là R 5/32 (.1563 inch) và dao phay ngón ∅ 5/16 (∅.3125). Đối với gia công thô, đây có thể là lựa chọn tốt, nhưng với gia công tinh, bán kính này phải hơi nhỏ hơn để dao có thể thực sự *cắt* vào các góc, thay vì chỉ tiếp xúc. Chọn dao ∅.250 là hợp lý, dao đó sẽ được sử dụng trong ví dụ lập trình.

Do cần cắt gọt toàn bộ vật liệu trong vùng khép kín (kể cả đáy), bạn hãy suy nghĩ về mọi vị trí dao có thể ăn vào chiều sâu bằng cách chuyển động song song hoặc nghiêng với trục Z. Sự ăn dao nghiêng phải được thực hiện từ khoảng trống, còn ăn xuống có thể thực hiện hầu như ở vị trí bất kỳ nhưng chỉ có hai vị trí có tính thực tiễn:

- Tâm hốc
- Góc của hốc

Mỗi vị trí đều có ưu điểm và nhược điểm riêng, khởi đầu ở tâm hốc, dao có thể theo quỹ đạo một chiều, và sau đường cắt đầu tiên, chỉ có thể cắt theo chế độ phay ngược hoặc phay thuận. Phương pháp này cần tính toán hơi nhiều. Phương pháp khởi đầu ở góc rất được ưa chuộng, sử dụng chuyển động gấp khúc, một đường cắt theo chế độ phay ngược đường thứ hai là phay thuận sau đó lặp lại, nhưng tính toán ít hơn. Trong ví dụ này, góc sẽ được chọn làm vị trí khởi đầu.



Hình 32.6. Bản vẽ hốc chữ nhật–chương trình O3203

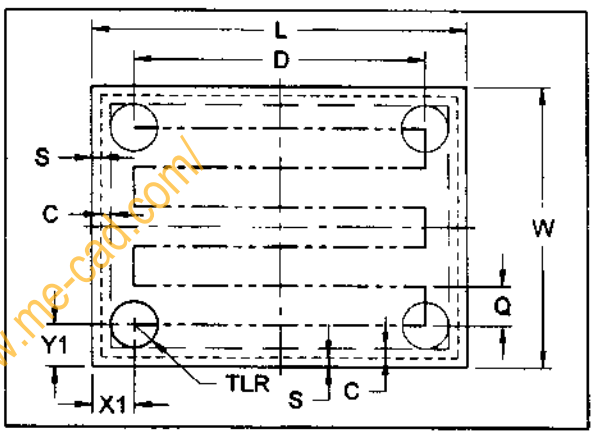
Mọi góc của hốc đều thích hợp như nhau để khởi đầu. Trong chương trình O3203 sẽ sử dụng góc dưới bên trái.

Nhà lập trình cần xét ba yếu tố quan trọng khi chọn vị trí khởi đầu cho dao cắt trong khu vực khép kín :

- Đường kính (hoặc bán kính) dao
- Lượng dư để lại cho gia công tinh
- Lượng dư để lại cho gia công bán tinh.

Các kích thước rất quan trọng trên bản vẽ bao gồm *chiều dài*, *chiều rộng* và *bán kính góc* của hốc. Chúng phải được biết trước, hai yếu tố tiếp theo là vị trí hốc và định hướng của hốc đối với các phần tử khác của chi tiết.

Trên Hình 32.7, điểm bắt đầu được xác định là khoảng cách X1 và Y1 từ góc đã cho (dưới bên trái), cùng với các dữ liệu cần thiết khác.



Hình 32.7. Điểm khởi đầu gia công thô hốc chữ nhật – sử dụng chuyển động gấp khúc.

Các ký tự biểu thị các xác lập cần thực hiện; nhà lập trình chọn giá trị của chúng, tùy theo điều kiện gia công cụ thể.

Ý nghĩa của các ký tự trên Hình 32.7:

X_1	= Vị trí X của dao khi bắt đầu
Y_1	= Vị trí Y của dao khi bắt đầu
TLR	= Bán kính dao
L	= Chiều dài hốc theo bản vẽ
W	= Chiều rộng hốc theo bản vẽ
Q	= Chiều rộng tính toán giữa các đường cắt
D	= Chiều dài tính toán của đường cắt thực
S	= Lượng dư để gia công tinh
C	= Lượng dư để gia công bán tinh (khoảng hở)

Lượng dư gia công

Có hai lượng dư - một liên quan đến gia công *tinh*, thường được thực hiện với dao riêng, và một liên quan với gia công *bán tinh*, thường thực hiện cùng với dao gia công thô. Dao cắt

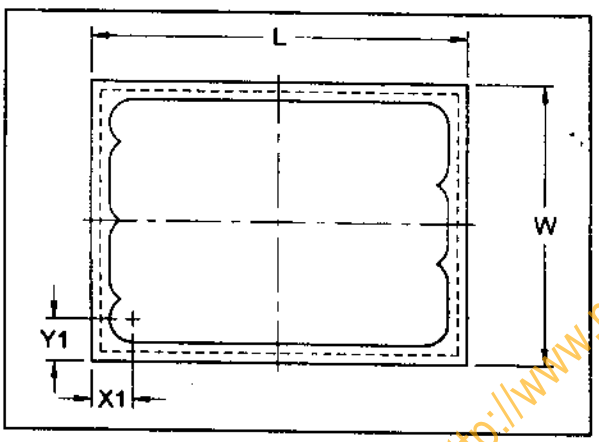
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

chuyển động tới lui theo đường gấp khúc, để lại các vết mấp mô "hình vỏ sò" trên bề mặt vách hốc. Điều này thường rất khó chấp nhận cho gia công tinh, do khó bảo đảm dung sai và độ bóng bề mặt sau khi hoàn tất.

Để tránh hoặc giảm khó khăn khi gia công tinh, cần thực hiện gia công bán tinh. Bạn nên áp dụng gia công bán tinh khi cắt gọt các vật liệu cứng hoặc sử dụng dao có đường kính nhỏ. Lượng dư gia công bán tinh, ký hiệu là C (Hình 32.7), cũng có thể bằng zero, nghĩa là không có lượng dư. Nói chung, lượng dư này thường có giá trị nhỏ.

Hình 32.8 minh họa kết quả gia công thô hốc chữ nhật, *chưa* gia công bán tinh. Bạn hãy chú ý vết mấp mô không đều trên vách hốc. Các điểm cao sẽ là chướng ngại cho chuyển động dao kế tiếp, do đó cần gia công bán tinh trước khi gia công tinh.



Hình 32.8. *Kết quả gia công thô hốc, chưa gia công bán tinh.*

Chiều rộng cắt

Hình dạng thực của hốc trước khi gia công bán tinh được xác định theo *chiều rộng cắt*. Có thể chọn giá trị này mà không cần tính toán, tuy nhiên *sự tính toán* dựa trên số lượng đường cắt thường cho kết quả tốt hơn. Theo đó, giá trị này là bằng nhau đối với mọi đường cắt. Nói chung, chiều rộng đường cắt được xác định theo *số phần trăm* đường kính dao, bạn hãy dùng phương pháp này chỉ để tham khảo, và vẫn *tính* chiều rộng (đường) cắt, sau đó chọn giá trị *gần sát* với số phần trăm đường kính dao mong muốn.

Trong ví dụ này sẽ sử dụng giá trị chiều rộng cắt trung bình, dựa trên năm đường cắt yêu cầu (chuyển động dao theo đường gấp khúc). Luôn luôn có sự khác biệt cơ bản giữa số lượng đường cắt được chọn là *số chẵn* hay *số lẻ*:

- Số lượng đường cắt là chẵn sẽ kết thúc gia công thô ở phía đối diện của hốc liên quan với vị trí khởi đầu.
- Số lượng đường cắt là lẻ sẽ kết thúc gia công thô cùng phía của hốc liên quan với vị trí khởi đầu.

Trong thực tế, hầu như không có vấn đề về chọn góc khởi đầu và chiều bắt đầu đường cắt thứ nhất. Vấn đề là chiều rộng cắt phải hợp lý và bằng nhau cho mọi đường cắt. Dưới đây là cách tính chiều rộng cắt đơn giản nếu biết số đường cắt. Nếu giá trị tính toán quá nhỏ hoặc quá lớn bạn hãy chọn lại số lượng đường cắt N.

Công thức tính chiều rộng cắt:

$$Q = \frac{W - 2 \times TLR - 2 \times S - 2 \times C}{N}$$

Với N là số lượng đường cắt đã chọn, còn các đại lượng khác đã nêu trên Hình 32.7.

➲ **Ví dụ:**

Trong ví dụ này, cần có năm chiều rộng cắt bằng nhau, dựa trên chiều rộng hốc 1.500 inch, đường kính dao 0.250 (TLR là 0.125), lượng dư gia công tinh S = 0.025 và lượng dư bán tinh là C = 0.010. Giá trị chiều rộng cắt sẽ là :

Q = (1.5 – 2 x 0.125 – 2 x 0.025 – 2 x 0.01)/5
Q = 0.2360

Giá trị này có thể hơi quá cao so với dao ∅0.250, nhưng sẽ rút ngắn chương trình ví dụ. Bảy đường cắt sẽ có giá trị chiều rộng cắt hợp lý hơn (0.1686, làm tròn đến 0.169).

Công thức nêu trên có thể được chỉnh sửa để sử dụng chiều dài hốc thay vì chiều rộng. Đây có thể là lựa chọn tốt hơn nếu hốc hẹp theo trục X và rộng theo trục Y.

Chiều dài cắt

Trước khi gia công bán tinh, chiều dài, khoảng cách số gia D của từng đường cắt, cần được tính toán hợp lý.

Nói chung, công thức tính chiều dài cũng tương tự công thức chiều rộng cắt.

$$D = L - 2 \times TLR - 2 \times 5 - 2 \times C$$

Trong ví dụ này, giá trị D sẽ là:

➲ **Ví dụ:**

D = 2.0 – 2 x 0.125 – 2 x 0.025 – 2 x 0.01
D = 1.6800

Đây là chiều dài *số gia* của đường cắt giữa các chiều rộng cắt (không bù bán kính dao).

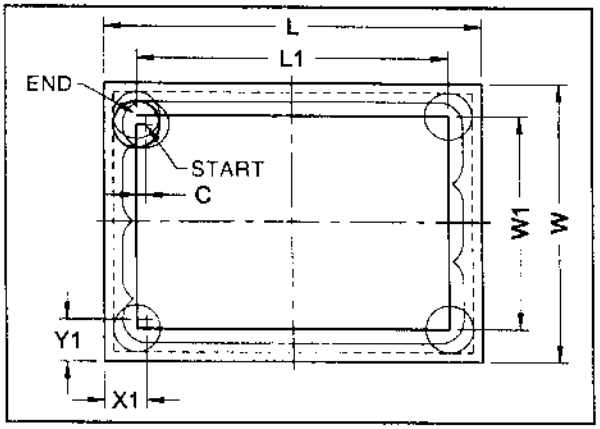
Chuyển động bán tinh

Mục đích duy nhất của gia công bán tinh là

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

loại bỏ lượng dư không đều. Do gia công bán tinh thường sử dụng dao như gia công thô, vị trí khởi đầu chuyển động bán tinh là vị trí dao cuối cùng trong chuỗi cắt thô. Trong trường hợp này, đó là góc trên bên trái của hốc. Hình 32.9 minh họa các chuyển động từ *Start* (bắt đầu) đến *End* (kết thúc) gia công bán tinh.



Hình 32.9. *Quỹ đạo dao cắt bán tinh bắt đầu từ vị trí dao kết thúc gia công thô, chừa lại lượng dư đồng đều cho gia công thô.*

Cần tính toán chiều dài L_1 và chiều rộng W_1, hiệu giữa vị trí Start và vị trí End là giá trị C dọc theo cả hai trục.

Công thức tính chiều dài và chiều rộng gia công bán tinh, khoảng cách cắt gọt thực, như sau:

$$L_1 = L - 2 \times TLR - 2 \times S$$

$$W_1 = W - 2 \times TLR - 2 \times S$$

➲ **Ví dụ:**

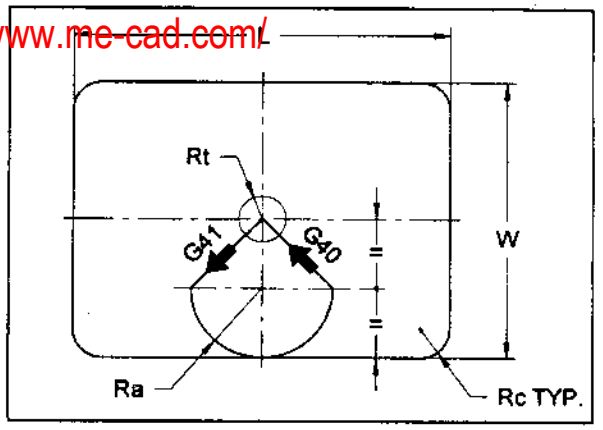
$L_1 = 2.0 - 2 \times 0.125 - 2 \times 0.025 = 1.7000$

$W_1 = 1.5 - 2 \times 0.125 - 2 \times 0.025 = 1.2000$

Quỹ đạo dao gia công tinh

Sau khi hốc được gia công thô và bán tinh, dao khác (hoặc giữ nguyên dao cũ, trong một số trường hợp) được dùng để gia công tinh hốc đến kích thước yêu cầu. Quỹ đạo lập trình sẽ được bù để duy trì khoảng dung sai, tốc độ trục chính và tốc độ cắt để bảo đảm độ bóng bề mặt theo yêu cầu. Vị trí khởi đầu quỹ đạo dao đối với hốc nhỏ đến trung bình thường là tâm điểm, đối với hốc lớn, vị trí này ở gần một trong các vách, nhưng không quá gần.

Đối với gia công tinh, bù bán kính dao phải có hiệu lực, chủ yếu để có tính linh hoạt nhằm đảm bảo dung sai trong khi gia công. Do bù bán kính dao không thể khởi động với chuyển động cung hoặc tròn, cần bổ sung chuyển động tuyến



Hình 32.10. *Quỹ đạo dao gia công tinh hốc chữ nhật*

tính vào và ra. Hình 32.10 minh họa quỹ đạo dao gia công tinh hốc chữ nhật (bắt đầu từ tâm hốc).

Trong những trường hợp này cần áp dụng một số điều kiện. Thứ nhất là tính toán bán kính cung dẫn vào, sử dụng phương pháp hoàn toàn như khi gia công rãnh:

$$R_a > R_t \quad \text{và} \quad R_t < R_c$$

Trong đó:

R_a = Bán kính cung tiếp cận
R_t = Bán kính dao cắt
R_c = Bán kính góc

Chế độ phay cắt thường là phay ngược và bù bán kính dao là G41, từ bên trái biên dạng.

➲ **Ví dụ:**

Tính toán bán kính tiếp cận (tiến vào) cho chi tiết trên bản vẽ (Hình 32.10), bắt đầu với bán kính góc. Bán kính này là 5/32 (.1563 inch) và bán kính dao được chọn là .125, do đó thỏa điều kiện $R_t<R_c$. Để thỏa điều kiện $R_a>R_t$, cần chọn bán kính tiếp cận lớn hơn bán kính dao một cách hợp lý. Chiều rộng và chiều dài hốc cũng rất quan trọng. Nếu có thể, hãy chọn bán kính tiếp cận bằng *một phần tư* chiều rộng hốc W để dễ tính toán chuyển động dao. Trong ví dụ này,

$R_a = W/4 = 1.5/4 = .375$ inch

Điều kiện được thỏa mãn, bán kính tiếp cận lớn hơn bán kính dao, có thể sử dụng một cách an toàn trong chương trình.

Chương trình phay hốc chữ nhật

Sau khi thực hiện mọi tính toán cần thiết, có thể viết chương trình O3203 cho bản vẽ ví dụ. Sử dụng hai dao phay có cùng đường kính Ø.250, dao cắt thô phải có khả năng khoan tâm. Zero chương trình là góc dưới bên trái của chi tiết. Các bước gia công thô và bán kính được chú thích rõ trong chương trình.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
O3203 (HỐC CHỮ NHẬT)
N1 G20
N2 G17 G40 G80 T01 (.250 DAO PHAY NGÓN)
N3 M06
N4 G90 G54 G00 X0.66 Y0.66 S1250 M03 T02
N5 G43 Z0.1 H01 M08
N6 G01 Z-0.15 F7.0
(-- KHỞI ĐẦU CẮT THÔ ----------------------)
N7 G91 X1.68 F10.0                    (CUT 1)
N8 Y0.236                        (STEPOVER 1)
N9 X-1.68 F12.0                       (CUT 2)
N10 Y0.236                       (STEPOVER 2)
N11 X1.68                             (CUT 3)
N12 Y0.236                       (STEPOVER 3)
N13 X-1.68                            (CUT 4)
N14 Y0.236                       (STEPOVER 4)
N15 X1.68                             (CUT 5)
N16 Y0.236                       (STEPOVER 5)
N17 X-1.68                            (CUT 6)
(-- KHỞI ĐẦU CẮT BÁN TINH ----------------)
N18 X-0.01              (SEMIFINISH STARTUP X)
N19 Y-0.01              (SEMIFINISH STARTUP Y)
N20 Y-1.19                    (LEFT Y- MOTION)
N21 X1.7                    (RIGHT X+ MOTION)
N22 Y1.2                      - (UP Y+ MOTION)
N23 X-1.7                     (LEFT X- MOTION)
N24 G90 G00 Z0.1 M09
N25 G28 Z0.1 M05
N26 M01

N27 T02                (.250 FINISHING END MILL)
N28 M06
N29 G90 G54 G00 X1.5 Y1.25 S1500 M03 T01
N30 G43 Z0.1 H02 M08
N31 G01 Z-0.15 F12.0
(-- GIA CÔNG TINH ------------------------)
N32 G91 G41 X-0.375 Y-0.375 D02 F15.0
N33 G03 X0.375 Y-0.375 R0.375 F12.0
N34 G01 X0.8437
N35 G03 X0.1563 Y0.1563 R0.1563
N36 G01 Y1.1874
N37 G03 X-0.1563 Y0.1563 R0.1563
N38 G01 X-1.6874
N39 G03 X-0.1563 Y-0.1563
N40 G01 Y-1.1874
N41 G03 X0.1563 Y-0.1563 R0.1563
N42 X0.8437
N43 G03 X0.375 Y0.375 R0.375
N44 G01 G40 X-0.375 Y0.375 F15.0
N45 G90 G00 Z0.1 M09
N46 G28 Z0.1 M05
N47 X-2.0 Y10.0
N48 M30
%
```

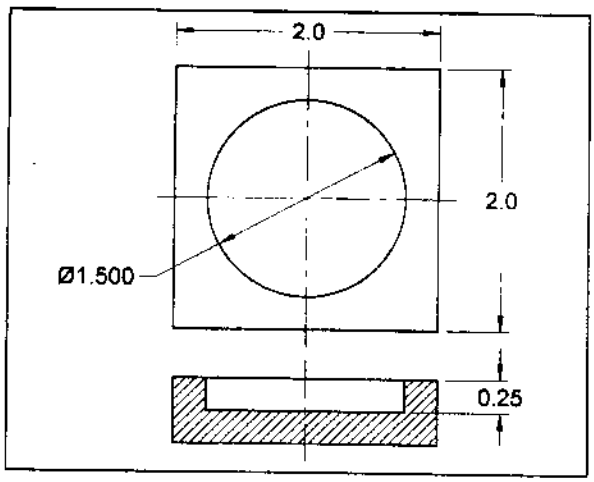
Bạn hãy nghiên cứu chương trình này một cách cẩn thận. Chương trình O3203 sử dụng mọi dữ liệu đã tính toán và nhiều chi tiết đã đề cập ở phần trên.

Trong chương trình này, các block N17 và N18 có thể kết hợp thành một. Điều này cũng có thể áp dụng cho cả block N19 và N20. Chúng được tách riêng để theo dõi các chuyển động dao đã minh họa. Có thể sử dụng chế độ số gia trong chương trình, nhưng chế độ tuyệt đối có lẽ dễ thực hiện hơn.

HỐC TRÒN

Một trong các kiểu hốc phổ biến là *hốc tròn*. Thuật ngữ *hốc* thường được hiểu là khu vực khép kín xung quanh với đáy ở dưới, nhưng phương pháp lập trình này còn cho phép đối với các hốc tròn có lỗ ở giữa, chẳng hạn một số nguyên công doa ngược.

Để minh họa ứng dụng lập trình thực tiễn đối với hốc tròn, Hình 32.11 là bản vẽ với các kích thước điển hình của loại hốc này.

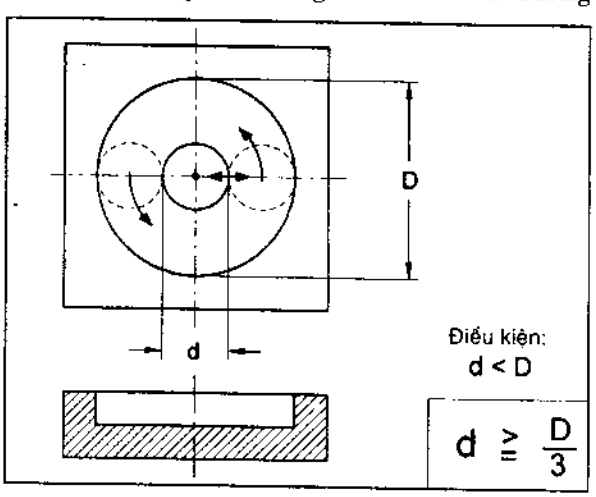


Hình 32.11. Bản vẽ hốc tròn (các ví dụ chương trình O3204-06)

Khi lập kế hoạch, việc đầu tiên là chọn đường kính dao. Bạn cần nhớ, để làm cho đáy hốc phẳng, không có các phần dư (không bị cắt gọt), điều quan trọng là duy trì chiều rộng cắt giữa các đường cắt theo khoảng cách giới hạn có thể tính toán được. Đối với hốc tròn, yêu cầu này ảnh hưởng đến đường kính *tối thiểu* của dao có thể dùng để cắt hốc tròn chỉ với một đường cắt 360^0.

Đường kính dao tối thiểu

Trong minh họa kế tiếp (Hình 32.12) sẽ nêu quan hệ giữa đường kính dao và đường



Hình 32.12. Quan hệ giữa đường kính dao cắt và đường kính hốc.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

kính hốc. Ngoài ra, còn có công thức xác định đường kính dao tối thiểu là *một phần ba* đường kính hốc. Phay sẽ bắt đầu ở tâm hốc tròn, với một chuyển động dao 360^0. Trong thực tiễn thường chọn dao hơi lớn hơn đường kính tối thiểu. Ưu điểm chính của tính toán này là khi hốc được gia công chỉ với một chuyển động dao theo đường tròn. Công thức này xác định chiều rộng tối đa của đường cắt.

Ví dụ, đường kính hốc trên bản vẽ là 1.5 inch. Sử dụng công thức, chọn dao phay cắt tâm có đường kính lớn hơn 1.5/3, tức là lớn hơn .500. Cỡ dao danh định thích hợp sẽ là Ø.625 (dao phay ngón 5/8).

Phương pháp ăn vào

Bước kế tiếp là xác định phương pháp ăn dao vào vật liệu gia công. Trong hốc tròn, vị trí tốt nhất để tiến vào theo trục Z, là *tâm* của hốc. Nếu tâm hốc cũng là zero chương trình X0Y0, và chiều sâu hốc là .250, sự bắt đầu chương trình có thể tương tự ví dụ dưới đây (dao đã lắp vào trục chính) :

```
O3204 (HỐC TRÒN - VERSION 1)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S1200 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.25 F8.0
N6 ...
```

Trong block kế tiếp (N6) dụng cụ cắt sẽ dịch chuyển từ tâm hốc hướng đến đường kính hốc và áp dụng bù bán kính dao dọc theo đường này. Chuyển động đó có thể thực hiện theo hai cách:

- Chuyển động tuyến tính đơn giản.
- Kết hợp chuyển động tuyến tính với tiếp cận tròn.

Tiếp cận tuyến tính

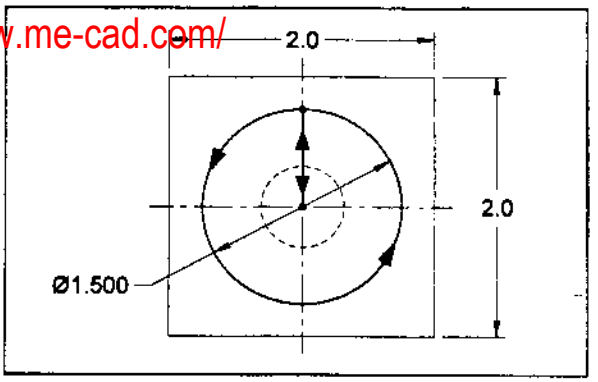
Sự xuất phát tuyến tính từ tâm hốc có thể theo chiều bất kỳ, nhưng chiều đến điểm phần tư có tính thực tiễn cao hơn. Trong ví dụ, chuyển động theo chiều Y dương, đến vị trí 90^0.

Theo đường này, sẽ lập trình bù bán kính dao G41 với chế độ phay ngược, tiếp theo là cung toàn phần 360^0 và chuyển động thẳng, trở lại tâm hốc. Trong chuyển động này sẽ xóa bù bán kính dao (Hình 32.13).

Đoạn chương trình tương ứng sẽ thực hiện các chuyển động gồm tiếp cận điểm phần tư, gia công biên dạng cung toàn phần, và trở lại tâm hốc:

```
N6 G41 Y0.75 D01 F10.0
N7 G03 J-0.75
N8 G01 G40 Y0 F15.0
```

Dao đến tâm điểm, hoàn tất gia công hốc



Hình 32.13. Tiếp cận tuyến tính khi phay hốc tròn – chương trình O3204

tròn. Dao phải lùi lại và trở về zero máy (chuyển động G28 luôn luôn theo chế độ chuyển động nhanh):

```
N9 G28 Z-0.25 M09
N10 G91 G28 X0 Y0 M05
N11 M30
%
```

Phương pháp này rất đơn giản, nhưng chưa hẳn là tốt nhất, đặc biệt đối với các yêu cầu cao về dung sai và độ bóng bề mặt. Các dung sai bản vẽ có thể đạt được bằng gia công thô với một dao và gia công tinh với một hoặc vài dao khác.

Các vết dao (vết gia công) có thể ở điểm tiếp xúc với đường kính hốc, là khả năng dễ xuất hiện khi tiếp cận thẳng đến đường kính hốc. Sự tiếp cận tuyến tính rất hiệu quả khi hốc không có các yêu cầu quá cao. Dưới đây là chương trình O3204 hoàn chỉnh:

```
O3204 (HỐC TRÒN - VERSION 1)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S1200 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.25 F8.0
N6 G41 Y0.75 D01 F10.0
N7 G03 J-0.75
N8 G01 G40 Y0 F15.0
N9 G28 Z-0.25 M09
N10 G91 G28 X0 Y0 M05
N11 M30
%
```

Kỹ thuật lập trình thứ hai đối với hốc tròn có tính thực tiễn cao hơn bảo đảm độ bóng bề mặt tốt hơn và dung sai chặt chẽ hơn. Thay vì một tiếp cận tuyến tính trực tiếp hướng đến đường kính hốc, dao cắt có thể áp dụng sự tiếp cận phối hợp tuyến tính - tròn.

Tiếp cận tuyến tính – tròn

Đối với phương pháp này, sẽ thay đổi chuyển động dao. Một cách lý tưởng, có thể thực hiện chuyển động một nửa cung nhỏ giữa tâm và điểm khởi đầu của hốc. Điều này chỉ có thể nếu *không* sử dụng bù bán kính dao. Một số

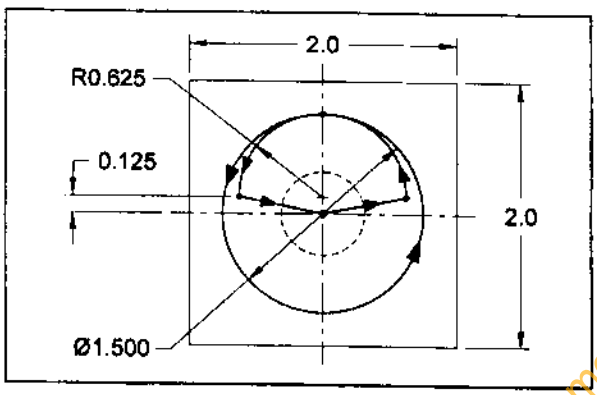
http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

bộ điều khiển áp dụng điều đó (xem ví dụ kế tiếp) khi sử dụng chu kỳ phay hốc tròn G12 hoặc G13. Nếu bộ điều khiển Fanuc có tùy chọn *User Macros*, chu kỳ phay hốc tròn G12 hoặc G13 có thể được lập trình từng bước, mỗi lần một block.

Do cần có bù bán kính dao để bảo đảm dung sai, và chế độ bù không thể khởi đầu trên một cung, trước hết cần lập trình sự tiếp cận tuyến tính và áp dụng bù bán kính dao. Sau đó, lập trình sự tiếp cận tiến vào cung tròn. Khi hốc hoàn tất, quy trình sẽ đảo lại, bù bán kính dao bị xóa khi chuyển động tuyến tính trở về tâm hốc. Tính toán bán kính tiếp cận trong ứng dụng này hoàn toàn như chuyển động dao hoàn tất gia công rãnh đã nêu ở phần trước (Hình 32.14).



Hình 32.14. *Tiếp cận phối hợp chuyển động tuyến tính-tròn để phay hốc tròn – chương trình O3205.*

Ví dụ này sử dụng bán kính tiếp cận .625. Bán kính bất kỳ *lớn hơn* bán kính dao (.3125) và *nhỏ hơn* bán kính hốc (.750) đều đúng. Chương trình O3205 sẽ theo chuyển động tiếp cận trên Hình 32.14.

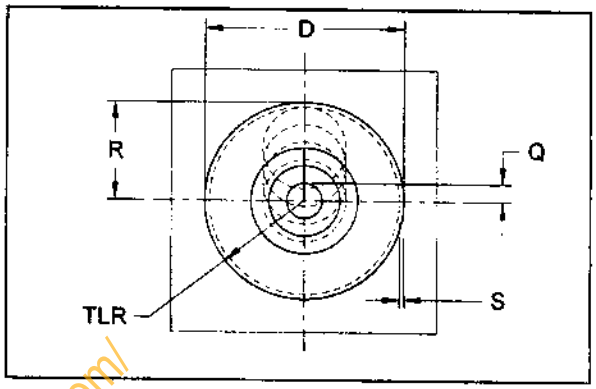
```
O3205 (HỐC TRÒN - VERSION 2)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S1200 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.25 F8.0
N6 G41 X0.625 Y0.125 D01 F10.0
N7 G03 X0 Y0.75 R0.625
N8 J-0.75
N9 X-0.625 Y0.125 R0.625
N10 G01 G40 X0 Y0 F15.0
N11 G28 Z-0.25 M09
N12 G91 G28 X0 Y0 M05
N13 M30
%
```

Kỹ thuật lập trình này ưu việt hơn sự tiếp cận tuyến tính trực tiếp, hoàn toàn không khó khi lập trình, một phần là nhờ tính đối xứng của các chuyển động dao. Phương pháp này có thể và nên áp dụng cho sự tiếp cận bất kỳ hướng đến gia công tinh biên dạng trong.

Gia công hốc tròn

Hốc tròn thường quá lớn đối với dao để bảo đảm đáy phẳng chỉ với một lần cắt gọt. Trong trường hợp đó, trước hết hốc được làm rộng bằng gia công thô, để loại bỏ phần vật liệu dư, sau đó áp dụng quỹ đạo dao gia công tinh. Một số bộ điều khiển có các chu kỳ đặc biệt, ví dụ gia công hốc theo đường xoắn. Trên bộ điều khiển Fanuc, các chu kỳ này có thể được viết với tùy chọn *User Macros*.

Ví dụ, sử dụng lại bản vẽ chi tiết trên Hình 32.11, gia công với dao Ø.375 (Hình 32.15).



Hình 33.15. *Gia công thô hốc tròn – chương trình O3206*

Dao phay Ø.375 tương đối nhỏ *không thể* làm sạch đáy lỗ với phương pháp đã nêu. Phương pháp gia công thô được minh họa trên Hình 32.15, và giá trị Q bằng chiều rộng đường cắt, được tính từ số lần cắt (bước) N, bán kính dao TLR và lượng dư S để gia công tinh.

Tính toán này về logic là tương tự hốc chữ nhật và giá trị chiều rộng cắt mong muốn có thể đạt được bằng cách thay đổi số lần cắt N.

Chương trình O3206 sử dụng ba đường cắt, tính toán từ công thức:

$$Q = \frac{R - TLR - S}{N}$$

Trong đó:

- Q = Chiều rộng tính toán giữa các đường cắt
- R = Bán kính hốc
- TLR = Bán kính dao
- S = Lượng dư gia công tinh
- N = Số đường cắt

Trong ứng dụng này, các giá trị như sau:

R = 1.5/2 = .75 inch *Đường kính D = 1.5 inch*
TLR = .375/2 = .1875
S = .025
N = 3

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Sử dụng công thức nêu trên, có thể tính giá trị Q:

Q = (.75 – .1875 - .025)/3 = .1792

Chương trình gia công thô rất đơn giản, không cần lập trình bù bán kính dao. Bạn hãy lưu ý ưu điểm của chế độ số gia G91, cho phép dễ dàng quan sát giá trị Q trong chương trình, theo chế độ tuyến tính G01. Các block kế tiếp chứa vector cung J, cắt đường tròn kế tiếp. Các bán tinh đường tròn (J) tăng dần theo giá trị chiều rộng cắt Q.

```
O3206 (CẮT THÔ HỐC TRÒN)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S1500 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.25 F7.0
N6 G91 Y0.1792 F10.0                (STEPOVER 1)
N7 G03 J-0.1792                  (ROUGH CIRCLE 1)
N8 G01 Y0.1792                      (STEPOVER 2)
N9 G03 J-0.3584                  (ROUGH CIRCLE 2)
N10 G01 Y0.1792                     (STEPOVER 3)
N11 G03 J-0.5376                 (ROUGH CIRCLE 3)
N12 G90 G01 X0 F15.0
N13 G28 Z-0.25 M09
N14 G91 X0 Y0 M05
N15 M30
%
```

CHU KỲ GIA CÔNG HỐC TRÒN

Trong Chương 28 đã đề cập chu kỳ gia công hốc tròn. Trong chương này, sẽ trình bày hai ví dụ có bổ sung các chi tiết. Fanuc không coi chu kỳ gia công hốc tròn G12 và G13 là tính năng tiêu chuẩn. Các bộ điều khiển có tính năng đó, chẳng hạn Yasrac, có chu kỳ macro viết sẵn bên trong, cho phép sử dụng dễ dàng. Người dùng Fanuc có thể tự viết các macro riêng (chu kỳ mã G đặc biệt), với tính năng tùy chọn *User Macros*, có tính linh hoạt cao hơn so với chu kỳ có sẵn trong bộ điều khiển.

Hai mã G này là đồng nhất, chỉ khác nhau về chiều cắt gọt. Ý nghĩa của mã G trong chu kỳ gia công hốc tròn như sau:

G12	Cắt hốc tròn CW
G13	Cắt hốc tròn CCW

Chu kỳ này *luôn luôn* lập trình với chế độ xóa bù bán kính dao G40 có hiệu lực, và có định dạng chương trình như sau :

```
G12 I.. D.. F.. (PHAY THUẬN)
```

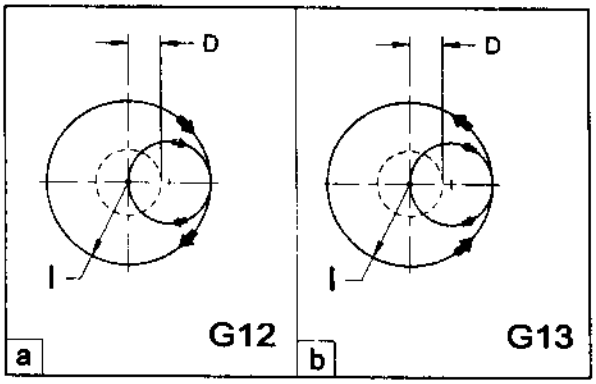
Hoặc

```
G13 I.. D.. F.. (PHAY NGƯỢC)
```

Trong đó:

I = Bán kính hốc
D = Giá trị (số) bù bán kính dao
F = Tốc độ cắt

Nói chung, chu kỳ này được gọi ở *tâm* và *đáy* của hốc. Mọi chuyển động dao đều là chuyển động cung, và có ba chuyển động. Không có chuyển động tuyến tính. Điểm khởi đầu bắt buộc (và điểm kết thúc) trên đường kính hốc là 0^0 (Hình 32.16).



Hình 32.16. Chu kỳ gia công hốc tròn G12 và G13

Ví dụ Hình 32.11 có thể được sử dụng để minh họa chu kỳ G12 hoặc G13. Để so sánh, dưới đây là chương trình O3205 sử dụng dao phay Ø.625:

```
O295 (HỐC TRÒN - VERSION 2)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S1200 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.25 F8.0
N6 G41 X0.625 Y0.125 D01 F10.0
N7 G03 X0 Y0.75 R0.625
N8 J-0.75
N9 X-0.625 Y0.125 R0.625
N10 G01 G40 X0 Y0 F15.0
N11 G28 Z-0.25 M09
N12 G91 G28 X0 Y0 M05
N13 M30
%
```

Nếu có chu kỳ G12, G13, hoặc macro tương tự, chương trình O3206 có thể được viết lại, sử dụng dao như cũ và chế độ phay ngược:

```
O3206 (HỐC TRÒN - SỬ DỤNG G13)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S1200 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.25 F8.0
N6 G13 I0.75 D1 F10.0                (HỐC TRÒN)
N7 G28 Z-0.25 M09
N8 G91 G28 X0 Y0 M05
N9 M30
%
```

Macro là công cụ lập trình rất mạnh, được trình bày chi tiết trong các tài liệu chuyên ngành.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

33 TIỆN VÀ DOA

Nội dung về tiện và doa CNC là rất nhiều đủ cho hàng chục cuốn sách. Chương này chỉ giới thiệu một số vấn đề cơ bản, những nội dung khác sẽ trình bày trong các chương về chu kỳ tiện, gia công rãnh, tiện cắt đứt, tiện ren,...

HÀM DỤNG CỤ CẮT - DAO TIỆN

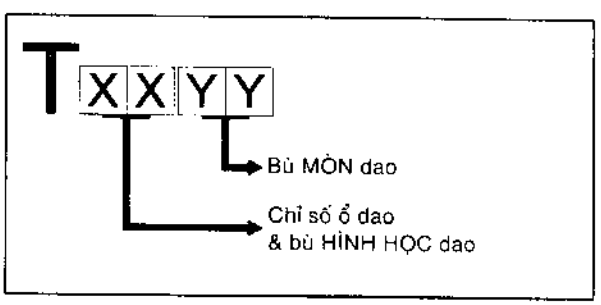
Nói chung, *tiện* và *doa* là các nguyên công hầu như đồng nhất, chỉ khác nhau về khu vực cắt gọt, nơi thực sự gia công. Trong thực tế thường sử dụng thuật ngữ *tiện trong* và *tiện ngoài*, có ý nghĩa tương tự tiện và doa. Dưới góc độ lập trình, các nguyên tắc hầu như đồng nhất, khi cần thiết sẽ trình bày rõ các khác biệt, nếu có.

Tiện CNC đòi hỏi lập trình dao đã chọn theo chỉ số dao, sử dụng địa chỉ T. Để so sánh, với trung tâm gia công CNC, hàm dụng cụ cắt trên máy tiện đòi hỏi nhiều thông tin chi tiết hơn. Một trong các khác biệt lớn giữa bộ điều khiển phay và tiện là địa chỉ T đối với máy tiện CNC *sẽ thực hiện sự thay dao thực sự*. Điều này không xảy ra đối với phay. Hàm M06 không hiện hữu trên máy tiện CNC tiêu chuẩn.

Địa chỉ T

Một khác biệt so với trung tâm gia công là dao được xác định T01 trong chương trình *phải* được lắp vào ổ dao #1 trên cụm ổ dao, dao T12 phải lắp trong ổ dao #12... Khác biệt thứ hai giữa dao phay và dao tiện là *định dạng* của địa chỉ T. Định dạng đối với hệ thống tiện là T4, nói một cách chính xác là T2+2. Hai chữ số đầu xác định chỉ số ổ dao và bù hình học, hai chữ số cuối là chỉ số bù mòn dao đối với ổ dao đã chọn (Hình 33.1).

Định dạng Txxyy biểu thị ổ dao xx và chỉ số



Hình 33.1. *Địa chỉ hàm dụng cụ cắt dùng cho máy tiện CNC.*

bù mòn dao yy. Ví dụ T0202 sẽ làm cho cụm ổ dao phân độ ổ dao #2 (hai chữ số đầu) sẽ là ổ dao làm việc (dao hoạt động). Đồng thời, chỉ số bù mòn dao tương ứng (cặp chữ số cuối) cũng trở nên có hiệu lực.

Sự lựa chọn chỉ số dao (cặp chữ số đầu) cũng chọn bù hình học trên hầu hết các máy tiện CNC hiện đại. Trong trường hợp đó, cặp chữ số thứ hai sẽ chọn chỉ số bù mòn dao. Ổ dao bắt đầu được chọn theo chỉ số cụm ổ dao có thể liên quan với chỉ số bù bất kỳ trong khoảng bù khả dụng. Trong hầu hết các ứng dụng, chỉ một chỉ số dao có hiệu lực đối với dao bất kỳ đã chọn. Trong trường hợp đó, cần lập trình chỉ số bù trùng với chỉ số dao. Phương pháp này giúp cho công việc của người vận hành trở nên dễ dàng hơn. Bạn hãy xét các lựa chọn dưới đây:

```
G00 T0214      Ổ dao 02, bù mòn dao 14
G00 T1105      Ổ dao 11, bù mòn dao 05
G00 T0404      Ổ dao 04, bù mòn dao 04
```

Dù ba ví dụ trên đều đúng về kỹ thuật, nhưng chỉ nên dùng định dạng thứ ba (T0404). Khi dùng nhiều dao trong chương trình, các chỉ số bù đối với từng dao có thể bị nhầm lẫn, nếu chúng không tương ứng với chỉ số ổ dao. Chỉ một trường hợp chỉ số bù khác với chỉ số dao, khi *hai hoặc nhiều chỉ số bù* được gán cho một dao, ví dụ T0202 đối với bù mòn dao thứ nhất, T0222 đối với bù mòn dao thứ hai.

Các chữ số zero đứng trước trong hàm dụng cụ cắt có thể bỏ qua khi chọn chỉ số dao, nhưng không được bỏ qua khi chọn chỉ số bù mòn dao. T0202 sẽ không thay đổi ý nghĩa khi viết là T202. Loại bỏ chữ số zero đứng trước đối với bù mòn dao là không hợp lệ:

T22 có nghĩa là T0022, định dạng không hợp lệ.

Tóm lại, phần hoạt động của ổ dao được lập trình bằng cặp chữ số thứ nhất, chỉ số bù mòn dao là cặp chữ số thứ hai trong hàm dụng cụ cắt.

```
G00 T0404
```

Trong thực tế, bạn không nên sử dụng tính năng loại bỏ chữ số zero đứng trước, hãy dùng hàm dụng cụ cắt với định dạng toàn phần.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tuy chế độ bù đã được trình bày trong phần hàm dụng cụ cắt, nhưng đây là tính năng rất quan trọng đối với hệ thống tiện, do đó cần nêu thêm các chi tiết.

Bù hình học được đo đối với *từng dao* theo *khoảng cách thực từ điểm quy chiếu (chuẩn) dao đến zero chương trình* (trục Z và đường kính X được lưu theo giá trị âm) (Hình 33.2)

Bù hình học xác định vị trí đo từ zero chương trình

Bù mòn dao dùng cho các kích thước tinh chỉnh

Cách tốt nhất để minh họa tầm quan trọng của bù mòn dao là xét chương trình *không* sử dụng tính năng đó. Mọi kích thước lập trình đều là các giá trị lý tưởng, dựa trên bản vẽ. Không xét sự mòn dao và các dung sai lưỡi cắt của dao. Độ sai lệch so với kích thước lập trình xảy ra do kích cỡ thực của dao sẽ dẫn đến kích thước không chính xác khi gia công chi tiết. Bù mòn dao được dùng để "tinh chỉnh" các kích thước thực khi gia công so với kích thước lập trình.

Mục đích của bù mòn dao là điều chỉnh sự khác biệt giữa kích thước lập trình và vị trí thực của dao trên chi tiết. Nếu bộ điều khiển không có tính năng bù mòn dao, các điều chỉnh chỉ có thể thực hiện với chế độ bù khả dụng – đó là bù hình học.

Nhập chế độ bù

Bù dụng cụ cắt có thể nhập vào chương trình theo hai cách :

- Theo lệnh *độc lập* với chuyển động dao.
- Theo lệnh được áp dụng *đồng thời* với lệnh chuyển động dao.

Bù dao độc lập

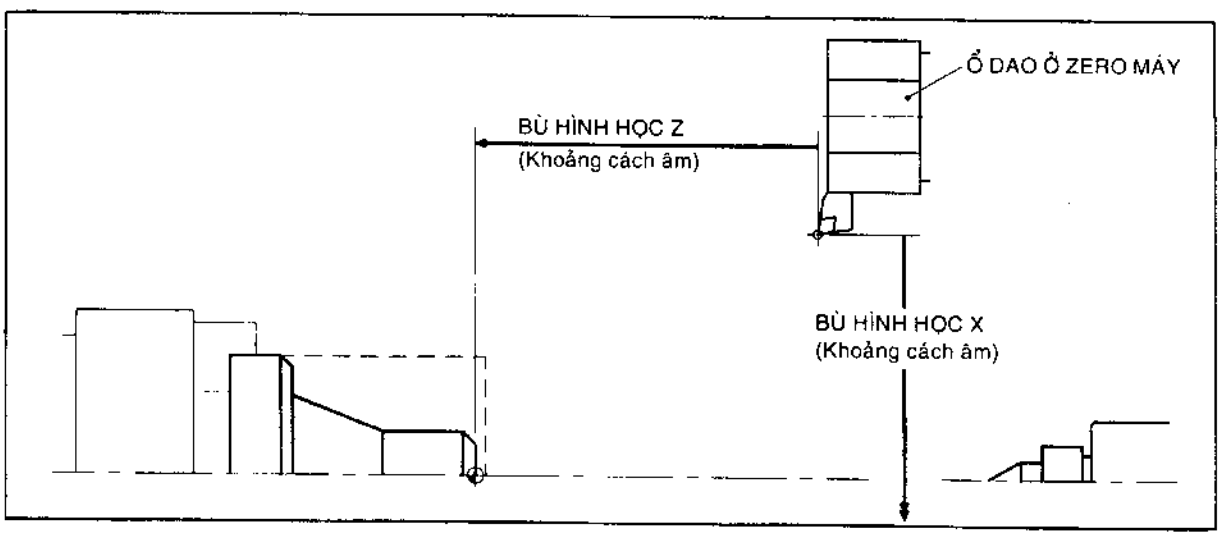
Đối với lệnh nhập bù dao độc lập trong chương trình, bù dao được áp dụng *cùng với phân độ dao*:

```
N34 G00 T0202
```

Lệnh này thường được lập trình là block thứ nhất cho từng dao (ở vị trí trống). Nếu sử dụng bộ ghi vị trí G50 cũ, lệnh bù được lập trình cùng với, hoặc ngay sau block ghi tọa độ. Tại điểm này, dao vẫn ở vị trí phân độ ban đầu. Khi lệnh bù dao được kích hoạt, dao sẽ *chuyển động thực sự* theo giá trị bù, đã lưu trong bộ ghi bù. Chú ý, lệnh chuẩn bị G00 đứng trước hàm dụng cụ cắt. Đây là lệnh rất quan trọng, cho phép xảy ra chuyển động bù dao. G00 quan trọng hơn đối với dụng cụ cắt thứ nhất, nhưng cũng nên lập trình cho dao bất kỳ. Trong Chương 4, *Hệ thống điều khiển*, đã nêu rõ trạng thái của bộ điều khiển khi được kích hoạt. Do hệ điều khiển thường giả thiết (mặc định) lệnh G01 (nội suy tuyến tính) khi khởi động, cần có tốc độ cắt kèm theo. Tuy nhiên, hệ điều khiển sẽ tìm T0202F0.025, dù đó là đúng. Chuyển động nhanh có tính thực tiễn cao hơn và tùy thuộc vào trạng thái điều khiển hiện hành lập trình lệnh G00 sẽ *luôn luôn* kích hoạt chế độ bù.

Bù dao với chuyển động

Phương pháp thứ hai là lập trình bù mòn dao đồng thời với chuyển động dao, thường trong khi dao tiếp cận chi tiết. Đây là phương pháp thông dụng. Hai ví dụ sẽ minh họa lập trình hàm T đối với hệ thống tiện, bù được kích hoạt khi cặp chữ số thứ hai trong nhóm chỉ số dao bằng hoặc lớn hơn 01:



Hình 33.2. *Bù hình học là khoảng cách từ điểm quy chiếu dao đến zero chương trình, đo theo trục từ zero máy.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N1 G20 T0100
N2 G96 S190 M03
N3 G00 X.. Z.. T0101 M08
...
```

Chú ý, sự thay dao trong block thứ nhất N1 – không sử dụng chỉ số bù mòn – chỉ số dao cũng là chỉ số bù hình học. Bù mòn dao được áp dụng trong block N3.

Trong hầu hết các trường hợp, sẽ không có sự khác biệt, khi lệnh bù được kích hoạt dù có hay không có lệnh chuyển động. Nhưng có thể có một số hạn chế khi lập trình nhập bù dao *không có* lệnh chuyển động. Ví dụ, nếu giá trị bù mòn được lưu là quá lớn và dao khởi hành từ vị trí zero máy, kiểu lập trình này có thể gây ra sự hành trình dao quá mức.

Ngay cả trong các trường hợp giá trị bù nhỏ, sẽ vẫn xảy ra chuyển động "nhảy" của cụm ổ dao khi kích hoạt lệnh bù. Một số nhà lập trình không thích chuyển động nhảy đó, dù không ảnh hưởng đến máy. Trong các trường hợp đó, cách tốt nhất có lẽ là kích hoạt bù mòn dao trong chuyển động thứ nhất, thường là chuyển động nhanh hướng đến chi tiết. Một khảo sát rất quan trọng khi bù mòn dao được kích hoạt cùng với chuyển động. Đầu chương này đã đề cập hàm dao tiện cũng là hàm gây ra sự phân độ dao. Tình huống cần tránh là phân độ dao đồng thời với chuyển động dao. Điều này có thể gây ra các hậu quả nghiêm trọng.

Phương pháp tốt nhất có lẽ là khởi động từng dao chỉ với sự phân độ, *không* kèm theo lệnh bù mòn dao:

```
N34 T0200 M42
```

Ví dụ nêu trên sẽ ghi trên xác lập tọa độ đối với dao 2, phân độ dao này vào vị trí làm việc, nhưng *không kích hoạt* lệnh bù bất kỳ (T0200 có nghĩa là phân độ dao 2 không bù mòn dao). Nếu cần, có thể bổ sung hàm khoảng bánh răng (hộp số). Block đó thường được tiếp theo bằng sự lựa chọn tốc độ trục chính, và chuyển động nhanh đến vị trí thứ nhất, gần sát chi tiết. Đó là block, nơi bù mòn dao sẽ được kích hoạt – *trên đường* đến vị trí thứ nhất:

```
N34 T0200 M42
N35 G96 S190 M03
N36 G00 G41 X12.0 Z0 T0202 M08
N37 G01 X1.6 F0.008
...
```

Ngoài ra bạn cần lưu ý, không cần G00 đối với block chứa sự phân độ dao có mục nhập bù mòn dao là zero (00). Ưu điểm của lập trình lệnh bù dao đồng thời với chuyển động là loại bỏ chuyển động "nhảy"; đồng thời không xảy ra hành trình quá mức, ngay cả khi giá trị bù quá cao. Giá trị bù mòn sẽ chỉ *kéo dài* hoặc *rút ngắn* chuyển động nhanh được lập trình, tùy theo giá trị bù thực đã lưu.

Nói chung, chỉ số đăng ký bù mòn dao được nhập *trước* hoặc *trong khi* chuyển động nhanh.

Thay đổi chế độ bù

Hầu hết các chương trình tiện đều yêu cầu một giá trị bù cho từng dao. Tuy nhiên, trong một số trường hợp chương trình có thể sử dụng hai hoặc nhiều giá trị bù gán cho một dao, mỗi lần kích hoạt một giá trị. Giá trị bù hiện hành có thể đổi sang giá trị khác cho cùng một dao để tăng tính linh hoạt. Điều này rất hữu ích, chủ yếu trong trường hợp khi các đường kính hoặc các chiều dài vai (bậc) phải được gia công theo dung sai chính xác. Giá trị bù mới phải được lập trình *nhưng không* xóa giá trị bù trước đó. Đây là phương pháp thông dụng để thay đổi giá trị bù. Lý do rất đơn giản- bạn hãy nhớ sự thay đổi giá trị bù bất kỳ chỉ phục vụ cho mục đích trong khi cắt gọt thực sự. Xóa bù có thể sẽ không an toàn nếu được lập trình trong chuyển động cắt gọt. Điều này là rất quan trọng - *và là kỹ thuật lập trình tương đối mới*, sẽ được làm sáng tỏ bằng vài ví dụ chi tiết.

LẬP TRÌNH NHIỀU GIÁ TRỊ BÙ

Hầu hết các công việc gia công trên máy tiện CNC đòi hỏi độ chính xác rất cao. Độ chính xác cao yêu cầu các khoảng dung sai được chuyên biệt trên bản vẽ kỹ thuật và các khoảng này có thể khác nhau. Do một giá trị bù cho một dao thường không đủ duy trì các dung sai đó, cần có hai hoặc nhiều giá trị bù cho một dao.

Ba ví dụ dưới đây được thiết kế để giới thiệu đầy đủ kỹ thuật lập trình với nhiều giá trị bù mòn dao sử dụng cùng một bản vẽ kỹ thuật.

Vấn đề rất đơn giản – lập trình và gia công ba đường kính theo bản vẽ đồng thời duy trì đúng các khoảng dung sai. Nguyên tắc đầu tiên – chương trình sẽ *không sử dụng* dung sai khoảng giữa các giá trị X hoặc Z. Đây là điều bất tiện do có thể gây ra khó khăn khi cần thay đổi chương trình, nếu kỹ sư thiết kế thay đổi các khoảng dung sai.

Dung sai trên bản vẽ gồm:

- Dung sai trên đường kính.
- Dung sai trên các vai (bậc).
- Dung sai trên các đường kính và các vai.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Phương pháp chung

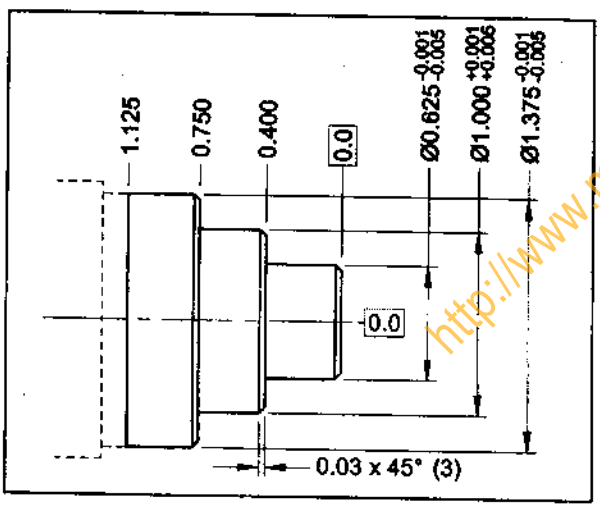
Các dung sai trong cả ba ví dụ đều chỉ nhằm mục đích minh họa, trong thực tiễn chúng nhỏ hơn nhiều. Mọi dung sai vạt góc đều là ±0.010, và các dung sai không ghi trên bản vẽ đều là ±0.005. Điều này cho phép tập trung vào vấn đề chính. Vật liệu là thanh nhôm Ø1.5 inch, sử dụng ba dao để gia công:

T01	Vạt mặt và gia công thô
T03	Gia công tinh các mặt trụ
T05	Dao cắt đứt, dày 0.125

Kỹ năng của nhà lập trình xác định kết quả cuối cùng – số lượng giá trị bù cần nhập đúng các vị trí trong chương trình, người vận hành CNC phải lưu các giá trị đúng cho từng chế độ bù. Trong mọi trường hợp, mục đích chính cần hướng đến là khoảng giữa dung sai trong gia công, không phải trong chương trình.

Dung sai đường kính

Hình 33.3 minh họa bản vẽ chi tiết với các dung sai chỉ khác nhau trên *đường kính*.



Hình 33.3. *Sử dụng nhiều giá trị bù – ví dụ đối với đường kính – O3301*

Giải pháp lập trình là sử dụng *hai* giá trị bù khi gia công tinh, T0313 và T0314. Trong bộ điều khiển, các giá trị bù chính xác cần được xác lập trước khi gia công – các giá trị lý tưởng đối với khoảng giữa dung sai như sau :

```
13   X-0.003  Z0.000
14   X+0.003  Z0.000
```

Chú ý, bù trục Z (kiểm soát các bậc) phải *như nhau đối với cả hai chế độ bù.*

Chương trình hoàn chỉnh – O3301 – như sau:

```
O3301
(1.5 ALUMINUM BAR - EXTEND 1.5 FROM JAWS)
(T01 - FACE AND ROUGH TURN)
N1 G20
N2 G50 S3000 T0100
N3 G96 S500 M03
N4 G00 G41 X1.7 Z0 T0101 M08
N5 G01 X-0.07 F0.005
N6 Z0.1
N7 G00 G42 X1.55
N8 G71 P9 Q16 U0.04 W0.004 D1000 F0.01
N9 G00 X0.365
N10 G01 X0.625 Z-0.03 F0.003
N11 Z-0.4
N12 X1.0 C-0.03 (K-0.03)
N13 Z-0.75
N14 X1.375 C-0.03 (K-0.03)
N15 Z-1.255
N16 U0.2
N17 G00 G40 X5.0 Z5.0 T0100
N18 M01

(T03 - FINISH TURN)
N19 G50 S3500 T0300
(-- OFFSET 00 AT THE START OF THE TOOL -----)
N20 G96 S750 M03
N21 G00 G42 X1.7 Z0.1 T0313 M08
(-- OFFSET 13 FOR THE 0.625 DIAMETER ------)
N22 X0.365
N23 G01 X0.625 Z-0.03 F0.002
N24 Z-0.4
N25 X1.0 C-0.03 (K-0.03) T0314
(-- OFFSET 14 FOR THE 1.0 DIAMETER --------)
N26 Z-0.75
N27 X1.375 C-0.03 (K-0.03) T0313
(-- OFFSET 13 FOR THE 1.375 DIAMETER ------)
N28 Z-1.255
N29 U0.2
N30 G00 G40 X5.0 Z5.0 T0300
(-- OFFSET 00 AT THE END OF TOOL ----------)
N31 M01

(T05 - 0.125 WIDE PART-OFF)
N32 T0500
N33 G97 S2000 M03
N34 G00 X1.7 Z-1.255 T0505 M08
N35 G01 X1.2 F0.002
N36 G00 X1.45
N37 Z-1.1825
N38 G01 X1.315 Z-1.25 F0.001
N39 X-0.02 F0.0015
N40 G00 X5.0
N41 Z5.0 T0500 M09
N42 M30
%
```

Đây là chương trình sử dụng cả ba dao cắt. Do T01 và T05 không thay đổi trong các ví dụ kế tiếp, từ đây trở đi sẽ chỉ nêu dao T03.

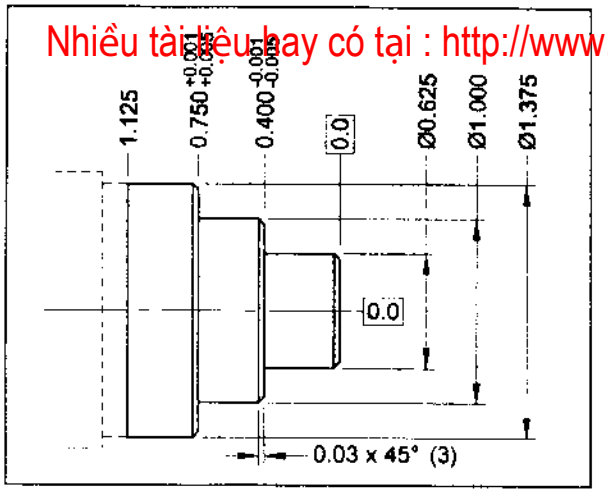
Dung sai trên bậc trụ

Hình 33.4 minh họa bản vẽ chi tiết với các dung sai khác nhau trên *bậc trụ*, giải pháp lập trình là sử dụng *hai* giá trị bù khi gia công tinh, T0313 và T0314. Trong bộ điều khiển, các giá trị chính xác cần được xác lập trước khi gia công – các giá trị bù lý tưởng đối với khoảng giữa dung sai như sau:

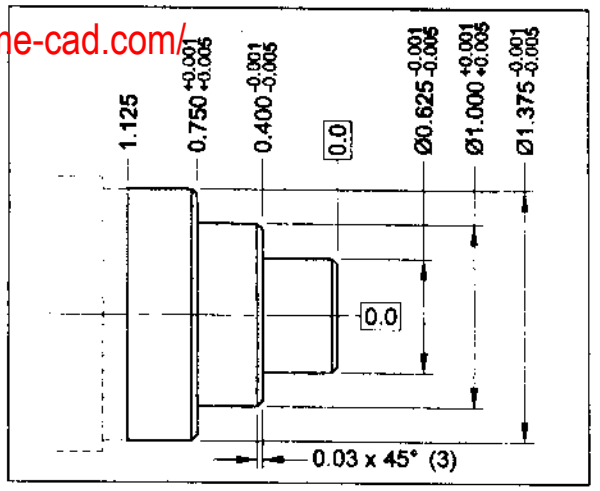
```
13   X0.0000  Z+0.0030
14   X0.0000  Z-0.0030
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 33.4. Sử dụng nhiều giá trị bù – ví dụ gia công các bậc trụ – O3302



Hình 33.5. Sử dụng nhiều giá trị bù – Ví dụ gia công các bậc và đường kính – O3303.

Chú ý, trong trường hợp này, bù X (kiểm soát kích cỡ đường kính) phải như nhau đối với cả hai chế độ bù. Dưới đây là dao T3 đối với chương trình O3302:

```
O3302
...
(T03 - FINISH TURN)
N19 G50 S3500 T0300
(-- OFFSET 00 AT THE START OF TOOL --------)
N20 G96 S750 M03
N21 G00 G42 X1.7 Z0.1 T0313 M08
(-- OFFSET 13 FOR THE 0.4 SHOULDER --------)
N22 X0.365
N23 G01 X0.625 Z-0.03 F0.002
N24 Z-0.4
N25 X1.0 C-0.03 (K-0.03)
N26 Z-0.75 T0314
(-- OFFSET 14 FOR THE 0.75 SHOULDER -------)
N27 X1.375 C-0.03 (K-0.03)
N28 Z-1.255
N29 U0.2
N30 G00 G40 X5.0 Z5.0 T0300
(-- OFFSET 00 AT THE END OF TOOL ----------)
N31 M01
...
```

Dung sai đường kính và bậc trụ

Hình 33.5 minh họa bản vẽ chi tiết với dung sai trên cả *đường kính* và *bậc trụ*.

Giải pháp lập trình là dùng *bốn* giá trị bù khi gia công tinh, ví dụ T0313, T0314, T0315, và T0316. Trong bộ điều khiển, các giá trị của chúng phải được xác lập trước khi gia công, giá trị lý tưởng đối với khoảng giữa dung sai như sau:

```
13  X-0.0030  Z+0.0030
14  X+0.0030  Z+0.0030
15  X+0.0030  Z-0.0030
16  X-0.0030  Z-0.0030
```

Đây là phiên bản mạnh nhất. Điều quan trọng không chỉ là *vị trí chính xác* của các chế độ bù trong chương trình, mà còn là các giá trị nhập của chúng.

Chú ý, bốn giá trị bù trục X (kiểm soát kích cỡ đường kính) ràng buộc chặt chẽ với bốn giá trị bù trục Z (kiểm soát chiều dài bậc trụ). Dưới đây là dao T03 trong chương trình O3303:

```
O3303
...
(T03 - FINISH TURN)
N19 G50 S3500 T0300
(-- OFFSET 00 AT THE START OF TOOL --------)
N20 G96 S750 M03
N21 G00 G42 X1.7 Z0.1 T0313 M08-
(-- OFFSET 13 FROM Z OVER TO Z UNDER ONLY --)
N22 X0.365
N23 G01 X0.625 Z-0.03 F0.002
N24 Z-0.4
N25 X1.0 C-0.03 (K-0.03) T0314
(-- OFFSET 14 FROM X UNDER TO X OVER ONLY --)
N26 Z-0.75 T0315
(-- OFFSET 15 FROM Z UNDER TO Z OVER ONLY --)
N27 X1.375 C-0.03 (K-0.03) T0316
(-- OFFSET 16 FROM X OVER TO X UMDER ONLY --)
N28 Z-1.255
N29 U0.2
N30 G00 G40 X5.0 Z5.0 T0300
(-- OFFSET 00 AT THE END OF TOOL ----------)
N31 M01
...
```

Người vận hành phải luôn chú ý đến sự hiện hữu của nhiều chế độ bù trong chương trình và lý do nhà lập trình sử dụng chúng. Các xác lập ban đầu và mọi thay đổi của chúng trong khi gia công đều rất quan trọng. Như có thể thấy trong hai ví dụ, chương trình O3301 và O3302, một nhóm giá trị bù phải luôn luôn không thay đổi (bù X hoặc bù Z). Ví dụ, trong chương trình O3301, các giá trị bù 03 và 13 kiểm soát đường kính. Điều đó có nghĩa là giá trị bù Z phải *luôn luôn* không đổi. Điều này còn có nghĩa là, nếu cần dịch chuyển các vai (bậc

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

trụ) sang trái 0.002, *mọi bậc trụ đều phải dịch chuyển theo cùng giá trị*:

```
13   Y-0.0030  Z-0.0020
14   Y+0.0030  Z-0.0020
```

Nếu không tuân thủ điều đó sẽ dẫn đến các kích thước không chính xác.

XÁC LẬP BÙ

Màn hình OFFSET (bù) được chọn bằng cách nhấn phím trên bảng điều khiển sẽ hiển thị bù hình học và bù mòn dao. Chúng là như nhau, trừ tiêu đề ở phía trên màn hình. Hiển thị sẽ tương tự hình dưới đây (chưa xác lập giá trị bù):

OFFSET (GEOMETRY)				
NO.	X AXIS	Z AXIS	RADIUS	TIP
01	0.0000	0.0000	0.0000	0
02	0.0000	0.0000	0.0000	0
03	0.0000	0.0000	0.0000	0
...	...	...	...	...

Trục X và trục Z thường chỉ ghi là X và Z, *Bán kính* là R và *Đỉnh dao* là T.

NO. là chỉ số bù, cặp thứ nhất của địa chỉ T – bù *Hình học*, hoặc cặp thứ hai, bù *Mòn* dao. Trục X và Z là các cột nhập giá trị bù tương ứng, các cột *Radius* (bán kính) và *Tip* (đỉnh dao) chỉ được sử dụng khi bù bán kính mũi dao được lập trình. Trong trường hợp đó, Radius sẽ là bán kính mũi dao và Tip là số tùy ý, do Fanuc xác định, chuyên biệt định hướng đỉnh dao. Điều này đã được trình bày trong Chương 29.

CÁC HÀM KHOẢNG TỐC ĐỘ

Một số máy tiện CNC được thiết kế để làm việc ở vài khoảng tốc độ (hộp số). Tính năng này cho phép nhà lập trình phối hợp tốc độ trục chính với các yêu cầu công suất máy. Nguyên tắc chung là yêu cầu tốc độ trục chính càng cao, định mức công suất cực đại càng thấp, và ngược lại. Các khoảng tốc độ trục chính và định mức công suất ứng với từng khoảng do nhà chế tạo máy xác định, và không thể thay đổi.

Có thể có hai, ba, hoặc bốn khoảng tốc độ, tùy theo kích cỡ máy tiện CNC. Các máy nhỏ, hoặc máy được thiết kế với tốc độ trục chính rất cao, có thể hoàn toàn không có khoảng tốc độ lập trình. Các máy lớn có thể có đến bốn khoảng tốc độ, tốc độ trục chính cực đại thường không cao. Phổ biến nhất là hai khoảng tốc độ.

Các hàm khoảng tốc độ gồm M41, M42, M43, M44, và định nghĩa liên quan đến sự khả dụng của khoảng tốc độ như sau:

Khoảng	Số khoảng khả dụng			
	1	2	3	4
Thấp	-	M41	M41	M41
Trung bình thấp	-	-	-	M42
Trung bình	-	-	M42	-
Trung bình cao	-	-	-	M43
Cao	-	M42	M43	M44

Sau khi chọn khoảng tốc độ, sẽ giới hạn tốc độ trục chính trong khoảng đó. Nếu khoảng chính xác tốc độ trục chính là quan trọng, cần tìm các tốc độ trục chính khả dụng trong từng khoảng. Tốc độ trục chính thấp nhất có thể là 20-30 r/min (vòng/phút), thường có sự chồng chập đối với các tốc độ trục chính trong hai khoảng liên tiếp. Ví dụ, số 1 có khoảng tốc độ 20-1400 r/min, số 2 có khoảng tốc độ 150-2500 r/min. Khi sử dụng tốc độ trục chính khả dụng trong các khoảng đó, lựa chọn khoảng (hộp số) nào là không quan trọng, khoảng thấp thường có công suất cao hơn.

Dưới đây là ví dụ thực

Khoảng tốc độ thấp: 20 – 1075 r/min (M41)
Khoảng tốc độ cao: 70 – 3600 r/min (M42)

VẠT GÓC TỰ ĐỘNG

Trong doa và tiện CNC, có các trường hợp gia công bậc trụ và đường kính (hoặc đường kính và bậc trụ) đòi hỏi phải cắt góc. Sự cắt góc sắc là yêu cầu phổ biến khi gia công giữa bậc trụ và mặt trụ (hoặc côn). Nhiều bản vẽ kỹ thuật yêu cầu mọi góc sắc đều phải vạt góc, thường không đưa ra kích cỡ cụ thể. Điều đó do nhà lập trình quyết định, thường trong khoảng 0.005 – 0.020 inch (0.125 – 0.500 mm). Sự cắt góc có thể là *vạt góc 45^0 hoặc bo tròn* với bán kính nhỏ. Nếu kích cỡ cắt góc được chuyên biệt, nhà lập trình phải sử dụng giá trị đó. Sự vạt góc có ba lý do thực tiễn:

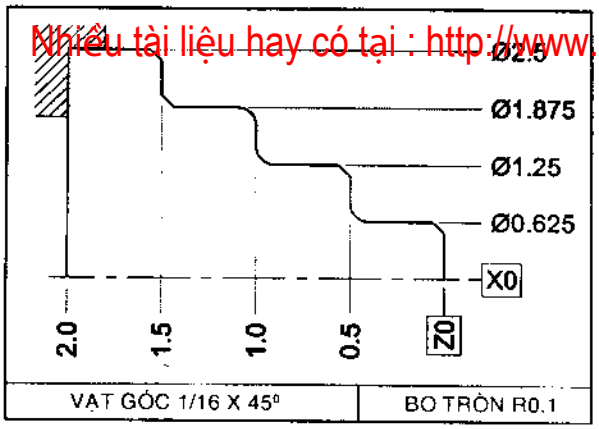
- Tính năng:
  ...đối với độ bền, dễ lắp ráp, và các khoảng hở
- An toàn
  ...các góc sắc thường rất nguy hiểm
- Dáng vẻ
  ...hình thức chi tiết hoàn tất.

Trong gia công tiện, nhiều chế độ cắt góc áp dụng cắt giữa bậc và đường kính kế cận (theo mặt vuông góc với một trục). Tính toán điểm bắt đầu và kết thúc không khó nhưng có thể hơi tốn thời gian, chẳng hạn tiện trục với nhiều đường kính.

Bản vẽ trên Hình 33.6 minh họa chi tiết đơn giản có nhiều góc (bậc) có thể áp dụng tính

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 33.6. Ví dụ về cắt góc tự động (vạt góc và bo tròn góc)

năng lập trình cắt góc tự động (trong bản vẽ có thể *không* ghi *tất cả* các góc).

Bạn hãy so sánh hai phương pháp, để hiểu rõ hơn các khác biệt được áp dụng trong lập trình. Nếu nhà lập trình *không sử dụng* tính năng cắt góc tự động, từng điểm thay đổi biên dạng phải được tính toán bằng tay và kết quả sẽ là chương trình O3304:

```
O3304 (TÍNH TOÁN VẠT GÓC BẰNG TAY)
...
N51 T0100
N52 G96 S450 M03
N53 G00 G42 X0.3 Z0.1 T0101 M08
N54 G01 X0.625 Z-0.0625 F0.003
N55 Z-0.4
N56 G02 X0.825 Z-0.5 R0.1
N57 G01 X1.125
N58 X1.25 Z-0.5625
N59 Z-0.9
N60 G02 X1.45 Z-1.0 R0.1
N61 G01 X1.675
N62 G03 X1.875 Z-1.1 R0.1
N63 G01 Z-1.4375
N64 X2.0 Z-1.5
N65 X2.375
N66 X2.55 Z-1.5875
N67 U0.2
N68 G00 G40 X10.0 Z5.0 T0100
N69 M01
```

Chỉ lập trình biên dạng hoàn tất (không vạt mặt), bắt đầu tại khoảng hở được chọn Z0.1, với đường kính X được tính toán ở X0.3. Từng điểm thay đổi biên dạng (đường kính) đều được tính toán cẩn thận. Tại cuối biên dạng, vạt góc cuối cùng được hoàn tất tại khoảng hở 0.025 phía trên đường kính lớn nhất, X2.55, và trục Z tính toán ở Z-1.5875.

Tính toán bằng tay, khả năng xảy ra sai số tương đối lớn. Ví dụ, sai số rất phổ biến trong kiểu lập trình này là giá trị đích của trục X. Khi tiện, thường dễ quên nhân đôi giá trị vạt góc hoặc bán kính bo tròn (hoặc chia đôi khi doa). Kết quả là block N56 có thể như sau:

```
N56 G02 X0.725 Z-0.5 R0.1 (LỖI GIÁ TRỊ X)
```

thay vì block *đúng*

```
N56 G02 X0.825 Z-0.5 R0.1 (X LÀ ĐÚNG)
```

Vậy cần thực hiện điều gì trong chương trình để cải thiện sự cắt góc tự động?

Hệ điều khiển Fanuc có hai phương pháp lập trình liên quan với sự cắt góc tự động trên máy tiện :

- ❑ Phương pháp vạt góc ...vạt góc 45^0
- ❑ Phương pháp bo tròn ...bo tròn góc 90^0

Hai phương pháp này có cơ chế hoạt động rất giống nhau, bạn cần chú ý các nguyên tắc của chúng.

Vạt góc 45 độ

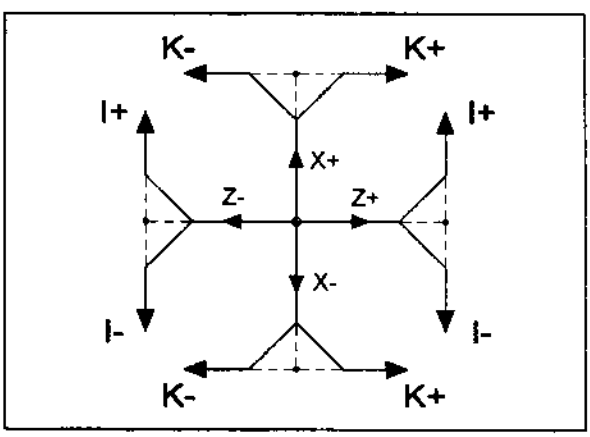
Vạt góc tự động luôn luôn thực thi trong chế độ G01, có hai vector I và K cho mục đích này, hoặc vector C trên một số bộ điều khiển.

Để tạo ra sự vạt góc tự động, các vector I và K chuyên biệt *chiều và lượng cắt gọt cần thiết cho vạt góc:*

Vector I
được dùng để tạo ra sự vạt góc bắt đầu từ trục X, theo chiều X+Z-, X-Z-, X+Z+ hoặc X-Z+

Vector K
được dùng để tạo ra sự vạt góc bắt đầu từ trục Z, theo chiều Z-X+, Z-X-, Z+X+, hoặc Z+X-

Hình 33.7 minh họa định nghĩa các vector I và K.



Hình 33.7. Vector I và K đối với sự vạt góc tự động

Khi bộ điều khiển gặp block chứa vector vạt góc I hoặc K, sẽ tự động *rút ngắn* chiều dài quỹ đạo dao lập trình đang có hiệu lực theo giá trị của I hoặc K được chuyên biệt trong chương trình. Nếu chưa hiểu rõ cách lập trình I hoặc K, bạn hãy xem Hình 33.7 và áp dụng các nguyên tắc dưới đây :

Vector I biểu thị *lượng vạt góc và chiều*

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

chuyển động khi chuyển động dao theo thứ tự *Đường kính – Vạt góc – Bậc trụ* nghĩa là cắt theo trục Z trước khi vạt góc. Độ lệch vạt góc chỉ có thể từ trục Z hướng đến trục X, với vector I được lập trình:

```
G01 Z-1.75 I0.125          (CẮT THEO TRỤC Z)
X4.0                       (TIẾP TỤC THEO TRỤC X)
```

Vector K biểu thị *lượng vạt góc và chiều chuyển động* khi chuyển động dao theo thứ tự *Bậc trục – Vạt góc – Đường kính* nghĩa là cắt theo trục X trước khi vạt góc. Độ lệch vạt góc chỉ có thể từ trục X hướng đến trục Z, khi vector K được lập trình:

```
G01 X2.0 K-0.125           (CẮT THEO TRỤC X)
Z-3.0                      (TIẾP TỤC THEO TRỤC Z)
```

Trong các trường hợp đó, dấu của vector I hoặc K xác định chiều vạt góc trong hệ tọa độ :

- Giá trị dương của I và K biểu thị chiều vạt góc theo chiều dương của trục *không* chuyên biệt trong block vạt góc.
- Giá trị âm của I hoặc K biểu thị chiều vạt góc theo chiều âm của trục *không* chuyên biệt trong block vạt góc.

Các giá trị của I và K luôn luôn là giá trị đơn (giá trị bán kính thay vì đường kính).

Nhiều bộ điều khiển hiện đại sử dụng các vector C+ và C – thay cho các vector I+, I-, K+, K- (Hình 33.8). Đây là phương pháp lập trình đơn giản hơn và các ứng dụng tương tự bán kính R bo tròn góc. Không có sự phân biệt giữa các trục chọn vector, chỉ cần xác định chiều.

- Vector C được sử dụng.
  - ... để tạo ra sự vạt góc bắt đầu tự trục X, theo chiều X+Z-, X-Z-, X+Z+, hoặc X-Z+,

  hoặc
  - ... để tạo ra sự vạt góc bắt đầu từ trục Z, theo chiều Z-X+, Z-X-, Z+X+, hoặc Z+X-

Nếu bộ điều khiển cho phép các vector C+ hoặc C-, sự lập trình còn dễ dàng hơn khi biết chiều chuyển động. Hai ví dụ nêu trên sẽ như sau (dùng vector C thay cho I và K):

```
G01 Z-1.75 C0.125          (CẮT THEO TRỤC Z)
X4.0                       (TIẾP TỤC THEO TRỤC X)
G01 X2.0 C-0.125           (CẮT THEO TRỤC X)
Z-3.0                      (TIẾP TỤC THEO TRỤC Z)
```

Cũng như trường hợp với các vector I và K, vector C cùng chuyên biệt giá trị đơn *theo một phía*, bán kính thay vì đường kính.

Bo tròn góc theo 90 độ

Bán kính bo tròn giữa bậc trụ và đường kính (hoặc ngược lại) được lập trình theo cách thức tương tự sự vạt góc 45^0. *Điều này cũng chỉ xảy ra trong chế độ G01*, chỉ cần một vector R. Đối với bo tròn góc, vector R *chuyên biệt chiều và lượng cắt gọt theo bán kính* (Hình 33.9)

- Vector R được sử dụng
  - ...để tạo ra sự bo tròn góc bắt đầu từ trục X, theo chiều X+Z-, X+Z+, X-Z-, X-Z+

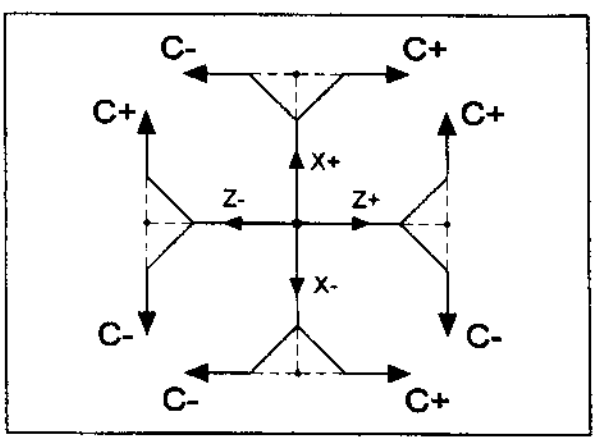
  hoặc
  - ...để tạo ra sự bo tròn góc bắt đầu từ trục Z theo chiều Z-X+, Z-X-, Z+X+, Z+X-

Khi hệ điều khiển gặp block chứa vector R, sẽ tự động *rút ngắn* chiều dài quỹ đạo dao đang hoạt động theo giá trị vector R được xác định trong chương trình. Để hiểu rõ hơn về lập trình vector R bạn hãy xem Hình 33.9 và áp dụng nguyên tắc dưới đây :

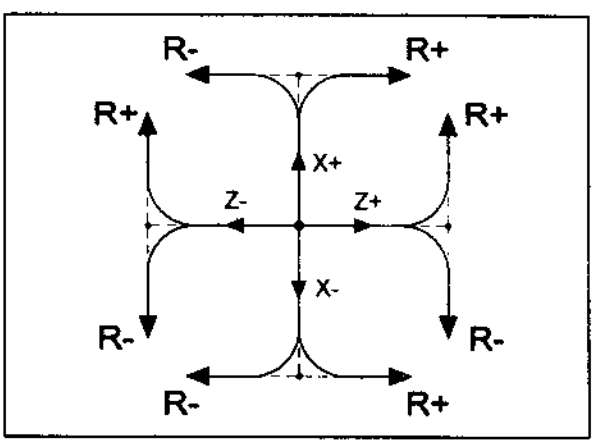
Vector R biểu thị *giá trị bán kính và chiều chuyển động* khi cắt gọt theo thứ tự *Bậc trụ – Bán kính – Đường kính*, nghĩa là cắt gọt theo trục X trước bán kính đó. Vector này còn được dùng khi *giá trị bán kính và chiều chuyển động* theo thứ tự ngược lại, *Đường kính – Bán kính – Bậc trụ*, nghĩa là cắt gọt theo trục Z trước bán kính đó.

Độ lệch bán kính có thể từ trục X hướng đến trục Z, khi vector R được lập trình:

```
G01 X2.0 R-0.125           (CẮT THEO TRỤC X)
Z-3.0                      (TIẾP TỤC THEO TRỤC Z)
```



Hình 33.8. Vector C đối với vạt góc tự động



Hình 33.9. Vector R đối với bo tròn góc tự động

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Độ lệch bán kính cũng có thể từ trục Z hướng đến trục X, khi vector R được lập trình:

```
G01 Z-1.75 R0.125          (CẮT THEO TRỤC Z)
X4.0                       (TIẾP TỤC THEO TRỤC X)
```

Trong các trường hợp đó, dấu của vecotr R xác định chiều cắt gọt bán kính trong hệ tọa độ:

- Giá trị dương của vector R biểu thị bán kính theo chiều *dương* của trục *không* được chuyên biệt trong block bán kính.
- Giá trị âm của vector R biểu thị bán kính theo chiều *âm* của trục *không* được chuyên biệt trong block bán kính.

Điều kiện lập trình

Sự cắt góc một cách tự động làm cho sự lập trình trên các máy tiện CNC hiện đại trở nên dễ dàng hơn, chỉ cần sử dụng các kích thước bản vẽ, không cần thêm các tính toán. Bất kể chương trình chứa vector I, K, hoặc C để vạt góc, hoặc R để bo tròn góc, các điều kiện cơ bản và các nguyên tắc tổng quát hầu như tương tự nhau:

- Vạt góc hoặc bo tròn góc phải hoàn toàn trong một góc phần tư (90^0)
- Vạt góc phải theo góc 45^0 và bo tròn theo 90^0 giữa bậc và đường kính, hoặc đường kính và bậc
- Các giá trị vector vạt góc I, K, C và vector bán kính R, luôn luôn là giá trị đơn, nghĩa là giá trị *theo một phía*, không phải là giá trị đường kính.
- Chiều cắt trước khi bo tròn góc phải vuông góc với chiều cắt sau khi bo tròn, chỉ theo một trục.
- Chiều cắt trước khi vạt góc hoặc bo tròn phải tiếp tục chỉ theo một trục, và phải có chiều dài tương đương ít nhất là chiều dài vạt góc hoặc giá trị bán kính – chiều cắt không thể đảo ngược.
- Cả vạt góc và bo tròn đều trong G01 (chế độ nội suy tuyến tính)
- Khi viết chương trình CNC, chỉ cần biết giao điểm – *điểm cắt giữa X và Z*, trên bản vẽ. Đây là điểm giữa bậc trụ và đường kính chứa vạt góc hoặc bo tròn.

Các nguyên tắc này áp dụng cho cả tiện và doa trên máy tiện CNC. Bạn cần nghiên cứu chúng một cách cẩn thận để tránh các vấn đề có thể xảy ra trong thực tiễn gia công.

Ví dụ lập trình

Chương trình O3305 sử dụng cả các vector vạt góc và bo tròn góc trong một ví dụ. Bản vẽ dùng cho ví dụ này, là bản vẽ cũ trên Hình 33.6.

Để hiểu rõ các khác biệt giữa hai phương pháp lập trình (cả hai đều đúng về kỹ thuật), bạn hãy so sánh chương trình O3305 với O3304. Các vector I và K được dùng cho vạt góc, khó hơn so với vector C:

```
O3305 (AUTOMATIC CORNER BREAKS USED)
N51 T0100
N52 G96 S450 M03
N53 G00 G42 X0.3 Z0.1 T0101 M08
N54 G01 X0.625 Z-0.0625 F0.003
N55 Z-0.5 R0.1
N56 X1.25 K-0.0625
N57 Z-1.0 R0.1
N58 X1.875 R-0.1
N59 Z-1.5 I0.0625
N60 X2.375
N61 X2.55 Z-1.5875
N62 U0.2
N63 G00 G40 X10.0 Z5.0 T0100
N64 M01
```

Chương trình này hơi ngắn hơn, do giảm bớt năm block. Bạn có thể tự hỏi, các lệnh G02 và G03 ở đâu? Tại sao không có tính toán các điểm thay đổi biên dạng (đường kính) và tâm điểm?

Ngoại trừ phần bắt đầu và kết thúc biên dạng, kiểu lập trình này cho phép triển khai chương trình rất nhanh, dễ dàng thực hiện các thay đổi trong khi gia công, nếu cần thiết. Nếu có thay đổi vạt góc hoặc bo tròn góc trên bản vẽ, chỉ cần thay đổi *một giá trị* trong chương trình, mà không phải tính toán lại, nhưng phải tuân thủ các nguyên tắc và điều kiện đã nên ở phần trên. Ưu điểm chính của gia công biên dạng tự động là dễ dàng thay đổi và không cần các tính toán bằng tay.

GIA CÔNG THÔ VÀ TINH

Hầu hết các nguyên công cắt gọt trên máy tiện CNC đều được thực hiện bằng cách sử dụng các chu kỳ, sẽ được trình bày chi tiết trong Chương 34. Các chu kỳ này đòi hỏi nhập dữ liệu dựa trên kiến thức về công nghệ, chẳng hạn chiều sâu cắt, lượng dư cho phép, tốc độ cắt, ...

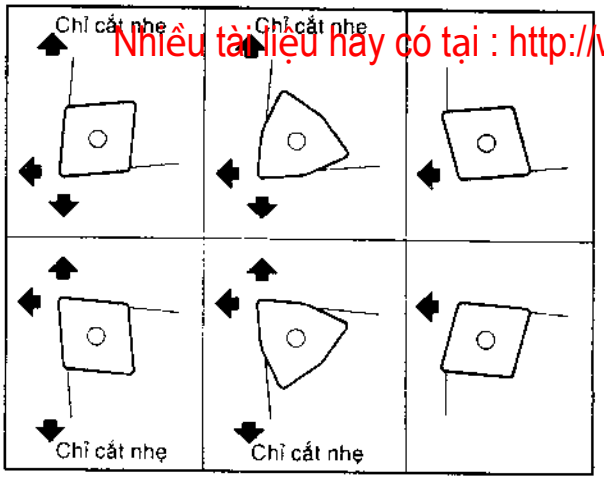
Các hình dạng thô và tinh thường đòi hỏi tính toán bằng tay, sử dụng đại số và lượng giác. Các tính toán này cần được thực hiện trên giấy riêng, thay vì trên bản vẽ. Theo đó, công việc sẽ được sắp xếp tốt hơn. Ngoài ra, sau này nếu có thay đổi, chẳng hạn thay đổi thiết kế kỹ thuật, cũng sẽ dễ theo dõi hơn.

Gia công thô

Hầu hết công việc trên máy tiện là cắt bỏ lượng dư lớn để tạo ra chi tiết hầu như hoàn tất. Dạng cắt gọt này được gọi là gia công *thô*, tiện thô hoặc doa thô. Là nguyên công cắt gọt, gia công thô không tạo ra chi tiết có độ chính xác cao, đó không phải là mục đích của gia công thô. Mục đích chính là cắt gọt lượng dư một cách hiệu quả, nhanh chóng, với tuổi bền dụng cụ cắt ở mức cao nhất, và để lại lượng dư thích

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 33.10. *Định hướng dao và chiều cắt. Hàng trên là dao tiện ngoài, hàng dưới là dao tiện (doa) trong.*

hợp cho gia công tinh. Dụng cụ cắt dùng trong gia công thô phải đủ bền, thường có bán kính mũi tương đối lớn. Các dao này phải có khả năng bảo đảm chiều sâu cắt lớn và tốc độ cắt cao. Các dao hình thoi thích hợp cho gia công tinh là các mảnh chắp 80^0 (đến 2+2 góc cắt), và mảnh tam giác (đến 3+3 góc cắt). 2+2 hoặc 3+3 nghĩa là *2 hoặc 3 lưỡi cắt trên từng mặt* của mảnh chắp. Không phải mọi mảnh chắp đều có thể sử dụng cả hai mặt. Hình 33.10 minh họa một số mảnh chắp và định hướng của chúng khi doa và tiện thô.

Mặc dầu một số dụng cụ cắt có thể lập trình theo nhiều hướng, nhưng có vài hướng hoàn toàn không nên dùng, hoặc chỉ dùng để cắt nhẹ hoặc hơi nhẹ.

Trong thực tiễn, bạn nên tuân theo một nguyên tắc gia công cơ bản- có giá trị đối với tất cả các máy công cụ:

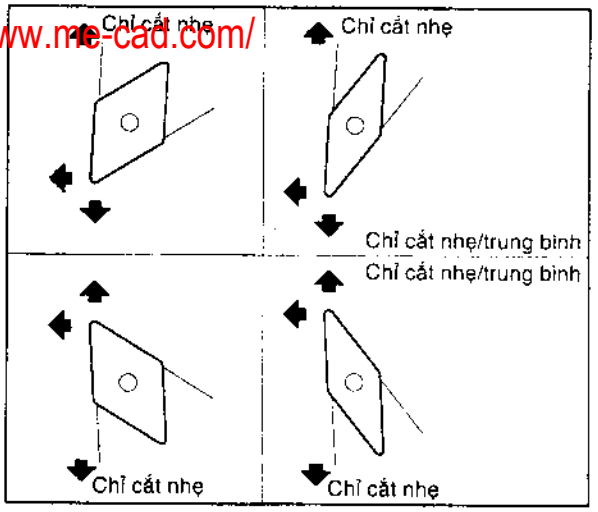
Luôn luôn thực hiện các cắt gọt nặng trước khi cắt gọt nhẹ

Nguyên tắc cơ bản này có nghĩa là *toàn bộ* cắt gọt thô phải được thực hiện trước khi lập trình gia công tinh. Ví dụ, yêu cầu là gia công thô và gia công tinh cả đường kính trong và đường kính ngoài. Gia công đường kính nào trước, điều đó không quan trọng, miễn là gia công tinh phải thực hiện sau khi gia công thô.

Sự mòn dao có thể giảm đến mức tối thiểu nếu đủ chiều sâu cắt và bán kính cắt ở dưới "lớp cắt gọt", thường trong khi cắt lần thứ nhất. Chất làm nguội được dùng cho hầu hết các vật liệu, và phải được cung cấp *trước khi* lưỡi cắt tiếp xúc với bề mặt gia công.

Gia công tinh

Gia công tinh thực hiện chuyển động cắt



Hình 33.11. *Định hướng dao và chiều cắt gọt khi gia công tinh với dao tiện thông dụng. Hàng trên là dao gia công ngoài, hàng dưới là gia công trong.*

gọt hoàn tất sau khi hầu hết lượng dư đã được cắt bỏ (gia công thô), chỉ để lại một lượng nhỏ cho gia công tinh. Dụng cụ cắt có thể có bán kính mũi dao nhỏ hơn, và để có độ bóng bề mặt tốt hơn, cần có tốc độ trục chính cao hơn với tốc độ cắt thấp hơn.

Nhiều loại dụng cụ cắt có thể được dùng cho gia công tinh, phổ biến nhất là hai mảnh chắp hình thoi với góc 55^0 và 35^0 (Hình 33.11).

Chú ý, có vài chiều cắt chỉ nên dùng cho cắt nhẹ hoặc hơi nhẹ.

Phôi và lượng dư gia công

Vật liệu được gia công thường gọi là *phôi*. Khi dao cắt gọt phôi theo hình dạng mong muốn, mỗi lần chỉ cắt được một lượng xác định. Hình dạng mảnh chắp, hướng đến chi tiết và chiều cắt gọt, kích cỡ và chiều dày mảnh chắp, tất cả đều tác động đến lượng cắt gọt. Điều này đặc biệt quan trọng trong gia công bán tinh và gia công tinh. Lượng dư gia công xác định lượng vật liệu để lại cho gia công tinh. Nếu lượng để lại này quá lớn hoặc quá nhỏ, độ chính xác gia công và độ bóng bề mặt có thể không đạt yêu cầu. Ngoài ra, cần xem xét cẩn thận không chỉ lượng dư chung trên chi tiết, mà còn *lượng dư riêng* theo trục X và trục Z.

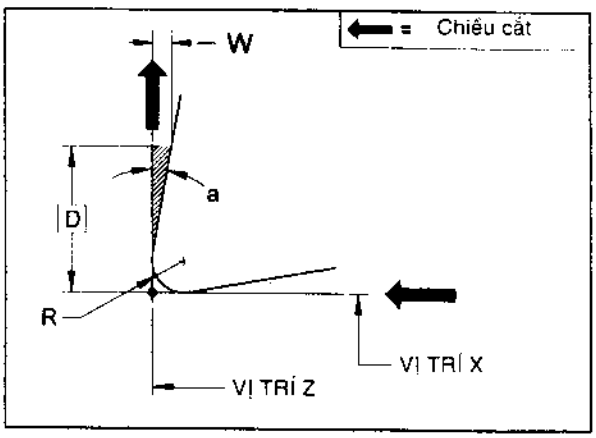
Nguyên tắc chung, trên trục X, cắt gọt đường kính, lượng dư để lại phải *tương đương* hoặc *hơi lớn hơn* bán kính mũi của dao cắt tinh kế tiếp. Ví dụ, nếu bán kính mũi dao là 0.030 inch (0.80 mm) được dùng để cắt tinh, lượng dư để gia công tinh là 0.030-0.040 (1mm). Lượng dư này là *theo một phía*, không phải trên đường kính.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lượng dư trên trục Z (ví dụ gia công mặt các bậc trục 90°) còn quan trọng hơn. Nếu cắt gọt chỉ theo chiều dương trục X (tiện ngoài), hoặc chiều âm trục Z (doa, tiện trong), với dao có góc dẫn 3-5°, không được để lại lượng dư lớn hơn .003 - .006 inch (0.080 - 0.150 mm) trên bậc (vai) vuông góc với đường tâm. Hình 33.12 minh họa tác hại của lượng dư quá cao đối với một số chiều cắt gọt và phương pháp khắc phục.



Hình 33.12. *Ảnh hưởng của lượng dư W đối với chiều sâu cắt D*

Trong minh họa này, chiều sâu cắt thực D tại mặt Z, được xác định bằng lượng dư W. Để tính D, bạn hãy dùng công thức:

$$D = tg\frac{A}{2} \times R + \frac{W}{tgA} + R$$

Trong đó:

D	= Chiều sâu cắt thực tại bề mặt (vuông góc với đường tâm)
A	= Góc dẫn của mảnh chắp
W	= Lượng dư trên bề mặt để gia công tinh
Vị trí X	= Vị trí đích đối với trục X
Vị trí Z	= Vị trí đích đối với trục Z

Minh họa này cũng áp dụng cho doa (tiện trong), khi chiều trục X ngược với chiều trên Hình 33.12. Để hiểu rõ hơn hậu quả của lượng dư quá lớn để lại trên bề mặt, bạn hãy xét ví dụ dưới đây:

➲ **Ví dụ**

Lượng dư trên bề mặt là .030, bán kính mũi dao là /I.03/ và góc dẫn dao là 3°:

W = .030, R = .031, A = 3

Các dữ liệu này đủ để tính chiều sâu cắt D, sử dụng công thức nêu trên:

D = tg3/2 × .031 + .006/tg3 + .031 = .60425

Chiều sâu cắt là .60425 inch, *quá lớn* đối với các dụng cụ cắt.

Phần trên đã đề cập, lượng dư theo trục Z chỉ trong khoảng .003 - .006 inch, bạn có thể tính lại ví dụ này với W = .006:

D = tg3/2 × .031 + .006/tg3 + .031 = .14630

Đây là chiều sâu cắt hợp lý hơn, có thể sử dụng lượng dư theo trục Z là .006 inch. Đối với gia công bề mặt theo chiều X ngược, hoặc đối với các bề mặt một chiều, có thể để lại lượng dư lớn hơn, gần với giá trị bán kính mũi dao.

LẬP TRÌNH MẶT LÕM

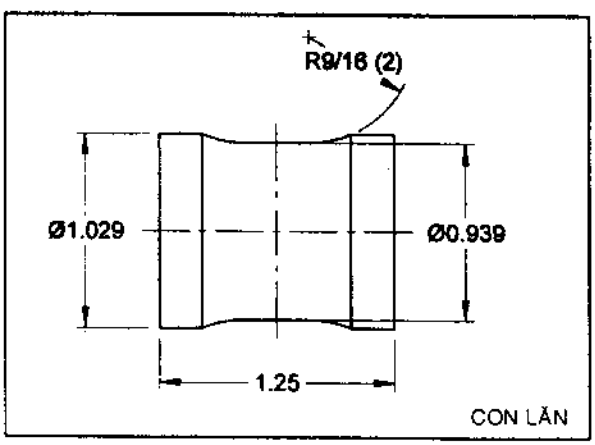
Một vấn đề rất quan trọng trong lập trình đối với máy tiện CNC là thay đổi chiều cắt gọt. Nói chung, lập trình chuyển động dao theo cách thức chiều chuyển động từ điểm khởi đầu sẽ là:

- Chiều X dương đối với gia công ngoài
  ...và/hoặc...
  Chiều Z âm đối với gia công ngoài
- Chiều X âm đối với gia công trong
  ...và/hoặc...
  Chiều Z âm đối với gia công trong

Ở đây còn có các nguyên công *tiện ngược* hoặc *doa ngược* được dùng trong lập trình CNC, nhưng chúng có liên hệ chặt chẽ với các phương pháp tiện thông dụng. Trong các nguyên công cắt gọt phổ biến trên máy tiện CNC, sự thay đổi chiều theo một trục sẽ tạo thành hốc hoặc *mặt lõm*.

Mặt lõm thường do kỹ sư thiết kế để cắt lõm một phần trên chi tiết, ví dụ để chi tiết lắp khớp với vai (bậc), mặt, hoặc bề mặt của chi tiết gia công.

Trong lập trình tiện CNC, mặt lõm có thể gia công với dao bất kỳ được dùng với chiều sâu cắt thích hợp, và *khoảng hở góc ngược thích hợp* (góc thoát dao). Đây là yêu cầu cần được xem xét chi tiết.



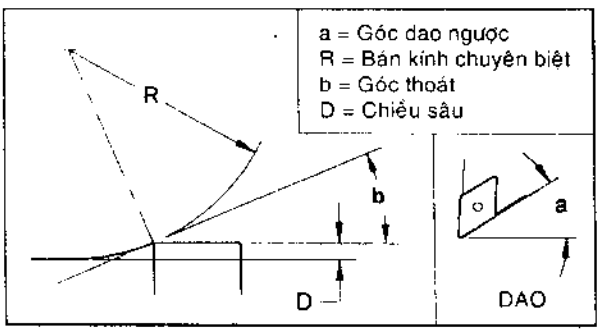
Hình 33.13. *Ví dụ tính toán khoảng hở góc ngược*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Hình 33.13 minh họa bản vẽ đơn giản một con lăn. Ở khoảng giữa con lăn có mặt trụ lõm giữa Ø1.029 và Ø.939. Vấn đề là tính toán, không ước tính, góc ngược cực đại của dao có thể sử dụng để gia công mặt lõm đó chỉ với một nguyên công.

Bước thứ nhất là nghiên cứu bản vẽ. Cần biết hiệu số giữa các đường kính và bán kính mặt lõm. Hình 33.14 minh họa chi tiết *các dữ liệu đó* (ngoại trừ góc b) từ bản vẽ.



Hình 33.14. *Dữ liệu cần thiết để tính góc b*

Công thức tính toán góc b sử dụng hàm lượng giác đơn giản. Trước hết, cần tính *chiều sâu* của mặt lõm D, là một nửa hiệu giữa hai đường kính:

$$D = \frac{\text{ĐƯỜNG KÍNH LỚN} - \text{ĐƯỜNG KÍNH NHỎ}}{2}$$

Khi biết D, có thể tính góc b theo công thức:

$$b = \cos^{-1}\left(\frac{R-D}{R}\right)$$

Ví dụ, tính toán theo dữ liệu (Hình 33.13) và D

$$b = \cos^{-1}\left(\frac{.5625-.045}{.5625}\right) = 23.07392$$

Đối với gia công thực tế, cần chọn dao có góc (thoát) ngược a lớn hơn góc tính toán b. Đối với bản vẽ (Hình 33.13), góc b = 23.07^0, dao được chọn có thể có mảnh chắp hình thoi 55^0 (khoảng hở góc ngược a là $30\text{-}32^0$), hoặc hình thoi 35^0 (góc ngược a là $50\text{-}52^0$), cả hai đều lớn hơn góc b đã tính. Các góc thực tế tùy thuộc vào nhà chế tạo dụng cụ cắt.

Kiểu tính toán này rất quan trọng đối với mặt lõm bất kỳ, được lập trình với chu kỳ gia công hoặc block. Ví dụ này chỉ minh họa một khả năng, nhưng có thể dùng cho các tính toán bất kỳ, yêu cầu khoảng hở góc ngược (góc thoát ngược).

TỐC ĐỘ TRỤC CHÍNH TRONG CHẾ ĐỘ CSS

Từ các phần đã nêu, bạn có thể nhớ CSS là viết tắt của *Constant Surface Speed* (tốc độ bề mặt không đổi). Tính năng này của máy tiện CNC liên tục tính toán tốc độ trục chính thực theo *số vòng quay/phút* (r/min), dựa trên giá trị nhập đã lập trình của tốc độ bề mặt. Tốc độ bề mặt lập trình được tính theo *feet/phút* - ft/min (hệ Anh) hoặc *mét/phút*-m/min (hệ mét).

Trong chương trình, giá trị nhập "*theo phút*" sử dụng lệnh chuẩn bị G96, nhập trực tiếp giá trị r/min sử dụng lệnh G97.

Constant Surface Speed (CSS) là tính năng rất mạnh của hệ điều khiển, có một vấn đề nhỏ liên quan đến tính năng này, thường bị bỏ qua, hoặc ít nhất không được đánh giá đúng mức. "Vấn đề nhỏ" này sẽ được minh họa trong chương trình đơn giản dưới đây.

Ví dụ này chỉ nêu ra vài block ở đầu chương trình, khi dao cắt tiếp cận chi tiết gia công, nhưng đủ dữ liệu để xét câu hỏi sau đây.

```
O3306
N1 G20 T0100
N2 G96 S450 M03
N3 G00 G41 X0.7 Z0 T0101 M08
N4 . . .
```

Câu hỏi là: Tốc độ trục chính thực (theo r/min) là bao nhiêu, khi thực thi block N2? Vào thời điểm đó, chưa thể biết tốc độ trục chính. Thậm chí không thể biết, trừ khi biết *đường kính* hiện hành, đường kính tại đó dao đang định vị. Hệ điều khiển liên tục theo dõi vị trí dao hiện hành vào mọi thời điểm. Do vậy, khi thực thi block N2, r/min thực của trục chính sẽ được tính toán đối với đường kính hiện hành, được lưu trong bộ điều khiển, chuyên biệt theo giá trị nhập bù hình học. Ví dụ, bạn hãy xét đường kính hiện hành có giá trị 23.5 inch (X23.5).

Từ công thức r/min tiêu chuẩn, tốc độ trục chính được tính toán với 450ft/min và Ø23.5 là 73r/min, tuy thấp, nhưng chính xác. Tại block kế tiếp, N3, vị trí dao gần với chi tiết hơn, tại đường kính .700 (X0.7). Từ cùng công thức tiêu chuẩn, tốc độ trục chính được tính toán cho đường kính đó là 2455 r/min – hơi nhanh, nhưng cũng chính xác. Đây là vấn đề? Có thể không phải là vấn đề đối với số máy công cụ, nhưng nếu là vấn đề, thì giải pháp sẽ như sau.

Vấn đề có thể liên quan với chuyển động nhanh từ Ø23.5 đến Ø.700. Khoảng cách hành trình thực (một phía của chi tiết) là (23.5 -0.700)/2 = 11.400. Với hành trình nhanh này, dao cắt sẽ chuyển động 11.400 inch – *đồng thời*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

– tốc độ trục chính tăng từ giá trị 73 r/min đến 2455 r/min. Tuỳ theo hệ điều khiển và khả năng xử lý của hệ đối với tình huống đó, dao có thể thực sự bắt đầu cắt gọt *ở tốc độ trục chính thấp hơn so với giá trị tính toán.*

Nếu xảy ra kiểu tình huống đó và dẫn đến vấn đề, bước duy nhất có thể thực hiện là *lập trình trước*, tốc độ trục chính mong muốn (r/min), *trước* chuyển động tiếp cận chi tiết của dụng cụ cắt, *sau đó* chuyển sang chế độ tốc độ bề mặt không đổi (CSS) và tiếp tục gia công.

```
O3307
N1 G20 T0100
N2 G97 S2455 M03                    (R/MIN PRESET)
N3 G00 G41 X0.7 Z0 T0101 M08
N4 G96 S450 M03
N5 ...
```

Vấn đề nêu trên cần được đánh giá rõ hơn. Điều đã thực hiện là trục chính khởi động với tốc độ r/min mong muốn, trước khi dao tiếp cận chi tiết, trong block N2. Trong block N3, dao chuyển sang bắt đầu cắt, trong khi trục chính đạt đến tốc độ lập trình cao nhất. Khi đạt đến vị trí đích dọc theo trục X (block N3), chế độ CSS tương ứng sẽ có hiệu lực cho mọi đường cắt kế tiếp.

Ví dụ này chưa hẳn đã phản ánh sự lập trình hàng ngày cho máy tiện CNC. Trong tình huống đó, cần thực hiện thêm vài tính toán để giải quyết vấn đề. Một số hệ thống CAD/CAM có thể thực hiện chính xác điều đó một cách hoàn toàn tự động. Nếu chưa biết vị trí X hiện hành của dụng cụ cắt, bạn có thể ước lượng dễ dàng.

ĐỊNH DẠNG LẬP TRÌNH TIỆN

Nếu xem kỹ các ví dụ đã nêu, bạn sẽ nhận thấy tính nhất quán trong các chương trình. Điều đó có thể được gọi là phong cách, định dạng, mẫu chuẩn, ... Từng nhà lập trình đều phát triển phong cách riêng theo thời gian và kinh nghiệm. Phong cách ổn định là rất quan trọng để xây dựng chương trình, thay đổi, hoặc diễn dịch chương trình một cách hiệu quả.

Định dạng chương trình – Mẫu chuẩn

Hầu hết các chương trình đều tuân theo dạng chương trình xác định. Bạn hãy lưu ý các chương trình tiện CNC thường bắt đầu với lệnh G20 hoặc G21và có lẽ là vài lệnh xóa. Block kế tiếp là chọn dụng cụ cắt, tiếp theo là dữ liệu tốc độ trục chính, v.v... Định dạng này, hầu như không thay đổi giữa các chương trình, tuân theo mẫu chuẩn xác định, tạo thành *khung sườn* để viết chương trình

Định dạng chương trình chung

Bạn nên thuộc lòng định dạng chuẩn này để viết các chương trình CNC. Các chi tiết chưa hiểu sẽ trở nên rõ ràng hơn sau khi bạn có kiến thức tổng quát về các quan hệ và các chi tiết được dùng trong nhiều phương pháp lập trình. Dưới đây là khung chuẩn về chương trình tiện CNC.

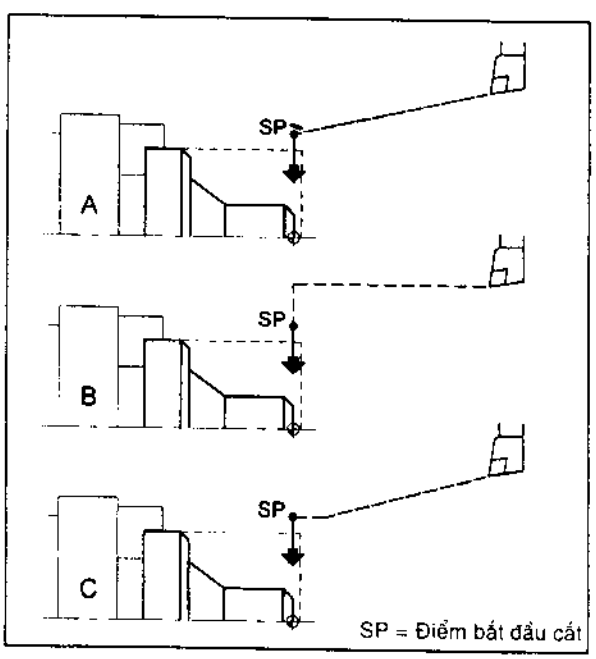
➲ **Khung chương trình chung – Tiện:**

```
O.. (TÊN CHƯƠNG TRÌNH)
N1 G20 G40 G99                      (PROGRAM START UP)
N2 T..00 M4..                   (TOOL AND GEAR RANGE)
N3 G97 S.. M03                    (STABILIZE R/MIN)
N4 G00 [G41/G42] X.. Z.. T.. M08  (APPROACH)
N5 G96 S..                          (CUTTING SPEED)
N6 G01 [X../Z..] F..  (FIRST CUTTING MOTION)
N7 ...
...
...                                    (MACHINING)
...
N.. G00 [G40] X.. Z.. T,.00 (TOOL CHG POSITION)
N.. M01                             (OPTIONAL STOP)
...
N.. M30                             (PROGRAM END)
%
```

Cấu trúc tổng quát này có thể áp dụng cho hầu hết các chương trình tiện. Bạn hoàn toàn có thể điều chỉnh theo các yêu cầu cụ thể. Ví dụ, không phải mọi công việc đều yêu cầu tính ổn định tốc độ trục chính, do đó có thể không cần block N3. Điều đó còn có ý nghĩa là sự quay M03 có thể chuyển sang block N5. Cấu trúc này chỉ là ví dụ, không phải là định dạng bất biến.

Tiếp cận chi tiết gia công

Phần quan trọng trong cấu trúc chương trình trên là phương pháp tiếp cận chi tiết



Hình 33.15. Tiếp cận an toàn đến chi tiết – ví dụ về gia công mặt đầu

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

đang quay. Nếu chi tiết là đồng tâm, tiếp cận tương tự tùy chọn A trên Hình 33.15. Tuy chỉ minh họa gia công mặt đầu, nhưng sự tiếp cận về logic cũng có thể áp dụng cho tiện mặt trụ ngoài hoặc doa lỗ. Bạn cần bảo đảm điểm khởi đầu SP ở phía trên đường kính, ít nhất là 0.100 inch theo một phía, nếu chưa biết *chính xác* đường kính thực. Tùy chọn B về tiếp cận của dao là hai trục đơn. Đây là biến thể của ví dụ A, chuyển động trục X có thể chia thành chuyển động nhanh và chuyển động cắt. Cuối cùng, các con sử dụng khoảng hở trên trục Z, cách xa mặt đầu. Chuyển động hướng đến mặt này cũng có thể chia thành chuyển động nhanh và chuyển động tuyến tính.

Các phương pháp nêu trên có nhiều biến thể. Yêu cầu chính về tiếp cận chi tiết là tính an toàn. Sự va chạm giữa dao và chi tiết quay có thể gây ra các hậu quả nghiêm trọng.

Tiện và doa là chủ đề lớn với nhiều nội dung. Chương 34 sẽ trình bày các chu kỳ tiện.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

34 CHU KỲ TIỆN

Chương 33 đã trình bày nhiều phương pháp lập trình quỹ đạo tiện và doa, giới thiệu một số kỹ thuật, đặc biệt là quỹ đạo dao gia công *tinh*, nhưng chưa tập trung chú ý vào quá trình cắt gọt lượng dư khi gia công thô. Chương này sẽ trình bày các phương pháp lập trình gia công thô và tinh.

SỰ CẮT GỌT TRÊN MÁY TIỆN

Một trong các công việc tốn nhiều thời gian khi lập trình bằng tay cho máy tiện CNC là cắt gọt lượng dư gia công, chủ yếu là từ phôi trụ, được gọi là tiện thô hoặc doa thô.

Lập trình bằng tay quỹ đạo dao gia công thô yêu cầu phối hợp nhiều đường cắt, với một block chương trình cho từng chuyển động dao. Đối với gia công thô biên dạng phức tạp, phương pháp này tốn nhiều thời gian và không hiệu quả, dễ dẫn đến các sai sót. Một số nhà lập trình, để giảm thời gian, đã hy sinh chất lượng chương trình bằng cách để lại lượng dư *không* đều cho gia công tinh, dẫn đến dụng cụ cắt bị mòn nhanh, thậm chí giảm độ bóng bề mặt.

Các bộ điều khiển tiện hiện đại rất mạnh và thuận tiện trong lĩnh vực gia công thô. Hầu như mọi hệ thống tiện CNC đều có tính năng cho phép quỹ đạo dao cắt thô được xử lý một cách tự động, sử dụng các *chu kỳ đặc biệt*. Gia công thô không chỉ là ứng dụng đối với các chu kỳ này, thực tế còn có các chu kỳ đặc biệt để *cắt ren* và *gia công rãnh đơn giản*. Chu kỳ cắt ren và rãnh sẽ được trình bày chi tiết trong các Chương 35, 36, và 37.

Các chu kỳ đơn giản

Các bộ điều khiển Fanuc hỗ trợ nhiều chu kỳ tiện đặc biệt. Có ba chu kỳ đơn giản trên bộ điều khiển Fanuc được trình bày trong Chương này. Chúng xuất hiện lần đầu tiên trong các bộ CNC cũ và bị giới hạn do trình độ công nghệ thời kỳ đó. Các tài liệu gọi chúng là *chu kỳ đơn giản*, *chu kỳ cố định*, hoặc *chu kỳ có sẵn*, về bản chất, tương tự các nguyên công khoan trên trung tâm gia công và máy phay CNC. Hai trong số các chu kỳ cũ này được dùng để tiện và doa, chu kỳ thứ ba là chu kỳ tiện ren rất đơn giản. Chương này sẽ trình bày chu kỳ tiện và chu kỳ doa.

Các chu kỳ phức tạp

Với sự tiến bộ của công nghệ máy tính, các nhà chế tạo bộ điều khiển đã phát triển khả năng các chu kỳ cắt gọt đáp ứng nhiều nguyên công tiện phức tạp và làm cho chúng trở thành bộ phận tích hợp trong hệ thống điều khiển trên máy tiện. Các chu kỳ đặc biệt này được Fanuc gọi là *Chu Kỳ Lặp Nhiều Lần*. Sự cải tiến chính của chúng so với các chu kỳ đơn giản là tính linh hoạt rất cao. Một số chu kỳ tiện tiến dùng cho tiện và doa, số khác là cắt ren và gia công rãnh.

NGUYÊN LÝ CỦA CÁC CHU KỲ TIỆN

Tương tự các nguyên công khác trên trung tâm gia công CNC, mọi chu kỳ tiện đều dựa trên các nguyên lý công nghệ. Nhà lập trình chỉ nhập các dữ liệu chung (thường là các tham số cắt gọt), hệ thống CNC sẽ tính toán các chi tiết cho từng đường cắt riêng rẽ. Các tính toán này dựa trên sự kết hợp các dữ liệu cố định và dữ liệu biến thiên. Các chuyển động dao trở về trong mọi chu kỳ này đều là tự động, và chỉ các giá trị sẽ thay đổi là được chuyên biệt khi gọi chu kỳ.

Các chu kỳ đơn giản chỉ được thiết kế để cắt theo đường thẳng, không có vạt góc, bo tròn góc, côn, hoặc mặt lõm. Các chu kỳ đơn giản chỉ có thể được dùng để cắt gọt theo chiều đứng, ngang, hoặc theo góc để cắt gọt côn. Các chu kỳ ban đầu này không thể thực hiện cùng các nguyên công như các chu kỳ lặp nhiều lần trên bộ điều khiển hiện đại - ví dụ, chúng không thể gia công thô các bán kính hoặc thay đổi chiều cắt. Đơn giản, chúng không thể cắt gọt biên dạng.

Trong nhóm các chu kỳ tiện đơn giản, có hai chu kỳ cho phép cắt gọt thô từ phôi trụ hoặc côn. Từng block của các chu kỳ này thay thế bốn block *bình thường* trong chương trình gia công. Trong nhóm các chu kỳ lặp nhiều lần, có nhiều chu kỳ cắt rãnh và gia công ren. Chu kỳ lặp nhiều lần có khả năng gia công một số biên dạng phức tạp.

G90 - CHU KỲ CẮT GỌT THẲNG

Bạn không nên nhầm lẫn giữa G90 đối với máy tiện và G90 trên trung tâm gia công. Trong tiện, G90 là chu kỳ tiện, còn trong phay G90 là chế độ tuyệt đối:

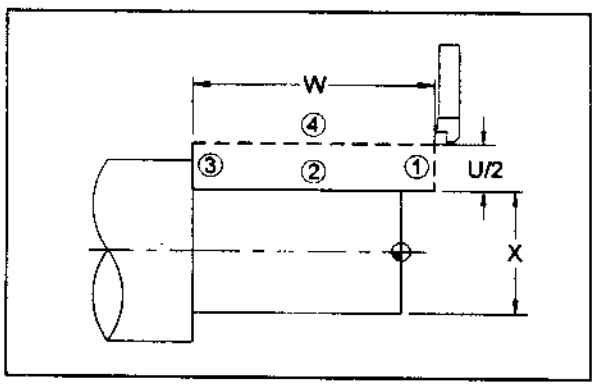
G90 là chế độ tuyệt đối trên máy phay, các trục X và Z là chế độ tuyệt đối trên máy tiện.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

(X, Z là chế độ tuyệt đối trên máy phay, các trục U và W là chế độ số gia trên máy tiện.

Chu kỳ xác định theo lệnh chuẩn bị G90 (nhóm mã G *Kiểu A*) được gọi là *chu kỳ cắt gọt thẳng*. Mục đích của chu kỳ này là cắt gọt lượng dư giữa điểm bắt đầu của dụng cụ cắt và các tọa độ được xác định theo trục X và Y. Kết quả cắt gọt là tiện hoặc doa, *thường song song* với đường tâm trục chính và trục Z là trục cắt chính. Chu kỳ G90 được dùng chủ yếu để cắt gọt phôi theo kiểu chữ nhật. Chu kỳ G90 cũng có thể được dùng để tiện côn. Hình 34.1 minh họa cấu trúc chu kỳ và các chuyển động.



Hình 34.1. *Cấu trúc chu kỳ đơn giản G90 – Ứng dụng cắt gọt thẳng*

Định dạng chu kỳ

Chu kỳ G90 có hai định dạng lập trình được xác định trước. Định dạng thứ nhất chỉ dùng cho cắt gọt thẳng, theo trục Z (Hình 34.1)

❑ Định dạng 1

```
G90 X(U).. Z(W).. F..
```

Trong đó:

X = Đường kính cần cắt gọt
Z = Kết thúc đường cắt theo vị trí Z
F = Tốc độ cắt (in/rev hoặc mm/rev)

Định dạng thứ hai bổ sung thêm tham số I hoặc R cho block và được thiết kế cho các chuyển động tiện côn, với ưu thế theo trục Z (Hình 34.2).

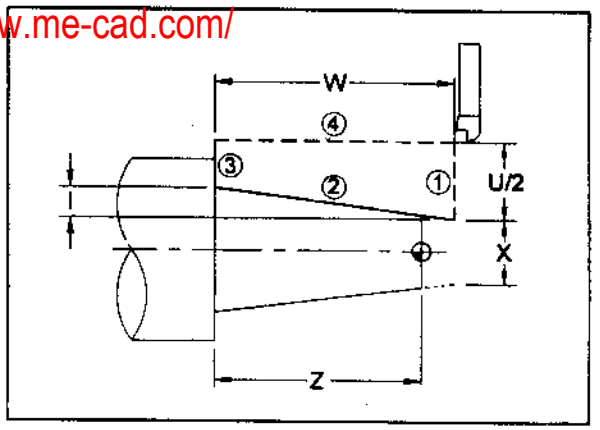
❑ Định dạng 2 (hai phiên bản)

```
G90 X(U).. Z(W).. I.. F..
G90 X(U).. Z(W).. R.. F..
```

Trong đó:

X = Đường kính cần cắt gọt
Z = Kết thúc cắt gọt theo vị trí Z
I(R) = Khoảng cách và chiều côn (đối với cắt gọt thẳng I = 0 hoặc R = 0)
F = Tốc độ cắt (theo in/rev hoặc mm/rev)

Trong cả hai ví dụ, ký hiệu các trục X và Z được dùng cho lập trình theo chế độ tuyệt đối,



Hình 34.2. *Cấu trúc chu kỳ G90 - ứng dụng tiện côn*

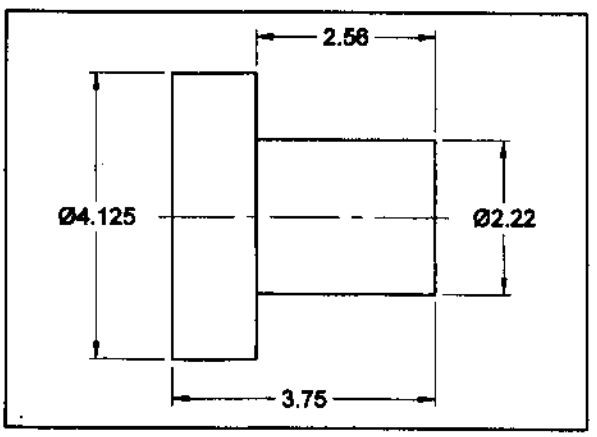
vị trí dao từ zero chương trình. Ký hiệu các trục U và W dùng cho lập trình số gia, biểu thị khoảng cách hành trình thực của dao từ vị trí hiện hành. Địa chỉ F là tốc độ cắt, thường là *inch/vòng quay* (in/rev) hoặc *milimét/vòng quay* (mm/rev). Địa chỉ I được dùng để tiện côn theo chiều ngang, có giá trị tương đương *một nửa* khoảng cách từ đường kính ở cuối phần côn, đến đường kính ở đầu phần côn. Địa chỉ R thay cho I, và chỉ khả dụng trên các *bộ phận điều khiển mới*.

Để xóa chu kỳ G90, toàn bộ điều cần thực hiện là sử dụng lệnh chuyển động bất kỳ – G00, G01, G02, hoặc G03. Nói chung thường là lệnh chuyển động nhanh G00 :

```
G90 X(U).. Z(W).. I.. F..
...
...
G00 ...
```

Ví dụ về tiện thẳng

Để minh họa ứng dụng thực tiễn của chu kỳ G90, bạn hãy nghiên cứu Hình 34.3, đây là nguyên công tiện đường kính đơn giản, từ đường kính phôi Ø4.125 xuống đường kính



Hình 34.3. *Ví dụ chu kỳ G90 tiện thẳng đơn giản – các chương trình O3401 và O3402*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ø2.22 inch, trên chiều dài .556 inch. Ở đây, cắt cho được côn, trình theo phương pháp block không có vạt góc, không có côn, cũng không bo tròn góc. Điều này giới hạn công dụng thực tiễn của chu kỳ G90, chỉ tiện thô rất đơn giản.

Do G90 là chu kỳ tiện thô, trước hết cần chọn *chiều sâu* của từng đường cắt, sau đó là *lượng dư* còn lại để gia công tinh. Để quyết định chiều sâu của từng đường cắt, cần tìm lượng dư thực sự cần cắt gọt từ đường kính phôi. Lượng dư thực sự được tính toán *theo một phía*, nghĩa là giá trị bán kính, dọc theo trục X:

(4.125 - 2.22) / 2 = .9525 inch

Với lượng dư .030 trên một phía để gia công tinh, cần trừ giá trị .030 khỏi lượng dư X, do đó tổng lượng gia công thô là .9525 inch. Kế tiếp, cần chọn số lần cắt cho chiều sâu toàn phần. Đối với năm lần cắt đều, mỗi chiều sâu cắt sẽ là .1845, đối với sáu lần cắt, .1538. Sẽ chọn sáu lần cắt và để lại .030 inch trên một phía, tương ứng .060 trên đường kính – đường kính thứ nhất sẽ là X3.8175. Ngoài ra, lượng dư .005 sẽ để lại trên mặt đầu, do đó vạt mặt trên trục Z sẽ ở Z-2.555. Khoảng hở phía trên đường kính và ở phía trước chi tiết thường là .100 inch.

```
O3401
(G90 CHU KỲ TIỆN THẲNG - TUYỆT ĐỐI)
N1 G20
N2 T0100 M41
N3 G96 S450 M03
N4 G00 X4.325 Z0.1 T0101 M08     (START POINT)
N5 G90 X3.8175 Z-2.555 F0.01          (PASS 1)
N6 X3.51                              (PASS 2)
N7 X3.2025                            (PASS 3)
N8 X2.895                             (PASS 4)
N9 X2.5875                            (PASS 5)
N10 X2.28                             (PASS 6)
N11 G00 X10.0 Z2.0 T0100 M09
N12 M01                        (END OF ROUGHING)
```

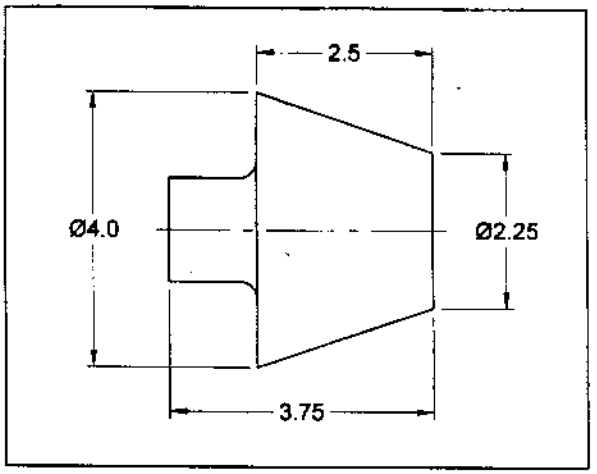
Nếu muốn, bạn có thể dùng phương pháp lập trình theo chế độ số gia. Tuy nhiên, sự theo dõi tiến độ chương trình với các tọa độ tuyệt đối dễ hơn so với các khoảng cách số gia. Dưới đây là phiên bản theo chế độ số gia:

```
O3402
(G90 CHU KỲ TIỆN THẲNG - SỐ GIA)
N1 G20
N2 T0100 M41
N3 G96 S450 M03
N4 G00 X4.325 Z0.1 T0101 M08     (START POINT)
N5 G90 U-0.5075 W-2.655 F0.01         (PASS 1)
N6 U-0.3075                           (PASS 2)
N7 U-0.3075                           (PASS 3)
N8 U-0.3075                           (PASS 4)
N9 U-0.3075                           (PASS 5)
N10 U-0.3075                          (PASS 6)
N11 G00 X10.0 Z2.0 T0100 M09
N12 M01                        (END OF ROUGHING)
```

Chu kỳ này rất đơn giản trong cả hai phiên bản – toàn bộ điều cần làm là tính toán đường kính mới cho từng lần cắt thô. Nếu quỹ đạo dao cắt cho được lập trình theo phương pháp block (không dùng chu kỳ G90), chương trình sẽ dài hơn gấp ba lần.

Ví dụ về tiện côn

Hình 34.4 minh họa bản vẽ chi tiết côn, áp dụng chu kỳ đơn giản G90.

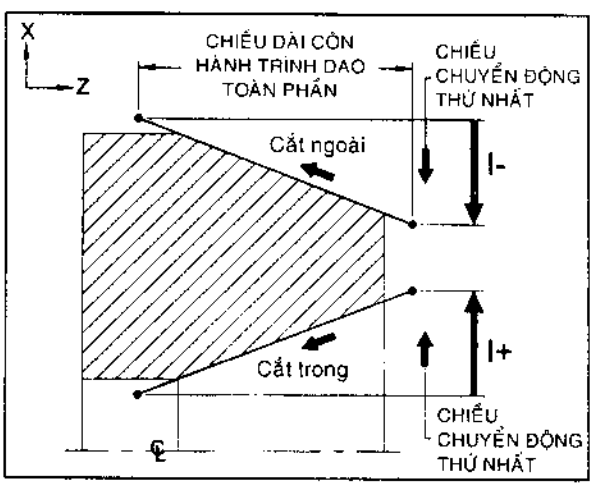


Hình 34.4. *Ví dụ về chu kỳ G90 trong tiện côn – chương trình O3403*

Để phân biệt giữa các phương pháp tiện thẳng và tiện côn, sử dụng chu kỳ G90, cần có cách thức phân biệt giữa hai dạng cắt gọt đó.

Sự khác biệt là bổ sung tham số I vào mục gọi chu kỳ, biểu thị *giá trị côn* và *chiều côn theo một phía*. Giá trị này được gọi là *giá trị bán kính có dấu*. Đây là giá trị I do liên quan với trục X. Đối với tiện thẳng, giá trị I luôn luôn là zero và không cần viết trong chương trình. Ý nghĩa của I là tiện côn nếu có giá trị khác zero (Hình 34.5).

Minh họa cho thấy giá trị I được tính toán theo giá trị đơn, nghĩa là theo một phía (giá trị



Hình 34.5. *Giá trị I được dùng cho chu kỳ G90 – tiện trong và tiện ngoài*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

bán kính), với chiều chuyển biệt dựa trên khoảng cách hành trình *toàn phần* và chiều của chuyển động thứ nhất từ vị trí khởi đầu.

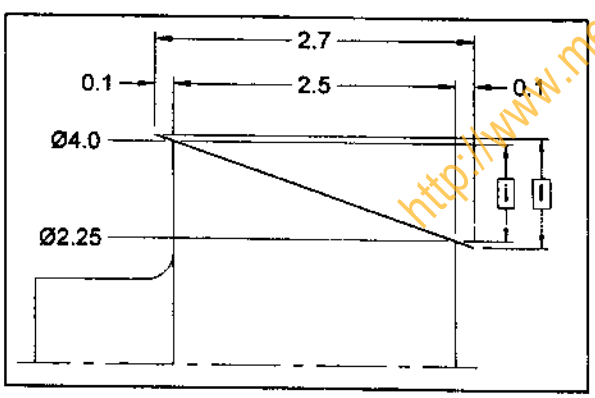
Có hai nguyên tắc cơ bản đối với cắt gọt côn sử dụng G90:

- Nếu chiều của chuyển động dao thứ nhất theo X âm, giá trị I là âm
- Nếu chiều của chuyển động dao thứ nhất theo X dương, giá trị I là dương

Trên máy tiện CNC với chiều dương X *phía trên* đường tâm trục chính, giá trị I là *âm* đối với *tiện côn ngoài* và *dương* đối với *tiện côn trong*.

Để lập trình chi tiết trên Hình 34.5, bạn cần nhớ đây là bản vẽ chi tiết gia công và không có các khoảng hở . Trước hết bạn phải bổ sung mọi khoảng hở cần thiết, sau đó tính giá trị I.

Trong ví dụ này, khoảng hở .100 được bổ sung cho cả hai đầu côn, tăng chiều dài dọc trục từ 2.5 lên 2.7 inch. Tính toán giá trị I đòi hỏi *chiều dài thực* của hành trình dao đồng thời duy trì côn không đổi. Có thể sử dụng phương pháp tam giác đồng dạng hoặc hàm lượng giác để tính toán. Hình 34.6 minh họa chi tiết các giá trị đã biết và chưa biết đối với tính toán giá trị I.



Hình 34.6. *Các giá trị đã biết và chưa biết đối với tiện côn-chương trình O3403. Biết "i", cần tính "I".*

Ví dụ này đưa ra phương pháp tính toán đơn giản nhất, áp dụng *định lý tam giác đồng dạng*.

Hai tam giác là đồng dạng, nếu các cạnh tương ứng của chúng tỷ lệ với nhau.

Trong lập trình, bạn thường gặp tình huống có thể giải quyết bằng nhiều phương pháp. Bạn hãy chọn phương pháp thích hợp nhất, sau đó áp dụng phương pháp khác để kiểm nghiệm kết quả. Ở đây sẽ dùng cả hai phương pháp để kiểm nghiệm tính chính xác của tính toán.

➲ Phương pháp tam giác đồng dạng

Trước hết tính hiệu số I giữa hai đường kính đã biết từ bản vẽ:

i = (4 - 2.25) / 2 = .875 inch

Do đó, tỷ số của hai tam giác đồng dạng sẽ là:

I / 2.7 = i / 2.5

Khi biết i, có thể dễ dàng tính I

I / 2.7 = .875 / 2.5
I = (.875 × 2.7) / 2.5
I = .945 ich *...là giá trị cần thiết để lập trình*

➲ Phương pháp lượng giác

Từ bản vẽ có thể nhận thấy

I = 2.7 × tga

cần tính giá trị hàm số tang trước

tga = i / 2.5
tga = .875 / 2.5 = .350

Giá trị I được tính như sau:

I = 2.7 × 0.35
I = 0.945 *...là giá trị cần thiết để lập trình*

Trong cả hai trường hợp, tính toán có cùng kết quả, khẳng định tính chính xác để lập trình. Chương trình O3403 là kết quả cuối cùng:

```
O3403
(G90 VÍ DỤ TIỆN CÔN - 1)
N1 G20
N2 T0100 M41
N3 G96 S450 M03
N4 G00 X4.2 Z0.1 T0101 M08                (START)
N5 G90 X3.662 Z-2.495 I-0.945 F0.01          (1)
N6 X3.324                                    (2)
N7 X2.986                                    (3)
N8 X2.648                                    (4)
N9 X2.310                                    (5)
N10 G00 X10.0 Z2.0 T0100 M09       (CLEAR POS.)
N11 M01                          (END OF ROUGHING)
```

Ví dụ tiện côn và tiện thẳng

Trong gia công bạn thường gặp chi tiết có phần côn và phần trục. Hình 34.7 minh họa bản vẽ đơn giản chi tiết có phần côn và bậc (vai) trụ.

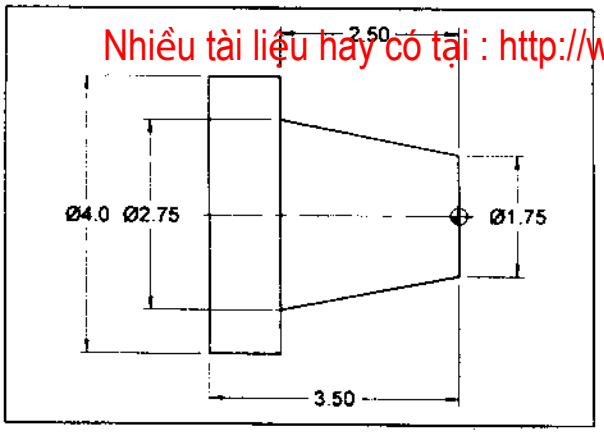
Sử dụng chu kỳ G90, nguyên công đòi hỏi tiện côn hướng đến bậc trục. Trong trường hợp này có thể dùng một chu kỳ G90 nhưng có thể dẫn đến sự cắt gọt *dư* hoặc *thiếu*. Cách tốt nhất có lẽ là sử dụng *hai chế độ* của chu kỳ này, một để tiện thô phần vai trụ và một để tiện thô phần côn.

Tương tự ví dụ trước, giá trị độ côn I cần được tính toán, sử dụng định lý tam giác đồng dạng. Chiều cao i trên chiều dài 2.50 inch là một nửa của hiệu giữa Ø2.750 và Ø1.750:

i = (2.75 - 1.75) / 2 = .500 inch

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 34.7. Ví dụ về sử dụng chu kỳ G90 để tiện côn và trụ thẳng – O3404

Đối với chiều dài côn, cần để lại lượng dư .005 ở bậc trụ để gia công tinh và phần côn cần tăng thêm .100 ở mặt đầu, do đó chiều dài côn toàn phần là 2.595:

2.5 - .005 + .100 = 2.595

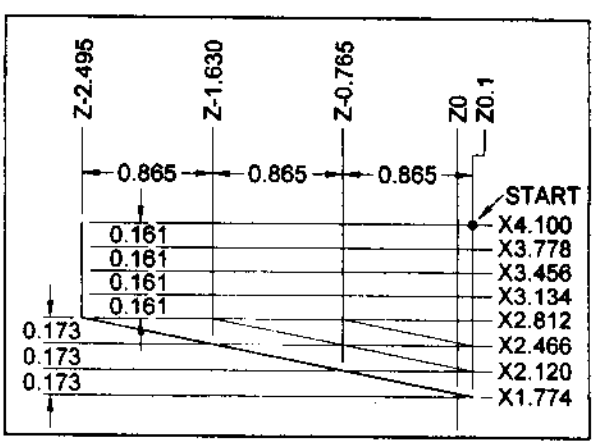
Có thể tính giá trị I dựa trên các giá trị đã biết:

I / 2.595 = .500/2.5
I = (.500 × 2.595) / 2.5
I = .519 inch ...*theo chiều âm*

Đối với gia công thô, cần để lại lượng dư .030 trên một phía dọc theo trục X. nghĩa là .060 trên đường kính.

Trong gia công thô, điều quan trọng là chọn chiều sâu cắt thích hợp, bảo đảm an toàn và các điều kiện cắt gọt. Trong ví dụ này có thể chọn chiều sâu cắt theo kỹ thuật lập trình đơn giản. Nếu chọn chiều sâu cắt một cách tùy ý, sẽ khó kiểm soát chiều sâu đường cắt thô *cuối cùng*. Bạn nên chọn *số lần cắt* có chiều sâu bằng nhau (Hình 34.8).

Đối với tính toán này, toàn bộ điều cần thiết là chia khoảng cách theo từng phía cho số lần cắt. Kết quả sẽ là chiều sâu cắt bằng nhau trong toàn bộ quá trình gia công thô. Nếu chiều sâu cắt quá nhỏ hoặc quá lớn, bạn chỉ cần tính lại với số lần cắt khác. Chiều sâu cắt thích hợp là kiến thức về công nghệ, nhà lập trình CNC phải biết rõ.



Hình 34.8. Chiều sâu cắt tính toán trong chương trình O3404

Trên Hình 34.8 có bốn lần cắt với chiều sâu .161 để tiện thô phần bậc trục và ba lần cắt với chiều sâu .173 cho phần côn.

Chương trình O3404 sẽ sử dụng các tính toán đó:

```
O3404
(G90 VÍ DỤ TIỆN CÔN - 2)
N1 G20
N2 T0100 M41
N3 G96 S450 M03
N4 G00 X4.1 Z0.1 T0101 M08                    (START)
N5 G90 X3.778 Z-2.495 F0.01            (STRAIGHT 1)
N6 X3.456                              (STRAIGHT 2)
N7 X3.134                              (STRAIGHT 3)
N8 X2.812                              (STRAIGHT 4)
N9 G00 X3.0       (CHANGE STRAIGHT TO TAPERED)
N10 G90 X2.812 Z-0.765 I-0.173          (TAPERED 1)
N11 Z-1.63 I-0.346                      (TAPERED 2)
N12 Z-2.495 I-0.519          (TAPERED 3 - FINAL)
N13 G00 X10.0 Z2.0 T0100 M09            (CLEAR POS.)
N14 M01                          (END OF ROUGHING)
```

Tóm lại, để tính toán giá trị của I hoặc R trong G90 khi tiện côn – *trong* hoặc *ngoài*, bạn hãy dùng công thức:

$$I(R) = \frac{\text{ĐƯỜNG KÍNH NHỎ} - \text{ĐƯỜNG KÍNH LỚN}}{2}$$

Kết quả sẽ có *cả dấu của giá trị I*.

G94 – CHU KỲ GIA CÔNG MẶT ĐẦU

Chu kỳ tiện đơn giản, tương tự G90, được lập trình với lệnh G94, được gọi là *chu kỳ gia công mặt đầu* (vạt mặt). Mục đích của G94 là cắt gọt lượng dư giữa vị trí khởi đầu của dụng cụ cắt và các tọa độ xác định theo trục X và Z. Kết quả là đường cắt (tiện) thẳng, *thường vuông góc* với đường tâm trục chính. Trong chu kỳ này, trục X là chiều cắt gọt chính. Chu kỳ G94 được dùng chủ yếu để gia công mặt đầu và có thể cắt gọt mặt côn đứng đơn giản.

Chu kỳ G94 về logic là đồng nhất với chu kỳ G90, nhưng chủ yếu cắt gọt theo trục X, hơn là theo trục Z.

G94 thường được dùng để gia công thô mặt đầu của chi tiết hướng đến đường tâm trục chính hoặc gia công mặt bậc trụ.

Định dạng chu kỳ

Tương tự các chu kỳ khác, chu kỳ gia công mặt đầu G94 cũng có định dạng lập trình được

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

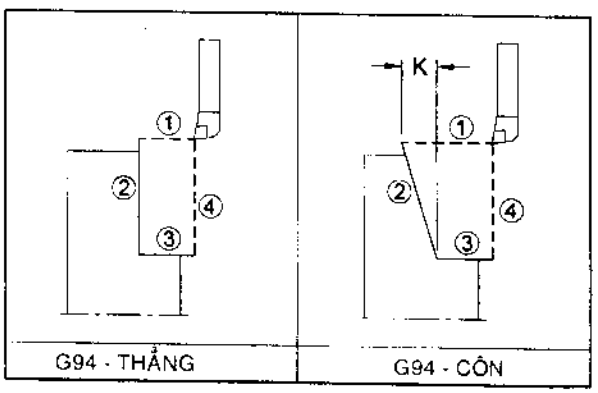
xác định sẵn. Đối với gia công mặt đầu vuông góc với đường tâm trục chính, định dạng như sau:

```
G94 X(U).. Z(W).. F..
```

Đối với mặt côn đứng:

```
G94 X(U).. Z (W).. K.. F..
```

Các trục X, Y được dùng cho lập trình tuyệt đối, các trục U, W dùng cho lập trình số gia, địa chỉ F là tốc độ cắt. Tham số K, nếu lớn hơn zero, được dùng để gia công mặt côn theo chiều đứng. Hình 34.9 minh họa tất cả các tham số lập trình và các bước cắt gọt, áp dụng cùng quy trình như chu kỳ G90.



Hình 34.9. *Cấu trúc chu kỳ tiện G94 – ứng dụng tiện thẳng và côn theo chiều đứng.*

CHU KỲ LẶP NHIỀU LẦN

Khác với chu kỳ cố định dùng cho các nguyên công khoan trên trung tâm gia công, hoặc chu kỳ tiện đơn giản G90 và G94, các chu kỳ tiên tiến trên máy tiện CNC có nhiều tính năng đặc biệt. Tính năng chính và nổi bật của các chu kỳ này là có thể vượt ra ngoài thứ tự lặp lại của các bước gia công. Tiện có thể rất phức tạp và các hệ điều khiển hiện đại phản ánh yêu cầu đó. Bạn không chỉ có thể lập trình tiện thẳng, côn mà còn cả bo tròn góc, vạt góc, gia công rãnh, gia công mặt lõm,..., do nhiều chu kỳ còn được dùng để gia công biên dạng. Có thể áp dụng cả bù bán kính mũi dao, nếu có yêu cầu.

Chu kỳ lặp nhiều lần, yêu cầu cần bộ nhớ máy tính, do đó các máy NC cũ, điều khiển bằng băng đục lỗ, *không thể* sử dụng các chu kỳ này. Bộ điều khiển CNC có thể đọc, đánh giá, và xử lý thông tin lưu trong bộ nhớ theo cả hai chiều, thuận và ngược, vào mọi thời điểm.

Khái quát

Nói chung hiện có bảy chu kỳ lặp khả dụng, được nhận biết theo địa chỉ lệnh chuẩn G:

Các chu kỳ gia công biên dạng – Thô

G71	Chu kỳ gia công thô – Chủ yếu theo chiều ngang
G72	Chu kỳ gia công thô – Chủ yếu theo chiều đứng
G73	Chu kỳ gia công thô theo mẫu lặp

Các chu kỳ gia công biên dạng – Tinh

G70	Chu kỳ gia công tinh đối với G71, G72, G73

Các chu kỳ bẻ gãy phoi

G74	Chu kỳ khoan lỗ nhiều lần theo trục Z - ngang
G75	Chu kỳ gia công rãnh nhiều lần theo trục X - đứng

Chu kỳ gia công ren

G76	Chu kỳ tiện ren – thẳng (trụ) hoặc côn

Chu kỳ tiện ren G76 được trình bày chi tiết trong Chương 37.

Các kiểu định dạng chu kỳ

Từng chu kỳ đều có các nguyên tắc riêng, có những điều *làm được* và *không làm được*. Phần kế tiếp sẽ trình bày chi tiết các chu kỳ lặp, ngoại trừ chu kỳ G76, được đề cập trong Chương 37.

Điều quan trọng cần chú ý là định dạng lập trình cho các chu kỳ này, phương pháp nhập dữ liệu, *hơi khác* đối với các bộ điều khiển Fanuc mức thấp, chẳng hạn 0T hoặc 16/18/20/21T, so với các bộ mức cao, chẳng hạn 10/11T hoặc 15T. Các chu kỳ này, nếu khả dụng trên bộ điều khiển mức thấp, đòi hỏi định dạng lập trình trong *hai block*, thay vì *một block* bình thường. Bạn hãy kiểm tra các xác lập tham số đối với từng bộ điều khiển để tìm các giải pháp tương thích. Chương này nêu rõ cả hai loại định dạng.

Các chu kỳ cắt gọt và biên dạng chi tiết

Có lẽ các chu kỳ lặp nhiều lần thông dụng nhất trong tiện và doa là loại *gia công cắt gọt biên dạng*. Có ba chu kỳ khả dụng trong nhóm gia công thô:

- G71, G72 và G73

và một chu kỳ gia công tinh:

- G70

Chu kỳ gia công tinh được thiết kế để gia công hoàn tất biên dạng đã được cắt gọt bằng *một* trong ba chu kỳ gia công thô.

Có một tình huống lý thú trong lập trình các chu kỳ lặp nhiều lần. Nói chung, cần lập trình gia công thô *trước* gia công tinh. Đây là cách thức logic duy nhất theo quan điểm *công nghệ*.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tuy nhiên, "nguyên tắc" này bị vi phạm khi thực hiện các tính toán trên máy tính. Ý nghĩa ở đây là khi lập trình ba chu kỳ lặp nhiều lần này, *biên dạng hoàn tất luôn luôn được xác định trước*, sau đó các yêu cầu gia công mới được áp dụng cho chu kỳ gia công thô. Thoạt nhìn, bạn sẽ cảm thấy ngạc nhiên, nhưng khi làm việc lâu hơn với các chu kỳ đó, sẽ dễ dàng nhận thấy đây là phương pháp thông minh.

Các chu kỳ bẻ gãy phoi

Hai chu kỳ "bẻ gãy phoi" được thiết kế để tạo ra sự *cắt gọt gián đoạn* dọc theo trục Z (G75), hoặc trục X (G74). Trong thực tế, chu kỳ G74 cung cấp nhiều ứng dụng thực tiễn hơn G75. Chu kỳ G74 cho phép khoan gián đoạn trên máy tiện CNC. Mặc dù yêu cầu khoan gián đoạn trên máy tiện ít hơn trên trung tâm gia công, nhưng vẫn được sử dụng trong thực tế. Tuy nhiên, chu kỳ G75, thực sự là chu kỳ gia công rãnh "gián đoạn", ít được sử dụng, do *không tạo ra rãnh chính xác*.

CÁC CHU KỲ CẮT GỌT BIÊN DẠNG

Cho đến nay, các chu kỳ cắt gọt biên dạng, là loại thông dụng nhất trong lập trình tiện CNC. Chúng được dùng để tiện ngoài, tiện trong (doa), dọc theo hầu như mọi biên dạng bất kỳ có thể gia công được.

Định nghĩa đường bao

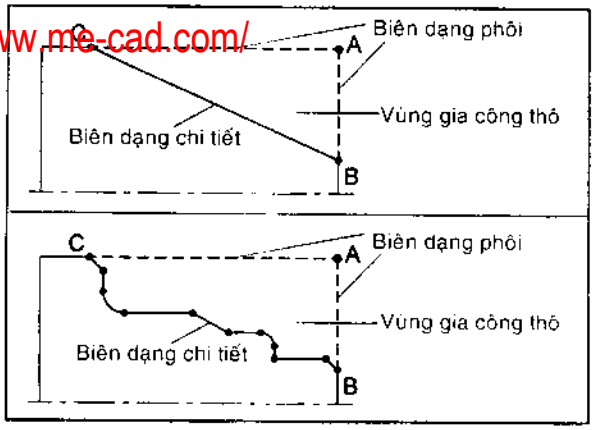
Các chu kỳ gia công thô dựa trên định nghĩa *hai* đường bao, thường được gọi là đường bao *vật liệu*, thực chất là biên dạng phôi, và đường bao *chi tiết*, biên dạng của chi tiết. Đây hoàn toàn không phải là khái niệm mới, nhiều ngôn ngữ lập trình cũ đã sử dụng phương pháp này, chẳng hạn Compact II, ngôn ngữ rất phổ biến trên hệ thống lập trình trong các năm 1970.

Hai đường bao này tạo ra vùng hoàn toàn khép kín xác định phần vật liệu dư. Từ vùng đó, vật liệu được cắt gọt theo cách thức có thứ tự, tuân theo các tham số gia công chuyên biệt trong (các) block gọi chu kỳ. Về toán học, số điểm tối thiểu có thể xác định diện tích phẳng là ba.

Ba điểm này phải không thẳng hàng. Hình 34.10 minh họa đường bao đơn giản chỉ với ba điểm và đường bao gồm nhiều điểm.

Trong các chu kỳ cắt gọt biên dạng, từng điểm đại diện cho vị trí dao, các điểm A, B và C là đỉnh các góc của diện tích gia công đã chọn.

Đường bao vật liệu (phôi) không cần xác định một cách thực sự, chỉ có tính ngụ ý, giữa



Hình 34.10. *Đường bao vật liệu và chi tiết khi áp dụng để tiện*

các điểm AB và AC. Đường bao vật liệu (phôi) *không thể có* các điểm khác; phải là đường thẳng nhưng không nhất thiết song song với một trong các trục.

Đường bao chi tiết được xác định giữa các điểm B và C, *có thể có* một số bất kỳ các điểm xem giữa hai điểm đó. Đối với lập trình CNC, có thể dùng các ký hiệu điểm khác nhau, thay vì các điểm A B C như trên Hình 34.10.

Điểm khởi đầu và các điểm P và Q

Điểm A trong minh họa là *điểm đầu* của chu kỳ gia công biên dạng, có thể định nghĩa đơn giản như sau:

Điểm bắt đầu được định nghĩa là vị trí tọa độ XZ cuối cùng, trước khi gọi chu kỳ gia công biên dạng.

Nói chung, điểm bắt đầu sẽ gần sát nhất với góc chi tiết, nơi bắt đầu gia công thô. Điều quan trọng là chọn điểm bắt đầu một cách rất cẩn thận, do đây "không chỉ là điểm bắt đầu". Điểm đặc biệt này kiểm soát mọi khoảng hở tiếp cận và chiều sâu thực của đường cắt gọt thô *thứ nhất*.

Các điểm B và C trong minh họa trở thành P và Q trong chương trình:

Điểm P đại diện cho số block của tọa độ XZ thứ nhất trên biên dạng hoàn tất

Điểm Q đại diện cho số block của XZ cuối cùng trên biên dạng hoàn tất.

Các vấn đề cần chú ý đối với các điểm biên dạng P và Q là rất quan trọng:

- Số lượng các điểm có thể được xác định giữa hai điểm P và Q, đại diện các tọa độ XZ của biên dạng hoàn tất. Biên dạng này được lập trình sử dụng các chuyển động dao G01, G02, và G03, cùng với tốc độ cắt.
- Lượng dư gia công được xác định bằng điểm bắt đầu và biên dạng P-Q phải gộp mọi khoảng hở cần thiết.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

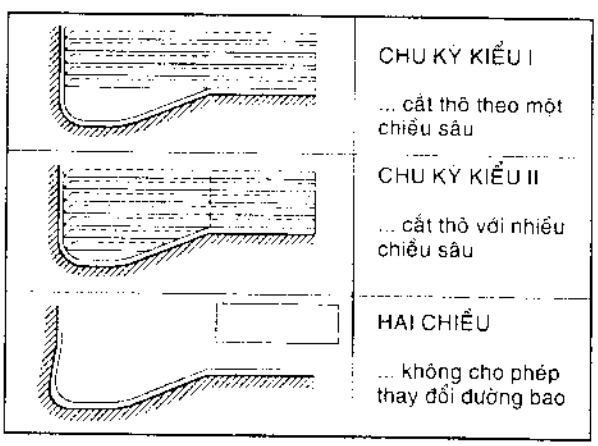
Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Bù bán kính mũi dao không cần gọi giữa các điểm P và Q, nhưng phải lập trình trước khi gọi chu kỳ, thường trong khi chuyển động đến điểm khởi đầu.
- Đối với gia công thô, vật liệu cần cắt gọt sẽ được chia thành chuỗi các đường cắt. Chu kỳ gia công thô chấp nhận các tham số cắt gọt do người dùng cung cấp.
- Vì các lý do an toàn, đường kính điểm khởi đầu phải ở *phía trên* đường kính phôi khi tiện ngoài và *phía dưới* đường kính lỗi khi tiện (doa) trong.
- Chuyển động dao giữa điểm P và điểm Q phải tăng đều khi tiện ngoài và giảm đều khi tiện trong.
- Chỉ cho phép đổi chiều giữa các điểm P và Q nếu chu kỳ Kiểu II khả dụng và được lập trình, và sau đó chỉ theo một chiều.
- Các block đại diện cho tọa độ XZ thứ nhất của biên dạng P, và tọa độ XZ cuối cùng của biên dạng Q, phải có số chuỗi thứ tự N không trùng lặp ở nơi bất kỳ trong chương trình.

CHU KỲ KIỂU I VÀ KIỂU II

Trong các phiên bản ban đầu của chu kỳ gia công biên dạng, sự thay đổi chiều biên dạng sang chiều *ngược lại* dọc theo một trục là không được phép. Các giới hạn này làm cho chu kỳ gia công biên dạng không thể áp dụng cho các mặt lõm trong chương trình, dù gia công mặt lõm khá phổ biến trong xưởng cơ khí.

Ngày nay, phương pháp cũ này được gọi là chu kỳ lặp *Kiểu I*. Các bộ điều khiển hiện đại sử dụng tính năng phần mềm cao cấp, cho phép thay đổi theo *một* chiều. Phương pháp này được gọi là *Kiểu II*, làm tăng tính linh hoạt trong lập trình khi cắt gọt các hốc hoặc mặt lõm. Hình 34.11 so sánh hai kiểu này và minh họa sự cấm thay đổi biên dạng theo hai chiều bên trong chu kỳ. Ví dụ này áp dụng chu kỳ G71 cắt gọt ngoài, nhưng có thể chỉnh sửa để cắt gọt trong.



Hình 34.11. *So sánh các chu kỳ Kiểu I và Kiểu II – không được phép thay đổi hai chiều dọc theo hai trục.*

Kiểu I cho phép biên dạng *tăng* ổn định (tiện ngoài) hoặc biên dạng *giảm* ổn định (cắt gọt trong) từ điểm P đến điểm Q (các chiều cắt gọt thông dụng). Trên các bộ điều khiển cũ, đảo chiều X hoặc Z là không được phép. Các bộ điều khiển hiện đại cho phép cắt gọt lõm với kiểu I nhưng chỉ với một đường cắt. Điều đó có thể gây ra lượng cắt gọt quá lớn trong một số trường hợp. Bạn cần biết chính xác hệ điều khiển hỗ trợ kiểu chu kỳ nào.

Kiểu II cho phép liên tục tăng biên dạng hoặc liên tục giảm biên dạng từ điểm P đến Q. Sự đổi sang chiều ngược lại chỉ được phép *trên một trục*, tùy theo chu kỳ hoạt động. Gia công thô hốc lõm có thể thực hiện với nhiều đường cắt. *Kiểu I* và *Kiểu II* đều áp dụng được cho chu kỳ, bằng cách lập trình *cả hai trục* trong block có điểm P. Đây thường là block ngay sau khi gọi chu kỳ trong chương trình (sau G71, G72,...)

Lập trình các chu kỳ Kiểu I và Kiểu II

Nếu hệ điều khiển hỗ trợ *Kiểu II* trong các chu kỳ tiện và doa, cũng sẽ hỗ trợ *Kiểu I*, nếu cần dùng Kiểu I trong một số ứng dụng đặc biệt. Điều đó có nghĩa là Fanuc không thay đổi kiểu này với kiểu khác, chỉ bổ sung *Kiểu II*. Câu hỏi sẽ là – làm thế nào để phân biệt giữa hai Kiểu trong chương trình? Điều cơ bản để chọn Kiểu là nội dung của block *ngay sau khi gọi chu kỳ*:

- Kiểu I ...*chỉ chuyên biệt một trục*
- Kiểu II ...*chuyên biệt hai trục*

➲ Ví dụ - Kiểu I

```
G71 U.. R..
G71 P10 Q.. U.. W.. F.. S..
N10 G00 X..          (MỘT TRỤC ĐỐI VỚI KIỂU I)
...
```

➲ Ví dụ - Kiểu II

```
G71 U.. R..
G71 P10 Q.. U.. W.. F.. S..
N10 G00 X.. Z..      (HAI TRỤC ĐỐI VỚI KIỂU II)
...
```

Nếu không có chuyển động theo trục Z trong block thứ nhất sau khi gọi chu kỳ và vẫn cần chu kỳ Kiểu II, bạn hãy lập trình W0 làm trục thứ hai.

Định dạng chu kỳ

Phần kế tiếp sẽ trình bày sáu chu kỳ tiện. Điều quan trọng là kiểu định dạng của từng chu kỳ khi áp dụng cho bộ điều khiển cụ thể. Nhiều bộ điều khiển Fanuc cho phép lập trình các chu kỳ lặp, chúng có thể chia thành hai nhóm:

- Hệ thống Fanuc 0T, 16T, 18T, 20T, 21T ...*mức thấp*
- Hệ thống Fanuc 10T,11T, 15T ...*mức cao*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trong thực tế điều này chỉ có nghĩa là sự thay đổi cách thức lập trình chu kỳ, nhưng cũng rất quan trọng để giải quyết một số vấn đề không tương thích. Chú ý, hàm dụng cụ cắt T không được chuyên biệt trong các ví dụ, mặc dầu vẫn được phép là tham số trong tất cả các chu kỳ lặp, do chỉ cần thiết để thay đổi chế độ bù dao.

G71 – CHU KỲ TIỆN

Chu kỳ tiện thô thông dụng nhất là G71. Mục đích của G71 là tiện thô theo chiều *ngang*, chủ yếu theo trục Z, thường từ phải sang trái. Chu kỳ này thường tiện thô phôi trục đặc. Tương tự mọi chu kỳ khác, G71 có hai định dạng – một block và hai block, tùy theo hệ điều khiển.

Định dạng chu kỳ G71 – 10T/11T/15T

Định dạng một – block của chu kỳ G71:

```
G71 P.. Q.. I.. K.. U.. W.. D.. F.. S..
```

Trong đó:

- P = Số block thứ nhất của biên dạng hoàn tất
- Q = Số block cuối cùng của biên dạng hoàn tất
- I = Khoảng cách và chiều gia công thô và bán tinh trên trục X – theo một phía
- K = Khoảng cách và chiều gia công thô và bán tinh trên trục Z – theo một phía
- U = Lượng dư để gia công tinh trên đường kính trục X.
- W = Lượng dư để gia công tinh trên trục Z
- D = Chiều sâu cắt thô
- F = Tốc độ cắt (in/rev hoặc mm/rev) vượt qua tốc độ cắt giữa block P và block Q
- S = Tốc độ trục chính (ft/min hoặc m/min) vượt qua tốc độ trục chính giữa block P và block Q

Các tham số I và K không khả dụng trên tất cả các máy tiện. Chúng điều khiển lượng cắt gọt bán tinh, đường cắt liên tục cuối cùng trước khi các chuyển động cắt thô hoàn tất.

Định dạng chu kỳ G71 – 0T/16T/18T/20T/21T

Nếu bộ điều khiển yêu cầu mục nhập hai – block cho chu kỳ G71, định dạng lập trình sẽ là:

```
G71 U.. R..
G71 P.. Q.. U.. W.. F.. S..
```

Trong đó:

Block thứ nhất:

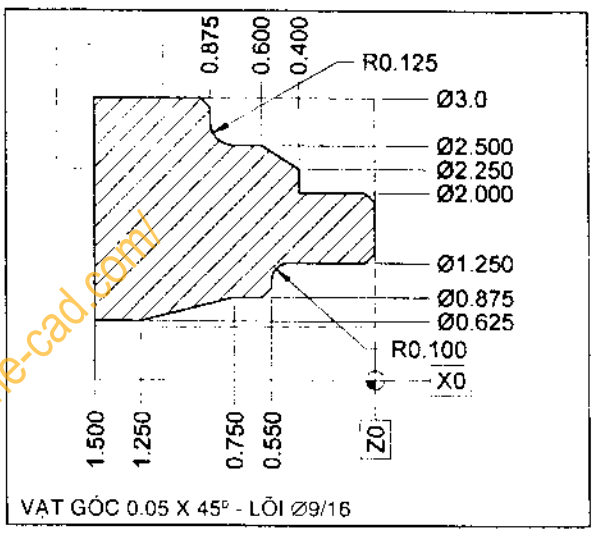
- W = chiều sâu cắt thô
- R = lượng lùi dao từ từng đường cắt

Block thứ hai:

- P = Số block thứ nhất của biên dạng hoàn tất
- Q = Số block cuối cùng của biên dạng hoàn tất
- U = Lượng dư để gia công tinh trên đường kính trục X.
- W = Lượng dư để gia công tinh trên trục Z
- F = Tốc độ cắt (in/rev hoặc mm/rev) vượt qua tốc độ cắt giữa block P và block Q
- S = Tốc độ trục chính (ft/min hoặc m/min) vượt qua tốc độ trục chính giữa block P và block Q

Bạn không nên nhầm U trong block thứ nhất, chiều sâu cắt theo một phía, và U trong block thứ hai, lượng dư trên đường kính. Các tham số I và K có thể chỉ có trên một số bộ điều khiển và lượng lùi dao R được xác lập theo tham số hệ thống.

Ứng dụng tiện trong và tiện ngoài của chu kỳ G71 sẽ sử dụng dữ liệu trên Hình 34.12.



Hình 34.12. Bản vẽ minh họa chu kỳ gia công thô G71 – Chương trình O3405

G71 tiện thô mặt ngoài

Phôi trong ví dụ này có sẵn lỗ Ø9/16 (.5625). Để tiện ngoài, bạn dùng dao tiêu chuẩn 80^0 với một lần cắt mặt đầu (vạt mặt), và tiện thô biên dạng ngoài.

Chương trình O3405 gồm các nguyên công nêu trên.

```
O3405   (G71 ROUGHING CYCLE - ROUGHING ONLY)
N1 G20
N2 T0100 M41          (OD ROUGHING TOOL + GEAR)
N3 G96 S450 M03     (SPEED FOR ROUGH TURNING)
N4 G00 G41 X3.2 Z0 T0101 M08 (START FOR FACE)
N5 G01 X0.36                  (END OF FACE DIA)
N6 G00 Z0.1                   (CLEAR OFF FACE)
N7 G42 X3.1         (START POSITION FOR CYCLE)
N8 G71 P9 Q17 U0.06 W0.004 D1250 F0.014
N9 G00 X1.7       (P POINT = START OF CONTOUR)
N10 G01 X2.0 Z-0.05 F0.005
N11 Z-0.4 F0.01
N12 X2.25
N13 X2.5 Z-0.6
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N14 Z-0.875 R0.125
N15 X2.0
N16 G01 X3.05 Z-0.95
N17 U0.2 F0.02        (Q POINT = END OF CONTOUR)
N18 G00 G40 X5.0 Z6.0 T0100
N19 M01
```

Tiện thô mặt ngoài hoàn tất tại điểm này trong chương trình và có thể lập trình gia công thô mặt trong cho dao kế tiếp. Trong tất cả các ví dụ có thay dao giữa dao ngắn (chẳng hạn dao tiện) và dao dài (chẳng hạn thanh doa), điều quan trọng là dịch chuyển dao ngắn ra xa mặt đầu. Chuyển động này phải đủ xa để bảo đảm cho dao dài tiến vào. Khoảng hở là 6.0 inch trong ví dụ nêu trên (block N18 với Z6.0).

G71 tiện thô mặt trong

Mặc đầu đã được gia công với dao trong phần trước, thanh doa thô có thể tiếp tục gia công:

```
N20 T0300                          (ID ROUGHING TOOL)
N21 G96 S400 M03       (SPEED FOR ROUGH BORING)
N22 G00 G41 X0.5 Z0.1 T0303 M08  (START POS.)
N23 G71 P24 Q31 U-0.06 W0.004 D1000 F0.012
N24 G00 X1.55      (P POINT = START OF CONTOUR)
N25 G01 X1.25 Z-0.05 F0.004
N26 Z-0.55 R-0.1 F0.008
N27 X0.875 K-0.05
N28 Z-0.75
N29 X0.625 Z-1.25
N30 Z-1.55
N31 U-0.2 F0.02      (Q POINT = END OF CONTOUR)
N32 G00 G40 X5.0 Z2.0 T0300
N33 M01
```

Chi tiết đã gia công thô xong, để lại lượng dư cần thiết trên các đường kính, mặt đầu, và bậc trụ để gia công tinh. Có thể gia công tinh với chu kỳ G70, sử dụng cùng một dao như gia công thô, nếu yêu cầu về dung sai và độ bóng bề mặt không quá cao. Nếu có yêu cầu cao, cần dùng (các) dao khác trong chương trình, sau khi thay dao.

Tại giai đoạn này, bạn hãy đánh giá những điều đã thực hiện và lý do. Nhiều nguyên lý áp dụng cho ví dụ nêu trên cũng rất thông dụng cho các nguyên công khác sử dụng các chu kỳ lặp. Điều quan trọng là nắm vững các nguyên tắc đó.

Chiều cắt trong G71

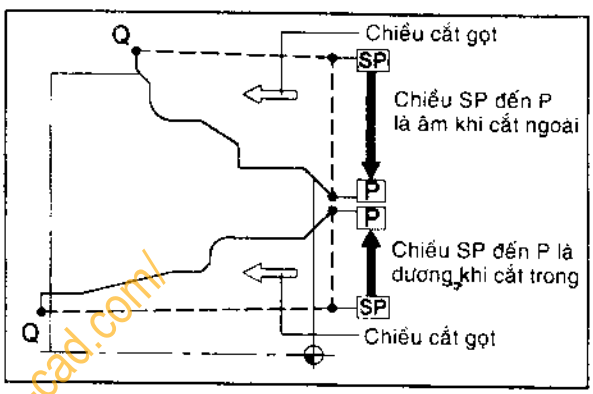
Ví dụ lập trình O3405, cho thấy G71 có thể dùng cho gia công thô *trong* và *ngoài*. Có hai khác biệt quan trọng:

- Điểm bắt đầu liên quan với điểm P (SP đến P và P đến SP).
- Dấu của địa chỉ U đối với lượng dư trên đường kính.

Hệ điều khiển sẽ xử lý chu kỳ để gia công *ngoài*, nếu chiều X từ điểm bắt đầu SP đến điểm P là *âm*. Trong ví dụ này, điểm bắt đầu là X3.1, điểm P là X1.7. Chiều X là *âm* hoặc giảm, thực hiện sự cắt gọt *ngoài*.

Hệ điều khiển sẽ xử lý chu kỳ để gia công *trong*, nếu chiều X từ điểm bắt đầu SP đến điểm P là *dương*. Trong ví dụ này, điểm bắt đầu là X0.5, điểm P là X1.55. Chiều X là *dương* hoặc tăng, thực hiện sự cắt gọt *trong*.

Hình 34.13 minh họa khái niệm của chu kỳ G71 khi áp dụng cho gia công trong và gia công ngoài. Mặc dầu *dấu* của giá trị lượng dư U là rất quan trọng đối với kích cỡ hoàn tất của chi tiết, nhưng *không* xác định chế độ cắt gọt. Điều này kết thúc phần liên quan đến chu kỳ lặp G71. Chu kỳ gia công thô mặt đầu G72 có nhiều điểm tương đồng, sẽ được trình bày kế tiếp.



Hình 34.13. Cắt trong và ngoài với chu kỳ G71

G72 - GIA CÔNG MẶT ĐẦU

Chu kỳ G72 có nhiều điểm tương tự chu kỳ G71 ngoại trừ sự cắt gọt chủ yếu theo chiều *đứng* (vạt mặt), chủ yếu từ đường kính lớn hướng đến đường tâm trục chính X0. Chu kỳ này được dùng để gia công thô phôi trụ đặc, sử dụng chuỗi đường cắt *đứng* (vạt mặt). Cũng như các chu kỳ khác trong nhóm này, G72 có hai định dạng – một block và hai block, tùy theo hệ thống điều khiển. Bạn hãy so sánh G72 với cấu trúc của G71 trong các ví dụ.

Định dạng chu kỳ G72 – 10T/11T/15T

Định dạng lập trình một – block của chu kỳ G72 như sau:

```
G72 P.. Q.. I.. K.. U.. W.. D.. F.. S..
```

Trong đó:

P = Số block thứ nhất của biên dạng hoàn tất
Q = Số block cuối cùng của biên dạng hoàn tất
I = Khoảng cách và chiều gia công thô và bán tinh trên trục X – theo một phía
K = Khoảng cách và chiều gia công thô và bán tinh trên trục Z – theo một phía
U = Lượng dư để gia công tinh trên đường kính trục X.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

W = Lượng dư để gia công tinh trên trục Z
D = Chiều sâu cắt thô
F = Tốc độ cắt (in/rev hoặc mm/rev) vượt qua tốc độ cắt giữa block P và block Q
S = Tốc độ trục chính (ft/min hoặc m/min) vượt qua tốc độ trục chính giữa block P và block Q

Ý nghĩa của từng địa chỉ hoàn toàn như với chu kỳ G71. Các tham số I và K có thể không khả dụng trên một số máy. Các tham số này kiểm soát lượng cắt gọt đối với gia công bán tinh, là đường cắt liên tục cuối cùng trước khi hoàn tất các chuyển động gia công thô.

Định dạng chu kỳ G72 – 0T/16T/18T/20T/21T

Nếu hệ điều khiển yêu cầu mục nhập hai-block cho chu kỳ G72, định dạng lập trình như sau:

```
G72 W.. R..
G72 P.. Q.. U.. W.. F.. S..
```

Trong đó:

Block thứ nhất :

W = chiều sâu cắt thô
R = lượng lùi dao từ từng đường cắt

Block thứ hai:

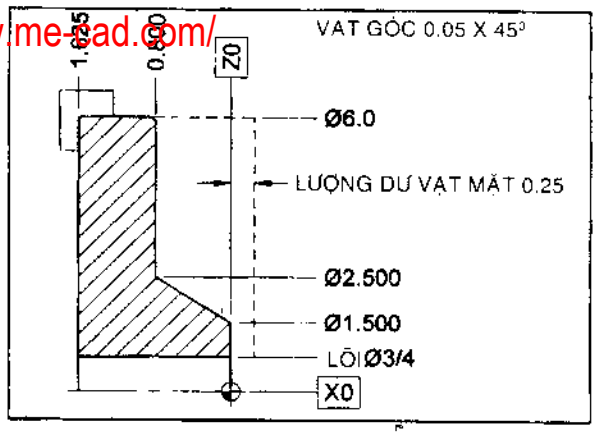
P = Số block thứ nhất của biên dạng hoàn tất
Q = Số block cuối cùng của biên dạng hoàn tất
U = Lượng dư để gia công tinh trên đường kính trục X.
W = Lượng dư để gia công tinh trên trục Z
F = Tốc độ cắt (in/rev hoặc mm/rev) vượt qua tốc độ cắt giữa block P và block Q
S = Tốc độ trục chính (ft/min hoặc m/min) vượt qua tốc độ trục chính giữa block P và block Q

Trong chu kỳ G71 với định dạng hai–block, có hai địa chỉ U. Trong định dạng hai–block của chu kỳ G72 có hai địa chỉ W. Bạn đừng nhầm W trong block thứ nhất, chiều sâu cắt (thực tế là *chiều rộng* cắt), và W trong block thứ hai – lượng dư để lại trên mặt đầu. Các tham số I và K có thể khả dụng, tùy theo bộ điều khiển.

Chương trình ví dụ O3406 với chu kỳ G72 sử dụng dữ liệu bản vẽ trên Hình 34.14.

Trong ứng dụng gia công mặt đầu, mọi dữ liệu chính sẽ được đảo ngược 90^0, do cắt gọt theo trục X. Chương trình gia công thô sử dụng chu kỳ G72 về logic là tương tự chu kỳ G71:

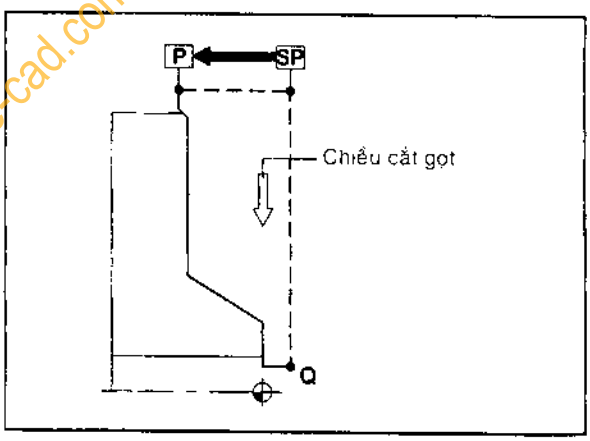
```
O3406 (G72 ROUGHING CYCLE - ROUGHING ONLY)
N1 G20
N2 T0100 M41              (OD FACING TOOL + GEAR)
N3 G96 S450 M03        (SPEED FOR ROUGH FACING)
N4 G00 G41 X6.25 Z0.3 T0101 M08  (START POS.)
N5 G72 P6 Q12 U0.06 W0.03 D1250 F0.014
N6 G00 Z-0.875   (P-POINT = START OF CONTOUR)
N7 G01 X6.05 F0.02
N8 X5.9 Z-0.8 F0.008
```



Hình 34.14. Bản vẽ minh họa chu kỳ gia công thô G72 – chương trình O3406

```
N9 X2.5
N10 X1.5 Z0
N11 X0.55
N12 W0.1 F0.02      (Q-POINT = END OF CONTOUR)
N13 G00 G40 X8.0 Z3.0 T0100
N14 M01
```

Khái niệm của chu kỳ G72 được minh họa trên Hình 34.15. Chú ý vị trí của điểm P khi liên quan với điểm bắt đầu SP và so sánh với chu kỳ G71.



Hình 34.15. Khái niệm cơ bản của chu kỳ lặp G72

G73 – CHU KỲ LẶP GIA CÔNG THEO MẪU CHUẨN

Chu kỳ lặp theo mẫu chuẩn còn được gọi là chu kỳ *Vòng kín* hoặc chu kỳ *Sao chép Biên dạng*. Mục đích của chu kỳ này là giảm đến mức thấp nhất thời gian cắt gọt thô các hình dạng không đều, chẳng hạn vật đúc, vật rèn, ...

Định dạng chu kỳ G73 – 10T/11T/15T

Định dạng lập trình một – block đối với chu kỳ G73 tương tự các ch kỳ G71 và G72:

```
G73 P... Q... I... K... U... W... D... F... S...
```

Trong đó:
P = Số block thứ nhất của biên dạng hoàn tất

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Q = Số block cuối cùng của biên dạng hoàn tất
I = Khoảng cách X và chiều phần bo tròn – theo một phía.
K = Khoảng cách trục Z và chiều phần bo tròn.
U = Lượng dư để gia công tinh trên đường kính trục X.
W = Lượng dư để gia công tinh trên trục Z
D = Số lượng phần chia cắt gọt.
F = Tốc độ cắt (in/rev hoặc mm/rev) vượt qua tốc độ cắt giữa block P và block Q
S = Tốc độ trục chính (ft/min hoặc m/min) vượt qua tốc độ trục chính giữa block P và block Q

Định dạng chu kỳ G73 – 0T/18T/18T/20T/21T

Nếu hệ điều khiển yêu cầu mục nhập hai – block cho chu kỳ G73, định dạng lập trình như sau:

```
G73 U.. W.. R..
G73 P.. Q.. U.. W.. F.. S..
```

Trong đó:

Block thứ nhất:

U = Khoảng cách và chiều trục X phần bo tròn – theo một phía
W = Khoảng cách và chiều trục Z phần bo tròn
R = Số lượng phần chia cắt gọt

Block thứ hai:

P = Số block thứ nhất của biên dạng hoàn tất
Q = Số block cuối cùng của biên dạng hoàn tất
U = Lượng dư để gia công tinh trên đường kính trục X.
W = Lượng dư để gia công tinh trên trục Z
F = Tốc độ cắt (in/rev hoặc mm/rev) vượt qua tốc độ cắt giữa block P và block Q
S = Tốc độ trục chính (ft/min hoặc m/min) vượt qua tốc độ trục chính giữa block P và block Q

Trong các mục nhập hai–block, bạn không nên nhầm lẫn giữa các địa chỉ trong block thứ nhất và trong block thứ hai (U và W trong ví dụ G73). *Chúng có ý nghĩa khác nhau.*

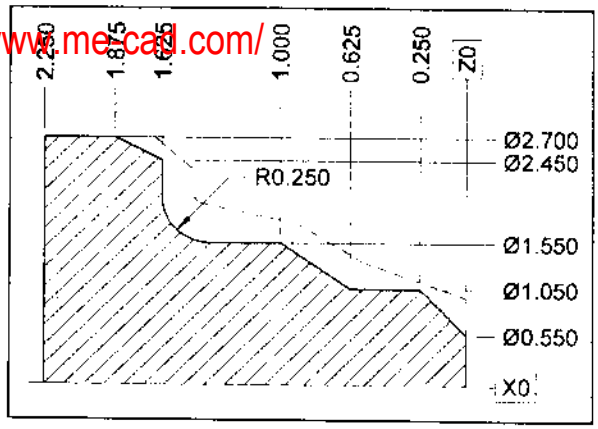
Ví dụ về chu kỳ lặp G73

Ví dụ chương trình chu kỳ lặp G73 sử dụng bản vẽ trên Hình 34.16.

Có ba tham số nhập quan trọng trong chu kỳ G73 – I, K và D. Một tham số đường như bị thiếu – *không có sự chuyên biệt chiều sâu cắt.* Trong chu kỳ G73, điều đó là không cần thiết. Chiều sâu cắt thực được tính toán một cách tự động, dựa trên ba tham số:

- I ... lượng cắt gọt thô theo trục X
- K ... lượng cắt gọt thô theo trục Z
- D ... số lần cắt gọt hoặc số lần lặp lại

Bạn hãy dùng chu kỳ này một cách cẩn thận – chu kỳ được thiết kế với giả thiết *lượng cắt gọt bằng nhau* dọc theo cả trục X và trục Z. Điều này ít xảy ra trong thực tế đối với vật đúc và vật rèn, chúng thường có lượng dư *khác nhau* trên toàn bộ phôi (Hình 34.16). Vẫn có thể sử dụng chu kỳ với hiệu suất hợp lý, nhưng có thể có hậu quả xấu đối với chi tiết có hình dạng không đều.



Hình 34.16. Chu kỳ lặp G73 – chương trình O3407

Trong ví dụ này, lượng cắt gọt *mong muốn lớn nhất* theo một phía sẽ được chọn là .200 (I0.2) và lượng dư *mong muốn lớn nhất* trên mặt đầu là .300 (R0.3). Số lần cắt có thể là hai hoặc ba, do đó chương trình sẽ sử dụng D3. Có thể cần điều chỉnh trên bộ điều khiển trong khi xác lập hoặc gia công, tùy theo điều kiện và kích cỡ cụ thể của phôi đúc hoặc rèn.

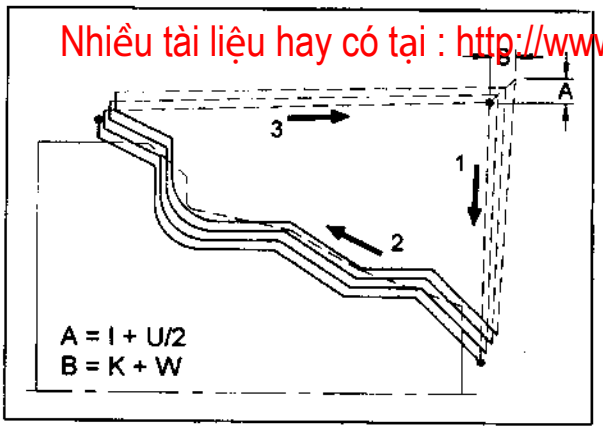
Chu kỳ này thích hợp để gia công các biên dạng khi biên dạng hoàn tất gần như tương hợp với biên dạng phôi rèn hoặc đúc. Ngay cả khi có sự cắt gọt "không tải" nào đó, chu kỳ này vẫn có thể hiệu quả hơn so với G71 hoặc G72. Chương trình O3407 minh họa gia công thô và gia công tinh với cùng một dao (chỉ có tính ví dụ).

```
O3407 (G73 PATTERN REPEATING CYCLE)
N1 G20 M42
N2 T0100
N3 G96 S350 M03
N4 G00 G42 X3.0 Z0.1 T0101 M08
N5 G73 P6 Q13 I0.2 K0.3 U0.06 W0.004 D3 F0.01
N6 G00 X0.35
N7 G01 X1.05 Z-0.25
N8 Z-0.625
N9 X1.55 Z-1.0
N10 Z-1.625 R0.25
N11 X2.45
N12 X2.75 Z-1.95
N13 U0.2 F0.02
N14 G70 P6 Q13 F0.006
N15 G00 G40 X5.0 Z2.0 T0100
N16 M30
%
```

Số lần cắt D3 có thể cần thiết để hỗ trợ độ lệch tâm quay, thường liên quan với phôi đúc hoặc rèn. Mặt khác, D2 có thể cần thiết đối với

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 34.17. Biểu diễn sơ đồ ứng dụng chu kỳ G73.

sự cắt gọt nặng, nhằm cải thiện tuổi bền dụng cụ cắt. Hình 34.17 minh họa sơ đồ lập trình ba đường cắt (D3).

Chú ý, chu kỳ này lặp lại sự gia công biên dạng một cách chính xác được chuyên biệt giữa các điểm P và Q. Từng quỹ đạo dao riêng rẽ đều được bù theo giá trị tính toán dọc theo các trục X và Z. Trên máy, bạn hãy quan sát tiến độ cắt gọt một cách cẩn thận – đặc biệt đối với quỹ đạo dao thứ nhất. Ở đây, sự vượt qua tốc độ cắt có thể hữu ích.

G70 - CHU KỲ GIA CÔNG TINH BIÊN DẠNG

Chu kỳ gia công biên dạng cuối cùng là G70. Tuy chu kỳ này có số mã G nhỏ hơn so với ba chu kỳ gia công thô, G71, G72, G73, nhưng thường được dùng *sau* các chu kỳ đó. G70 chỉ được dùng để *gia công tinh biên dạng trước đó đã được gia công thô.*

Định dạng chu kỳ G70 – tất cả các bộ điều khiển

Đối với chu kỳ này, không có sự khác biệt về định dạng lập trình giữa các bộ điều khiển – tất cả đều như nhau, gọi chu kỳ với lệnh một–block.

Định dạng lập trình của chu kỳ G70 :

```
G70 P.. Q.. F.. S..
```

Trong đó:

P = Số block thứ nhất của biên dạng hoàn tất
Q = Số block cuối cùng của biên dạng hoàn tất
F = Tốc độ cắt (in/rev hoặc mm/rev)
S = Tốc độ trục chính (ft/min hoặc m/min)

Chu kỳ G70 chấp nhận biên dạng hoàn tất được xác định trước đó từ một trong ba chu kỳ gia công thô. Biên dạng hoàn tất này được xác định theo P và Q của các chu kỳ lặp, và thường được lặp lại trong chu kỳ G70, dù có thể thay đổi.

Để an toàn, bạn hãy dùng cùng điểm khởi đầu cho G70 như đối với các chu kỳ gia công thô.

Chương trình gia công thô O3405, sử dụng chu kỳ lặp G71 để tiện thô ngoài và trong, có thể hoàn tất bằng cách dùng thêm hai dao khác, một để gia công tinh mặt ngoài, một để gia công tinh lỗ (mặt trong):

```
(O3405 TIẾP TỤC...)
....
N34 T0500 M42        (OD FINISHING TOOL + GEAR)
N35 G96 S530 M03   (SPEED FOR FINISH TURNING)
N36 G42 X3.1 Z0.1 T0505 M08       (START POS.)
N37 G70 P9 Q17           (FINISHING CYCLE - OD)
N38 G00 G40 X5.0 Z6.0 T0500
N39 M01

N40 T0700                    (ID FINISHING TOOL)
N41 G96 S475 M03      (SPEED FOR ROUGH BORING)
N42 G00 G41 X0.5 Z0.1 T0707 M08  (START POS.)
N43 G70 P24 Q31          (FINISHING CYCLE - ID)
N44 G00 G40 X5.0 Z2.0 T0700
N45 M30                            (END OF PROGRAM)
%
```

Ngay cả đối với gia công tinh mặt ngoài, dao cắt vẫn được lập trình để khởi động *phía trên* đường kính phôi ban đầu, dù đã hoàn tất sự gia công thô. Phương pháp này cũng áp dụng cho tiện trong. Vì các lý do an toàn, đây là kinh nghiệm nên áp dụng.

Trong chu kỳ G70, không cần lập trình tốc độ cắt, dù định dạng chu kỳ chấp nhận tốc độ cắt. Các block từ P đến Q đối với dao cắt thô đã có các tốc độ cắt. Các tốc độ cắt này sẽ bỏ qua trong chế độ gia công thô và sẽ có hiệu lực chỉ trong chu kỳ G70 khi gia công tinh. Nếu biên dạng hoàn tất không có tốc độ cắt, bạn hãy lập trình *tốc độ cắt chung* cho gia công tinh tất cả các biên dạng trong khi xử lý chu kỳ G70. Ví dụ, block chương trình

```
M37 G70 P9 Q17 F0.007
```

sẽ là sự lãng phí thời gian, do tốc độ cắt .007 in/rev sẽ *không* được sử dụng. Tốc độ cắt đã xác định giữa các block N9 và N17 của chương trình O3405 sẽ vượt qua tốc độ này. Mặt khác, nếu hoàn toàn không có tốc độ cắt được lập trình để gia công tinh biên dạng, thì:

```
N.. G70 P.. Q.. F0.007
```

sẽ sử dụng .007 in/rev cho quỹ đạo dao gia công tinh.

Logic áp dụng cho chu kỳ G71, cũng áp dụng cho chu kỳ G72. Chương trình gia công thô O3406, dùng để tiện thô mặt đầu của chi tiết, có thể hoàn tất bằng cách dùng một dao khác để gia công tinh mặt này với chu kỳ G70:

```
(O3406 TIẾP TỤC...)
....
N15 T0500 M42          (OD FACING TOOL + GEAR)
N16 G96 S500 M03    (SPEED FOR FINISH FACING)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N17 G00 G41 X6.25 Z0.3 T0505 M08 (START POS.)
N18 G70 P10 Q20                  (FINISHING CYCLE)
N19 G00 G40 X8.0 Z3.0 T0500
N20 M30
%
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Các nguyên tắc đã trình bày ở phần trên cũng áp dụng cho gia công tinh biên dạng được xác định từ chu kỳ G72 chương trình O3407, sử dụng chu kỳ G73 cũng có thể được lập trình bằng cách sử dụng dao khác để gia công tinh, áp dụng các nguyên tắc tương tự chu kỳ G72.

NGUYÊN TẮC CƠ BẢN CỦA CÁC CHU KỲ G70-G73

Để giúp cho các chu kỳ lặp gia công biên dạng (thô và tinh) hoạt động ổn định và hiệu quả, bạn cần nắm vững các nguyên tắc cơ bản của chúng. Một sơ suất hoặc nhầm lẫn nhỏ cũng có thể làm cho bạn tốn nhiều thời gian chỉnh sửa.

Dưới đây là một số nguyên tắc quan trọng:

- ❑ Luôn luôn áp dụng bù bán kính mũi dao trước khi gọi chu kỳ lặp.
- ❑ Luôn luôn xóa bù bán kính mũi dao sau khi hoàn tất chu kỳ lặp.
- ❑ Chuyển động trở về vị trí ban đầu là tự động, do đó không cần lập trình.
- ❑ Block P trong G71 phải không có giá trị Z (Z hoặc W) đối với chu kỳ Kiểu I
- ❑ Sự đổi chiều chỉ được phép đối với chu kỳ G71 kiểu II, và chỉ theo một trục (W0).
- ❑ Lượng dư U được lập trình trên đường kính, dấu của U biểu thị phía của phôi sẽ áp dụng giá trị U (dấu là chiều trục Y, về phía hoặc ra xa đường tâm trục chính).
- ❑ Tốc độ cắt lập trình cho gia công tinh biên dạng (được xác định giữa P và Q) sẽ bị bỏ qua trong khi gia công thô.
- ❑ Địa chỉ D không sử dụng dấu thập phân, được lập trình theo định dạng bỏ qua các số zero đứng trước:

D0750 hoặc D750 tương đương chiều sâu .0750

Chỉ một số hệ điều khiển cho phép sử dụng dấu thập phân cho địa chỉ D (chiều sâu cắt) trong các chu kỳ G71 và G72

G74 - CHU KỲ KHOAN GIÁN ĐOẠN

G74 là một trong hai chu kỳ thường dùng cho các chi tiết không gia công tinh. Cùng với chu kỳ G75, G74 được dùng để cắt gián đoạn, chẳng hạn bẻ gãy phôi trong chuyển động cắt gọt dài. *G74 được dùng dọc theo trục Z.*

Đây là chu kỳ thường dùng để cắt gián đoạn dọc theo trục Z. Tên của chu kỳ là *Chu kỳ khoan gián đoạn*, tương tự chu kỳ khoan G73, được dùng để gia công các tâm. Đối với gia công tiện, ứng dụng chu kỳ G74 không đa năng bằng chu kỳ khoan gián đoạn G73 trên trung tâm gia công. Tuy công dụng chính của chu kỳ G74 là khoan gián đoạn trên máy tiện CNC, nhưng cũng có thể dùng để cắt gián đoạn khi tiện hoặc doa (chẳng hạn, các vật liệu rất cứng), cắt rãnh sâu ở mặt đầu, gia công cắt đứt, và nhiều ứng dụng khác.

Định dạng chu kỳ G74 - 10T/11T/15T

Định dạng lập trình một-block của chu kỳ G74:

```
G74 X..(U..) Z..(W..) I.. K.. D.. F.. S..
```

Trong đó:

- X (U) = Đường kính lỗ hoàn tất cần gia công
- Z (W) = Vị trí Z của chiều sâu khoan lỗ cuối cùng.
- I = Chiều sâu mỗi lần khoan (không có dấu)
- K = Khoảng cách mỗi lần khoan (không có dấu)
- D = Lượng để lại ở cuối đường cắt (bằng zero đối với lỗ mặt đầu)
- F = Tốc độ cắt (in/rev hoặc mm/rev)
- S = Tốc độ trục chính (ft/min hoặc m/min)

Định dạng chu kỳ G74 - 0T/16T/20T/21T

Định dạng lập trình hai -- block của chu kỳ G74

```
G74 R..
G74 X ..(U..) Z..(W..) P.. Q.. R.. F.. S..
```

Trong đó:

Block thứ nhất:

- R = Khoảng lùi dao (khoảng hở đối với từng lần cắt)

Block thứ hai:

- X (U) = đường kính lỗ hoàn tất cần gia công
- Z (W) = Vị trí Z của lần cắt cuối cùng (chiều sâu lỗ)
- P = Chiều sâu mỗi lần cắt (không có dấu)
- Q = Khoảng cách mỗi lần cắt (không có dấu)
- R = Lượng để lại ở cuối đường cắt (bằng zero đối với khoan mặt đầu)
- F = Tốc độ cắt (in/rev hoặc m/rev)
- S = Tốc độ trục chính (ft/min hoặc m/min)

Nếu không có X(U) và I (hoặc P) trong chu kỳ, sự gia công sẽ chỉ theo trục Z (khoan gián đoạn). Trong nguyên công khoan gián đoạn, chỉ lập trình các giá trị Z, K, và F (Hình 34.18).

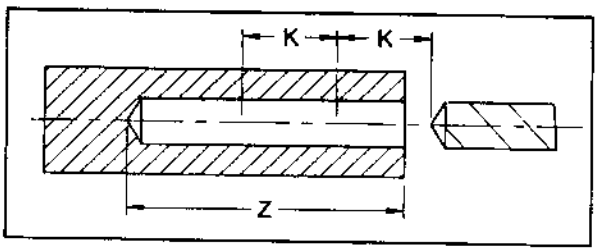
Ví dụ dưới đây minh họa chu kỳ G74:

```
O3407 (G74 PECK DRILLING)
N1 G20
N2 T0200
N3 G97 S1200 M03                (SPEED IN RPM)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N4 G00 X0 Z0.2 T0202 M08        (START POSITION)
N5 G74 Z-3.0 K0.5 F0.002        (PECK DRILLING)
N6 G00 X6.0 Z2.0 T0200          (CLEAR POSITION)
N7 M30                          (END OF PROGRAM)
%
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 34.18. Sơ đồ định dạng đối với chu kỳ G74

Khoan được thực hiện theo ba chiều sâu, số gia chiều sâu là 1/2 inch. Chú ý, chiều sâu lần khoan thứ nhất được tính từ điểm bắt đầu. Lập trình rãnh gián đoạn cũng tương tự.

CHU KỲ CẮT RÃNH – G75

G75 là một trong hai chu kỳ tiện dùng để gia công chi tiết đơn giản không đòi hỏi độ chính xác cao. Cùng với G74, G75 được dùng cho các nguyên công đòi hỏi sự cắt gọt gián đoạn, ví dụ, để bẻ gãy phoi trong khi chuyển động cắt sâu hoặc dài. *Chu kỳ G75 được sử dụng dọc theo trục X.*

Đây còn là chu kỳ rất đơn giản, được thiết kế để bẻ gãy phoi trong khi gia công thô dọc theo trục X, chủ yếu để cắt rãnh. Chu kỳ G75 đồng nhất với G74 nhưng trục X thay cho trục Z.

Định dạng chu kỳ G75 – 10T/11T/15T

Định dạng lập trình một – block của chu kỳ G75:

```
G75 X ..(U..) Z..(W..) I.. K.. D.. F.. S..
```

Trong đó:

- X (U) = Đường kính rãnh hoàn tất cần cắt gọt
- Z (W) = Vị trí Z của rãnh cuối (chỉ khi gia công nhiều rãnh)
- I = Chiều sâu mỗi lần cắt (không có dấu)
- K = Khoảng cách giữa các rãnh (không có dấu) (chỉ khi gia công nhiều tãnh)
- D = Lượng để lại ở cuối đường cắt (bằng zero hoặc không sử dụng khi cắt rãnh mặt đầu)
- F = Tốc độ cắt (in/rev hoặc mm/rev)
- S = Tốc độ trục chính (ft/min hoặc m/min)

Định dạng chu kỳ G75 – 0T/16T/18T/20T/21T

Định dạng lập trình hai – block của chu kỳ G75:

```
G75 R..
G75 X..(U..) Z..(W..) P.. Q.. R.. F.. S..
```

Trong đó:

Block thứ nhất:

- R = Khoảng lùi dao (khoảng hở đối với từng lần cắt)

Block thứ hai:

- X(U) = Đường kính rãnh hoàn tất cần cắt gọt
- Z (W) = Vị trí Z của rãnh cuối
- P = Chiều sâu mỗi lần cắt (không có dấu)
- Q = Khoảng cách giữa các rãnh (không có dấu)
- R = Lượng để lại ở cuối đường cắt (bằng zero hoặc không sử dụng khi cắt rãnh mặt đầu)
- F = Tốc độ cắt (in/rev hoặc mm/rev)
- S = Tốc độ trục chính (ft/min hoặc m/min)

Nếu không có Z(W) và K (hoặc Q) trong chu kỳ, sự gia công chỉ dọc theo trục (cắt rãnh gián đoạn). Ví dụ thực tiễn về chu kỳ G75 sẽ được giới thiệu trong Chương 35.

CÁC NGUYÊN TẮC CƠ BẢN CỦA G74 VÀ G75

Các nguyên tắc cơ bản chung cho cả hai chu kỳ bao gồm:

- Trong cả hai chu kỳ, các giá trị X và Z đều có thể lập trình theo chế độ tuyệt đối hoặc số gia.
- Cả hai chu kỳ đều cho phép lượng để lại ở cuối đường cắt theo chế độ tự động
- Có thể bỏ qua lượng để lại ở cuối đường cắt nếu giả thiết giá trị đó là zero
- Lượng lùi dao (khoảng hở đối với mỗi lần cắt) chỉ có thể lập trình trong định dạng hai – block. Với định dạng một – block, giá trị này được xác lập theo tham số bên trong bộ điều khiển.
- Nếu lập trình lượng lùi dao (định dạng hai-block), và lập trình cả lượng để lại, sự hiện diện của X sẽ xác lập ý nghĩa. Nếu lập trình giá trị X, giá trị R có nghĩa là lượng để lại ở cuối đường cắt.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

35 CẮT RÃNH TRÊN MÁY TIỆN

Cắt rãnh trên máy tiện CNC là nguyên công cắt gọt nhiều bước. Thuật ngữ *cắt rãnh* thường được áp dụng cho quá trình tạo hình hốc hẹp có chiều sâu xác định trên mặt trụ, côn, hoặc mặt đầu của chi tiết. Hình dạng rãnh, hoặc phần cơ bản của hình dạng rãnh, sẽ do dụng cụ cắt quyết định. Dụng cụ cắt rãnh còn được dùng cho nhiều nguyên công cắt gọt khác.

Dụng cụ cắt rãnh thường là mảnh carbide gắn trên cán dao đặc biệt. Có nhiều dạng mảnh chắp cắt rãnh, từ một đỉnh đến nhiều đỉnh, được chế tạo theo kích thước danh định. Dao cắt với mảnh chắp nhiều lưỡi cắt được dùng để giảm chi phí và tăng năng suất.

CÁC NGUYÊN CÔNG CẮT RÃNH

Các dụng cụ cắt rãnh *ngoài* hoặc *trong* sử dụng nhiều mảnh chắp với các cấu hình khác nhau. Sự khác biệt quan trọng nhất giữa cắt rãnh và tiện là *chiều cắt gọt*. Dao tiện có thể cắt theo nhiều chiều, dao cắt rãnh thường chỉ dùng để cắt theo một chiều. Ngoại lệ là nguyên công cắt rãnh thoát (giữa các bậc trục) thực hiện theo góc 45^0, góc mảnh chắp và góc ăn dao phải *đồng nhất* (thường là 45^0). Còn có ứng dụng thứ hai của chuyển động hai trục đồng thời trong cắt rãnh, cắt góc cho rãnh. Nói một cách chính xác, đây là nguyên công tiện. Tuy dao cắt rãnh không được thiết kế để tiện, nhưng vẫn có thể dùng gia công nhẹ, chẳng hạn cắt góc (vạt góc) nhỏ. Trong khi cắt góc rãnh, lượng cắt gọt (ăn dao) luôn luôn rất nhỏ và tốc độ cắt thường chậm.

Các chuẩn rãnh

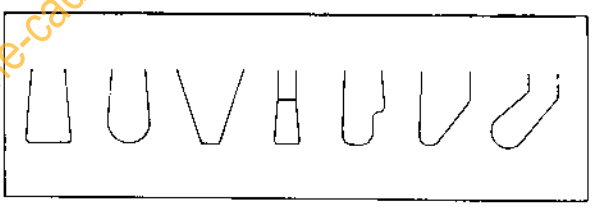
Đối với lập trình CNC, gia công rãnh thường không gây ra các khó khăn đặc biệt. Một số rãnh có thể lập trình dễ hơn số khác. Trước khi lập trình gia công rãnh, cần xem xét kỹ các yêu cầu kỹ thuật trên bản vẽ. Trên một chi tiết có thể có vài rãnh ở các vị trí khác nhau, bạn có thể viết chương trình con cho chúng. Khi chuẩn bị viết chương trình, bạn cần xem xét rãnh một cách cẩn thận. Khi chuẩn bị, bạn nên đánh giá rãnh theo ít nhất là ba tiêu chí dưới đây:

- ❑ Hình dạng rãnh
- ❑ Vị trí rãnh trên chi tiết
- ❑ Các kích thước và dung sai rãnh

Trong thực tế, một số rãnh có thể không đạt chất lượng cao nhất. Có lẽ do nhiều rãnh không yêu cầu độ chính xác cao, và khi cần gia công rãnh chính xác, nhà lập trình có thể thiếu kinh nghiệm. Bạn hãy đặc biệt chú ý độ bóng bề mặt và dung sai của rãnh.

HÌNH DẠNG RÃNH

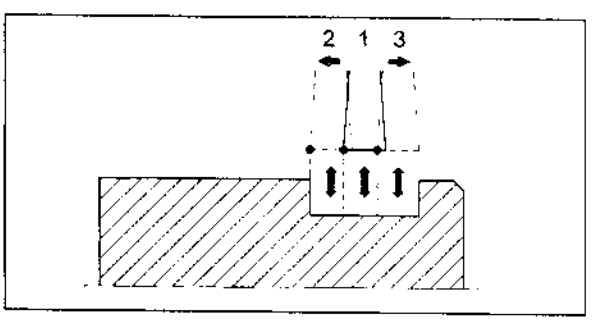
Đánh giá thứ nhất khi lập trình gia công rãnh là hình dạng rãnh. Hình dạng này được xác định trên bản vẽ chi tiết, đáp ứng công dụng của rãnh. Hình dạng rãnh là yếu tố quyết định khi chọn dao cắt rãnh. Rãnh có các góc sắc và song song với trục máy đòi hỏi mảnh chắp vuông, rãnh có đầu tròn (bán kính) đòi hỏi mảnh chắp có bán kính bằng hoặc hơi nhỏ hơn. Các rãnh công dụng đặc biệt, ví dụ dạng rãnh góc, cần mảnh chắp với các góc tương ứng với góc rãnh trên bản vẽ. Các rãnh hình dạng đặc biệt đòi hỏi mảnh chắp có hình dạng tương ứng. (Hình 35.1).



Hình 35.1. *Hình dạng của một số dao cắt rãnh thông dụng*

Kích cỡ mảnh chắp danh định

Trong nhiều nguyên công cắt rãnh, chiều rộng rãnh thường lớn hơn kích cỡ danh định của mảnh chắp trên dao cắt rãnh. Kích cỡ danh định thường có chiều rộng 1mm, 2mm, 3mm, hoặc 1/32, 3/64, 1/16, 1/8 inch...



Hình 35.2. *Phân bố đường cắt khi gia công rãnh rộng hơn mảnh chắp.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ví dụ, chiều rộng rãnh .276 inch có thể cắt với mảnh chắp có chiều rộng danh định *nhỏ hơn* gần nhất là .250 inch. Trong các trường hợp đó, chương trình gia công rãnh phải có ít nhất hai lần cắt – một hoặc vài lần cắt thô, và ít nhất một lần cắt tinh. Dao thứ hai có thể dùng để gia công tinh, nếu rãnh có dung sai cao hoặc để giảm sự mòn dao (Hình 35.2).

Chỉnh sửa mảnh chắp

Đôi khi nhà lập trình gặp rãnh đòi hỏi mảnh chắp có kích cỡ hoặc hình dạng đặc biệt. Có hai khả năng cần xem xét. Thứ nhất là đặt chế tạo mảnh chắp theo yêu cầu, nếu có thể và có tính thực tiễn. Đối với số lượng rãnh lớn, điều này có thể sẽ kinh tế. Thứ hai là chỉnh sửa hoặc mài lại lưỡi cắt có sẵn.

Nói chung, trong lập trình CNC, cần sử dụng các dao tiêu chuẩn thông dụng ở mức cao nhất. Trong các trường hợp đặc biệt, có thể chỉnh sửa dao hoặc mảnh chắp theo yêu cầu gia công cụ thể. Đối với cắt rãnh, có thể tăng đôi chút chiều sâu cắt hoặc sửa lại bán kính dao. Giải pháp cuối cùng là chỉnh sửa lại hình dạng rãnh. Sự chỉnh sửa hoặc mài lại dao tiêu chuẩn có thể dẫn đến giảm năng suất và tăng chi phí.

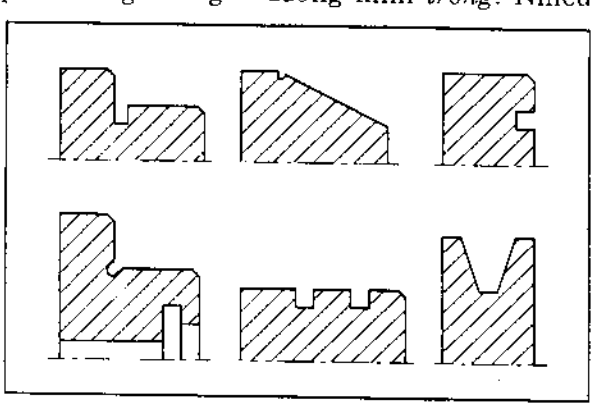
VỊ TRÍ RÃNH

Vị trí rãnh trên chi tiết được xác định từ bản vẽ. Vị trí rãnh có thể thuộc một trongba nhóm:

- ❑ Rãnh trên mặt trụ ...cắt đường kính
- ❑ Rãnh trên mặt côn ...cắt côn
- ❑ Rãnh trên mặt phẳng ...cắt bậc (vai)

Tuy có thể có các biến thể, nhưng trong thực tiễn chỉ xét ba nhóm nêu trên. Các vị trí này có thể là *trong* hoặc *ngoài*.

Hai vị trí rãnh phổ biến nhất là trên mặt trụ, phía ngoài thẳng – đường kính *ngoài*, hoặc phía trong thẳng – đường kính *trong*. Nhiều rãnh cắt có thể ở trên mặt phẳng (mặt đầu, bậc), trên mặt côn, hoặc ở góc (Hình 35.3)



Hình 35.3. Các vị trí rãnh trên chi tiết

KÍCH THƯỚC RÃNH

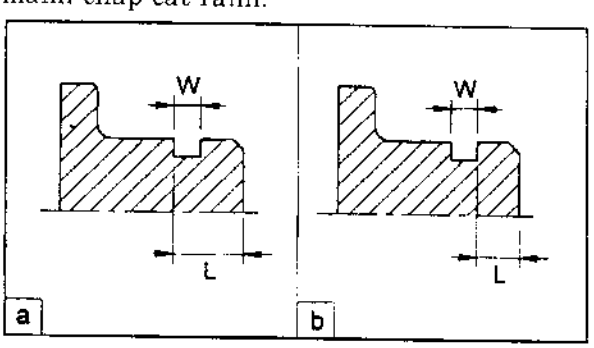
Các kích thước rãnh là rất quan trọng khi chọn mảnh chắp dao cắt rãnh. Các kích thước này gồm *chiều rộng, chiều sâu* rãnh, và các yêu cầu về góc rãnh. Nói chung, thường không có vấn đề nảy sinh khi sử dụng lưỡi cắt hẹp để làm rộng rành với nhiều đường cắt. Điều này cũng đúng khi dùng dao cắt sâu để cắt rãnh cạn. *Các kích thước rành xác định phương pháp gia công*. Rãnh có chiều rộng *bằng* chiều rộng lưỡi cắt được chọn cho hình dạng rành, yêu cầu chỉ *một* đường cắt. Toàn bộ yêu cầu là chuyển động dao *ăn vào* đơn giản và chuyển động *ra nhanh*. Để lập trình rãnh một cách chính xác, cần biết *chiều rộng, chiều sâu* và *vị trí* của rãnh liên quan đến điểm quy chiếu đã biết trên chi tiết. Vị trí này là khoảng cách đến một phía – hoặc một vách – của rãnh.

Một số rãnh rất lớn đòi hỏi phương pháp đặc biệt. Ví dụ, rãnh rộng 10 mm và sâu 8 mm không để gia công bằng một đường cắt. Trong trường hợp đó, các đường cắt thô không chỉ kiểm soát chiều rộng mà cả chiều sâu rãnh. Thậm chí có thể dùng nhiều dao để gia công rãnh. Có thể phải thiết kế chương trình theo từng phần. Nếu lưỡi dao bị gãy, chỉ cần lặp lại phần chương trình bị ảnh hưởng.

Vị trí rãnh

Hình 35.4 minh họa hai phương pháp ghi kích thước rãnh. Chiều rộng rãnh trong cả hai trường hợp là W, nhưng khoảng cách L từ mặt đầu là khác nhau giữa ví dụ a và ví dụ b.

Trong ví dụ 35.4a, kích thước L được đo đến *bên trái* rãnh. Đối với lập trình, kích thước này thuận tiện hơn, do sẽ thực sự xuất hiện trong chương trình như được chuyên biệt trên bản vẽ. Nói chung, điểm quy chiếu dao tiêu chuẩn của dao cắt rãnh được xác lập theo *bên trái* của mảnh chắp cắt rãnh.



***Hình 35.4.** Kích thước vị trí rãnh – hai phương pháp thông dụng.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

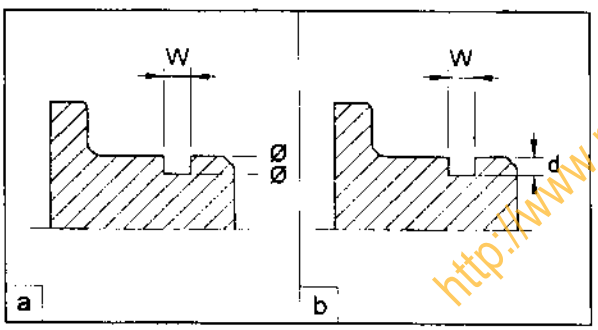
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trong ví dụ 36.4b, kích thước L là *bên phải* rãnh. Có thể dễ dàng xác định kích thước bên trái, bằng cách cộng với chiều rộng rãnh W. Các xem xét lập trình sẽ hơi khác, đặc biệt đối với dung sai kích thước. Bạn nên chọn kích thước chuyên biệt biểu thị kích thước *quan trọng*. Nếu khoảng dung sai được chuyên biệt cho kích thước bất kỳ, dung sai đó phải được duy trì trên rãnh hoàn tất, và sẽ phản ánh phương pháp lập trình chung. Rãnh cũng có thể ghi kích thước từ vị trí quy chiếu khác, tùy theo điều kiện cụ thể.

Chiều sâu rãnh

Hình 35.5 minh họa hai phương pháp ghi kích thước chiều sâu rãnh.

Trong Hình 35.5a, ghi cả đường kính đỉnh và đường kính đáy rãnh. Phương pháp này có ưu điểm là đường kính đáy của rãnh sẽ thực sự xuất hiện theo giá trị trục X trong chương trình. Nhược điểm là phải tính chiều sâu rãnh và chọn dao cắt rãnh thích hợp. Ví dụ trên Hình 35.5b chỉ nêu chiều sâu rãnh, cần tính đường kính đáy. Cả hai ví dụ này đều khá phổ biến trong lập trình CNC.



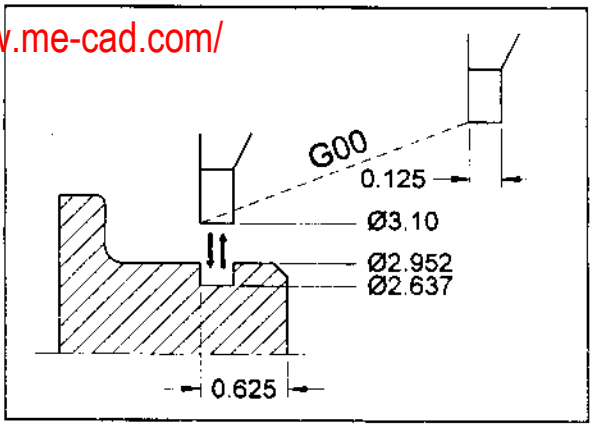
Hình 35.5. *Ghi kích thước chiều sâu rãnh – hai phương pháp thông dụng*

Các rãnh sâu thường là rãnh có chiều rộng hợp lý nhưng có tỷ số giữa đường kính đỉnh và đường kính đáy khá lớn.

LẬP TRÌNH RÃNH ĐƠN GIẢN

Đơn giản nhất trong tất cả các rãnh là loại có chiều rộng và hình dạng như lưỡi cắt của dụng cụ cắt (Hình 35.6).

Chương trình gia công loại rãnh này rất đơn giản. Trong chế độ nhanh, dịch chuyển dao cắt rãnh đến vị trí khởi đầu, ăn vào chiều sâu rãnh, sau đó lùi nhanh về vị trí ban đầu, và rãnh hoàn tất. Không có cắt gọt, không cần kiểm soát độ bóng bề mặt, cũng không sử dụng các kỹ thuật đặc biệt. Có người sẽ nói, như vậy sẽ không chất lượng. Sự tạm dừng ở đáy rãnh có thể là sự cải thiện duy nhất. Đúng, chất



Hình 35.6. *Ví dụ rãnh đơn giản – chương trình O3501. Chiều rộng lưỡi cắt bằng chiều rộng rãnh.*

lượng rãnh không cao, nhưng vẫn là rãnh, dù không đáp ứng yêu cầu gia công chính xác. Tuy nhiên lập trình gia công rãnh đơn giản là sự khởi đầu quan trọng để nghiên cứu các kỹ thuật tiên tiến hơn.

Chương trình O3501, sử dụng dao cắt rãnh vuông tiêu chuẩn. 125 inch, cắt rành có cùng chiều rộng. Chiều sâu rãnh là hiệu giữa hai đường kính ghi trên bản vẽ:

(2.952 – 2.637) / 2 = .1575 inch

Chương trình này sử dụng dao T08:

```
O3501 (RÃNH ĐƠN GIẢN)
(G20)
...
N33 T0800 M42                          (TOOL 8 ACTIVE)
N34 G97 S650 M03                       (650 RPM SPEED)
N35 G00 X3.1 Z-0.625 T0808 M08  (START POINT)
N36 G01 X2.637 F0.003          (FEED-IN TO DEPTH)
N37 G04 X0.4                  (DWELL AT THE BOTTOM)
N38 X3.1 F0.05               (RETRACT FROM GROOVE)
N39 G00 X6.0 Z3.0 T0800 M09 (CLEAR POSITION)
N40 M30                              (END OF PROGRAM)
%
```

Chương trình O3501 hoạt động như sau. Thứ nhất, nhập đơn vị Anh được xác lập trước từ đầu chương trình với G20. N33 và N34 là các block khởi động dao T08, chọn r/min trực tiếp. Thay vào đó, có thể chọn *Tốc độ Bề mặt không đổi* (CSS) trong chế độ G96. N35 là block, dao chuyển đến vị trí từ đó sẽ bắt đầu gia công rãnh (điểm khởi đầu). Khoảng hở ở vị trí an toàn này là phía trên đường kính chi tiết, trong ví dụ là .074 inch:

(3.1 – 2.952) / 2 = .074

Chất làm nguội cũng được áp dụng trong block đó, trong khi dao chuyển động. Block N36 là sự cắt rãnh với tốc độ cắt chậm .003 in/rev. Block N37 là thời gian tạm dừng 0.4 giây, sau đó dao trở về đường kính ban đầu, và chương trình hoàn tất.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tuy ví dụ cắt rãnh này rất đơn giản nhưng bạn hãy đánh giá chương trình chi tiết hơn. Chương trình này hàm chứa vài nguyên tắc quan trọng có thể áp dụng cho phương pháp lập trình rãnh bất kỳ, với yêu cầu cao về dung sai và độ bóng bề mặt.

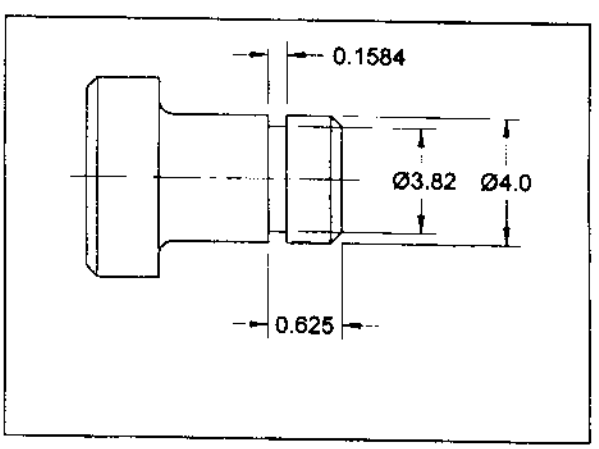
Bạn hãy chú ý khoảng hở *trước khi* bắt đầu cắt rãnh. Dao ở vị trí .074 inch phía trên đường kính chi tiết Ø3.100. Khoảng cách này là sự an toàn tối thiểu. Các rãnh thường được gia công với tốc độ cắt chậm và có thể tốn nhiều thời gian. Ngoài ra bạn cần lưu ý tốc độ cắt thực tăng từ .003 in/rev trong block N36 đến .050 in/rev trong block N38. Thay vào đó, đã có thể sử dụng lệnh chuyển động nhanh G00. Sự cắt gọt với tốc độ cắt đủ cao (thay vì sử dụng chuyển động nhanh) có thể cải thiện độ bóng bề mặt rãnh do giảm sự chà xát cạnh dao với vách (hai bên) rãnh.

Chiều rộng dao .125 không xuất hiện trong chương trình, dù trực tiếp hay gián tiếp. Điều đó có nghĩa là chiều rộng và hình dạng lưỡi cắt sẽ trở thành chiều rộng và hình dạng rãnh. Chiều rộng rãnh sẽ khác nếu dùng kích cỡ dao khác, dù cấu trúc chương trình không thay đổi, kể cả khi *hình dạng* dao cắt rãnh thay đổi. Kết hợp thay đổi hình dạng *và* kích cỡ dao sẽ có vô số cơ hội khả dĩ mà không cần thay đổi chương trình.

KỸ THUẬT GIA CÔNG RÃNH CHÍNH XÁC

Rãnh gia công theo kỹ thuật nêu trên thường không chính xác. Các vách có thể có bề mặt nhám, các góc ngoài có cạnh sắc, chiều rộng rãnh phụ thuộc chiều rộng lưỡi cắt và độ mòn của dao. Trong nhiều trường hợp, loại rãnh này là không thể chấp nhận.

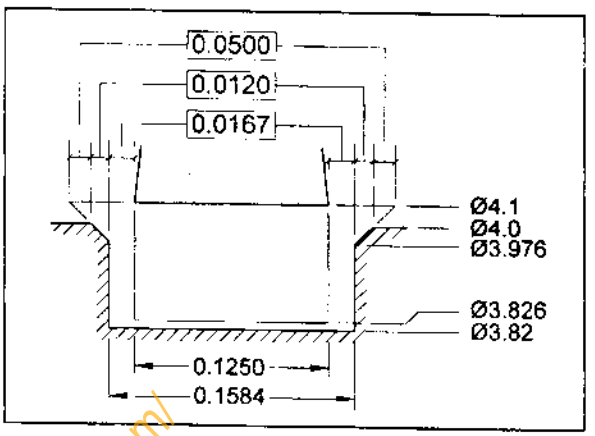
Lập trình và gia công rãnh chính xác đòi hỏi thêm công sức, để có rãnh chất lượng cao.



Hình 35.7. Bản vẽ ví dụ rãnh chính xác – chương trình O3502

Hai minh họa dưới đây nêu rõ các kích thước rãnh và chi tiết lập trình liên quan. Hình 35.7 minh họa rãnh chính xác, chiều rộng được vẽ lớn một cách cố ý để dễ hình dung.

Phương pháp cắt gọt nào là tốt nhất? Một đường cắt thô ăn vào và hai đường cắt tinh cho hai vách là hợp lý; do đó cần để lại lượng dư 0.006 inch ở đường kính đáy để gia công tinh. Ngoài ra cần vạt góc cho các góc ở đường kính Ø4.0. Hình 35.8 minh họa sự phân bố các đường kính cắt.



Hình 35.8. Rãnh chính xác, phân bố các đường cắt trong chương trình O3502.

Trước khi lập trình block thứ nhất, bạn cần chọn dao cắt và phương pháp gia công. Đây là các quyết định quan trọng, do có ảnh hưởng trực tiếp đến chất lượng rãnh hoàn tất.

Chọn chiều rộng rãnh

Dao gia công rãnh được chọn cho ví dụ trong chương trình O3502 sẽ là dao ngoài được gán số ổ dao ba – T03. Điểm quy chiếu (chuẩn) dao được chọn tại *lưỡi bên trái* của mảnh chắp, đây cũng là lựa chọn tiêu chuẩn. Ngoài ra còn cần lựa chọn chiều rộng mảnh chắp. Các mảnh chắp gia công rãnh có chiều rộng tiêu chuẩn, thường với số gia 1mm (hệ mét), 1/32 hoặc 1/16 inch (hệ Anh). Trong trường hợp này, chiều rộng rãnh phi – tiêu chuẩn là .1584 inch. Chiều rộng lưỡi cắt tiêu chuẩn gần nhất là 5/32 inch (0.15625 inch). Câu hỏi là, có nên chọn chiều rộng mảnh chắp 5/32 inch? Không nên. Về lý thuyết, mảnh chắp này *có thể* gia công rãnh, nhưng do hiệu số thực tế giữa chiều rộng rãnh và mảnh chắp quá nhỏ (.10215 inch trên hai vách), lượng dư còn lại hầu như không đáng kể.

Hiệu số kích thước cho phép chỉ hơi lớn hơn .001 trên từng vách rãnh, có thể làm cho cạnh mảnh chắp chà xát thay vì cắt gọt phần vách đó. Lựa chọn tốt hơn có thể là giảm chiều rộng

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

mảnh thiết kế yêu cầu đến bốn phần mười (.4) inch tiếp, đó là 1/8 inch (.1250). Tính linh hoạt của chiều rộng 1/8 inh cao hơn loại 5/32 inch. Sau khi chọn dao cắt rãnh, có thể gán các giá trị ban đầu, trị số bù dao (03), tốc độ chính (400 ft/min), khoảng tốc độ (M42):

- T0303 = DAO CẮT RÃNH VUÔNG .1250

Có thể viết vài block đầu tiên của chương trình:

```
O3502 (PRECISION GROVE)
(G20)
...
N41. T0303 M42
N42 G96 S400 M03
...
```

Phương pháp gia công

Sau khi chọn dao cắt rãnh và gán trị số ổ dao, cần quyết định *phương pháp* gia công rãnh. Phần trên đã khái quát phương pháp gia công, phần này sẽ trình bày chi tiết.

Một phương pháp lập trình đơn giản là *không* chọn kỹ thuật *vào - ra* cơ bản (chương trình O3501). Điều đó có nghĩa là cần tìm phương pháp *tốt hơn*, bảo đảm chất lượng cao cho rãnh gia công. Bước thứ nhất hướng đến mục đích này là áp dụng yếu tố chiều rộng lưỡi cắt nhỏ hơn chiều rộng rãnh, dao sẽ ăn vào rãnh vài lần. Không khó để tính rãnh rộng .1584 được gia công với lưỡi cắt rộng .1250 sẽ cần *ít nhất* hai đường cắt rãnh. Nhưng khi rãnh rộng hơn nhiều so với rãnh nêu trên thì sao?

Phương pháp tính *số lần cắt rãnh tối thiểu* có công thức như sau:

$$C_{min} = \frac{G_w}{T_w}$$

Trong đó:

C_{min} = Số lần cắt rãnh tối thiểu
G_W = Chiều rộng rãnh gia công
T_W = Chiều rộng lưỡi cắt rãnh

Áp dụng công thức cho ví dụ, dữ liệu ban đầu là chiều rộng rãnh .1584 inch và chiều rộng lưỡi cắt .1250 inch, do đó cần tối thiểu *hai* đường cắt rãnh. Cần làm tròn số tăng, đến số nguyên gần nhất

.1584 / .1250 = 1.2672 ⇒ 2 đường cắt

Các bước gia công gồm ăn vào một lần cho vách bên trái của rãnh, lần thứ hai cho vách bên phải, và vạt góc. Rãnh được lập trình theo cách này có thể chấp nhận được, nhưng *không* đạt chất lượng cao.

Ngay cả khi chất lượng rãnh là *chấp nhận được*, nhưng cả nhà lập trình và người vận hành CNC đều không thực sự hài lòng. Vậy, biện pháp để gia công rãnh chất lượng *cao* nhất có thể đạt được là gì?

Để viết các chương trình tối ưu, cần nỗ lực để đạt đến chất lượng chấp nhận được ở mức *lập trình*, nhằm tránh các vấn đề ở mức *gia công*.

Có thể áp dụng nguyên tắc nêu trên vào ví dụ này. Cốt lõi là kiến thức về quy trình công nghệ. Kinh nghiệm gia công cho thấy cắt gọt lượng dư *bằng nhau* từ từng vách (phía) rãnh sẽ có các điều kiện cắt gọt tốt hơn, kiểm soát độ bóng bề mặt tốt hơn, và tuổi bền dụng cụ cắt dài hơn.

Nếu áp dụng nhận xét đó trong ví dụ hiện hành, cần có kết luận quan trọng. Nếu hai lần cắt với chiều rộng không đều dẫn đến kết quả *chấp nhận được*, thì *ba* lần cắt *phân bố đều* sẽ cho kết quả tốt hơn.

Nếu sử dụng *ít nhất ba* đường cắt để tạo rãnh thay vì tối thiểu hai lần, nhà lập trình CNC sẽ có tố thể kiểm soát hai yếu tố quan trọng:

- Kiểm soát VỊ TRÍ rãnh
- Kiểm soát CHIỀU RỘNG rãnh

Trong gia công rãnh chính xác, hai yếu tố này quan trọng như nhau và phải được xem xét cùng với nhau.

Bạn hãy xét phương pháp áp dụng hai yếu tố nêu trên trong ví dụ. Yếu tố thứ nhất áp dụng với sự điều khiển chương trình là vị trí rãnh.*Vị trí* rãnh trên bản vẽ là .625 inch cách mặt đầu của chi tiết, đến vách trái của rãnh. Không có các dung sai kích thước cộng và trừ, do đó kích thước trên bản vẽ được sử dụng và được lập trình trực tiếp. Yếu tố thứ hai được chương trình điều khiển là *chiều rộng* rãnh. Đây là giá trị .1584 inch trên bản vẽ và chiều rộng dao đã chọn là .1250. Mục đích là lập trình các chuyển động cắt theo *ba bước*, sử dụng kỹ thuật đã nêu:

➲ Bước 1

Gia công thô vào giữa rãnh, để lại lượng dư bằng nhau trên cả hai vách rãnh nhằm gia công tinh, và để lại lượng dư nhỏ ở đáy rãnh.

➲ Bước 2

Lập trình dao gia công từ vách trái của rãnh, kể cả vạt góc

➲ Bước 3

Lập trình dao gia công vách bên phải của rãnh, kể cả vạt góc và vét đáy rãnh hướng về vách trái.

Hai bước cuối yêu cầu cắt (hoặc vạt) các góc,

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

chiều rộng vật liệu tương với phải cắt kế tiếp không lớn hơn một nửa đến ba phần tư chiều rộng lưỡi cắt. Trong bước thứ ba, cần vét (gia công) phần đáy rãnh, do đó phải xét lượng dư để gia công tinh.

Lượng dư để gia công tinh

Trong bước thứ nhất, dao đi xuống *đúng* tâm rãnh. Để tính toán vị trí trục Z khi khởi đầu, trước hết cần tìm lượng dư trên từng vách được để lại cho gia công tinh. Lượng dư này là một nửa của hiệu giữa chiều rộng rãnh và chiều rộng lưỡi cắt (Hình 35.8).

(.1584 - .1250) / 2 = .0167

Vị trí Z của dao sẽ là .0167 theo phía dương của vách trái. Nếu vách này là Z-0.625, vị trí khởi đầu của dao cắt rãnh sẽ là Z-0.6083. Khi dao hoàn tất lần cắt thứ nhất, sẽ để lại lượng dư bằng nhau trên cả hai vách của rãnh.

Bạn nên tránh làm tròn số .0167, ví dụ, thành .0170 inch. Tuy điều này không gây ra sự khác biệt trong gia công, nhưng lập trình nói chung đều yêu cầu chỉ sử dụng các giá trị tính toán. Ích lợi của phương pháp này là dễ kiểm tra chương trình và tính ổn định trong lập trình. Lượng dư bằng nhau sẽ bảo đảm tính ổn định, .0167 và .0167 là lựa chọn tốt hơn so với .0170 và .0164, dù kết quả thực tế là như nhau.

Kế tiếp, bạn cần xem xét các vị trí X. Vị trí thứ nhất là nơi bắt đầu cho dao ăn vào, vị trí thứ hai là đường kính cuối đối với đường cắt ăn xuống. Vị trí khởi đầu có thể là .050 *theo một phía* ở trên đường kính hoàn tất, trong trường hợp này sẽ là đường kính khoảng hở tính toán từ Ø4.0:

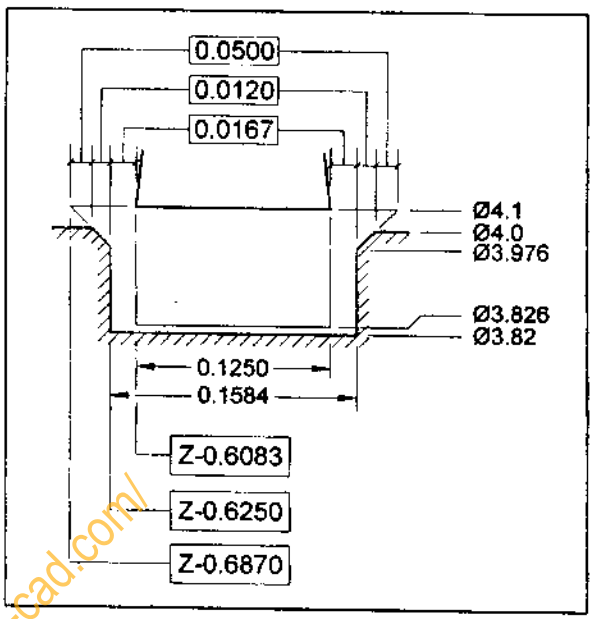
4.0 - .05 x 2 = 4.1 (X4.1)

Bạn không nên bắt đầu cắt với khoảng hở lớn hơn .050 inch (1.27 mm) – với các tốc độ cắt chậm, thường dùng để gia công rãnh, khoảng cách dao chạy mà không cắt gọt sẽ quá lớn, do đó sẽ không hiệu quả. Đường kính cuối là đáy rãnh, trên bản vẽ là 3.82. Cần lập trình kích thước X3.82 là đường kính đích, nhưng phải để lại lượng dư nhỏ, chẳng hạn .003 theo một phía (.006 theo đường kính), để *gia công tinh* đáy rãnh. Điều đó sẽ cộng thêm hai lần .003 cho đường kính rãnh 3.82, do đó đích X được lập trình là 3.826. Sau một lần ăn xuống, dao sẽ trở lại đường kính ban đầu:

```
N43 G00 X4.1 Z-0.6083 T0303 M08
N44 G01 X3.826 F0.004
N45 G00 X4.1
```

Trong trường hợp đó, chuyển động nhanh trở lại phía trên rãnh (N45) là lựa chọn tốt, do các vách sau này được gia công tinh, độ bóng bề mặt vách vào thời điểm này là không quan trọng. Rãnh sau khi được gia công thô, sẽ bắt đầu gia công tinh.

Có thể đưa tất cả các giá trị đã tính toán vào Hình 35.8, từ đó sẽ có các dữ liệu được nêu trên Hình 35.9.



Hình 35.9. *Rãnh chính xác – dữ liệu gia công rãnh được dùng trong chương trình O3502.*

Dung sai rãnh

Cũng như mọi nguyên công khác, chương trình gia công rãnh phải có cấu trúc sao cho có thể duy trì các dung sai tại máy công cụ. Tuy không có dung sai chuyên biệt trong ví dụ, nhưng được ngầm hiểu bằng kích thước bốn chữ số thập phân. Khoảng dung sai, chẳng hạn 0.0 đến +0.01, có lẽ là cách thức phổ biến khi chuyên biệt dung sai. Chỉ giá trị kích thước *bên trong* khoảng chuyên biệt này mới được sử dụng trong chương trình. Trong ví dụ này, mục tiêu là kích thước trên bản vẽ .1584 (được chọn một cách cố ý).

Có thể gặp một vấn đề trong khi gia công, ảnh hưởng đến chiều rộng rãnh, đó là *sự mòn dao*. Khi dao cắt gọt, sẽ mòn dần, lưỡi cắt ngày càng *thu hẹp*. Tuy vẫn còn khả năng cắt gọt, nhưng chiều rộng rãnh có thể bị lệch ra ngoài khoảng dung sai. Nguyên nhân thứ hai có thể làm cho chiều rộng rãnh không đạt yêu cầu là *chiều rộng mảnh chắp*. Các mảnh chắp được chế tạo với độ chính xác rất cao, nhưng vẫn trong khoảng dung sai xác định. Nếu thay mảnh chắp, chiều rộng rãnh có thể thay đổi đôi chút,

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

do mảnh tiếp theo có thể không cắt chính xác rộng như mảnh cũ. Để loại bỏ, hoặc ít nhất là giảm thiểu khả năng *lệch khỏi khoảng dung sai*, bạn cần áp dụng kỹ thuật đơn giản, chỉ lập trình sự bù *bổ sung* cho các nguyên công cắt gọt tinh.

Ở phần trước, khi lập kế hoạch gia công rãnh chính xác, đã chọn bù 03 cho dao cắt rãnh. Vậy, tại sao phải bù *bổ sung*? Giả sử, vào thời điểm nào đó, mọi xác lập máy chỉ sử dụng một giá trị bù trong chương trình, nhưng trong khi gia công, rãnh bị hẹp do dao bị mòn. Giải pháp sẽ là gì? Nếu điều chỉnh xác lập bù trục Z, theo chiều âm hoặc dương, sẽ thay đổi *vị trí* rãnh so với zero chương trình nhưng *không thay đổi chiều rộng rãnh*. Điều cần thiết là sự bù thứ hai, chỉ kiểm soát chiều rộng rãnh. Trong chương trình O3502, vạt góc trái và vách sẽ hoàn tất với một giá trị bù (03), vạt góc và vách phải sẽ sử dụng sự bù thứ hai. Để dễ nhớ sự bù thứ hai, bạn có thể chọn số 13.

Bước tiếp theo cần hoàn thành là tính toán vị trí bắt đầu vạt góc trái. Hiện tại, dao ở vị trí Z-0.683 nhưng sẽ dịch chuyển theo lượng dư vách .0167, chiều rộng vạt góc .012 và khoảng hở .050. Tổng cộng là .0787 đến vị trí Z-0.687. Với tốc độ cắt chậm, vạt góc được thực hiện trước và tiếp tục cắt để hoàn tất vách trái, *đến cùng đường kính* như đối với gia công thô, là X3.826:

```
N46 Z-0.687
N47 G01 X3.976 Z-0.625 F0.002
N48 X3.826 F0.003
```

Bước tiếp theo là đưa dao trở về phía trên đường kính chi tiết. Chuyển động này rất quan trọng. Trong chương trình, bạn cần bảo đảm vách trái đã hoàn tất không bị hư hại khi rút dao ra từ đáy rãnh. Ngoài ra, cần bảo đảm dao không tiếp xúc phần chưa gia công bên vách phải. Điều đó có nghĩa là không rút dao ra xa hơn vị trí Z-0.6083. Điều đó còn có ý nghĩa là *không rút quá nhanh* (Chương 19). Cách tốt nhất là đưa dao về vị trí khởi động ban đầu với tốc độ tương đối *cao nhưng không cắt gọt*:

```
N49 X4.1 Z-0.6083 F0.04
```

Tại điểm này, vách trái của rãnh sẽ hoàn tất. Để lập trình các chuyển động cho vách phải, dao phải cắt với lưỡi cắt *bên phải*. Một phương pháp là thay đổi các tọa độ G50 trong chương trình, nếu xác lập cũ này vẫn còn được sử dụng, hoặc dùng sự bù tọa độ làm việc mới. Phương pháp được dùng ở đây có lẽ là đơn giản nhất và an toàn nhất. Mọi chuyển động liên quan đến vạt góc phải và vách bên phải của rãnh đều được lập trình theo *chế độ số gia*, chỉ áp dụng cho trục Z, sử dụng địa chỉ W:

```
N50 W0.0787 T0313
N51 X3.976 W-0.062 F0.002
```

Trong block N50, dao hành trình với khoảng cách tương đương tổng lượng dư trên vách phải .0167, vạt góc .012 và khoảng hở .050. Lệnh bù thứ hai cũng được lập trình trong block này. Đây là block duy nhất có thể *áp dụng* bù 13, block đứng trước, sẽ quá sớm, còn block kế tiếp, sẽ quá trễ.

Block N51 chứa vị trí vạt góc đích và chế độ tuyệt đối cho trục X kết hợp chế độ số gia cho trục Z.

Để hoàn tất vách bên phải của rãnh, cần gia công đường kính đáy rãnh, block 52, sau đó tiếp tục cắt lượng dư .003 ở đường kính đáy (block 53) – *gia công tinh đáy rãnh*:

```
N52 X3.82 F0.003
N53 Z-0.6247 T0303
```

Bạn hãy xem xét giá trị cuối trên trục Z – đây là giá trị rất nhỏ .0003 so với kích thước bản vẽ .625. Mục đích ở đây là bù cho áp suất dao. *Đây không phải là bậc ở góc rãnh*. Do gia công tinh sẽ kết thúc ở phía *trái* của rãnh, cần chú ý đến giá trị bù ban đầu (03). Block N53 là block duy nhất, nơi sự thay đổi bù là đúng. Bạn cần bảo đảm không thay đổi chỉ số dao – *cụm ổ dao sẽ phân độ dao*!

Lúc này có thể hoàn tất chương trình O3502. Toàn bộ điều còn lại cần thực hiện là trở về vị trí khởi đầu của rãnh, tiếp theo là các block kết thúc chương trình.

```
N54 X4.1 Z-0.6083 F0.04
N55 G00 X10.0 Z2.0 T0300 M09
N56 M30
%
```

Tại điểm này, chương trình O3502 có thể *hoàn tất*. Bạn hãy chú ý các block chương trình, nơi thay đổi giá trị bù:

```
O3502 (RÃNH CHÍNH XÁC)
(G20)
...
N41 T0300 M42                          (NO OFFSET)
N42 G96 S400 M03
N43 G00 X4.1 Z-0.6083 T0303 M08        (OFFSET 03)
N44 G01 X3.826 F0.004
N45 G00 X4.1
N46 Z-0.687
N47 G01 X3.976 Z-0.625 F0.002
N48 X3.826 F0.003
N49 X4.1 Z-0.6083 F0.04
N50 W0.0787 T0313                      (OFFSET 13)
N51 X3.976 W-0.062 F0.002
N52 X3.82 F0.003
N53 Z-0.6247 T0303                     (OFFSET 03)
N54 X4.1 Z-0.6083 F0.04
N55 G00 X6.0 Z3.0 T0300 M09            (NO OFFSET)
N56 M30
%
```

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

CHÚ Ý: Điều quan trọng là bạn phải cẩn thận khi sử dụng hai giá trị bù cho cùng một dao trong khi gia công (điều này áp dụng chung – không chỉ khi gia công rãnh).

Bạn cần nhớ, *mục đích* của chế độ bù trong ví dụ này là kiểm soát *chiều rộng* rãnh, *không phải* đường kính.

Bạn hãy luôn luôn tuân thủ các chú ý dưới đây, dựa trên chương trình ví dụ O3502:

- Bắt đầu gia công với các giá trị ban đầu như nhau được gán cho cả hai chế độ bù (cùng các giá trị XY như nhau cho các chế độ bù 03 và 13).
- Các giá trị bù X của 03 và 13 phải luôn luôn như nhau. Nếu xác lập X của một chế độ bù thay đổi, xác lập của chế độ bù kia cũng phải thay đổi theo cùng giá trị. Bạn hãy điều chỉnh cả hai giá trị bù X để kiểm soát dung sai chiều sâu rãnh.
- Nếu chiều rộng rãnh trở nên quá hẹp và cần điều chỉnh, chỉ thay đổi giá trị bù Z.
- Để điều chỉnh vị trí vách trái của rãnh, hãy thay đổi giá trị Z của bù 03.
- Để điều chỉnh vị trí vách phải của rãnh, hãy thay đổi giá trị Z của bù 13.
- Không xóa chế độ bù hiện hành – chỉ đổi sang chế độ bù thứ hai một cách trực tiếp.
- Bảo đảm chỉ số dao (hai chữ số thứ nhất của địa chỉ T) không thay đổi; nếu thay đổi, ĐÓ SẼ LÀ SỰ THAY DAO.

Ngoài ra còn có các chú ý khác, tùy theo các điều kiện cụ thể. Về nguyên tắc, cần kiểm tra chương trình một cách cẩn thận trước khi đưa vào sản xuất.

Độ bóng bề mặt rãnh

Đến đây có thể bạn đã hiểu kỹ thuật lập trình gia công rãnh chính xác. Điều còn lại là độ bóng bề mặt của rãnh sau khi gia công. Bạn nên tuân theo các phương pháp phân bố đường cắt bằng nhau, tốc độ cắt và tốc độ trục chính thích hợp, dụng cụ cắt và mảnh chắp chuẩn, chất làm nguội thích hợp, và các kỹ thuật khác được dùng trong ví dụ, độ bóng bề mặt sẽ đạt yêu cầu.

Bạn cần nhớ, thuật ngữ "*rãnh chính xác*" không chỉ có nghĩa là vị trí và kích thước rãnh chính xác, mà còn phải bảo đảm chất lượng bề mặt rãnh.

NHIỀU RÃNH

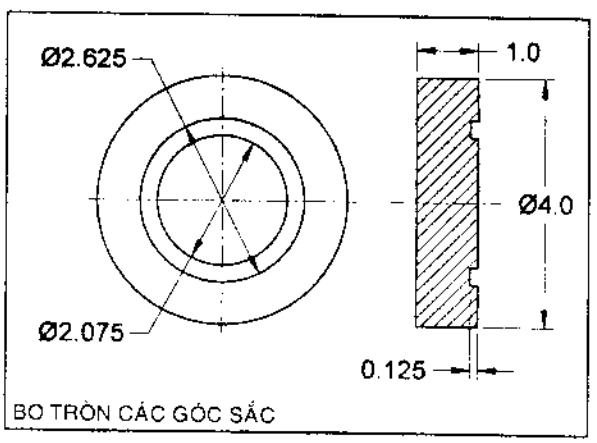
Gia công nhiều rãnh là thuật ngữ thường dùng với ý nghĩa cắt cùng một rãnh ở các vị trí khác nhau trên chi tiết. Trong các trường hợp đó, chương trình sẽ rút gọn bằng cách dùng chương trình con cho nhiều rãnh, sẽ được gọi lại các vị trí rãnh tương ứng. Các chương trình con sẽ tiết kiệm thời gian, chúng dễ thiết kế và dễ chỉnh sửa. Tuy chương trình con sẽ được trình bày trong Chương 38, nhưng ví dụ lập trình gia công nhiều rãnh sử dụng chương trình con được nêu ở cuối chương này, có thể được coi là sự giới thiệu cơ bản.

Khi cắt các rãnh, lượng phoi cũng nhiều hơn. Trên các rãnh đường kính ngoài, không cần các khảo sát đặc biệt, trọng tâm là loại bỏ lượng phoi lớn. Tình huống sẽ hoàn toàn khác khi gia công rãnh trong. Khi gia công rãnh trong, phoi có thể tích tụ trong lỗ doa. Các phoi này có thể làm hư hại đường kính lỗ, thậm chí cả dao cắt rãnh. Để giải quyết vấn đề đó, bạn hãy xét sự gia công vài rãnh, kéo dao ra và thổi phoi ra khỏi lỗ. Trong trường hợp này, có thể dùng lệnh dừng M01. Sau khi loại bỏ hết toàn bộ phoi, sẽ tiếp tục cắt rãnh kế tiếp.

RÃNH Ở MẶT ĐẦU

Cắt rãnh mặt đầu là quá trình gia công rãnh với dao chuyển động theo trục Z. Dao được lập trình theo cùng các nguyên tắc như khi gia công rãnh dọc theo trục X. Do bản chất của rãnh ở mặt đầu, *hướng* dao là vấn đề quan trọng nhất. Giải pháp là *khoảng hở hướng tâm* của dao trong khi gia công. Không cần chú ý quá mức về khoảng hở hướng tâm khi cắt rãnh theo trục X, do lưỡi cắt trên cùng một mặt phẳng với đường tâm của máy. Tuy nhiên, khi cắt rãnh mặt đầu (trục Z), khoảng hở dao dọc theo bán kính cắt là quan trọng nhất.

Ví dụ kế tiếp minh họa phương pháp lập trình gia công rãnh mặt đầu (Hình 35.10).



Hình 35.10. Ví dụ cắt rãnh mặt đầu – chương trình O3503

Tuy cả hai đường kính trong và ngoài của rãnh đều là lựa chọn kỹ thuật trên bản vẽ, nhưng cũng cần chiều rộng thực của rãnh để lập

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

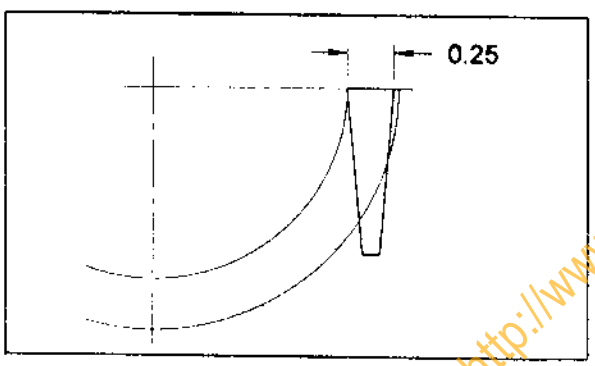
trình. Để tính chiều rộng rãnh, bạn hãy tính một nửa hiệu số của hai đường kính rãnh:

(2.625-2.075) / 2 = .275 inch.

Đây là giá trị chiều rộng thực (.275) của rãnh trong ví dụ. Bạn cần nhớ, chương trình sẽ sử dụng dao cắt rãnh có chiều rộng 0.250 nhỏ hơn chiều rộng rãnh. Bạn hãy xem lại ví dụ về rãnh chính xác đã nêu ở phần trên, cần ba lần cắt – một lần cắt thô vào giữa rãnh và hai lần cắt tinh, với sự vạt góc nhỏ. Nhưng trước hết, bạn hãy xét *khoảng hở hướng tâm* của dao cắt rãnh. Đây là sự xem xét lập trình rất quan trọng, đặc trưng cho hầu hết các nguyên công cắt rãnh mặt đầu, nhưng dễ bị bỏ qua.

Khoảng hở hướng tâm

Dao cắt rãnh mặt đầu được lắp vuông góc (90^0) với mặt đầu của chi tiết (song song với đường tâm trục chính). Dao cắt rãnh tiêu chuẩn hầu như không có khoảng hở hướng kính, có thể va chạm với chi tiết ở đầu thấp (Hình 35.11).



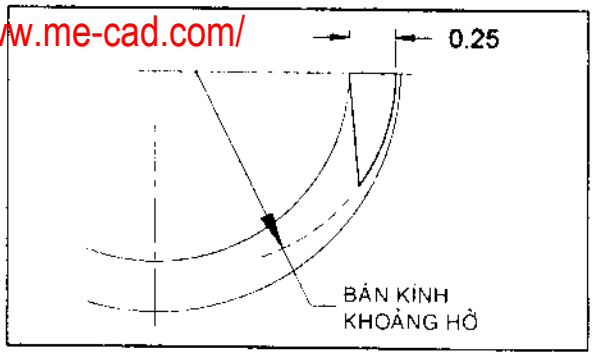
Hình 35.11. Va chạm giữa dao cắt rãnh tiêu chuẩn và rãnh mặt đầu

Đây là nguyên công đơn giản, nhưng dao phải được mài hợp lý. Bạn cần bảo đảm mài dao không ảnh hưởng đến chiều rộng lưỡi cắt và lượng mài phải ở mức tối thiểu để không làm giảm độ bền của dao.

Ví dụ chương trình cắt rãnh mặt đầu

Chương trình O3503 sử dụng dao cắt rãnh cải tiến (Hình 35.12) và vạt góc .012 để loại bỏ các góc sắc. Chương trình này chỉ sử dụng một chế độ bù. Điểm xác lập dao là cạnh dưới của mảnh chắp, tương ứng ∅2.075. Tất cả các tính toán đều dễ dàng thực hiện, hoàn toàn tương tự như gia công rãnh theo trục X:

```
O3503 (FACE GROOVE)
(G20)
...
N21 T0400 M42
N22 G96 S450 M03
```



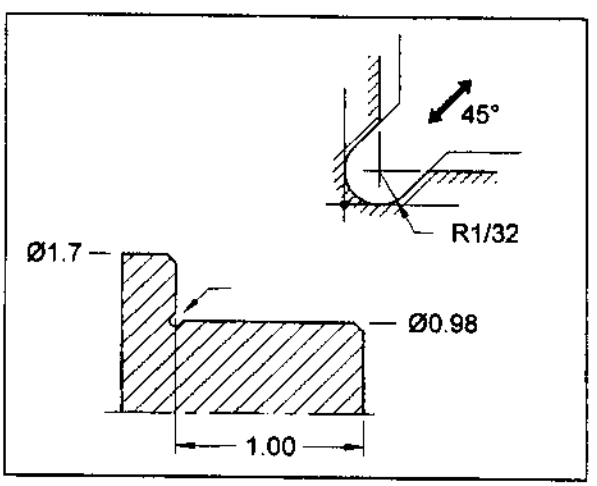
Hình 35.12. Chỉnh sửa dao cắt rãnh tiêu chuẩn để gia công rãnh mặt đầu

```
N23 G00 X2.1 Z0.05 T0404 M08
N24 G01 Z-0.123 F0.003
N25 Z0.05 F0.04
N26 X1.951
N27 X2.075 Z-0.012 F0.001
N28 Z-0.123 F0.003
N29 X2.1 Z0.05 F0.04
N30 U0.149
N31 U-0.124 Z-0.012 F0.001
N32 Z-0.125 F0.003
N33 X2.0755
N34 X2.1 Z0.05 F0.04 M09
N35 G00 X8.0 Z3.0 T0400
N36 M30
%
```

RÃNH GÓC/RÃNH BẬC

Cắt rãnh góc cũng là sự gia công rãnh, sử dụng dao đặc biệt được thiết kế để cắt theo chuyển động góc 45^0. Rãnh có thể vuông hoặc với bán kính, tùy theo thiết kế rãnh và loại dao. Dao cắt rãnh cũng có thể là loại mảnh chắp tiêu chuẩn, lắp vào cán nghiêng 45^0. Mục đích của kiểu rãnh này là để gia công các bậc lõm, thường ở góc chi tiết, để bảo đảm sự tương hợp bậc (vai) của hai chi tiết lắp ghép.

Để lập trình rãnh góc, cần biết bán kính của mảnh chắp gia công rãnh, trong ví dụ này là .031 (1/32) inch. Chiều sâu cắt được thiết lập



Hình 35.13. Rãnh góc – chương trình O3504

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

từ bản vẽ. Mối quan tâm này đặc biệt là "rãnh tối thiểu". Trong trường hợp này, tâm rãnh phải là giao của bậc và đường kính. Chuyển động cắt vào và ra rãnh phải nghiêng 45^0, nghĩa là cùng giá trị hành trình trên cả hai trục X và Z. Hình 35.13 minh họa rãnh góc với đường cắt có bán kính .031.

Chương trình này tương đối dễ hiểu

```
O3504 (RÃNH GÓC)
(G20)
...
N217 G50 S1000 T0500 M42
N218 G96 S375 M03
N219 X1.08 Z-0.95 T0505 M08
N220 G01 X0.918 Z-1.031 F0.004
N221 G04 X0.1
N222 X1.08 Z-0.95 F0.04
N223 G00 X6.0 Z3.0 T0500 M09
N224 M30
%
```

Block N219 định vị dao sao cho tâm của mảnh chắp (và điểm gá lắp dao) ở ngay trên đường tâm của rãnh góc (khoảng hở .050 theo các trục X và Y). Các block N220 và N222 là hai chuyển động cắt – một là ăn vào rãnh (N220), và thứ hai là đi ra khỏi rãnh (N222). Khoảng hành trình là như nhau theo cả hai chiều. Sự tạm dừng 0.1 giây để tăng tính thuận tiện ở đáy rãnh. Block N220 còn có thể được lập trình theo chế độ *số gia*:

```
N220 G01 U-0.162 W-0.081 F0.004
N221 G04 X0.1
N222 U0.162 W0.081 F0.04
```

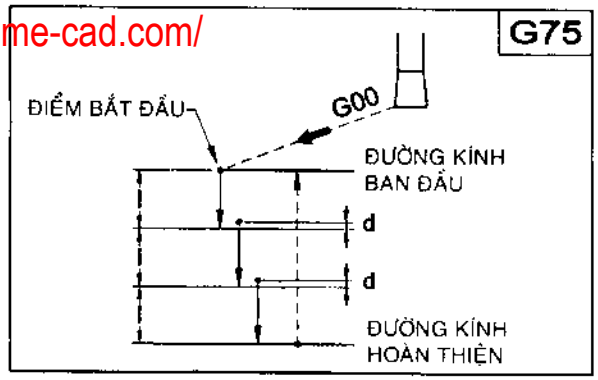
CHU KỲ GIA CÔNG RÃNH

Các bộ điều khiển Fanuc trên máy tiện có hai chu kỳ lặp G74 và G75 có thể sử dụng cho sự cắt gọt gián đoạn theo một trục. Các định dạng lập trình của cả hai chu kỳ đã được nêu trong Chương 34. Chu kỳ G74 được dùng để cắt gọt theo trục Z, chủ yếu là khoan lỗ gián đoạn, G75 được dùng để cắt theo trục X, chủ yếu để gia công rãnh đơn giản.

Ứng dụng chu kỳ G75

Tuy được dùng chủ yếu để gia công rãnh, nhưng chu kỳ G75 còn có thể được áp dụng để cắt gián đoạn trên mặt đầu. Mặc dầu chu kỳ này quá đơn giản, không thể gia công các bề mặt có độ bóng cao, nhưng vẫn có các ưu điểm riêng, chủ yếu là bẻ gãy phoi trong khi cắt theo trục X, đặc biệt hữu dụng trong một số nguyên công cắt rãnh, cắt đứt, và gia công bề mặt. Ứng dụng thứ hai là gia công thô lõi trong các rãnh sâu, sau đó có thể gia công tinh bằng các phương pháp chính xác hơn.

Trong G75, sự bẻ gãy phoi được thực hiện



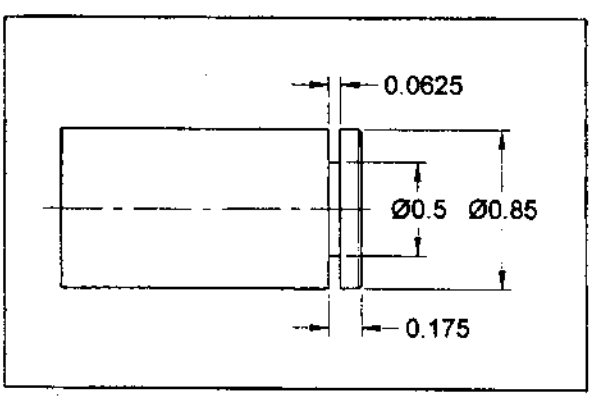
Hình 35.14. Sơ đồ biểu diễn chu kỳ G75

bằng cách lần lượt thay đổi giữa chuyển động cắt theo một chiều và chuyển động lùi nhanh theo chiều ngược lại. Điều này có nghĩa là một chuyển động cắt luôn luôn tiếp theo bằng chuyển động nhanh theo chiều ngược lại, trên cơ sở nguyên tắc *cắt vào và ra nhanh* kết hợp với khoảng hở. (Hình 35.14).

Lượng chuyển động lùi dao được thiết lập bên trong chu kỳ và được xác lập bằng tham số bên trong hệ thống điều khiển. Trên Hình 35.14 đó là giá trị d (trong bộ điều khiển, d thường từ 0.010 đến 0.020 inch). Hai ví dụ kế tiếp sẽ minh họa ứng dụng thực tiễn của chu kỳ cắt rãnh G75.

Cắt rãnh đơn với G75

Rãnh đơn yêu cầu tọa độ X và Z của điểm khởi đầu, đường kính rãnh hoàn tất X, và chiều sâu mỗi lần cắt I. Đối với rãnh đơn, vị trí trục X và khoảng cách R không thể lập trình. Vị trí Z được cho theo điểm khởi đầu và không thay đổi.



Hình 35.15. Ví dụ gia công rãnh đơn với chu kỳ G75 – Chương trình O3505.

Chương trình O3505 cắt rãnh đơn dựa trên Hình 35.15.

```
O3505
(G75 MỘT RÃNH)
(G20)
...
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

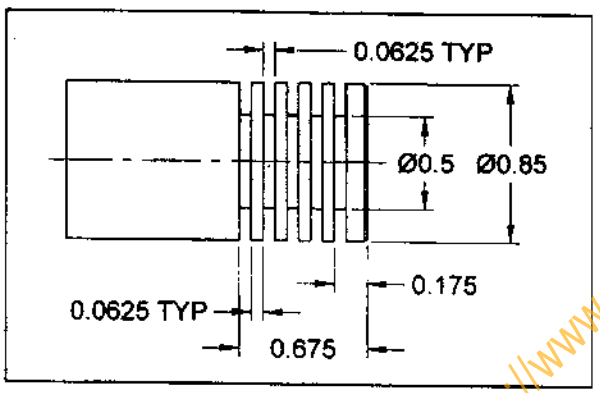
Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N43 G50 S1250 T0300 M42
N44 G96 S375 M03
N45 G00 X1.05 Z-0.175 T0303 M08
N46 G75 X0.5 I0.055 F0.004
N47 G00 X6.0 Z2.0 T0300 M09
N48 M30
%
```

Chú ý, giá trị I là .055. Đây là giá trị có ý nghĩa, đó là chiều sâu cắt rãnh đã được tính toán cẩn thận. Hành trình dao sẽ từ Ø1.050 đến Ø.500, hoặc .275 theo một phía (1.05-.50)/2 = .275, do đó sẽ có đúng năm lần cắt rãnh, mỗi lần sâu .055 (.275/5 = .055)

Gia công nhiều rãnh với G75

Lập trình gia công nhiều rãnh, sử dụng chu kỳ G75, là tương đối dễ dàng. Trong trường hợp này, khoảng cách rãnh, bước giữa các rãnh phải luôn luôn bằng nhau, nếu không bằng nhau, không thể sử dụng chu kỳ G75. Chuyên biệt khoảng hở d trong Hình 35.14 thường không được lập trình.



Hình 35.16. Gia công nhiều rãnh sử dụng chu kỳ G75. Chương trình O3506.

Chương trình O3506 gia công nhiều rãnh hở, sử dụng chu kỳ G75, dựa trên Hình 35.16.

```
O3506
(G75 NHIỀU RÃNH)
(G20)
...
N82 G50 S1250 T0300 M42
N83 G96 S375 M03
N84 G00 X1.05 Z-0.175 T0303 M08
N85 G75 X0.5 Z-0.675 I0.055 K0.125 F0.004
N86 G00 X6.0 Z2.0 T0300 M09
N87 M30
%
```

Xác lập và điều kiện đối với nhiều rãnh hoàn toàn như một rãnh. Khác biệt duy nhất là các mục nhập bổ sung trong block gọi chu kỳ G75.

Có thể sử dụng kỹ thuật này không chỉ cho nhiều rãnh, mà còn để mở rộng một rãnh có chiều rộng lớn hơn nhiều so với mảnh chắp gia công rãnh. Sự khác biệt duy nhất trong lập trình sẽ là giá trị K – khoảng cách giữa các rãnh. Nếu K lớn hơn chiều rộng lưỡi cắt, sẽ gia công nhiều rãnh riêng rẽ. Nếu K bằng hoặc nhỏ hơn chiều rộng mảnh chắp, sẽ cắt một rãnh rộng.

CÁC RÃNH ĐẶC BIỆT

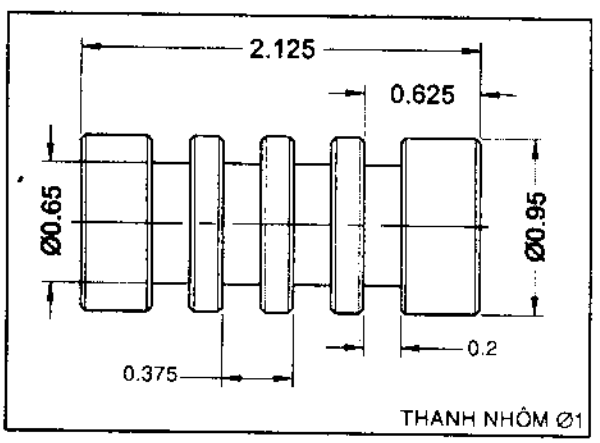
Trong thực tế có rất nhiều kiểu rãnh. Chúng chủ yếu là các rãnh có hình dạng đặc biệt, đáp ứng yêu cầu sử dụng cụ thể, chẳng hạn rãnh đáy tròn, rãnh pulley, rãnh vòng – O, ... Một số rãnh, loại tuân theo các tiêu chuẩn công nghiệp, có thể gia công với các mảnh chắp tiêu chuẩn, chẳng hạn rãnh pulley. Các nguyên tắc lập trình đối với rãnh *"phi tiêu chuẩn"* không khác biệt so với các nguyên tắc đã nêu ở phần trên.

RÃNH VÀ CHƯƠNG TRÌNH CON

Lập trình nhiều rãnh với chu kỳ G75 là phương pháp không đáp ứng các yêu cầu gia công chính xác. Hai nhược điểm chính là chất lượng rãnh và yêu cầu khoảng cách đồng đều giữa các rãnh. Phương pháp lập trình sử dụng chương trình con sẽ hiệu quả hơn nhiều.

Gia công nhiều rãnh có thể được lập trình rất hiệu quả và với độ chính xác cao bằng cách sử dụng kỹ thuật chương trình con, được trình bày trong Chương 38. Nguyên tắc cơ bản là lập trình mọi chuyển động rãnh chung trong chương trình con, và mọi chuyển động khác nhau giữa các rãnh trong chương trình chính. Theo cách đó, cùng một rãnh có thể được lặp lại theo các khoảng cố định hoặc biến thiên, tùy theo yêu cầu.

Trên Hình 35.17 là ví dụ đơn giản về lập trình nhiều rãnh, sử dụng chương trình con. Chỉ sử dụng hai dao cắt – dao tiện để vạt mặt



Hình 35.17. Lập trình nhiều rãnh sử dụng chương trình con – O3507 là chương trình chính và O3557 là chương trình con.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

đầu và tiện tinh đạo cắt đứt rộng 0.125 inch để gia công bốn rãnh, vạt góc, và cắt đứt chi tiết đã hoàn tất. Nguyên công cắt đứt được trình bày trong Chương 36 và chương trình con trong Chương 38. Bạn hãy lưu ý, mọi chuyển động dao liên quan đến vị trí rãnh được lập trình trong chương trình chính O3507, và mọi chuyển động liên quan đến cắt rãnh được lập trình trong chương trình con O3557. Trong ví dụ này sử dụng khoảng cách đều giữa các rãnh.

```
O3507 (GRV W/SUB-PROG)
(T01 - 55 DEGREE DIAMOND INSERT)
N1 G20 T0100
N2 G96 S500 M03
N3 G00 X1.2 Z0 T0101 M08
N4 G01 X-0.07 F0.006                (FACE OFF)
N5 G00 Z0.1
N6 G42 X0.7                  (START OF CHAMFER)
N7 G01 X0.95 Z-0.025 F0.003          (CHAMFER)
N8 Z-2.285                           (TURN OD)
N9 U0.2 F0.03
N10 G00 G40 X4.0 Z4.0 T0100 M09
N11 M01

(T05 - 0.125 PART-OFF TOOL)
N12 G50 S2500 T0500
N13 G96 S500 M03
N14 G00 Z-0.5875 T0505 M08          (POS-GRV1)
N15 X1.0
N16 M98 P3657                       (CUT GRV 1)
N17 G00 W-0.375 M98 P3657           (CUT GRV 2)
N18 G00 W-0.375 M98 P3657           (CUT GRV 3)
N19 G00 W-0.375 M98 P3657           (CUT GRV 4)
N20 G00 Z-2.285          (OPEN UP FOR PART-OFF)
N21 G01 X0.8 F0.006
N22 X1.1
N23 G00 X1.0 Z-2.2        (CHAMFER BACK START)
N24 G01 X0.9 Z-2.25 F0.003           (CHAMFER)
N25 X-0.02 F0.005                   (PART-OFF)
N26 G00 X1.2                          (CLEAR)
N27 G40 X4.0 Z4.0 T0500 M09
N28 M30
%

O3557 (SUB-PROG FOR 03607)
N1 G01 X0.66 F0.004          (FEED TO ROUGH OD)
N2 G00 X1.0                          (CLEAR OUT)
N3 W-0.0875               (SHIFT TO LEFT CHFR)
N4 G01 X0.9 W0.05 F0.002             (LEFT CHFR)
N5 X0.66 F0.004           (FEED TO ROUGH OD)
N6 X1.0 W0.0375 F0.03       (BACK TO START)
N7 W0.0875               (SHIFT TO RIGHT CHFR)
N8 X0.9 W-0.05 F0.002              (RIGHT CHFR)
N9 X0.65 F0.004           (FEED TO FINISH OD)
N10 W-0.075                     (SWEEP BOTTOM)
N11 X1.0 W0.0375 F0.03      (BACK TO START)
N12 M99                     (RETURN TO MAIN)
%
```

Ví dụ này kết thúc Chương 35 trình bày phương pháp lập trình gia công rãnh. Tuy gia công rãnh là nguyên công tương đối đơn giản, nhưng khi lập trình cần có các chú ý đặc biệt.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

36 CẮT ĐỨT

Cắt đứt, là nguyên công cắt gọt phổ biến trên máy tiện, thường sử dụng đồ gá cấp phôi thanh. Trong khi cắt đứt, dụng cụ cắt (dao cắt đứt) tách chi tiết hoàn tất ra khỏi phôi thanh. Chi tiết hoàn tất sau khi cắt đứt sẽ rơi xuống dưới, thường là hộp đặc biệt để tránh hư hỏng.

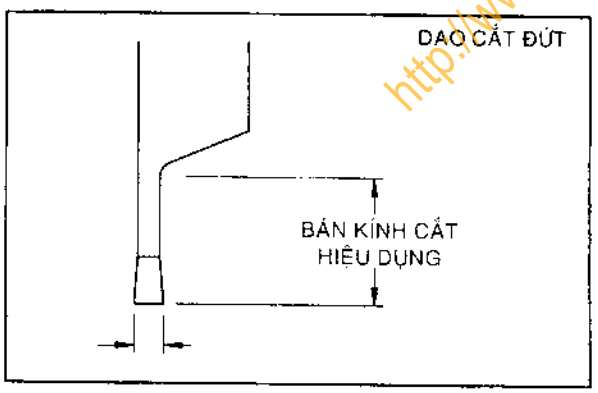
QUY TRÌNH CẮT ĐỨT

Lập trình quỹ đạo dao cắt đứt tương tự quy trình tiện rãnh. Mục đích của cắt đứt hơi khác, do phải tách chi tiết ra khỏi vật liệu phôi, thay vì tạo rãnh với kích thước và chất lượng xác định. Phôi thường là thanh tròn dài 8, 10, 12 feet (2-4 mét)

Hai vấn đề quan trọng trong cắt đứt, cũng như trong tiện rãnh, là thoát phoi và chất làm nguội.

Dao cắt đứt

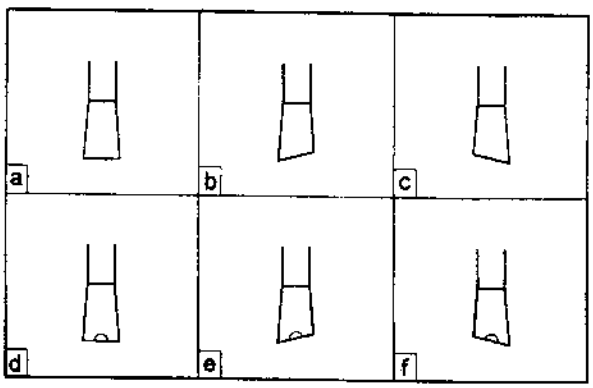
Nguyên công cắt đứt sử dụng dao cắt đặc biệt, được gọi là dao cắt đứt. Dao cắt đứt có cấu tạo tương tự dao tiện (cắt) rãnh, nhưng có một khác biệt cơ bản. *Chiều dài* của lưỡi cắt lớn hơn nhiều so với dao tiện rãnh (Hình 36.1).



Hình 36.1. Dao cắt đứt – cấu hình đầu cắt

Ở đầu cán (thép) thường là mảnh chắp carbide, với các góc thoát trên cả hai phía. Đầu cắt của dao có nhiều cấu hình khác nhau, luôn luôn ở đỉnh của phần carbide. Các cấu hình đầu dao phổ biến được nêu trên Hình 36.2.

Trên Hình có hai dạng thiết kế mảnh chắp – dạng không có vết lõm (a, b, và c), và dạng có vết lõm gần lưỡi cắt (d, e, f). Vết lõm ở phần giữa lưỡi cắt sẽ làm biến dạng phoi, do đó phoi dễ cuộn lại. Kết quả là phoi hẹp hơn chiều rộng



Hình 36.2. Dao cắt đứt – các cấu hình đỉnh cắt

cắt, không bị kẹt trong rãnh, kéo dài tuổi bền dụng cụ cắt.

Ngoài ra bạn hãy lưu ý góc hơi nghiêng ở các lưỡi cắt b, c, e, và f. Góc này giúp kiểm soát hình dạng và kích cỡ phần lõi còn lại trên mặt chi tiết khi tách khỏi thanh phôi, kể cả kích cỡ vành còn lại trên chi tiết. Tuy từng thiết kế đều có ứng dụng riêng, nhưng có lẽ lựa chọn đa năng nhất là dạng f, đặc biệt khi cắt các đường kính lớn. Khác với các kiểu gia công khác, phoi cắt khi cắt đứt phải *cuộn lại*, nhưng không gãy. Mảnh chắp với vết lõm hoặc thiết kế tương tự rất thích hợp cho mục đích này.

Nói chung, các nhà lập trình thường chỉ dùng một dao cắt đứt cho toàn bộ công việc. Họ chọn dao cắt đứt đủ dài cho đường kính thanh phôi cực đại và để dao lắp lâu dài trong ổ dao, kể cả cho các đường kính nhỏ. Lý do đơn giản là tiết kiệm thời gian. Điều này là đúng trong một số trường hợp, nhưng vẫn có nhược điểm. Dao cắt đứt dài thường có mảnh chắp rộng hơn loại dao ngắn, để tăng độ bền và độ cứng vững. Khi phôi lớn, cần dùng dao cắt đứt dài, với mảnh chắp tương đối rộng. Nếu dao này được dùng cho các chi tiết ngắn, chẳng hạn phôi ống với thành mỏng, sự cắt đứt sẽ không chuẩn và lãng phí vật liệu. Dao cắt đứt ngắn với mảnh chắp hẹp sẽ phù hợp hơn.

Cần cung cấp chất làm nguội ở lưỡi cắt, tương tự khi tiện rãnh. Dầu hòa tan là chất làm nguội thích hợp do có cả khả năng làm nguội và bôi trơn. Hỗn hợp thường dùng là một phần dầu hòa tan và 15-20 phần nước. Bạn cần bảo đảm chất làm nguội được cung cấp với áp suất toàn phần, đặc biệt là khi cắt các đường

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

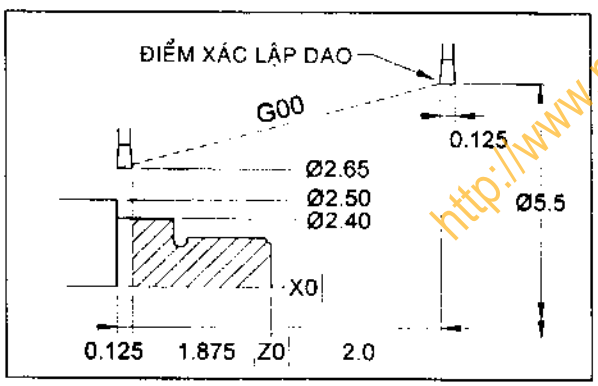
Nhiều tài liệu hay có tại : http://www.me-cad.com/

kính lớn. Áp suất này giúp chất lắm người tràn tới lưỡi cắt và đẩy phoi ra ngoài, do phoi có thể tích tụ trong rãnh cắt.

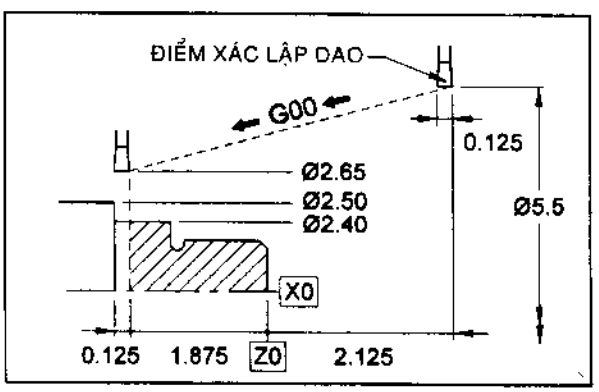
Chuyển động tiếp cận của dao

Bước thứ nhất trong lập trình quỹ đạo dao là chọn dao cắt đứt có đủ khả năng cắt hoàn toàn chi tiết ra khỏi thanh phôi. Quyết định kế tiếp là chọn chiều rộng mảnh chắp và vị trí quy chiếu (chuẩn) dao. Dao cắt đứt quá ngắn sẽ không vươn đến đường tâm trục chính một cách an toàn. Dao quá dài có thể sẽ không đủ cứng vững, gây ra rung động, thậm chí bị gãy trong khi cắt. Chiều rộng mảnh chắp cũng quan trọng đối với các điều kiện cắt. Chiều rộng dao tỷ lệ thuận với khả năng cắt sâu.

Sự lựa chọn điểm quy chiếu (chuẩn) dao cắt đứt cũng theo các nguyên tắc như dao tiện rãnh. Sẽ thuận lợi khi có cùng phía cho cả tiện rãnh và cắt đứt, nhằm duy trì sự gá lắp ổn định. Các chương trình sau đây sẽ minh họa sự khác biệt giữa điểm quy chiếu dao theo bên *trái* và bên *phải* đính dao. Hình 36.3 minh họa chương trình O3601 và Hình 36.4 là chương trình O3602. Trong cả hai trường hợp, zero chương trình là mặt đầu của chi tiết hoàn tất.



Hình 36.3. Tiếp cận dao cắt – quy chiếu dao theo bên trái – chương trình O3601.



Hình 36.4. Tiếp cận dao cắt đứt – quy chiếu dao theo bên phải – chương trình O3602

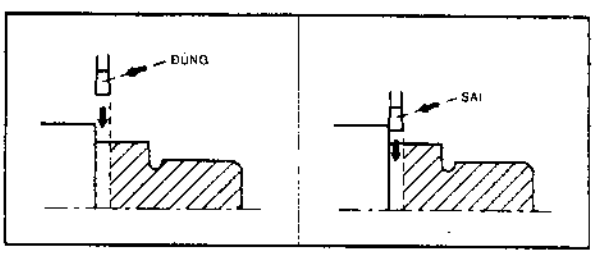
Trong các ví dụ này, vị trí thay dao và các kết quả cuối cùng là đồng nhất. So sánh hai chương trình cho thấy các giá trị trục X không thay đổi, nhưng các giá trị trục Z là khác nhau (các block N122 và N125). Điều này phản ánh phía cắt của đỉnh dao.

```
O3601 (CẮT ĐỨT/LƯỠI CẮT BÊN TRÁI)
...
N120 G50 S1250 T0800 M42
N121 G96 S350 M03
N122 G00 X2.65 Z-2.0 T0808 M08
N123 G01 X-0.03 F0.004
N124 G00 X2.65 M09
N125 X5.5 Z2.0 T0800
N126 M30
%
```

Ví dụ này phù hợp với các đề nghị đã nêu về gia công rãnh chính xác. Xác lập vị trí quy chiếu dao theo *bên trái* của dao sẽ dễ dàng hơn cho người vận hành CNC. Nếu có lý do chính đáng, bạn hãy xác lập điểm quy chiếu dao theo bên phải và chương trình sẽ dựa theo Hình 36.4.

```
O3602 (CẮT ĐỨT/LƯỠI CẮT BÊN PHẢI)
...
N120 G50 S1250 T0800 M42
N121 G96 S350 M03
N122 G00 X2.65 Z-1.875 T0808 M08
N123 G01 X-0.03 F0.004
N124 G00 X2.65 M09
N125 X5.5 Z2.125 T0800
N126 M30
%
```

Nhược điểm của xác lập dao theo bên trái là chiều rộng mảnh chắp luôn luôn cộng vào vị trí Z trong chương trình. Trong ví dụ thứ hai, chiều dài hoàn tất của chi tiết được sử dụng trực tiếp, nhưng có thể xảy ra va chạm với mâm cặp hoặc giá đỡ quay. Bạn cần cẩn thận khi chọn vị trí tiếp cận dao theo trục X, hãy lập trình dao *phía trên* đường kính phôi, dù trước đó, các nguyên công tiện đã cắt gọt xong. Hình 36.5 minh họa sự tiếp cận đúng và sai của dao cắt đứt.



Hình 36.5. Sự tiếp cận đúng và sai của dao đến đường kính phôi.

Lượng dư gia công

Nguyên công cắt đứt, trong nhiều trường hợp không phải là bước gia công cuối cùng để hoàn tất chi tiết. Nói chung, cắt đứt có thể chỉ hoàn tất nguyên công thứ nhất, chi tiết còn tiếp

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tục định kích cỡ chi tiết gia công qua nhiều bước khắt. Trong trường hợp đó, cần để lại lượng dư để tiếp tục gia công tinh. Lượng dư ở mặt đầu thường khoảng .010 đến .020 inch (0.2 ÷ 0.4 mm). Theo đó, block N122 trong cả hai chương trình cần phải thay đổi, ví dụ từ Z-2.0 sang Z-2.01 trong chương trình O3601, và từ Z-1.875 sang Z-1.895 trong chương trình O3602.

Mục nhập chương trình kế tiếp cần chú ý là giá trị X trong block N122, trong ví dụ là X2.65, để lại khoảng hở thực .125 phía trên ∅2.400. Nếu giá trị này có vẻ quá lớn, bạn cần suy nghĩ kỹ trước khi quyết định. Vì các lý do an toàn, bạn phải luôn xét đường kính phôi *thực*. Trong ví dụ này, đường kính thanh phôi là 2.500 inch và khoảng hở thực sẽ là .075 inch theo từng phía của phôi.

Chuyển động trả dao

Vấn đề an toàn tiếp theo trong lập trình dao cắt đứt là phương pháp đưa dao về vị trí thay dao, khi hoàn tất sự cắt đứt. Có vẻ rất hấp dẫn nếu thay hai block chương trình N124 và N125 bằng một block, sau đó đưa dao trở về vị trí thay dao ngay sau khi cắt đứt:

```
N124 G00 X5.5 Z2.0 (HOẶC Z2.125) T0800 M09
```

Như vậy, ngay sau khi được cắt rời, chi tiết sẽ rơi xuống hộp chứa, và có thể bớt một block trong chương trình. *Nhưng bạn không nên làm điều đó*, do có thể rất nguy hiểm. Chi tiết *sẽ* được dao cắt rời khỏi thanh phôi – và *sẽ phải* rơi xuống hộp chứa - nhưng toàn bộ điều này có thực sự xảy ra? Nhiều lý do có thể làm cho chi tiết *không* được cắt rời hoàn toàn. Kết quả sẽ là dao bị gãy, chi tiết bị xước, thậm chí có thể làm hư hại máy.

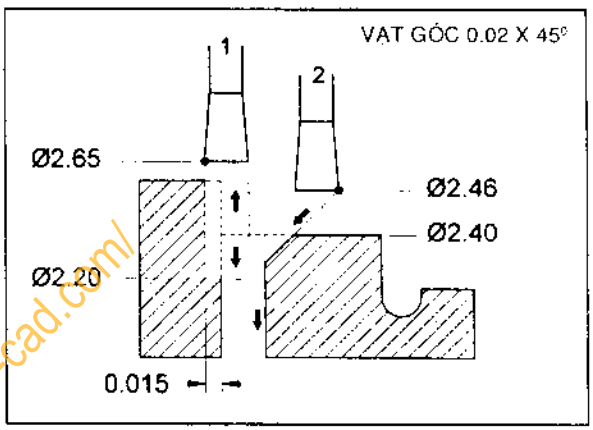
Luôn luôn trả dao về theo trục X trước, và luôn luôn phía trên đường kính phôi.

Cắt đứt có vạt góc

Khi quy trình gia công hoàn tất với dao cắt đứt, sẽ yêu cầu chất lượng bề mặt tốt nhất có thể. Một trong các yêu cầu đó là xử lý các góc sắc (vạt góc hoặc bo tròn). Trong ví dụ này, góc sắc là giao điểm của X2.4 và Z-1.875. Nếu dao tiện không thể vạt góc trong nguyên công tiện, dao cắt đứt có thể là lựa chọn tốt. Hầu hết các dao cắt đứt đều không được thiết kế để cắt vát (theo trục Z), nhưng sự vạt góc chỉ cắt gọt một lượng nhỏ vật liệu trong phạm vi khả năng của dao. Bạn cần tránh vạt góc rộng hơn 75% chiều rộng mảnh chắp hoặc sử dụng vài lần cắt, nếu cần thiết. Vạt góc cần được thực hiện *trước khi cắt rời* và phải cắt từ ngoài vào trong, *không* được cắt từ trong ra ngoài. Kỹ thuật lập trình để vạt góc trong khi cắt đứt được tóm tắt theo các bước sau:

- Định vị dao theo trục Z xa hơn so với vị trí cắt đứt thông thường
- Khởi động sự cắt đứt và dừng lại ngay dưới đường kính nơi sẽ kết thúc sự vạt góc.
- Trở lại đường kính khởi đầu và đưa dao đến vị trí bắt đầu vạt góc.
- Lập trình vạt góc trong một block và tiếp tục cắt đứt trong block kế tiếp.

Để minh họa kỹ thuật lập trình này, bạn hãy nghiên cứu chương trình O3603 và Hình 36.6 – điểm quy chiếu dao ở bên trái, vạt góc là .020 inch × 45^0.



***Hình 36.6.** Vạt góc với dao cắt đứt – chương trình O3603.*

```
O3603 (CẮT ĐỨT CÓ VẠT GÓC)
(G20)
...
N120 G50 S1250 T0800 M42
N121 G96 S350 M03
N122 G00 X2.65 Z-2.015 T0808 M08
N123 G01 X2.2 F0.004
N124 X2.46 F0.03
N125 Z-1.95                    (LEFT SIDE OF TOOL)
N126 U-0.1 W-0.05 F0.002
N127 X-0.03 F0.004
N128 G00 X2.65
N129 X5.5 Z2.0 T0800 M09
N130 M30
%
```

Trong block N122, dao được định vị với .015 inch phía sau vị trí Z-2.0. Block N123 chỉ tạm thời gia công rãnh (đến ∅2.200). Block kế tiếp, N124, là chuyển động ra khỏi rãnh, đến đường kính bắt đầu vạt góc (∅2.460). Trong block N125, dao dịch chuyển theo trục Z, đến vị trí bắt đầu vạt góc. Giá trị 1.950 được tính như sau:

1.875 - .020 - .030 + .125 = 1.250

Giá trị 1.875 là mặt cuối của chi tiết (theo bản vẽ), .020 là giá trị vạt góc; .030 là khoảng

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

hở, về kích thước cần máy mòn chặt. Chú ý điều chỉnh chiều rộng dao .125, để duy trì điểm quy chiếu dao theo *bên trái* lưỡi cắt trong khi cắt thực sự với phần lưỡi bên phải. Block N126 là vạt góc, sử dụng chế độ số gia, có thể giảm bớt vài tính toán. Nếu dùng chế độ tuyệt đối, block N126 sẽ như sau:

```
N126 X2.36 Z-2.0 F0.002
```

Ngoài ra, cần chú ý, chỉ giảm tốc độ cắt khi vạt góc, để bảo đảm độ bóng cao. Sự giảm tốc độ cắt chỉ có ý nghĩa đối với các vạt góc rất nhỏ. Phần còn lại của chương trình không thay đổi.

Trong một số trường hợp, có thể dùng hai dao để cắt đứt. Gá lắp hai dao phải chính xác. Dao cắt rãnh nhỏ và cứng vững có thể tạo rãnh ban đầu và vạt góc, dao thứ hai sẽ cắt phần còn lại. Khi hoàn toàn cắt rời, thanh phôi chìa ra từ trục chính sẽ có bậc nhỏ. Bạn cần lập trình cắt (vạt) mặt đầu cho từng chi tiết kế tiếp để cắt bỏ phần bậc này.

Tránh hư hại chi tiết

Khi chi tiết được cắt rời khỏi phôi thanh sẽ rơi xuống. Nếu bị va đập, chi tiết có thể bị hư hại hoặc có các vết trầy xước. Để tránh khả năng hư hỏng, người vận hành máy tiện CNC cần đặt thùng hoặc bể chứa chất làm nguội để hứng chi tiết rơi xuống. Cách thứ hai là lấy dao cắt đứt ra xa đường tâm, đủ xa để chưa cắt rời chi tiết. Sau đó, khi máy dừng lại, người vận hành CNC có thể gõ nhẹ và lấy chi tiết bằng tay.

Trong mọi trường hợp, cần luôn luôn tuân thủ các quy định về an toàn trong xưởng cơ khí.

Không chạm vào chi tiết trong khi chương trình đang hoạt động hoặc trục chính đang quay

Giải pháp tốt nhất để tránh hư hại chi tiết là máy tiện CNC được trang bị đồ gá chuyên dùng để lấy chi tiết, có thể đi kèm với máy, hoặc tự thiết kế và chế tạo theo yêu cầu cụ thể.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

37 GIA CÔNG REN MỘT LƯỠI CẮT

Gia công ren là quy trình công nghệ được dùng để tạo ra rãnh xoắn với hình dạng xác định, thường trên mặt trụ. Mục đích chính của ren là nối ghép hai chi tiết với nhau, không bị hư hại trong khi lắp và tháo. Các ứng dụng phổ biến của ren thuộc bốn nhóm chính:

- + Chi tiết lắp ghép ...vít và đai ốc
- + Dụng cụ đo ...vi kế
- + Truyền chuyển động ...trục vít, thấu kính camera
- + Tăng ngẫu lực ...vít nâng, con đội

Cắt ren là quy trình chế tạo rất đa năng. Có hai nhóm chính: cắt ren cho vật liệu kim loại, và tạo ren bằng khuôn cho chất dẻo. Phương pháp tạo ren bằng khuôn cho chất dẻo chiếm lĩnh ngành công nghiệp nhựa. Số lượng chai nhựa, đựng nước uống, mỹ phẩm, dầu gội, ..., sử dụng hàng ngày là rất lớn.

Trong lĩnh vực gia công kim loại, có thể tạo ren bằng một trong các nhóm phương pháp sau:

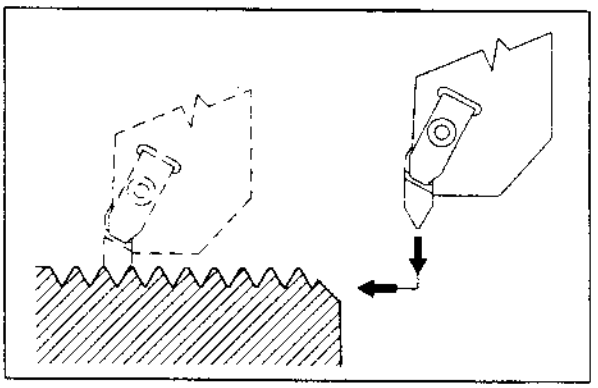
- Cán ren hoặc tạo hình ren.
- Tarô và bàn ren.
- Phay ren.
- Mài ren.
- Tiện ren với dao một lưỡi cắt.

Đối với nhà lập trình CNC, các lĩnh vực cần quan tâm thường là tarô ren, phay, và tiện ren. Các phương pháp tarô ren được trình bày trong Chương 24 và 25, phay ren được nêu trong Chương 44. Chương này chỉ đề cập các phương pháp lập trình *gia công ren một lưỡi cắt*.

GIA CÔNG REN TRÊN MÁY TIỆN CNC

Máy tiện CNC có thể gia công ren chất lượng rất cao kết hợp các nguyên công tiện và doa, trong *một xác lập* gá lắp chi tiết gia công. Đây là tính năng rất hấp dẫn đối với các nhà chế tạo và các xưởng cơ khí.

Tiện ren một lưỡi cắt – còn gọi là *gia công ren một lưỡi* – sử dụng cán dao tương tự các loại dao khác, nhưng có mảnh chắp tiện ren đặc biệt, với một, hai, hoặc ba đỉnh. Nói chung, hình dạng và kích cỡ mảnh chắp tiện ren phải tương ứng hình dạng và kích cỡ ren hoàn tất (Hình 37.1).



Hình 37.1. *So sánh dạng ren và hình dạng dao tiện ren*

Theo định nghĩa, gia công ren một lưỡi cắt là quy trình cắt gọt rãnh xoắn có hình dạng đặc biệt có bước tiến đồng nhất theo vòng quay trục chính. Dạng ren chủ yếu được xác định bằng hình dạng và vị trí gá lắp dao cắt. Tính đồng nhất của bước tiến được điều khiển bằng tốc độ cắt lập trình.

Dạng ren

Dạng ren phổ biến nhất trong lập trình CNC là ren V với góc 60^0. Có nhiều dạng ren V trong sản xuất, gồm cả ren hệ Anh và hệ mét. Các dạng khác gồm ren hình thang, ren Acme, ren vuông, ren tròn, ... Ngoài các dạng ren tương đối phổ biến, còn có các ren chuyên dùng, chẳng hạn trong công nghiệp ôtô, hàng không, dầu mỏ, ... Dạng ren có thể trên mặt trụ, mặt côn, ren trong, ren ngoài. Ren có thể được cắt trên mặt phẳng (ren cuộn) kể cả trên bề mặt tròn. Ren có thể có một hoặc nhiều đầu mối, chiều phải hoặc chiều trái, bước không đổi hoặc biến thiên, ...

Gia công ren

Phần này đưa ra danh sách các nguyên công cắt gọt ren có thể lập trình trên máy tiện CNC. Nhiều nguyên công đòi hỏi kiểu dao tiện ren đặc biệt, một số nguyên công chỉ có thể được lập trình nếu hệ điều khiển được trang bị các tính năng đặc biệt.

- Ren bước tiến không đổi.
- Ren bước tiến biến thiên.
- Ren trong và ren ngoài.
- Ren trụ (ren thẳng)

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Ren bề mặt
- Ren phải (R/H) và ren trái (L/H)
- Ren bề mặt (ren cuộn)
- Ren một đầu mối
- Ren nhiều đầu mối
- Ren tròn
- Ren nhiều block

Trong gia công ren dường như có vô số các phối hợp và các khả năng, nhưng kiến thức lập trình và kinh nghiệm trong một nhóm ren sẽ là vô giá đối với các nhóm khác. Chương trình gia công ren chuẩn dựa trên kiến thức vững chắc về các nguyên tắc gia công ren cơ bản.

THUẬT NGỮ VỀ REN

Gia công ren, có nội dung tương đối rộng, đủ để trình bày trong một cuốn sách, với các thuật ngữ chuyên môn được dùng trong nhiều tài liệu kỹ thuật và công nghệ. Nhà lập trình CNC và người vận hành cần hiểu rõ ý nghĩa của các thuật ngữ đó.

Dưới đây là một số thuật ngữ về ren và gia công ren:

- GÓC REN

  ...là góc giữa hai cạnh (sườn) của ren, đo trên mặt phẳng trục

- ĐỈNH REN

  ...là bề mặt trên cùng của ren, giao giữa hai cạnh (sườn) ren

- CHIỀU SÂU REN

  ...nói chung đây là khoảng cách giữa đỉnh và chân ren, đo vuông góc với trục (trong lập trình, chiều sâu được coi là giá trị đo được theo một phía ren)

- REN NGOÀI

  ...là ren được gia công trên mặt ngoài của chi tiết, ví dụ, bu-lông

- REN TRONG

  ...là ren được gia công trên mặt trong của chi tiết, ví dụ, đai ốc

- GÓC XOẮN

  ...là góc tạo thành giữa đường xoắn của ren tại đường kính bước và mặt phẳng vuông góc với trục.

- HƯỚNG REN

  ...là khoảng cách dao gia công ren tiến dọc trục trong một vòng quay trục chính. Hướng ren luôn luôn xác định tốc độ cắt ren, có thể là hằng hoặc biến.

- ĐƯỜNG KÍNH CHÍNH

  ...là đường kính lớn nhất của ren

- ĐƯỜNG KÍNH PHỤ

  ...là đường kính nhỏ nhất của ren

- REN NHIỀU ĐẦU MỐI

  ...là ren có hơn một đầu mối, dịch chuyển theo giá trị bước ren

- BƯỚC REN

  ...là khoảng cách từ điểm chuyên biệt của một ren đến điểm tương ứng của ren kế cận, khi đo song song với trục máy.

- ĐƯỜNG KÍNH BƯỚC

  ...trên ren thẳng, đường kính bước là đường kính tưởng tượng, "bề mặt đi qua các ren tại những điểm sao cho tạo thành chiều rộng của các ren đó bằng chiều rộng của các khoảng cách được cắt bằng bề mặt trụ".

- CHÂN REN

  ...là bề mặt đáy của ren nối các cạnh (sườn) của hai ren kế cận

- REN CUỘN

  ...còn gọi là ren mặt, ren được gia công theo trục X thay vì theo trục Z

- ĐỘ DỊCH CHUYỂN

  ...trong gia công ren nhiều đầu mối, đây là khoảng cách dao cắt dịch chuyển để cắt đầu mối kế tiếp; khoảng cách này luôn luôn bằng bước ren. Số dịch chuyển luôn luôn bằng số đầu mối trừ một.

- REN CÔN

  ...là ren có đường kính bước tăng hoặc giảm theo tỷ số không đổi (chẳng hạn ren ống)

- TPI

  ....theo đơn vị Anh, số ren trên chiều dài một inch (1/bước ren) – ren hệ mét được định nghĩa theo bước – đường lượng TPI (threads per inch) không ứng dụng được.

QUY TRÌNH GIA CÔNG REN

Gia công ren là một trong các nhiệm vụ lập trình tự động hóa phổ biến nhất trong xưởng cơ khí hiện đại, một trong các nguyên công tương đối khó trên máy tiện CNC. Thoạt nhìn, dường như sự lập trình quỹ đạo có các tham số được xác định rõ ràng là tương đối dễ thực hiện. Tuy nhiên, các ứng dụng thực tiễn hơi khác so với lý thuyết.

Điều làm cho gia công ren trở nên khó khăn là sự sử dụng dao cắt ren. Dao tiện ren một lưỡi cắt có cấu trúc khác với các loại dao khác. Tuy các dao cũng được lắp vào ổ dao như các dụng cụ cắt khác, nhưng lưỡi cắt có các đặc trưng riêng. Dao gia công ren không chỉ *cắt gọt* mà còn *tạo ra* hình dạng ren. Nói chung, lưỡi dao tiện ren có hình dạng của ren hoàn chỉnh. Gá lắp dao tiện ren trong ổ dao có thể vuông

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

góc, hoặc song song với đường tâm xác định theo góc ren, so với đường tâm trục chính. Điều quan trọng là dao được lắp *vuông góc* trong ổ dao. Ngay cả độ lệch góc rất nhỏ cũng ảnh hưởng xấu đến ren hoàn tất.

Các bước tiện ren

Khi so sánh lưỡi dao tiện ren với mảnh chắp hình thoi 80^0 dùng để tiện tinh, bạn sẽ thấy các khác biệt:

Bán kính dao:

Tiện ren = đỉnh rất sắc
Tiện = trung bình .0313 (0.8mm)

Góc dao:

Tiện ren = 60^0 và phần đỡ yếu
Tiện = 80^0 và phần đỡ mạnh

Tốc độ cắt:

Tiện ren = đến .25 in/rev (6.5 mm/rev) hoặc cao hơn
Tiện = .015 in/rev đến .030 in/rev
(0.4 mm/rev – 0.6mm/rev)

Chiều sâu cắt

Tiện ren = nhỏ
Tiện = trung bình đến lớn

So sánh này cho thấy ngay cả ren bước mịn cũng không thể gia công với một đường cắt ren. Nếu chỉ áp dụng một đường cắt ren, chất lượng ren sẽ rất xấu, thậm chí không sử dụng được, đồng thời làm giảm tuổi bền của dụng cụ cắt.

Phương pháp thông dụng là cắt ren với nhiều đường, tăng dần chiều sâu ren cho từng đường cắt. Đối với tiện ren nhiều đường cắt, sự quay trục chính phải đồng bộ với sự khởi đầu của từng đường cắt, sao cho từng chiều sâu ren ở cùng vị trí trên mặt trụ ren. Ren chất lượng sẽ hoàn tất khi đường cắt cuối cùng tạo ra kích cỡ ren thích hợp, hình dạng, độ bóng và dung sai theo yêu cầu. Do tiện ren một lưỡi cắt gồm nhiều đường cắt cho một ren, các nhà lập trình phải hiểu rõ từng đường cắt đó.

Trong lập trình, cấu trúc từng đường cắt (áp dụng cho ren thẳng):

Chuyển động 1	Từ vị trí khởi đầu, chuyển dao đến đường kính ngoài theo chế độ chuyển động nhanh
Chuyển động 2	Tiện ren – cắt ren một trục (tốc độ cắt bằng giá trị dẫn hướng)
Chuyển động 3	Lùi dao nhanh ra khỏi ren
Chuyển động 4	Lùi dao nhanh về vị trí khởi đầu

Quá trình chuyển động dao bốn–bước sẽ được gộp trong các xem xét cơ bản về chương trình CNC.

Chuyển động 1

Trước bước thứ nhất, dao phải chuyển động từ vị trí phân độ đến gần chi tiết gia công. Đây là chuyển động nhanh, bạn cần tính toán chính xác các tọa độ XY của vị trí này. Các tọa độ đó được gọi là *vị trí khởi đầu gia công ren*, do chúng xác định nơi sự cắt ren sẽ bắt đầu và sẽ trở về sau khi cắt. Vị trí khởi đầu phải được chọn hơi xa chi tiết nhưng gần ren, giao điểm của khoảng hở trục X và khoảng hở trục Z.

Chuyển động dao thứ nhất liên quan trực tiếp với ren, là chuyển động *từ* vị trí khởi đầu *đến* đường kính cắt ren. Do một đường cắt không thể cắt ren đến chiều sâu toàn phần, chiều sâu này phải được chia thành các đoạn nhỏ. Từng đoạn chiều sâu đó phụ thuộc vào kiểu dao, vật liệu gia công, và độ cứng vững gá lắp. Chuyển động tiếp cận này được lập trình trong chế độ chuyển động nhanh.

Chuyển động 2

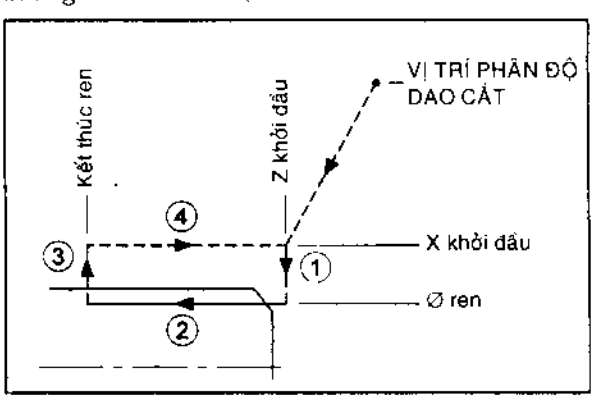
Khi dao đến đường kính cắt ren với chiều sâu đã biết, *chuyển động thứ hai* sẽ bắt đầu. Đây là sự cắt ren thực tế, theo tốc độ cắt đã xác định và chỉ khi tốc độ trục chính *đồng bộ* với tốc độ cắt ren. Không cần các biện pháp đặc biệt để duy trì tính đồng bộ – trong chế độ cắt ren, sự đồng bộ là tự động. Ren sẽ được cắt đến vị trí kết thúc ren đã lập trình.

Chuyển động 3

Trong *chuyển động thứ ba*, khi hoàn tất đường kính cắt ren, dao phải lùi ra khỏi ren, theo tốc độ nhanh, đến vị trí khoảng hở trục X. Vị trí dao này thường là đường kính được lập trình *phía ngoài* khu vực ren.

Chuyển động 4

Đường cắt ren hoàn tất với *chuyển động thứ tư* khi dao trở về vị trí khởi đầu theo chế độ nhanh. Mọi đường cắt còn lại đều được lập trình theo cách thức nêu trên, chỉ thay đổi đường kính cắt ren (điều khiển chiều sâu ren).



Hình 37.2. *Các bước cơ bản khi tiện ren một lưỡi cắt.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lưu ý chu kỳ lập trình *tuyến động* 2 trong chế độ cắt ren, sử dụng lệnh G tương ứng. Các chuyển động 1, 3, và 4 đều trong chế độ (nhanh) G00.

Vị trí khởi đầu

Vị trí khởi đầu của dao là *vị trí hở*. Đối với ren trên mặt trụ thẳng, khoảng hở thích hợp tối thiểu dọc theo trục X là khoảng .100 (2.5 mm) theo một phía, có thể lớn hơn khi cắt ren thô. Đối với ren côn, khoảng hở cũng tương tự nhưng ở đầu (đường kính) lớn.

Cần xét một số điều kiện đối với khoảng hở dọc theo trục Z. Khi dao tiếp xúc với vật liệu, sẽ phải tiến với đúng 100% tốc độ cắt lập trình. Do tốc độ cắt ren là tương đương với *dẫn hướng* ren, cần có thời gian để đạt đến tốc độ cắt toàn phần *trước khi* tiếp xúc với vật liệu. Cần xét tác động của gia tốc này khi chọn giá trị khoảng hở.

Khi lập trình ren thô, giá trị khoảng hở phía trước thường phải lớn hơn ren trung bình và ren mịn. Ví dụ ren với TPI 8 yêu cầu tốc độ cắt .1250 in/rev. Nếu khoảng hở trục Z quá nhỏ, quá trình gia tốc sẽ không đủ khi dao tiếp xúc với vật liệu, có thể làm cho ren bị hư. Để tránh vấn đề này, cần áp dụng nguyên tắc:

Khoảng hở trục Z đối với điểm khởi đầu phải gấp 3-4 lần chiều dài dẫn hướng (bước tiến) ren.

Đây chỉ là nguyên tắc chung, rút ra từ kinh nghiệm, sổ tay hướng dẫn đi kèm bộ điều khiển CNC thường cung cấp phương pháp tính toán khoảng hở này dựa trên lý thuyết.

Trong một số trường hợp, cần giảm khoảng hở trục Z do không gian quá chật, chẳng hạn khi ren bắt đầu ở rất gần ụ động hoặc các giới hạn máy. Do thời gian gia tốc phụ thuộc trực tiếp vào tốc độ trục chính, giải pháp duy nhất cho trường hợp này là *giảm* tốc độ trục chính (r/min) – *tốc độ cắt không cần giảm*.

Đối với các phương pháp ăn dao phức tạp, vị trí khởi đầu thay đổi tương ứng từng lần cắt với giá trị tính toán trước.

Chiều sâu và đường kính cắt ren

Đối với cắt ren côn và trụ sử dụng block trong lập trình (không dùng các chu kỳ), bạn hãy chọn đường kính cắt cho *từng* đường cắt của dao trong chương trình. Từ vị trí khởi đầu của ren, dao cắt sẽ chuyển động *hướng vào* đường tâm trục chính đối với ren ngoài và *ra xa* đường tâm chính đối với ren trong. Đường kính cắt thực đối với từng lần cắt phải được chọn không chỉ theo đường kính ren mà còn theo các điều kiện gia công.

Trong gia công ren, tải phoi trên lưỡi cắt tăng dần khi tăng chiều sâu cắt. Có thể tránh nguy hiểm cho ren và dao bằng cách duy trì *tải phoi ổn định* trên lưỡi cắt. Một phương pháp đạt được tính ổn định này là *giảm* dần chiều sâu cắt ren qua từng lần cắt, thứ hai là áp dụng phương pháp ăn dao thích hợp. Cả hai kỹ thuật cắt ren này thường được sử dụng đồng thời.

Để tính toán chiều sâu của từng đường cắt, không cần các công thức phức tạp, chỉ yêu cầu đôi chút kinh nghiệm. Mọi chu kỳ gia công ren đều có thuật toán trong bộ điều khiển, tính từng chiều sâu cắt một cách tự động. Đối với các tính toán bằng tay, bạn có thể áp dụng phương pháp sau. Biết *chiều sâu ren toàn phần* (đo theo một phía), nhà lập trình sẽ chọn *số đường cắt ren* thích hợp cho chi tiết cụ thể. Giá trị thứ hai cần xác định là chiều sâu cắt *cuối cùng*, chiều sâu hoàn tất ren. Các giá trị này thường được chọn theo kinh nghiệm.

Sau khi thiết lập ba tham số (giá trị) đó, cần phân chia chiều sâu cắt toàn phần cho từng đường cắt kể cả đường cắt cuối cùng. Bắt đầu với tính toán gần đúng, sao cho chiều sâu cắt giảm dần qua từng đường cắt nhưng vẫn không vượt quá chiều sâu ren toàn phần. Từng chiều sâu này có thể không bảo đảm tải phoi ổn định tuyệt đối, nhưng từng đường cắt vẫn trong phạm vi chấp nhận được.

Hình 37.3 minh họa ren ngoài. Đây là ren 12 TPI (12 ren/inch), nhà lập trình cần tìm chiều sâu ren, bằng cách dùng công thức tiêu chuẩn. Chiều sâu ren ngoài, biên dạng toàn phần, là 0.0511 inch, dựa trên công thức chiều sâu ren – chỉ áp dụng cho ren *ngoài*, hệ mét và ren thống nhất.

$$D = \frac{0.61343}{TPI} = 0.61343 \times P$$

Trong đó:

D = Chiều sâu ren ngoài
TPI = Số ren/inch
P = Bước ren (1/TPI)

Theo tiêu chuẩn dạng ren UN, hằng số trong công thức này là 0.64952, chiều sâu ren sẽ là 0.0541.

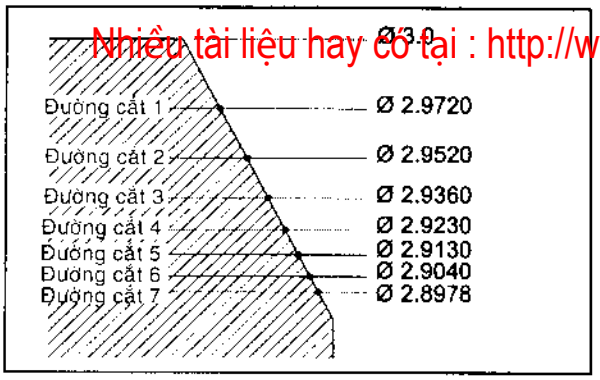
Đối với ren trong biên dạng toàn phần, công thức tính chiều sâu chỉ dùng cho ren hệ mét và AN (tiêu chuẩn quốc gia Mỹ) – giá trị D là chiều sâu *trong*:

$$D = \frac{0.54127}{TPI} = 0.54127 \times P$$

Nếu chọn bảy đường cắt ren, với đường cắt

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 37.3. Phân phối các đường kính cắt ren để tải phoi cân bằng.

cuối cùng là 0.0031 (để tiện lập trình), các chiều sâu riêng rẽ có thể được phân phối như sau:

Chiều sâu đường cắt #1 - 0.0140 Chiều sâu tích lũy = 0.0140
Chiều sâu đường cắt #2 - 0.0100 Chiều sâu tích lũy = 0.0240
Chiều sâu đường cắt #3 - 0.0080 Chiều sâu tích lũy = 0.0320
Chiều sâu đường cắt #4 - 0.0065 Chiều sâu tích lũy = 0.0385
Chiều sâu đường cắt #5 - 0.0050 Chiều sâu tích lũy = 0.0435
Chiều sâu đường cắt #6 - 0.0045 Chiều sâu tích lũy = 0.0480
Chiều sâu đường cắt #7 - 0.0031 Chiều sâu tích lũy = 0.0511

Các tính toán này sẽ được dùng khi thực sự viết chương trình. Chiều sâu ren phụ thuộc vào số ren/inch dù là ren ngoài hay ren trong. Khi tính chiều sâu ren, đường kính ren là không quan trọng. Khi biết đường kính này, các chiều sâu tính toán có thể được sử dụng để tìm đường kính của từng đường cắt ren. Ví dụ, đối với ren ngoài 3.0-12, từng đường kính sẽ được tính dựa trên kích cỡ ren danh định 3 inch:

Đường kính cắt ren # 1 3.0 – 2 × 0.0140 = 2.9720
Đường kính cắt ren # 2 3.0 – 2 × 0.0240 = 2.9520
Đường kính cắt ren # 3 3.0 – 2 × 0.0320 = 2.9360
Đường kính cắt ren # 4 3.0 – 2 × 0.0385 = 2.9230
Đường kính cắt ren # 5 3.0 – 2 × 0.0435 = 2.9130
Đường kính cắt ren # 6 3.0 – 2 × 0.0480 = 2.9040
Đường kính cắt ren # 7 3.0 – 2 × 0.0511 = 2.8978

Phương pháp tính đường kính cắt ren này không có gì sai. Điều còn thiếu là *sự kiểm tra độ chính xác*. Do từng đường kính đều được tính từ đường kính danh định, sai số tính toán là *không* tích lũy và có thể khó xác định. Phương pháp tốt hơn có lẽ là tính từng đường kính ren dựa trên tính toán *trước đó*, sử dụng một chiều sâu cắt, *thay vì* chiều sâu tích lũy – bạn hãy so sánh với kết quả từ phương pháp nêu trên:

Đường kính cắt ren # 1 3.0000 – 2 × 0.0140 = 2.9720
Đường kính cắt ren # 2 2.9720 – 2 × 0.0100 = 2.9520
Đường kính cắt ren # 3 2.9520 – 2 × 0.0080 = 2.9360
Đường kính cắt ren # 4 2.9360 – 2 × 0.0065 = 2.9230
Đường kính cắt ren # 5 2.9230 – 2 × 0.0050 = 2.9130
Đường kính cắt ren # 6 2.9130 – 2 × 0.0045 = 2.9040
Đường kính cắt ren # 7 2.9040 – 2 × 0.0031 = 2.8978

Ưu điểm của phương pháp này là khi tìm được đường kính cuối cùng (2.8978 trong ví dụ), *cộng* hai lần chiều sâu cho đường kính đó, kết quả *phải* bằng đường kính danh định của ren, (trong ví dụ là 3.000 inch):

2.8978 + 2 × 0.0511 = 3.0000

Nếu kết quả không phải là đường kính ren danh định, đã có sai số trong các tính toán. Sử dụng cả hai phương pháp và so sánh kết quả sẽ bảo đảm tính chính xác cao hơn.

Chuyển động cắt ren

Khi dao cắt đạt đến chiều sâu đường cắt ren, ren đó sẽ được cắt. Cắt gọt bắt đầu ở vị trí khoảng hở trục Z và kết thúc ở cuối ren với tốc độ cắt vẫn còn hiệu lực. Dù ren được cắt, nhưng chuyển động tuyến tính, không sử dụng lệnh G01 khi cắt ren. Nếu dùng G01, sự khởi đầu từng đường cắt sẽ *không* đồng bộ với sự khởi đầu đường cắt trước đó. Thay cho lệnh G01, cần dùng mã G chuyên biệt cho gia công ren. G32 là mã thông dụng nhất trên các bộ điều khiển Fanuc đối với gia công ren. Trong quá trình chuyển động cắt ren G32, hệ thống điều khiển tự động xóa tác dụng của chế độ vượt qua tốc độ cắt. Người vận hành CNC cần rất cẩn thận khi xác lập dao cắt ren, đặc biệt khi ren ở gần các bậc (vai) trên chi tiết. Để minh họa sự lập trình cho đến phần này, bạn hãy xem đoạn chương trình dưới đây:

```
...
N61 G00 X3.3 Z0.3 (START POINT XZ)
N62 X2.972 (THREAD DIA START)
N63 G32 Z-1.75 F0.0833 (THREAD TO END)
...
```

Lùi dao ra khỏi ren

Thời điểm ren đạt đến vị trí cuối dọc theo trục chính, dao phải lùi ngay ra khỏi vật liệu, để tránh hư hại ren. Đây là chuyển động thứ ba trong quy trình tiện ren cơ bản. Chuyển động lùi dao có thể có hai dạng - lùi thẳng theo một trục (thường là trục X), hoặc lùi từ từ theo hai trục (đồng thời dọc theo các trục XZ) (Hình 37.4).

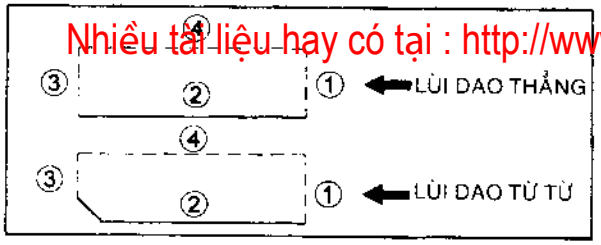
Nói chung, nên lập trình *lùi dao theo một trục* mỗi khi dao kết thúc cắt gọt trong *khoảng trống*. Đối với các ren không kết thúc trong khoảng trống, cần *lùi dao từ từ theo hai trục*. Chuyển động lùi dao từ từ tạo ra ren chất lượng tốt hơn và tuổi bền của dao dài hơn. Để lập trình lùi dao thẳng (theo một trục), chế độ cắt ren G32 phải được xóa và thay bằng chế độ chuyển động nhanh, sử dụng lệnh G00.

```
N64 G00 X3.3                    (RAPID OUT)
```

Đối với lùi dao từ từ (theo hai trục), mã G tiện ren và tốc độ cắt *phải giữ nguyên hiệu lực*.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

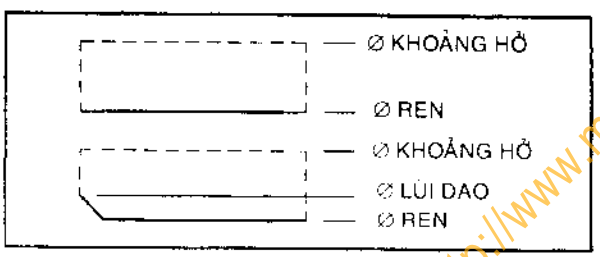


Hình 37.4. Lùi dao ra khỏi ren theo một trục và theo hai trục.

Khi hoàn tất chiều dài ren bình thường – nhưng *trước khi* lùi dao- dao tiện ren chuyển động đồng thời theo hai trục, kết thúc *phía ngoài* ren. Chiều dài bình thường khi lùi dao là 1-1.5 lần bước tiến dao (không phải bước ren) theo góc 45^0. Điều quan trọng cần chú ý là đường kính khoảng hở.

```
N64 U0.2 W-0.1                (GRADUAL PULLOUT)
N65 G00 X3.3                  (RAPID OUT)
```

Đối với ren *ngoài*, đường kính khoảng hở cách đường tâm trục chính phải luôn luôn *xa hơn* đường kính lùi ra từ từ. Đối với ren *trong*, đường kính khoảng hở cách đường tâm trục chính phải *gần hơn* so với đường kính lùi ra từ từ. (Hình 37.5).



Hình 37.5. Lùi dao ra khỏi ren và đường kính khoảng hở (ren ngoài)

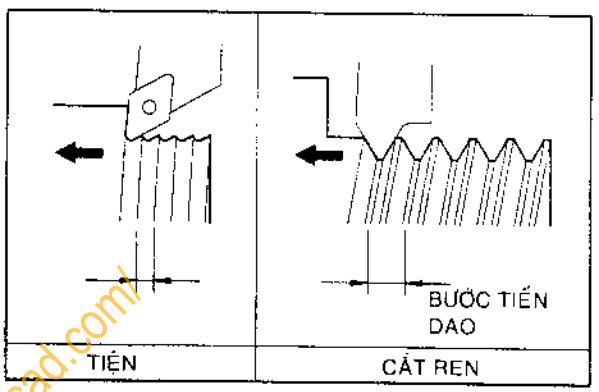
Trở về vị trí khởi đầu

Bất kể cách thức lập trình lùi dao ra khỏi ren, thẳng hay từ từ, bước cuối cùng trong quy trình tiện ren là đưa dao về vị trí khởi đầu. Chuyển động dao này hoàn toàn ở khoảng trống, do đó được lập trình theo chế độ chuyển động nhanh G00. Nói chung, chuyển động trở về vị trí khởi đầu chỉ theo một trục, thường là trục Z. Điều này là do trong hầu hết các chương trình, sự lùi dao ra khỏi ren thường đến đường kính trục X. Dưới đây là phần chương trình lùi dao (theo hai trục) và trở về vị trí khởi đầu:

```
...
N61 G00 X3.3 Z0.3             (START POINT XZ)
N62 X2.972                    (THREAD DIA START)
N63 G32 Z-1.75 F0.0833        (THREAD TO END)
N64 U0.2 W-0.1                (GRADUAL PULLOUT)
N65 G00 X3.3                  (RAPID OUT)
N66 Z0.3                      (RETURN TO Z-START)
...
```

TỐC ĐỘ CẮT REN VÀ TỐC ĐỘ TRỤC CHÍNH

Trong gia công ren, sự lựa chọn dao, tốc độ trục chính, và tốc độ cắt thường bị hạn chế. Cả dao và tốc độ cắt đều được xác định theo ren hoàn chỉnh, chuyên biệt trên bản vẽ kỹ thuật. Dao tiện ren là một trong các dụng cụ cắt *yếu nhất* được dùng trên máy tiện CNC – nhưng các ứng dụng lại yêu cầu tốc độ cắt *nặng* trong lập trình máy tiện CNC đối với dao bất kỳ. Các yếu tố khác có thể ảnh hưởng đến ren hoàn tất gồm tốc độ trục chính, chiều sâu từng đường cắt, sự chuẩn bị lưỡi cắt, gá lắp dao... Nói chung, chỉ cần thay đổi một yếu tố cũng có thể giải quyết vấn đề. Hình 37.6 so sánh tốc độ cắt khi tiện mặt trụ và khi tiện ren.



Hình 37.6. So sánh tốc độ cắt khi tiện mặt trụ và khi tiện ren.

Chọn tốc độ cắt ren

Chọn tốc độ cắt khi tiện hoặc doa dựa trên vật liệu được gia công, bán kính mũi dao, độ bóng bề mặt... Theo ý nghĩa đó, tốc độ cắt "thích hợp" khi tiện hoặc doa bao quát một khoảng rộng. Trong tiện ren, tính linh hoạt này bị giới hạn. Tốc độ cắt ren luôn luôn được xác định bằng *bước tiến* (dẫn hướng) ren – *thay vì* bước ren. Trên bản vẽ dùng đơn vị Anh, ren được cho theo số ren/inch, viết tắt là TPI (theads per inch) và đường kính danh định. Ví dụ, ren được ghi trên bản vẽ là 3.75–8, có nghĩa là *8 ren/inch*, và đường kính danh định (ví dụ, đường kính chính) là Ø3.750. Mọi ren hệ mét một đầu mối đều có bước ren tiêu chuẩn hóa, tùy theo đường kính ren. Ví dụ, ren M24×3 là ren hệ mét một đầu mối với bước 3 mm trên đường kính 24 mm. M24×1 là ren một đầu mối có bước ren 1 mm.

Bất kể đơn vị kích thước, điều quan trọng nhất khi chọn tốc độ cắt là *bước tiến* và *số đầu mối* của ren.

Phần này sẽ tóm tắt quan hệ cơ bản giữa *bước tiến* (dẫn hướng) và *bước ren*. Trong

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

xưởng cơ khí, các thuật ngữ *bước tiến* và *bước ren* đôi khi bị sử dụng sai. Lý do là đối với ren một đầu mối, giá trị bước tiến *bằng* giá trị bước ren. Do xưởng cơ khí thường chỉ gia công ren một đầu mối, sự khác biệt giữa hai khái niệm này ít được chú ý. Ngoài ra, hầu như mọi mũi tarô đều chỉ có một đầu mối. Nếu không phân biệt rõ giữa các khái niệm đó, sẽ lập trình không chuẩn. Mỗi khái niệm đều có ý nghĩa riêng, cần sử dụng chúng một cách chính xác.

$$\text{BƯỚC TIẾN} = \frac{\text{Số đầu mối ren}}{\text{TPI}} = F$$

Tốc độ cắt ren được lập trình luôn luôn là bước tiến, không phải bước ren.

$$\text{BƯỚC REN} = \frac{1}{\text{TPI}}$$

Từ hai công thức này có thể dễ dàng nhận thấy nếu số đầu mối ren là *một*, bước tiến và bước ren có cùng giá trị.

Khi tính tốc độ cắt ren, cần áp dụng công thức:

$$F = L = P \times n$$

Trong đó:

F = Tốc độ cắt (in/rev hoặc mm/rev)
L = Bước tiến của ren(inch hoặc mm)
P = Bước ren (inch hoặc mm)
n = Số đầu mối (số nguyên dương)

Ví dụ, ren một đầu mối có bước ren 3 mm, tốc độ cắt sẽ là:

$3 \times 1 = F3.0$

Đối với các chương trình gia công ren sử dụng đơn vị Anh, công thức nêu trên vẫn có giá trị, do

$$P = \frac{1}{\text{TPI}}$$

Trong đó:

P = Bước ren
TPI = Số ren/inch

Ví dụ, ren một đầu mối, 8 TPI, tốc độ cắt sẽ là:

$1/8 \times 1 = .125 \times 1 = F0.125$

Các ren nhiều đầu mối có các đặc tính riêng, nhưng lựa chọn tốc độ cắt vẫn luôn luôn là *bước tiến* thay vì bước ren.

Chọn tốc độ trục chính

Tốc độ trục chính để tiện ren luôn luôn được lập trình trực tiếp với r/min, không dùng tốc độ bề mặt không đổi (CSS). Điều đó có nghĩa là phải dùng lệnh chuẩn bị G97 với địa chỉ S, chuyên biệt số vòng quay trong một phút. Ví dụ, G97S500M03, tốc độ trục chính là 500 r/min.Thực tế là sự tiện ren một lưỡi cắt thực hiện qua vài đường kính giữa đường cắt thứ nhất và chân ren, do đó lựa chọn G96 có lẽ là logic. Nhưng điều đó không đúng. Thứ nhất, ngay cả đối với ren thô hơi sâu, hiệu giữa đường kính thứ nhất và đường kính cuối cũng không đáng kể. Thứ hai – và lý do này còn quan trọng hơn – chương trình gia công ren yêu cầu tốc độ trục chính và tốc độ cắt phải đồng bộ với nhau một cách hoàn hảo khi bắt đầu từng đường cắt. Sự đồng bộ đó chỉ có thể đạt được một cách chính xác với r/min không đổi thay vì tốc độ bề mặt không đổi (CSS).

Đối với đa số các ren, sự lựa chọn r/min chỉ đòi hỏi xem xét các điều kiện gia công tổng quát, tương tự các nguyên công tiện khác. Đồng thời, sự lựa chọn tốc độ trục chính phải xét đến tốc độ cắt. Do sử dụng các chế độ cắt nặng khi tiện ren, sẽ có khả năng một số ren không thể cắt với *tốc độ trục chính khả dụng bất kỳ*. Từng máy tiện CNC đều có giá trị tốc độ cắt có thể lập trình, được chuyên biệt theo in/min hoặc mm/min, cho đến giá trị max trên từng trục.

Giả sử tốc độ cắt max có thể lập trình trên trục X là 250 in/min (6350 mm/min); giá trị max trên trục Z có thể là 450 in/min (11430 mm/min). Bạn hãy nhớ lại quan hệ giữa tốc độ trục chính và tốc độ cắt theo vòng quay. Kết quả của quan hệ này là *tốc độ cắt biểu thị theo thời gian*, không theo vòng quay. Tốc độ cắt theo thời gian luôn luôn là kết quả của tốc độ trục chính r/min nhân với tốc độ cắt in/rev hoặc mm/rev.

➲ **Ví dụ đơn vị Anh:**

700 r/min x .125 in/rev = 87.500 in/min

➲ **Ví dụ đơn vị hệ mét**

700 r/min x 3mm/rev = 2100 mm/min

Trong lập trình tiện CNC nói chung, không chỉ tiện ren, cần bảo đảm tốc độ cắt theo vòng quay kết hợp với tốc độ trục chính sẽ *nhỏ hơn hoặc bằng* tốc độ cắt *khả dụng* max theo thời gian đối với trục có định mức thấp hơn, thường là trục X.

Dựa trên nguyên tắc đơn giản này, tốc độ trục chính max đối với bước tiến cho trước có thể chọn theo công thức dưới đây:

$$R_{max} = \frac{Ft_{max}}{L}$$

Trong đó:

R_{max} = r/min cho phép cực đại

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ft_{max} = tốc độ cắt/thời gian cực đại (trục X)
L = Bước tiến của ren

Nhiều tài liệu hay có tại : http://www.me-cad.com/

➲ **Ví dụ đơn vị Anh**

Nếu bước tiến ren L là .125 và tốc độ cắt max đối với trục X là Ft_{max} = 250 in/min, tốc độ cắt ren max R_{max} sẽ là:

R_{max} = 250 / .125 = 2000 r/min

➲ **Ví dụ đơn vị hệ mét**

Nếu bước tiến ren L là 2.5 mm và tốc độ cắt max đối với trục X là Ft_{max} = 6350 mm/min, tốc độ cắt ren max R_{max} sẽ là:

R_{max} = 6350 / 2.5 = 2540 r/min

Tốc độ r/min cho phép cực đại chỉ phản ánh khả năng của máy CNC. Tốc độ cắt được sử dụng thực tế trong chương trình phải tính toán các điều kiện gia công và gá lắp, tương tự các nguyên công cắt gọt khác. Trong thực tiễn, đa số các tốc độ trục chính (r/min) lập trình đều thấp hơn khả năng cực đại của máy công cụ CNC.

Tốc độ cắt ren cực đại

Sựa lựa chọn tốc độ cắt tổng quát đã được trình bày trong Chương 12. Sau khi nghiên cứu phần lựa chọn tốc độ r/min cực đại (tốc độ trục chính), bạn sẽ thấy các hạn chế tương tự cũng áp dụng cho sự xác định *tốc độ cắt ren cực đại* đối với tốc độ trục chính cho trước (lập trình theo r/m). Các giới hạn của máy công cụ CNC là rất quan trọng, cần đặc biệt chú ý khi viết chương trình gia công ren.

Tốc độ cắt ren cực đại có thể lập trình tương ứng tốc độ trục chính cho trước (r/min) được tính từ công thức:

$$Fr_{max} = \frac{Ft_{max}}{S}$$

Trong đó:
Fr_{max} = Tốc độ cắt cực đại tương ứng tốc độ trục chính cho trước
Ft_{max} = Tốc độ cắt cực đại theo thời gian (trục X)
S = Tốc độ trục chính được lập trình

➲ **Ví dụ đơn vị Anh**

Nếu tốc độ cắt max theo trục X là 500 in/min và tốc độ trục chính được chọn là 2000 r/min, tốc độ cắt max có thể lập trình sẽ là:

250 / 2000 = .125 in/rev

Do đó, bước tiến ren max có thể cắt với tốc độ 2000 r/min là .125 inch, cho phép cắt 8 ren/inch hoặc mịn hơn.

Sự thay đổi tốc độ trục chính (tốc độ cắt không đổi) sẽ cho phép lập trình các ren thô hơn trên máy tiện CNC. Ví dụ, nếu chỉ chọn 1500 r/min, thay vì 2000, bước tiến ren max sẽ tăng đến .1670 inch, tương ứng 6 ren/inch.

➲ **Ví dụ đơn vị hệ mét**

Trong ví dụ tương tự, sử dụng các đơn vị hệ mét, tốc độ cắt cực đại trên trục X là 6350 mm/min và tốc độ trục chính lập trình được chọn là S = 1600 r/min. Trong trường hợp này, tốc độ cắt ren cực đại có thể lập trình sẽ là:

6350 / 1600 = 3.969 mm/rev

Điều đó có nghĩa là bước tiến ren cực đại tương ứng tốc độ 1600 r/min sẽ nhỏ hơn 4 mm.

Các giá trị tính toán chỉ nêu ra các khả năng thực của bộ điều khiển và máy, không bảo đảm xác lập an toàn cho chi tiết và chưa hẳn là tốc độ gia công thích hợp.

Sai số bước tiến

Nói chung, tốc độ cắt ren yêu cầu địa chỉ F, với độ chính xác đến bốn chữ số thập phân đối với ren hệ Anh (định dạng F2.4), và ba chữ số thập phân đối với ren hệ mét (định dạng F3.3). Đa số các ren đều ngắn và độ chính xác này là đủ. Không có vấn đề đối với ren hệ mét, bất kể chiều dài ren, do ren được xác định theo bước tiến trên bản vẽ. Đối với ren lập trình theo đơn vị Anh, bước tiến ren phải được tính từ số ren/inch (TPI) trên bản vẽ. Đối với nhiều ren hệ Anh, bước tiến được tính toán một cách chính xác trong phạm vi bốn chữ số thập phân khả dụng cho địa chỉ F. Ren 10TPI đòi hỏi tốc độ lập trình F0.1, ren 16 TPI yêu cầu tốc độ lập trình F0.0625, ... Đây là các ren theo tỷ số 1/TPI trong phạm vi độ chính xác với bốn chữ số thập phân, với TPI là 8, 10, 16, 20, 40, ...

Không phải mọi ren đều thuộc nhóm thuận tiện này. Đối với nhiều ren khác, giá trị tính toán cần được làm tròn.

Ví dụ, xét ren 14 TPI. Tốc độ cắt ren chính xác sẽ là 1/14 = .071428571 inch/rev, giá trị làm tròn được dùng trong chương trình sẽ là F0.0714. Nếu chiều dài ren ngắn sẽ hoàn toàn không có sai số rõ rệt, ren trong khoảng dung sai cho phép. *Sai số tích lũy*, được gọi là *sai số bước tiến ren*, có thể dẫn đến phế phẩm do ren không chính xác. Bằng cách dùng giá trị làm tròn .0714, tổn thất sẽ là .000028571 inch đối với từng vòng ren. Sai số bước tiến trên một inch (hoặc dài hơn) có thể tính theo công thức:

$$L_e = (F_a - F_p) \times TPI$$

Trong đó:
L_e = Sai số bước tiến max/inch

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

F_a = Tốc độ cắt tính toán
F_p = Tốc độ cắt được làm tròn
TPI = Số ren/inch

Trên một inch, sai số trong ví dụ này là 0.0004 inch, và 50 inch sẽ là 0.0200 inch, Ví dụ thứ hai, giả sử làm tròn tăng. Một cách lý tưởng, ren vơi 11.5 ren/inch cần lập trình với tốc độ cắt .086956522. Nếu giá trị này được làm tròn đến .0870, sai số tích lũy sẽ là .0005 inch/inch và sai số qua 50 inch sẽ là .0250 inch. Dù đối với tốc độ cắt ren, máy CNC không cho phép sáu chữ số thập phân, sự làm tròn giá trị tính toán cũng rất quan trọng.

Bạn hãy so sánh các giá trị làm tròn dưới đây và sai số tương ứng (11.5 TPI với chiều dài 50 inch):

.0869 ... sai số .0325
.0870 ... sai số .0250
.0871 ... sai số .0825

Sự khác biệt giữa các giá trị làm tròn chỉ là 0.0001 inch.

Các nhà chế tạo bộ điều khiển Fanuc đã nhận thức được vấn đề này và đưa ra địa chỉ E đối với tốc độ cắt ren trên các bộ điều khiển CNC cũ của họ. Ưu điểm của địa chỉ E đối với tiện ren là cho phép lập trình với *sáu* chữ số thập phân thay vì *bốn* đối với ren hệ Anh với sự làm tròn số hợp lý, sai số tích lũy sẽ rất nhỏ.

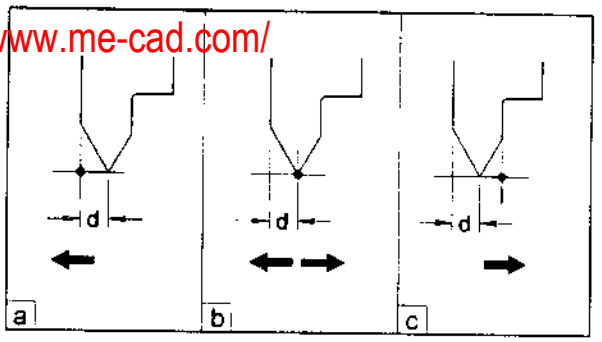
Sử dụng ví dụ 14 TPI trên chiều dài 50 inch, sai số đối với toàn bộ chiều dài này chỉ là .0003 inch, nếu thay F0.0714 bằng E0.071429. Ví dụ thứ hai, sử dụng ren 11.5 TPI, cần lập trình với tốc độ cắt E0.086957. Sai số tích lũy trên chiều dài 50 inch chỉ là .000275 inch, hoàn toàn có thể bỏ qua.

Các hệ thống CNC hiện nay cho phép sử dụng độ chính xác đến sáu chữ số cho địa chỉ F.

Sai số bước tiến luôn luôn là vấn đề tiềm ẩn khi lập trình các bước tiến ren dài. Tùy theo ứng dụng ren trong xưởng cơ khí, sai số tích lũy của bước tiến ren có thể trở thành vấn đề cần giải quyết chính xác.

ĐIỂM QUY CHIẾU DAO

Sự xác lập dao chuẩn là vấn đề cơ bản trong môi trường gia công. Tuy sự xác lập chuẩn là rất quan trọng đối với mọi dụng cụ cắt, nhưng duy trì sự xác lập đó đối với dao tiện ren, trong và ngoài, còn quan trọng hơn. Lưỡi cắt cần được định hướng đúng, gá lắp chắc chắn trong ổ dao, và phải phù hợp với loại ren cần gia công. Điểm quy chiếu (chuẩn dao), được dùng để gá lắp dao, cũng rất quan trọng.



Hình 37.7. Các điểm quy chiếu để gá lắp dao tiện ren.

Điểm quy chiếu của dao tiện ren đòi hỏi xem xét cẩn thận hơn so với các dao tiện khác. Trên Hình 37.7, có ba khả năng, theo mức độ thường xuyên khi lập trình. Phiên bản (C) rất ít sử dụng, ngoại trừ một số trường hợp tiện ren trái. Đối với hầu hết các ren trái, phiên bản (a) hoặc (b) có thể thuận tiện hơn.

Xác lập dao theo Hình 37.7a thích hợp nhất cho ứng dụng chung và cho các ren kết thúc ở vai hoặc bậc. Cấu hình trên Hình 37.7b thích hợp cho các ren kết thúc trên đường kính hở. Hình 37.7c là sự xác lập có thể dùng cho ren trái.

Lựa chọn điểm quy chiếu dao (G50 hoặc xác lập bù hình học) theo Hình 37.7a là rất thích hợp khi muốn *tiêu chuẩn hóa* gá lắp dao đối với kiểu ren *bất kỳ*. Đây là xác lập thuận tiện nhất, bất kể sự kết thúc ren, đồng thời an toàn nhất. Trong một số trường hợp, cần có khoảng hở đối với sự khác biệt giữa lưỡi cắt được lập trình và lưỡi cắt thực. Bạn nên sử dụng giá trị một nửa chiều rộng lưỡi cắt ren (nếu có thể áp dụng).

LẬP TRÌNH TIỆN REN THEO BLOCK

Phương pháp cũ nhất trong tiện ren một lưỡi cắt là tính toán từng chuyển động liên quan với tiện ren và lập trình theo block chương trình. Phương pháp này được gọi là lập trình tiện ren theo block.

Mỗi bước trong bốn bước cơ bản chiếm một block chương trình, kết quả là bốn block cho từng đường cắt ren. Nếu dùng sự lùi dao từ từ (theo hai trục) ra khỏi ren, sẽ có năm block cho từng đường cắt ren. Khi cắt ren thô, ren trên vật liệu cứng, hoặc ren nhiều đầu mối, chương trình tiện ren theo block sẽ rất dài. Chiều dài của chương trình, khó khăn khi chỉnh sửa, khả năng xảy ra sai sót, dung lượng nhớ nhỏ trong bộ điều khiển, là các nhược điểm của phương pháp này.

Về ưu điểm, nhà lập trình *hoàn toàn kiểm soát sự lập trình gia công ren*. Sự kiểm soát đó,

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

cho phép áp dụng một số kỹ thuật gia công ren đặc biệt, ví dụ cắt hình dạng ren với dao nhỏ hơn ren, hoặc gia công các ren ăn khớp lớn với dao cắt rãnh tròn.

Phương pháp lập trình sử dụng kỹ thuật block đối với ren bước tiến không đổi khả dụng trên mọi máy tiện CNC.

G32	Lệnh cắt gọt ren.

Lệnh chuẩn bị đối với kiểu ren này là G32. Có thể dùng lệnh G33 trên một số bộ điều khiển, nhưng G32 là mã tiêu chuẩn đối với Fanuc và các bộ điều khiển tương thích.

Ví dụ dưới đây sẽ sử dụng ren ngoài 3.0 – 12 TPI. Sự cắt ren được phân chia theo bảy đường cắt, chiều sâu tosàn phần là .0511 inch:

Chiều sâu đường cắt #1 = .0140	Chiều sâu tổng	.0140
	Đường kính	2.9720
Chiều sâu đường cắt #2 = .0100	Chiều sâu tổng	.0240
	Đường kính	2.9520
Chiều sâu đường cắt #3 = .0080	Chiều sâu tổng	.0320
	Đường kính	2.9360
Chiều sâu đường cắt #4 = .0065	Chiều sâu tổng	.0385
	Đường kính	2.9230
Chiều sâu đường cắt #5 = .0050	Chiều sâu tổng	.0435
	Đường kính	2.9130
Chiều sâu đường cắt #6 = .0045	Chiều sâu tổng	.0480
	Đường kính	2.9040
Chiều sâu đường cắt #7 = .0031	Chiều sâu tổng	.0511
	Đường kính	2.8978

Cần tính toán cẩn thận tất cả các đường kính sao cho không có sai số. Sai số dù nhỏ cũng có thể dẫn đến phế phẩm.

Nguyên công tiện ren trong chương trình O3701 sẽ sử dụng dao và số bù dao 5 (T0505), với tốc độ trục chính 450 r/min (G97S450):

```
O3701
...
(N45 G50 X12.0 Z4.5)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X3.2 Z0.25 T0505 M08
```

Lúc này dao đạt đến điểm khởi đầu tiện ren. Giai đoạn kế tiếp là thực thi tất cả bốn bước, mỗi bước một block, cho đường cắt *thứ nhất*:

```
N49 X2.972                          (PASS 1)
N50 G32 Z-1.6 F0.0833          (or F/E0.083333)
N51 G00 X3.2
N52 Z0.25
```

Có thể lập trình tiếp 6 đường cắt còn lại, chỉ cần thay đổi giá trị đường kính. Bạn hãy lưu ý, tốc độ cắt ren *không* lặp lại – do đây là lệnh chế độ từ block N50 trở đi

```
N53 X2.952                          (PASS 2)
N54 G32 Z-1.6
N55 G00 X3.2
N56 Z0.25
N57 X2.9360                         (PASS 3)
N58 G32 Z-1.6
N59 G00 X3.2
N60 Z0.25
N61 X2.9230                         (PASS 4)
N62 G32 Z-1.6
N63 G00 X3.2
N64 Z0.25
N65 X2.9130                         (PASS 5)
N66 G32 Z-1.6
N67 G00 X3.2
N68 Z0.25
N69 X2.9040                         (PASS 6)
N70 G32 Z-1.6
N71 G00 X3.2
N72 Z0.25
N73 X2.8978                         (PASS 7)
N74 G32 Z-1.6
N75 G00 X3.2
N76 Z0.25
```

Block N76 kết thúc sự gia công ren, từ đây có thể đóng chương trình nếu không dùng thêm các dao khác.

```
N77 X12.0 Z4.5 T0500 M09
N78 M30
%
```

Nhược điểm trong chương trình này là lặp lại quá nhiều. Bạn hãy quan sát ba block tiếp sau từng đường cắt mới. Chúng luôn luôn như nhau. Đối với ren cần nhiều đường cắt, sự lặp lại này sẽ rất lớn. Phương pháp lập trình block có một ưu điểm, hoàn toàn do nhà lập trình kiểm soát, có thể thực hiện các điều chỉnh về ren và chiều sâu từng đường cắt. Có thể bổ sung phương pháp ăn dao phi tiêu chuẩn và rút dao từ từ (theo hai trục) ra khỏi ren. Nhưng sẽ rất bất tiện khi cần chỉnh sửa chương trình.

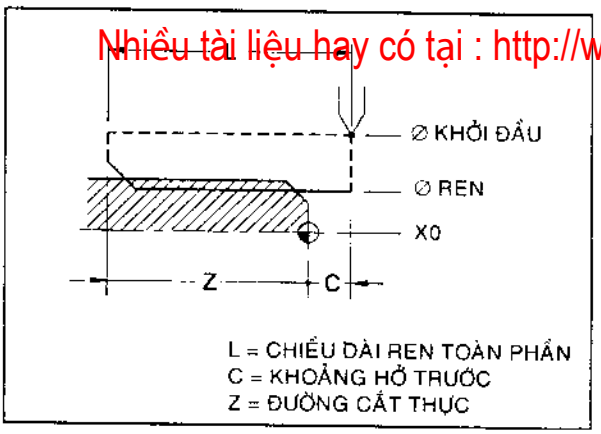
CHU KỲ TIỆN REN CƠ BẢN – G92

Hệ thống điều khiển có thể thực hiện nhiều tính toán và lưu kết quả trong bộ nhớ để sử dụng khi cần thiết. Tính năng này đặc biệt hữu dụng trong gia công ren. Do có thể tránh lặp lại chuyển động dao theo block và chương trình sẽ được rút ngắn rõ rệt.

Chương trình ví dụ O3701 minh họa lệnh G32 được sử dụng lại nhưng sẽ áp dụng chu kỳ gia công ren đơn giản. Đây là chu kỳ gia công ren G92 trên bộ điều khiển Fanuc. Bạn hãy chú ý, G92 dùng để tiện ren *hoàn toàn khác* với lệnh cùng tên, G92 cũ đã lạc hậu là lệnh xác lập *đăng ký hệ thống*. Nếu bộ điều khiển trên máy tiện sử dụng G92 cho chu kỳ tiện ren đơn giản, bạn hãy dùng G50 làm lệnh đăng ký hệ thống. Lệnh này chỉ áp dụng trên các bộ điều khiển kiểu cũ, do các bộ điều khiển hiện đại sử dụng các chế độ bù hình học tiên tiến.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 37.8. *G92 – Chu kỳ tiện ren đơn giản*

Hình 37.8 minh họa sơ đồ chu kỳ tiện ren G92.

Để so sánh với phương pháp lập trình G32, cùng loại ren 3.0 – 12 sẽ được lập trình, ren này có 12 ren/inch trên đường kính ngoài 3.0 inch. Chương trình mới có cùng tác dụng như chương trình cũ O3701 nhưng có cấu trúc khác.

Sử dụng chu kỳ G92, dưới đây sẽ liệt kê các đường kính đã tính cho từng đường cắt ren, theo tứ thự xuất hiện trong chương trình (vào giai đoạn này vẫn chưa thay đổi):

Chiều sâu đường cắt # 1 = Ø 2.9720
Chiều sâu đường cắt # 2 = Ø 2.9520
Chiều sâu đường cắt # 3 = Ø 2.9360
Chiều sâu đường cắt # 4 = Ø 2.9230
Chiều sâu đường cắt # 5 = Ø 2.9130
Chiều sâu đường cắt # 6 = Ø 2.9040
Chiều sâu đường cắt # 7 = Ø 2.8978

Cũng như trước, dao tiện ren được gán là T0505 và tốc độ trục chính là 450 r/min:

```
O3702
...
(N45 G50 X12.0 Z4.5)
N45 T0500 M42
N47 G97 S450 M03
N48 G00 X3.2 Z0.25 T0505 M08  (START POSITION)
```

Bốn block đầu là đồng nhất với phương pháp lập trình tiện ren theo block. Trong bước kế tiếp, dao sẽ được định vị ở đường kính đường cắt thứ nhất, cắt ren, lùi ra khỏi ren, và trở về vị trí khởi đầu. Ba block này lặp lại đối với từng đường cắt. Ưu điểm chính của chu kỳ tiện ren G92 là loại bỏ các dữ liệu lặp lại đó và làm cho chương trình dễ chỉnh sửa hơn.

Định dạng của chu kỳ tiện ren trực tiếp G92 là:

```
G92 X.. Z.. F..
```

Trong đó:
X = đường kính hiện hành của đường cắt ren
Z = vị trí kết thúc ren
F = tốc độ cắt ren in/rev

Đường cắt ren thứ nhất sẽ được lập trình trong block này – N49. Chú ý, các giá trị nhập trục X, trục Z, và tốc độ cắt:

```
N49 G92 X2.972 Z-1.6 F0.0833          (PASS 1)
```

Hệ điều khiển sẽ lấy giá trị X cuối cùng và giá trị Z cuối cùng *trước khi gọi chu kỳ* làm vị trí khởi đầu của ren. Vị trí này là điểm khởi động của chu kỳ. Trong ví dụ, vị trí khởi đầu là X3.2 Z0.25 (block N48). Sáu đường cắt ren còn lại được lập trình bằng cách thay đổi giá trị đường kính, không cần lặp lại giá trị Z *hoặc tốc độ cắt*.

```
N50 X2.9520                           (PASS 2)
N51 X2.9360                           (PASS 3)
N52 X2.9230                           (PASS 4)
N53 X2.9130                           (PASS 5)
N54 X2.9040                           (PASS 6)
N55 X2.8978                           (PASS 7)
```

Block N55 sẽ hoàn tất bằng chuyển động tự động trả về vị trí khởi đầu của ren. Từ vị trí đó, chương trình kết thúc hoàn toàn như với phương pháp sử dụng G32.

```
N56 G00 X12.0 Z4.5 T0500 M09
N57 M30
%
```

Một nhầm lẫn hoặc sai sót lập trình thường xảy ra với chu kỳ này là bỏ qua lệnh G00 trong block N56. Chu kỳ G92 chỉ có thể xóa bằng *lệnh chuyển động khác*, trong trường hợp này là chuyển động nhanh G00. Nếu bỏ sót G00 trong chương trình, hệ điều khiển sẽ dự đoán còn có các đường cắt ren khác, trong khi cắt ren đã hoàn tất trong block trước đó (N55).

Chu kỳ tiện ren đơn giản G92, không có các phương pháp ăn dao đặc biệt, chỉ đẩy dao thẳng vào bề mặt gia công. Cuối Chương này sẽ giải thích phương pháp ăn dao thẳng là *không* thích hợp cho hầu hết các nguyên công tiện ren.

Có thể lập trình sự lùi dao từ từ (theo hai trục) với G92 bằng cách dùng hàm M24 trước khi gọi chu kỳ G92. Nếu hệ điều khiển hỗ trợ tính năng này (hầu hết các bộ điều khiển đều có), bạn hãy sử dụng chu kỳ tiện ren G76.

CHU KỲ LẶP NHIỀU LẦN – G76

Trong Chương 34 đã trình bày nhiều chu kỳ trên máy tiện CNC, chủ yếu dùng cho tiện và doa. Trong phần này, sẽ trình bày một chu kỳ lặp dùng cho các ứng dụng tiện ren.

Trong các giai đoạn đầu của sự phát triển CNC, chu kỳ gia công ren đơn giản G92 là kết quả trực tiếp của công nghệ máy tính hóa thời

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

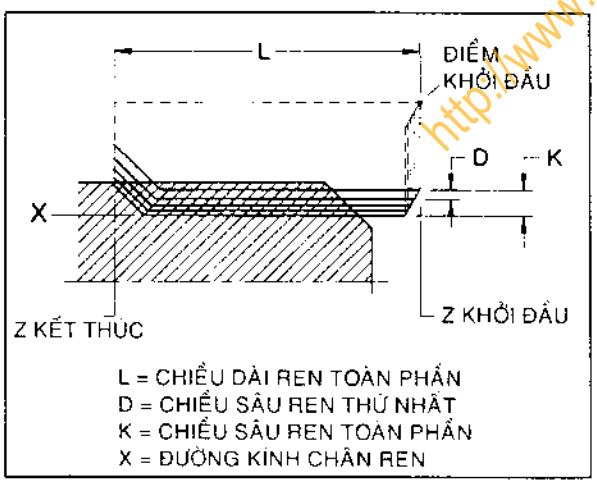
kỳ đó. Công nghệ máy tính đã phát triển rất nhanh, cung cấp nhiều tính năng mới cho nhà lập trình. Một trong các bổ sung chính là chu kỳ gia công ren lặp lại nhiều lần G76. Đây được coi là chu kỳ phức tạp, không phải do khó sử dụng (mà ngược lại, rất dễ sử dụng), chủ yếu do có một số tính năng rất mạnh.

Để nhận thức rõ tác dụng của chu kỳ gia công ren G76, bạn hãy so sánh với phương pháp dùng G32, thậm chí với chu kỳ G92. Trong khi chương trình sử dụng G32 đòi hỏi bốn hoặc năm block chương trình cho từng đường cắt ren, chu kỳ G76 *thực hiện sự cắt ren đơn bất kỳ* chỉ trong *một* block mã chương trình (một số bộ điều khiển yêu cầu hai block). Với chu kỳ G76, *số lượng bất kỳ* các đường cắt ren vẫn chỉ chiếm một phần rất nhỏ trong chương trình, cho phép chỉnh sửa trên máy (nếu cần thiết) rất nhanh và dễ dàng.

Hiện có hai định dạng lập trình, tùy theo bộ điều khiển, tương tự sự lập trình các chu kỳ tiện khác.

Định dạng chu kỳ G72 – 10T/11T/15T

Chu kỳ tiện ren yêu cầu các dữ liệu nhập ban đầu – thông tin cung cấp cho bộ điều khiển xác định ren theo các đại lượng gia công. Hình 37.9 minh họa G76 đối với các bộ điều khiển Fanuc 10/11/15T.



Hình 37.9. *G76 - Chu kỳ tiện ren lặp nhiều lần (10T/11T/15T)*

Các tham số này tạo thành cấu trúc của chu kỳ một – block (ren ngoài hoặc ren trong):

```
G76 X.. Z.. I.. K.. D.. F.. A.. P..
```

Trong đó:

X = Đường kính lần cắt ren cuối cùng
Z = Vị trí biểu thị sự kết thúc ren
I = Giá trị côn trên toàn bộ chiều dài
K = Chiều sâu đơn của ren – dương
D = Chiều sâu đường cắt thứ nhất – dương
A = Góc mảnh chắp – dương
P = Phương pháp ăn vào – dương

Bạn hãy quan sát các khác biệt trong cấu trúc định dạng của chu kỳ lặp nhiều lần G76 với chu kỳ cơ bản G92. Chu kỳ G76 xuất hiện có vẻ đơn giản nhưng bên trong rất phức tạp, hệ điều khiển phải thực hiện nhiều tính toán và kiểm tra. Đây là một lý do để sử dụng máy tính, cho phép chúng thực hiện công việc khó khăn. Các tính toán này cần dữ liệu (thông tin lặp lại), ở dạng tham số nhập, thiết lập các đặc tính kỹ thuật của ren. Tuy cần nhiều dữ liệu nhập, nhưng G76 là chu kỳ rất dễ sử dụng trong lập trình CNC.

Định dạng chu kỳ G76 – 0T/16T/18T

Trên các bộ điều khiển Fanuc thông dụng 0T, 16T, 18T, chu kỳ G76 hơi khác so với các bộ điều khiển 10/11/15T. Mục đích và chức năng của G76 là như nhau, sự khác biệt chỉ là cách thức thiết lập cấu trúc cho dữ liệu nhập. Các bộ điều khiển Fanuc 10/11/15T sử dụng nhập chu kỳ một dòng, còn 0/16/18T yêu cầu nhập hai dòng.

```
G76 P.. Q.. R..
G76 X.. Z.. R.. P.. Q.. F..
```

Trong đó:

Block thứ nhất

P = ...mục nhập dữ liệu sáu chữ số theo ba cặp:
Các chữ số 1 và 2 - số đường cắt tinh (01-99)
Các chữ số 3 và 4 – số bước tiến đối với rút dao từ từ (0.0 – 9.9 nhân với bước tiến), không dùng dấu thập phân (00-99)
Các chữ số 5 và 6 – góc ren (chỉ gồm 00, 29, 30, 55, 60 và 80 độ)
Q= Chiều sâu cắt tối thiểu (giá trị bán kính dương, không có dấu thập phân)
R = Giá trị xác định của lượng dư để gia công tinh (cho phép dấu thập phân)

Block thứ hai

X = (a) Đường kính ren cuối cùng (đường kính tuyệt đối)
... hoặc ...
(b) Khoảng cách từ điểm khởi đầu đến đường kính ren cuối cùng (chế độ số gia)
Z = Kết thúc ren theo trục Z (có thể là khoảng cách số gia W)
R = Hiệu số bán kính giữa các vị trí đầu và cuối của ren trong đường cắt cuối cùng (R0, dùng cho ren thẳng, có thể bỏ qua)
P = Chiều cao ren (giá trị bán kính dương, không có dấu thập phân)
Q = Chiều sâu đường cắt ren thứ nhất (giá trị dương – không có dấu thập phân)
F = Tốc độ cắt ren (bằng bước tiến ren)

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Bạn cần nhận thấy rằng thứ tự nhập các địa chỉ chu kỳ tiện khác đã trình bày trong Chương 34. Bạn không nên nhầm lẫn giữa các địa chỉ P/Q/R trong block thứ nhất với P/Q/R trong block thứ hai, chúng có ý nghĩa riêng – chỉ trong từng block tương ứng.

➲ **Ví dụ – Đơn vị Anh**
(Ren ngoài 1 – 11/6 với 20 TPI)

```
N10 G76 P011060 Q005 R0.003
N11 G76 X1.6261 Z-1.5 P0307 Q0100 F0.05
```

➲ **Ví dụ – Đơn vị hệ mét**
(Ren trong M76 x 1.5)

```
N20 G76 P011060 Q050 R0.05
N21 G76 X76.0 Z-30.0 P812 Q250 F1.5
```

Ví dụ lập trình

Ví dụ về ren với 12TPI trên đường kính ngoài 3.000 inch, có thể dễ dàng sử dụng phương pháp lập trình G76. Ở đây sẽ trình bày các ví dụ cho cả hai loại bộ điều khiển, chỉ sử dụng số lượng *tối thiểu* các block chương trình (trong đó chỉ trình bày dao tiện ren):

```
O3703 (G76 METHOD - ONE BLOCK METHOD)
...
(N45G50 X12.0 Z4.5)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X3.2 Z0.25 T0505 M08
N49 G76 X2.8978 Z-1.6 I0 K0.0511 D0140 A60
    P4 F0.0833                (or F/E0.083333)
N50 G00 X12.0 Z4.5 T0500 M09
N51 M30
%
```

Cần làm rõ một số điểm liên quan đến chương trình.Thực tế chương trình chỉ cần sáu hoặc bảy block cơ bản. Có thể thực hiện sự thay đổi bất kỳ trong chương trình bằng cách chỉnh sửa tham số trong block N49, nơi gọi chu kỳ gia công ren. Ví dụ, để thay đổi chiều sâu đường cắt ren thứ nhất từ giá trị hiện hành .0140 sang .0160, toàn bộ điều cần làm là thay D0140 bằng D0160.

Sự so sánh chu kỳ G76 với G92 là không hợp lý, do từng chu kỳ này là sản phẩm của gia đoạn công nghệ tương ứng. Chúng cùng hiện diện trong một bộ điều khiển hiện đại, chủ yếu để *tương thích ngược* với các chương trình cũ. Hai chương trình này là minh họa cho sự khác biệt giữa các kỹ thuật lập trình.

Ví dụ, trong ứng dụng chu kỳ tiện ren G92, nhập *từng* đường kính đường cắt ren là rất quan trọng, trong chu kỳ G76, chỉ giá trị nhập đường kính đường cắt *cuối cùng* là quan trọng.

Hệ CNC thực hiện mọi tính toán cần thiết. Thông tin cung cấp được chứa trong chương trình. Thứ nhất, bộ điều khiển ghi lại vị trí khởi đầu của dao như này, tương tự chu kỳ G92. Trong ví dụ này (block N48), là vị trí X3.2Z0.25. Bước kế tiếp, bộ điều khiển đánh giá tất các các tham số G76 (dữ liệu lập trình trong block N49). Giá trị X là đường kính của đường cắt ren cuối cùng, giá trị K là chiều sâu ren đơn. Điều này cung cấp đủ thông tin để bộ điều khiển "biết" đường kính chi tiết *lý thuyết* đã gia công sơ bộ (không thể biết đường kính *thực* đã gia công sơ bộ). Quan hệ này là quan trọng để chọn *chiều* dao tiếp cận nhanh. Nếu đường kính khởi đầu X của ren *lớn hơn* đường kính đường cắt cuối cùng, đây là gia công *ren ngoài*. Nếu đường kính khởi đầu X của ren *nhỏ hơn* đường kính đường cắt cuối cùng, đây sẽ là *ren trong*.

Giá trị Z trong chu kỳ G76 có cùng ý nghĩa như giá trị Z trong lệnh cắt ren G32 hoặc chu kỳ cắt ren G92, biểu thị điểm *kết thúc* ren và kiểm soát chiều dài ren.

Hai tham số đặc thù trong chu kỳ G76 là I và K. Giá trị I luôn luôn bằng zero nếu cắt ren đường kính thẳng (trên mặt trụ). Giá trị khác zero được dùng cho ren côn, I biểu thị hiệu số giữa đường kính đầu và đường kính cuối.

Trong phiên bản hai block, chương trình cũng tương tự định dạng một block, áp dụng các nguyên tắc logic như nhau.

```
O3704 (G76 METHOD - TWO BLOCK METHOD)
...
(N45 G50 X12.0 Z4.5)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X3.2 Z0.25 T0505 M08
N49 G76 P011060 Q005 R0.003
N50 G76 X2.8978 Z-1.6 P0511 Q0140 F0.083333
N51 G00 X12.0 Z4.5 T0500 M09
N52 M30
%
```

Còn vài thông số cần giải thích, nhưng trước hết bạn hãy xem xét cách thức chu kỳ tính toán chiều sâu ren thứ nhất. *Các bộ điều khiển hiện đại sử dụng nhập một – block sẽ được dùng cho các giải thích, trừ khi có các đề nghị khác.*

Tính toán ren thứ nhất

Đối với gia công ren block N32, cũng như chu kỳ gia công ren đơn giản G92, vị trí khởi đầu của ren chỉ được xác định *một cách hợp lý*, áp dụng cho cả hai trục nhằm cung cấp khoảng hở dao thích hợp. Khoảng hở trục Z trong block vị trí khởi đầu chỉ xét bước tiến của ren và tốc độ trục chính. Mục đích là tránh cắt các ren không hoàn chỉnh, do gia tốc máy đối với tốc độ cắt. Khoảng hở đối với trục X là khoảng hở bắt buộc để dao dịch chuyển ra xa ren. Các nguyên tắc này cũng áp dụng cho chu kỳ G76

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

và có thể sử dụng theo cùng cách thức như các phương pháp đã nêu.

Các phương pháp lập trình G32 và G92 có một khác biệt *lớn* so với G76. Trong các ví dụ gia công ren đã nêu, vị trí khởi đầu trên trục X là X3.2. Trong block G32, và trong chu kỳ G92, đường kính gia công ren *thứ nhất luôn luôn* được lập trình (trong các ví dụ đó, giá trị là X2.972). Đây không phải là trường hợp trong chu kỳ G76. Trong chu kỳ này, đường kính *cuối* được lập trình – *không phải đường kính thứ nhất* – nghĩa là đường kính cắt gọt thứ nhất do hệ hệ điều khiển tính toán.

Hệ điều khiển tính đường kính ren *thứ nhất*, nếu được cung cấp đủ thông tin:

- ❑ Đường kính chân ren [Giá trị X]
- ❑ Chiều sâu ren toàn phần [Giá trị K]
- ❑ Chiều sâu ren thứ nhất [Giá trị D]

Dựa trên các giá trị được cung cấp, đường kính thứ nhất T_f của ren được tính như sau:

$$T_f = X + (K \times 2) - (D \times 2)$$

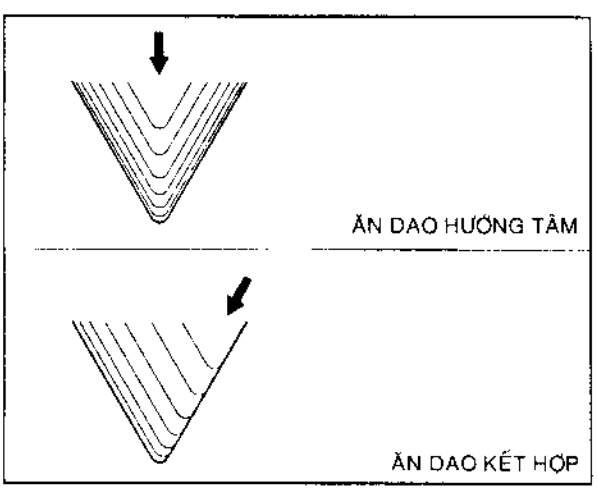
Trong ví dụ này, X là 2.8978, K là .0511 (hoặc P0511), chiều sâu cắt ren thứ nhất D = .0140, nhập vào chương trình là D0140 hoặc Q0140, tùy theo bộ điều khiển. Do đó Tf sẽ là:

$$T_f = 2.8978 + (.0511 \times 2) - (.0140 \times 2) = 2.9720$$

Kết quả hoàn toàn như các ví dụ đã nêu, nhưng do bộ điều khiển tính toán.

PHƯƠNG PHÁP ĂN DAO

Sự ăn dao vào vật liệu có thể được lập trình theo nhiều cách. Một trong các tùy chọn quan trọng nhất là phương pháp điều khiển dao tiến đến ren, còn gọi là *ăn dao tiện ren*. Đây là phương pháp chi tiết hóa các chuyển động của dao, sử dụng một trong *hai* cách ăn dao (Hình 37.10).



Hình 37.10. *Ăn dao hướng tâm và ăn dao kết hợp khi tiện ren*

Một phương pháp trong lập trình ren là ăn dao vào, đôi khi được gọi là phương pháp *hướng tâm*, còn được gọi là ăn dao *thẳng* hoặc *vuông góc*; phương pháp thứ hai là ăn dao *theo góc*, còn được gọi là ăn dao *kết hợp* hoặc ăn dao theo *mặt nghiêng* của ren.

Yêu cầu kiểm soát chiều ăn dao khi tiện ren là cung cấp các điều kiện cắt gọt tối ưu cho lưỡi cắt. Ngoại trừ các ren với bước tiến rất mịn và một số kim loại mềm, đa số các quy trình tiện ren đều sử dụng ăn dao kết hợp (theo góc). Một số loại ren bị loại trừ do dạng hình học của chúng – chẳng hạn ren vuông luôn luôn yêu cầu ăn dao vuông góc (ăn dao hướng tâm). Góc ăn dao được lập trình với tham số A của chu kỳ G76.

Từng phương pháp tiện ren đều có quy trình riêng, sử dụng các tính năng dưới đây:

- ❑ Lượng cắt gọt không đổi.
- ❑ Chiều sâu cắt không đổi.
- ❑ Cắt một lưỡi cắt.
- ❑ Cắt cả hai lưỡi cắt.

Tham số P của chu kỳ G76 lựa chọn tính năng theo yêu cầu.

Ăn dao hướng tâm

Phương pháp ăn dao *hướng tâm* của dao tiện ren là một trong các phương pháp gia công ren thông dụng nhất, có thể áp dụng *chuyển động thẳng* của dao cắt theo một chiều hướng đến đường kính được cắt gọt. Dao tiện ren sẽ ăn thẳng cho từng đường cắt mới. Đường kính đường cắt này được chuyên biệt theo dữ liệu X trong chương trình. Trong chu kỳ G76, tham số A0 được dùng cho sự ăn dao hướng tâm. Trong lập trình block G32 và chu kỳ G92 đơn giản, không có tham số để lập trình. Vị trí khởi đầu trục Z là không đổi đối với tất cả các đường kính ren và dễ lập trình hơn. Ăn dao hướng tâm thích hợp cho vật liệu mềm (đồng thau, nhôm, ...) nhưng có thể làm hư hại ren khi gia công vật liệu cứng.

Kết quả của chuyển động ăn dao hướng tâm là *cả hai* lưỡi cắt của dao đều đồng thời cắt gọt vật liệu. Do các lưỡi cắt đối diện với nhau, sự cuộn phoi sẽ đối diện với nhau. Trong nhiều ứng dụng, điều này sẽ làm tăng nhiệt độ và gây ra vấn đề mòn dao do nhiệt. Ngay cả khi giảm chiều sâu ăn dao cũng có thể không giải quyết

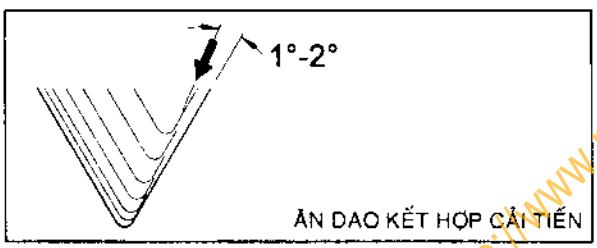
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

được vấn đề. Nếu ăn dao hướng tâm thông thường tạo ra ren chất lượng cao, cần áp dụng phương pháp ăn dao kết hợp.

Ăn dao kết hợp

Phương pháp ăn dao kết hợp, sử dụng chiều *góc* của dao khi chuyển động đến đường kính cắt ren. Hình dạng phoi tạo thành trong phương pháp ăn dao kết hợp tương tự hình dạng phoi tiện. Chỉ một lưỡi cắt của dao là thực sự cắt gọt, do đó nhiệt tiêu tán *ra xa* lưỡi cắt và phoi cuộn xoắn ra ngoài, kéo dài tuổi bền của dao. Chiều sâu của phoi có thể lớn hơn và cần ít đường cắt hơn. Hình 37.10 minh họa sự ăn dao kết hợp, một lưỡi cắt liên tục tiếp xúc với sườn ren. Ở đó không có sự cắt gọt, chỉ chà xát giữa hai bề mặt có thể làm giảm chất lượng bề mặt của ren. Để tránh vấn đề này, cần lập trình góc ăn dao hơi nhỏ hơn góc sườn ren (một nửa góc ren). Ren –V thông dụng, có góc ren 60^0 và góc sườn ren 30^0, góc ăn dao phải hơi nhỏ hơn, bạn có thể chọn giá trị 29^0. *Bạn cần nhớ* – dạng hình học của ren không được thay đổi, điều này do hình dạng dao cắt quyết định. Điều thay đổi là *cách thức* cắt gọt của dao (Hình 37.11).



Hình 37.11. *Góc ăn dao kết hợp được cải tiến để tăng chất lượng ren.*

Trong chu kỳ gia công ren G76, có các công cụ rất mạnh ở dạng tham số cắt gọt, hai tham số liên quan với phương pháp ăn dao. Một là *địa chỉ A* và thứ hai là *địa chỉ P*. Sự mô tả góc chỉ khả dụng trong định dạng hai-block, cặp chữ số cuối của địa chỉ P trong lệnh G76 thứ nhất – N49 G76P.. 60 Q... R..

Góc dao cắt ren – Tham số A

Đối với sự ăn dao kết hợp (mọi bộ điều khiển), giá trị *khác zero* được gán cho tham số biểu thị góc dao, giá trị bằng góc *gộp* của mảnh chắp cắt ren. Sự tiếp cận của dao hướng đến chi tiết sẽ hơi nhỏ hơn gộp A. Ví dụ, nếu A60 tiêu chuẩn được lập trình trong chu kỳ G76, góc ăn dao sẽ hơi nhỏ hơn 30^0, để lại khoảng hở nhỏ.

Chỉ có sáu xác lập góc A là được phép trong chu kỳ gia công ren G76:

A0	A29	A30	A55	A60	A80

A0	Góc dao công	ISO
A29	Ren ACME	ANSI
A30	Ren hình thang hệ mét	DIN103
A55	Ren Whitworth 55^0	BSW, BSP
A60	Ren V tiêu chuẩn 60^0	Hệ Anh hoặc hệ mét
A80	Ren PG Đức	Panserrohrgewinde

Để bổ sung thông tin và quy chiếu các dạng ren nêu trên, bạn hãy lưu ý ren ACME (góc 29^0) được dùng phổ biến trong các hệ truyền động, ví dụ con đội thay bánh xe hơi sử dụng ren ACME. Ren *hình thang hệ mét* là phiên bản hệ mét của ren ACME, với góc 30^0. Ren kiểu Whitworth có góc 55^0, có nguồn gốc từ Vương Quốc Anh; sự sử dụng loại ren này đang giảm dần, do các ren hệ mét trở thành tiêu chuẩn toàn cầu. Ren A80 PS, là ren ống đặc biệt của Đức (Panserrohrgewinde), với góc 80^0, ít được sử dụng ở Bắc Mỹ.

Trên các bộ điều khiển hiện đại (mức cao), có tham số P trong chu kỳ G76, hoạt động rất gần gũi với tham số A, và xác định *kiểu cắt gọt*.

Kiểu cắt gọt ren – Tham số P

Trong chu kỳ gia công ren G76, sự ăn dao có thể được lập trình với địa chỉ P, bổ sung cho địa chỉ A. Mục đích của tham số A là điều khiển phương pháp ăn dao cắt ren, cho đến chiều sâu xác định – dựa trên *góc* của mảnh chắp cắt ren. Đối với phương pháp ăn dao được điều khiển cao hơn, điều khiển *chiều sâu* cắt ren, cần sử dụng tham số P, lập trình trong định dạng chu kỳ G76, khả dụng trên các bộ điều khiển Fanuc 10T trở lên, xác định *kiểu cắt gọt ren*, liên quan với chiều sâu lập trình của ren.

Cùng với sự ăn dao hướng tâm, lập trình với tham số A0 và ăn dao kết hợp (tham số A khác zero), còn có hai kiểu cắt gọt khác có thể được dùng trong lập trình ăn dao cắt ren – *cắt một phía* và *cắt zig-zag* (chữ chi). Các thuật ngữ này biểu thị số lưỡi cắt được dùng vào một thời điểm. Cắt một phía là cắt gọt với *một lưỡi cắt*, cắt zig-zag là cắt gọt với *hai lưỡi cắt*. Từng kiểu cắt gọt này đều có thể được sử dụng phối hợp với tham số góc ren A đã chọn và chiều sâu cắt – *giá trị không đổi* hoặc *chiều sâu không đổi*.

Các bộ điều khiển Fanuc trên máy tiện CNC có bốn phương pháp điều khiển ăn dao và chiều sâu cắt ren (Hình 37.12).

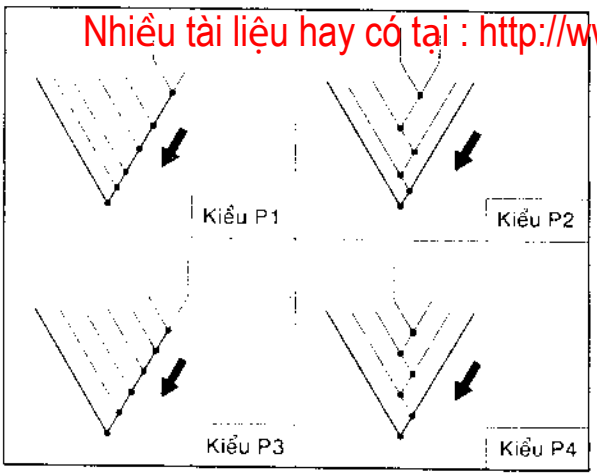
P1 Cắt gọt một lưỡi.. với lượng cắt gọt không đổi
P2 Cắt gọt hai lưỡi.. với lượng cắt gọt không đổi
P3 Cắt gọt một lưỡi.. với chiều sâu cắt gọt không đổi
P4 Cắt gọt hai lưỡi.. với chiều sâu cắt gọt không đổi

Trên các bộ điều khiển Fanuc được chế tạo

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 37.12. Các kiểu cắt gọt trong chu kỳ gia công ren G76 (tham số P) sử dụng trên các bộ điều khiển Fanuc 10/11/15T

trước loại 10T, không có tham số P trong chu kỳ G76. Sự tương đương với tham số P1 hiện nay trong các bộ điều khiển đó là mặc định. Trên các bộ điều khiển *có hỗ trợ* tham số P, nếu P bị bỏ qua trong lệnh gọi chu kỳ G76, phương pháp cắt gọt P1 sẽ là mặc định. Đây là ứng dụng gia công ren phổ biến nhất là thích hợp với nhiều loại ren, P1 sẽ áp dụng *một lưỡi cắt* của dao tiện ren và *lượng cắt gọt không đổi*, kết quả sẽ là lượng phoi đồng đều. Bạn nên thực nghiệm với ba kiểu P2, P3, P4 còn lại.

CÁC TÍNH TOÁN TRONG PHƯƠNG PHÁP MỘT BLOCK

Sự ăn dao kết hợp không gây ra các vấn đề lập trình khi sử dụng chu kỳ gia công ren G76. Nếu hệ thống CNC có chu kỳ G76, có thể áp dụng cho khoảng 95% các công việc gia công ren. Vậy, điều gì với 5% còn lại? Cần phải làm gì nếu không thể sử dụng chu kỳ G76 và chương trình cần sự ăn dao kết hợp được điều khiển hoàn toàn? Làm thế nào để điều khiển các phương pháp ăn dao khác khả dụng đối với G76, nhưng không có chu kỳ G76 hoặc không thể sử dụng chu kỳ đó?

Thật không may, chỉ có cách duy nhất – Dùng máy tính cầm tay tính toán từng vị trí dao và từng chuyển động dao một cách riêng rẽ. Điều này tuy tốn nhiều công sức nhưng đáng để thực hiện. Đây là công việc nên làm, do sự cải tiến nhỏ ở máy công cụ cũng có thể rất khó. Công việc lập trình cao nhất luôn luôn tốn thời gian và công sức nhưng chất lượng và độ chính xác của chi tiết phụ thuộc vào điều đó.

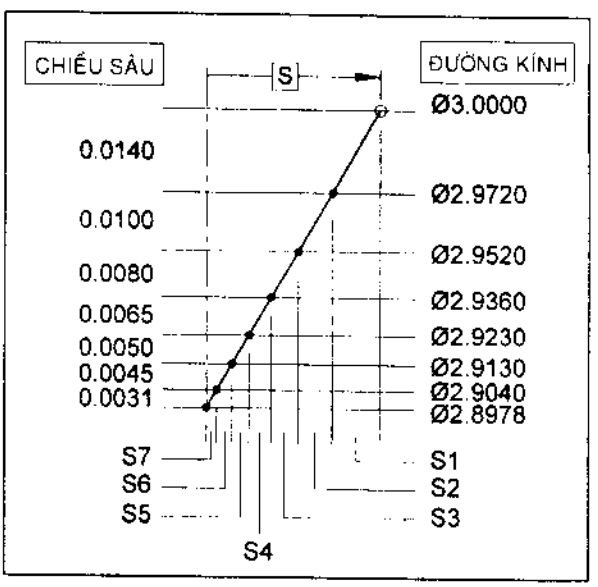
Các nguyên tắc của tiện ren kết hợp khi áp dụng cho lập trình theo block là khá đơn giản, nhưng công việc lập trình có thể nhàm chán và sự thể hiện có máy sẽ rất khó hoặc không khả thi. Từng đường tiện ren được tính toán theo vị trí khởi đầu khác nhau trên trục Z. Điều này được gọi là vị trí *dịch chuyển*, phải được tính một cách chính xác, nếu không đủ chính xác, chương trình sẽ bị lỗi. Vị trí này cũng phải đúng ở ngay lần tính đầu tiên, nếu không chuẩn, các thay đổi và chỉnh sửa sẽ tốn nhiều thời gian. Trong ví dụ này, ren sẽ được gia công như ví dụ trước (3.0-12 TPI). Chương trình sẽ sử dụng lệnh G32, với góc ăn dao kết hợp 29^0.

Các xem xét ban đầu

Ren được dùng trong các ví dụ của phần này là ren ngoài 3.0 – 12 TPI. Mọi đường kính riêng rẽ đã được tính toán ở phần trước, và mọi chiều sâu của từng đường cắt cũng đã được thiết lập. Các giá trị đó sẽ được dùng trong ví dụ này. Toàn bộ có bảy đường cắt và bảy chiều sâu cần lập trình với giá trị Z (tổng chiều sâu là .0511 theo bán kính) (Hình 37.13).

Đường cắt #1	chiều sâu đến Ø2.9720	(Chiều sâu cắt .0140)
Đường cắt #2	chiều sâu đến Ø2.9520	(Chiều sâu cắt .0100)
Đường cắt #3	chiều sâu đến Ø2.9360	(Chiều sâu cắt .0080)
Đường cắt #4	chiều sâu đến Ø2.9230	(Chiều sâu cắt .0065)
Đường cắt #5	chiều sâu đến Ø2.9130	(Chiều sâu cắt .0050)
Đường cắt #6	chiều sâu đến Ø2.9040	(Chiều sâu cắt .0045)
Đường cắt #7	chiều sâu đến Ø2.8978	(Chiều sâu cắt .0031)

Ngoài các chiều sâu cắt và đường kính, Hình 37.13 còn nêu các giá trị *dịch chuyển* S từ S1 đến S7. Khi vị trí khởi đầu Z dịch chuyển, khoảng cách dịch chuyển phải được tính toán trên cơ sở góc kết hợp và chiều sâu đường cắt ren. Tính toán kế tiếp phải dựa trên kết quả trước đó.



Hình 37.13. Các tính toán ăn dao kết hợp chương trình tiện ren theo block với G32.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Khoảng cách minh họa S biểu thị *tổng* giá trị dịch chuyển từ vị trí khởi đầu trục Z danh định, trong ví dụ này, giá trị lập trình là Z0.25. Sự dịch chuyển theo chiều dương trục Z.

Về lý thuyết, không có sự khác biệt về *chiều* dịch chuyển được lập trình – hướng vào ren hay ra xa ren. Trong thực tiễn – nên lập trình sự dịch chuyển *ra xa* ren, nếu có thể – ụ động có thể là vật cản, do đó cần lưu ý vị trí của ụ động. Theo đó, khoảng cách gia tăng tốc độ cắt sẽ *tăng* thay vì giảm.

Mặc dầu vẫn có thể chọn cách tiếp cận khác, nhưng cần tính toán khoảng cách S trước. Chiều sâu ren toàn phần là .0511 và góc ăn dao kết hợp được chọn là 29^0, do đó sử dụng hàm lượng giác sẽ tính được khoảng cách S:

$S = .0511 \times \tan 29^0 = .028325193$

Khoảng cách S biểu thị *tổng* giá trị dịch chuyển của dao tiện ren. Độ dịch chuyển đối với từng đường cắt ren sẽ là một phần của S, ký hiệu là S_x (Hình 37.13) trong khoảng S1 đến S7.

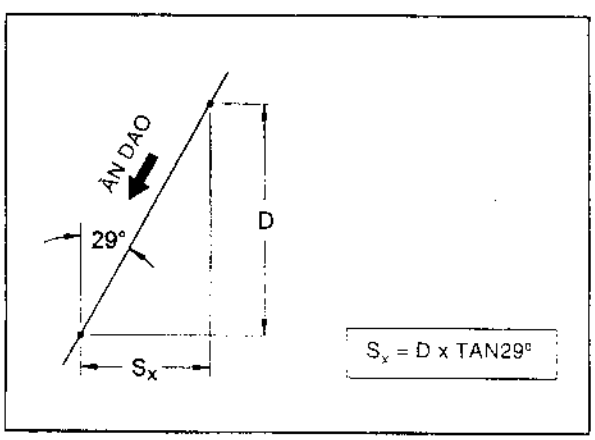
$$S_x = D \times \tan 29^0$$

Trong đó:

Sx = Độ dịch chuyển của đường cắt ren hiện hành – số gia

D = Chiều sâu của đường cắt ren hiện hành.

Tính toán từng đường cắt sử dụng cùng công thức nêu trên, chỉ thay đổi giá trị D. Bạn cần nhớ mục đích của quá trình này là tìm vị trí khởi đầu Z mới cho *từng đường cắt ren* nghĩa là giá trị Z đối với đường kính ren đã biết (Hình 37.14)



Hình 37.14. Tính toán vị trí khởi đầu của ren – trục

Sau khi biết vị trí khởi đầu trục Z đối với chiều sâu đường cắt thứ nhất, sẽ dễ dàng tính toán các vị trí khởi đầu cho từng chiều sâu cắt kết tiếp. Bạn đã biết vị trí khởi đầu trục Z đối với dao cắt ren đã được thiết lập là Z0.25 *cộng* với giá trị tính toán .028325193 được làm tròn từ giá trị tính toán .028325193. Vị trí khởi đầu *lý thuyết* sẽ là Z0.2783, được tính từ ∅3.000, nhưng *không được sử dụng trong chương trình*. Giá trị ban đầu này là cần thiết cho tất cả các giá trị còn lại. Đối với từng tính toán kế tiếp, giá trị S_x sẽ *trừ* từ vị trí khởi đầu Z *hiện hành*. Danh mục dưới đây sẽ liệt kê từng giá trị dịch chuyển S_x (được làm tròn theo đơn vị Anh):

```
S1 =  .0140 x tan29   = .0078
S2 =  .0100 x tan29   = .0055
S3 =  .0080 x tan29   = .0044
S4 =  .0065 x tan29   = .0036
S5 =  .0050 x tan29   = .0028
S6 =  .0045 x tan29   = .0025
S7 =  .0031 x tan29   = .0017
Tổng                    .0283
```

Bảy vị trí dịch chuyển đối với vị trí trục Z khởi đầu có thể tính toán dễ dàng, dựa trên vị trí khởi đầu lý thuyết Z0.2783 tại ∅3.000:

```
#1 =     S - S1  = .2783 - .0078 = .2705
#2 = .2705 - S2  = .2705 - .0055 = .2650
#3 = .2650 - S3  = .2650 - .0044 = .2606
#4 = .2606 - S4  = .2606 - .0036 = .2570
#5 = .2570 - S5  = .2570 - .0028 = .2542
#6 = .2542 - S6  = .2542 - .0025 = .2517
#7 = .2517 - S7  = .2517 - .0017 = .2500
```

Ví dụ này biểu thị bù vị trí ban đầu cách xa ren, sau đó mỗi lần sẽ dịch chuyển một bậc về vị trí gốc Z0.25. Sử dụng phương pháp này, bạn sẽ nhận thấy giá trị được xác lập ban đầu .250 inch là khoảng hở tối thiểu và không được phép nhỏ hơn giá trị đó. Chỉ giá trị trục Z là thay đổi – các giá trị lập trình khác hoàn toàn không bị ảnh hưởng trong phương pháp lập trình này.

Chương trình hoàn chỉnh hơi dài (đặc trưng của lập trình với G32), nhưng sẽ minh họa phương pháp ăn dao kết hợp (theo góc) khi không có chu kỳ khả dụng hoặc khó sử dụng. Trong ví dụ này chỉ lập trình dao cắt ren:

```
O3705
(COMPOUND INFEED EXAMPLE)
...
(N45 G50 X12.0 Z4.5)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X3.2 Z0.2705 T0505 M08      (START 1)
N49 X2.972                           (PASS 1)
N50 G32 Z-1.6 F0.0833        (or F/E0.083333)
N51 G00 X3.2
N52 Z0.265                           (START 2)
N53 X2.952                           (PASS 2)
N54 G32 Z-1.6
N55 G00 X3.2
N56 Z0.2606                          (START 3)
N57 X2.9360                          (PASS 3)
N58 G32 Z-1.6
N59 G00 X3.2
N60 Z0.257                           (START 4)
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N61 X2.9330                          (PASS 4)
N62 G32 Z-1.6
N63 G00 X3.2
N64 Z0.2542                          (START 5)
N65 X2.9130                          (PASS 5)
N66 G32 Z-1.6
N67 G00 X3.2
N68 Z0.2517                          (START 6)
N69 X2.9040                          (PASS 6)
N70 G32 Z-1.6
N71 G00 X3.2
N72 Z0.25                            (START 7)
N73 X2.8978                          (PASS 7)
N74 G32 Z-1.6
N75 G00 X3.2
N76 Z0.25 M09
N77 X12.0 Z4.5 T0500
N78 M30
%
```

Trong chương trình O3705, phương pháp ăn dao là tương đương với tham số P1 trong chu kỳ G76. Kiểu cắt ren này chỉ sử dụng một lưỡi cắt của dao, với lượng ăn dao không đổi trong từng đường cắt ren, biểu thị phương pháp lập trình tiện ren phổ biến và có thể dùng làm mẫu cho nhiều ứng dụng cắt ren khác. Lập trình cắt ren theo block sẽ khá dài, cần kiểm tra độ chính xác một cách rất cẩn thận.

CHUYỂN ĐỘNG LÙI DAO

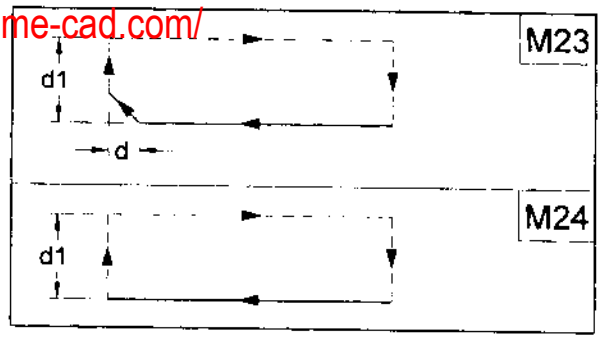
Ở phần đầu đã nêu rõ chỉ có hai phương pháp lùi dao ra khỏi ren - chuyển động thẳng theo *một trục* và chuyển động từ từ đồng thời theo *hai trục*. Cả hai đều được dùng trong lập trình tiện ren. Các ứng dụng của chúng minh chứng cho các hàm M trong hệ điều khiển là tính năng tiêu chuẩn. Các hàm lùi dao tiện ren này được gọi là *hàm vạt góc ren* hoặc hàm *hoàn tất ren*.

Các hàm lùi dao tiện ren

Khi sử dụng các chu kỳ gia công ren G92 và G76 đối với máy tiện CNC, phần kết thúc ren (giá trị trục Z) sẽ có thể trong phần ren đã gia công hoặc ở bề mặt phôi. Sự lùi dao thực tế có thể được lập trình theo một trục hoặc đồng thời theo hai trục. Các hàm của Fanuc được thiết kế cho mục đích này là M23 và M24. Chúng điều khiển chuyển động lùi dao ra khỏi ren ở phần kết thúc ren:

M23	Hoàn tất ren ON	(hai trục)
M24	Hoàn tất ren OFF	(một trục)

Các bộ điều khiển khác cũng có thể có các hàm tương tự. Mục đích của các hàm này là kích hoạt hoặc ngắt *sự chèn tự động* lùi dao giữa chuỗi thứ tự chuyển động cắt ren 2 và 3, như đã đề cập ở đầu Chương. Hình 37.15 so sánh chuyển động cắt ren có và không có sự lùi dao.



Hình 37.15. Các hàm M điều khiển sự lùi dao từ từ.

Lùi dao một trục

Sự lùi dao một trục (M24, hoàn tất ren OFF) là chuyển động nhanh đơn giản được lập trình ở cuối đường cắt ren, được coi là *chuyển động thứ ba* trong chuỗi – thứ tự bốn chuyển động cơ bản. Chiều lùi dao luôn luôn là 90^0 so với trục ren. Đối với chu kỳ gia công ren G92 hoặc G76, đây là điều kiện mặc định, do đó M24 là không cần thiết, trừ khi sử dụng M23, thường là cho ren khác trong cùng một chương trình. Hai hàm này xóa lẫn nhau. Nếu dùng M24, cần phải lập trình *trước* chu kỳ gia công ren áp dụng hàm đó. Ví dụ, chương trình tiện ren O3703 sử dụng chu kỳ G76 sẽ được chỉnh sửa đôi chút trong O3706:

```
O3706
...
N45 (G50 X12.0 Z4.5) M24 (THREAD PULLOUT OFF)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X3.2 Z0.25 T0505 M08
N49 G76 X2.8978 Z-1.6 I0 K0.0511 D0140 A60
    P4 F0.0833                  (or F/E0.083333)
N50 G00 X12.0 Z4.5 T0500 M09
N51 M30
%
```

Hàm M24 xuất hiện trong block N45, block khả dụng duy nhất mà không có hàm M khác.

Lùi dao hai trục

Lùi dao hai trục là chuyển động dao từ từ theo góc, dọc theo hai trục để ra khỏi ren (hoàn tất ren ON, M23). Chương trình O3707 tương tự chương trình O3706.

```
O3707
...
N45 (G50 X12.0 Z4.5) M23 (THREAD PULLOUT ON)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X3.2 Z0.25 T0505 M08
N49 G76 X2.8978 Z-1.6 I0 K0.0511 D0140 A60
    P4 F0.0833                  (or F/E0.083333)
N50 G00 X12.0 Z4.5 T0500 M09
N51 M24                              (CANCEL M23)
N52 M30
%
```

Trong trường hợp này, M23 được áp dụng

Nhiều tài liệu hay có tại : http://www.me-cad.com/

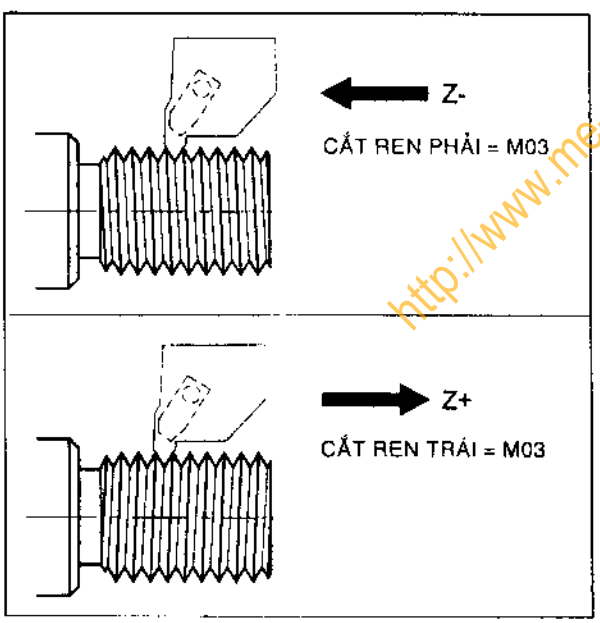
Nhiều tài liệu hay có tại : http://www.me-cad.com/

trong block tại điểm bắt đầu M51 được dùng để xóa sự lùi dao. Sự xóa là không cần thiết trong chương trình này, nhưng trong thực tế nên xóa các hàm chỉ dùng cho những mục đích đặc biệt.

Có một số điều kiện đặc biệt áp dụng cho hàm M23. Trên Hình 37.15, khoảng cách hoàn tất d được xác lập bằng tham số điều khiển, trong khoảng .100× đến 12.700× bước tiến ren. Xác lập điều khiển bình thường *đúng bằng một* bước tiến ren. Góc lùi dao ra khỏi ren thường là 45^0, hoặc hơi nhỏ hơn do sự trễ trong hệ thống trợ động. Nếu khoảng cách hoàn tất d lớn hơn khoảng cách lùi dao d1, sự lùi dao sẽ *không thể* thực hiện.

CHIỀU REN

Ren bất kỳ có thể được cắt theo *chiều phải* (ren phải) hoặc *chiều trái* (ren trái). Sự lựa chọn bất kỳ đều không ảnh hưởng đến biên dạng và/hoặc chiều sâu ren, nhưng các yếu tố khác là rất quan trọng. Hầu hết các ứng dụng ren đều sử dụng ren phải. *Ren phải* và *ren trái* có quan hệ chặt chẽ với đường cong xoắn của ren (Hình 37.16).



Hình 37.16. *Cắt ren phải (trên) và ren trái (dưới) sử dụng giá dao cắt ren phải (lắp ngược).*

Chiều ren được xác định theo hai điều kiện:

- ❑ Chiều cắt của dao (Z+ hoặc Z-)
- ❑ Chiều quay trục chính (M03 hoặc M04)

Các điều kiện này được dùng phối hợp để lập trình gia công ren cụ thể. Các yếu tố ảnh hưởng đến phương pháp lập trình đối với ren phải và ren trái gồm:

- ❑ Thiết kế dao tiện ren – trái hoặc phải
- ❑ Chiều quay trục chính – M03 hoặc M04
- ❑ Chiều cắt gọt – Z+ hoặc Z-
- ❑ Định hướng mũi dao trong ổ dao

Về lý thuyết, ren phải hoặc trái đều có thể cắt với dao tiện ren *bất kỳ*, nhưng điều này khó áp dụng trong thực tế. Sự lựa chọn không chuẩn sẽ tác động xấu đến chất lượng ren, tuổi bền của dụng cụ cắt, làm tăng chi phí,... Khi đầu ren ở gần vai hoặc bậc, khoảng hở để tăng tốc độ cắt bị hạn chế. Phương pháp duy nhất để tránh gia công ren kém chất lượng do sự gia tốc trong không gian hẹp là *giảm tốc độ* trục chính.

TIỆN REN GẦN VAI TRỤC

Lập trình gia công ren ở sát vai hoặc bậc có một khó khăn đặc thù. Khó khăn đó là bậc hoặc vai của chi tiết. Sẽ là không đủ nếu chỉ lập trình điểm kết thúc của ren một cách hợp lý, cần phải lập trình *một cách chính xác*. Ngay cả khi đó vẫn có thể xảy ra va chạm nếu gá lắp dao không chuẩn. Ba vấn đề trong lĩnh vực lập trình ren này, bao gồm:

- ❑ Rãnh thoát dao quá hẹp hoặc không có rãnh đó.
- ❑ Lưỡi dao tiện ren quá rộng
- ❑ Ren quá sâu

Vấn đề thứ nhất khi tiện ren hướng đến bậc, rãnh thoát dao quá hẹp, tương đối dễ giải quyết, chỉ cần tăng chiều rộng này trong chương trình. Hầu hết các rãnh thoát dao đều có thể được điều chỉnh cho dao tiện ren, mà không ảnh hưởng đến ý đồ kỹ thuật của thiết kế. Điều này có thể phải sửa đổi đôi chút trên bản vẽ – nhưng bạn hoàn toàn có thể kiểm tra trước khi chỉnh sửa.

Vấn đề thứ hai và thứ ba có thể không liên quan với nhau, nhưng giải pháp thường là chung cho cả hai. Nếu lưỡi dao tiện ren quá rộng hoặc ren quá sâu, bạn hãy thử tăng chiều rộng rãnh thoát dao trước, nếu có thể. Nếu không thể tăng chiều rộng rãnh, vì lý do nào đó, bạn có lựa chọn thứ hai là *giảm* chiều rộng lưỡi dao cắt ren. Giải pháp đơn giản là dùng dao nhỏ hơn vẫn có thể cắt chiều sâu ren theo yêu cầu. Đây có thể là dao với kích cỡ nhỏ hơn một bậc, và có thể dùng cán dao khác.

Nếu không thể dùng dao nhỏ hơn, bạn hãy lập trình *dao tiện ren hiện hữu được chỉnh sửa*. Sự chỉnh sửa trong trường hợp này có nghĩa là mài bớt một phần lưỡi dao theo chiều cắt gọt, mà không ảnh hưởng đến phần thực sự cắt gọt. Trước khi quyết định chỉnh sửa bằng phương pháp mài, bạn hãy xét các lựa chọn khác một cách cẩn thận – sửa đổi các dao tiêu chuẩn được thiết kế cho máy CNC luôn luôn là giải

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

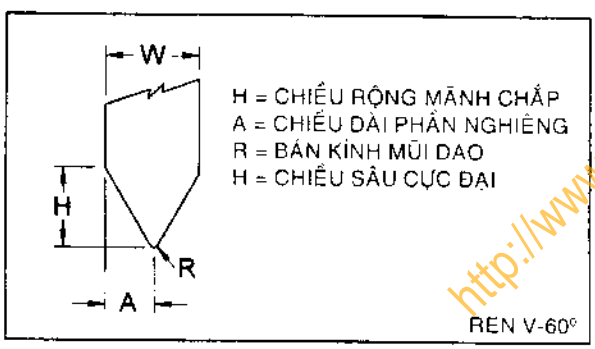
pháp *cuối cùng*. Mảnh chắp có lớp tráng phủ sẽ bị giảm các ưu thế cắt gọt, nếu lớp này bị mất trong khi mài. Cần rất cẩn thận, không mài lớp tráng phủ trong phần cắt gọt của lưỡi cắt. Trong trường hợp chương trình sử dụng mảnh chắp tiện ren được chỉnh sửa, bạn hãy xét vài đề nghị dưới đây.

Luôn luôn cẩn thận khi dùng các dao được chỉnh sửa.

Chỉnh sửa mảnh chắp

Hiện có nhiều loại mảnh chắp tiện ren tiêu chuẩn, bạn có thể tìm được mảnh chắp thích hợp cho công việc của mình. Nếu mảnh chắp tiện ren tiêu chuẩn cần được chỉnh sửa, ví dụ dưới đây sẽ minh họa vài xem xét lập trình – bất kể *có* hay *không có* rãnh thoát dao trên chi tiết.

Để chỉnh sửa mảnh chắp tiện ren tiêu chuẩn, trước hết bạn hãy quan sát cấu hình bình thường của mảnh chắp. Hình 37.17 minh họa mảnh chắp tiện ren, biết chiều rộng W, chiều dài phần nghiêng A, bán kính mũi dao R, nhưng chưa biết chiều cao phần nghiêng H.



Hình 37.17. *Các kích thước cơ bản của mảnh chắp tiện ren*

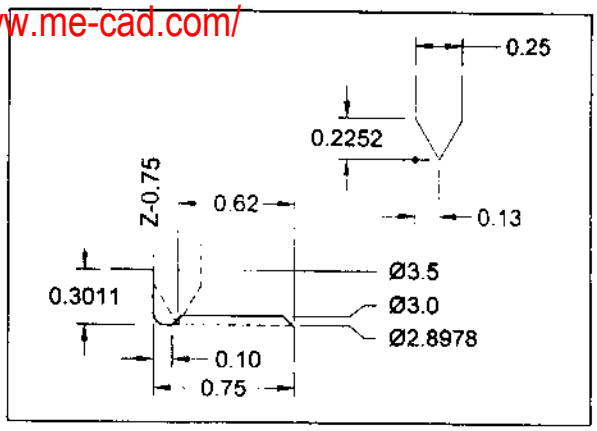
Trong ví dụ này, kích thước W là .250 inch và A là .130. Góc xen giữa của mảnh chắp là 60^0 và bán kính mũi dao R là .012 không thích hợp trong trường hợp này. Kích thước H biểu thị chiều sâu ren cực đại và thường được đo từ đầu nhọn của mũi dao, được tính bằng hàm lượng giác:

$H = A / \tan 30^0$
$H = .130 / .577350269$
$H = .225166605$
$H = .2252$

Vấn đề được minh họa trên Hình 37.18

Công việc là lập trình ren với chiều rộng rãnh thoát dao .100, sử dụng dao có chiều dài phần nghiêng A = .130. Mảnh dao này là không thích hợp, và không thể hoàn tất chiều dài ren tối thiểu với chiều sâu toàn phần. Hiệu số giữa chiều dài bậc và chiều rộng rãnh thoát dao là:

.750 - .100 = .650



Hình 37.18. *Mảnh chắp tiện ren trước khi chỉnh sửa. không vừa với phần rãnh thoát dao.*

Chiều dài ren tối thiểu trong minh họa chỉ là .620. Không có các khoảng hở và chiều dài ren quá ngắn. Để giải quyết vấn đề này, bạn hãy chọn cỡ dao nhỏ hơn, nếu có thể. Nếu không, *chỉnh sửa* mảnh chắp là giải pháp duy nhất.

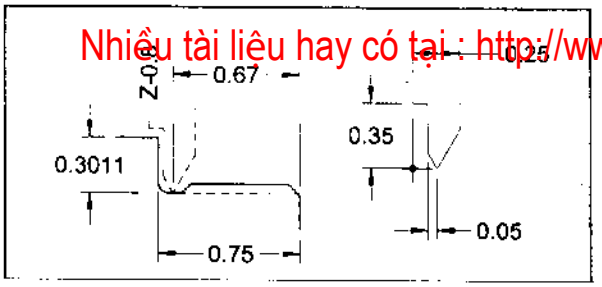
Sự chỉnh sửa này đòi hỏi phải mài mảnh chắp ở phần không quan trọng, để cho phép dao hoàn tất chiều dài ren tối thiểu .650 inch. Về lý thuyết, lượng mài tối thiểu của mảnh chắp là .030, hiệu số giữa chiều dài ren thực và chiều dài yêu cầu. Sự chỉnh sửa này không tạo ra *khoảng hở* bất kỳ ở bậc ren hoặc ở mũi dao. Cả hai khoảng hở này đều rất quan trọng đối với chất lượng ren. Ngay cả sai số gá lắp nhỏ trên máy cũng có thể gây ra khó khăn nghiêm trọng.

Luôn luôn tính toán lượng chỉnh sửa, không được ước lượng.

Trong ví dụ này, có ba kích thước ảnh hưởng đến lượng chỉnh sửa mảnh chắp. Tổng của chúng sẽ là giá trị cần mài trên mảnh chắp. Một, chiều dài ren sẽ được kéo dài thêm .030 để đạt được chiều dài đến tối thiểu .650. Hai, khoảng hở này do nhà lập trình quyết định. Giải pháp là tổng giá trị chỉnh sửa dao sẽ là .080. Nói cách khác, mảnh chắp tiện ren cần mài bớt .080 inch. Điều này sẽ rút ngắn chiều dài phần nghiêng ban đầu của dao là .130 - .080 = .050. Bạn phải bảo đảm có thể đạt được chiều sâu ren với dao được chỉnh sửa. Chương trình gia công sẽ phản ánh sự chỉnh sửa này theo vị trí kết thúc ren trên trục Z, và được viết là Z-0.8 (vị trí gá lắp dao không đổi), như minh họa trên Hình 37.19

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 37.19. Mảnh chắp tiện ren được chỉnh sửa để có đủ khoảng trống ở rãnh thoát dao

Trong gia công ren, chiều dài ren là *chiều dài thực của ren có chiều sâu toàn phần*. Thiết kế chi tiết có thể cho phép ren hơi dài hơn, nhưng không được ngắn hơn. Chiều cao của bậc cũng rất quan trọng. Trong ví dụ này, chiều cao là .3011 và sự chỉnh sửa mảnh chắp là có thể. Mảnh chắp tiện ren lớn có thể không cần chỉnh sửa, giải pháp duy nhất là sử dụng kích cỡ dao nhỏ hơn.

Kiểm nghiệm chương trình

Dù dao tiện ren là tiêu chuẩn hay được chỉnh sửa, thì tiện ren đến vai hoặc bậc cũng làm cho người vận hành CNC lo lắng, khi gia công chi tiết thứ nhất. Do sự vượt qua tốc độ cắt và các công tắc duy trì cắt gọt đều không hoạt động trong khi tiện ren, sự kiểm nghiệm chương trình trên máy tiện trở nên khó khăn hơn. Ngay cả các phương pháp kiểm nghiệm bằng đồ họa (mô phỏng) trên máy tính cũng có thể không phát hiện được sự va chạm tiềm ẩn.

Phương pháp kiểm nghiệm chương trình ren đơn giản nhưng rất hiệu quả luôn luôn khả dụng ngay trên máy tiện CNC. Phương pháp này đòi hỏi người vận hành CNC phải có tay nghề cao, hiểu rõ cả chương trình và các nguyên tắc tiện ren, kiến thức về công nghệ cũng rất quan trọng.

Phương pháp này sử dụng một số đặc tính trên các bộ điều khiển CNC hiện đại. Mục đích của kiểm nghiệm chương trình là xác định khả năng dao tiện ren va chạm với bậc *trước khi* xảy ra sự tiện ren thực tế.

Các bước sau đây chỉ có tính tổng quát, cần chỉnh sửa phù hợp với các điều kiện cụ thể , khi kiểm nghiệm chương trình gia công ren:

- ❑ Sử dụng chế độ BLOCK ĐƠN và lần lượt kiểm tra cho đến vị trí khởi đầu của ren.
- ❑ Chuyển từ chế độ AUTO sang MANUAL – trục chính dừng và dao ở khoảng trống
- ❑ Chọn biểu thị màn hình XZ (chế độ tuyệt đối)
- ❑ Chuyển sang chế độ HANDLE cho trục Z
- ❑ Trong khi quan sát hiển thị vị trí XZ, hãy dịch chuyển handle cùng chiều với ren, cho đến khi dao tới vị trí Z lập trình, hoặc không thể dịch chuyển xa hơn.
- ❑ Nếu dao đến vị trí Z lập trình trước, sự gá lắp dao là an toàn đối với tiện ren.
- ❑ Nếu dao chỉ chạm vào chi tiết nhưng không đạt đến vị trí Z cuối được lập trình của ren, cần điều chỉnh gá lắp dao theo hiệu số giữa vị trí lập trình và vị trí thực, công thêm khoảng hở nhỏ.

Còn có các phương pháp kiểm nghiệm khác, ví dụ, tạm thời sử dụng *lệnh chuyển động tuyến tính G01*, thay cho *lệnh gia công ren G32*, mà không có chi tiết lắp trong trục chính.

Trong chế độ không – tiện ren, sự vượt qua tốc độ cắt sẽ có hiệu lực, nhưng không có hiệu lực trong chế độ tiện ren.

Bằng cách đọc vị trí dao hiện hành trên màn hình và so sánh với vị trí lập trình, có thể xác định dao có bị va chạm hay không. Trong khi kiểm nghiệm, tốc độ cắt có thể chậm dần hoặc dừng hẳn vào thời điểm bất kỳ. Mục đích của kiểm nghiệm chương trình là thiết lập các điều kiện làm việc an toàn *trước khi* thực hiện tiện ren.

CÁC DẠNG REN KHÁC

Tuy ren V tiêu chuẩn với góc đỉnh 60^0 là dạng ren thông dụng nhất, nhưng không phải là duy nhất. Có nhiều dạng ren, nhà lập trình có thể gặp trong xưởng cơ khí.

Ví dụ, bạn hãy xét ren ACME. Trong hệ mét, ren tương đương được gọi là *ren thang hệ mét*. Từ góc độ lập trình, cả hai loại ren này hầu như đồng nhất với nhau. Ren ACME có góc 29^0, và ren thang hệ mét có góc 30^0 với dạng hình học hơi khác.

Ứng dụng chính của ren thang là *truyền chuyển động*, thường với sự nhả khớp. Một số loại trục vitme trên máy tiện sử dụng loại ren này. Lập trình ren thang có thể phải sử dụng giá đỡ cố định, do các ren này có thể khá dài. Điều quan trọng là xem xét sai số tích lũy trên khoảng cách dài.

Chiều sâu ren

Mỗi loại ren đều có công thức và quan hệ toán học riêng. Có hai công thức cơ bản liên quan với chiều sâu ren ACME. Một dùng cho ren 10TPI và *thô hơn*, thứ hai dùng cho ren 12TPI và *mịn hơn*. Đối với các ren ACME không quá 10TPI, công thức chiều sâu ren như sau:

$$T_d = .500 \times P + .010$$

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Đối với ren ACME từ 12TPI trở lên, công thức tính chiều sâu ren như sau:

$$T_d = .500 \times P + .005$$

Trong đó:

T_d = Chiều sâu ren
P = Bước ren

Các ren khác trong nhóm hình thang gồm *Stub ACME* hoặc 60^0 *Stub ACME*. Lập trình ren thang không khó hơn ren V, nhưng nhà lập trình phải biết các công thức và chi tiết hình học của thiết kế ren.

Các ren không thuộc nhóm 60^0 gồm *ren vuông*, ren API (dùng trong công nghiệp dầu khí), ren *Buttress*, ren *Aero*, ren tự khóa *Dardelet*, ren *tròn*, ren *Lebus* (yêu cầu các tính năng điều khiển đặc biệt), và nhiều loại khác.

REN CÔN

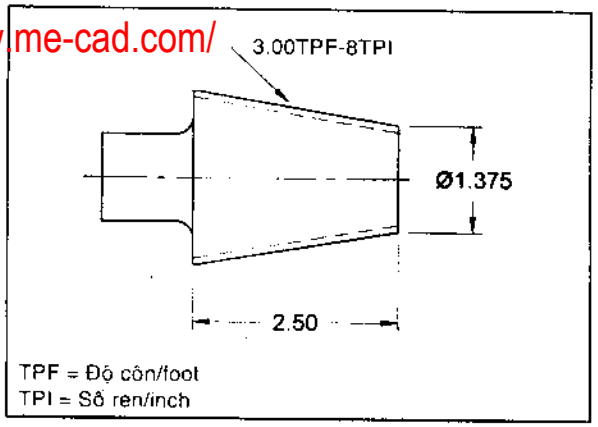
Các bước lập trình đối với ren côn hầu như không khác biệt so với ren thẳng. Chuyển động tiện ren côn đồng thời theo *hai trục*, thay vì một trục. Bốn bước chuyển động cơ bản hầu như đồng nhất với các chuyển động khi tiện ren trục:

Chuyển động 1	Chạy dao nhanh từ vị trí khởi đầu đến đường kính tiện ren
Chuyển động 2	Cắt ren (cắt theo hai trục)
Chuyển động 3	Lùi dao ra khỏi ren
Chuyển động 4	Trở về vị trí khởi đầu

Khi so với ren trụ, các khác biệt lập trình đối với ren côn là ở hai chuyển động đầu – *chuyển động 3* và *chuyển động 4* không thay đổi. Trong *chuyển động 1*, vị trí khởi đầu của dao được xác định theo chiều định hướng của dao tiện ren, dù là ren trong hay ren ngoài.

Đối với các dạng ren *ngoài*, vị trí khởi đầu của dao tiện ren phải luôn luôn *phía trên đường kính lớn nhất* của ren. Đối với các dạng ren *trong*, vị trí khởi đầu phải *ở dưới đường kính nhỏ nhất của ren*. Đây cũng là yêu cầu đối với ren thẳng, nhưng đối với ren côn điều này có tầm quan trọng lớn hơn. Đối với các ví dụ về ren côn, bạn hãy xét bản vẽ trên Hình 37.20.

Ren được xác định theo chiều dài toàn phần (2.500), đường kính ở đầu chi tiết phôi (1.375), góc côn (3.000 inch/foot), và bước ren (8 TPI). Đây là ren một đầu mối và zero chương trình sẽ ở mặt đầu của chi tiết hoàn tất. Mọi nguyên công cắt gọt trước đó đều đã được thực hiện cho chi tiết. Xem xét lập trình thứ nhất đối với kiểu gia công này sẽ là chiều sâu ren.



Hình 37.20. *Ví dụ về ren côn – Chương trình O3708*

Chiều sâu và các khoảng hở

Từ công thức ở phần trước, chiều sâu D của ren được dùng trong chương trình sẽ là:

D = .61343 / 8 = .0766788 = .0767

Chiều sâu ren này được đo hướng trục và không liên quan với góc ren. Sau khi biết chiều sâu, có thể xác lập các khoảng hở – một ở đầu và một ở cuối ren. Giá trị khoảng hở trục Z sẽ phụ thuộc sự tăng tốc độ của dao. Do tốc độ cắt sẽ được lập trình là F0.125, và theo nguyên tắc bốn lần bước tiến ren, khoảng hở yêu cầu sẽ là .500 inch. Chỉ cần .400 inch là đủ với tốc độ trục chính tương đối chậm (450 r/min). Khoảng hở cuối có thể nhỏ hơn, .200 là khoảng hở hợp lý ở cuối ren, dù vẫn có thể sử dụng giá trị nhỏ hơn.

Đối với ren côn, bạn hãy xét chiều dài *toàn phần* của hành trình dao dọc theo từng trục, không phải chiều dài ren thực trên bản vẽ. Chiều dài hành trình dao trong ví dụ này sẽ là sự kết hợp của hai khoảng hở đã chọn cộng với chiều dài ren đã biết (dọc theo trục Z):

.400 + 2.500 + .200 = 3.100

Bước kế tiếp không phải lúc nào cũng cần thiết, tùy thuộc vào phương pháp lập trình. Nếu dùng phương pháp block, cần *cả hai* đường kính đầu và cuối cho từng đường cắt. Nếu dùng chu kỳ G76 hoặc G92. Chỉ cần khoảng cách giữa đường kính đầu và cuối của ren côn. Khoảng cách này sẽ được lập trình theo tham số I của chu kỳ tiện ren và là một phần trong tính toán độ côn. Tất cả các ví dụ đã nêu ở phần trên đều là ren thẳng với giá trị I là zero (I0).

Tính toán độ côn

Cần tính độ côn của ren để thiết lập các đường kính đầu và cuối. Phương pháp tính toán phụ thuộc vào cách thức xác định độ côn và ghi kích thước trên bản vẽ. Kích thước độ côn trên bản vẽ thường không đủ cho lập trình – cần

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

phải tính toán chính xác bằng một trong hai phương pháp.

Phương pháp thứ nhất sử dụng chiều dài ren và góc côn theo các hàm lượng giác cơ bản. Phương pháp thứ hai xác định độ côn theo *tỷ số của các cạnh*. Các tỷ số này được ghi trên bản vẽ, ví dụ 1:12, 1:16,... hoặc *độ côn/foot*, *độ côn/inch*. Bạn cần nhớ nguyên tắc:

Độ côn luôn luôn đo theo đường kính.

Ren ống tiêu chuẩn Bắc Mỹ là dạng ren côn điển hình, được xác định theo tỷ số côn 1:16, tương đương độ côn *3/4 inch/foot* đo trên đường kính và vuông góc với trục. Ren ống còn được định nghĩa theo góc cho trước, ví dụ $1^0 47'23''$. Tương đương 1.789910608^0 tính theo số thập phân. Đối với lập trình CNC, thường dùng đơn vị độ theo số thập phân thay vì độ – phút – giây. Để hiểu nguyên tắc độ côn tính theo tỷ số giữa các cạnh, bạn hãy xem định nghĩa và ví dụ.

TỶ SỐ biểu thị quan hệ giữa hai giá trị, viết dưới dạng phân số.

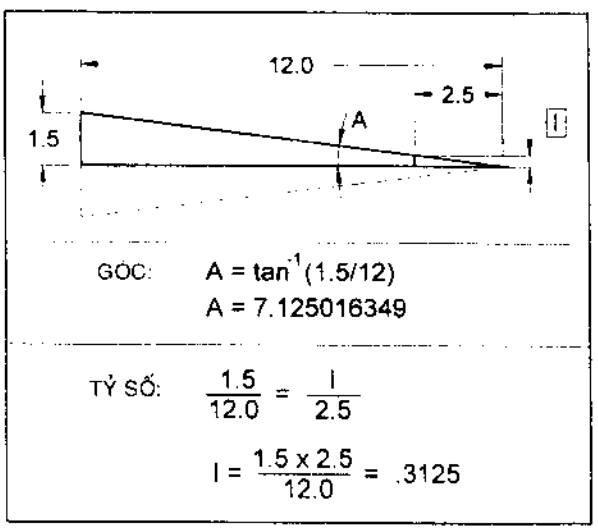
Cả hai giá trị trong tỷ số phải có cùng đơn vị và phải là phân số tối giản (1/4 thay vì 2/8 hoặc 4/16).

ĐỘ CÔN/FOOT biểu thị hiệu số giữa hai đường kính trên chiều dài một foot hoặc 12 inch

Ví dụ, độ côn 3 inch/foot tương đương tỷ số côn 1/4, do

3/12 = 1/4 = 1:4

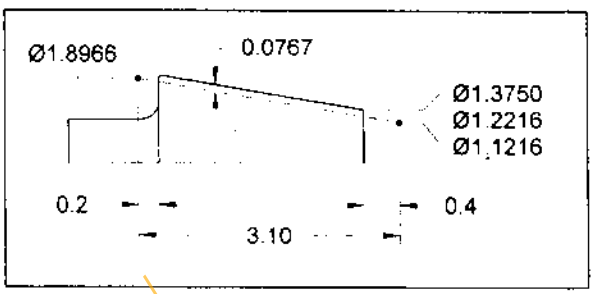
Trong lập trình CNC, bạn chỉ cần tính các đường kính ở *đầu* và *cuối* ren. Các tính toán có thể thực hiện theo hàm lượng giác, hoặc theo tỷ số (Hình 37.21).



Hình 37.21. Tính toán ren côn - không xét các khoảng hở.

Dựa trên các nguyên tắc đó, có thể tính các giá trị cần thiết cho ví dụ nêu trên. Chú ý, chỉ sử dụng các kích thước *theo cạnh* hoặc *hướng kính*. Trong nhiều ứng dụng lập trình, chỉ cần dùng một trong hai phương pháp nêu trên, góc *hoặc* tỷ số. Có thể dùng phương pháp còn lại để kiểm chứng độ chính xác của các tính toán.

Trên Hình 37.22, các đường kính đầu và cuối đã được tính, sử dụng phương pháp góc và/hoặc tỷ số giữa các cạnh. Kết quả tính toán sẽ được dùng trong chương trình tùy thuộc vào kỹ thuật lập trình, chẳng hạn phương pháp block hoặc phương pháp chu kỳ. Các chi tiết tùy thuộc vào đặc tính kỹ thuật của ren, các tính năng của máy và bộ điều khiển.



***Hình 37.22.** Các giá trị tính toán đối với ren côn trong chương trình O3708.*

Ren côn lập trình theo block

Lập trình gia công ren côn theo block cũng đơn giản như đối với ren thẳng. Để đơn giản, ví dụ này sẽ sử dụng ăn dao thẳng (một trục) và chín đường cắt ren, chiều sâu toàn phần là .0767 inch. Cần áp dụng chín chiều sâu cho *cả hai* đầu ren. Cột thứ nhất là chiều sâu của từng đường cắt, cột thứ hai là đường kính đầu (nhỏ) của ren, cột thứ ba là đường kính cuối (đầu lớn) của ren côn. Đường kính đầu (nhỏ) được tính theo tọa độ tuyệt đối Z0.4, đường kính cuối ở Z-2.7:

Chiều sâu	∅ đầu	∅ cuối
.0165	1.2420	2.0170
.0145	1.2130	1.9880
.0120	1.1890	1.9640
.0100	1.1690	1.9440
.0080	1.1530	1.9280
.0060	1.1410	1.9160
.0040	1.1330	1.9080
.0030	1.1270	1.9020
.0027	1.1216	1.8966

Mọi yêu cầu đều đã có đủ để viết chương trình O3708:

```
O3708
...
(G32 - TAPERED THREAD)
(N45 G50 X12.0 Z4.5)
N46 T0500 M42
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N47 G97 S450 M03
N48 G00 X2.5 Z0.4 T0505 M08
N49 X1.242                          (PASS 1)
N50 G32 X2.017 Z-2.7 F0.125
N51 G00 X2.5
N52 Z0.4
N53 X1.213                          (PASS 2)
N54 G32 X1.988 Z-2.7
N55 G00 X2.5
N56 Z0.4
N57 X1.189                          (PASS 3)
N58 G32 X1.964 Z-2.7
N59 G00 X2.5
N60 Z0.4
N61 X1.169                          (PASS 4)
N62 G32 X1.944 Z-2.7
N63 G00 X2.5
N64 Z0.4
N65 X1.153                          (PASS 5)
N66 G32 X1.928 Z-2.7
N67 G00 X2.5
N68 Z0.4
N69 X1.141                          (PASS 6)
N70 G32 X1.916 Z-2.7
N71 G00 X2.5
N72 Z0.4
N73 X1.133                          (PASS 7)
N74 G32 X1.908 Z-2.7
N75 G00 X2.5
N76 Z0.4
N77 X1.127                          (PASS 8)
N78 G32 X1.902 Z-2.7
N79 G00 X2.5
N80 Z0.4
N81 X1.1216                         (PASS 9)
N82 G32 X1.8966 Z-2.7
N83 G00 X2.5
N84 Z0.4
N85 G00 X12.0 Z4.5 T0500 M09
N86 M30
%
```

Trong ví dụ này, sự ăn dao thẳng và lùi dao được sử dụng để chương trình dễ hiểu hơn. Chương trình sẽ không quá thay đổi nếu sử dụng ăn dao kết hợp và/hoặc lùi dao theo góc (hai trục) ra khỏi ren, nhưng cần tính toán nhiều hơn.

Ren côn sử dụng chu kỳ đơn giản

Trong chu kỳ gia công ren G92, độ côn của ren được lập trình theo giá trị bán kính I, với chiều chuyên biệt từ đường kính cuối (đầu lớn) đến đường kính đầu (nhỏ):

G92 X.. Z.. I.. F..

X là đường kính ren hiện hành ở *cuối đường cắt*, Z là vị trí cuối của ren, I là hiệu số *theo một phía* giữa đường kính ren ở *cuối* và đường kính ren ở *đầu*. Giá trị I phải có dấu đại số (chỉ phải viết dấu trừ, nếu có), chuyên biệt *chiều* côn, trong trường hợp này là giá trị âm. Chương trình O3709 sẽ cắt ren côn sử dụng chu kỳ G92.

```
O3709
...
(G92 - TAPERED THREAD)
(N45 G50 X12.0 Z4.5)
(N46 T0505 M42)
N47 G97 S450 M03
N48 G00 X2.5 Z0.4 T0505 M08
N49 G92 X2.017 I-0.3875 Z-2.7 F0.125 (PASS 1)
N50 X1.988                          (PASS 2)
N51 X1.964                          (PASS 3)
N52 X1.944                          (PASS 4)
N53 X1.928                          (PASS 5)
N54 X1.916                          (PASS 6)
N55 X1.908                          (PASS 7)
N56 X1.902                          (PASS 8)
N57 X1.8966                         (PASS 9)
N58 G00 X12.0 Z4.5 T0500 M09
N59 M30
%
```

Chú ý, khoảng cách I của độ côn là hiệu số giữa đường kính *cuối*, 1.8966, và đường kính *đầu*, 1.1216, chia cho 2. Kết quả sẽ là:

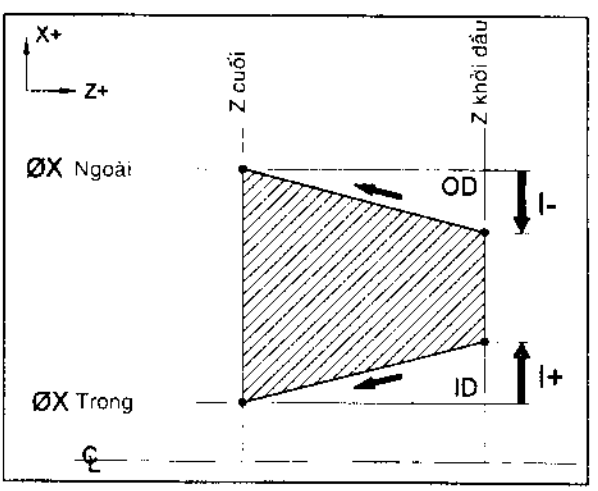
(1.8966 – 1.1216) / 2 = .3875

Giá trị I (.3875) phải có dấu, biểu thị chiều côn (chiều từ điểm cuối). Trong ví dụ này, giá trị I phải âm do đường kính đầu của đoạn côn *ở dưới* đường kính cuối của đoạn côn khi chiếu từ ụ động đến mâm cặp máy tiện. Trong chương trình, mục nhập này sẽ là I-0.3875.

Ren côn và chu kỳ lặp

Chu kỳ lặp G76 yêu cầu giá trị I *khác zero*, nếu gia công ren côn. Giá trị I trong chương trình xác định hiệu số theo một phía (bán kính), được gọi là khoảng cách bán kính, và chiều giữa đường kính đầu và cuối của ren côn.

Bạn cần nhớ, đường kính R luôn luôn lập trình ở *cuối* ren và giá trị I chiều cao và độ dốc của đoạn côn (tỷ số côn theo cạnh). Trên máy tiện CNC với chiều X+ hướng lên từ đường tâm, sự tăng đường kính côn sẽ yêu cầu giá trị I âm, sự giảm đường kính côn sẽ yêu cầu giá trị dương. I luôn luôn có giá trị đơn, đo trên bán kính thay vì đường kính (Hình 37.23).



Hình 37.23. *Chiều ren côn I trong chu kỳ gia công ren G76*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chu kỳ cơ bản G76 được duy trì nhưng có thêm giá trị I khác zero.

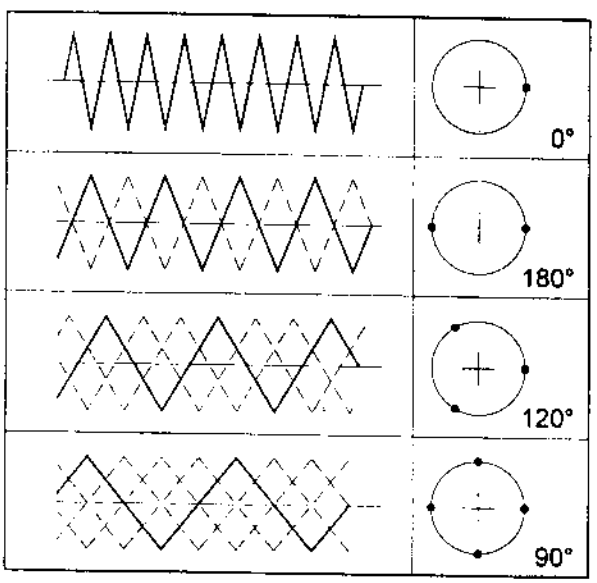
```
O3710
...
(G76 - TAPERED THREAD)
(N45 G50 X12.0 Z4.5)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X2.5 Z0.4 T0505 M08
N49 G76 X1.8966 Z-2.7 I-0.3875 K0.0767
    D0140 F0.125
N50 G00 X12.0 Z4.5 T0500 M09
N51 M30
%
```

Nếu có thể dùng phương pháp này để tiện ren, chu kỳ G76 là lựa chọn tốt nhất, có thể viết chương trình rất nhanh, và dễ dàng chỉnh sửa ngay tại máy tiện CNC.

REN NHIỀU ĐẦU MỐI

Hầu hết các ren đều chỉ có một đầu mối, thích hợp cho nhiều ứng dụng. Công dụng chính của ren nhiều đầu mối là truyền chuyển động *chính xác* rất nhanh qua khoảng cách tương đối dài. Bạn hãy lưu ý thuật ngữ *chính xác* – ren thô cũng có thể sử dụng để truyền chuyển động một cách nhanh chóng, nhưng không chính xác. Ví dụ về ren nhiều đầu mối chính xác là phần ren điều khiển thấu kính trong các camera.

Đối với nhà lập trình, có vài xem xét đặc thù đối với ren nhiều đầu mối. Điều quan trọng là vị trí khởi đầu đối với từng ren phải ở vị trí, khi nhìn từ mối ren của vít hoặc đai ốc, từng đầu mối trên chu vi sẽ được chia theo các số gia góc bằng nhau. Điều quan trọng thứ hai là biên dạng ren bằng nhau khi nhìn từ tiết diện ren.



Hình 37.24. *Biểu diễn ren nhiều đầu mối (điểm đen là đầu mối ren)*

Để đạt được các điều kiện đó, có hai công cụ lập trình khả dụng.

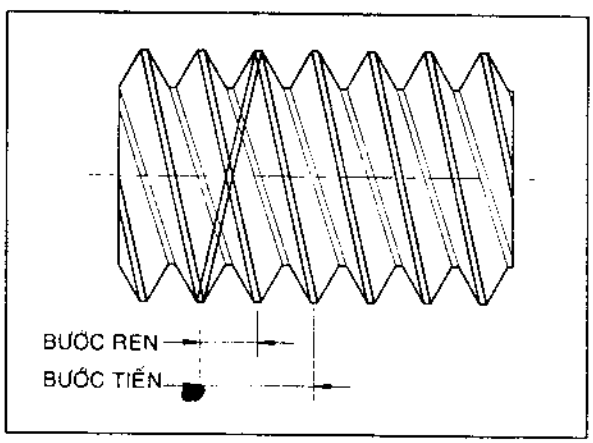
Các công cụ này là tính toán *vị trí đầu mối ren* và *tốc độ cắt ren* (Hình 37.24).

Trong minh họa này là bốn ví dụ tiết diện (trái) và hình chiếu tương ứng (phải) của ren một đầu mối (trên), hai đầu mối (thứ hai), ba đầu mối (thứ ba), và bốn đầu mối (dưới cùng).

Mặc dầu các ví dụ này chỉ có tính biểu tượng, nhưng chúng có *bước ren* như nhau. Ngoài ra, bạn cần lưu ý sự phân bố *đều* của các đầu mối ren, được biểu thị bằng điểm đen. Giá trị góc là khoảng cách giữa các đầu mối khi chiếu phần ren theo đường tâm. Khoảng cách này là tự động, trong chế độ gia công ren chỉ cần lập trình giá trị dịch chuyển chính xác từ một đầu mối ren đến đầu mối kế tiếp.

Tính tốc độ cắt ren

Tốc độ cắt ren luôn luôn là *bước tiến* của ren, không phải *bước* ren. Đối với ren một đầu mối, bước tiến và bước ren bằng nhau – nhưng khác nhau đối với ren nhiều đầu mối. Bạn hãy xét ren một đầu mối 16TPI. Bước tiến *và* bước ren đều bằng .0625, do đó tốc độ cắt là F0.0625. Nếu trên bản vẽ là ren 16TPI *hai đầu mối* (ví dụ 3.0–16TPI 2 START), *bước ren* không thay đổi (.0625), nhưng *bước tiến* sẽ gấp đôi, đến .1250. Do đó, tốc độ cắt lập trình của ren hai đầu mối với bước ren .0625 sẽ là F0.125. Sự nhân giá trị bước ren luôn luôn phụ thuộc vào *số đầu mối ren*. Điều đó có nghĩa là ren ba đầu mối có tốc độ cắt bằng ba lần bước ren, ren bốn đầu mối có tốc độ cắt bằng bốn nhân với bước ren... (Hình 37.25).



Hình 37.25. *Quan hệ giữa bước tiến và bước ren của ren hai đầu mối*

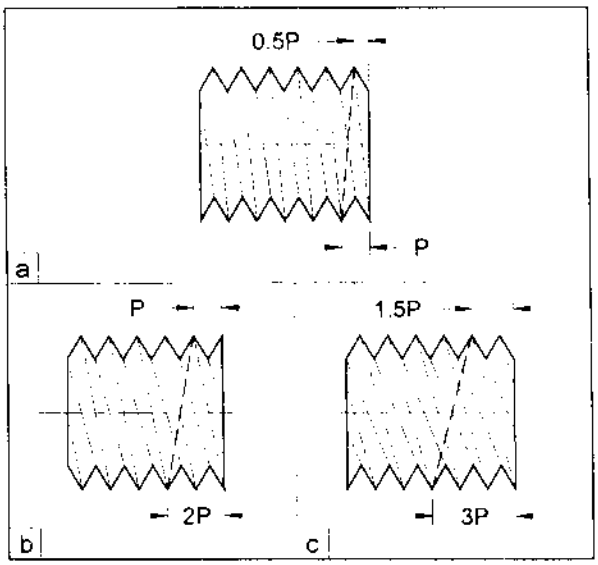
Tốc độ cắt được tính theo công thức:

$$\text{Tốc độ cắt} = \frac{\text{Số đầu mối ren}}{\text{TPI}}$$

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Hình 37.26 minh hoạ quan hệ giữa bước ren và bước tiến của một số ren nhiều đầu mối – quan hệ này được duy trì theo tỷ lệ.



Hình 37.26. *Ren nhiều đầu mối – quan hệ giữa bước tiến và bước ren.*
(a) Ren một đầu mối Bước tiến = Bước ren = 1P
(b) Ren hai đầu mối Bước tiến = 2P
(c) Ren ba đầu mối Bước tiến = 3P

Lượng dịch chuyển

Tốc độ cắt không phải là xem xét duy nhất khi lập trình ren nhiều đầu mối. Yếu tố thứ hai là giá trị lập trình của *độ dịch chuyển mũi dao*. Sự dịch chuyển này bảo đảm từng đầu mối ren sẽ trong quan hệ thích hợp với tất cả đầu mối ren còn lại. Khi hoàn tất một đầu mối, vị trí khởi đầu của dao cần *dịch chuyển* (chỉ theo trục Z), luôn luôn bằng giá trị *bước ren*. Công thức tính giá trị dịch chuyển mũi dao như sau:

Lượng dịch chuyển = Bước ren

Lượng dịch chuyển lập trình *đối với từng* đầu mối ren phía trên đầu mối thứ nhất. Điều đó có nghĩa là số lượng dịch chuyển trong chương trình bằng *số đầu mối ren trừ một*:

Số lần dịch chuyển = Số đầu mối ren – 1

Công thức này cũng đúng cho ren một đầu mối, nhưng số lần dịch chuyển là zero (1-1=0)

Vài phương pháp có thể xác định *thời điểm* lập trình sự dịch chuyển dao. Phương pháp thứ nhất, đối với ren hai đầu mối, là lập trình một ren đến chiều sâu toàn phần, sau đó dịch chuyển và cắt ren thứ hai đến chiều sâu toàn phần. Phương pháp thứ hai, cũng áp dụng cho loại ren này, là một đường cắt cho ren thứ nhất, *dịch chuyển ra*, áp dụng đường cắt đó cho ren thứ hai, *dịch chuyển vào*, cắt đường cắt thứ hai cho ren thứ nhất, *dịch chuyển ra* và lặp lại cho đến khi cả hai ren đạt được chiều sâu toàn phần. Phương pháp này có thể áp dụng cho ren có số đầu mối bất kỳ.

Ưu thế rõ ràng của phương pháp thứ nhất là dễ lập trình. Nhược điểm là, nếu lưỡi cắt bị mòn trên ren thứ nhất, ren thứ hai sẽ không chính xác. Ưu điểm của phương pháp thứ hai là dao sẽ mòn đều trên cả hai ren, nhưng đòi hỏi nhiều công sức hơn khi lập trình. Vấn đề tiếp theo là đối với nhiều vật liệu cứng, tuổi bền của lưỡi cắt có thể bị giảm.

Để minh họa ứng dụng ren nhiều đầu mối, sẽ sử dụng loại ren dưới đây:

- Số ren/inch là 12 (12 TPI).
- Số đầu mối ren là hai.
- Cắt ren ngoài trên đường kính danh định 3.000 inch.
- Chiều sâu ren tính toán .0511 (.61343/12).
- Số đường cắt là 7 (chu kỳ G92).

Phương pháp lập trình block sử dụng G32 có thể áp dụng cho một số trường hợp đặc biệt, nhưng cũng có thể đạt kết quả tốt khi sử dụng chu kỳ G92 và G76, chương trình sẽ ngắn hơn và biên tập tại máy CNC dễ dàng hơn.

Ví dụ ứng dụng

Ren được dùng trong ví dụ này là loại 12 TPI hai đầu mối trên đường kính Ø3.00, với số đường cắt là bảy, và các chiều sâu cắt như đã nêu ở phần trước, chương trình thứ nhất O3711 sẽ hoàn tất một ren *trước* ren thứ hai (tốc độ cắt F0.25, *thay vì* G0.125), P là số đường cắt, T là số ren, thứ nhất hoặc thứ hai.

```
O3711
...
(G92 - DOUBLE START TREAD - 1)
(N45 G50 X12.0 Z4.5)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X2.5 Z0.4 T0505 M08    (--- THREAD 1)
N49 G92 X2.017 Z-2.7 F0.25           (T1 - P1)
N50 X1.988                           (T1 - P2)
N51 X1.964                           (T1 - P3)
N52 X1.944                           (T1 - P4)
N53 X1.928                           (T1 - P5)
N54 X1.916                           (T1 - P6)
N55 X1.908                           (T1 - P7)
N56 X1.902                           (T1 - P8)
N57 X1.8966                          (T1 - P9)
N58 G00 X2.5 Z0.525            (--- THREAD 2)
N59 G92 X2.017 Z-2.7                 (T2 - P1)
N60 X1.988                           (T2 - P2)
N61 X1.964                           (T2 - P3)
N62 X1.944                           (T2 - P4)
N63 X1.928                           (T2 - P5)
N64 X1.916                           (T2 - P6)
N65 X1.908                           (T2 - P7)
N66 X1.902                           (T2 - P8)
N67 X1.8966                          (T2 - P9)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N68 G00 X12.0 Z4.5 T0520 M09
N69 M30
%
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Phiên bản này có thể được chỉnh sửa để lần lượt chuyển đổi đường cắt từ ren thứ nhất sang ren thứ hai, chương trình O3712. Áp dụng kỹ thuật này, sự mòn dao sẽ được phân bố đồng đều giữa tất cả các ren.

```
O3712
...
(G92 - DOUBLE START THREAD - 2)
(N45 G50 X12.0 Z4.5)
N46 T0500 M42
N47 G97 S450 M03
N48 G00 X2.5 Z0.4 T0505 M08     (--- THREAD 1)
N49 G92 X2.017 Z-2.7 F0.25           (T1 - P1)
N50 G00 Z0.525                       (START 1)
N51 G92 X2.017 Z-2.7                 (T2 - P1)
N52 G00 Z0.4                         (START 2)
N53 G92 X1.988 Z-2.7                 (T1 - P2)
N53 G00 Z0.525                       (START 1)
N54 G92 X1.988 Z-2.7                 (T2 - P2)
N55 G00 Z0.4                         (START 2)
N56 G92 X1.964 Z-2.7                 (T1 - P3)
N57 G00 Z0.525                       (START 1)
N58 G92 X1.964 Z-2.7                 (T2 - P3)
N59 G00 Z0.4                         (START 2)
N60 G92 X1.944 Z-2.7                 (T1 - P4)
N61 G00 Z0.525                       (START 1)
N62 G92 X1.944 Z-2.7                 (T2 - P4)
N63 G00 Z0.4                         (START 2)
N64 G92 X1.928 Z-2.7                 (T1 - P5)
N65 G00 Z0.525                       (START 1)
N66 G92 X1.928 Z-2.7                 (T2 - P5)
N67 G0D Z0.4                         (START 2)
N68 G92 X1.916 Z-2.7                 (T1 - P6)
N69 G00 Z0.525                       (START 1)
N70 G92 X1.916 Z-2.7                 (T2 - P6)
N71 G00 Z0.4                         (START 2)
N72 G92 X1.908 Z-2.7                 (T1 - P7)
N73 G00 Z0.525                       (START 1)
N74 G92 X1.908 Z-2.7                 (T2 - P7)
N75 G00 Z0.4                         (START 2)
N76 G92 X1.902 Z-2.7                 (T1 - P8)
N77 G00 Z0.525                       (START 1)
N78 G92 X1.902 Z-2.7                 (T2 - P8)
N79 G00 Z0.4                         (START 2)
N80 G92 X1.8966 Z-2.7                (T1 - P9)
N81 G00 Z0.525                       (START 1)
N82 G92 X1.8966 Z-2.7                (T2 - P9)
N83 G00 X12.0 Z4.5 T0500 M09
N84 M30
%
```

Chu kỳ G92 và chuyển động G00 xóa lẫn nhau. Đó là lý do để lặp lại mã G. Giá trị duy nhất còn giữ hiệu lực là tốc độ cắt F0.25, chỉ cần lập trình một lần trong ví dụ nêu trên.

CẮT LẠI REN

Sau khi gia công xong, cần kiểm tra chất lượng ren *trước khi* tháo chi tiết ra khỏi máy. Khi chi tiết được tháo ra khỏi máy, nếu cần cắt lại ren, sự gá lắp và định vị lại chi tiết sẽ rất khó bảo đảm độ chính xác. Đường cắt ren thứ nhất sẽ bắt đầu tại vị trí ngẫu nhiên trên chu vi trục. Từng đường cắt kế tiếp sẽ tự động đồng bộ hóa để bắt đầu ở cùng vị trí. Khi chi tiết ren vẫn được kẹp chặt, sự đồng bộ hóa này sẽ được bảo đảm.

Có hai phương pháp cho phép tránh cắt gọt lại ren. Thứ nhất, lập trình sự bù mòn dao. Thứ hai, lập trình hàm M00 ở cuối từng thao tác cắt ren, trước khi tiếp tục gia công, kể cả đối với dao cuối cùng. Nếu cần cắt lại ren sau khi tháo chi tiết, người vận hành nên áp dụng các bước dưới đây:

1. Định vị và kẹp chặt lại chi tiết ren để chạy đồng tâm giữa chi tiết và trục chính.
2. Xác lập bù trục X đủ lớn, sao cho dao tiện ren dịch chuyển phía trên ren (ren ngoài) hoặc phía dưới ren (ren trong).
3. Chỉnh thẳng hàng mũi dao tiện ren với ren đã hoàn tất (chỉ với độ chính xác của thị giác – mắt người vận hành).
4. Lặp lại các bước cắt ren đã thực hiện trước đó, nhưng lần này không cắt gọt, cẩn thận điều chỉnh giá trị bù mòn dao cho đến khi dao thực sự cắt lại ren.

Về nguyên tắc, cần tránh cắt gọt lại ren. Rất khó gá lắp và kẹp chặt lại chi tiết một cách chính xác, do đó rất khó bảo đảm chất lượng ren.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

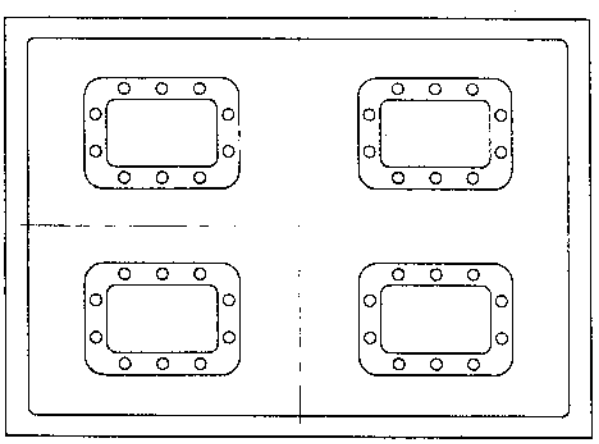
Nhiều tài liệu hay có tại : http://www.me-cad.com/

38 CHƯƠNG TRÌNH CON

Độ dài của chương trình CNC thường được đo theo số lượng ký tự trong chương trình. Số lượng này tương tự byte, nếu chương trình được lưu trên đĩa máy tính. Chiều dài thực của chương trình thường không gây ra vấn đề đối với sự gia công cụ thể. Chiều dài chương trình sẽ thay đổi, tùy theo độ phức tạp của chi tiết gia công, số dụng cụ cắt được sử dụng, phương pháp lập trình, và các yếu tố khác. Nói chung, chương trình càng ngắn, thời gian viết chương trình càng ngắn. Các chương trình ngắn cũng giảm khả năng xảy ra sai sót của con người, do dễ dàng kiểm tra, chỉnh sửa, và tối ưu hóa. Hầu như mọi hệ thống CNC đều có các tính năng được thiết kế để rút ngắn chương trình và làm cho quá trình lập trình trở nên dễ dàng hơn, hiệu quả hơn, và ít lỗi sai hơn. Các ví dụ của kiểu lập trình này là chu kỳ cố định, chu kỳ lặp và các macro riêng. Chương này trình bày cấu trúc, sự triển khai, và các ứng dụng của phương pháp chuẩn bị chương trình có tính hiệu quả cao - sử dụng *chương trình con*.

CHƯƠNG TRÌNH CHÍNH VÀ CÁC CHƯƠNG TRÌNH CON

Chương trình CNC là chuỗi các lệnh, gán cho các dụng cụ cắt và các nguyên công. Nếu chương trình có nhiều lệnh *lặp lại*, có thể thay đổi cấu trúc từ một chương trình dài thành hai hoặc nhiều chương trình riêng rẽ. Từng nhóm lệnh lặp lại chỉ cần viết một lần và được gọi khi có yêu cầu. Đó là khái niệm cơ bản của chương trình con. Hình 38.1 minh họa sơ đồ chi tiết lặp lại ở các vị trí khác nhau.



Hình 38.1. *Ví dụ về yêu cầu chi tiết thích hợp để sử dụng chương trình con*

Từng chương trình phải có chỉ số chương trình riêng và được lưu trong bộ nhớ của hệ điều khiển. Nhà lập trình sử dụng hàm mã M đặc biệt để gọi chương trình tương ứng. Chương trình *thứ nhất* gọi chương trình khác, được gọi là *chương trình chính*, mọi chương trình còn lại đều là *chương trình con*. Chương trình con không thể gọi chương trình chính, do đây là chương trình ở mức cao nhất của mọi chương trình còn lại. Chương trình con cũng có thể được gọi từ các chương trình con khác có mức cao hơn.Khi sử dụng chương trình có các chương trình con, bạn phải chọn chương trình chính, không chọn chương trình con. Khả năng duy nhất để chọn chương trình con trên bộ điều khiển là nhằm các mục đích biên tập. Trong một số tài liệu, chương trình con (subprogram) còn được gọi là *subroutine* hoặc *macros*, nhưng thông dụng nhất vẫn là thuật ngữ chương trình con, và macros đôi khi có thể có ý nghĩa hơi khác.

Công dụng của chương trình con

Thứ tự lập trình lặp lại các nhóm lệnh hoặc các chuỗi thứ tự block thay đổi sẽ rất có ích nếu được chuyển thành chương trình con. Các ứng dụng phổ biến của chương trình con trong lập trình CNC bao gồm:

- Các chuyển động gia công lặp lại.
- Các hàm liên quan đến sự thay dao
- Các sơ đồ gia công lỗ
- Gia công các rãnh và các ren
- Các bước làm nóng máy
- Thay đổi giá đỡ định vị
- Các hàm đặc biệt
- ...

Về cấu trúc, chương trình con hoàn toàn tương tự chương trình tiêu chuẩn. Chúng sử dụng cùng các nguyên tắc cú pháp. Nói chung, có thể khó phân biệt giữa chương trình bình thường và chương trình con nếu không xem xét kỹ. Chương trình con có thể sử dụng dữ liệu nhập theo chế độ số gia hoặc chế độ tuyệt đối, tùy theo yêu cầu. Chương trình con được tải vào bộ nhớ hệ thống CNC như mọi chương trình khác. Khi được thực thi hợp lý, chúng có các tác dụng sau:

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Giảm chiều dài chương trình.
- Giảm lỗi sai trong chương trình.
- Giảm thời gian và công sức lập trình.
- Chỉnh sửa dễ dàng và nhanh chóng.

Không phải mọi chương trình con đều có đủ các tác dụng nêu trên, nhưng dù chỉ một tác dụng cũng đủ để sử dụng chương trình con.

Nhận biết chương trình con

Bước thứ nhất để sử dụng thành công chương trình con là *nhận biết và tách biệt* các chuỗi chương trình *lặp lại*. Ví dụ, sáu block chương trình biểu thị sự trở về zero máy (chuẩn máy) trên trung tâm gia công ngang, ở đầu chương trình.

```
N1 G20
N2 G17 G40 G80 (STATUS BLOCK)
N3 G91 G28 Z0 (Z AXIS RETURN)
N4 G28 X0 Y0 (X AND Y AXES RETURN)
N5 G28 B0 (B AXIS RETURN)
N6 G90 (ABSOLUTE MODE)
N7 ...
```

Những block này là chuỗi thứ tự các lệnh lặp lại *mỗi khi* viết chương trình mới cho máy đó. Loại chương trình này có thể được viết nhiều lần trong tuần, lặp lại cùng một chuỗi thứ tự các lệnh. Để giảm khả năng xảy ra lỗi, thứ tự các block thường dùng có thể lưu lại theo chương trình riêng và được ký hiệu bằng chỉ số chương trình đặc thù. Sau đó có thể gọi lại ở phần đầu của chương trình chính bất kỳ. Thứ tự lập trình này được gọi là *chương trình con* - nhánh hoặc phần phụ của chương trình chính.

CÁC HÀM CHƯƠNG TRÌNH CON

Chương trình con phải được hệ điều khiển nhận biết theo kiểu chương trình *đặc thù, không phải là chương trình chính*. Đặc tính này được thực hiện với hai hàm M, thường chỉ áp dụng cho chương trình con:

M98	Hàm gọi chương trình con
M99	Hàm kết thúc chương trình con

Tiếp sau *hàm M98 gọi* chương trình con luôn luôn là chỉ số chương trình con P--. *Hàm M99 kết thúc* chương trình con sẽ chấm dứt chương trình đó và chuyển sự xử lý cho chương trình gốc chứa chương trình con đó (chương trình chính hoặc chương trình con có mức cao hơn). Tuy M99 chủ yếu để kết thúc chương trình con, nhưng đôi khi cũng được sử dụng trong chương trình chính thay cho hàm M30. Trong trường hợp đó, chương trình sẽ "lặp lại" liên tục, cho đến khi nhấn phím *Reset*.

Hàm gọi chương trình con

Hàm M98 gọi chương trình con đã lưu trước đó từ chương trình khác. Nếu chỉ sử dụng một mình hàm này trong block, sẽ dẫn đến lỗi sai. M98 là hàm chưa hoàn chỉnh, đòi hỏi *thêm* hai tham số để có hiệu lực thực sự:

- Địa chỉ P nhận biết chỉ số chương trình con được chọn.
- Địa chỉ L hoặc K xác định số lần lặp lại chương trình con (mặc định là K1 hoặc L1).

Ví dụ, block gọi chương trình con có hàm M98 và chỉ số chương trình con:

```
M167 M98 P3851
```

Trong block N167, chương trình con O3851 được gọi từ bộ nhớ CNC, lặp lại *một lần* – do L1 (K1) là mặc định, tùy theo hệ điều khiển. Chương trình con này phải được lưu trong bộ điều khiển, trước khi gọi từ chương trình khác.

Block M98, gọi chương trình con, còn có thể chứa các lệnh khác, chẳng hạn chuyển động chạy dao nhanh, tốc độ trục chính, tốc độ cắt, bù bán kính dao, v.v... Trên hầu hết các bộ điều khiển, nếu được gộp trong block gọi chương trình con, dữ liệu đó sẽ chuyển nội dung cho chương trình con. Block gọi chương trình con dưới đây còn có chuyển động dao theo hai trục:

```
N460 G00 X28.373 Y13.4193 M98 P3851
```

Block này trước hết sẽ thực thi chuyển động chạy dao nhanh, *sau đó* gọi chương trình con. Thứ tự các lệnh trong block không gây ra sự khác biệt đối với sự thực thi block:

```
N460 M98 P3851 G00 X28.373 Y13.4193
```

sẽ có cùng thứ tự gia công cũng như khi lệnh chuyển động chạy dao đứng trước lệnh gọi chương trình con, nhưng có vẻ không logic.

Hàm kết thúc chương trình con

Khi cả chương trình chính và chương trình con cùng hiện hữu trong bộ điều khiển, chúng phải có chỉ số chương trình khác nhau. Trong khi xử lý, chúng sẽ hoạt động như một chương trình liên tục, do đó phải có sự phân biệt rõ ràng đối với hàm kết thúc chương trình. Hàm kết thúc chương trình là M90, hoặc M02, nhưng ít thông dụng. Chương trình con phải được kết thúc bằng hàm khác. Fanuc sử dụng M99 cho mục đích này:

```
O3851 (SUB-1)          Khởi động chương trình con
...
...
M99                    Kết thúc chương trình con
%
```

Khi kết thúc chương trình con, bộ điều

Nhiều tài liệu hay có tại : http://www.me-cad.com/

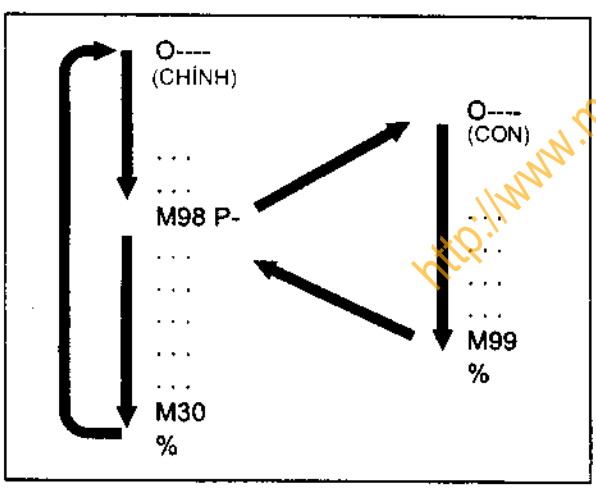
Nhiều tài liệu hay có tại : http://www.me-cad.com/

khiển quay lại xử lý chương trình gốc – sẽ *không* kết thúc chương trình chính – chỉ kết thúc với hàm M30. Có thể gộp thêm vài tham số vào M99 kết thúc chương trình con, chẳng hạn mã bỏ qua block, số block trở về để thoát. Chú ý, ký hiệu mã dừng (dấu %) cũng được dùng cho chương trình con như chương trình chính. Sự kết thúc chương trình con là rất quan trọng và phải thực hiện chuẩn xác. Lệnh này gởi hai yêu cầu rất quan trọng cho bộ điều khiển:

- Kết thúc chương trình con.
- Trở về block sau block gọi chương trình con.

Không được phép dùng hàm kết thúc chương trình M30 (hoặc M02) để kết thúc chương trình con – lệnh này sẽ *xóa toàn bộ sự xử lý chương trình* và xác lập lại bộ điều khiển. Hàm kết thúc chương trình này không cho phép thực thi block bất kỳ đứng sau block chứa lệnh đó.

Nói chung, hàm kết thúc chương trình con M99 đưa sự xử lý trở lại block *ngay sau* block chứa lệnh gọi chương trình con M98. Khái niệm này được minh họa trên Hình 38.2 (không ghi số block).



Hình 38.2. Lưu đồ xử lý chương trình với một chương trình con.

Số block trở về

Trong hầu hết các chương trình, hàm M99 được lập trình là mục nhập độc lập và là lệnh *cuối cùng* trong chương trình con. Nói chung, thường không có các lệnh khác trong block này. Hàm M99 kết thúc chương trình con và chuyển sự thực thi sang block kế tiếp của chương trình chứa chương trình con đó. Ví dụ,

```
N67 M98 P3852                    (SUBPROGRAM CALL)
N68 ...          (BLOCK TO RETURN TO FROM O3852)
N69 ...
N70 ...
```

thực thi block N67 bằng cách gọi chương trình con O3852. Sau khi xử lý chương trình con O3852, bộ điều khiển trở lại chương trình gốc và tiếp tục xử lý các lệnh từ block N68, là block trở về.

Ứng dụng đặc biệt

Đối với một số ứng dụng đặc biệt, có thể phải xác định chỉ số block *khác* để trở về, thay vì sử dụng mặc định *block kế tiếp*. Nếu nhà lập trình nhận thấy tùy chọn này là hữu ích cho một số công việc cụ thể và sử dụng kỹ thuật đó, địa chỉ P gộp vào block N99:

```
M99 P..
```

Trong định dạng này, địa chỉ P biểu thị chỉ số block trở về – *từ* chương trình con đã hoàn tất. Chỉ số block đó phải có trong chương trình gốc. Ví dụng, nếu chương trình chính có các block:

```
(CHƯƠNG TRÌNH CHÍNH)
...
N67 M98 P3852
N68 ...
N69 ...
N70 ...
```

và chương trình con O3852 kết thúc như sau:

```
O3852 (SUB)
...
...
M99 P70
%
```

sự xử lý chương trình gọi sẽ tiếp tục từ block N70 (của chương trình chính trong ví dụ nêu trên), *bỏ qua* các block N68 và N69. Dạng ứng dụng này, nói chung, ít được dùng trong thực tế, do đòi hỏi kiểu công việc thích hợp, được nêu ở đây để hiểu sâu hơn các nguyên tắc sử dụng chương trình con.

Địa chỉ P có ý nghĩa khác khi được dùng với hàm M98 và M99

Nói chung, kỹ thuật lập trình này ít được sử dụng trong thực tiễn, nhưng nhà lập trình cần nắm vững. Các ứng dụng liên quan sẽ gồm các công cụ lập trình khác, chẳng hạn kết hợp với hàm bỏ quan block, sử dụng mã gạch nghiêng (/).

Số lần lặp lại chương trình con

Tính năng gọi chương trình con rất quan trọng là địa chỉ L hoặc K, tùy theo kiểu loại bộ điều khiển. Địa chỉ này xác định số lần lặp lại chương trình con – *số lần* chương trình con được lặp lại trước khi quay trở lại xử lý chương trình gốc. Trong hầu hết các chương trình, chương trình con chỉ được gọi một lần, sau đó sẽ tiếp tục xử lý chương trình gốc.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Các chương trình con hỏi lặp lại trong chương trình con *nhiều lần* trước khi chuyển sang phần còn lại của chương trình gốc, là khá phổ biến. Để so sánh, sử dụng chương trình con O3852 một lần, có thể gọi từ chương trình gốc như sau:

```
N167 M98 P3852 L1 (K1)
```

Đây là block chương trình đúng, nhưng hoàn toàn không cần lập trình L1/K1. Có thể bỏ qua địa chỉ đó một cách an toàn, mặc định của bộ điều khiển là xử lý chương trình con *một lần*.

Nếu không chuyên biệt L/K, giá trị mặc định luôn luôn là L1/K1

```
N167 M98 P3852 L1 (K1) đồng nhất với
N167 M98 P3852
```

Chú ý – Trong các ví dụ kế tiếp, bạn hãy thay L bằng K, nếu bộ điều khiển yêu cầu.

Số lần lặp lại đối với một số kiểu bộ điều khiển trong khoảng L0 – L9999, và phải lập trình khi địa chỉ L khác với L1. Một số nhà lập trình viết block đầy đủ, kể cả khi chương trình con chỉ thực thi một lần, thay vì dựa vào điều kiện mặc định của bộ điều khiển. Lựa chọn này chỉ là sở thích cá nhân.

Biến thể địa chỉ số lần lặp lại

Một số bộ điều khiển Fanuc *không* chấp nhận địa chỉ L/K là số lần lặp lại, chúng sử dụng định dạng khác. Trên các bộ điều khiển này, lệnh gọi chương trình con một lần là không đổi:

```
N342 M98 P3852
```

Block này chỉ gọi chương trình con một lần, không sử dụng yêu cầu đặc biệt. Để lặp lại chương trình con, chẳng hạn 4 lần, thay vì lập trình.

```
N342 M98 P3852 L4 (K4)
```

sẽ viết số lần lặp lại cần thiết *trực tiếp ngay sau địa chỉ P*, trong một câu lệnh:

```
N342 M98 P43852          hoàn toàn tương tự với
N342 M98 P00043842
```

Kết quả là đồng nhất, chương trình con sẽ lặp lại 4 lần. *Bốn chữ số đầu được dùng cho số lần lặp lại, bốn chữ số cuối là chỉ số chương trình.* Ví dụ

```
M98 P3850                hoàn toàn tương tự với
M98 P00013850
```

Kết quả là chương trình con O3850 chỉ thực thi một lần. Để lặp lại chương trình O0050, ví dụ 39 lần, bạn cần viết:

```
M98 P380050  hoặc  M98 P00380050
```

Đối với các bộ điều khiển 0T/16T/18T/20T/21T, số lần lặp tối đa được biểu thị bằng bốn chữ số đầu trong địa chỉ P là 9999.

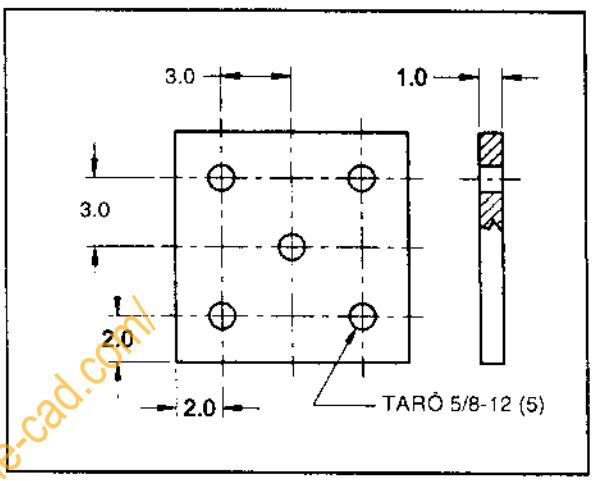
```
M98 P99993852
```

Sẽ lặp lại sự thực thi chương trình O3852 là 9999 lần (một số bộ điều khiển cũ có số lần lặp lại tối đa là 999 lần).

L0/K0 trong lệnh gọi chương trình con

Nói chung, bạn dễ dàng sử dụng L/K lớn hơn 1 để lặp lại chương trình con. Fanuc còn cung cấp số lần lặp lại *zero*, dạng K0/L0. Khi nào thì có thể lập trình L0/K0? Có ai đó muốn lặp lại chương trình con *zero lần*?

Có một số lý do. Bạn hãy xem Hình 38.3. Sơ đồ 5 lỗ cần được khoan miệng, và tarô ren.



Hình 38.3. Bản vẽ mẫu dùng để phát triển chương trình con, được dùng cho các chương trình O3801, O3802, và O3853

Đối với mũi khoan miệng (∅0.750) sẽ dùng chu kỳ G82 với khoảng tạm dừng 0.2 giây ở chiều sâu Z-0.3275. Đối với khoan để tarô ren, sử dụng chu kỳ G81, và tarô ren, với mũi tarô 5/8–12, sẽ dùng chu kỳ G84. Mũi khoan miệng chuẩn bị lỗ để khoan và tạo ra sự lã miệng 0.015 inch. Mũi khoan để tarô ren là loại 35/64 (∅0.5469), tạo lỗ cho mũi tarô 5/8-12:

```
O3801
(TOOL 1 - 90-DEG SPOT DRILL - 3/4 DIA)
N1 G20
N2 G17 G40 G80 T01
N3 M06
N4 G90 G00 G54 X2.0 Y2.0 S900 M03 T02
N5 G43 H01 Z1.0 M08
N6 G99 G82 R0.1 Z-0.3275 P200 F3.0  (LL HOLE)
N7 X8.0                             (LR HOLE)
N8 Y8.0                             (UR HOLE)
N9 X2.0                             (UL HOLE)
N10 X5.0 Y5.0                    (MIDDLE HOLE)
N11 G80 Z1.0 M09
N12 G28 Z1.0 M05
N13 M01

(TOOL 2 - 35/64 DRILL)
N14 T02
N15 M06
N16 G90 G00 G54 X2.0 Y2.0 S840 M03 T03
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N17 G43 H02 Z1.0 M08
N18 G99 G81 R0.1 Z-1.214 F11.0
N19 X8.0
N20 Y8.0
N21 X2.0
N22 X5.0 Y5.0
N23 G80 Z1.0 M09
N24 G28 Z1.0 M05
N25 M01

(TOOL 3 - 5/8-12 TAP)
N26 T03
N27 M06
N28 G90 G00 G54 X2.0 Y2.0 S500 M03 T01
N29 G43 H03 Z1.0 M08
N30 G99 G84 R0.4 Z-1.4 F41.0
N31 X8.0
N32 Y8.0
N33 X2.0
N34 X5.0 Y5.0
N35 G80 Z1.0 M09
N36 G28 Z1.0 M05
N37 G28 X5.0 Y5.0
N38 M30
%
```

Kiểu chương trình này sử dụng các tọa độ XY lặp lại cho từng dụng cụ cắt (mũi khoan lỗ miệng, khoan lỗ, và tarô ren). Để chương trình hiệu quả hơn, tất cả các block *lặp lại* trong chương trình sẽ được tập hợp vào chương trình con. Dưới đây là sơ đồ các lỗ được tách ra từ chương trình dài, có thêm G80Z1.0M09, là phần kết thúc tiêu chuẩn của chu kỳ cố định bất kỳ:

```
X2.0 Y2.0
X8.0
Y8.0
X2.0
X5.0
Y5.0
G80 Z1.0 M09
```

Chỉ cần nỗ lực đôi chút để tái định dạng chương trình hiện hữu, tách chương trình chính và chương trình con, lưu sơ đồ gia công lặp lại. Các tọa độ XY riêng của cả năm lỗ trong sơ đồ gộp lại:

```
O3853 (SUBPROGRAM)
(FIVE HOLE PATTERN)
N1 X2.0 Y2.0
N2 X8.0
N3 Y8.0
N4 X2.0
N5 X5.0 Y5.0
N6 G80 Z1.0 M09
N7 M99
%
```

Chương trình con này được gọi từ chương trình, trong ví dụ sẽ là từ chương trình mới O3802. Địa chỉ L0 sẽ ngăn chặn sự gia công hai lần cho lỗ thứ nhất:

```
O3802 (MAIN PROGRAM)
(TOOL 1 - 90-DEG SPOT DRILL - 3/4 DIA)
N1 G20
N2 G17 G40 G80 T01
N3 M06
N4 G90 G00 G54 X2.0 Y2.0 S900 M03 T02
N5 G43 H01 Z1.0 M08
N6 G99 G82 R0.1 Z-0.3275 P200 F3.0 L0
N7 M98 P3853
N8 G28 Z1.0 M05
N9 M01

(TOOL 2 - 35/64 DRILL)
N10 M06
N11 T02
N12 G90 G00 G54 X2.0 Y2.0 S840 M03 T03
N13 G43 H02 Z1.0 M08
N14 G99 G81 R0.1 Z-1.214 F11.0 L0
N15 M98 P3853
N16 G28 Z1.0 M05
N17 M01

(TOOL 3 - 5/8-12 TAP)
N18 M06
N19 T03
N20 G90 G00 G54 X2.0 Y2.0 S500 M03 T01
N21 G43 H03 Z1.0 M08
N22 G99 G84 R0.4 Z-1.4 F41.0 L0
N23 M98 P3853
N24 G28 Z1.0 M05
N25 G28 X5.0 Y5.0
N26 M30
%
```

Trong chương trình này, chuyển động dao XY ban đầu cho từng dao sẽ định vị dụng cụ cắt ở lỗ *thứ nhất* của sơ đồ gia công. Tất cả các chu kỳ cố định trong chương trình đều khởi động từ lỗ thứ nhất trong sơ đồ. Do định nghĩa lỗ thứ nhất được gộp trong chương trình con, *và trong chương trình chính*, bạn phải lập trình L0 trong block gọi chu kỳ cố định, nếu không dùng L0, lỗ thứ nhất sẽ được gia công hai lần. Đây là ứng dụng điển hình của L0 liên quan đến các chu kỳ cố định, nhưng không liên quan với chương trình con. Ngoài ra, block G82Z1.0M05 trả về zero máy tiêu chuẩn có thể gộp trong chương trình con O3853, do block này lặp lại sau từng lần gọi M98 trong chương trình chính O3802. Điều này tuy đúng nhưng không nên lạm dụng.

ĐÁNH SỐ CHƯƠNG TRÌNH CON

Sự theo dõi thường xuyên chương trình con luôn luôn quan trọng hơn sự theo dõi chương trình bình thường. Bạn cần biết *chính xác* các chương trình con khả dụng, phương pháp sử dụng, và công dụng của chúng. Một chương trình con có thể được sử dụng trong nhiều chương trình khác, do đó kỹ thuật nhận biết chương trình con là rất quan trọng.

Thư mục chương trình của bộ điều khiển không phân biệt giữa chỉ số chương trình và chỉ số chương trình con. Hệ điều khiển chỉ nhận biết lệnh gọi chương trình con theo định dạng lập trình, hàm M98, tiếp theo là chỉ số chương trình con P..

Toàn bộ điều này có nghĩa là chỉ số chương trình con được gán *ở mức lập trình*, không theo

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

mức có thể *bằng nhau*. Thách thức của nhà lập trình, *không phải* của người vận hành CNC, là gán chỉ số chương trình con. Nhà lập trình có tính linh hoạt cao trong sắp xếp chương trình con và ký hiệu của chúng, có thể thiết kế và thiết lập một số nguyên tắc cơ bản và các tiêu chuẩn liên quan. Một số nguyên tắc về định dạng chương trình chính cũng áp dụng cho chương trình con. Bạn hãy nhớ bốn điểm chính:

- Nếu được dùng trong chương trình, chỉ số chương trình thường được chuyên biệt bằng chữ O, tiếp theo là bốn hoặc năm chữ số, tùy theo hệ điều khiển.
- Nếu được dùng trong chương trình, chỉ số chương trình có thể chuyên biệt theo dấu hai chấm (:) đối với định dạng ISO, tiếp theo là bốn hoặc năm chữ số, tùy theo xác lập của hệ điều khiển.
- Chỉ số chương trình chính (O hoặc:) không thể âm hoặc bằng zero.
- Chỉ số chương trình con không thể âm hoặc bằng zero.

Trong khoảng cho phép, số *bất kỳ* có thể gán cho chương trình chính hoặc chương trình con *bất kỳ*. Một số nhà lập trình hoàn toàn không sử dụng chỉ số chương trình. Điều này có thể chấp nhận trên một số bộ điều khiển, nhưng chỉ khi ứng dụng đó không yêu cầu chương trình con. Trong hầu hết các trường hợp, người vận hành máy có thể gán chỉ số cho chương trình chính. Mặt khác, để kiểm soát các chương trình con, chỉ số chương trình trở nên rất quan trọng. Thứ nhất là để có sự tổ chức hợp lý. Điều này càng quan trọng nếu các chương trình con được thiết kế cho *nhiều chương trình* và được gọi *vào các thời điểm khác nhau*. Thực tế, không có phương pháp nào là *tốt nhất*, nhưng một số gợi ý có thể cung cấp ý tưởng để tiếp cận vấn đề đánh số chương trình và phát triển phong cách riêng.

Ví dụ, tất cả các chương trình chính đều được đánh số theo thứ tự liên tiếp, chẳng hạn trong cuốn sách này, hai chữ số đầu là số chương. Trong chương này, cũng áp dụng phương pháp đó cho các chương trình con, còn hai chữ số cuối là tùy ý tăng theo giá trị 50, ví dụ O3853 sẽ là ví dụ chương trình con thứ ba trong Chương 38. Bạn có thể chọn phương pháp riêng của mình, bảo đảm tính hợp lý và có tổ chức.

Phương pháp có tổ chức

Sự tiếp cận lập trình này dựa trên giả thiết là bộ nhớ CNC *không* được dùng làm phương tiện lưu trữ mọi chương trình gia công đã thực hiện. Dung lượng nhớ của hệ điều khiển *luôn luôn* có giới hạn. Khi đạt đến giới hạn đó, sẽ không còn chỗ cho các chương trình mới. Sự tổ chức chương trình hợp lý có lẽ là chỉ sử dụng bộ nhớ hệ thống CNC cho *chương trình hiện hành* và một vài chương trình sắp sử dụng.

Nếu *người vận hành máy công cụ* gán chỉ số chương trình đặc thù trong khi xác lập máy, tình huống này cần có sự kiểm soát. Trên một số bộ điều khiển, chỉ số chương trình chính trên bản copy thường không được tải lên một cách tự động, khi không thực sự cần thiết. Điều đó có nghĩa là, nếu sự sắp xếp được thực hiện với quản đốc phân xưởng, người vận hành CNC lưu các chương trình chính chỉ sử dụng *ba* chữ số cho các chỉ số chương trình bình thường, 1-999, sẽ có các số bốn chữ số 1000-9999 khả dụng cho các chương trình con. Khoảng khả dụng này là quá đủ cho hầu hết các ứng dụng sản xuất. Phương pháp này cho phép dễ dàng kiểm soát các chương trình con có chỉ số đã chọn. Mọi chỉ số bốn chữ số của chương trình con đều có thể được ghi lại, đánh giá, và gọi từ chương trình bất kỳ, chương trình chính hoặc chương trình con khác, mà không sợ trùng lắp hoặc nhầm lẫn.

Các chương trình con phải được ghi lại đầy đủ và chi tiết, độc lập với mọi chương trình gốc. Theo cách đó, các chương trình con có thể được sử dụng khi cần thiết, bất kể chương trình gốc. Phương pháp này cho phép sắp xếp mọi chương trình con theo số series (1000, 2000, ..., hoặc 1100, 1200, ...) cho từng kiểu máy, kiểu chương trình con, hoặc kiểu nguyên công cắt gọt

Các chương trình con riêng rẽ phải được gán chỉ số chương trình đặc thù. Chỉ số này được gọi cùng với hàm M98 và địa chỉ P. Sự kết hợp M98 và P.. là *yêu cầu tối thiểu* để gọi chương trình con từ chương trình khác.

Sử dụng ví dụ về chuỗi thứ tự trở về zero máy trên trung tâm gia công đứng bốn – trục, có thể viết chương trình con (với chỉ số được gán là O3854) đối với các block biểu thị mọi lệnh cần thiết – *không* gộp các lệnh G20, G21 chọn đơn vị.

```
O3854 (MACHINE ZERO RETURN)
N101 G17 G40 G49 G80
N102 G91 G28 Z0
N103 G28 X0 Y0
N104 G28 B0
N105 G90
N106 M99
%
```

Sự chọn đơn vị được đặt trong chương trình chính, để tăng tính linh hoạt. Sau khi viết và lưu chương trình con trở về zero máy trong bộ nhớ, từng chương trình chính đều có thể khởi đầu bằng cách gọi chương trình con O3854:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
O3803 (MAIN PROGRAM)
(PART ABC 123)
N1 G2 0                          Units used for this program
N2 M9 8 P3854                    Subprogram O3854 call
N3 G90 G54 G00 X. . Y. .         Nomal program start
N4 . . .
. . .
< ... Machining ... >
. . .
N45 M3 0                         Main program end
%
```

Để hình dung sự thực thi hai chương trình này từ hệ thống CNC, bạn hãy theo dõi mọi bước gia công theo thứ tự thực hiện chương trình. Trong khi thực thi chương trình O3803, hệ điều khiển sẽ tuân theo thứ tự các thao tác dưới đây:

1. Xác lập chỉ số chương trình O3803 là chỉ số chương trình hiện hành.
2. Hiển thị tên chương trình trên màn hình.
3. Xác lập đơn vị đo (trong ví dụ này là inch).
4. Chuyển sang đầu chương trình con O3854.
5. Thực thi mọi block trong chương trình con O3854.
6. Khi xử lý M99, chương trình con kết thúc, trở về chương trình chính.
7. Bắt đầu xử lý chương trình chính từ block N3
8. Khi xử lý M30, kết thúc chương trình chính, và trở về phần đầu.
9. Khi kích hoạt công tắc CYCLE START, sẽ lặp lại các bước từ 1 đến 8.

Như ví dụ này cho thấy, chương trình chính sử dụng số gia 1, chương trình con cũng sử dụng số gia 1, nhưng bắt đầu từ số block N101. Lý do thứ nhất là chương trình con được thiết kế thích hợp sẽ hầu như không thay đổi, do đó không cần bổ sung thêm các block sau khi gỡ rối chương trình đó. Lý do thứ hai còn quan trọng hơn. Sự không có các số chuỗi thứ tự trùng lặp sẽ thấy rõ trên màn hình của bộ điều khiển. Sự hiển thị các số block hoạt động sẽ nhanh chóng thông báo cho người vận hành CNC chương trình chính hay chương trình con đang thực thi. Các bộ điều khiển Fanuc rất linh hoạt với các số block, cho phép nhận biết các chuỗi thứ tự block một cách tự do, trong khoảng cho trước.

Ví dụ dưới đây sẽ minh họa khái niệm nêu trên. Trong ứng dụng đơn giản, chương trình chính gọi một chương trình con, sẽ không xảy ra vấn đề đánh số block. Ngay cả khi các số thứ tự trùng lặp trong cả hai chương trình, cũng không có sự nhầm lẫn. Mặc khác, khi nhiều chương trình con được gọi từ một chương trình chính, các số block trùng lặp xuất hiện trong khi xử lý chương trình chính, và cả khi xử lý các chương trình con. Tình huống đó có thể làm cho người vận hành CNC bị nhầm lẫn thậm chí không thể theo dõi điều thực sự đang diễn ra trong hệ điều khiển vào thời điểm bất kỳ đã cho.

Để tránh vấn đề đó, bạn hãy xét quá trình gán *các số block đặc thù* cho từng chương trình con, để không bị trùng lặp. Một phương pháp là xác định các số chương trình con theo các dãy hàng ngàn đủ cao, ví dụ O6100, O6200, O6310, ... Sau đó, đánh số block trong chương trình con dựa trên số chương trình con. Ví dụ:

```
O6100 (SUB 1)
N6101 . . .
N6102 . . .
N6103 . . .                      ...etc

O6200 (SUB 2)
N6201 . . .
N6202 . . .
N6203 . . .                      ...etc
```

Phương pháp này chỉ cho phép tối đa một trăm block, thích hợp cho nhiều chương trình con. Người vận hành sẽ dễ dàng giám sát chương trình với nhiều chương trình con. Đây tuy không phải là phương pháp có thể áp dụng cho mọi chương trình, nhưng ý tưởng tốt cho nhiều ứng dụng.

Bảo vệ chương trình con

Các chương trình con là chương trình đặc biệt được thiết kế để sử dụng thường xuyên. Các chương trình con có thể được lưu lâu dài trong bộ nhớ hệ thống, được gọi ra từ nhiều chương trình khác. Sự can thiệp bất kỳ vào các chương trình này, vô tình hay cố ý, đều có thể gây ra các hậu quả xấu. Nếu chỉ mất một chương trình con trong bộ nhớ, vẫn có thể làm dừng hàng trăm chương trình khác, tùy theo mức độ sử dụng chương trình con bị mất đó.

Các bộ điều khiển Fanuc hướng đến vấn đề này bằng cách cho phép gán các series chuyên biệt cho chỉ số chương trình, có thể khóa bằng xác lập tham số hệ thống. Ví dụ, series chỉ số chương trình 9000 (trong khoảng 09000 đến 09999) sẽ không biểu thị trên màn hình điều khiển khi được khóa bằng tham số hệ thống. Ngoài ra, các chương trình trong series này cũng không thể biên tập hoặc in, v.v...

Nếu không xác lập tham số khóa, các chương trình thuộc series 9000 sẽ bình thường như mọi chương trình khác. Để tận dụng ưu thế của tính năng này nhằm bảo vệ các chương trình quan trọng, bạn hãy xem các tài liệu liên quan của Fanuc.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trước khi viết chương trình con cần suy nghĩ và lập kế hoạch cẩn thận. Do ứng dụng chủ yếu của chương trình con là sơ đồ gia công lặp lại, nhà lập trình phải có khả năng *phát hiện* sơ đồ gia công sẽ sử dụng trong chương trình con.

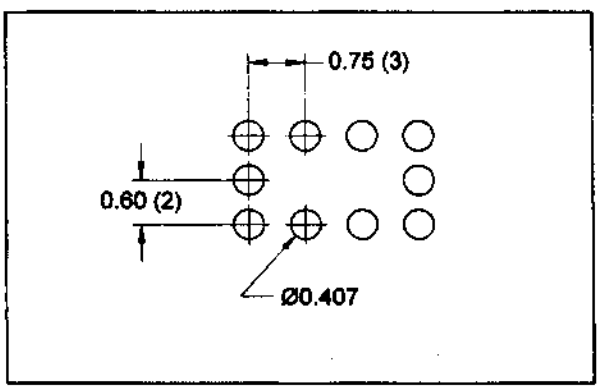
Nhận biết sơ đồ lặp

Khả năng nhận biết sơ đồ lặp là vấn đề kinh nghiệm. Các dấu hiệu đầu tiên sẽ đến trong khi viết chương trình bình thường theo từng block. Nếu có các nhóm block liên tiếp *lặp lại* và chứa dữ liệu *như nhau*, bạn nên đánh giá chương trình cẩn thận hơn và có thể viết chương trình con.

Nhà lập trình giàu kinh nghiệm sẽ không viết chương trình theo cách dài dòng trước, đó là sự lãng phí thời gian. Kinh nghiệm lập trình tăng cường khả năng nhận biết các dấu hiệu về chương trình con ngay từ các giai đoạn đầu khi quy hoạch lập trình. Tuy nhiên, đối với nhà lập trình ít kinh nghiệm, sẽ là bình thường khi viết chương trình dài trước. Điều này sẽ tốn thêm thời gian và không hiệu quả, nhưng sẽ tích lũy kinh nghiệm chuyên nghiệp. Với kinh nghiệm hạn chế, nên viết lại chương trình từ dạng dài để có một chương trình chính và một hoặc nhiều chương trình con. Nhà lập trình sẽ có khả năng nhận biết các phần trong chương trình dài có thể chuyển thành chương trình con. Sau khi nhận biết chuỗi dữ liệu lặp lại trong chương trình đó, vấn đề duy nhất là thực hiện các điều chỉnh nhỏ để tách các cụm block lặp lại này và xác định chúng theo chương trình con.

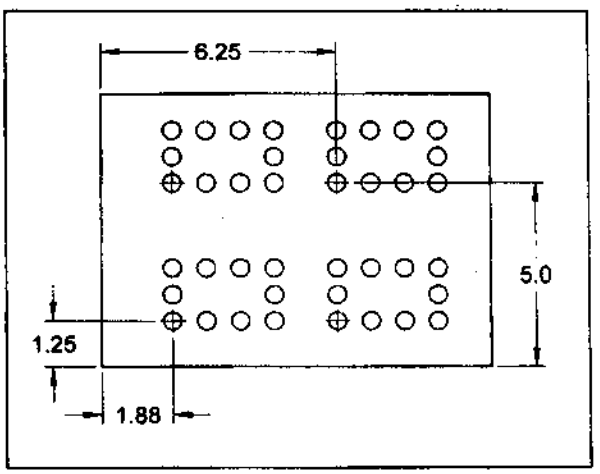
Chuyển động dao và chương trình con

Một trong các ứng dụng chương trình con phổ biến là quỹ đạo gia công ở các vị trí khác nhau trên chi tiết. Ví dụ, sơ đồ 10 lỗ xếp theo hình chữ nhật cần được lập trình (Hình 38.4).



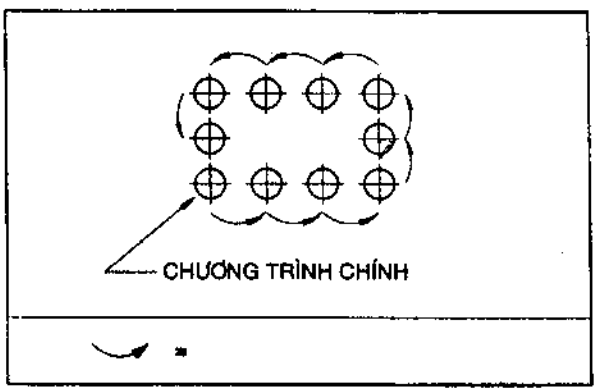
Hình 38.4. Chi tiết về sơ đồ các lỗ được dùng trong chương trình O3804.

Sơ đồ các lỗ này lặp lại ở bốn vị trí trên chi tiết (Hình 38.5).



Hình 38.5. Sắp xếp sơ đồ các lỗ trong các ví dụ chương trình O3804 và O3805 (cả hai đều sử dụng chương trình con O3855).

Chương trình con O3855 chứa sơ đồ này và sử dụng địa chỉ L để thiết lập số lần lặp chu kỳ cố định. Trong chương trình thứ nhất, O3804, chuyển động dao trước block chương trình con. Để bắt đầu viết chương trình, bạn hãy tập trung vào sơ đồ các lỗ. Thứ nhất, có thể chọn chế độ số gia G91 cho sơ đồ này, sau đó lập trình các giá trị số gia X và Y, bắt đầu từ lỗ bất kỳ, chẳng hạn góc dưới bên trái và tiếp tục theo một chiều (Hình 38.6).



Hình 38.6. Lưu đồ xử lý chương trình con O3855

```
O3855 (SUBPROGRAM)
(FOUR-CORNER LOCATIONS)
N551 G91 X0.75 L3
N552 Y0.6 L2
N553 X-0.75 L3
N554 Y-0.6
N555 M99
%
```

Chương trình con được thiết kế để gia công

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

chín lỗ trong sơ đồ thứ nhất. Lỗ thứ mười thực tế là lỗ thứ nhất – được gia công trong block với lệnh gọi chu kỳ hoặc chuyển động nhanh. Bốn vị trí sơ đồ các lỗ không gộp trong chương trình con, chúng phải được xếp trong chương trình chính. Do chương trình chính sử dụng chế độ tuyệt đối G90, có thể thiết lập các vị trí riêng rẽ:

```
O3804 (MAIN PROGRAM)
(FOUR-CORNER PATTERN)
N1 G20
N2 G17 G40 G80
N3 G90 G00 G54 X1.88 Y1.25
N4 G43 Z1.0 S350 M03 H01
N5 G99 G81 R0.1 Z-0.269 F3.5        (LL HOLE 1)
N6 M98 P3855                      (LL PATTERN)
N7 G90 X6.25 Y1.88                  (LR HOLE 1)
N8 M98 P3855                      (LR PATTERN)
N9 G90 X6.25 Y5.0                   (UR HOLE 1)
N10 M98 P3855                     (UR PATTERN)
N11 G90 X1.88 Y5.0                  (UL HOLE 1)
N12 M98 P3855                     (UL PATTERN)
N13 G80 G90 G28 Z1.0 M05
N14 G91 G28 X0 Y0
...
```

Trong ví dụ này chỉ sử dụng một dụng cụ cắt, các dụng cụ cắt khác cũng tuân theo cùng quy trình viết chương trình đó. Phương pháp này rất thông dụng – trong chế độ tuyệt đối từ chương trình chính, dao được định vị ở góc dưới bên trái của sơ đồ và lỗ thứ nhất được khoan tại vị trí đó. Sau khi gọi chương trình con và khoan xong chín lỗ, sử dụng các lệnh định vị theo số gia và số lần lặp lại chu kỳ. Số lần lặp lại trong chương trình con là số khoảng cách giữa các lỗ, *không* phải số lỗ.

Phương pháp đơn giản hơn, đặc biệt hữu dụng đối với số lượng lớn các vị trí sơ đồ, là *kết hợp* chuyển động nhanh đến vị trí khởi đầu sơ đồ với lệnh gọi chương trình con. Điều này là chấp nhận được đối với hầu hết các hệ điều khiển:

```
O3805 (MAIN PROGRAM)
(FOUR-CORNER PATTERN)
N1 G20
N2 G17 G40 G80
N3 G90 G00 G54 X1.88 Y1.25
N4 G43 Z1.0 S350 M03 H01
N5 G99 G81 R0.1 Z-0.269 F3.5 M98 P3855
N6 G90 X6.25 Y1.88 M98 P3855
N7 G90 X6.25 Y5.0 M98 P3855
N8 G90 X1.88 Y5.0 M98 P3855
N9 G80 G90 G28 Z1.0 M05
N10 G91 G28 X0 Y0
...
```

Ưu điểm chính của O3805 là rút ngắn chương trình O3804. Cả hai đều cho cùng kết quả, chọn phương pháp nào là tùy theo sở thích cá nhân. Bạn hãy chú ý, dường như không cần lặp lại lệnh chế độ G90 và các trục X, Y. Các giá trị chế độ trong chương trình con cần được chú ý đặc biệt.

Giá trị chế độ và chương trình con

Mọi giá trị chế độ có hiệu lực khi gọi chương trình con sẽ giữ nguyên hiệu lực trong chương trình con đó, trừ khi có sự thay đổi bên trong.

Trong ví dụ O3804 và O3805, bạn hãy chú ý sự lặp lại của G90, X6.25, và Y5.0. Chúng *rất quan trọng*. Chương trình con O3855 thay đổi trạng thái điều khiển sang chế độ số gia G91 và lỗ cuối cùng của sơ đồ mười lỗ được gia công khi hoàn tất chuyển động nhanh đến lỗ đó trong chế độ tuyệt đối. Điều này xảy ra trong chương trình chính, *không* trong chương trình con.

Tiếp theo là một vấn đề khác. Chương trình con gia công tinh biên dạng sử dụng bù bán kính dao G41 hoặc G42 với địa chỉ D. Nếu chương trình con này, ví dụ, được dùng để gia công bán tinh và để lại lượng dư, chương trình đó sẽ không hoạt động. Lý do là địa chỉ D là cố định và được lưu trong bộ điều khiển theo bán kính dao toàn phần. Giải pháp? Bạn hãy dùng hai giá trị bù D và lấy địa chỉ D ra khỏi chương trình con, sau đó gọi địa chỉ D cùng với M98, ví dụ:

```
M98 P. . D. .
```

Bằng cách đó, giá trị bù D có thể thay đổi vào thời điểm bất kỳ gọi chương trình con, mà không cần thay đổi trong chương trình con đó. Phương pháp này rất hữu ích nếu biên dạng lập trình yêu cầu *hai* hoặc *nhiều* giá trị bù, nhưng chỉ khả dụng trên một số bộ điều khiển. Dưới đây là nội dung của chương trình con gia công biên dạng đơn giản có bù D bên trong. Giá trị xác lập D51 là bằng bán kính dao:

```
O3856 (CONTOUR SUBPROGRAM - A)
N561 G41 G01 X0 D51 F10.0          (D.. INCLUDED)
N562 Y1.75
N563 G02 X0.25 Y2.0 R0.25
N564 G01 X1.875
N565 Y0
N566 X-0.75
N567 G00 G40 Y-0.75
N568 M99
%
```

Để gia công tinh biên dạng, chương trình con này được gọi theo cách thức bình thường, từ chương trình chính:

```
M98 P3856
```

Chương trình con này có thể dùng để gia công tinh *và* gia công bán tinh, để lại lượng dư nhỏ, nhưng phải sử dụng hai giá trị bù D, chẳng hạn D51 và D52. Trong trường hợp đó, D51 lưu giá trị bán kính dao và lượng dư để gia công tinh (D51 = bán kính dao + lượng dư), D52 chỉ lưu bán kính hoàn tất (D52 = bán kính dao). Đối với dao phay Ø.500, các giá trị có thể là:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
D51 = bán kính .250 + lượng dư .007 = .257
D52 = bán kính .250 + lượng dư .000 = .250
```

Kế tiếp, D.. phải được *loại ra* khỏi chương trình con:

```
O3857 (CONTOUR SUBPROGRAM - B)
N561 G41 G01 X0 F10.0        (D.. NOT INCLUDED)
N562 Y1.75
N563 G02 X0.25 Y2.0 R0.25
N564 G01 X1.875
N565 Y0
N566 X-0.75
N567 G00 G40 Y-0.75
N568 M99
%
```

Bộ điều khiển *yêu cầu* giá trị bù D nhưng không nhất thiết phải trong block chứa G41/G42. Khi D được chuyên biệt trước G41/G42, có thể chuyển vào chương trình con từ chương trình chính, tùy theo nguyên công cắt gọt:

```
M98 P3857 D51          ...gia công bán tinh
M98 P3851 D52          ...gia công tinh
```

Đây là phương pháp lập trình rất mạnh sử dụng các chương trình con cho hơn một nguyên công, nếu bộ điều khiển hỗ trợ điều đó.

Trở về từ chương trình con

Khi hoàn tất chương trình con, cần xóa các giá trị chế độ hiện hành trong chương trình chính. Các giá trị có thể thay đổi trong chương trình con gồm chế độ số gia hoặc tuyệt đối, lệnh chuyển động, chất làm nguội, ... Chương trình con luôn luôn là *nhánh* của chương trình khác – *kéo dài liên tục* của chương trình gốc, và là phần tích hợp trong chương trình đó. Mọi giá trị chế độ xác lập ở vị trí bất kỳ trong chương trình đều có hiệu lực cho đến khi thay đổi hoặc được xóa bằng lệnh trong cùng nhóm. Hàm kết thúc chương trình con M99 không xóa giá trị chế độ bất kỳ đang có hiệu lực.

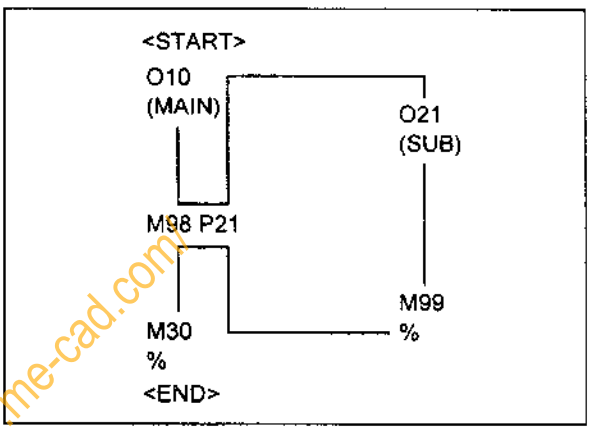
Từ các ví dụ O3804 và O3805 có thể thấy chu kỳ cố định được gọi từ chương trình chính chỉ một lần. Mọi dữ liệu chu kỳ chế độ đều được chuyển đến các chương trình con. Chương trình chính nêu rõ các giá trị chế độ hiện hành.

SẮP XẾP CHƯƠNG TRÌNH CON ĐA MỨC

Ví dụ nêu trên cho thấy chương trình chính chỉ gọi một chương trình con, và chương trình con này không gọi chương trình con khác. Điều đó được gọi là sắp xếp *một mức*, hoặc sắp xếp sâu một mức. Điều đó có nghĩa là, nếu chương trình chính gọi chương trình con số một, chương trình con đó có thể gọi chương trình con số hai, có thể gọi chương trình con số ba, và gọi chương trình con số bốn. Điều này được gọi là sắp xếp *bốn mức*. Ứng dụng thực tế ít khi sử dụng cả bốn mức, nhưng chúng là các công cụ lập trình khả dụng. Các ví dụ tiếp theo sẽ minh họa lưu đồ xử lý chương trình cho từng mức sắp xếp.

Sắp xếp một mức

Sắp xếp một mức có nghĩa là chương trình chính chỉ gọi một chương trình con. Chương trình con sắp xếp sâu một mức là rất phổ biến trong lập trình CNC. Sự xử lý chương trình bắt đầu từ đầu chương trình chính. Khi gọi chương trình con từ chương trình chính bằng block M98 P.., bộ điều khiển chuyển sang phần đầu của chương trình con đó và xử lý nội dung trong chương trình này, sau đó trở về chương trình chính để xử lý các block còn lại của chương trình chính (Hình 38.7).



Hình 38.7. Sắp xếp chương trình con một mức.

Sắp xếp hai mức

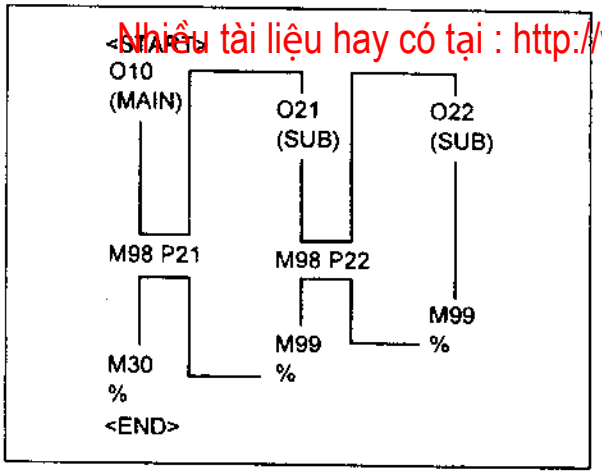
Sự xử lý chương trình con được sắp xếp sâu hai mức cũng bắt đầu từ phần đầu của chương trình chính. Khi bộ điều khiển gặp lệnh gọi chương trình con ở mức thứ nhất, sẽ tách khỏi chương trình chính và bắt đầu xử lý các block trong chương trình con thứ nhất, từ block đầu tiên của chương trình này. Trong khi xử lý chương trình con mức thấp, hệ thống CNC gặp block gọi chương trình con mức thứ hai.

Tại điểm này, sự xử lý chương trình con thứ nhất *tạm thời* dừng lại, hệ thống CNC chuyển sang mức thứ hai. Do không có lệnh gọi chương trình con từ mức thứ hai, mọi block chương trình con này sẽ được xử lý. Khi gặp block chứa hàm M99, hệ thống sẽ tự động trở về chương trình *đã rẽ nhánh* và *khôi phục* sự xử lý chương trình tạm thời cũng dừng lại trước đó.

Sự trở về chương trình gốc thường là đến block ngay sau khi gọi chương trình con trong chương trình đó. Mọi block còn lại trong chương trình con thứ nhất sẽ được thực thi cho

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



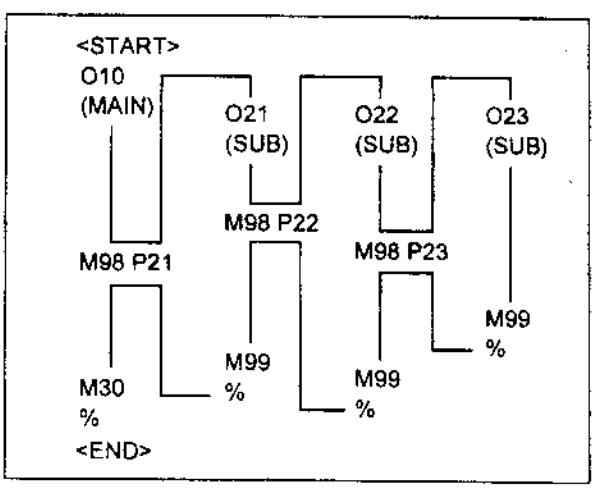
Hình 38.8. Sắp xếp chương trình con hai mức.

đến khi gặp hàm M99. Khi điều này xảy ra, hệ điều khiển sẽ trở về chương trình đã rẽ nhánh (chương trình gốc) trong trường hợp này là chương trình chính.

Do vẫn còn các block trong chương trình chính, chúng sẽ được xử lý cho đến khi gặp hàm M30. Hàm này kết thúc sự thực thi chương trình chính. Hình 38.8 minh họa sơ đồ khái niệm sắp xếp chương trình con hai lớp.

Sắp xếp ba mức

Sắp xếp đến chiều sâu ba mức là sự mở rộng logic của sắp xếp hai mức. Cũng như hai mức, khởi động từ đầu chương trình chính (Chương trình 010 trên Hình 38.9), nhánh thứ nhất sẽ là mức thứ nhất (021), nhánh kế tiếp (022) và nhánh thứ ba là 023. Từng chương trình con được xử lý cho đến lệnh gọi chương trình con kế tiếp, hoặc đến cuối chương trình con. Sự xử lý chương trình luôn luôn trở lại block ngay sau block gọi chương trình con, kết thúc trong chương trình chính.



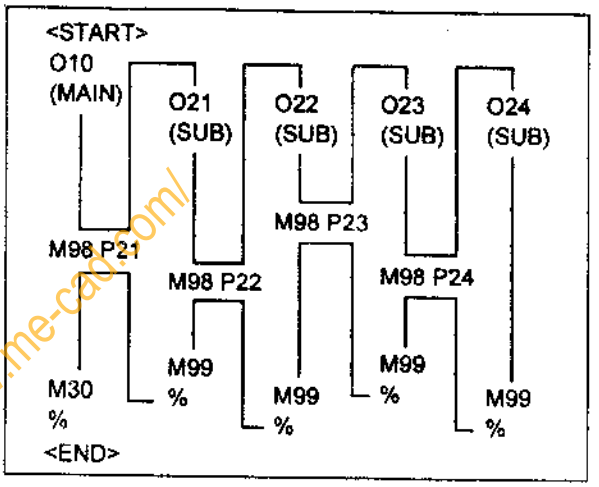
Hình 38.9. Sắp xếp chương trình con ba mức.

Sắp xếp bốn mức

Logic của sắp xếp chương trình con đa mức tương đối dễ hiểu. Sắp xếp bốn mức là sự mở rộng đa mức của sắp xếp đơn và đồng nhất về logic với mọi ví dụ đã nêu ở phần trên.

Sự bổ sung thêm các nhánh một cách không cần thiết trong sắp xếp chương trình con đa mức chỉ làm cho ứng dụng lập trình trở nên phức tạp hơn và khó quản lý hơn.

Lập trình sắp xếp chương trình con đến bốn mức (hoặc thậm chí chỉ ba mức) đòi hỏi kiến thức đầy đủ về thứ tự xử lý chương trình – và có ứng dụng thích hợp cho sự sắp xếp đó. Trong lập trình ở xưởng cơ khí, hầu như không cần sử dụng sắp xếp ba hoặc bốn mức. Lưu đồ chương trình sắp xếp bốn mức được nêu trên Hình 38.10.



Hình 38.10. Sắp xếp chương trình con bốn mức.

Các ứng dụng với nhiều mức chương trình con

Khi xét thực tế từng chương trình con có thể lặp lại đến 9999 lần trong chương trình bất kỳ gọi chương trình con đó, bạn sẽ thấy năng lực lập trình rất mạnh có thể sử dụng trong sản xuất. Bạn hãy chú ý các khó khăn tiềm ẩn khi viết chương trình với nhiều mức chương trình con. Phương pháp này có thể tạo ra chương trình ngắn nhưng tốn nhiều thời gian. Thời gian chuẩn bị chương trình, viết và gỡ rối, thường lâu hơn so với các chương trình bình thường. Không chỉ sự triển khai logic phức tạp và tốn nhiều thời gian, mà còn phải rất cẩn thận khi ghi lại lưu đồ xử lý tất cả các chương trình, xác lập các điều kiện ban đầu, kiểm tra giá trị và hiệu lực của dữ liệu ...

Nguyên tắc chung đối với kỹ thuật sắp xếp nhiều mức chương trình con – chỉ sử dụng trong

Nhiều tài liệu hay có tại : http://www.me-cad.com/

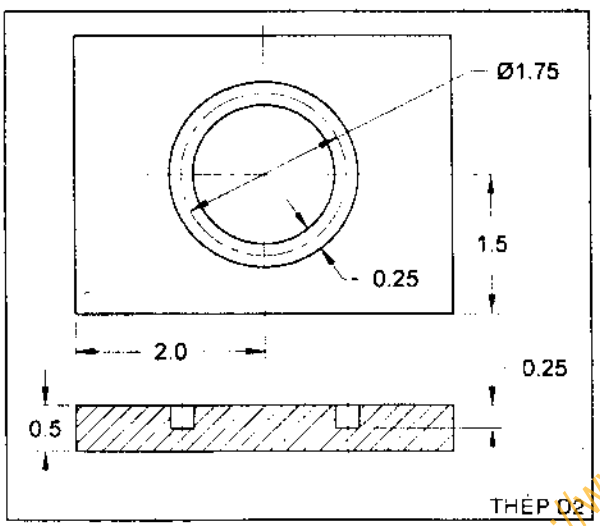
Nhiều tài liệu hay có tại : http://www.me-cad.com/

những trường hợp khi tần số sử dụng loại chương trình đó xứng đáng với chi phí về thời gian và công sức. Nói chung, sắp xếp nhiều mức luôn luôn có các ưu điểm và nhược điểm, cần cân nhắc cẩn thận cho từng trường hợp cụ thể.

GIA CÔNG BIÊN DẠNG VỚI CHƯƠNG TRÌNH CON

Các ví dụ chương trình sử dụng chương trình con cho đến phần này đều liên quan đến gia công lỗ, chúng cung cấp đủ nội dung để hiểu khái niệm chương trình con. Còn có nhiều ví dụ khác về sử dụng chương trình con trong các chương kế tiếp.

Dưới đây là ví dụ lập trình chương trình con gia công biên dạng XY đơn giản theo chiều sâu Z (Hình 38.11).

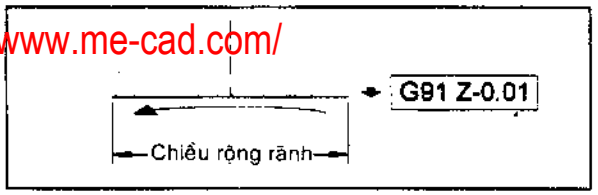


Hình 38.11. *Chương trình chính O3806 sử dụng chương trình con O3858*

Bản vẽ yêu cầu rãnh với bước Ø 1.750 được gia công đến chiều sâu .250 inch. Đây là rãnh thô, không yêu cầu dung sai chính xác và độ bóng bề mặt cao. Toàn bộ điều cần thiết là dao phay mặt đầu cắt tâm Ø .250 (khoan rãnh), ăn vào chiều sâu, lập trình quỹ đạo tròn 360^0.

Ngay cả với vật liệu dễ cắt gọt, chẳng hạn đồng thanh, chia một chiều sâu cắt .250 thành hai lần cắt sâu .125 cũng vẫn có lợi. Vật liệu là *thép dụng cụ D2*, tương đối cứng. Tốc độ dao là 630 r/min và mỗi lần chỉ ăn vào chiều sâu .010 inch, lặp lại biên dạng rãnh 25 lần, đối với 25 × .010 = .250 inch chiều sâu cắt yêu cầu. Chế độ gia công này rất thích hợp với chương trình con. Sơ đồ chi tiết cắt chiều sâu theo một số gia (.010) được minh họa trên Hình 38.12.

Chương trình con O3858 chỉ chứa *các chuyển động dao chung cho một đường cắt rãnh*. Điều đó có nghĩa là cắt sâu theo số gia.



Hình 38.12. *Chi tiết về chương trình con O3858.*

010 và cắt tròn 360^0. Mọi chuyển động khác sẽ trong chương trình chính O3806. Bạn hãy lưu ý thuật ngữ số gia theo chiều sâu cắt. Giá trị .010 phải được lập trình theo *số gia*, nếu không dùng chế độ số gia, dao sẽ cắt theo chiều sâu tuyệt đối Z-0.01 – đến 25 lần. Dưới đây là chương trình chính đầy đủ O3806 và chương trình con O3858 (giả thiết dao T01 đã lắp trong trục chính):

```
O3806 (MAIN FOR SIMPLE DEEP GROOVE)
(T01 - 0.250 DIA CENTER CUTTING END MILL)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X2.875 Y1.5 S630 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z0 F10.0      (START Z POSITION AT Z0 !)
N6 M98 P3858 L25   (CALL SUBPROGRAM 25 TIMES)
N7 G90 G00 Z1.0 M09
N8 G28 Z1.0 M05
N9 M30
%

O3858 (SUB FOR O3806)
N581 G91 G01 Z-0.01 F0.5 (INCREMENT BY -0.01)
N582 G03 I-0.875 F2.0  (FULL CIRCLE CONTOUR)
N583 M99
%
```

Chương trình này khá đơn giản, nhưng minh họa hai yêu cầu quan trọng cần duy trì trong khi viết chương trình con bất kỳ. Các vấn đề này liên quan đến sự bảo đảm *quan hệ liên tục giữa chương trình chính và chương trình con*. Đó là các yêu cầu đặc biệt:

- ... duy trì sự chuyển từ chương trình chính sang chương trình con (trước khi gọi chương trình con).
- ... duy trì sự chuyển từ chương trình con trở lại chương trình chính (sau khi hoàn tất chương trình con).

Yêu cầu thứ nhất được đáp ứng trong block N5. Vị trí trục Z phải là Z0, không thể ở vị trí khác. Vị trí Z0, sẽ cho phép dao gia tăng 25 lần khoảng cách .010, do đó đạt chiều sâu .250. Nói một cách khác, vị trí khởi đầu của dao trước khi gọi chương trình con phải là vị trí bảo đảm quỹ đạo dao chính xác.

Yêu cầu thứ hai được đáp ứng trong block N7. Lệnh G90 làm cho block này trở nên đặc biệt. Tại sao? Do chương trình con sử dụng chế độ số gia G91. Khi sự xử lý chương trình con chuyển sang chương trình chính, cần chuyển chế độ số gia sang chế độ tuyệt đối với lệnh G90.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chuỗi thứ tự lập trình để thay dao tự động (ATC) thường ngắn và đơn giản. Đối với hệ thống phay CNC, hàm M06 sẽ thực thi công việc này, còn đối với máy tiện CNC là hàm T. không thể lập trình sự thay dao nếu không thiết lập các điều kiện xác định. Các hàm chương trình liên quan với sự trở về zero (chuẩn) máy, xóa chất làm nguội, dừng trục chính, ..., đều là phần tích hợp của chương trình thay dao. Cần khoảng ba, bốn, năm, ..., block chương trình để thiết lập đủ các điều kiện, *mỗi khi* lập trình thay dao tự động. Điều cơ bản là các block này luôn luôn có nội dung như nhau, bất kể chương trình đang được sử dụng.

Ví dụ, bạn hãy xét thứ tự các bước dưới đây, rất phổ biến, cần thực hiện để thay dao nhiều lần trong một chương trình.

Ví dụ này dựa trên trung tâm gia công đứng CNC, sử dụng hàm thay dao tự động (ATC):

1. Dừng chất làm nguội.
2. Xóa chế độ chu kỳ cố định.
3. Xóa chế độ bù bán kính dao.
4. Tắt trục chính.
5. Trả về vị trí quy chiếu máy trên trục Z.
6. Xóa các giá trị bù.
7. Thực hiện sự thay dao.

Bảy bước này xảy ra trong *từng chương trình* yêu cầu sự thay dao cụ thể và chúng sẽ xảy ra cho *từng dao* trong *chương trình*. Chương trình thay dao đơn giản lặp lại nhiều lần và được sử dụng thường xuyên. Để lập trình dễ dàng hơn, bạn có thể viết chương trình con gồm cả bảy bước nêu trên, và gọi vào chương trình chính mỗi khi cần thay dao:

```
O3859 (TOOL CHANGE VERTICAL MACHINING CENTER)
N1 M09
N2 G80 G40 M05
N3 G91 G28 Z0
N4 G49 D00 H00
N5 G90 M06
N6 M99
%
```

Ví dụ này dễ dàng chỉnh sửa để phù hợp với các loại máy công cụ khác, kể cả trung tâm gia công CNC ngang. Có thể gộp cả các yêu cầu đặc biệt, chẳng hạn một số tùy chọn của nhà chế tạo. Thậm chí có thể lập trình sự thay dao tại vị trí bàn máy xác định. Chỉnh sửa duy nhất là bổ sung block G53X.. Y.. trước block thay dao. Ví dụ thứ hai là mã đặc biệt để thay dao *và* hàm chất làm nguội ON (hoạt động). Một số nhà chế tạo máy đưa ra hàm M đặc biệt, kết hợp hai hàm tiêu chuẩn, ví dụ M16 – kết hợp các hàm tiêu chuẩn M06 và M08.

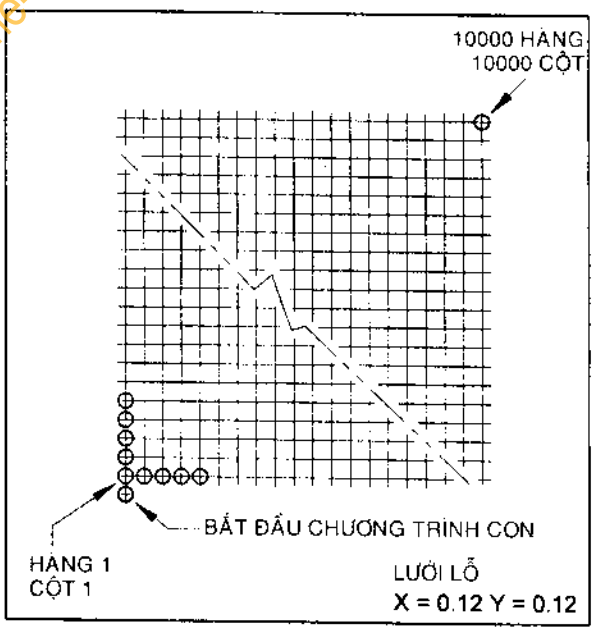
Ngoài ra, bạn hãy chú ý các hàm xóa – chỉ có vài hàm xóa trong chương trình con O3859. Khi thiết kế chương trình con, nhà lập trình hầu như không biết khi nào hàm chất làm nguội là ON hay OFF, không biết chu kỳ cố định hoặc bù bán kính dao có hoạt động hay không. Ngoài ra, nhà lập trình cũng có thể không biết trạng thái hiện hành là chế độ G90 hay G91.

Trạng thái *thực* của chúng nói chung không quá quan trọng. Các hàm xóa này được đưa vào chương trình con, tận dụng ưu thế xóa một hàm đã bị xóa sẽ được hệ điều khiển bỏ qua.

LƯỚI 100 000 000 LỖ

Phần này sẽ xem xét các chương trình con từ góc độ khác, nhưng với ví dụ *thực*, nhằm chứng minh năng lực rất mạnh của kỹ thuật ứng dụng chương trình con.

Ví dụ này minh họa phương pháp có thể khoan một trăm triệu lỗ (vâng, *một trăm triệu* lỗ) sử dụng chương trình chỉ có 29 block với hai dụng cụ cắt, kể cả chỉ số chương trình và mã dừng (dấu %). Hình 38.13 minh họa sơ đồ lưới đơn giản gồm 10000 hàng (X) và 10000 cột (Y).



Hình 38.13. *Sơ đồ lưới vuông – 100.000.000 lỗ*

Để làm cho ví dụ này trở nên hợp lý, đơn giản, và thú vị, các lỗ sẽ rất nhỏ, chỉ có ∅5/64 (.0781), với bước .120 dọc theo từng trục, tạo thành sơ đồ lưới vuông gồm các lỗ rất gần nhau.

Chỉ sử dụng hai dụng cụ cắt, *mũi khoan*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

miệng với góc đỉnh 120^0 để khởi đầu lỗ khoan và mũi khoan ∅5/64. Cả hai dụng cụ cắt đều bắt đầu gia công từ vị trí chu kỳ R0.06 phía trên tấm chi tiết đến chiều sâu tương ứng của chúng: Z-0.04 đối với khoan miệng và Z-0.215 đối với khoan lỗ.

Từ quan điểm lập trình, thiết kế chương trình không khó, chỉ sử dụng chương trình chính và một chương trình con. Các bước lập trình gia công 100.000.000 lỗ hoàn toàn tương tự lưới 100 lỗ.

Chương trình chính chứa các xác lập tiêu chuẩn và gọi chương trình con. Chương trình con sẽ lặp lại chu kỳ cố định hoạt động 9999 lần cho hai hàng, mỗi hàng theo một chiều.

Vị trí khởi đầu của chuyển động dao thứ nhất là X1.0 Y1.0 (dịch chuyển .120 theo Y-). Chu kỳ cố định khoan lỗ thứ nhất, lặp lại 9999 lần, chuyển một bước sang trục Y+, khoan lỗ và lặp lại 9999 lần theo trục X-. Sơ đồ chương trình con này lặp lại 5000 lần trong chương trình chính.

```
O3860 (SUBPROGRAM)
N601 G91 Y0.12
N602 X0.12 L9999
N603 Y0.12
N604 X-0.12 L9999
N605 M99
%

O3807 (MAIN PROGRAM)
N1 G20
N2 G17 G40 G80 T01                    (SPOT DRILL)
N3 M06
N4 G90 G00 G54 X1.0 Y1.0 S3000 M03 T02
N5 G43 Z1.0 H01 M08
N6 G99 G82 R0.06 Z-0.04 P30 F5.0 L0
N7 M98 P3860 L5000
N8 G90 G80 Z1.0
N9 G28 Z1.0
N10 M01

N11 T02                               (5/64 DRILL)
N12 M06
N13 G90 G00 G54 X1.0 Y1.0 S3000 M03 T01
N14 G43 Z1.0 H02 M08
N15 G99 G81 R0.06 Z-0.215 F4.0 L0
N16 M98 P3860 L5000
N17 G90 G80 Z1.0
N18 G28 Z1.0
N19 G91 G28 X0 Y0
N20 M30
%
```

Thiết kế chương trình tận dụng ưu thế của chương trình con và số lần lặp lại tối đa.

Điều làm cho chương trình còn thú vị hơn là xác định *thời gian gia công*. Bạn thử ước tính *thời gian gia công tất cả các lỗ đó với hai dụng cụ cắt*. Tốc độ và lượng ăn dao hợp lý đối với hầu hết các vật liệu, thời gian dịch chuyển dao, thời gian tạm nghỉ đối với khoan miệng, ... Giả sử tốc độ chuyển động nhanh trên tất cả hai trục là 475 in/min, giá trị này là hợp lý. Để thuận tiện, có thể bỏ qua các chuyển động giữa zero máy và vị trí thứ nhất.

Tính toán đầu tiên là tìm thời gian để thực hiện chuyển động nhanh giữa tất cả các lỗ. Một trăm triệu khoảng cách đều (trừ một) nhân với .120 chia cho 475 in/min là 25263, 1576 phút, và nhân với hai, do có hai dao, sẽ là 50526, 3163 phút.

Mũi khoan miệng sẽ dịch chuyển .060 từ phía trên đến bề mặt chi tiết và chiều sâu cắt.040, tổng chiều dài là .100, nhân với một trăm triệu lỗ với tốc độ 5 in/min, do đó thời gian khoan miệng là 2.000.000 phút. Mũi khoan miệng rút nhanh ra khỏi lỗ một trăm triệu lần khoảng cách .100 với tốc độ 475 in/min, tổng cộng 21052, 6316 phút, thời gian tạm dừng ở từng lỗ là 0.030 giây, đổi thành phút sẽ là 50.000 phút.

Khoan lỗ đến chiều sâu.215 từ mức .060 phía trên, hành trình sẽ là .275 với tốc độ 4.0 in/min, với một trăm triệu lỗ cần 6.875.000 phút. Mũi khoan rút nhanh một trăm triệu lần khoảng cách .275 với tốc độ 475 in/min, cần 57894, 7368 phút.

Tổng các kết quả nêu trên là 9054473, 6837 phút, tương ứng 150907, 8947 giờ, là 6287, 829 ngày, nghĩa là 17, 22 69 năm. Tin hay không là tùy bạn, cần đến hơn 17 *năm* gia công *liên tục* để khoan miệng và khoan một trăm triệu lỗ, và tất cả có thể thực hiện với chương trình chính và chương trình con có không đến ba chục block nhập.

Chuyển sang các chi tiết liên quan khác, kích cỡ tấm chi tiết chưa tính các biên sẽ là 100 × 100 ft, do đó hành trình gia công thực phải lớn hơn 100 ft dọc theo trục X và trục Y. Thật khó có máy CNC nào trên thị trường có thể giải quyết nhiệm vụ này. Ví dụ, gá lắp tấm chi tiết như thế nào? Đó lại là câu hỏi khác.

Một vấn đề thú vị khác, bạn hãy xét thời gian cần thiết để lập trình, *không có* chương trình con và *không có* sự đếm lặp (địa chỉ L). Giả sử mỗi block cần 6 giây để viết, và 55 block là đủ trên một trang giấy (khổ A4); sẽ cần đến 19 năm(vâng, *mười chín năm*!) liên tục để viết chương trình cho hai dụng cụ cắt đó. Bạn cần đến 1818182 tờ giấy, tương ứng chồng giấy cao 705 ft (215 mét). Chừng đó cũng đủ để thấy *sức mạnh của chương trình con*.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

39 DỊCH CHUYỂN MỐC CHUẨN

Đa số các chương trình CNC đều là chương trình liên quan đến một loại công việc, trên máy cụ thể trong xưởng. Công việc đó có các tính chất đặc thù, các yêu cầu chuyên biệt và quỹ đạo chạy dao riêng. Quỹ đạo dao là quan trọng nhất trong mọi tính năng của chương trình CNC.

Trách nhiệm chính của nhà lập trình CNC là triển khai quỹ đạo chạy dao cho sự gia công bất kỳ, không có lỗi và theo cách thức hiệu quả nhất. Sự triển khai quỹ đạo dao là rất quan trọng, do sẽ biểu thị sơ đồ gia công đặc thù cho chi tiết cụ thể và không liên quan với chương trình CNC khác. Nói chung, nhà lập trình sẽ gặp các cơ hội, với sơ đồ gia công hiện hữu có thể sử dụng cho nhiều chi tiết khác. Điều này đòi hỏi sự triển khai các chương trình hiệu quả hơn, tạo ra các chương trình CNC cho nhiều ứng dụng và không bị lỗi.

Kỹ thuật lập trình hướng đến vấn đề này được gọi là *chuyển đổi sơ đồ gia công*, hoặc *dịch chuyển mốc chuẩn*. Ví dụ điển hình của kỹ thuật này là tạm thời thay đổi điểm quy chiếu chương trình (chuẩn chương trình, zero chương trình) từ vị trí ban đầu đến vị trí mới, được gọi là *dịch chuyển chuẩn* (gia công). Các kỹ thuật lập trình khác bao gồm *Hình ảnh đối xứng gương*, sẽ được trình bày trong Chương 40, *Quay tọa độ*, và *Hàm lập tỷ lệ*, sẽ được nêu trong các chương kế tiếp.

Chương này trình bày chi tiết chủ đề *dịch chuyển mốc chuẩn*, còn gọi là *chuyển đổi sơ đồ gia công*. Đây là tính năng cơ bản của mọi hệ thống CNC, có thể áp dụng theo nhiều phương cách khác nhau.

DỊCH CHUYỂN MỐC CHUẨN VỚI G92 HOẶC G50

Về bản chất, dịch chuyển mốc chuẩn là tái định vị tạm thời hoặc lâu dài cho chuẩn chi tiết (điểm quy chiếu chương trình) bên trong chương trình. Khi sử dụng kỹ thuật lập trình này, sẽ định vị lại sơ đồ gia công hiện hữu (quỹ đạo dao) trong chương trình ở các vị trí khác bên trong vùng làm việc của máy CNC.

Chương 15 đã giải thích lệnh G92 (phay) và lệnh G50 (tiện), trước khi tiếp tục, bạn nên xem lại các lệnh này. Bạn cần nhớ, các lệnh này *không* gây ra chuyển động chạy dao trực tiếp, chỉ tác động đến chuyển động dao bất kỳ *theo sau* lệnh đó. Lệnh đăng ký vị trí G92 và G50 đăng ký các tọa độ *tuyệt đối* của vị trí dao hiện hành nhưng hoàn toàn không ảnh hưởng đến các kích thước số gia (tương đối), khi sử dụng lệnh G91 đối với phay hoặc các trục U/W đối với tiện. Mục đích của chúng là "báo" cho hệ điều khiển *vị trí dao hiện hành*. Bước này là cần thiết ít nhất một lần ở phần đầu của từng dao để thiết lập quan hệ giữa zero chương trình cố định (chuẩn chi tiết) và vị trí thực của dao. Ví dụ,

```
G92 X10.0 Y6.5
```

"báo" cho hệ điều khiển dao cắt được xác lập tại 10.0 đơn vị dương cách xa zero chương trình theo trục X và 6.5 đơn vị dương theo trục Y.

Điều gì sẽ xảy ra nếu đăng ký sai vị trí? Nếu các giá trị trong lệnh G92 hoặc G50 không phản ánh *đúng* vị trí của dao cắt? Quỹ đạo dao sẽ sai và hậu quả sẽ là phế phẩm, gãy dao, thậm chí máy bị hư hại.

Nhà lập trình CNC giàu trí tưởng tượng luôn luôn cố gắng tìm các phương pháp để sử dụng tốt nhất các công cụ lập trình hiện có. Lệnh G92 và G50 chỉ là hai trong nhiều công cụ hữu ích cho nhà lập trình CNC.

Đối với các công việc đơn giản, có lẽ không cần các sáng tạo đặc biệt. Sẽ không thật tinh tế khi đầu tư nhiều thời gian để bổ sung các tính năng cho chương trình mà không đem lại các ưu thế thật sự. Khi cần, hoàn toàn có thể tối ưu hóa chương trình dễ dàng.

Dịch chuyển zero chương trình

Nếu sử dụng lệnh G92 trên trung tâm gia công hoặc G50 trên máy tiện CNC, thay vì các lệnh bù G54 đến G59 mới hơn và rất hiệu quả, chỉ cần *một* lệnh đăng ký vị trí G92 (G50) *cho một dao*, với giả thiết không sử dụng các lệnh bù chi tiết.

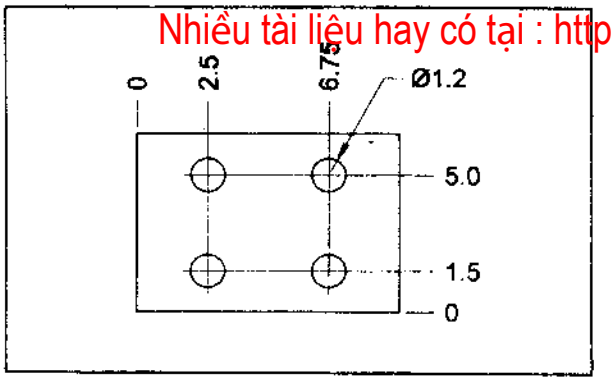
Nếu có hơn một lệnh đăng ký vị trí đối với một dao trong chương trình, điều đó sẽ có nghĩa là *sự dịch chuyển zero chương trình*.

Để minh họa khái niệm dịch chuyển zero chương trình, bạn hãy xem bản vẽ đơn giản trên Hình 39.1.

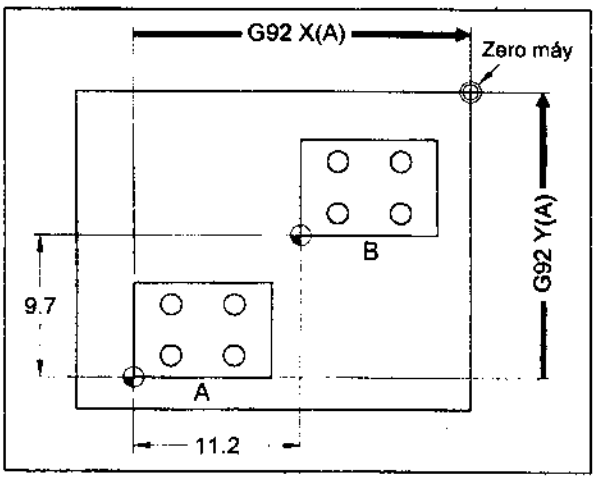
Dựa trên bản vẽ này, bốn lỗ sẽ được gia công ở *hai vị trí độc lập* trên bàn máy, như minh họa trên Hình 39.2.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 39.1. Bản vẽ minh họa sự dịch chuyển zero chương trình – chương trình O3901



Hình 39.2. Dịch chuyển zero chương trình sử dụng lệnh G92 cho hai chi tiết – O3901

G92X(A) biểu thị khoảng cách X từ chuẩn chi tiết của *Chi tiết A* đến zero (chuẩn) máy, G92Y(A) là khoảng cách Y từ chuẩn chi tiết của *Chi tiết A* đến zero (chuẩn) máy. Chú ý, các khoảng cách này là từ zero chương trình đến zero máy. Nếu cần, chúng có thể đến vị trí khác, nhưng *phải* bắt đầu từ zero (chuẩn) chi tiết. Để sử dụng G92, *cần biết* các khoảng cách giữa hai chi tiết. Sử dụng các giá trị làm tròn để đơn giản hóa ví dụ:

Chi tiết A: `G92 X22.7 Y19.8 Z12.5`

Chi tiết B: `X-11.2 Y-9.7 Z0` *từ Chi tiết A*

Ngoài ra, bạn cần lưu ý, giá trị Z là *như nhau* đối với cả chi tiết A và chi tiết B, do chỉ dùng một dụng cụ cắt cho hai chi tiết đó. Để khoan điểm bốn lỗ ở hai vị trí, chương trình O3901 có thể viết như sau:

```
O3901
(G92 USED FOR TWO TABLE LOCATIONS)
N1 G20 G90
N2 G92 X22.7 Y19.5 Z12.5     (TOOL AT M/C ZERO)
N3 S1200 M03
N4 M08
N5 G99 G82 X2.5 Y1.5 R0.1 Z-0.2 P200 F8.0
N6 X6.75
N7 Y5.0
N8 X2.5                  (TOOL AT LAST HOLE OF PART A)
N9 G80 Z1.0
N10 G92 X-8.7 Y-4.7     (SET AT LAST HOLE OF A)
N11 G99 G81 X2.5 Y1.5 R0.1 Z-0.2 P200
N12 X6.75
N13 Y5.0
N14 X2.5                 (TOOL AT LAST HOLE OF PART B)
N15 G80 Z1.0
N16 G92 X-9.0 Y-4.8       (TOOL FROM M/C ZERO)
N17 G00 Z12.5 M09
N18 X0 Y0                    (TOOL AT M/C ZERO)
N19 M30
%
```

Cần làm rõ một số block, gồm các block N2, N8, N10, N14, N16, và N18. Từng block đó đều có quan hệ với vị trí dao hiện hành. *Ở đây bạn cần rất cẩn thận*. Nhà lập trình sẽ gặp khó khăn nếu chưa hiểu rõ các nguyên tắc tính toán của G92.

Dao cắt khởi đầu từ vị trí zero máy mỗi khi thực thi chương trình. Dao được lắp trong trục chính trước khi gia công. Trong block N2, zero (chuẩn) chi tiết được thiết lập cho chi tiết A. Dao tại điểm này cách zero chương trình 22.7 inch theo trục X, và 19.5 inch theo trục Y. Xác lập tọa độ trong block N2 phản ánh điều đó. Trong các block N7 và N8, dao hoàn tất lỗ cuối cùng của chi tiết A (tại X2.5Y5.0 của xác lập G92 hiện hành).

Block quan trọng kế tiếp là N10. Tại điểm này trong chương trình, chi tiết A đã hoàn tất, nhưng chưa gia công chi tiết B. Bạn hãy xem xét vị trí dao sau khi thực thi block N9. Dao ở vị trí X2.5 Y5.0, chương trình cần "báo" cho hệ điều khiển vị trí dao vào đúng thời điểm, nhưng liên quan đến chi tiết B. Điều này được thực hiện bằng phép tính số học đơn giản.

```
G92 (X) = 11.5 + 2.5 - 22.7 = -8.7
G92 (Y) =  9.8 + 5.0 - 19.5 = -4.7
```

Bạn hãy xem Hình 39.3 để hình dung tính toán này. Chiều của các mũi tên trong minh họa là rất quan trọng để xác định dấu (chiều) trục trong block G92.

Các block N13 và N14 chứa tọa độ đối với vị trí dao cuối cùng của chi tiết B. Từ minh họa, bạn sẽ hiểu ý nghĩa của các giá trị tọa độ trong block N16. Để hoàn tất chương trình dao cần trở về vị trí ban đầu (zero máy). Sự trở về này sẽ thực hiện từ X2.5Y5.0 của chi tiết B, cách zero máy 9.000 inch theo trục X và 4.800 inch theo trục Y:

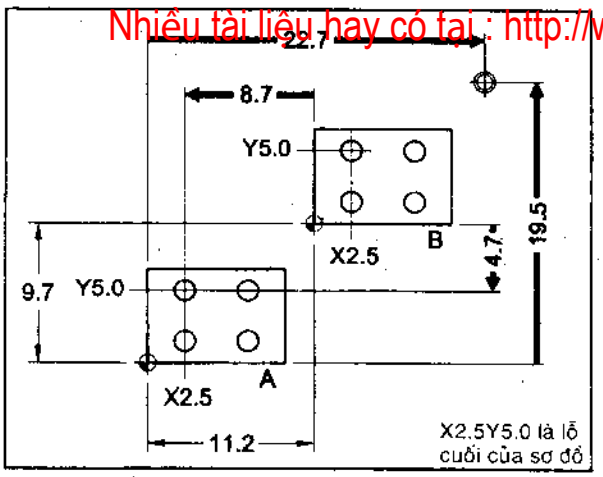
```
G92 (X) = 11.2 + 2.5 - 22.7 = -9.0
G92 (Y) =  9.7 + 5.0 - 19.5 = -4.8
```

Cả hai tọa độ lập trình X và Y đều âm.

Sau khi vị trí dao hiện hành được xác lập tại lỗ cuối cùng trên chi tiết B, có thể thực hiện

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 39.3. *Tính toán các tọa độ (XY) của G92 cho chương trình O3901*

sự trở về zero máy. Sự trở về này là cần thiết, do đây là vị trí của dao thứ nhất. Vị trí đích đối với zero máy X0Y0 không phải do đây là zero máy mà là do các tọa độ G92 được đo từ đó. Chuyển động X và Y thực tế đến zero máy được lập trình trong block N18.

HỆ TỌA ĐỘ CỤC BỘ

Lệnh G92 về đăng ký vị trí dùng cho lập trình tuyệt đối và đã lạc hậu. Theo thời gian, đã có các lệnh mới được bổ sung để điều khiển hệ tọa độ. Hệ tọa độ chính (bù chi tiết G54 đến G59) đã được trình bày trong các Chương trước, và phần trên đã nêu rõ không được dùng G92 khi lệnh bù bất kỳ đang có hiệu lực. Điều này nhằm tránh sự thay đổi zero chương trình dù chỉ là tạm thời. Giải pháp ở đây là *tập hợp con* có thể lập trình của hệ tọa độ chính (bù chi tiết) được gọi là *hệ tọa độ cục bộ* hoặc *hệ tọa độ con*.

Có nhiều trường hợp, khi bản vẽ được ghi kích thước theo cách thức các lệnh bù G54 đến G59 trở nên khó áp dụng. Ví dụ, sơ đồ lỗ bulông. Nếu chi tiết là hình tròn, có thể chọn zero chương trình tại *tâm* của sơ đồ lỗ. Tuy nhiên, nếu sơ đồ lỗ bulông trong diện tích hình chữ nhật, chuẩn chi tiết có thể ở góc *ngoài* của tấm phôi.

Thông thường, các vị trí tuyệt đối của các lỗ bulông sẽ được tính từ zero chương trình, trừ khi sử dụng *sự dịch chuyển* zero chương trình (lệnh G92) hoặc chọn hệ tọa độ đặc biệt.

Khi làm việc với các lệnh bù chi tiết, có ba phương pháp lập trình giúp cho công việc trở nên thuận tiện hơn và có thể giảm các sai sót tính toán, nếu có:

- Sử dụng tâm đường tròn phân bố các lỗ làm zero chương trình.
- Sử dụng các lệnh bù khác nhau trong chương trình, ví dụ, G54 để quy chiếu đến cạnh biên của chi tiết và G55 để quy chiếu theo tâm sơ đồ lỗ bulông.
- Sử dụng hệ tọa độ cục bộ, bên trong hệ độ chính hiện hành (bù chi tiết) được chọn ở đầu chương trình.

Trong mọi trường hợp, luôn luôn có một ưu điểm, nhà lập trình sử dụng các tính toán liên quan đến tọa độ tâm đường tròn bulông, *một cách trực tiếp trong chương trình CNC*, mà không cần các phép tính cộng và trừ. Phương pháp này thậm chí cho phép đơn giản hóa sự gá lắp trên máy.

Phương pháp thứ nhất, lập trình tâm đường tròn bulông, là phương pháp thông dụng đã được trình bày trong phần gia công sơ đồ các lỗ.

Phương pháp thứ hai, sử dụng sự thay đổi giữa các lệnh bù chi tiết, cũng đã được đề cập trong các Chương trước, nói chung tương đối dễ áp dụng. Hạn chế của phương pháp này là chỉ có *sáu* lệnh bù chi tiết khả dụng trên bộ điều khiển Fanuc – từ G54 đến G59. Nếu cần cả 6 lệnh bù cho chi tiết nào đó, sẽ không còn lệnh bù "dự trữ" để sử dụng cho các tình huống, chẳng hạn sơ đồ lỗ bulông. (Còn có các lệnh bù bổ sung, thường là tính năng tùy chọn của hệ điều khiển Fanuc)

Phương pháp thứ ba, sử dụng *hệ tọa độ cục bộ*, cho phép sử dụng hệ tọa độ *phụ thuộc*, còn gọi là hệ tọa độ *con*, *bên trong* sự bù chi tiết hiện hành. Có thể xác định số hệ tọa độ con bất kỳ bên trong hệ tọa độ chính, nhưng mỗi lần chỉ có thể làm việc trong một hệ tọa độ. *Chú ý*:

> Hệ tọa độ cục bộ không thay thế, ***chỉ bổ sung cho***, hệ tọa độ chính.

Hệ tọa độ cục bộ là tập hợp con của sự bù chi tiết hiện hành, được lập trình chỉ khi đã chọn sự bù chi tiết tiêu chuẩn hoặc bổ sung. Có nhiều ứng dụng có thể tận dụng ưu thế của tính năng điều khiển này.

Lệnh G52

Hệ tọa độ cục bộ là gì và hệ đó vận hành như thế nào? Về hình thức, có thể coi đây là hệ thống các tọa độ *liên hệ* với sự bù chi tiết đang có hiệu lực. Hệ này được lập trình bằng lệnh chuẩn bị G52.

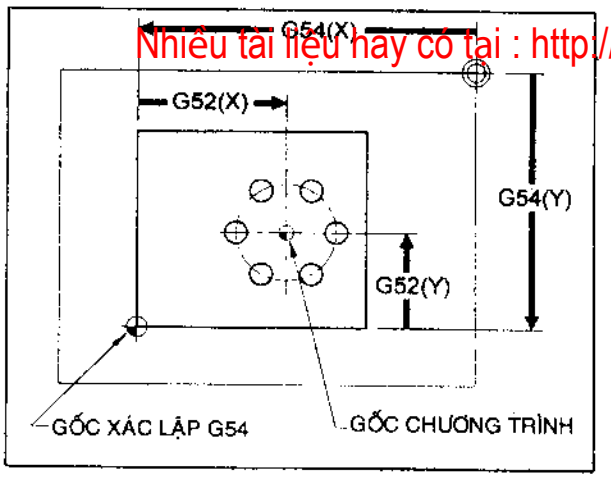
G52	Hệ tọa độ cục bộ

Lệnh G52 luôn luôn được bổ sung bằng các tọa độ chi tiết thực đã biết, xác lập zero chương trình mới – *có tính tạm thời* – như minh họa trên Hình 39.4.

Minh họa này hiển thị đường tròn bulông

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 39.4. Định nghĩa hệ tọa độ cục bộ sử dụng lệnh G52

gồm 6 lỗ trên tấm chữ nhật. Zero chương trình thông dụng thường là cạnh dưới của tấm phôi và tâm đường tròn bulông cách cạnh đó X8.0 và Y3.0 inch, sẽ trở thành giá trị dịch chuyển G52. Đường tròn bulông có đường kính ∅4.500 inch và lỗ thứ nhất ở vị trí 00 trên đường tròn đó. Các lỗ kế tiếp được gia công ngược chiều đồng hồ (CCW) theo thứ tự 2, 3, 4, 5, và 6.

Chương trình sẽ tạm thời chuyển zero (chuẩn) chi tiết từ cạnh dưới của tấm phôi đến tâm đường tròn bulông, *trong chương trình*. Với minh họa này bạn hãy theo dõi các block chương trình, trong quan hệ với đường tròn bulông và thứ tự logic chúng xuất hiện trong chương trình:

```
G90 G54 G00 X8.0 Y3.0     (BOLT CIRCLE CENTER)
 (--WORK COORDINATE SYSTEM POSITION ------)

G52 X8.0 Y3.0
(-- NEW PROGRAM ZERO ESTABLISHED ---------)

(G81) X2.25 Y0 (HOLE 1 LOCATION FROM NEW ZERO)
(-- COORDINATES FROM NEW ZERO ------------)

G52 X0 Y0
(-- CANCEL LOCAL OFFSET AND RETURN TO G54 ---)
```

Lệnh G52 có tính chế độ sẽ hoạt động cho đến khi bị xóa trong chương trình. Để xóa hệ tọa độ cục bộ và trở về chế độ bù chi tiết hoạt động trước đó, điều cần thực hiện là lập trình các giá trị zero với G52:

```
G52 X0 Y0                    ...ví dụ cuối
```

Mọi chuyển động dao sau khi xóa hệ tọa độ cục bộ sẽ liên quan với bù chi tiết ban đầu, đã được chuyên biệt bằng cách chọn G54 ở phần đầu chương trình này.

Chương trình đường tròn bulông sử dụng các kỹ thuật nêu trên. Bạn hãy so sánh kiểu lập trình này với việc chọn góc dưới bên trái là zero (chuẩn) chi tiết duy nhất.

Trước hết, sai sót khả dĩ của người vận hành CNC trong khi gá lắp sẽ giảm đến mức thấp nhất. Quả vậy, người vận hành chỉ cần xác lập G54 ở góc dưới bên trái của tấm phôi, và không cần điều chỉnh bất kỳ theo tâm của đường tròn bulông. Sự lập trình cũng dễ dàng hơn, do các giá trị tọa độ của đường tròn bulông xuất phát từ tâm đường tròn đó, thay vì từ cạnh dưới của tấm phôi.

```
O3902 (G54 AND G52 EXAMPLE)
N1 G20
N2 G17 G40 G80 T01
N3 M06
N4 G90 G54 G00 X8.0 Y3.0 S1200 M03 T02 (CNTR)
N5 G43 Z1.0 H01 M08
N6 G52 X8.0 Y3.0     (TEMP PRG ZERO AT BC CNTR)
N7 G99 G82 R0.1 Z-0.2 P100 F10.0 L0 (NO HOLE)
N8 X2.25 Y0                              (HOLE 1)
N9 X1.125 Y1.9486                        (HOLE 2)
N10 X-1.125                              (HOLE 3)
N11 X-2.25 Y0                            (HOLE 4)
N12 X-1.125 Y-1.9486                     (HOLE 5)
N13 X1.125                               (HOLE 6)
N14 G80 Z1.0 M09
N15 G52 X0 Y0            (RETURN TO G54 SYSTEM)
N16 G28 Z1.0 M05
N17 M01

N18 T02
N19 M06
N20 (... Tiếp tục gia công...)
```

HỆ TỌA ĐỘ MÁY

Cho đến phần này, *hệ tọa độ chính* (các lệnh bù chi tiết G54 đến G59) và *hệ tọa độ cục bộ*, G52, đã được trình bày chi tiết. Chúng là các công cụ lập trình mạnh và rất hữu ích. Hệ điều khiển Fanuc còn có hệ tọa độ khác, ít thông dụng hơn, có thể gọi là *hệ tọa độ thứ ba*.

Sự lựa chọn hệ tọa độ này chỉ có thể thực hiện với các tọa độ máy và lệnh chuẩn bị G53.

G53	Hệ tọa độ máy.

Hệ tọa độ máy *luôn luôn* sử dụng các tọa độ được đo từ zero (chuẩn) máy làm dữ liệu nhập.

Thoạt nhìn, sự sử dụng hệ tọa độ đặc thù này dường như không có lợi. Trước khi kết luận, bạn hãy đánh giá các nguyên tắc của hệ tọa độ máy, một số ứng dụng sẽ trở nên rõ ràng hơn:

- ❑ Lệnh G53 chỉ có hiệu lực trong block chuyên biệt lệnh đó.
- ❑ Các tọa độ lập trình luôn luôn liên quan với vị trí zero (chuẩn) máy
- ❑ G53 chỉ được sử dụng trong chế độ tuyệt đối (G90)
- ❑ Hệ tọa độ chính hiện hành (bù chi tiết) không thể xóa bằng lệnh G53.
- ❑ Bù bán kính dao phải được xóa trước lệnh G53.

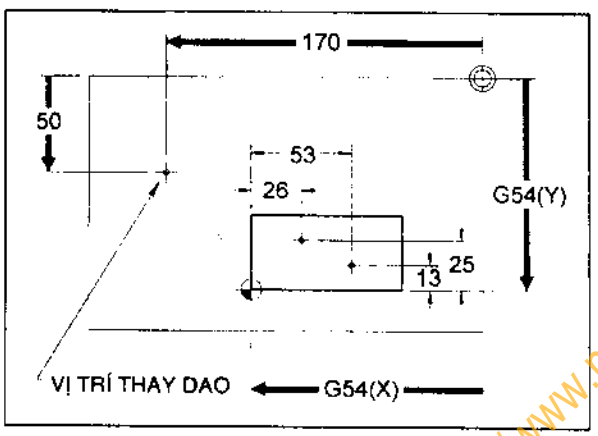
Từ các nguyên tắc này có thể thấy ít nhất là một công dụng khả dĩ của G53. Hệ tọa độ máy

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

có thể được dùng để báo đảm sự thay dao tại *cùng một vị trí bàn máy* mỗi khi sự thay dao được lập trình, bất kể chi tiết nào đang trên bàn máy và lệnh bù nào đang hoạt động. Điều này có thể áp dụng cho một chương trình hoặc là tiêu chuẩn cho mọi chương trình trên máy công cụ xác định. Bạn cần nhớ, vị trí thay dao luôn luôn được xác định bằng khoảng cách thực của dao từ vị trí zero (chuẩn) máy, *không phải* zero chương trình và cũng *không thể* từ vị trí khác. Trên nhiều máy, hoặc trong các gá lắp phức tạp, nên thiết lập vị trí thay dao cố định, bất kể vị trí của chi tiết gia công, chẳng hạn khi gia công với bàn xoay hoặc đồ gá cố định trên bàn máy.

Chương trình tiếp theo sẽ minh họa công dụng của lệnh G53, thực hiện sự thay dao tại vị trí cố định trên bàn máy, *không* quan hệ trực tiếp với chương trình hoặc chi tiết (Hình 39.5).



Hình 39.5. Hệ tọa độ máy G53 – chương trình O3903

```
O3903 (G53 COMMAND USAGE)
N1 G20
N2 G17 G40 G80 T01
N3 G91 G28 Z0
N4 G90 G53 G00 X-170.0 Y-50.0  (TOOL CHG POS)
N5 M06                         (ACTUAL TOOL CHANGE)
N6 G54 G00 X26.0 Y25.0 S1000 M03 T02
N7 G43 Z1.0 H01 M08
N8 G99 G82 R0.1 Z-0.2 P100 F8.0
N9 X53.0 Y13.0
N10 G80 G28 Z1.0 M05
N11 G53 G00 X-170.0 Y-50.0 (TOOL CHANGE POS)
N12 M01

N13 T02
N14 M06                        (ACTUAL TOOL CHANGE)
M15 G90 G54 G00 X53.0 Y13.0 S780 M03 T03
N16 G43 Z1.0 H02 M08
N17 G99 G81 R0.1 Z-0.836 F12.0
N18 X26.0 Y25.0
N19 G80 G28 Z1.0 M05
N20 G53 G00 X-170.0 Y-50.0 (TOOL CHANGE POS)
N21 M01

N22 T03
N23 M06                        (ACTUAL TOOL CHANGE)
...
%
```

Nguyên tắc thứ tư nêu trên, khẳng định bù chi tiết hiện hành *không* bị xóa bằng lệnh hệ tọa độ máy. Do ví dụ lập trình O3903 không minh họa tình huống này, thứ tự các chuyển động dao dưới đây (không liên quan với O3903) sẽ cho thấy sự độc lập giữa G53 và G54:

```
N1 G21                                  (METRIC)
...
N250 G90 G54 G00 X17.7 Y35.3
N251 G01 Z-5.0 F200.0
N252 G00 Z500.0
N253 G53 X-400.0 Y-100.0     (FIXED POSITION)
N254 M00                  (MANUAL TOOL CHANGE)

N255 S1200 M03
N256 X50.0 Y35.0    (IN ORIGINAL WORK OFFSET)
N257          (tiếp tục gia công...)
```

Thứ tự gia công trong chương trình rất đơn giản. Dao chuyển động đến vị trí XY của chi tiết trong N250, thực hiện sự gia công theo yêu cầu, chẳng hạn khoan, trong N251, chuyển động nhanh đến vị trí Z trống trong N252, *sau đó* đến vị trí thay dao cố định trong N253. Trong block kế tiếp, người vận hành CNC thay dao bằng tay, block N254, sau đó khôi phục tốc độ trục chính và trong N255. Trong block N256 chỉ chuyên biệt các tốc độ X và Y. Mọi giá trị khác đều là mặc định, kể cả lệnh bù G54. Block N256 có ý nghĩa đầy đủ như sau:

```
N256 G90 G54 G00 X50.0 Y35.0
```

Theo kinh nghiệm, bạn nên luôn luôn lập trình block hoàn chỉnh chứa mọi thông tin xác lập, và thực hiện điều đó mỗi khi gọi dao mới.

XÁC LẬP DỮ LIỆU

Trong các xưởng cơ khí nhỏ hoặc trung bình, nơi có các máy CNC độc lập, người vận hành máy thường xác lập mọi giá trị bù để nhập vào hệ thống CNC trong khi gá lắp, định vị dao và chi tiết. Phương pháp này rất hữu ích khi nhà lập trình CNC không biết các giá trị xác lập – các giá trị thực tế – của các chế độ bù vào thời điểm viết chương trình.

Trong môi trường sản xuất linh hoạt hoặc sản xuất lớn, phương pháp này không kinh tế và không hiệu quả. Sự sản xuất lớn sử dụng công nghệ hiện đại, chẳng hạn CAD/CAM để thiết kế và tính toán quỹ đạo dụng cụ cắt, sản xuất theo module, robot, xác lập dao trước, thay dao tự động, đồ gá lập trình, dây chuyền tự động, ... Trong môi trường đó, không thể có các phần tử chưa biết – luôn luôn biết quan hệ của mọi vị trí quy chiếu, không cần tìm các giá trị bù, cũng không cần xác lập giá trị bù cho từng máy riêng rẽ. Nhà lập trình phải biết mọi giá trị bù trước khi gá lắp dụng cụ và xác lập máy.

Biết các thông tin đó sẽ cho phép nhà lập

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

trình đưa dữ liệu bù vào chương trình và chuyển đến các bộ ghi (đăng ký) thích hợp thông qua dòng chương trình, không cần sự can thiệp của người vận hành, sự gia công được tự động hóa hoàn toàn, kể cả bảo trì dụng cụ cắt và các chế độ bù liên quan. Mọi chế độ bù đều được chương trình liên tục kiểm soát, *kể cả cập nhật các giá trị bù* được yêu cầu khi thay đổi vị trí, thay đổi bán kính và chiều dài dao.

Sự tự động hóa này được thực hiện thông qua tính năng điều khiển tùy chọn *Xác lập Dữ liệu*. Nhiều bộ điều khiển có tính năng này, ngay cả xưởng cơ khí chỉ có một máy CNC độc lập cũng có thể áp dụng tốt tính năng đó.

Lệnh xác lập dữ liệu

Để chọn tùy chọn xác lập dữ liệu và xác lập dữ liệu bù thông qua chương trình, bạn có thể sử dụng lệnh G cơ bản của Fanuc:

G10	Xác lập dữ liệu

Ở dạng cơ bản, G10 là lệnh *không chế độ*, chỉ có hiệu lực trong block được lập trình. Nếu cần lệnh này trong các block kế tiếp, phải lặp lại trong block đó.

Lệnh G10 có định dạng đơn giản, khác nhau giữa trung tâm gia công và máy tiện CNC, thậm chí có thể có các khác biệt định dạng trên các kiểu bộ điều khiển Fanuc khác nhau, dù các phương pháp lập trình về logic là như nhau. Các định dạng còn khác nhau đối với các chế độ bù, chẳng hạn giữa chế độ bù chi tiết và bù chiều dài dao.

Các ví dụ trong phần này chỉ áp dụng trên các bộ điều khiển Fanuc phổ biến, đã được kiểm nghiệm trên Fanuc 16 Model B, bộ điều khiển tiện và phay thông dụng.

Chế độ tọa độ

Sự lựa chọn chế độ lập trình số gia hay tuyệt đối có ảnh hưởng lớn đến sự nhập các giá trị bù trong suốt chương trình CNC. Bất kể kiểu bù được nhập với lệnh G10, giá trị bù lập trình sẽ *thay thế* giá trị bù hiện hành được lưu trong bộ điều khiển, nếu chương trình trong chế độ tuyệt đối (G90 đối với điều khiển phay và XZ đối với điều khiển tiện).

Trong chế độ số gia G91 đối với điều khiển phay và các trục UW đối với tiện, giá trị bù được lập trình *không thay thế* mà chỉ *cập nhật* giá trị bù lưu trong bộ điều khiển đó:

G90 với G10 = Giá trị bù sẽ được THAY THẾ

G91 với G10 = Giá trị bù sẽ được CẬP NHẬT

G90 hoặc G91 có thể được xác lập ở nơi bất kỳ trong chương trình, miễn là block chứa lệnh đã chọn đó được gán *trước* khi gọi lệnh xác lập dữ liệu G10.

Mọi kiểu bù khả dụng đều có thể được xác lập trong chương trình, sử dụng lệnh G10:

- Bù chi tiết G54 đến G59 và G54.1 P..
- Bù chiều dài dao G43 và G44.
- Bù bán kính dao G41 và G42.

Nhóm này gồm tất cả các chế độ bù liên quan, nếu khả dụng.

BÙ CHI TIẾT

Trước khi nghiên cứu phần này, bạn hãy xem lại Chương 17 về các khái niệm bù chi tiết.

Nhập bù chi tiết tiêu chuẩn

Sáu lệnh bù chi tiết tiêu chuẩn G54 đến G59 khả dụng trên cả bộ điều khiển phay và tiện. Do các yêu cầu gia công, chúng thường liên quan với điều khiển phay. Định dạng lập trình là như nhau:

```
G10 L2 P.. X.. Y.. Z..    Trung tâm gia công = phay
G10 L2 P.. X.. Z..        Trung tâm tiện = tiện
```

L2 là *chỉ số nhóm bù cố định* cho biết mục nhập là xác lập bù chi tiết. Địa chỉ P trong trường hợp này có thể có giá trị từ 1 đến 6, gán cho sự chọn G54 đến G50:

P1 = G54	P2 = G55	P3 = G56
P4 = G57	P5 = G58	P6 = G59

Ví dụ:

```
G90 G10 L2 P1 X-450.0 Y-375.0 Z0
```

Sẽ nhập các tọa độ X-450.0Y-375.0Z0 vào bộ đăng ký bù chi tiết G54 (mọi ví dụ trong phần này đều sử dụng đơn vị hệ mét).

L2 = Bù chi tiết tiêu chuẩn

Nhập bù chi tiết bổ sung

Ngoài 6 chế độ bù chi tiết tiêu chuẩn đối với điều khiển phay, Fanuc còn có các lệnh bù bổ sung G54.1 P1 đến G54.1 P48. Lệch G10 còn có thể được sử dụng để nhập các giá trị bù cho một trong 48 lệnh bù đó, với định dạng tương tự định dạng nêu trên:

```
G10 L20 P.. X.. Y.. Z..
```

Riêng chỉ số nhóm bù cố định được đổi thành L20, chỉ chọn lệnh bù chi tiết bổ sung.

L20 = Bù chi tiết bổ sung

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhập bù chi tiết bên ngoài

Sự bù kế tiếp thuộc hệ tọa độ chi tiết được gọi là bù *Ngoài* hoặc bù *Chung*. Chế độ bù này, không thể lập trình với mã G tiêu chuẩn bất kỳ, được dùng để cập nhật tổng quát mọi chế độ bù chi tiết, tác động đến tất cả các giá trị bù.

Để nhập giá trị bù vào bù bên ngoài, G10 sử dụng nhóm bù L2 và P0 để chọn bù bên ngoài:

```
G90 G10 L2 P0 X-10.0
```

sẽ đưa X-10.0 vào bù chi tiết bên ngoài, nhưng vẫn duy trì mọi xác lập khác (trục Y, trục Z, và trục bổ sung bất kỳ). Trong thực tiễn, khi sử dụng xác lập đã nêu, *từng* giá trị bù chi tiết trong chương trình sẽ dịch chuyển 10 mm theo chiều âm trên trục X.

BÙ CHIỀU DÀI DAO

Giá trị bù chiều dài dao đối với điều khiển phay có thể lập trình với lệnh G10 kết hợp nhóm bù. Tùy theo *kiểu* bộ nhớ điều khiển, nhóm bù L sẽ có các ý nghĩa khác nhau.

Có ba kiểu nhớ trên các bộ điều khiển Fanuc đối với bù chiều dài và bù bán kính dao:

Bộ nhớ A - chỉ một cột cho bù chiều dài dao

Nhập: Kết hợp dạng hình học + bù mòn dao
Giá trị: Xác lập giá trị bằng block G10 L11 P.. R..

Bộ nhớ B: hai cột cho bù dao

Nhập 1: Giá trị bù hình học riêng rẽ
Giá trị 1: Xác lập giá trị bằng block G10 L10 P.. R..

Nhập 2: Giá trị bù mòn dao riêng rẽ
Giá trị 2: Xác lập giá trị bằng block G10 L11 P.. R..

Bộ nhớ C - hai cột cho bù dao và hai cột cho bù bán kính

Nhập 1:	Giá trị bù hình học riêng rẽ
Được dùng cho:	Mã bù M
Giá trị 1:	Xác lập giá trị bằng block G10 L10 P.. R..
Nhập 2:	Giá trị bù hình học riêng rẽ
Được dùng cho:	Mã bù D
Giá trị 2:	Xác lập giá trị bằng block G10 L12 P.. R..
Nhập 3:	Giá trị bù mòn dao riêng rẽ
Được dùng cho:	Mã bù H
Giá trị 3:	Xác lập giá trị bằng block G10 L11 P.. R..
Nhập 4:	Giá trị bù mòn dao riêng rẽ
Được dùng cho:	Mã bù D
Giá trị 4:	Xác lập giá trị bằng block G10 L13 P.. R..

Trong tất cả các trường hợp đó, số nhóm bù được Fanuc gán một cách tùy ý và địa chỉ P là số đăng ký bù trong hệ thống CNC. R là giá trị bù được xác lập trong số bù đã chọn. Các chế độ tuyệt đối và số gia có tác dụng như nhau đối với nhập lập trình bù chiều dài dao như đối với bù chi tiết.

Ví dụ đối với trung tâm gia công CNC, block dưới đây sẽ nhập giá trị âm 468 mm vào chỉ số đăng ký bù chiều dài dao 5 (năm):

```
G90 G10 L10 P5 R-468.0
```

Nếu cần điều chỉnh giá trị bù này để cắt với chiều sâu giảm 0.5mm đối với bù chiều dài dao 5, bạn hãy chuyển sang chế độ số gia và lập trình như sau:

```
G91 G10 L10 P5 R-0.5
```

Chú ý, G91 là chế độ số gia. Nếu hai block này được dùng theo thứ tự nêu trên, giá trị cuối cùng của chỉ số bù 5 sẽ là -467.5 mm.

Các bộ điều khiển Fanuc kiểu cũ sử dụng địa chỉ L1 thay vì L11. Các bộ điều khiển này không có bù mòn dao như mục nhập riêng rẽ. Để tương thích với các bộ điều khiển cũ, L1 được chấp nhận trên mọi bộ điều khiển hiện đại ở vị trí L11.

Khoảng nhập có hiệu lực

Trên hầu hết các trung tâm gia công CNC, khoảng giá trị bù chiều dài dao bị giới hạn:

±999.999 mm	Nhập bù hình học hệ mét
±99.999 inch	Nhập bù hình học hệ Anh
±99.999 mm	Nhập bù mòn dao hệ mét
±9.999 inch	Nhập bù mòn dao hệ Anh

Số chế độ bù khả dụng cũng có giới hạn, tùy theo kiểu bộ điều khiển. Số bù tối thiểu là 32. Một cách tùy chọn, hệ thống CNC có thể có 64, 99, 200, 400, chế độ bù (thậm chí nhiều hơn), hầu hết đều là tùy chọn đặc biệt.

BÙ BÁN KÍNH DAO

Đối với bộ nhớ bù kiểu C, có thể nhập giá trị bù bán kính dao (D) thông qua chương trình, sử dụng G10 với các nhóm bù L12 và L13:

```
G90 G10 L12 P7 R5.0
```

sẽ nhập giá trị bán kính 5.000 vào chỉ số đăng ký bù hình học bán kính dao 7.

```
G90 G10 L13 P7 R-0.03
```

sẽ nhập giá trị bán kính - 0.030 vào chỉ số đăng ký bù mòn bán kính dao 7.

Nếu chỉ cần *điều chỉnh* giá trị bù hiện hữu, bạn hãy dùng chế độ lập trình theo số gia. Ví dụ cuối về bù mòn dao sẽ được cập nhật bằng cách cộng thêm 0.010 mm:

```
G91 G10 L13 P7 R-0.01 (Xác lập mới là 0.02 mm)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Cần rất cẩn thận với chế độ G90 và G91 bạn phải khôi phục lại chế độ cho các phần kế tiếp của chương trình.

BÙ KHI TIỆN

Bù chiều dài dao không áp dụng cho bộ điều khiển tiện, do có cấu trúc bù khác. Có thể dùng lệnh G10 để xác lập dữ liệu cho điều khiển tiện, sử dụng định dạng dưới đây:

```
G10 P.. X(U).. Z(W).. R(C).. Q..
```

Địa chỉ P là số *bù hình học* hoặc chỉ số *bù mòn dao* cần xác lập. Các địa chỉ X, Z, và R là giá trị tuyệt đối, còn U, W, C là các đương lượng số gia tương ứng của chúng. Không dùng chế độ G90 hoặc 91, chỉ sử dụng các mã G thuộc *Nhóm A*.

Để phân biệt giữa bù hình học và bù mòn dao, chỉ số bù hình học phải tăng theo giá trị tùy ý từ 10000:

P10001 sẽ là chỉ số bù hình học 1
P100012 sẽ là chỉ số bù hình học 12, v.v...

Nếu không có số 10000, địa chỉ P sẽ là chỉ số bù mòn dao.

Phần này sẽ nêu các ví dụ xác lập dữ liệu bù trên máy tiện CNC, theo thứ tự liên tiếp, dựa trên thứ tự nhập:

```
G10 P10001 X 0Z0 R0 Q0
```

... khử toàn bộ bù hình học đối với các xác lập *G 01*
(Đăng ký bù hình học 1)

```
G10 P1 X 0Z0 R0 Q0
```

... khử toàn bộ bù mòn dao đối với xác lập *W 01*
(Đăng ký bù mòn dao 1)

Chú ý - *Q0 cũng xóa giá trị chỉ số đỉnh dao trong G 01*

```
G10 P10001 X-Z00.0 Z-150 R0.8 Q3
```

... xác lập nội dung của bù hình học *G 01* theo:
`X-200.0 Z-150.0 R0.8 T3`

... đồng thời xác lập T3 trong bù mòn dao - *một cách tự động*

```
G10 P1 R0.8          Xác lập T hiện hành (mặc định)
```

...xác lập giá trị R0.8 trong bù mòn dao *W 01*

Chú ý, có thể sẽ an toàn hơn nếu lập trình:

```
G10 P1 R0.8 Q3  xác lập hiện hành không được giả định
G10 P1 X-0.12
```

... bù mòn dao W01 được xác lập là X-0.12, bất kể xác lập trước đó

```
G10 P1 U0.05
```

... cập nhật X-0.12 với + 0.05 để có giá trị mới là X-0.07

Chú ý, chỉ số mũi dao (được lập trình trong ứng dụng G10 theo mục nhập Q) sẽ luôn luôn thay đổi cả bù hình học và bù mòn dao, bất kể kiểu hoặc giá trị bù. Lý do là sự an toàn được thiết lập bên trong bộ điều khiển nhằm loại bỏ sai sót nhập dữ liệu.

XÁC LẬP DỮ LIỆU MDI

Lập trình các giá trị bù thông qua xác lập chương trình đòi hỏi hiểu rõ định dạng nhập đối với hệ điều khiển cụ thể. Sẽ là quá trễ khi xác lập sai gây ra hư hại máy hoặc chi tiết gia công.

Một phương pháp có thể sử dụng để bảo đảm xác lập dữ liệu bù là đúng, là sự kiểm tra đơn giản. Kiểm tra các mục nhập G10 trong chế độ MDI trên CNC trước, sau đó kiểm lại kết quả:

- ❑ Xác lập chế độ Program
- ❑ Xác lập chế độ MDI
- ❑ Chèn dữ liệu kiểm tra

Ví dụ, nhập:

```
G90 G10 L10 P12 R-106.475
```

- ❑ Nhấn INSERT
- ❑ Nhấn CYCLE START

Để kiểm nghiệm, bạn hãy kiểm tra bù chiều dài dao H12 - giá trị lưu sẽ phải là -106.475

Trong khi vẫn trong chế độ MDI, chèn dữ liệu kiểm tra khác, ví dụ:

```
G91 G10 L10 P12 R-1.0
```

- ❑ Nhấn INSERT
- ❑ Nhấn CYCLE START

Để kiểm nghiệm, kiểm tra xác lập bù chiều dài dao H12 - giá trị mới sẽ phải là -107.475.

Bạn hãy tiếp tục kiểm tra theo quy trình nêu trên, tuy tốn thêm thời gian nhưng sẽ chắc chắn hơn.

NHẬP THAM SỐ LẬP TRÌNH

Phần này trình bày đặc tính thứ hai khi lập trình lệnh G10-trong đó lệnh G10 có *tính chế độ*, được dùng để thay đổi tham số hệ thống, thông qua chương trình. Lệnh này đôi khi được gọi là "*Viết hàm tham số*", ít dùng trong lập trình hàng ngày. Điều quan trọng là hiểu khái niệm tham số điều khiển hệ thống để có thể sử dụng tốt lệnh chế độ G10.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

CẢNH BÁO
Xác lập sai các tham số hệ thống có thể làm cho máy CNC bị hư đến mức không thể sửa chữa.

Công dụng của lệnh này là thay đổi điều kiện gia công, ví dụ các hằng thời gian trục chính và tốc độ cắt, dữ liệu bù sai số bước,... Lệnh này thường xuất hiện trong *User Macros* (được áp dụng bằng lệnh G65) và mục đích là điều khiển một số thao tác máy.

Lệnh chế độ G10

Khi lệnh G10 được dùng cho xác lập dữ liệu bù, cần lặp lại trong từng block. G10 đối với mục nhập bù chỉ có thể sử dụng theo lệnh *không chế độ*. Các bộ điều khiển Fanuc hiện đại còn cho phép một kiểu thay đổi thông qua chương trình - thay đổi *tham số hệ thống CNC* thông qua lệnh chế độ G10.

Nhiều mục nhập trong chương trình sẽ được bộ điều khiển tự động chuyển thành tham số hệ thống. Ví dụ, lập trình G54, giá trị xác lập hiển thị trên màn hình bù chi tiết. Sự lưu thực tế giá trị G54 xảy ra trong tham số hệ thống, được nhận biết bằng chỉ số tham số xác định. Xác lập G54 có thể thay đổi thông qua dữ liệu bù hoặc thông qua thay đổi tham số, nhưng phải biết chỉ số của tham số đó. Một số tham số hệ thống không thể thay đổi một cách dễ dàng (số khác hoàn toàn không thể thay đổi), do đó lệnh chế độ G10 rất hữu ích. Cần có hai lệnh liên quan với nhau. G10 để khởi đầu xác lập và G11 để xóa xác lập đó:

```
G10 L50
(... xác lập dữ liệu...)
G11
```

Block xác lập dữ liệu có ba mục nhập:

```
G10 L50
.. P.. R..
G11
```

Trong trường hợp kết hợp lệnh chế độ G10 và G11, các lệnh này có ý nghĩa như sau:

G10	Chế độ xác lập dữ liệu
L50	Chế độ cố định nhập tham số lập trình
.. P.. R..	Chuyên biệt mục nhập dữ liệu
G11	Xóa chế độ xác lập dữ liệu

Giữa block G10 L50 và block G11 là danh sách tham số hệ thống cần xác lập, mỗi block một tham số. Chỉ số của tham số sử dụng địa chỉ N, dữ liệu sử dụng các địa chỉ P và R có vài kiểu nhập tham số hệ thống:

Kiểu nhập tham số	Khoảng nhập cho phép
Kiểu bit	0 hoặc 1
Kiểu trục bit	0 hoặc 1
Kiểu byte	0 đến +127
Kiểu trục byte	0 đến 255
Kiểu từ ngữ	0 đến ±32767
Kiểu trục từ ngữ	0 đến ±32767
Kiểu hai từ ngữ	0 đến +99999999
Kiểu trục hai từ ngữ	0 đến ±99999999

Bạn hãy chú ý các tham số kiểu bit - số dữ liệu đơn luôn luôn được gán 8 bit. Mỗi bit đều có ý nghĩa riêng, do đó cần rất cẩn thận khi thay đổi một bit mà không thay đổi bit khác. Kiểu từ ngữ còn được gọi là *kiểu số nguyên* và kiểu hai từ ngữ là *kiểu số nguyên dài*.

Ghi tham số

Đánh số các tham số kiểu bit và kiểu trục bit là tiêu chuẩn từ 0 đến 7 (*máy tính bắt đầu đếm từ zero thay vì từ một*), từ phải sang trái:

Số	#7	#6	#5	#4	#3	#2	#1	#0

trong đó *Số* là chỉ số tham số bốn chữ số và #7 đến #0 là các vị trí bit riêng rẽ - *chú ý thứ tự đánh số và phương pháp đếm*. Các tham số không phải kiểu bit được nhập theo mục nhập byte, từ ngữ, hoặc hai từ ngữ (các phiên bản có trục và không có trục).

Địa chỉ P

Địa chỉ P chỉ được dùng cho các tham số liên quan với trục (trục bit, trục byte, từ ngữ trục, hai từ ngữ trục). Nếu tham số không liên quan với trục, địa chỉ P là dư và không cần lập trình. Nếu cần hơn một trục để xác lập, bạn hãy dùng nhiều mục nhập .. P.. R .. giữa G10 và G11.

Địa chỉ R

Địa chỉ R là giá trị mới cần được đăng ký trong chỉ số tham số chọn và luôn luôn phải nhập. Khoảng có hiệu lực nêu trên phải được bảo đảm. Chú ý, không có dấu thập phân trong mục nhập này.

Sự chuyển đổi chương trình

Các chương trình, dù chỉ chứa một mục nhập tham số lập trình, chỉ được sử dụng với máy và bộ điều khiển mà chương trình đó được thiết kế.

Cần rất cẩn thận khi chương trình đã chỉnh sửa các tham số hệ thống được sử dụng trên nhiều máy.

Các chỉ số tham số và ý nghĩa của chúng trên các kiểu bộ điều khiển khác nhau có thể không như nhau. Cần biết rõ kiểu bộ điều khiển và các chỉ số tham số trong khi lập trình.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ví dụ, trên bộ điều khiển Fanuc Model 15, tham số điều khiển ý nghĩa của địa chỉ không có dấu thập phân là số 2400 (Bit #0). Tham số điều khiển xác lập này trên Fanuc Model 16 là số 3401 (Bit #0).

Các ví dụ dưới đây sẽ minh họa mục nhập tham số lập trình đã được kiểm tra trên bộ điều khiển Fanuc 16 Model B-phiên bản tiện và phay. Các tham số đã chọn chỉ dùng để minh họa. *Không nên kiểm tra các tham số này trên máy.*

Ví dụ thứ nhất thay đổi xác lập tốc độ baud của thiết bị Input/Output với giao diện RS-232, nếu *Kênh I/O* được xác lập là 0:

```
G10 L50
N0103 R10
G11
```

Tham số điều khiển xác lập tốc độ baud đối với thiết bị này có chỉ số #103. Từ bảng do Fanuc cung cấp, có thể nhập giá trị R:

Xác lập giá trị - R	Tốc độ
1	50 baud
2	100 baud
3	110 baud
4	150 baud
5	200 baud
6	300 baud
7	600 baud
8	1200 baud
9	2400 baud
10	4800 baud
11	9600 baud
12	19200 baud

Trong ví dụ nêu trên,

```
G10 L50
N0103 R10
G11
```

tốc độ baud được chọn là 4800 ký tự /giây

Trong ví dụ thứ hai, tham số #5130 điều khiển khoảng cách vạt góc đối với các chu kỳ gia công ren G92 và G76 (khoảng cách lùi dao từ từ chỉ có thể áp dụng trên bộ điều khiển tiện). Kiểu dữ liệu là byte không có trục, đơn vị của dữ liệu là 0.1 trước ren và khoảng giá trị là 0-127:

```
G10 L50
N5103 R1
G11
```

Đoạn chương trình này sẽ thay đổi tham số #5130 sang giá trị 1. Lượng vạt góc sẽ tương đương một bước ren. Bạn không nên nhầm lẫn giữa bit và byte, trong đó byte là giá trị 0 đến 127 hoặc 0 đến 255 đối với kiểu byte trục, bit chỉ là trạng thái (0 hoặc 1, OFF hoặc ON,....), cho phép chọn chỉ một trong hai tùy chọn khả dụng. Thuật ngữ BIT là viết tắt của hai từ:

Bit = Binary digit (*chữ số nhị phân*)

Ví dụ thứ ba là mục nhập kiểu tham số hai từ ngữ, đổi bù chi tiết G54 sang X-250.000.

```
G90
G10 L50
N1221 P1 R-250000
G11
```

Tham số #1221 điều khiển G54, #1222 điều khiển G55, v.v... P1 quy chiếu trục X, P2 quy chiếu trục Y, đến 8 trục. Do cần khoảng hiệu lực của số nguyên dài (kiểu hai từ ngữ), không thể sử dụng dấu thập phân. Do xác lập theo hệ mét, một micron (0.001mm) là số gia nhỏ nhất, giá trị - 250.000 sẽ được nhập là 250000. ví dụ dưới đây là KHÔNG chính xác và sẽ dẫn đến lỗi:

```
G90
G10 L50
N1221 P1 R-250.0    (không được dùng dấu thập phân)
G11
```

Mục nhập đúng phải *không có* dấu thập phân. Điều kiện lỗi cũng sẽ phát sinh nếu địa chỉ P không được chuyên biệt. Ví dụ:

```
G90
G10 L50
N1221 R-250000
G11
```

sẽ đưa đến trạng thái lỗi. Ví dụ tiếp theo được thay đổi để nhập hai trục:

```
G90
G10 L50
N1221 P1 R-250000
N1221 P2 R-175000
G11
```

Nếu ví dụ này được dùng trên bộ điều khiển tiện, P1 là trục X, P2 là trục Z. Trên trung tâm gia công, P1 là trục X, P2 là trục Y, và P3 là trục Z. Trong các trường hợp đó, hai trục đầu của xác lập bù chi tiết G54 sẽ là -250.000 và -175.000.

Đôi khi cần xác lập tất cả các trục theo zero. Điều này có thể được thực hiện với xác lập bù tiêu chuẩn:

```
G90 G10 L2 P1 X0 Y0 Z0          (Điều khiển phay)
```

hoặc viết tham số, cũng trên điều khiển phay:

```
G90
G10 L50
N1221 P1 R0       (Xác lập tọa độ X của G54 theo zero)
N1221 P2 R0       (Xác lập tọa độ Y của G54 theo zero)
N1221 P3 R0       (Xác lập tọa độ Z của G54 theo zero)
```

Tham số kiểu bit

Ví dụ kế tiếp là vô hại và có thể sử dụng để

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

kiểm tra, nhưng cần cẩn thận với các tham số khác. Mục đích duy nhất là xác lập ON cho chuỗi thứ tự block tự động trong khi nhập chương trình CNC tại bộ điều khiển. Ví dụ này còn là minh họa tham số kiểu bit khi chuẩn bị lập trình sử dụng chế độ tham số có thể lập trình.

Trên Fanuc 16 Model B (và hầu hết các model khác) có tính năng cho phép nhập tự động chuỗi các số, nếu chương trình được nhập từ bàn phím. Tính năng này nhằm tiết kiệm thời gian khi nhập dữ liệu bằng tay. Để kích hoạt tính năng đó, bạn hãy chọn tham số điều khiển trạng thái ON và OFF của tính năng. Trên Fanuc 16, đây là chỉ số tham số 0000. Đây là tham số kiểu bit, nghĩa là có chứa 8 bit. Mỗi bit đều có ý nghĩa riêng. Bit #5 (SEQ) điều khiển trạng thái của chuỗi đánh số tự động (ON hoặc OFF, tương ứng 1 hoặc 0, nhưng chỉ có thể nhập một số). Không thể lập trình từng bit riêng rẽ, chỉ được nhập số dữ liệu đơn gồm tất cả tám bit. Điều đó có nghĩa là phải biết tất cả các bit còn lại để thay đổi một bit. Trong ví dụ này, xác lập hiện hành của tham số 0 sẽ như sau:

			SEQ			INI	ISO	TVC
0000	#7	#6	#5	#4	#3	#2	#1	#0
	0	0	0	0	1	0	1	0

Ý nghĩa cụ thể của các tham số khác là chưa rõ ràng trong ví dụ này. Bit #5 được xác lập là 0, có nghĩa là xóa sự đánh số block tự động.

Đoạn chương trình dưới đây sẽ kích hoạt bit #5, mà không thay đổi các bit khác:

```
G10 L50
N0 R00101010
G11
```

Mục nhập trên màn hình tham số sẽ phản ánh thay đổi đó:

			SEQ			INI	ISO	TVC
0000	#7	#6	#5	#4	#3	#2	#1	#0
	0	0	1	0	1	0	1	0

Chú ý, tất cả các bit đều được viết. Tuy nhiên, điều này là không cần thiết. Fanuc đưa ra tính năng bổ sung - có thể chọn *số gia* để đánh số, ví dụ, chọn 10 sẽ sử dụng các mục nhập N10, N20, N30, chọn 1 sẽ sử dụng N1, N2, N3,... Ví dụ, chọn số gia là năm, sẽ có N5, N10, N15,.... Số gia có thể được xác lập bằng chỉ số tham số khác. Trên Fanuc 16, chỉ số tham số chứa giá trị đánh số tự động và # 3216. Đây là tham số kiểu từ ngữ có khoảng hiệu lực là 0 đến 9999. Chỉ có thể kích hoạt tham số này bằng xác lập bit #5 trong tham số 0000 là 1. Đoạn chương trình sẽ như sau:

```
G10 L50
N3216 R5
G11
```

Sau khi hoàn tất các xác lập này, sẽ không cần nhập các số block trong chương trình bất kỳ được nhập thông qua bàn phím bộ điều khiển. Thời điểm bất kỳ, nếu nhấn phím EOB (End-of-block, kết thúc block), số N sẽ xuất hiện trên màn hình một cách tự động, theo số gia là năm, tiết kiệm thời gian bấm phím trong khi nhập chương trình bằng tay.

Ý tưởng đằng sau G10 có tính chế độ trong chế độ nhập tham số lập trình là vài tham số có thể xác lập theo nhóm. Do hai tham số có quan hệ logic với nhau, có thể tạo ra một đoạn chương trình với cùng các kết quả như hai đoạn chương trình nhỏ đã nêu. Lệnh chế độ G10 sẽ như sau:

```
G10 L50
N0000 R00101010
N3216 R5
G11
```

Do không có tham số kiểu trục, có thể bỏ qua địa chỉ P. N0000 là hoàn toàn như N0, được sử dụng chỉ để tăng tính rõ ràng.

Những người dùng Fanuc 15 cần lưu ý (hệ Fanuc 15 *cao hơn* Fanuc 16) - chỉ số tham số chọn kích hoạt đánh số chuỗi thứ tự một cách tự động là 0010, bit #1 (SQN). Trên Fanuc tính linh hoạt cao hơn - sự khởi động số chuỗi thứ tự có thể được điều khiển với tham số # 0031, và chỉ số tham số lưu giá trị gia là # 0032, với cùng phong cách mục nhập chương trình như đã nêu. Ngoài ra, trên Fanuc 15, khoảng các số thứ tự được phép đến 99999. Đây là ví dụ về sự khác biệt giữa Fanuc 15 và Fanuc 16, dù chúng là sản phẩm của cùng một nhà chế tạo.

Tác dụng của các số block

Nhiều chương trình có các số block. Hoàn toàn có thể gán các số block cho ví dụ nêu trên:

```
...
N121 G10 L50
N122 N0000 R00101010
N123 N3216 R5
N124 G11
...
```

Đoạn chương trình này hoạt động như thế nào? Lúc này có *hai* địa chỉ N khác nhau trong các block N122 và N123. Bộ điều khiển xử lý tình huống này ra sao? Ở đây hoàn toàn không có mâu thuẫn. Trong trường hợp hai địa chỉ N trong một block *bên trong đoạn G10 đến G11*, địa chỉ N thứ nhất là số block, N thứ hai trong cùng một block sẽ được diễn dịch là chỉ số tham số.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

40 HÌNH ẢNH GƯƠNG

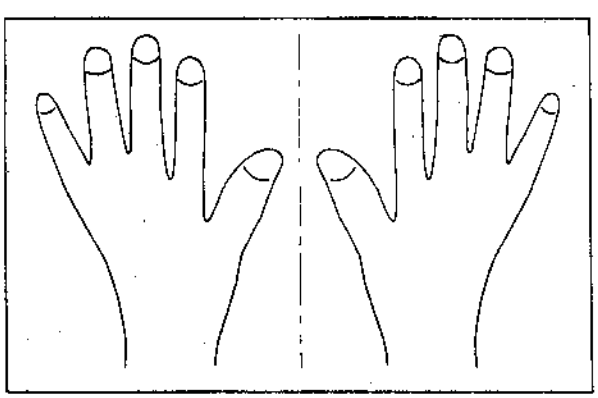
Mục đích chính của phát triển chương trình CNC là tạo ra quỹ đạo dao trong vị trí cụ thể của chi tiết hoặc máy. Nếu quỹ đạo dao yêu cầu cả chiều trái và chiều phải, thời gian lập trình có thể rút ngắn bằng cách dùng tính năng *hình ảnh gương*.

Chuỗi thứ tự gia công bất kỳ đều có thể lặp lại một cách đối xứng bằng cách sử dụng tính năng *hình ảnh gương* của hệ điều khiển. Không cần các tính toán mới, do đó kỹ thuật lập trình này sẽ giảm thời gian lập trình và giảm khả năng xảy ra các sai sót. Hình ảnh (đối xứng) gương đôi khi còn được gọi là hàm *Đảo ngược trục*, nhưng điều này không hoàn toàn đúng. Mặc dầu trong chế độ hình ảnh gương, các trục máy sẽ được đảo ngược, nhưng còn có vài thay đổi khác, do đó thuật ngữ *Hình ảnh gương* có tính chính xác cao hơn. Nếu bạn thông thạo hệ thống CAD, bạn sẽ thấy hàm hình ảnh gương trong CNC cũng dựa trên các nguyên tắc như trong CAD.

Hình ảnh gương dựa trên nguyên tắc các chi tiết đối xứng, đôi khi được gọi là chi tiết *Phải* (R/H) và chi tiết *Trái* (L/H) (Hình 40.1).

Lập trình hình ảnh gương yêu cầu kiến thức về hệ tọa độ vuônggóc cơ bản, đặc biệt là cách thức áp dụng các góc phần tư, nội suy đường tròn và các ứng dụng bù bán kính dao.

Hệ tọa độ vuông góc có bốn góc phần tư trên mặt phẳng. Vùng trên bên phải là *Góc phần tư I*, trên bên trái là *Góc phần tư II*, dưới bên trái là *Góc phần tư III*, vùng dưới bên phải là *Góc phần tư IV*. Nếu zero chương trình ở góc dưới bên trái của chi tiết, bạn đang lập trình trong góc phần tư thứ nhất.

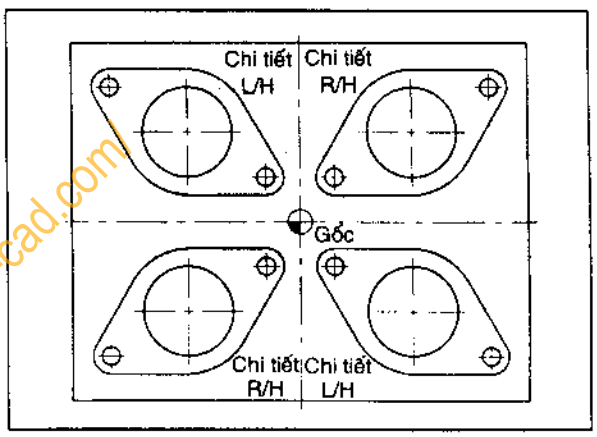


Hình 40.1. *Tay phải và tay trái theo nguyên tắc đối xứng gương.*

NGUYÊN TẮC CƠ BẢN CỦA HÌNH ẢNH GƯƠNG

Nguyên tắc cơ bản của hình ảnh gương dựa trên nhận xét sự gia công với quỹ đạo dao đã cho trong một góc phần tư hầu như không khác biệt khi gia công với cùng quỹ đạo dao đó trong góc phần tư khác. Khác biệt chính là *sự đảo ngược* của một số chiều chuyển động. Điều đó có nghĩa là chi tiết được gia công trong một góc phần tư có thể lặp lại trong góc phần tư khác *sử dụng cùng một chương trình với hàm hình ảnh gương có hiệu lực*.

Nguyên tắc chiều *Phải* và chiều *Trái* có thể được áp dụng cho chiều chi tiết gia công (Hình 40.2).



Hình 40.2. *Nguyên tắc chiều phải và chiều trái áp dụng cho chi tiết gia công.*

Bạn cần nhớ, mỗi góc phần tư đều có các dấu trục khác nhau. Hàm hình ảnh gương cho phép sự đảo các trục và các thay đổi chiều khác có thể thực hiện một cách tự động.

Chiều quỹ đạo dao

Tùy theo góc phần tư được chọn cho hình ảnh gương, sự thay đổi chiều quỹ đạo dao có thể tác động đến *một số* hoặc *tất cả* các yếu tố sau:

- ❑ Dấu số học của trục (cộng hoặc trừ).
- ❑ Chiều phay (thuận hoặc ngược).
- ❑ Chiều chuyển động cung (CW hoặc CCW).

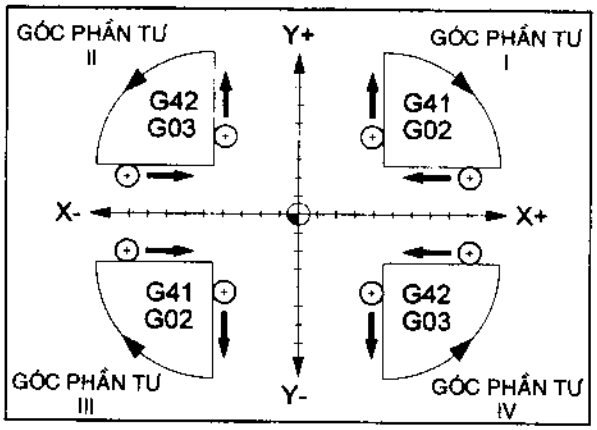
Một hoặc nhiều trục máy có thể bị tác động. Nói chung, các trục đó chỉ gồm trục X và Y. Trục Z thường *không* được dùng trong các ứng dụng hình ảnh gương.

Không phải tất cả các yếu tố nêu trên đều

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

bị ảnh hưởng đồng thời. Nếu không có sự nội suy đường tròn, không cần xét chiều của cung tròn. Hình 40.3 minh họa tác động của hình ảnh gương đối với quỹ đạo dao trong cả bốn góc phần tư.



Hình 40.3. *Tác động của hình ảnh gương đối với quỹ đạo dao trong các góc phần tư*

Quỹ đạo dao gốc

Chương trình quỹ đạo dao gốc có thể được triển khai trong góc phần tư bất kỳ. Nếu không áp dụng hình ảnh gương (trạng thái mặc định), quỹ đạo dao chỉ gia công trong góc phần tư xác định. Đây là cách thức lập trình hầu hết các ứng dụng. Khi khởi động sự đối xứng gương, sẽ luôn luôn lấy chiếu gương sơ đồ gia công *gốc* - quỹ đạo dao gốc - bất kể quỹ đạo đó được xác định trong góc phần tư nào.

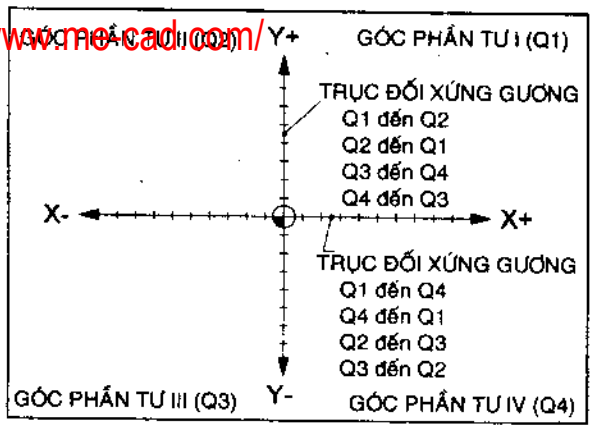
Sự chiếu gương sẽ luôn luôn truyền sơ đồ gia công (quỹ đạo dao) sang một hoặc vài góc phần tư khác. Đây là mục đích của hàm hình ảnh gương. Lập trình hình ảnh gương đòi hỏi đáp ứng một số điều kiện. Một trong các điều kiện đó là xác định trục gương (trục đối xứng).

Trục gương

Do có *bốn* góc phần tư, chúng sẽ cung cấp *bốn* vùng gia công. Các vùng này được phân chia bằng *hai* trục máy. Trụcđối xứng gương là trục máy qua đó mọi chuyển động lập trình sẽ được *chiếu* đối xứng. Hình 40.4 minh họa các trục gươngvà tác động của chúng đối với sự định hướng chi tiết trong các góc phần tư. Trục đối xướng gương có thể được xác định theo hai cách:

- ❑ Tại máy ... do người vận hành CNC thực hiện
- ❑ Thông qua chương trình ... do người lập trình CNC thực hiện

Các phương pháp trên cho phép chọn một trong các khả năng:



Hình 40.4. *Trục đối xứng gương và tác động của trục đối với chiều của chi tiết*

1. Gia công bình thường - không xác lập trình ảnh gương
2. Gia công đối xứng gương qua trục X
3. Gia công đối xứng gương qua trục Y
4. Gia công đối xứng gương qua trục X và Y

Hình ảnh gương lập trình phải được hệ điều khiển hỗ trợ

Gia công *bình thường* theo đúng chương trình đã viết. Ví dụ, nếu quỹ đạo lập trình xảy ra trong góc phần tư thứ hai (sử dụng chế độ tuyệt đối G90), các giá trị X bình thường sẽ là âm và các giá trị Y là dương. Dấu của các điểm tọa độ sẽ luôn luôn bình thương trong góc phần tư gốc đã lập trình, khi không sử dụng hình ảnh gương. Khi sự gia công xảy ra trong góc phần tư đối xứng gương, một hoặc cả hai dấu sẽ thay đổi.

Dấu của các tọa độ

Dấu "bình thường" tùy thuộc vào góc phần tư của hệ tọa độ được dùng trong lập trình. Nếu lập trình trong *Góc phần tư I*, cả X và Y đều có giá trị dương. Đây là danh sách các giá trị trong bốn góc phần tư.

Góc phần tư I	X+ Y+
Góc phần tư II	X- Y+
Góc phần tư III	X- Y-
Góc phần tư IV	X+ Y-

Chiếu gương quỹ đạo dao lập trình, hệ điều khiển sẽ *tạm thời* thay đổi một hoặc cả hai dấu, tùy theo trục đối xứng. Ví dụ, nếu chuyển động dao được lập trình trong Góc phần tư I (X+Y+), và được chiếu gương qua trục X, sẽ lấy các dấu của Góc phần tư IV (X+Y-). Trong thường hợp này, X là trục đối xứng. Ví dụ thứ hai, cũng dựa trên chương trình gốc trong Góc phần tư I, nhưng chiếu gương qua trục Y. Khi đó, các dấu

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tạm thời sẽ của Góc phần tư II (X-Y+). Nếu chiếu gương chương trình trong Góc phần tư I theo cả hai trục X và Y, chương trình sẽ được thực thi trong Góc phần tư III (X-Y-).

Chiều phay

Phay chu vi có thể được lập trình theo chế độ *phay thuận* hoặc *phay ngược*. Khi quan sát chuyển động dao gốc được xác định trong chế độ phay ngược trong *Góc phần tư I*, sự gia công chiếu gương trong các góc phần tư còn lại sẽ như sau:

- Chiếu gương vào góc phần tư II ... phay thuận
- Chiếu gương vào góc phần tư III ... phay ngược
- Chiếu gương vào góc phần tư IV ... phay thuận

Điều quan trọng là hiểu chế độ gia công khi sử dụng hình ảnh gương. Chế độ phay thuận có thể không đạt kết quả mong muốn, do không bảo đảm chất lượng bề mặt và dung sai kích thước.

Chiều chuyển động cung

Sự thay đổi thứ hai đối với quỹ đạo dao, sẽ chỉ xảy ra khi chiếu gương qua một trục, là chiều quay của cung. Cung thuận chiều kim đồng hồ bất kỳ được lập trình sẽ trở thành cung ngược chiều khi chiếu gương qua một trục, và ngược lại. Đây là kết quả của chiều chuyển động cung, dựa trên góc phần tư I:

- Góc phần tư I - cung gốc là CW

  Góc phần tư II - cắt CCW
  Góc phần tư III - cắt CW
  Góc phần tư IV - cắt CCW

- Góc phần tư I - cung gốc là CCW

  Góc phần tư II - cắt CW
  Góc phần tư III - cắt CCW
  Góc phần tư IV - cắt CW

Hệ điều khiển sẽ tự động thực thi G02 theo G03 và G03 theo G02 khi được yêu cầu. Đối với hầu hết các ứng dụng gia công, sự thay đổi chiều chuyển động cung sẽ không ảnh hưởng đến chất lượng gia công. Đối với cả chiều phay và chiều cung, bạn xem lại Hình 40.3

Khởi động và kết thúc chương trình

Khi chi tiết được lập trình với ý định sử dụng hình ảnh gương, bạn cần suy nghĩa cẩn thận về phương pháp lập trình, sử dụng kỹ thuật hơi khác so với lập trình trong một góc phần tư (không dùng hình ảnh gương). Trong khi chiếu gương, mọi chuyển động trong chuyển động, trừ sự trở về zero máy, sẽ được chiếu đối xứng, khi kích hoạt tính năng chiếu gương. Điều đó có nghĩa là, bạn cần xét các *yếu tố* dưới đây:

1. CÁCH THỨC khởi động chương trình
2. VỊ TRÍ áp dụng hình ảnh gương
3. THỜI ĐIỂM xóa hình ảnh gương

Khởi đầu và kết thúc chương trình có chiếu gương thường ở cùng vị trí, tại X0Y0 của chi tiết.

CHIẾU GƯƠNG BẰNG XÁC LẬP

Hình ảnh gương có thể được xác lập trên bộ điều khiển, không yêu cầu các mã đặc biệt. Chương trình tương đối ngắn, do chỉ chứa chuyển động dao trong một góc phần tư. Khi viết chương trình có chiếu gương cần lập kế hoạch và xây dựng cấu trúc chương trình một cách hợp lý.

Xác lập điều khiển

Hầu hết các bộ điều khiển đều có xác lập màn hình hoặc các công tắc chuyên dùng cho xác lập chiếu gương trên bộ điều khiển. Cả hai thiết kế này đều cho phép người vận hành dễ dàng xác lập một số tham số, mà không bị nguy cơ ghi đè lên các tham số khác. Trong trường hợp xác lập màn hình, sự hiển thị tương tự dưới đây:

```
MIRROR IMAGE X-AXIS = 0 (0:OFF 1:ON)
MIRROR IMAGE Y-AXIS = 0 (0:OFF 1:ON)
```

Đây là hiển thị mặc định, chiếu gương cả hai trục đều không hoạt động (chế độ xóa). Để chỉ áp dụng chiếu gương trục X, cần bảo đảm hiển thị:

```
MIRROR IMAGE X-AXIS = 1 (0:OFF 1:ON)
MIRROR IMAGE Y-AXIS = 0 (0:OFF 1:ON)
```

Để chỉ áp dụng chiếu gương trục Y, hiển thị sẽ là:

```
MIRROR IMAGE X-AXIS = 0 (0:OFF 1:ON)
MIRROR IMAGE Y-AXIS = 1 (0:OFF 1:ON)
```

Và cuối cùng, để chiếu gương đồng thời cả hai trục:

```
MIRROR IMAGE X-AXIS = 1 (0:OFF 1:ON)
MIRROR IMAGE Y-AXIS = 1 (0:OFF 1:ON)
```

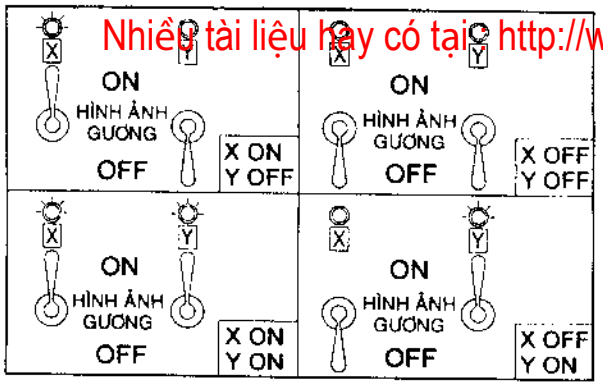
Để xóa hình ảnh gương và trở về chế độ chương trình bình thường, xác lập cho cả hai trục X và Y là zero:

```
MIRROR IMAGE X-AXIS = 0 (0:OFF 1:ON)
MIRROR IMAGE Y-AXIS = 0 (0:OFF 1:ON)
```

Hình 40.5 minh họa các xác lập hình ảnh gương sử dụng các công tắc chế độ ON/OFF. Hầu hết các máy đều có đèn báo, sẽ sáng (ON) đối với trục chiếu gương hiện hành.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

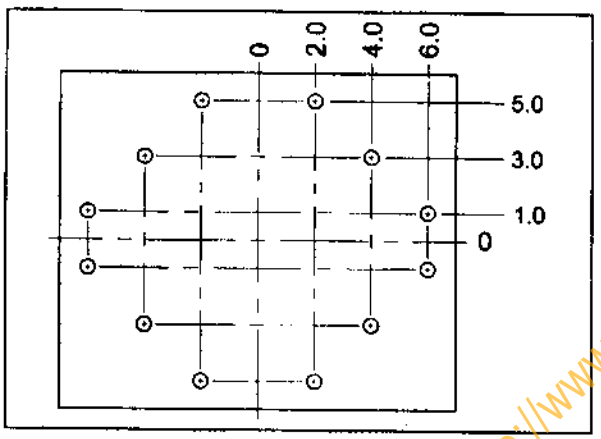
Nhiều tài liệu hay có tại: http://www.me-cad.com/



Hình 40.5. Công tắc xác lập hình ảnh gương bằng tay.

Lập trình - xác lập chiếu gương bằng tay

Hình 40.6 là bản vẽ với lỗ được gia công trong cả 4 góc phần tư, được dùng để minh họa quá trình xác lập và lập trình hình ảnh gương.

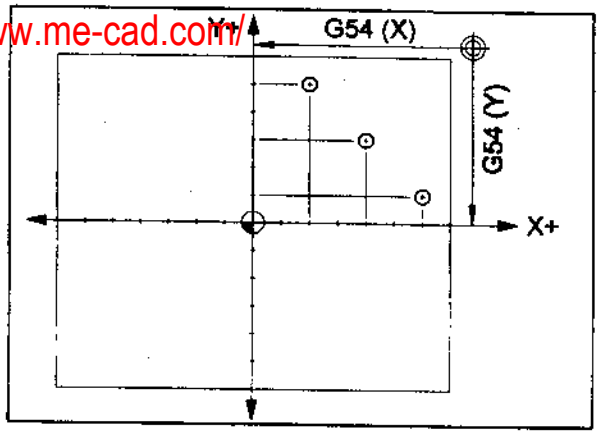


Hình 40.6. Bản vẽ minh họa lập trình hình ảnh gương bằng tay.

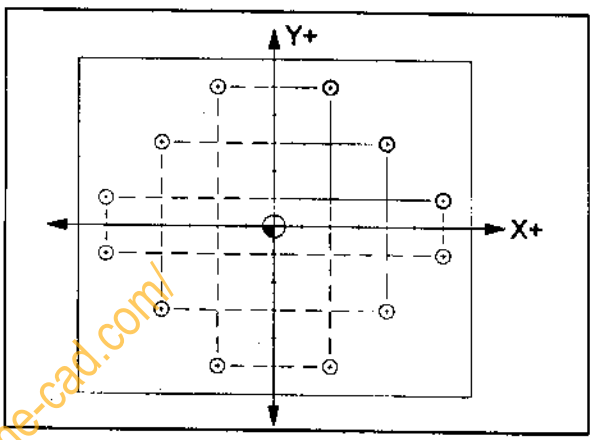
Đối với hình ảnh gương bằng tay, chuyển động dao sẽ chỉ trong một góc phần tư - Hình 40.7 - được chiếu gương vào các góc phần tư khác - Hình 40.8 và chương trình O4001:

```
O4001 (CENTER DRILL THREE HOLES)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S900 M03              (X0Y0)
N4 G43 Z1.0 H01 M08
N5 G99 G82 X6.0 Y1.0 R0.1 Z-0.269 P300 F7.0
N6 X4.0 Y3.0
N7 X2.0 Y5.0
N8 G80 Z1.0 M09
N9 G28 Z1.0 M05
N10 G00 X0 Y0                  (MUST RETURN TO X0Y0)
N11 M30
%
```

Bạn hãy quan sát chuyển động dao thứ nhất trong N3. Dao cắt định vị tại X0Y0, *nơi không có lỗ*. Đây là block quan trọng nhất trong chương trình đối với hình ảnh gương, do vị trí này là *chung cho cả bốn góc phần tư.*



Hình 40.7. Chuyển động dao lập trình cho ba lỗ trong góc phần tư I



Hình 40.8. Chuyển động dao kết quả trong cả bốn góc phần tư sử dụng hình ảnh gương

http://www.me-cad.com/

HÌNH ẢNH GƯƠNG LẬP TRÌNH

Hầu hết các bộ điều khiển đều có hình ảnh gương có thể xác lập nhưng *không thể lập trình*. Hình ảnh gương kích hoạt bằng xác lập của bộ điều khiển được thực hiện trên máy CNC thay vì trong chương trình. Mặt khác, hình ảnh gương lập trình sử dụng các hàm M (hoặc đôi khi là mã G) và hầu như luôn luôn sử dụng chương trình con.

Các xác lập điều khiển là tự động trong chương trình. Các chương trình thực tế đối với hình ảnh gương có thể khác nhau giữa các máy, nhưng nguyên tắc ứng dụng như nhau.

Các hàm hình ảnh gương

Trong các ví dụ này, sẽ sử dụng các hàm:

M21	Chiếu gương theo trục X
M22	Chiếu gương theo trục Y
M23	Xóa chiếu gương (OFF cho các trục)

Hàm M xác lập hình ảnh gương cho từng trục. Nếu một hàm đang có hiệu lực khi lập

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

trình hàm kia, cả *hai* sẽ có hiệu lực. Để chỉ một trục có hiệu lực, cần xóa hàm gương trước.

Xóa chế độ hình ảnh gương khi hoàn tất chuyển động dao.

Ví dụ hình ảnh gương đơn giản

Chương trình O4002 cho ba lỗ trên Hình 40.6, có thể được thay đổi có hình ảnh gương lập trình. Các vị trí tuyệt đối của lỗ được lưu trong chương trình con O4051:

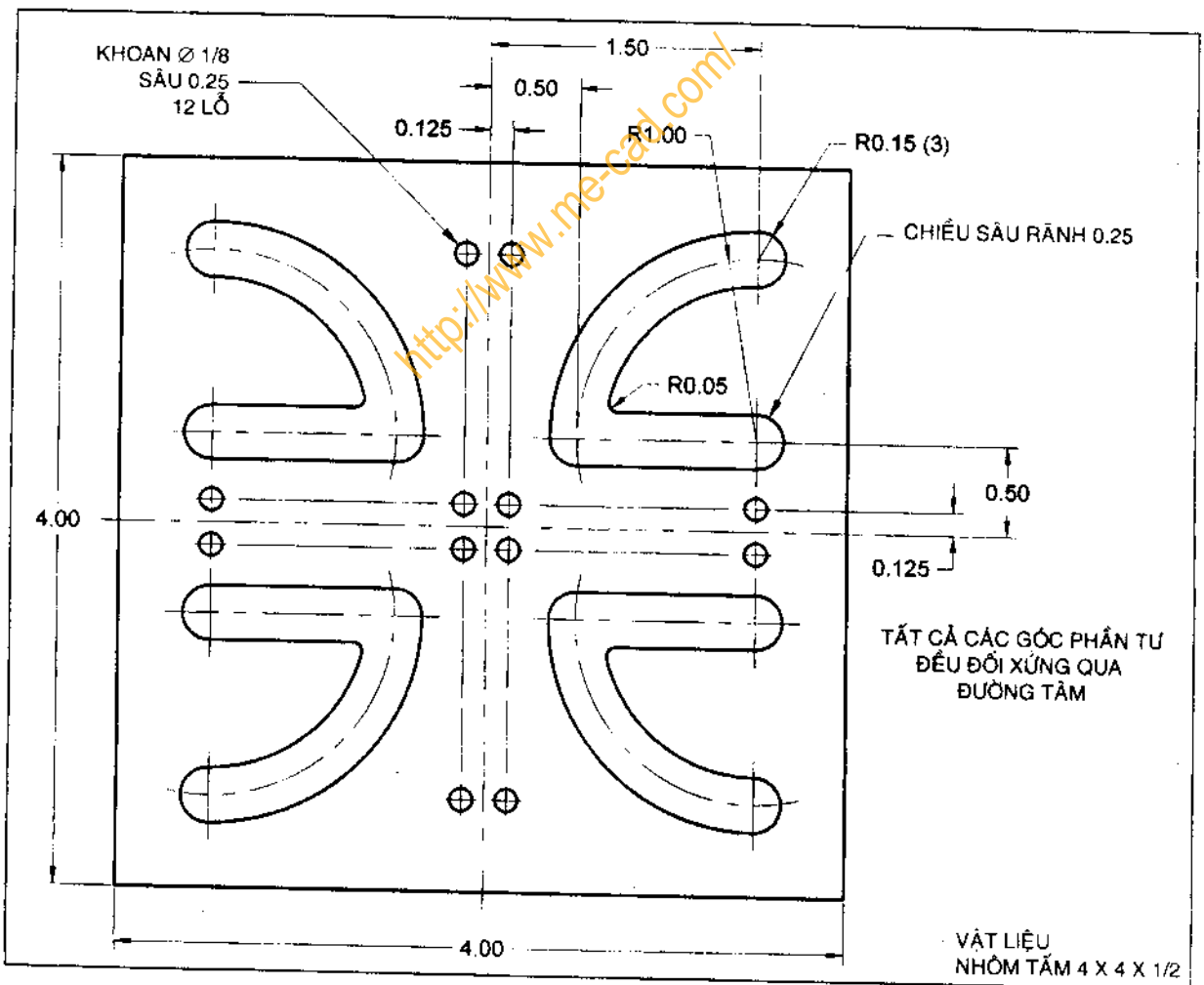
```
O4051
N1 X6.0 Y1.0
N2 X4.0 Y3.0
N3 X2.0 Y5.0
N4 M99
%
```

Chương trình chính O4002 gọi chương trình con O4051 trong các góc phần tư khác nhau, sử dụng các hàm hình ảnh gương. Chú ý, vị trí X0Y0 là chung cho cả bốn góc phần tư.

```
O4002                                   (MAIN PROGRAM)
N1 G20
N2 G17 G40 G80
N3 M23                                  (MIRROR OFF)
N4 G90 G54 G00 X0 Y0 S900 M03           (X0Y0)
N5 G43 Z1.0 H01 M08
N6 G99 G82 R0.1 Z-0.269 P300 F7.0 L0
N7 M98 P4051                            (QUADRANT I)
N8 M21                                  (X-MIRROR ON)
N9 M98 P4051                            (QUADRANT II)
N10 M22                                 (Y-MIRROR ON)
N11 M98 P4051                           (QUADRANT III)
N12 M23                                 (MIRROR OFF)
N13 M22                                 (Y-MIRROR ON)
N14 M98 P4051                           (QUADRANT IV)
N15 G80 Z1.0 M09                        (CYCLE CANCEL)
N16 M23                                 (MIRROR OFF)
N17 G28 Z1.0 M05                        (Z MACHINE ZERO)
N18 G00 X4.0 Y6.0                       (CLEAR ATC LOCATION)
N19 M30                                 (PROGRAM END)
%
```

Chương trình hình ảnh gương hoàn chỉnh

Chương trình hoàn chỉnh về ứng dụng hình ảnh gương với nhiều chuyển động dao sẽ sử dụng hai dao cắt để triển khai chương trình theo bản vẽ trên Hình 40.9. Chương trình cũng sẽ sử dụng sự dịch chuyển tọa độ G52, thay dao tự động, chu kỳ cố định, các chuyển động nội suy, và bù bán kính dao. Cần có hai chương



Hình 40.9. *Ví dụ về hình ảnh gương lập trình, sử dụng chương trình chính O4003 và các chương trình con O4052 và O4053*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

trình con, một để khoan ba lỗ trong O4052 và một để phay rãnh trong O4053.

```
O4052 (SUBPROGRAM - DRILLING)
N1 X0.125 Y0.125                    (MIDDLE HOLE)
N2 X1.5                               (HOLE IN X)
N3 X0.125 Y1.5                        (HOLE IN Y)
N4 X0 Y0 L0              (NO HOLE AT PLATE CENTER)
N5 M99                     (SUBPROGRAM 04152 END)
%
```

Chương trình con O4052 chỉ chứa ba vị trí lỗ trong *góc phần tư I*. Sự gọi chu kỳ không được gộp vào chương trình con này, sự trở về tâm của tấm phôi (N4) vẫn trong chu kỳ nhưng với địa chỉ L0.

```
O4053 (SUBPROGRAM - MILLING)
N1 G00 X1.5 Y1.5                  (CENTER OF SLOT)
N2 G01 Z-0.25 F3.0
N3 G03 X0.5 Y0.5 I0 J-1.0 F5.0
N4 G01 X1.5
N5 G41 D01 X1.365 Y0.485             (SLOT START)
N6 G03 X1.5 Y0.35 I0.135 J0
N7 X1.65 Y0.5 I0 J0.15
N8 X1.5 Y0.65 I-0.15 J0
N9 G01 X0.7254
N10 G02 X0.6754 Y0.7 I0 J0.05
N11 X0.677 Y0.7125 I0.05 J0
N12 X1.5 Y1.35 I0.823 J-0.2125
N13 G03 X1.65 Y1.5 I0 J0.15
N14 X1.5 Y1.65 I-0.15 J0
N15 X0.35 Y0.5 I0 J-1.15
N16 X0.5 Y0.35 I0.15 J0
N17 G01 X1.5
N18 G03 X1.635 Y0.485 I0 J0.135
N19 G01 G40 X1.5 Y0.5                  (SLOT END)
N20 G00 Z0.1
N21 X0 Y0                (MOTION TO PLATE CENTER)
N22 M99                    (SUBPROGRAM O4053 END)
%
```

Góc phần tư I cũng được dùng trong chương trình con O4053 cho một rãnh. Sự gia công bắt đầu với dao cắt ở đường tâm rãnh, gia công thô bán kính và các vách. Sau đó sử dụng bù bán kính và rãnh được gia công tinh theo kích thước. Chương trình con kết thúc ở tâm phôi trong N21, tương tự như khi khoan. Chương trình O4003 sử dụng hai chương trình con. Nếu dùng nhiều dao, kỹ thuật lập trình vẫn không thay đổi.

```
O4003 (MAIN PROGRAM)
(USES SUBPROGRAMS O4052 AND O4053)
(X0 Y0 LOWER LEFT CORNER - Z0 WORK TOP)
(M21 = X-MIRROR ON  ----------------------)
(M22 = Y-MIRROR ON  ----------------------)
(M23 = MIRROR OFF   ----------------------)

(T01 - 1/8 DIA SHORT DRILL)
N1 G17 G20 G40 G80 G49           (STARTUP BLOCK)
N2 T01 M06                        (TOOL CHANGE)
N3 G52 X2.0 Y2.0 M23               (MIRROR OFF)
N4 G90 G54 G00 X0 Y0 S1800 M03 T02
N5 G43 Z1.0 H01 M08
N6 G99 G81 R0.1 Z-0.269 F4.0 L0
N7 M98 P4052                      (QUADRANT I)
N8 M21                          (X-MIRROR ON)
N9 M98 P4052                     (QUADRANT II)
N10 M22                          (Y-MIRROR ON)
N11 M98 P4052                   (QUADRANT III)
N12 M23                           (MIRROR OFF)
N13 M22                            (MIRROR ON)
N14 M98 P4052                    (QUADRANT IV)
N15 G80 M09                     (CYCLE CANCEL)
N16 M23                           (MIRROR OFF)
N17 G52 X0 Y0
N18 G28 Z0.1 M05
N19 G00 X4.0 Y6.0          (CLEAR ATC LOCATION)
N20 M01                       (OPTIONAL STOP)

(T02 - 1/4 DIA CENTER CUTTING END MILL)
N21 T02 M06               (TOOL T02 TO SPINDLE)
N22 G52 X2.0 Y2.0 M23              (MIRROR OFF)
N23 G90 G54 G00 X0 Y0 S2500 M03 T01
N24 G43 Z0.1 H02 M08
N25 M98 P4053                     (QUADRANT I)
N26 M21                         (X-MIRROR ON)
N27 M98 P4053                    (QUADRANT II)
N28 M22                          (Y-MIRROR ON)
N29 M98 P4053                   (QUADRANT III)
N30 M23                           (MIRROR OFF)
N31 M22                          (Y-MIRROR ON)
N32 M98 P4053                    (QUADRANT IV)
N33 M23                           (MIRROR OFF)
N34 G52 X0 Y0 M09
N35 G28 Z0.1 M05
N36 G00 X4.0 Y6.0          (CLEAR ATC LOCATION)
N37 M30                       (PROGRAM END)
%
```

Chú ý cách thức sử dụng G52. Để sử dụng hình ảnh gương một cách chính xác, zero chương trình phải được xác định trên đường chiếu gương (trục đối xứng). Do cần hai đường (trục) cho phép chiếu này, tâm của tấm phôi là zero chương trình. Không cần trả về zero máy X và Y khi kết thúc dao hoặc kết thúc chương trình. Điều cần thiết là vị trí trong không gian trống để thay dao.

HÌNH ẢNH GƯƠNG TRÊN MÁY TIỆN CNC

Hàm *hình ảnh gương* có công dụng chủ yếu trên trung tâm gia công CNC. Trên máy tiện, ứng dụng này bị giới hạn với máy có hai ụ dao, mỗi ụ trên một phía của đường tâm trục chính. Sự chiếu gương thực sẽ sử dụng trục X (đường tâm trục chính) làm trục đối xứng và cho phép cùng một phương pháp lập trình cho cả hai ụ dao.

Gia công với hình ảnh gương có thể được sử dụng độc lập hoặc kết hợp với các tính năng tiết kiệm thời gian khác, chẳng hạn *Quay tọa độ* và *Hàm lập tỷ lệ*.

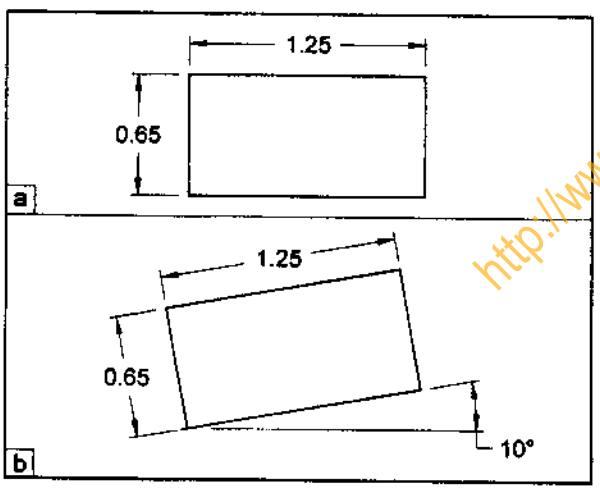
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

41 QUAY TỌA ĐỘ

Chuyển động dao lập trình tạo ra hình dạng, biên dạng, hoặc hốc có thể quay xung quanh điểm xác định theo góc chuyên biệt. Với tính năng điều khiển này, có nhiều cơ hội làm cho sự lập trình trở nên linh hoạt và hiệu quả hơn. Tính năng lập trình rất mạnh này, thường là tùy chọn điều khiển đặc biệt, được gọi là *Quay hệ tọa độ*, hoặc đơn giản là *quay tọa độ*.

Một trong các ứng dụng quan trọng nhất của quay tọa độ là chương trình được xác định theo hướng chuẩn trực nhưng được gia công theo góc (theo yêu cầu kỹ thuật trên bản vẽ). Chế độ chuẩn trực chỉ xác định chiều ngang và chiều đứng, có nghĩa là chương trình dao song song với các trục máy. Lập trình chế độ chuẩn trực đơn giản hơn so với tính toán các vị trí dao ở nhiều điểm thay đổi biên dạng trong định hướng góc. Bạn hãy so sánh hai hình chữ nhật trên Hình 41.1.



Hình 41.1. Vật thể chuẩn trực gốc (a) và vật thể quay (b)

Hình (a) biểu thị chiều chuẩn trực của hình chữ nhật, hình (b) biểu thị cũng hình chữ nhật đó, được quay 10^0 ngược chiều kim đồng hồ. Bằng tay, lập trình quỹ đạo dao cho hình (a) dễ dàng hơn và để hệ điều khiển chuyển quỹ đạo dao theo hình (b). Tính năng quay tọa độ là tùy chọn đặc biệt và phải là một phần của hệ điều khiển.

Về toán học, quay tọa độ là tính năng chỉ yêu cầu ba đại lượng để xác định chi tiết quay - tâm quay, góc quay, và quỹ đạo dao cần quay.

CÁC LỆNH QUAY

Quay tọa độ sử dụng hai lệnh chuẩn bị để chuyển tính năng này ON hoặc OFF. Hai lệnh G điều khiển quay tọa độ gồm:

G68	Quay hệ tọa độ ON
G69	Quay hệ tọa độ OFF

Lệnh G68 sẽ *kích hoạt* quay hệ tọa độ, dựa trên *tâm quay* và *giá trị góc quay (độ)*:

```
G68 X.. Y.. R..
```

Trong đó:

X = Tọa độ X tuyệt đối của tâm quay
Y = Tọa độ Y tuyệt đối của tâm quay
R = Góc quay

Tâm quay

Các tọa độ XY thường là tâm quay. Đây là điểm đặc biệt để quay, điểm có thể được xác định bằng hai trục khác nhau, tùy theo mặt phẳng gia công đã chọn. X và Y là chọn tâm tâm quay tuyệt đối với mặt phẳng G17 hoạt động. G18 sử dụng XZ làm các tọa độ điểm quay, và G19 sử dụng YZ làm các tọa độ điểm quay. Lệnh chọn mặt phẳng G17, G18, hoặc G19 phải được nhập vào chương trình ở thời điểm bất kỳ *trước khi* đưa ra lệnh quay G68.

Nếu các vị trí tọa độ X và Y không được chuyên biệt với lệnh G68 để có tâm quay (trong mặt phẳng G17), *vị trí dao hiện hành* sẽ được dùng làm tâm quay *mặc định*, nhưng phương pháp này thiếu tính thực tiễn và không nên sử dụng.

Bán kính quay

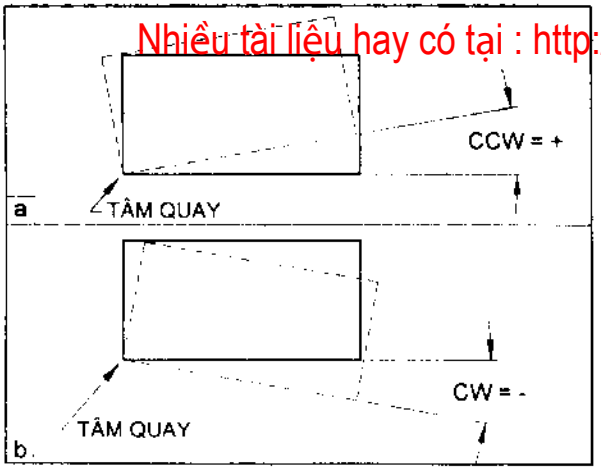
Sự biểu thị góc G68 được chuyên biệt bằng giá trị R. Đơn vị là *độ* (góc), đo từ *tâm xác định*. Số lượng các chữ số thập phân của giá trị R sẽ trở thành giá trị góc. R dương xác định chiều CCW, R âm xác định chiều CW (Hình 41.2).

Đối với ví dụ lập trình cơ bản, bạn sử dụng hình dạng chi tiết đơn giản để dễ hình dung. Chẳng hạn hình chữ nhật với bán kính bo tròn góc (Hình 41.3).

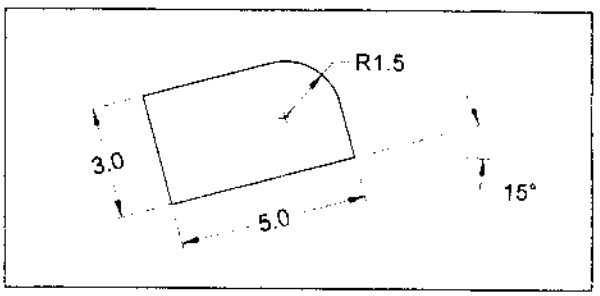
Quỹ đạo dao thực, kể cả sự tiếp cận đến chi tiết và lùi xa chi tiết, thường không ghi trên bản vẽ kỹ thuật. *Ở đây bạn cần cẩn thận*, nếu sự tiến đến và/hoặc lùi xa chi tiết được gộp vào

Nhiều tài liệu hay có tại : http://www.me-cad.com/

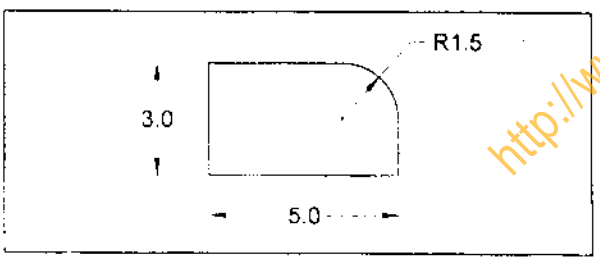
Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 41.2. Chiều quay tọa độ, dựa trên tâm quay: a) Ngược chiều kim đồng hồ (CCW) có góc R dương b) Thuận chiều kim đồng hồ (CW) có góc R âm



Hình 41.3. Chi tiết định hướng theo yêu cầu trên bản vẽ kỹ thuật.

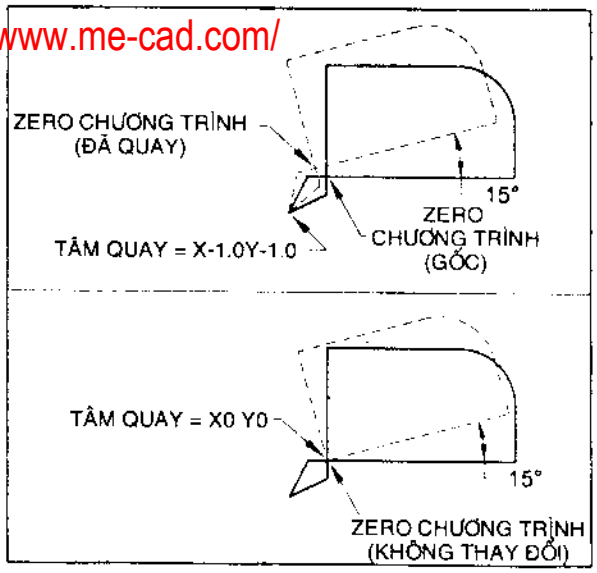


Hình 41.4. Chi tiết định hướng theo chương trình, sử dụng lệnh G68

sự quay tọa độ, *zero chương trình cũng có thể quay*. Trên Hình 41.4, định hướng của chi tiết và 15° ngược chiều kim đồng hồ, dựa theo góc dưới bên trái.

Vào thời điểm này, bạn hãy bỏ qua góc quay chỉ lập trình chi tiết tựa như được định hướng theo vị trí chuẩn trực, nghĩa là vuông góc với các trục (Hình 41.4).

Đối với cắt gọt thực, bạn có thể quyết định gộp hay không gộp các chuyển động dao tiếp cận chi tiết trong sự quay tọa độ. Trên Hình 41.5 là hai khả năng và tác động của sự quay tọa độ đối với zero chương trình. Trong cả hai trường hợp, quỹ đạo dao tiếp cận chi tiết sẽ bắt



Hình 41.5. So sánh quỹ đạo dao lập trình (đường nét liền) và quỹ đạo dao đã quay (đường nét đứt) (a) Zero chương trình gộp trong sự quay tọa độ (b) Zero chương trình không gộp trong sự quay tọa độ

đầu và kết thúc ở cùng vị trí X-1.0 và Y-1.0 (vị trí ở khoảng trống).

Chương trình O4101 minh họa ví dụ (a) trên Hình 41.5, *gộp* sự quay zero chương trình. Nếu zero chương trình không cần quay, bạn chỉ gộp quỹ đạo dao biên dạng chi tiết giữa các lệnh G68 và G69, đưa các chuyển động dao tới gần và ra xa chi tiết ra ngoài phần đó. Bạn nên chú ý G69 trong block N2 - sự xóa được bố trí ở đây để tăng tính an toàn.

```
O4101
N1 G20
N2 G69          (ROTATION CANCELED IF NEEDED)
N3 G17 G80 G40
N4 G90 G54 G00 X-1.0 Y-1.0 S800 M03
N5 G43 Z0.1 H01 M08
N6 G01 Z-0.375 F10.0
N7 G68 X-1.0 Y-1.0 R15.0
N8 G41 X-0.5 Y-0.5 D01 F20.0
N9 Y3.0
N10 X3.5
N11 G02 X5.0 Y1.5 R1.5
N12 G01 Y0.5
N13 X-0.5
N14 G40 X-1.0 Y-1.0 M09
N15 G69                    (ROTATION CANCELED)
N16 G28 X-1.0 Y-1.0 Z1.0 M05
N17 M30
%
```

Chương trình này được triển khai đối với định hướng chuẩn trực của chi tiết (= quay 0°), nhưng được gia công ở 15°, sử dụng tùy chọn quay hệ tọa đó.

Trong ví dụ này, block N8 chứa bù bán kính dao G41. Bù dao bất kỳ được lập trình sẽ được gộp vào khi thực hiện sự quay tọa độ.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Xóa quay tọa độ

Lệnh G69 *xóa* hàm quay tọa độ và đưa hệ điều khiển trở về trạng thái chuẩn trực bình thường. Lệnh G89 phải được đặt trong block riêng rẽ, như trong ví dụ O4101.

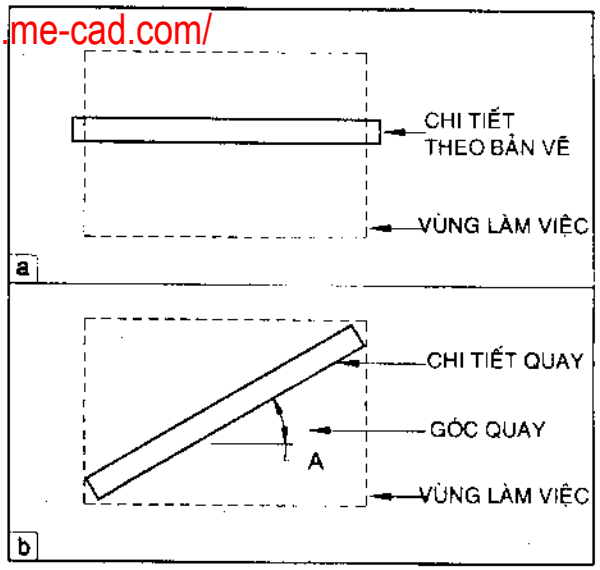
Các ứng dụng chung

Như đã đề cập, đa số các máy CNC không có hàm quay tọa độ hoặc chỉ có hàm này ở dạng tính năng tùy chọn. Hàm này có thể rất hữu dụng trong hai lĩnh vực gia công.

- Nếu bản chất của công việc yêu cầu các chi tiết chuẩn trực được gia công theo góc (theo yêu cầu trên bản vẽ). Ví dụ nêu trên thuộc nhóm này.
- Nếu có hành trình X và/hoặc Y ngắn trên trung tâm gia công và chi tiết định vị trên bàn máy theo góc đã biết, do hành trình máy bị giới hạn.

Ứng dụng thứ hai là ví dụ rất tốt về quay hệ tọa độ, nhưng phải thỏa hai điều kiện:

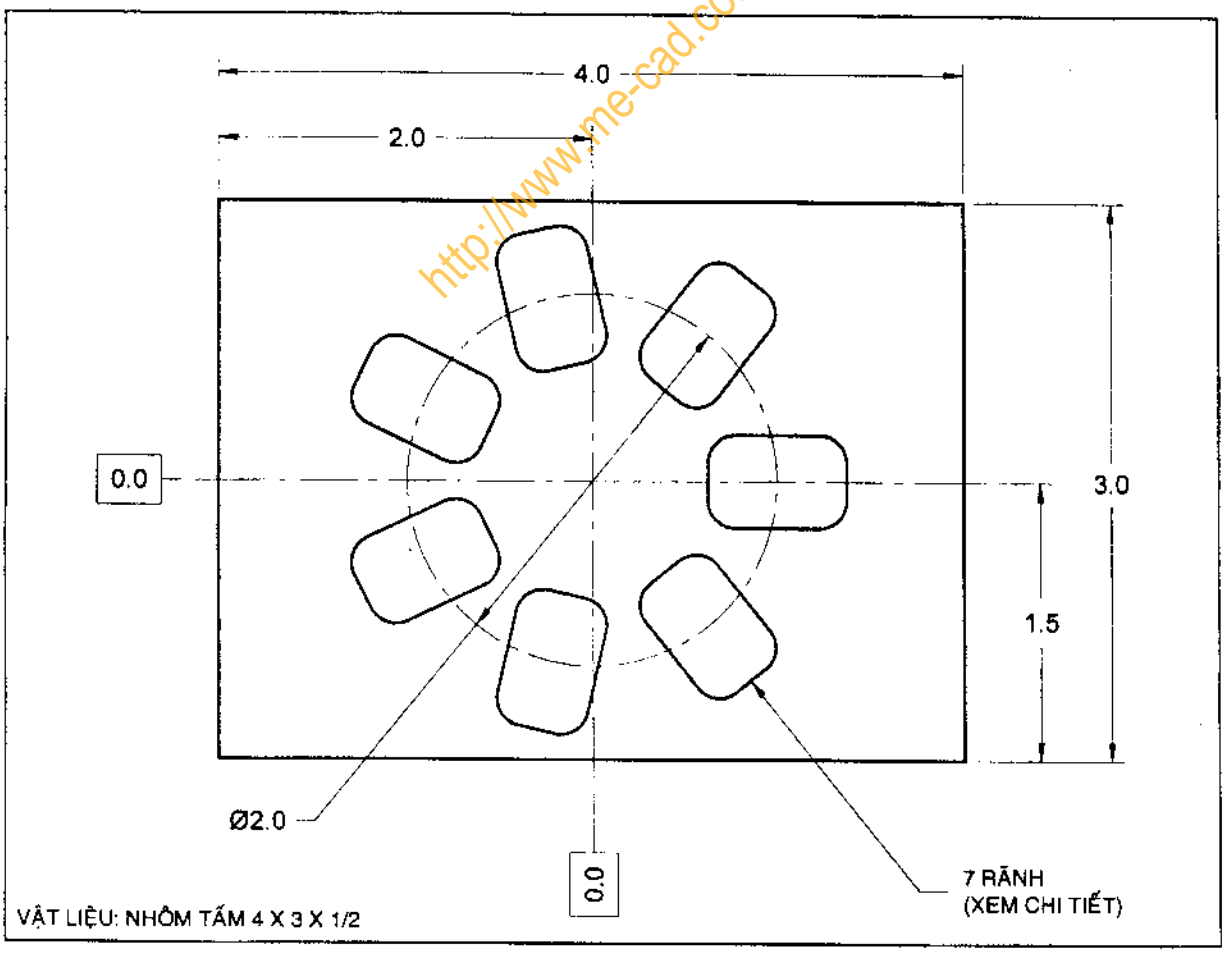
- Chi tiết quay phải trong phạm vi không gian làm việc (gia công)
- Phải biết góc gá lắp chi tiết



Hình 41.6. *Quay tọa độ được áp dụng để khớp chi tiết dài trong không gian làm việc*

Trên Hình 41.6, chi tiết không khớp trong không gian làm việc chuẩn trực, nhưng có thể trong phạm vi không gian đó sau khi quay.

Phương pháp này rất thú vị nhưng đôi khi



Hình 41.7. *Ví dụ quay tọa độ - chương trình O4102*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

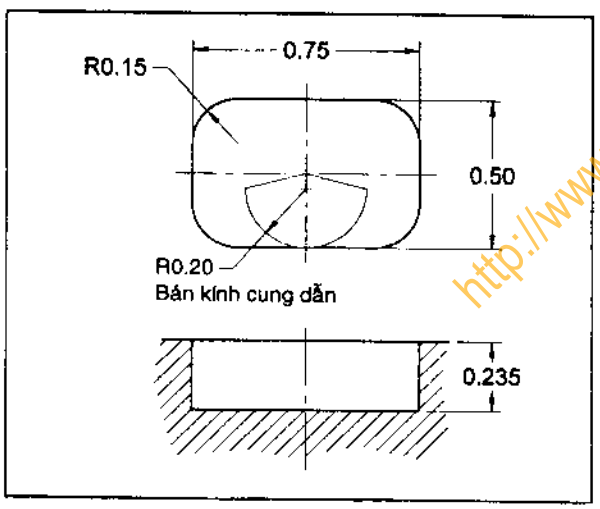
không thể đặt vào diện tích làm việc chỉ dài 20 mm. Tuy nhiên, có các trường hợp cho phép áp dụng kỹ thuật lập trình này. Minh họa chỉ nêu ra các nguyên tắc ứng dụng tổng quát. Nếu không biết góc định vị, bạn có thể dùng đồng hồ so để đo và tính toán bằng các hàm lượng giác. Trong một số trường hợp, có thể dùng đồ gá cho chi tiết gia công.

Không nên nhầm lẫn giữa không gian làm việc và kích cỡ bàn máy.

Kích cỡ bàn máy thường lớn hơn so với thể tích làm việc thực sự để có thể gá lắp chi tiết. Không gian làm việc được dùng cho lập trình và cho gá lắp chi tiết gia công, luôn luôn được xác định bằng các giới hạn của chuyển động dao. Không gian làm việc phải có khả năng bảo đảm mọi chuyển động dao lập trình và các khoảng hở, kể cả khi bù bán kính dao có hiệu lực.

ỨNG DỤNG THỰC TIỄN

Trong nhiều trường hợp, có thể sử dụng chương trình con rất hiệu quả cùng với sự quay tọa độ. Các ứng dụng, chẳng hạn phay các hình đa giác hoặc gia công tại các vị trí đường tròn lỗ chỉ là vài khả năng cụ thể. Ví dụ chi tiết dưới đây (Hình 41.7) nêu ra bản vẽ chi tiết đơn giản nhưng sẽ ứng dụng nhiều kỹ thuật lập trình đã đề cập.



Hình 41.8. Hình chiếu của rãnh gia công trong chương trình O4102

Trước hết cần đánh giá các yêu cầu và điều kiện để triển khai chương trình cốt lõi của chương trình là gia công 7 rãnh với dao phay ngón Ø .250 (kiểu cắt tâm). Để tăng tính hiện thực cho chương trình, thay vì ăn vào chiều sâu toàn phần .235 inch, có thể chọn 0.050 làm chiều sâu cực đại của đường cắt. Chương trình sẽ để lại lượng dư cho gia công tinh các vách rãnh (.0075 trên mỗi phía). Ngoài ra, mọi góc sắc phải được xử lý với sự vạt góc tối thiểu. Về tổng thể, chỉ sử dụng ba dao cắt:

DAO PHAY MẶT Ø3.0
DAO PHAY NGÓN ĐỂ CẮT TÂM Ø1/4
DAO VẠT GÓC Ø3/8

Đây là ứng dụng lập trình tiên tiến. Thoạt đầu, có thể chưa hiểu rõ chương trình, nhưng với thời gian và kinh nghiệm, sẽ dễ dàng diễn dịch chương trình theo từng block.

Chương trình chính O4102 có bốn chương trình con. Mặc dầu một số phần có thể hơi khó hiểu, nhưng có một yếu tố chính cần nắm rõ. Trong hai chương trình con sẽ có block sau đây:

```
G91 G68 X0 Y0 R51.429
```

Mục đích của block này là dịch chuyển sang rãnh kế tiếp theo góc. X0Y0 giữ không đổi, chúng *luôn luôn* là giá trị tuyệt đối, chỉ có góc là số gia, do lệnh G91.

Ví dụ này không chỉ là minh họa tốt về quay hệ tọa độ, mà còn nêu ra các kỹ thuật tiên tiến sử dụng chương trình con và các tính năng bổ sung. Không có các kỹ thuật lập trình tiên tiến, vẫn có thể viết chương trình này, nhưng sẽ dài hơn nhiều và hầu như không thể tối ưu hóa tại máy CNC. Chương trình hoàn tất, O4102, được chú thích rõ ràng từng block, cho phép theo dõi cấu trúc chương trình một cách dễ dàng.

```
O4102 (COORDINATE SYSTEM ROTATION)

(7 POCKETS - PETER SMID - VERIFIED ON FANUC 15M CNC SYSTEM)
(PARAMETER #6400 BIT #0 - RIN - MUST BE SET TO 1 TO ALLOW G90 AND G91)
(MATERIAL 4 X 3 X 1/2 ALUMINUM PLATE - HORIZONTAL LAYOUT)
(X0Y0 IS CENTER OF 2.0 DIA CIRCLE - Z0 AT THE FINISHED TOP OF THE PLATE)

(T01 ..... 3.0 DIA FACE MILL - SKIM CUT TO CLEAN TOP FACE)
(T02 ..... 1/4 DIA CENTER CUTTING END MILL - MAX DEPTH OF CUT 0.05)
(T03 ..... 3/8 DIA CHAMFERING TOOL - 90 DEGREES - MINIMUM CHAMFER)
(T02 / D51 - OFFSET FOR ROUGHING POCKET WALLS ..... 0.140 SUGGESTED - 0.0075 PER SIDE)
(T02 / D52 - OFFSET FOR FINISHING POCKET WALLS .... 0.125 SUGGESTED)
(T03 / D53 - OFFSET FOR CHAMFERING ..... 0.110 SUGGESTED - TO BE ADJUSTED)
(INCREMENT OF ROTATION ..... 360/7 = 51.429 DEGREES)

(T01 - 3.0 DIA FACE MILL - SKIM CUT TO CLEAN TOP FACE)
N1 G20                                    (ENGLISH UNITS)
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N2 G69                                          (CANCEL COORDINATE ROTATION IF ACTIVE)
N3 G17 G40 G80 T01                              (SEARCH FOR T01 IF NOT READY)
N4 M06                                          (T01 TO THE SPINDLE)
N5 G90 G54 G00 X-1.375 Y-3.25 S3500 M03 T02     (XY START POSITION FOR FACE MILLING)
N6 G43 Z1.0 H01 M08                             (Z CLEARANCE FOR SETUP - COOLANT ON)
N7 G01 Z0 F30.0                                 (TOP OF FINISHED PART FOR FACE MILLING)
N8 Y3.125 F15.0                                 (FACE MILL LEFT SIDE)
N9 G00 X1.375                                   (MOVE TO THE RIGHT SIDE)
N10 G01 Y-3.25                                  (FACE MILL RIGHT SIDE)
N11 G00 Z1.0 M09                                (Z AXIS RETRACT - COOLANT OFF)
N12 G28 Z1.0 M05                                (Z AXIS HOME FOR TOOL CHANGE)
N13 M01                                         (OPTIONAL STOP)

(T02 - 1/4 DIA CENTER CUTTING END MILL - MAX DEPTH OF CUT 0.05)
N14 T02                                         (SEARCH FOR T02 IF NOT READY)
N15 M06                                         (T02 TO THE SPINDLE)
N16 G69                                         (CANCEL COORDINATE ROTATION IF ACTIVE)
N17 G90 G54 G00 X1.0 Y0 S2000 M03 T03           (XY START POSITION FOR THE CENTER OF POCKET 1)
N18 G43 Z1.0 H02 M08                            (Z CLEARANCE FOR SETUP - COOLANT ON)
N19 G01 Z0.02 F30.0                             (CONTROLS 0.005 LEFT ON THE POCKET BOTTOM)
N20 M98 P4152 L7                                (ROUGH AND FINISH MILLING OF SEVEN POCKETS)
N21 G69                                         (CANCEL COORDINATE ROTATION IF ACTIVE)
N22 G90 G00 Z1.0 M09                            (Z AXIS RETRACT - COOLANT OFF)
N23 G28 Z1.0 M05                                (Z AXIS HOME FOR TOOL CHANGE)
N24 M01                                         (OPTIONAL STOP)

(T03 - 3/8 DIA CHAMFERING TOOL - 90 DEGREES)
N25 T03                                         (SEARCH FOR T03 IF NOT READY)
N26 M06                                         (T03 TO THE SPINDLE)
N27 G69                                         (CANCEL COORDINATE ROTATION IF ACTIVE)
N28 G90 G54 G00 X-2.5 Y-2.0 S4000 M03 T01       (XY START POSITION FOR PERIPHERAL CHAMFERING)
N29 G43 Z1.0 H03 M08                            (Z CLEARANCE FOR SETUP - COOLANT ON)
N30 G01 Z-0.075 F50.0                           (ABSOLUTE DEPTH FOR CHAMFERING Z-0.075)
N31 G41 X-2.0 D53 F12.0                         (APPROACH MOTION AND RADIUS OFFSET)
N32 Y1.5                                        (CHAMFER LEFT EDGE)
N33 X2.0                                        (CHAMFER TOP EDGE)
N34 Y-1.5                                       (CHAMFER RIGHT EDGE)
N35 X-2.5                                       (CHAMFER BOTTOM EDGE)
N36 G00 G40 Y-2.0                               (RETURN TO START POINT AND CANCEL OFFSET)
N37 Z0.1                                        (CLEAR ABOVE PART)
N38 X1.0 Y0                                     (MOTION TO THE CENTER OF POCKET 1)
N39 M98 P4154 L7                                (CHAMFER SEVEN POCKETS)
N40 G69                                         (CANCEL COORDINATE ROTATION IF ACTIVE)
N41 G90 G00 Z1.0 M09                            (Z AXIS RETRACT - COOLANT OFF)
N42 G28 Z1.0 M05                                (Z AXIS HOME FOR TOOL CHANGE)
N43 X-2.0 Y8.0                                  (PART CHANGE POSITION)
N44 M30                                         (END OF MAIN PROGRAM O4102)
%


O4151                                           (POCKET TOOL PATH AT ZERO DEGREES - POCKET 1)
N101 G91 Z-0.05                                 (START AT POCKET CENTER - FEED-IN BY 0.05)
N102 M98 P4153                                  (POCKET CONTOUR - O4153 USED FOR ROUGHING)
N103 M99                                        (END OF SUBPROGRAM O4151)
%


O4152                                           (SUBPROGRAM FOR MILLING POCKETS)
N201 M98 P4151 D51 F5.0 L5                      (ROUGH TO ABS. DEPTH Z-0.230 IN FIVE STEPS)
N202 Z-0.005                                    (FINISH TO FINAL ABSOLUTE DEPTH Z-0.235)
N203 M98 P4153 D52 F4.0                         (POCKET CONTOUR - 04253 USED AT FULL DEPTH)
N204 G90 G00 Z0.02                              (RETURN TO ABS. MODE AND Z AXIS CLEAR POS.)
N205 G91 G68 X0 Y0 R51.429                      (NEXT POCKET ANGLE INCREMENT)
N206 G90 X1.0 Y0                                (MOVE TO NEXT ROTATED XY AXES START POSITION)
N207 M99                                        (END OF SUBPROGRAM O4152)
%


O4153                                           (POCKET TOOL PATH AT ZERO DEGREES - POCKET 1)
N301 G41 X-0.2 Y-0.05                           (LEAD-IN LINEAR MOTION)
N302 G03 X0.2 Y-0.2 I0.2 J0                     (LEAD-IN CIRCULAR MOTION)
N303 G01 X0.225 Y0                              (CONTOUR BOTTOM WALL ON THE RIGHT)
N304 G03 X0.15 Y0.15 I0 J0.15                   (CONTOUR LR CORNER RADIUS)
N305 G01 X0 Y0.2                                (CONTOUR RIGHT SIDE WALL)
N306 G03 X-0.15 Y0.15 I-0.15 J0                 (CONTOUR UR CORNER RADIUS)
N307 G01 X-0.45 Y0                              (CONTOUR TOP SIDE WALL)
N308 G03 X-0.15 Y-0.15 I0 J-0.15                (CONTOUR UL CORNER RADIUS)
```

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N309 G01 X0 Y-0.2                          (CONTOUR LEFT SIDE WALL)
N310 G03 X0.2 Y0.2 R0.2                    (CONTOUR LL CORNER RADIUS)
N311 G01 X0.225 Y0                         (CONTOUR BOTTOM WALL ON THE LEFT)
N312 G03 X0.2 Y0.2 I0 J0.2                 (LEAD-OUT CIRCULAR MOTION)
N313 G01 G40 X-0.2 Y0.05                   (LEAD-OUT LINEAR MOTION)
N314 M99                                   (END OF SUBPROGRAM O4153)
%

O4154                                      (SUBPROGRAM FOR CHAMFERING POCKETS)
N401 G91 G01 Z-0.175 F50.0                 (CHAMFERING DEPTH FOR POCKET AT ABS. Z-0.075)
N402 M98 P4153 D53 F8.0                    (POCKET CONTOUR - O4153 USED FOR CHAMFERING)
N403 G90 G00 Z0.1                          (RETURN TO ABS. MODE AND Z AXIS CLEAR POS.)
N404 G91 G68 X0 Y0 R51.429                 (NEXT POCKET ANGLE INCREMENT)
N405 G90 X1.0 Y0                           (MOVE TO NEXT ROTATED XY AXES START POSITION)
N406 M99                                   (END OF SUBPROGRAM O4154)
%
```

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

42 HÀM LẬP TỶ LỆ

Nói chung, quỹ đạo dao lập trình đối với trung tâm CNC biểu thị các kích thước trên bản vẽ, có lẽ với bù bán kính dao cắt có hiệu lực. Đôi khi, có những thời điểm khi quỹ đạo dao gia công đã được lập trình phải được lặp lại, nhưng gia công *nhỏ hơn* hoặc *lớn hơn* so với chi tiết ban đầu, đồng thời vẫn duy trì tỷ lệ. Để đạt được mục đích này, cần sử dụng tính năng *Hàm lập tỷ lệ*. Bạn hãy chú ý hai yếu tố:

- Hàm lập tỷ lệ là tùy chọn trên nhiều bộ điều khiển và có thể không khả dụng trên một số máy.
- Hàm này có thể sử dụng một số tham số hệ thống.

Để tăng tính linh hoạt trong lập trình, hàm lập tỷ lệ có thể được dùng kết hợp với các hàm lập trình khác, chẳng hạn *Dịch chuyển mốc chuẩn*, *Hình ảnh gương*, và *Quay hệ tọa độ*, đã được trình bày trong các Chương trước.

MÔ TẢ

Hệ điều khiển áp dụng hệ số tỷ lệ chuyên biệt cho mọi chuyển động lập trình, có nghĩa là giá trị lập trình của tất cả các trục sẽ thay đổi. Quá trình lập tỷ lệ thực chất là nhân giá trị trục lập trình với hệ số tỷ lệ, dựa trên điểm tâm tỷ lệ. Nhà lập trình phải cung cấp cả *tâm tỷ lệ* và *hệ số tỷ lệ*. Thông qua tham số hệ thống điều khiển, sự lập tỷ lệ có thể được thực hiện hiệu quả hoặc không hiệu quả đối với từng trục chính, nhưng không thể đối với các trục bổ sung bất kỳ. Đa số sự lập tỷ lệ chỉ áp dụng cho các trục X và Y.

Điều quan trọng cần nhận biết là một số giá trị và các đại lượng được xác lập trước không bị tác động từ hàm lập tỷ lệ, chẳng hạn các giá trị bù. Các hàm bù dưới đây *không* thay đổi nếu hàm lập tỷ lệ có hiệu lực:

- Giá trị bù bán kính dao ... G41 - G42/D
- Giá trị bù chiều dài dao ... G43 - G44/H
- Giá trị bù vị trí dao ... G45 - G48/H

Trong các chu kỳ cố định, còn có ba tình huống *không* bị tác động từ hàm lập tỷ lệ:

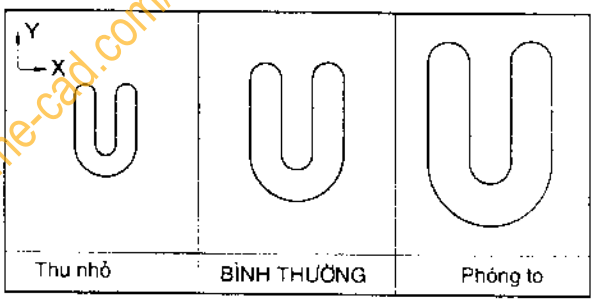
- Các giá trị dịch chuyển X và Y trong các chu kỳ G76 và G87
- Chiều sâu khoan gián đoạn Q trong các chu kỳ G83 và G73
- Giá trị vạt góc đã lưu đối với các chu kỳ G83 và G73

Sử dụng hàm lập tỷ lệ

Trong công nghiệp, có nhiều ứng dụng cần lập tỷ lệ quỹ đạo hiện hữu, cho phép tiết kiệm thời gian và công sức. Dưới đây là một số khả năng áp dụng hàm lập tỷ lệ.

- Các chi tiết có dạng hình học tương tự nhau hoặc đồng dạng.
- Gia công với hệ số co được thiết lập sẵn.
- Gia công khuôn mẫu.
- Chuyển đổi hệ Anh sang hệ mét và ngược lại.
- Thay đổi kích cỡ các ký tự chạm khắc.

Bất kể loại ứng dụng, sự lập tỷ lệ được dùng để tạo ra quỹ đạo dao mới lớn hơn hoặc nhỏ hơn quỹ đạo ban đầu. Lập tỷ lệ do đó được sử dụng để *phóng to* (tăng kích cỡ) hoặc *thu nhỏ* (giảm kích cỡ) của quỹ đạo dao hiện hữu (Hình 42.1).



Hình 42.1. So sánh sự thu nhỏ (trái) và phóng to (phải) chi tiết với chi tiết bình thường (giữa)

ĐỊNH DẠNG LẬP TRÌNH

Để cung cấp thông tin cần thiết cho bộ điều khiển, nhà lập trình phải nhập các dữ liệu dưới đây:

- Tâm tỷ lệ ... Điểm quay
- Hệ số tỷ lệ ... Phóng to hoặc thu nhỏ

Lệnh chuẩn bị đối với hàm lập tỷ lệ là G51, và lệnh xóa là G50:

G50	Xóa chế độ tỷ lệ	Tỷ lệ OFF
G51	Kích hoạt chế độ tỷ lệ	Tỷ lệ ON

Hàm lập tỉ lệ sử dụng định dạng lập trình như sau:

```
G51 I.. J.. K.. P..
```

Trong đó:

I = Tọa độ X của tâm tỷ lệ (tuyệt đối)

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

J = Tọa độ Y của tâm tỷ lệ (tuyệt đối)
K = Tọa độ Z của tâm tỷ lệ (tuyệt đối)
P = Hệ số tỷ lệ (số gia 0.001 hoặc 0.00001)

Lệnh G51 phải được lập trình trong block riêng rẽ. Các lệnh liên quan với sự trở về zero máy, G27, G28, G29 và G30, phải được lập trình trong chế độ tỷ lệ OFF. Nếu dùng G92 để đăng ký vị trí, lệnh này cũng phải lập trình trong chế độ tỷ lệ OFF. Bù bán kính dao G41/G42 phải được xóa bằng G40 trước khi kích hoạt hàm lập tỷ lệ. Các lệnh và các hàm khác có thể hoạt động, kể cả các lệnh bù từ G54 đến G59.

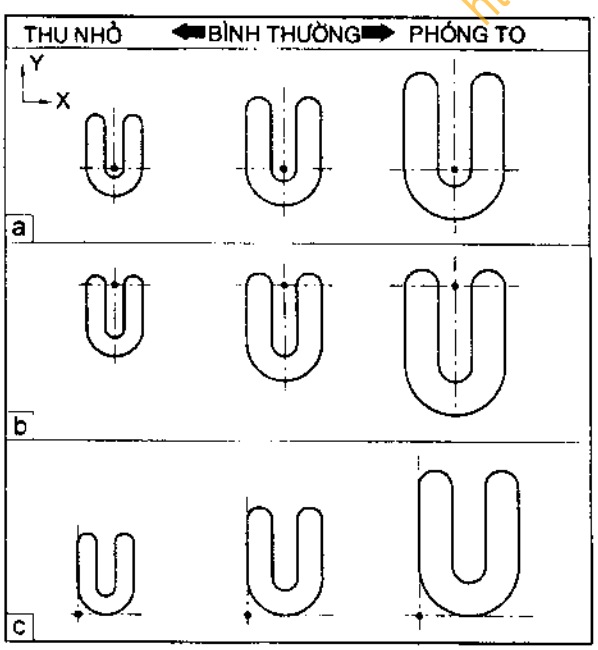
Tâm tỷ lệ

Tâm tỷ lệ xác định vị trí của quỹ đạo dao được lập tỷ lệ.

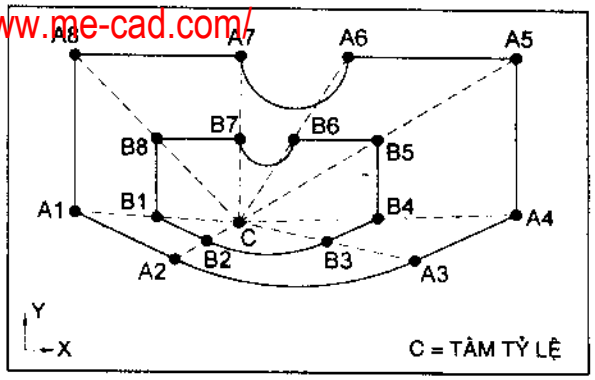
Fanuc 15M sử dụng I/J/K để chuyên biệt tâm tỷ lệ theo các trục X/Y/Z tương ứng. Các giá trị này luôn luôn được lập trình theo chế độ tuyệt đối. Cần biết một nguyên tắc quan trọng khi tâm tỷ lệ điều khiển vị trí của quỹ đạo dao được lập tỷ lệ:

Chi tiết lập tỷ lệ sẽ luôn luôn *mở rộng ra xa* hoặc *thu hẹp về phía* tâm tỷ lệ với lượng bằng nhau dọc theo các trục (Hình 42.2)

Để hiểu biên dạng tương đối phức tạp, bạn hãy so sánh biên dạng đầu và biên dạng sau khi lập tỷ lệ được nêu trên Hình 42.3. Hình này biểu thị hai quỹ đạo dao gia công (A và B) với tâm tỷ lệ C. Tùy theo giá trị của hệ số tỷ lệ, kết quả sẽ là quỹ đạo A1 đến A8 hoặc B1 đến B8.



Hình 42.2. So sánh vị trí chi tiết lập tỷ lệ dựa trên tỷ lệ



Hình 42.3. Tác dụng của tâm tỷ lệ đối với chi tiết được lập tỷ lệ.

Các điểm A1 đến A8 và B1 đến B8 trong minh họa biểu thị các vị trí thay đổi biên dạng của quỹ đạo dao.

- Nếu quỹ đạo dao A1 đến A8 là quỹ đạo ban đầu, quỹ đạo B1 đến B8 là quỹ đạo dao được lập tỷ lệ qua tâm C, với hệ số tỷ lệ NHỎ HƠN 1.
- Nếu quỹ đạo dao B1 đến B8 là quỹ đạo ban đầu, quỹ đạo A1 đến A8 là quỹ đạo dao được lập tỷ lệ qua tâm C, với hệ số tỷ lệ LỚN HƠN 1.

Đường nét đứt nối các điểm riêng rẽ được sử dụng để dễ hình dung hàm lập tỷ lệ. Bắt đầu từ tâm C, đường này luôn luôn nối đến điểm thay đổi biên dạng. Điểm B luôn luôn là điểm trung bình giữa C và A tương ứng. Trong thực tế, điều đó có nghĩa là khoảng cách CB5 và khoảng cách B5A5 là bằng nhau.

Hệ số tỷ lệ

Hệ số tỷ lệ xác định kích cỡ của quỹ đạo dao được lập tỷ lệ.

Hệ số tỷ lệ cực đại liên quan với hệ số tỷ lệ nhỏ nhất. Các hệ thống CNC hiện đại có thể được xác lập bên trong – thông qua tham số hệ thống – để xác lập sẵn hệ số tỷ lệ nhỏ nhất theo 0.001 hoặc 0.00001. Một số hệ CNC cũ chỉ có thể xác lập 0.001 là hệ số tỷ lệ nhỏ nhất. Hệ số tỷ lệ là độc lập với đơn vị đo được dùng trong chương trình – G20 hoặc G21.

Khi hệ số tỷ lệ nhỏ nhất là 0.001, tỷ lệ lớn nhất có thể lập trình là 999.999. Nếu hệ số tỷ lệ nhỏ nhất là 0.00001, tỷ lệ lập trình lớn nhất là 9.9999. Với sự lựa chọn cho trước, bạn cần quyết định giữa các tỷ lệ lớn sẽ giảm độ chính xác, hoặc ngược lại. Đối với hầu hết các ứng dụng, hệ số nhỏ nhất 0.001 là hoàn toàn đủ.

- ❑ Hệ số tỷ lệ > 1 ... Phóng to
- ❑ Hệ số tỷ lệ = 1 ... Không đổi
- ❑ Hệ số tỷ lệ <1 ... Thu nhỏ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Khi không có địa chỉ D trong khối G51 theo mặc định, xác lập tham số hệ thống sẽ có hiệu lực.

Sai số làm tròn khi lập tỷ lệ

Quá trình chuyển đổi bất kỳ đều có thể dẫn đến sai số, thường do làm tròn số. Ví dụ, chuyển từ inch sang milimét sử dụng thừa số 25.4, chẳng hạn chuyển giá trị lập trình 1.5 inch sang milimét tương đương:

1.5 inch x 25.4 = 38.1 mm

Kết quả trong trường hợp này là hoàn toàn chính xác. Ví dụ thứ hai, 1.5625 inch:

1.5625 inch x 25.4 = 39.6875 mm

Giá trị theo hệ mét là chính xác trong phạm vi bốn chữ số thập phân đối với lập trình theo đơn vị Anh.

Chuyển từ milimét sang inch sẽ hơi khác. Hệ số tỷ lệ từ mm sang inch (với độ chính xác 9 chữ số thập phân) là 0.039370079. Tuy nhiên trong lập trình chỉ có thể sử dụng hệ số với ba hoặc năm chữ số thập phân. Điều đó có nghĩa là làm tròn hệ số này sẽ dẫn kết quả chuyển đổi không chính xác. Trong nhiều trường hợp, kết quả làm tròn là chấp nhận được, tuy nhiên, điều quan trọng là xét khả năng sai số.

Bạn hãy so sánh sai số với các hệ số làm tròn khác nhau đối với 12.7 mm, đúng bằng 0.500 inch.

➲ **Sử dụng hệ số tối thiểu 0.001**

inch = 12.7 mm x 0.039
= 0.4953 ...sai số 0.0047 in.

Inch = 12.7 mm x 0.038
= 0.4826 ...sai số 0.0174 in.

Inch = 12.7 mm x 0.040
= 0.5080 ...sai số 0.0080 in.

➲ **Sử dụng hệ số tối thiểu 0.00001**

Inch = 12.7 mm x 0.03937
= 0.499999 ... sai số 0.000001

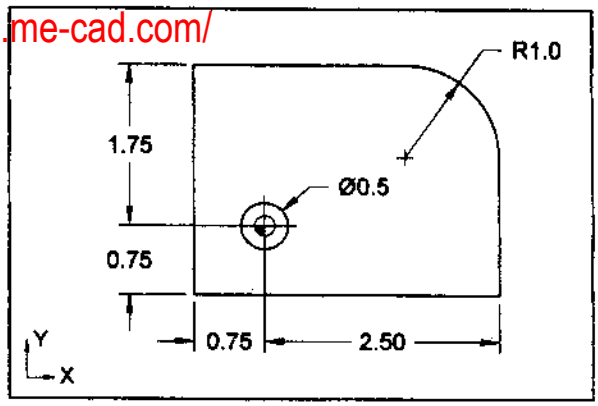
Inch = 12.7 mm x 0.03958
= 0.500126 ... sai số 0.000126

Inch = 12.7 mm x 0.03936
= 0.499812 ... sai số 0.000128

Các vị dụ này có tính điển hình. Nếu áp dụng độ co 5%, ví dụ hệ số tỷ lệ 1.05 (phóng to) hoặc 0.95 (thu nhỏ), độ chính xác của chi tiết vẫn được bảo đảm.

VÍ DỤ LẬP TRÌNH

Ví dụ thứ nhất là rất đơn giản (Hình 42.4)



Hình 42.4. Bản vẽ minh họa hàm lập tỷ lệ – Chương trình O4201 và O4202.

O4201 là chương trình gia công biên dạng cơ bản, sử dụng một dao và chỉ một đường cắt xung quanh chu vi chi tiết. Đây là chương trình bình thường, không lập tỷ lệ.

```
04301 (BASIC PROGRAM USING G54 - NOT SCALED)
N1 G20
N2 G17 G40 G80
N3 G90 G00 G54 X-1.25 Y-1.25 S800 M03
N4 G43 Z1.0 H01 M08
N5 G01 Z-0.7 F50.0
N6 G41 X-0.75 D01 F25.0
N7 Y1.75 F15.0
N8 X1.5
N9 G02 X2.5 Y0.75 I0 J-1.0
N10 G01 Y-0.75
N11 X-1.25
N12 G40 Y-1.25 M09
N13 G00 Z1.0
N14 G28 Z1.0
N15 G28 X-1.25 Y-1.25
N16 M30
%
```

Chương trình O4202 là phiên bản chỉnh sửa của O4201, có hệ số tỷ lệ 1.05 – phóng to 5% - và tâm tỷ lệ tại X0Y0Z0. Có thể bỏ qua KD trong G51

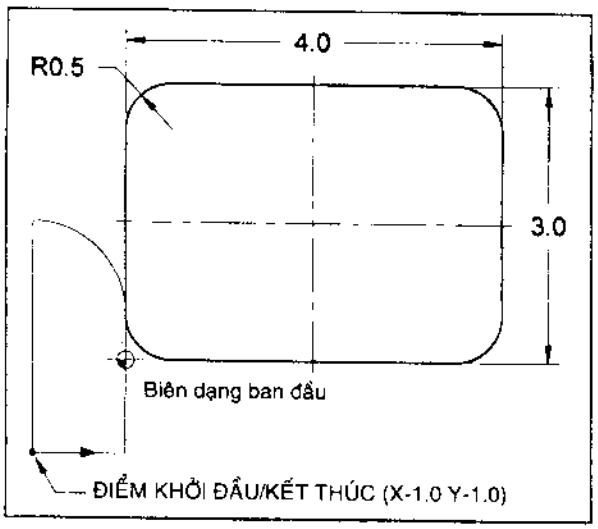
```
O4202
(PROGRAM 04301 SCALED BY 1.05 FACTOR)
N1 G20
N2 G17 G40 G80
N3 G50                                  (SCALING OFF)
N4 G90 G00 G54 X-1.25 Y-1.25 S800 M03
N5 G43 Z1.0 H01 M08
N6 G51 I0 J0 K0 P1.050                  (FROM X0Y0Z0)
N7 G01 Z-0.7 F50.0
N8 G41 X-0.75 D01 F25.0
N9 Y1.75 F15.0
N10 X1.5
N11 G02 X2.5 Y0.75 I0 J-1.0
N12 G01 Y-0.75
N13 X-1.25
N14 G40 Y-1.25 M09
N15 G50                                 (SCALING OFF)
N16 G00 Z1.0
N17 G28 Z1.0
N18 G28 X-1.25 Y-1.25
N19 M30
%
```

Chương trình O4203 phức tạp hơn. Hình

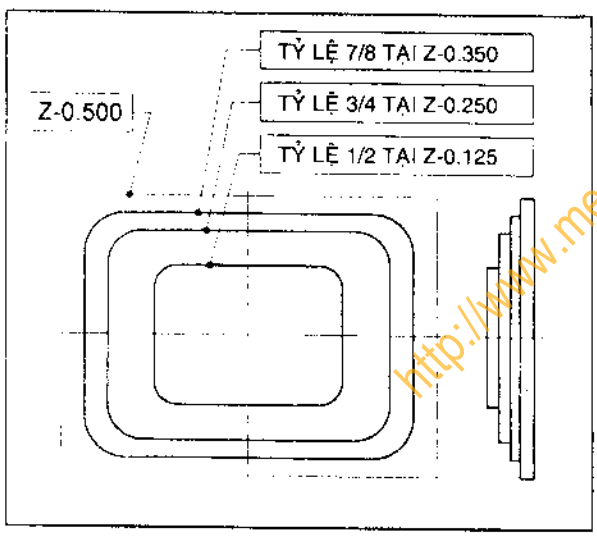
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

42.5 Biên dạng ban đầu. Hình 42.5 minh họa các chi tiết biên dạng với chiều sâu và các tỷ lệ mới. Chương trình bắt đầu từ tỷ lệ nhỏ nhất. Bạn hãy chú ý block quan trọng N712 và N713.



Hình 42.5. Biên dạng ban đầu với tỷ lệ toàn phần



Hình 42.6. Các biên dạng tỷ lệ ở ba chiều sâu

Tất cả các lệnh đều phải khởi đầu từ điểm gốc ban đầu.

```
O4203 (MAIN PROGRAM)
(SCALING FUNCTION - VERIFIED ON YASNAC I80)
(T01 = 1.0 DIA END MILL)
N1 G20
N2 G50                                  (SCALING OFF)
N3 G17 G40 G80 T01
N4 M06
N5 G90 G54 G00 X-1.0 Y-1.0 S2500 M03
N6 G43 Z0.5 H01 M08
N7 G01 Z-0.125 F12.0                    (SET DEPTH)
N8 G51 I2.0 J1.5 P0.5                (0.5X AT Z-0.125)
N9 M98 P7001                      (RUN NORMAL CONTOUR)
N10 G01 Z-0.25                          (SET DEPTH)
N11 G51 I2.0 J1.5 P0.75            (0.75X AT Z-0.250)
N12 M98 P7001                     (RUN NORMAL CONTOUR)
N13 G01 Z-0.35                          (SET DEPTH)
N14 G51 I2.0 J1.5 P0.875 (0.875X AT Z-0.350)
N15 M98 P7001                     (RUN NORMAL CONTOUR)
N16 M09
N17 G28 Z0.5 M05
N18 G00 X-2.0 Y10.0
N19 M30
%
```

```
O7001 (SUBPROGRAM FOR G51 SCALE)
(D51 = CUTTER RADIUS)
N701 G01 G41 X0 D51
N702 Y2.5 F10.0
N703 G02 X0.5 Y3.0 R0.5
N704 G01 X3.5
N705 G02 X4.0 Y2.5 R0.5
N706 G01 Y0.5
N707 G02 X3.5 Y0 R0.5
N708 G01 X0.5
N709 G02 X0 Y0.5 R0.5
N710 G03 X-1.0 Y1.5 R1.0
N711 G01 G40 Y-1.0 F15.0
N712 G50                                (SCALING OFF)
N713 X-1.0 Y-1.0  (RETURN TO ORIGINAL START)
N714 M99
%
```

Hàm lập tỷ lệ cung cấp nhiều khả năng. Bạn hãy kiểm tra các tham số điều khiển liên quan và bảo đảm chương trình phản ánh các xác lập điều khiển. Hiện có các khác biệt rõ rệt giữa các kiểu bộ điều khiển.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

43 PHỤ TÙNG MÁY TIỆN CNC

Máy CNC bất kỳ đều có thể được trang bị các phụ tùng để tăng hoặc mở rộng các tính năng của máy. Hầu hết các máy CNC đều có ít nhất vài phụ tùng là thiết bị tiêu chuẩn hoặc tùy chọn. Các trung tâm gia công có bàn máy quay và phân độ, giá đỡ, các đầu vuông góc, v.v... Tất cả đều là phụ tùng tương đối phức tạp, đòi hỏi thời gian để hiểu rõ nguyên lý và cơ chế hoạt động của chúng. Nhiều máy tiện còn được trang bị các phụ tùng với sự lập trình tương đối đơn giản. Các tính năng hoặc phụ tùng thuộc loại này có thể lập trình bao gồm:

- Điều khiển mâm cặp
- Ụ động
- Phân độ ổ dao hai chiều
- Bộ cấp phôi thanh

Nhiều tính năng khả dụng khác cũng có thể cho phép lập trình.

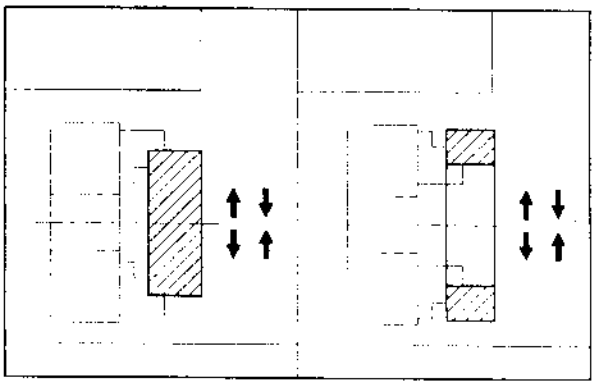
- Cơ cấu lấy chi tiết
- Thanh kéo chi tiết
- Thân và chuôi ụ động
- Gối đỡ cố định/di động
- Cữ chặn chi tiết
- ... và các loại khác tùy theo thiết kế máy

Một số phụ tùng loại này rất phổ biến, do đó cần nghiên cứu chúng một cách chi tiết cùng với vài ví dụ về ứng dụng lập trình.

ĐIỀU KHIỂN MÂM CẶP

Trong các thao tác bằng tay, mâm cặp, đai kẹp, hoặc đồ gá đặc biệt lắp trên đầu máy tiện thường mở và đóng khi người vận hành CNC nhấn bàn đạp chân. Vì các lý do an toàn, không thể mở mâm cặp đang quay, do được bảo vệ bằng khóa liên động an toàn đặc biệt. Tính năng quan trọng thứ hai của mâm cặp là thuật ngữ *mở* và *đóng* phụ thuộc vào phương pháp kẹp chặt, ngoài hoặc trong. Công tắc khóa được dùng để chọn kiểu mâm cặp (Hình 43.1).

Chú ý, các thuật ngữ *mở* và *đóng* liên quan với công tắc hai trạng thái hoặc công tắc khóa, được bố trí trên máy, thường ghi rõ CHUCK CLOSED (đóng mâm cặp). Có hai xác lập - IN và OUT.



Hình 43.1. *Kẹp chi tiết - ngoài và trong, chú ý xác lập của công tắc CHUCK CLOSED.*

Trong một số ứng dụng, chẳng hạn, cấp phôi thanh, cần mở và đóng mâm cặp với sự điều khiển của chương trình. Hai hàm M được dùng để mở và đóng mâm cặp.

Các hàm mâm cặp

Mặc dầu các số được gán (thường cho hàm M) có thể khác nhau giữa các máy, nhưng ứng dụng lập trình hoàn toàn như nhau. Một hàm xóa hàm kia. Các hàm M liên quan đến điều khiển mâm cặp bao gồm:

M10	Mở mâm cặp
M11	Đóng mâm cặp

➲ **Ví dụ:**

Quy trình lập trình phải có sự dừng trục chính và lệnh tạm dừng:

```
M05                (dừng trục chính)
M10                (mở mâm cặp)
G04 U0.1           (tạm dừng 01 giây)
M11                (đóng mâm cặp)
M03                (khởi động lại trục chính)
```

Đây là chuỗi thứ tự được đơn giản hóa, trong đó sự tạm dừng (G04) là thời gian để, ví dụ, phôi thanh đi qua vị trí dừng. Một số bộ cấp phôi thanh không yêu cầu dừng trục chính để cấp phôi, số khác có chương trình con đặc biệt để cấp phôi.

Các hàm M10 và M11 còn có thể sử dụng trên máy, trong khi gá lắp, sử dụng chế độ xác lập MDI trong chế độ vận hành bằng tay. Cuối chương này, các hàm M10 và M11 sẽ được sử dụng cho các ứng dụng liên quan với sự cấp phôi thanh.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Áp lực kẹp trong mâm cặp

Lực cần thiết để kẹp chi tiết trong mâm cặp được gọi là *áp lực kẹp*. Trên hầu hết các máy tiện CNC, áp lực này được điều khiển bằng van điều chỉnh, thường trong vùng ụ động. Sau khi xác lập áp lực kẹp, giá trị áp lực hầu như không thay đổi. Tuy nhiên, cũng có các công việc đòi hỏi áp lực kẹp *tăng* (kẹp chặt hơn) hoặc *giảm* (kẹp lỏng hơn) một cách thường xuyên. Các công việc cụ thể đó yêu cầu sự điều khiển áp lực kẹp có thể lập trình.

Vài nhà chế tạo máy tiện CNC cung cấp áp lực kẹp có thể lập trình. Nếu có điều đó, sẽ sử dụng hai hàm M *phi - tiêu chuẩn*, ví dụ:

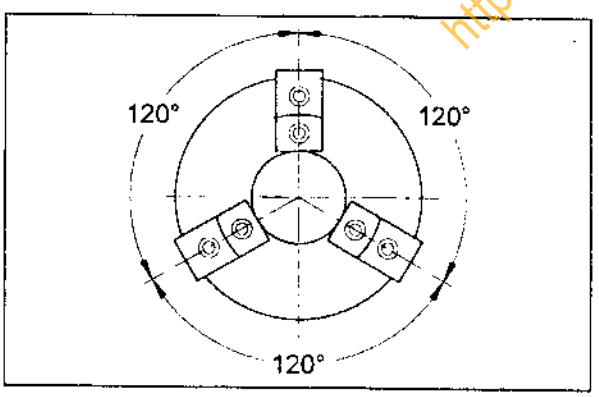
M15	Áp lực kẹp thấp
M16	Áp lực kẹp cao

Về nguyên tắc, chi tiết cần kẹp chặt lại trong mâm cặp trước khi hàm này có thể thay thế hàm kia, có thể làm lệch chi tiết trong thiết bị kẹp giữ. Nếu trên máy tiện có tính năng áp lực kẹp, bạn hãy xem kỹ tài liệu do nhà chế tạo máy tiện cung cấp.

Khi thay đổi áp lực kẹp - bằng tay hoặc thông qua chương trình - luôn luôn bảo đảm chi tiết được kẹp chặt một cách an toàn.

Chấu mâm cặp

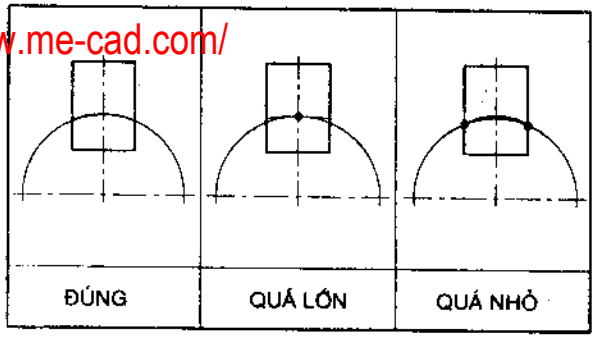
Phần này không liên quan trực tiếp với lập trình, nhưng rất hữu ích đối với nhà lập trình. Hầu hết các mâm cặp đều có ba chấu, cách nhau 120^0 (Hình 43.2).



Hình 43.2. *Mâm cặp ba chấu trên máy tiện CNC.*

Các chấu có thể *cứng* (có các răng cưa để kẹp chặt tốt hơn) hoặc *mềm* (thường được người vận hành điều chỉnh để thích hợp với đường kính chi tiết), nhưng chỉ có các chấu mềm mới có thể điều chỉnh.

Nói chung, các chấu cứng chỉ có hai loại, kẹp chặt ngoài hoặc trong. Các chấu mềm được thiết kế để *điều chỉnh* và khả năng thực hiện



Hình 43.3. *Đường kính các chấu mềm được điều chỉnh đúng (trái) và sai (phải)*

điều đó là yêu cầu cơ bản đối với người vận hành CNC. Có nhiều kỹ thuật để điều chỉnh các chấu mềm. Điều quan trọng là hiểu những điều có thể xảy ra nếu các chấu bị điều chỉnh không chuẩn.

Hình 43.3 minh họa ba khả năng - một, các chấu được điều khiển đúng đường kính, và hai trường hợp điều chỉnh sai. Nếu điều chỉnh sai, sự kẹp chặt, độ đồng tâm, hoặc cả hai đều không chuẩn.

Ụ ĐỘNG VÀ CHUÔI Ụ ĐỘNG

Ụ động là bộ phận rất phổ biến trên máy tiện CNC. Công dụng chính là đỡ chi tiết quá dài, quá lớn, hoặc cần ép chặt hơn vào mâm cặp, chẳng hạn trong một số nguyên công tiện thô. Ụ động còn có thể được sử dụng để gia công tinh các phôi ống mỏng, hoặc để đỡ chi tiết có phần kẹp chặt ngắn trong mâm cặp, để bảo đảm độ đồng tâm. Tuy nhiên, ụ động thường có thể là vật cản các chuyển động dao, do đó cần bảo đảm tránh va chạm. Ụ động có ba bộ phận chính:

- Thân ụ động
- Trục ụ động
- Chuôi hoặc mũi chống tâm.

Tất cả các bộ phận này đều quan trọng trong gá lắp và lập trình.

Thân ụ động

Thân là bộ phận nặng nhất của ụ động trên máy tiện, được lắp trên sống trượt máy tiện, bằng tay hoặc thông qua tùy chọn lập trình, bằng thủy lực. Ụ động lập trình thường chỉ khả dụng theo tùy chọn lắp đặt tại nơi chế tạo và phải đặt hàng khi mua máy.

Trục ụ động

Đây là ống trục dịch chuyển vào và ra bên trong thân ụ động, với khoảng hành trình xác định, chẳng hạn 3 inch trên các máy tiện cỡ

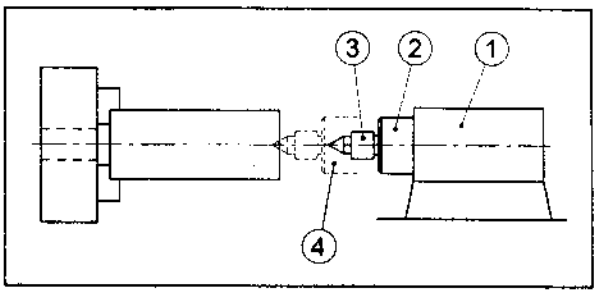
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

trung. Khi thân ụ động được lắp vào vị trí cố định trên sống trượt máy tiện, trục ụ động chuyển động *ra* để đỡ chi tiết, hoặc *vào*, để thay chi tiết. Chi tiết được đỡ bằng mũi tâm lắp trong trục ụ động.

Mũi tâm

Mũi tâm là bộ phận lắp trong trục ụ động với đầu côn, được đỡ bằng phần côn trong tương hợp và tiếp xúc với chi tiết. Tùy theo thiết kế, nếu ụ động có ổ đỡ bên trong có thể sử dụng mũi chống tâm chết (cố định), còn nếu không có ổ đỡ, phải dùng mũi tâm sống (quay). Các chi tiết gia công cần phải khoan tâm trước (trên máy CNC hoặc trước đó), sử dụng cùng góc dao như mũi tâm ụ động (thường là 60^0). Ụ động thông dụng được minh họa trên Hình 43.4.



Hình 43.4. *Ụ động trên máy tiện CNC. (1) Thân (2) Trục ụ động - RA (lùi lại để thay chi tiết). (3) Mũi tâm (4) Trục ụ động - VÀO (vị trí đỡ chi tiết)*

Các hàm trục ụ động

Trên hầu hết các máy tiện CNC, sự lập trình chuyển động trục ụ động là như nhau. Có hai hàm M hoạt động như nhau đối với thân ụ động lập trình và không lập trình. Hai hàm M đó như sau:

M12	Trục ụ động VÀO hoặc ON = hoạt động
M13	Trục ụ động RA hoặc OFF = không hoạt động

Nếu trục ụ động đỡ chi tiết, là vị trí VÀO, sử dụng hàm M12. Nếu trục ụ động không đỡ chi tiết, là vị trí RA, sử dụng hàm M13. Đối với gá lắp, có thể dùng các hàm M12 và M13, trên nhiều máy tiện, công tắc hai trạng thái (on/off) trên bộ điều khiển, được dùng để vận hành trục ụ động.

Trục chính phải ON khi trục ụ động đỡ chi tiết.

Ụ động lập trình

Thân ụ động thường không lập trình (chỉ lập trình trục ụ động), nhưng tính năng này khả dụng trên nhiều máy tiện CNC theo *tùy chọn được lắp đặt tại nơi chế tạo*. Điều đó có nghĩa là cần đặt hàng ụ động lập trình vào thời điểm đặt mua máy. Hiện có nhiều kiểu ụ động lập trình, chẳng hạn kiểu trượt chỉ dịch chuyển sang trái và phải, kiểu xoay, có thể xoay ra ngoài khi không cần sử dụng.

Ụ động được coi là *lập trình*, có thể lập trình sử dụng hai hàm M *phi-tiêu chuẩn*. Ví dụ, máy tiện CNC sử dụng hai hàm M:

M21	Thân ụ động tiến về phía trước
M22	Thân ụ động lùi về phía sau

Trên một số máy tiện CNC, có thể có thêm hai hàm M, một dùng để *kẹp chặt* ụ động, và một để *mở kẹp chặt*. Trong nhiều trường hợp, hai hàm ụ động này có các tính năng kẹp chặt/mở kẹp chặt được thiết lập sẵn.

Dưới đây là các bước lập trình để dịch chuyển ụ động về phía chi tiết, thực hiện sự gia công, và đưa ụ động trở về vị trí ban đầu. Thay vì giới thiệu ví dụ lập trình thực, ở đây chỉ nêu các bước có tính hướng dẫn, bạn có thể điền các hàm M cần thiết đối với máy CNC cụ thể:

1. Mở kẹp chặt thân ụ động
2. Dịch chuyển thân ụ động về phía trước
3. Kẹp chặt thân ụ động
4. Dịch chuyển trục ụ động vào chi tiết
5. *...thực hiện các nguyên công cắt gọt theo yêu cầu.*
6. Lùi trục ụ động ra khỏi chi tiết
7. Mở kẹp thân ụ động
8. Lùi thân ụ động về vị trí ban đầu
9. Kẹp chặt thân ụ động.

Một số quy trình đòi hỏi thời gian để hoàn tất, dù thời gian đó chỉ tính theo giây. Nói chung, nên lập trình hàm tạm dừng (dwell) để bảo đảm hoàn tất một bước trước khi bắt đầu bước kế tiếp (xem Chương 23).

Các vấn đề an toàn

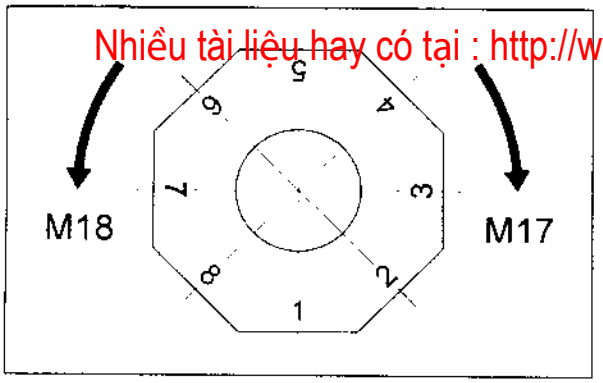
Khi lập trình gia công chi tiết có sử dụng ụ động, an toàn là rất quan trọng. Chuyển động dao hướng đến chi tiết ở đầu quỹ đạo dao và dao trở về vị trí thay dao là rất quan trọng. An toàn nhất là tiếp cận từ vị trí thay dao hướng đến chi tiết theo trục Z trước, *sau đó* là chuyển động theo trục X. Khi trở về từ vị trí gần chi tiết, cần thực hiện theo thứ tự ngược lại, trước hết lùi dao theo trục X phía trên chi tiết, *sau đó* chuyển động theo trục Z (cả hai trục thường dịch chuyển đến vị trí thay dao an toàn).

PHÂN ĐỘ Ổ DAO HAI CHIỀU

Tính năng hiệu quả kế tiếp là *phân độ ổ dao hai chiều*. Nhiều máy tiện CNC có *sự phân độ hai chiều* được xây dựng sẵn, đó là phương pháp phân độ ổ dao tự động (bộ điều khiển

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 43.5. Phân độ ổ dao hai chiều lập trình

quyết định chiều). Tuy nhiên, chắc chắn sẽ có lợi khi có chiều phân độ *lập trình*. Nếu tính năng này khả dụng trên máy tiện CNC, sẽ có hai hàm M để lập trình sự phân độ ổ dao. Cả hai hàm này đều phi - tiêu chuẩn (Hình 43.5).

Các hàm M dùng để phân độ ổ dao:

M17	Phân độ thuận	T01-T02-T03 ...
M18	Phân độ ngược	... T03-T02-T01

Trong ví dụ dưới đây, nhà lập trình làm việc với máy tiện có ổ dao có thể lắp 8 dao. Trong đó, dao T01 được dùng trước, sau đó là dao T08 và trở lại dao T01. Không có vấn đề gì khi viết chỉ số từ T01 đến T08 *hoặc* từ T08 đến T01, sử dụng chiều phân độ ổ dao tự động. Điều đó có nghĩa là, để đạt hiệu quả cao cần sử dụng phân độ ổ dao hai chiều. Tuy T01 và T08 có thể khác biệt về số nhưng chúng kế nhau trên ổ dao đa giác có 8 vị trí lắp dao. Hệ điều khiển sẽ *luôn luôn chọn phương pháp ngắn nhất*, trong trường hợp này, từ T01 đến T08 theo chiều ngược, sau đó từ T08 đến T01 theo chiều thuận.

Nếu máy không có tính năng phân độ hai chiều tự động, cần phải lập trình, nếu bộ điều khiển cho phép điều đó. Ngược lại, trong lập trình bình thường, khi chuyển từ T08 đến T01, chuyển động ổ dao phải *đi qua cả 6 vị trí lắp dao còn lại*, điều này là không hiệu quả. Ví dụ kế tiếp sẽ minh họa phương pháp và vị trí đặt hàm M trong chương trình.

Ví dụ lập trình

Ví dụ này là chương trình hoàn chỉnh, phối hợp tính năng phân độ hai chiều và phương pháp sử dụng ụ động lập trình toàn phần. Mọi chuyển động dao đều là thực nhưng không quá quan trọng trong ví dụ này. *Thứ tự đánh số các dụng cụ cắt trên ổ dao có thể khác nhau* giữa các máy. Thuật ngữ *chiều thuận* và *chiều ngược* liên quan với thứ tự đó. Ở đây sẽ sử dụng các hàm M đã nêu ở phần trên:

```
O4301
(BI-DIRECTIONAL INDEXING AND TAILSTOCK)
N1 G20 G99 M18              (SET INDEX BACKWARD)
N2 G50 S1200                      (LIMIT MAX RPM)
N3 T0100       (SHORT FROM T02 TO T01 WITH M18)
N4 G96 S500 M03
N5 G00 G41 X3.85 Z0.2 T0101 M08
N6 G01 Z0 F0.03
N7 X-0.07 F0.007
N8 G00 Z0.2
N9 G40 X10.0 Z5.0 T0100
N10 M01

N11 T0800      (SHORT FROM T01 TO T08 WITH M18)
N12 G97 S850 M03
N13 G00 X0 Z0.25 T0808 M08
N14 G01 Z-0.35 F0.005
N15 G04 U0.3
N16 G00 Z0.25
N17 X15.0 Z3.0 T0800
N18 M05           (SPINDLE STOP FOR TAILSTOCK)
N19 M01                          (OPTIONAL STOP)

N20 M21                        (TAILSTOCK FORWARD)
N21 G04 U2.0                         (2 SEC. DWELL)
N22 M12                                 (QUILL IN)
N23 G04 U1.0                         (1 SEC. DWELL)

N24 G50 M17    (NO MAX RPM - SET INDEX FORWARD)
N25 T0100      (SHORT FROM T08 TO T01 WITH M17)
N26 G96 S500 M03
N27 G00 G42 X3.385 Z0.1 T0101 M08
N28 G01 X3.685 Z-0.05 F0.008
N29 Z-2.5 F0.012
N30 U0.2
N31 G00 G40 X10.0 Z5.0 T0100
N32 M01                          (OPTIONAL STOP)

N33 T0200      (SHORT FROM T01 TO T02 WITH M17)
N34 G96 S600 M03
N35 G00 G42 X3.325 Z0.1 T0202 M08
N36 G01 X3.625 Z-0.05 F0.004
N37 Z-2.5 F0.006
N38 U0.2 F0.015
N39 G00 G40 X15.0 Z5.0 T0200
N40 M05           (SPINDLE STOP FOR TAILSTOCK)
N41 M01                          (OPTIONAL STOP)

N42 M13                                 (QUILL OUT)
N43 G04 U1.0                         (1 SEC. DWELL)
N44 M22                    (TAILSTOCK BACKWARD)
N45 G04 U2.0                         (2 SEC. DWELL,)
N46 M30                              (END OF PROGRAM)
%
```

Trong ví dụ này, T01 được dùng để vạt mặt phôi theo đường tâm trục chính, sau đó là T08, mũi khoan tâm, được dùng để gia công lỗ tâm. Khi mũi khoan tâm lùi ra khoảng trống, thân ụ động tiến tới và kẹp chặt, sau đó trục ụ động tiến vào chi tiết. T01 gia công thô phần vạt góc và đường kính, sau đó T02 gia công tinh phần vạt góc và đường kính. Khi hoàn tất gia công tinh, trục chính dừng lại, trục ụ động chuyển động ra, thân ụ động lùi lại. Người vận hành xác lập vị trí ụ động.

Ở cuối quá trình này, T02 ở vị trí hoạt động. Điều đó có nghĩa là M18 cần được lập trình ở *đầu* chương trình, để có thể phân độ ngắn nhất từ T02 đến T01.

Bạn hãy chú ý cách thức lập trình các hàm

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

M17 hoặc M18 - vị trí của chúng trong block là rất quan trọng! Các hàm này không phân độ ổ dao, chúng chỉ xác lập chiều. *Txx00 sẽ thực hiện sự phân độ thực tế.*

Toàn bộ điều này dẫn đến một câu hỏi - làm thế nào để biết máy tiện CNC có chiều phân độ tự động được thiết lập bên trong (chiều ngắn nhất) hay chiều có thể lập trình? Có một khả năng, trên các máy tiện CNC nơi chỉ xảy ra chiều *tiến* (không có sự phân độ tự động), đó là tính năng *chiều có thể lập trình*, khả dụng với dạng M17 và M18 hoặc các hàm tương tự.

Mặc dù xu hướng trên các máy tiện CNC hiện đại là tích hợp chiều phân độ ổ dao tự động vào hệ điều khiển (nghĩa là *hệ điều khiển đưa ra quyết định*), nhưng vẫn có một số ưu điểm khi phương pháp lập trình chiều phân độ khả dụng đối với các trường hợp gia công đặc biệt. Ví dụ, bạn hãy xét dao quá khổ lắp trên ổ dao. Dao này tuyệt đối an toàn khi *không phân độ* theo khoảng quay toàn phần của ổ dao. Sự phân độ tự động *không xét đến* tình huống đó.

Với sự phân độ lập trình, nhà lập trình có toàn quyền kiểm soát. Lập trình gá lắp theo cách thức không làm cho ổ dao phân độ đủ 360^0 vào thời điểm bất kỳ là hoàn toàn có thể được. Đây có thể không phải là trường hợp điển hình - chỉ cần thêm vài giây, nhưng có thể xảy ra thường xuyên.

ĐỒ GÁ CẤP PHÔI THANH

Cấp phôi thanh là đồ gá trên máy tiện CNC cho phép các chi tiết trụ nhỏ và trung bình có thể gia công liên tục, cho đến số lượng có thể gia công từ một thanh dài vài mét. Sử dụng bộ cấp phôi thanh có nhiều ưu điểm, đặc biệt là kiểu thiết kế thủy động lực hiện đại, thay vì kiểu cơ học cũ. Ví dụ, các nguyên công cưa sẽ được loại bỏ (thay bằng dao cắt đứt có độ chính xác cao hơn), không cần chỉnh đường kính cho các chấu mềm trong mâm cặp, quá trình gia công không cần sự can thiệp của người vận hành, lượng dư gia công nhỏ, và tốc độ trục chính cao, v.v...

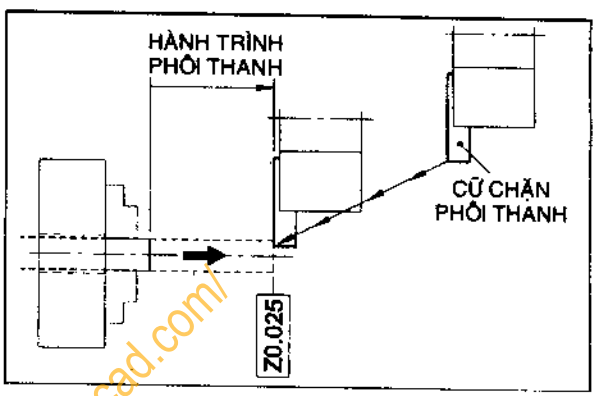
Các thanh phôi được đặt trong ống dẫn hướng đặc biệt (đẩy hoặc kéo phôi) từ ống đến khu vực gia công. Các hạn chế là chiều dài và đường kính thanh phôi, do nhà chế tạo bộ cấp phôi thanh quy định và đường kính lỗ trục chính của máy tiện CNC.

Hiện nay có nhiều thiết kế cấp phôi thanh và phương pháp lập trình chịu ảnh hưởng lớn từ thiết kế cấp phôi thanh cụ thể.

Các hàm điều khiển mở và đóng mâm cặp, hàm bỏ qua block, hàm M99 và nhiều hàm đặc biệt khác, là các công cụ khả dụng để lập trình cấp phôi thanh. Một số các hàm này đã được trình bày ở phần trước.

Cữ chặn phôi thanh

Tuy chuyển động của phôi thanh từ ống dẫn hướng được điều khiển bằng các hàm *mở* và *đóng* mâm cặp (M10 và M11), vẫn phải đưa ra vị trí đích của thanh, theo khoảng cách dịch chuyển ra khỏi ống dẫn hướng. Vị trí này phải thấp hơn đường kính thanh và về phía dương của trục Z (chẳng hạn .025 inch). Đây là lượng cần vạt mặt (Z0 ở mặt hai đầu) (Hình 43.6).



Hình 43.6. *Vị trí dừng đối với hành trình của phôi thanh.*

Chương trình rất đơn giản, sử dụng các hàm M10 và M11 nhưng còn hai hàm khác có *thể hoặc không được yêu cầu* đối với bộ cấp phôi thanh cụ thể. Các hàm M phi tiêu chuẩn này (trong ví dụ) như sau:

M71	Cấp phôi thanh ON - khởi động
M72	Cấp phôi thanh OFF - dừng

Các hàm này chỉ là ví dụ, có thể khác tùy theo cơ cấu cấp phôi thanh. Dưới đây là chương trình ví dụ:

```
O4302
N1 G20 T0100 M05          (T01 IS BAR STOPPER)
N2 G00 X0.125 Z0.025 T0101   (STOP POSITION)
N3 M10                          (CHUCK OPEN)
N4 G04 U1.0                   (1 SEC. DWELL)
N5 M71                      (BARFEEDER ON)
N6 G04 U2.0                   (2 SEC. DWELL)
N7 M11                         (CHUCK CLOSE)
N8 G04 U1.0                   (1 SEC. DWELL)
N9 M72                      (BARFEEDER OFF)
N10 G00 X10.0 Z5.0 T0100    (CLEAR POSITION)
N11 M01                      (OPTIONAL STOP)
```

Vài chú ý quan trọng liên quan đến cữ chặn phôi thanh có thể rất có ích để viết chương trình tốt hơn:

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Vị trí dao 1 (T01) kẹp giữ cữ chặn phôi thanh (N1).
- Ban đầu, sự kẹp chặt phôi thanh (cho từng chi tiết thứ nhất từ phôi thanh) được thực hiện bằng tay.
- Sự quay trục chính phải dừng lại trước khi mở mâm cặp.
- Mọi hàm M liên quan với sự cấp phôi thanh phải được lập trình trong từng block riêng rẽ.
- Sự tạm dừng phải đủ cho công việc nhưng không quá dài.

Trên đây là các xem xét tổng quát về lập trình cữ chặn phôi thanh, nhưng bạn cần kiểm tra lại quy trình ứng với thiết kế phôi thanh cụ thể.

CÁC TÙY CHỌN BỔ SUNG

Có nhiều tùy chọn khác (các tính năng phi - tiêu chuẩn) trên máy tiện CNC được coi là những phụ tùng có thể lập trình. Một số có thể hơi bất thường chẳng hạn *bộ chuyển tải phôi*, *áp lực ụ động* lập trình, số khác có thể phổ biến hơn, chẳng hạn *giá đỡ di động* (thay cho giá đỡ cố định), được dùng để đỡ phía trước dụng cụ cắt cho các chi tiết dài. Giá đỡ cố định và di động cho phép tránh hoặc giảm độ lệch trên chi tiết tương đối dài hoặc chi tiết có thành mỏng.

Hai phụ tùng khác, liên quan với nhau và với *sự cấp phôi thanh*, gồm:

- Bộ lấy sản phẩm, còn gọi là bộ đỡ tải chi tiết.
- Tay kéo ra ngoài

Cả hai đều được sử dụng chung với các thao tác cấp phôi thanh và dùng hai hàm M.

Bộ lấy sản phẩm

Phụ tùng rất phổ biến để gia công liên tục, sử dụng bộ cấp phôi thanh, là *bộ lấy sản phẩm*. Mục đích là lấy chi tiết hoàn tất sau khi được cắt rời. Thay vì để chi tiết rơi vào khu vực máy và có thể gây ra hư hỏng, bộ phận này sẽ tiếp nhận chi tiết một cách an toàn và chuyển chi tiết vào hộp chứa. Hộp chứa thường ở khu vực máy tiện CNC nơi người vận hành có thể với tới mà không bị nguy hiểm, và không có chất làm nguội. Có hai hàm M phi - tiêu chuẩn được dùng cho bộ lấy sản phẩm:

M73	Bộ lấy sản phẩm tiến tới	... vào hoặc tiến
M74	Bộ lấy sản phẩm lùi lại	... ra hoặc lùi

Ví dụ chương trình dưới đây minh họa phương pháp lập trình từng hàm M nêu trên cho dao cắt đứt.

```
O4303
N1 G20                          (TOP OF PROGRAM)
...
N81 T0700                       (PART-OFF TOOL ACTIVE)
N82 G50 S1500                   (LIMIT MAXIMUM RPM)
N83 G96 G52 G0 G03                (SPINDLE SPEED)
N84 G00 X2.2 Z-2.625 T0707 M08  (START POS.)
N85 M73                      (PART CATCHER ADVANCE)
N86 G01 X-0.01 F0.004          (PART-OFF MOTION)
N87 G00 X2.2 M09            (MOVE ABOVE STOCK DIA)
N88 X10.0 Z5.0 T0700          (SAFE XZ POSITION)
N89 M74                     (PART CATCHER RETRACT)
N90 M01                           (OPTIONAL STOP)
/ N91 N30              (CONTROLLED END OF PROGRAM)
N92 M99       (RESTART FROM THE TOP OF PROGRAM)
%
```

T07 là dao cắt đứt rộng .125, chứa thanh phôi ϕ2.0 theo chiều dài 2.5, đây là quy trình tiêu chuẩn. Trong chương trình, sử dụng một kỹ thuật lập trình đặc biệt, liên quan đến sự gia công liên tục. Bạn hãy tập trung chú ý vào ba block cuối, N90, N91, và N92.

Gia công liên tục

Block N90 là sự dừng tùy chọn, thường dùng để gá lắp và kiểm tra ngẫu nhiên. Block 91 chứa M30 hàm dừng chương trình. Chú ý ký hiệu vạch nghiêng ở phía trước block này. Đây là hàm *bỏ qua block*, đã đề cập trong Chương 22. Khi công tắc bỏ qua block trên bảng điều khiển được xác lập theo theo vị trí ON, hệ thống điều khiển sẽ *không* xử lý các lệnh trong block N91. Điều đó có nghĩa là chương trình sẽ không kết thúc ở đây và sự xử lý và tiếp tục ở block N92, nơi M99 được lập trình.

Mặc dầu hàm M99 chủ yếu được dùng để kết thúc chương trình con, nhưng cũng có thể được sử dụng trong chương trình chính (như trong ví dụ này). Trong trường hợp đó, M99 tạo ra vòng xử lý *liên tục*, làm cho chương trình trở lại từ đầu, và - không có sự gián đoạn sẽ lặp lại sự xử lý. Do dao thứ nhất thường có cữ chặn phôi thanh được lập trình, bộ cấp phôi thanh dịch chuyển phôi ra khỏi ống dẫn hướng và toàn bộ chương trình sẽ lặp lại, cho đến khi công tắc bỏ qua block trên bộ điều khiển được xác lập theo vị trí OFF. Khi đó M30 hoạt động và M99 trong block kế tiếp sẽ không được xử lý.

Bộ đếm chi tiết

Dạng gia công liên tục này thường sử dụng thêm một tính năng trên bộ điều khiển - *đếm số chi tiết*. Các chi tiết có thể được đếm thông qua chương trình (thường là user macro), hoặc bằng xác lập số lượng các chi tiết cần thiết trên hệ thống điều khiển. Chúng còn có thể được lập trình bằng các hàm M phi - tiêu chuẩn, ví dụ:

M88	Đếm thuận	... thứ tự tăng
M89	Đếm ngược	... thứ tự giảm

Số lượng xác lập sẵn đối với bộ đếm thường là dung lượng của phôi thanh hoặc số lượng chi

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tiết được làm ra. Một chương trình mẫu lập trình ở cuối chương này sẽ minh họa hàm đếm số lượng chi tiết và các tính năng khác.

Tay kéo ra

Tay kéo ra là thiết bị (phụ tùng máy tiện CNC) *kẹp và kéo* phôi thanh ra ngoài ống dẫn hướng cấp phôi thanh (trong khi mâm cặp mở). Đây là phương pháp phổ biến đối với các bộ cấp phôi thanh "*kiểu kéo*". Nói chung, tay kéo phôi được lắp trên ổ dao, như một "công cụ" riêng rẽ, hoặc bổ sung trên dao hiện hữu, theo thứ tự cho trước đối với số lượng vị trí dao khả dụng. Do các phụ tùng này không thể sử dụng khi quay trục chính, nhưng thường cần tốc độ cắt, chúng được lập trình trong chế độ G98-tốc độ cắt tính theo thời gian (in/min hoặc mm/min).

Bất kể kiểu loại tay kéo phôi khả dụng, các bước lập trình là như nhau - thanh chìa ra từ trục chính không dài hơn mặt trục chính sau khi cắt đứt:

01. Tại vị trí khởi đầu an toàn, phân độ vị trí dao nơi lắp tay kéo ra. Vào thời điểm này, trục chính phải dừng lại với M05.
02. Với tốc độ chạy dao nhanh, dịch chuyển đến đường tâm trục chính (X0), và vị trí trục Z khoảng một nửa chiều dài phôi thanh chìa ra từ trục chính.
03. Trong chế độ "ăn dao theo thời gian", ăn vào về phía phôi thanh như được chìa ra sau khi cắt đứt.
04. Tạm dừng khoảng 0.5 giây để bộ kéo ra kẹp chặt phôi thanh.
05. Mở mâm cặp với M10.
06. Kéo phôi thanh ra ngoài từ ống dẫn hướng.
07. Tạm dừng khoảng 0.5 giây để bộ kéo hoàn tất sự kéo ra.
08. Đóng mâm cặp với M11.
09. Tạm dừng khoảng 1 giây để đóng chặt mâm cặp.
10. Dịch chuyển bộ kéo phôi ra xa phôi thanh.
11. Đưa bộ kéo phôi trở về vị trí khởi đầu an toàn.
12. Trở lại chế độ "ăn dao theo số vòng quay"

Trong lập trình, cấu trúc tổng quát sẽ tương tự định dạng nêu trên.

```
O4304
...
N.. ...
N.. Txx.. M05                    (ITEM 01)
N.. G00 X0 Z..                   (ITEM 02)
N.. G98 G01 Z.. F..              (ITEM 03)
N.. G04 U0.5                     (ITEM 04)
N.. M10                          (ITEM 05)
N.. G01 Z.. F..                  (ITEM 06)
N.. G04 U0.5                     (ITEM 07)
N.. M11                          (ITEM 08)
N.. G04 U1.0                     (ITEM 09)
N.. G00 Z..                      (ITEM 10)
N.. X.. Z.. Txx00                (ITEM 11)
N.. G99                          (ITEM 12)
N.. ...
```

Bạn hoàn toàn có thể chỉnh sửa cấu trúc chương trình để phù hợp với các yêu cầu của gá lắp cụ thể trong xưởng cơ khí của bạn.

VÍ DỤ LẬP TRÌNH

Ví dụ dưới đây minh họa chương trình hoàn chỉnh gia công tự động cấp phôi thanh, cho đến khi gia công đủ số lượng chi tiết. Người vận hành xác lập số lượng chi tiết cần thiết khi bắt đầu thanh phôi mới. Chương trình này cần được nghiên cứu cẩn thận, do có một số tính năng cao cấp và rất thực tiễn, tất cả đều đã được trình bày, chủ yếu trong chương này:

```
O4305
(N1 TO N18 FOR NEW BAR ONLY - 1.5 CUT-OFF)
N1 M18                          (INDEX T03 TO T01)
N2 G20 T0100 M05               (T01 - BAR STOPPER)
N3 G00 X0.1 Z1.5 T0101          (NEW BAR OUT 1.5)
N4 M10                             (CHUCK OPEN)
N5 G04 U1.0                       (1 SEC. DWELL)
N6 M71                          (BARFEEDER ON)
N7 G04 U1.0                       (1 SEC. DWELL)
N8 M11                            (CHUCK CLOSE)
N9 X5.0 Z2.0 T0100             (CLEAR POSITION)
N10 M01                         (OPTIONAL STOP)

N11 M17                        (INDEX T01 TO T03)
N12 T0300    (T03 - 0.125 WIDE PART-OFF TOOL)
N13 G97 S1400 M03                 (CUTTING RPM)
N14 G00 X1.25 Z0 T0303 M08     (START POSITION)
N15 G01 X-0.02 F0.004        (PART-OFF BAR END)
N16 G00 X1.25 M09              (MOVE ABOVE BAR)
N17 X5.0 Z2.0 T0300            (CLEAR POSITION)
N18 M01                         (OPTIONAL STOP)

N19 M18                        (INDEX T03 TO T01)
N20 T0100 M05                 (T01 - BAR STOPPER)
N21 G00 X0.1 Z0.05 T0101 (0.05 STOCK ON FACE)
N22 M10                            (CHUCK OPEN)
N23 G04 U1.0                      (1 SEC. DWELL)
N24 M71                         (BARFEEDER ON)
N25 G04 U1.0                      (1 SEC. DWELL)
N26 M11                           (CHUCK CLOSE)
N27 X5.0 Z2.0 T0100            (CLEAR POSITION)
N28 M01                         (OPTIONAL STOP)
N29 M17                        (INDEX T01 TO T02)
N30 T0200        (T02 - FACE-CHAMFER-TURN OD)
N31 G96 S400 M03                  (CUTTING SPEED)
N32 G00 G41 X1.25 Z0 T0202 M08    (START FACE)
N33 G01 X-0.07 F0.007          (FACE-OFF FRONT)
N34 G00 Z0.1                          (CLEAR Z+)
N35 G42 X0.67                  (CHAMFER START)
N36 G01 X0.92 Z-0.025 F0.003    (CUT CHAMFER)
N37 Z-1.26 F0.01                (CUT DIAMETER)
N38 U0.2 F0.02                 (CLEAR ABOVE BAR)
N39 G00 G40 X5.0 Z2.0 T0200 (CLEAR POSITION)
N40 M01                         (OPTIONAL STOP)

N41 T0300    (T03 - 0.125 WIDE PART-OFF TOOL)
N42 G97 S1400 M03                 (CUTTING RPM)
N43 G00 X1.25 Z-1.125 T0303 M08   (START POS.)
```

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N44 G01 X1.1 F0.004          (PART CUTOFF)
N45 G00 X1.25                (MOVE ABOVE BAR)
N46 X5.0 Z2.0 T0300          (CLEAR POSITION)
N47 M01                      (OPTIONAL STOP)
N48 M89          (INCREASE PART COUNTER BY 1)
/ N49 M30         (CONTROLLED END OF PROGRAM)
N50 M99 P19          (RESTART FROM BLOCK N19)
%
```

Đối với các phụ tùng và các tùy chọn, nhà chế tạo máy công cụ sử dụng một số hàm M để kích hoạt và khử kích hoạt phụ tùng cụ thể. Không thể trình bày quy trình đặc biệt bất kỳ trong tài liệu tham khảo tổng quát. Hy vọng các ý tưởng được giới thiệu trong chương này sẽ có ích và hữu dụng đối với các đề nghị của nhà chế tạo và giúp hiểu tốt hơn các đề nghị đó.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

44 PHAY XOẮN

Phay xoắn sử dụng tính năng hệ điều khiển gọi là *nội suy xoắn*. Theo định nghĩa đơn giản nhất, nội suy xoắn là thao tác trong đó nội suy tròn sử dụng *đồng thời ba trục*. Điều này có thể dẫn đến hiểu sai do ngụ ý cung hoặc hình tròn ba chiều. Cung hoặc hình tròn kiểu đó không có trong toán học. Trong thực tế, lệnh nội suy tròn G02 hoặc G03 sử dụng *cả ba* trục, ví dụ:

```
G03 X.. Y.. Z.. F..
```

Kiểu thao tác này chỉ khả dụng trên các trung tâm gia công CNC như một tính năng tùy chọn. Phần kế tiếp sẽ trình bày chủ đề phay xoắn chi tiết hơn.

NGUYÊN CÔNG PHAY XOẮN

Phay xoắn là gì? Về cơ bản, đây là dùng nội suy tròn, kỹ thuật lập trình, để gia công các cung và đường tròn *kết hợp* với nội suy tuyến tính trong một block, trong cùng một chuyển động.

Các chương trước, liên quan với nội suy tròn, đã giới thiệu một tính năng chính của chủ đề này. Trong nội suy tròn, có hai trục chính được dùng trong mặt phẳng đã chọn, với ý đồ lập trình chuyển động cung hoặc chuyển động tròn.

Ví dụ, trong mặt phẳng G17 XY (mặt phẳng phổ biến nhất), định dạng của nội suy tròn có hai dạng:

➲ **Sử dụng các tâm cung IJK cho chuyển động CW và CCW:**

```
G02 X.. Y.. J.. F..
G03 X.. Y.. J.. F..
```

➲ **Sử dụng bán kính R cho chuyển động CW và CCW:**

```
G02 X.. Y.. R.. F..
G03 X.. Y.. R.. F..
```

Chú ý, ở đây trục Z không được lập trình, nếu đưa trục Z vào cùng một block chứa lệnh phay đường tròn, sẽ không hoạt động một cách bình thường. Thực tế block này sẽ không hoạt động, *trừ khi* hệ điều khiển có tính năng đặc biệt được gọi là tùy chọn *nội suy xoắn*.

Nội suy xoắn

Nội suy xoắn thường là tùy chọn hệ điều khiển đặc biệt được thiết kế để sử dụng cho chế độ cắt đường tròn hoặc cung với chiều thứ ba. Chiều thứ ba luôn luôn do mặt phẳng hoạt động xác định.

- Trong mặt phẳng G17XY - chiều thứ ba là trục Z
- Trong mặt phẳng G18ZX - chiều thứ ba là trục Y
- Trong mặt phẳng G19YY - chiều thứ ba là trục X

Trong mặt phẳng đặt biệt G17 (XY), chiều thứ ba là trục Z. Trong mặt phẳng hoạt động G18 (ZX), chiều thứ ba là trục Y, và trong mặt phẳng hoạt động G19 (YZ), chiều thứ ba là trục X.

Trong mọi trường hợp, chiều thứ ba - *chuyển động trục thứ ba* - sẽ *luôn luôn* là chuyển động tuyến tính, vuông góc với mặt phẳng hoạt động.

Từ nội dung nêu trên, có thể định nghĩa nội suy xoắn như sau:

> Nội suy xoắn là chuyển động tròn đồng thời trên hai trục trong mặt phẳng làm việc, với chuyển động tuyến tính dọc theo trục còn lại.

Chuyển động ba trục này được hệ điều khiển đồng bộ hóa với tất cả các trục đến vị trí đích cùng một thời điểm.

Định dạng lập trình

Các định dạng *tổng quát* đối với nội suy xoắn trong chương trình, tương tự các định dạng khả dụng cho nội suy tròn. Sự chọn mặt phẳng là *rất* quan trọng:

➲ **Sử dụng các tâm cung IJK cho chuyển động CW và CCW:**

```
G02 X.. Y.. Z.. I.. J.. K.. F..
G03 X.. Y.. Z.. I.. J.. K.. F..
```

➲ **Sử dụng bán kính R cho chuyển động CW và CCW**

```
G02 X.. Y.. Z.. R.. F..
G03 X.. Y.. Z.. R.. F..
```

Sự chọn mặt phẳng lập trình trước block nội suy xoắn xác định các trục sẽ hoạt động trong chương trình và chức năng của chúng.

Các vector cung đối với nội suy xoắn

Các hàm vector cung được lập trình sử dụng cùng các nguyên tắc như nội suy tròn nhưng sẽ *khác nhau đối với từng mặt phẳng*. Dưới đây là bảng tóm tắt.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Mặt phẳng hoạt động	Chuyển động tròn	Chuyển động tuyến tính	Vector cung
G17	X và Y	Z	I và J
G18	X và Z	Y	I và K
G19	Y và Z	X	J và K

Chú ý, các vector cung áp dụng cho hai trục tạo thành *chuyển động tròn* - chuyển động tuyến tính hoàn toàn không có ảnh hưởng. Nếu hệ điều khiển hỗ trợ nhập bán kính trực tiếp R (thay vì các vector IJK), tâm của chuyển động cung được tính toán một cách tự động, trong mặt phẳng hiện hành.

Ứng dụng

Mặc dầu tùy chọn nội suy xoắn không phải là phương pháp lập trình thông dụng, nhưng có thể là phương pháp duy nhất cho một số ứng dụng gia công đặc biệt:

- Phay ren
- Phay biên dạng xoắn
- Tạo hốc xoắn

Từ ba nhóm này, *phay ren* là phương pháp phổ biến nhất của nội suy xoắn được áp dụng trong công nghiệp. Hai nhóm ứng dụng còn lại tương tự nhau, nhưng ít phổ biến hơn.

PHAY REN

Có hai phương pháp gia công ren thông dụng trên máy CNC. Trên trung tâm gia công CNC, phương pháp gia công ren chủ yếu là ta rô ren, thường sử dụng các chu kỳ cố định G84 hoặc G74. Trên máy tiện CNC, cũng sử dụng ta rô ren (không dùng chu kỳ), nhưng hầu hết các ren được gia công bằng phương pháp tiện ren, sử dụng phương pháp block với G32, chu kỳ đơn giản G92, và chu kỳ lặp G76.

Áp dụng phay ren

Có nhiều trường hợp, trong đó ta rô ren hoặc gia công ren với dao một lưỡi cắt là không thực tiễn, khó khăn,hoặc không thể thực hiện trong tình huống cụ thể. Có thể khắc phục các khó khăn đó bằng cách chọn *phương pháp phay ren*. Phay ren có lẽ là ứng dụng công nghiệp phổ biến nhất của tính năng nội suy xoắn trên bộ điều khiển.

Có thể sử dụng phay ren trong lập trình để đạt được kết quả mong muốn. Các ưu điểm của phay ren bao gồm:

- Đường kính ren lớn hầu như mọi đường kính bất kỳ đều có thể được phay ren (với độ đồng tâm cao).
- Tạo ra ren đều và chính xác (chỉ có mài ren là có độ chính xác cao hơn).
- Kết hợp phay ren trong một lần gá lắp, loại bỏ các thao tác phụ.
- Có thể cắt ren chiều sâu toàn phần.
- Không cần dùng mũi ta rô.
- Sự ta rô ren là không thực tiễn.
- Ta rô ren khó và gây ra các vấn đề.
- Không thể ta rô ren trong các vật liệu cứng.
- Ta rô ren lỗ cụt gây ra các vấn đề.
- Chi tiết không thể quay trên máy tiện CNC.
- Có thể gia công ren phải và ren trái bằng một dao.
- Có thể gia công ren trong và ren ngoài bằng một dao.
- Giảm hoặc không có ba-via khi cắt ren.
- Chất lượng bề mặt cao, đặc biệt với các vật liệu mềm.
- Kéo dài tuổi bền của dao cắt ren.
- Không cần dùng các đầu ta rô đắt tiền.
- Không cần dùng các mũi ta rô lớn.
- Không cần đảo chiều trục chính (như khi ta rô).
- Định mức công suất của dao tốt hơn.
- Một cán dao có thể tiếp nhận các mảnh chắp với bước ren khác nhau.
- Giảm chi phí gia công ren.

Phay ren chỉ bổ sung mà không thay thế các nguyên công cắt ren khác, sử dụng dao cắt ren đặc biệt, được gọi là *dao phay ren* nhiều răng, trong đó *bước ren được thiết lập ngay trên dao cắt*.

Điều kiện phay ren

Để phay ren thành công, cần có ba điều kiện trước khi viết chương trình:

- Hệ điều khiển phải hỗ trợ nguyên công phay ren.
- Phải gia công trước đường kính cắt ren.
- Phải chọn dao phay ren thích hợp.

Ba điều kiện này phải được đáp ứng đồng thời.

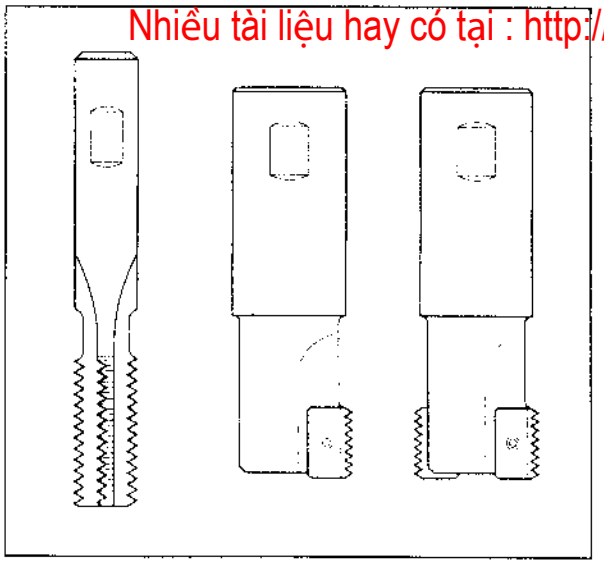
Dao phay ren

Dao phay ren ít nhất có hai dạng - một số được chế tạo từ carbide nguyên khối, số khác sử dụng các mảnh chắp. Trong các thiết kế đó, bước dao phải tương ứng với bước ren trên bản vẽ. Dao phải đủ nhỏ để khớp với không gian trong khả dụng và đủ lớn để bảo đảm độ cứng khi cắt ren ngoài, đối với phay ren trong, dao cắt khả dụng để phay ren trong các lỗ nhỏ đến .250 inch (6.35 mm).

Khác với ta rô, dao phay ren không có góc xoắn, chỉ có bước. Góc xoắn được yêu cầu đối với cắt ren và được kiểm soát trong khi chuyển

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 44.1. Dao phay ren. Carbide nguyên khối (trái), một mảnh chắp (giữa), và hai mảnh chắp (phải).

động nội suy xoắn bằng chuyển động tuyến tính. Các dao phay ren được minh họa trên Hình 44.1.

Các yêu cầu gia công trước

Lỗ để ta rô không thể có cùng đường kính với mũi ta rô, phải nhỏ hơn để *bảo đảm chiều sâu ren*. Nguyên tắc này cũng áp dụng cho phay xoắn.

- Nếu phay ren trên đường kính trong của chi tiết (ren trong), đường kính gia công trước phải nhỏ hơn cỡ ren danh định.
- Nếu phay ren trên đường kính ngoài (ren ngoài), đường kính gia công trước phải bằng cỡ ren danh định.

Đường kính (trong hoặc ngoài) có thể hơi lớn hơn hoặc nhỏ hơn kích cỡ "danh định", độ lệch này được quyết định theo yêu cầu "lắp ghép" ren.

Bán kính khoảng hở

Bán kính khoảng hở bảo vệ ren, tránh hư hỏng do dụng cụ cắt. Mỗi lưỡi cắt trên dụng cụ hoặc mảnh chắp đều được mài với *góc giảm* theo chiều cắt - được gọi là *góc hở*. Góc này bảo đảm các điều kiện cắt ổn định trong khi phay ren.

Năng suất phay ren

Một trong các lý do lựa chọn nguyên công phay ren là mong muốn cải thiện năng suất gia công. Có nhiều kích cỡ dao phay ren khả dụng với cỡ bước. Để đạt được hiệu quả cao nhất, bạn hãy dùng dao phay ren đủ lớn cắt ren trong một vòng quay. Đồng thời, dao phải có tất cả các khoảng hở cần thiết.

Tốc độ cắt, độ mảnh trình và lựa chọn tốc độ cắt có ảnh hưởng lớn đến năng suất phay ren. Dao đường kính lớn có thể phay hiệu quả hơn (tốc độ cắt cao) nhưng không vừa trong những nơi chật hẹp. Dao đường kính nhỏ có tác dụng ngược lại - có thể dùng ở nơi chật, nhưng tốc độ cắt thấp. Dao nhỏ còn có thể sử dụng với tốc độ trục chính cao hơn và tốc độ cắt tương ứng - do đó có thể giảm thời gian chu kỳ cắt gọt.

ĐƯỜNG XOẮN

Có thể định nghĩa đường xoắn như sau:

"*Đường xoắn là đường cong được tạo ra khi một điểm chuyển động xung quanh mặt trục (tưởng tượng hoặc thực) với tốc độ không đổi theo chiều trục của hình trụ đó.*"

Định nghĩa này có nghĩa là đường xoắn là *đường cong được tạo ra bằng chuyển động tròn của một điểm trên mặt trụ hoặc côn, kết hợp đồng thời với sự tịnh tiến*. Đường cong ren vít là ví dụ điển hình của đường xoắn trên mặt trụ.

Chuyển động của dao cắt dựa trên định nghĩa toán học (sử dụng ba trục), tạo ra chuyển động xoắn, còn gọi là nội suy xoắn.

Đường xoắn là chuyển động gia công có *bốn* biến thể:

- Cắt tròn thuận chiều (CW) với chuyển động tuyến tính dương.
- Cắt tròn thuận chiều (CW) với chuyển động tuyến tính âm.
- Cắt tròn ngược chiều (CCW) với chuyển động tuyến tính dương.
- Cắt tròn ngược chiều (CCW) với chuyển động tuyến tính âm.

Đường cong xoắn trên Hình 44.2 minh họa đường xoắn (vật thể ba chiều) theo bốn hình chiếu tiêu chuẩn. Đường cong này được biểu diễn theo các hình chiếu:

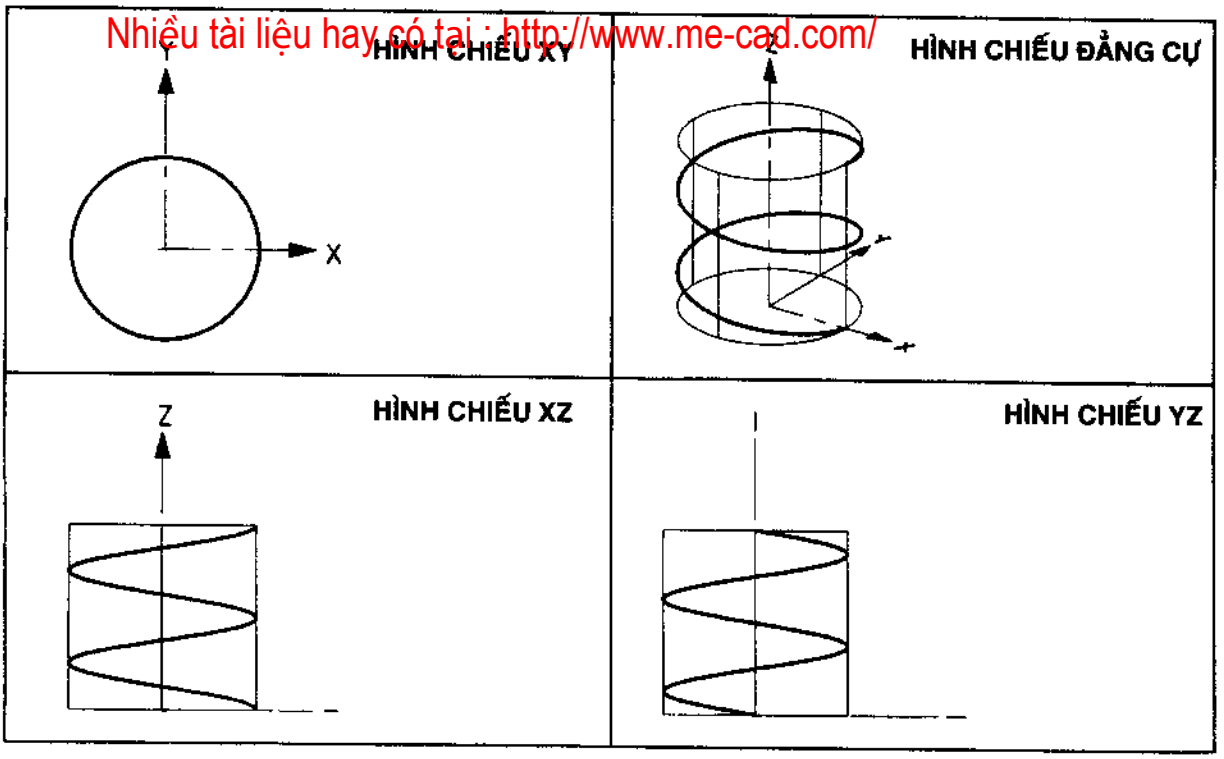
- Hình chiếu bằng (XY), chỉ thấy đường tròn.
- Hình chiếu trước (XZ), đường cong xoắn nhìn từ mặt trước tiêu chuẩn.
- Hình chiếu cạnh (YZ), đường cong xoắn nhìn từ mặt bên phải tiêu chuẩn.
- Hình chiếu đẳng cự (XYZ), biểu thị đường xoắn hai vòng theo ba chiều.

Hình chiếu đường cong xoắn rất hữu ích là *hình chiếu phẳng* (còn gọi là *hình khai triển*), thường dùng để biểu thị đường xoắn theo vật thể phẳng có bọc xung quanh hình trụ. Hình 44.3 minh họa hình khai triển phẳng của đường cong xoắn (một vòng).

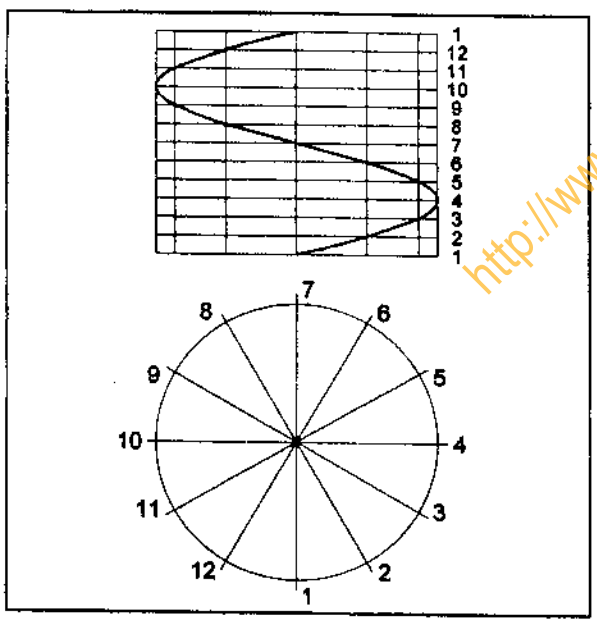
http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

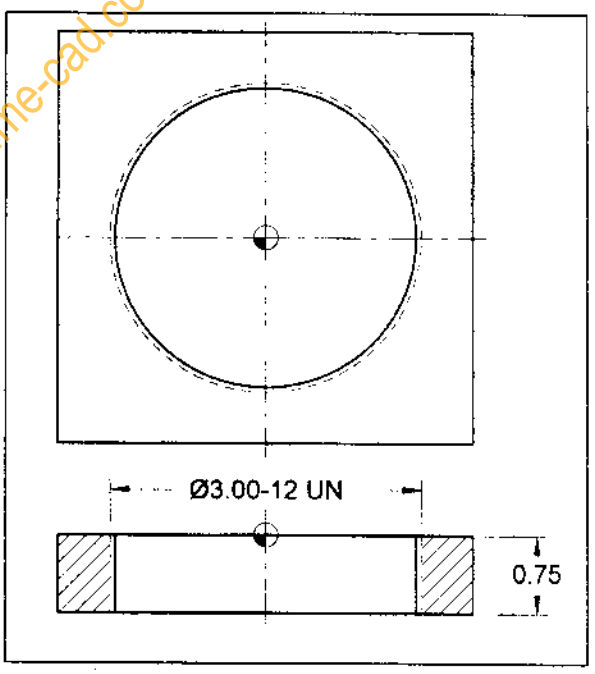
Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 44.2. *Đường cong xoắn biểu diễn trên bốn hình chiếu - chỉ nêu hai vòng quay giữa đỉnh và đáy của đường cong xoắn.*



Hình 44.3. *Khai triển đường cong xoắn phải với một vòng 360°.*



Hình 44.4. *Ví dụ phay ren trong, chương trình O4401*

VÍ DỤ PHAY REN

Nguyên công phay ren trên trung tâm gia công CNC có thể được lập trình rất hiệu quả bằng cách sử dụng tính năng nội suy xoắn trên bộ điều khiển. Phương cách đơn giản nhất và dễ hiểu nhất để trình bày và giải thích chế độ phay ren là sử dụng ví dụ minh họa (Hình 44.4).

Ren thẳng

Thông tin dưới đây là dữ liệu ban đầu, dựa trên bản vẽ và các dao cắt hiện có:

1. Ren trong có đường kính 3.00 inch - suốt qua phôi tấm.
2. Chiều dày tấm 0.75 inch.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

3. 12 TPI, 12 ren/inch.
4. Dao phay ren đường kính 1.500.
5. Dao TO3, và các chế độ bù H03, D03.
6. Đường kính lỗ 2.9000 inch.

Đây là các số liệu cơ bản, từ đó có thể bắt đầu lập trình.

Các tính toán ban đầu

Trong ví dụ này có sáu điều cần khảo sát. Các mục 1, 2 và 3 là từ bản vẽ, còn các mục 4, 5, 6 được lựa chọn hoặc tính toán như một phần trong khi lập trình. Bạn nên xem xét các mục được chọn hoặc tính toán một cách riêng rẽ.

Mục 4 là kích cỡ dao phay ren. Cần xét hai đặc tính của dao phay ren, đường kính và *bước* lưỡi cắt (các răng). Lựa chọn đường kính dao phải được thực hiện cẩn thận, giá trị phải nhỏ hơn đường kính *lỗ*. Thứ hai là chọn dao phay ren có số răng/inch thích hợp (bước). Đường kính phay ren là quan trọng hơn đối với ren trong, nhưng bước dao phay ren phải được duy trì, bất kể cắt ren trong hay ren ngoài.

Mục 5 là chỉ số dao và các chỉ số bù liên quan. Trong trường hợp này, chỉ số dao là 3, được lập trình theo T03. Chỉ số bù chiều dài dao H03 và bù bán kính dao D03. Xác lập bù D03 sẽ chứa *bán kính* dao cắt ren, trong trường hợp này, giá trị danh nghĩa sẽ là .6250. Trong ví dụ này các chỉ số bù là tùy ý, thực tế có thể khác. Bạn chỉ cần nhớ đường kính được gia công đối với cắt ren trong *phải* nhỏ hơn đường kính ren danh định - tương tự khoan lỗ để ta-rô ren.

Mục 6 cho biết đường kính lỗ là 2.9000 inch. Tại sao là số này mà không phải là số khác? Bạn cần nhớ chiều sâu ren trong được thiết lập từ công thức. Công thức tổng quát để tính chiều sâu D của ren trong là nhân hằng số với bước ren:

D = BƯỚC x .54127

Nếu áp dụng công thức này cho ∅3.000, với ren 12 TPI (1/12 = .0833333 bước), chiều sâu ren sẽ là:

.0833333 x .54127 = .0451058

Khi áp dụng để tính đường kính lỗ gia công trước, phải lấy đường kính danh định *trừ đi hai lần* giá trị này:

3.0000 - 2 x .0451058 = 2.9097884

Do đó, đường kính lỗ đối với ren này sẽ là ∅2.9098

Tại điểm này, cần xem xét tiếp một vấn đề, đó là dao phay tròn. Về cơ bản, đây là dao tạo hình, đỉnh và đáy dao sẽ tạo thành trên ren hoàn tất. Tính năng này có một ưu điểm. Bằng cách lập trình đường kính trong hơi nhỏ hơn, kích cỡ hoàn chỉnh sẽ được hình thành với độ bóng bề mặt cao. Để lại lượng dư khoảng .003 đến .006 trên một phía sẽ có tác dụng tốt. Ví dụ, khoảng .003 - .006 được sử dụng và giá trị tính toán ∅2.9097884 có thể được làm tròn đến ∅2.9, chỉ để lại lượng dư .0097844 trên đường kính, tương ứng .0048942 theo một phía, để gia công tinh.

Vị trí bắt đầu

Sau khi thu thập và tính toán mọi dữ liệu cần thiết, bước thứ hai là tính toán vị trí bắt đầu gia công ren.

Điều này là dễ dàng đối với các trục X và Y tâm đường kính ren là vị trí khởi đầu thuận lợi. Trong ví dụ này, và để đơn giản, vị trí XY này còn tương đương với vị trí X0Y0.

Điểm bắt đầu của dao phay ren được đo dọc theo trục Z trong phay xoắn quan trọng hơn nhiều so với các kiểu phay khác. Vị trí bắt đầu trên trục Z phải luôn luôn *đồng bộ* với bước ren, khi cắt gọt đồng thời theo cả ba trục. Zero của trục này (ZO) phải ở trên mặt phôi tấm.

Vị trí bắt đầu trên trục Z được xác định bằng vài yếu tố - *kích cỡ dao phay ren*, *bước ren*, *chiều chuyển động theo trục Z* (xuống hoặc lên), và *phương pháp ăn dao vào* theo các trục XY.

Khi cắt ren sử dụng tính năng nội suy xoắn, cả ba trục được khảo sát như nhau. Phương pháp tiếp cận cung nội suy xoắn hoàn toàn như phương pháp tiếp cận cung nội suy tròn - quy trình là như nhau.

Chiều và chuyển động quay

Trong nội suy xoắn, điều *cực kỳ quan trọng* là phối hợp, đồng bộ hóa ba yếu tố:

- Sự quay trục chính
- Chiều cắt gọt tròn
- Chiều chuyển động trục Z

Tại sao ba yếu tố này lại quan trọng? Tại sao phải phối hợp đồng bộ cả ba? Bạn hãy lần lượt đánh giá từng yếu tố đó.

Sự quay trục chính

Sự quay trục chính có thể là M03 (thuận chiều, CW) hoặc M04 (ngược chiều, CCW).

Chiều cắt tròn

Chiều chuyển động cắt tròn tuân theo các

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

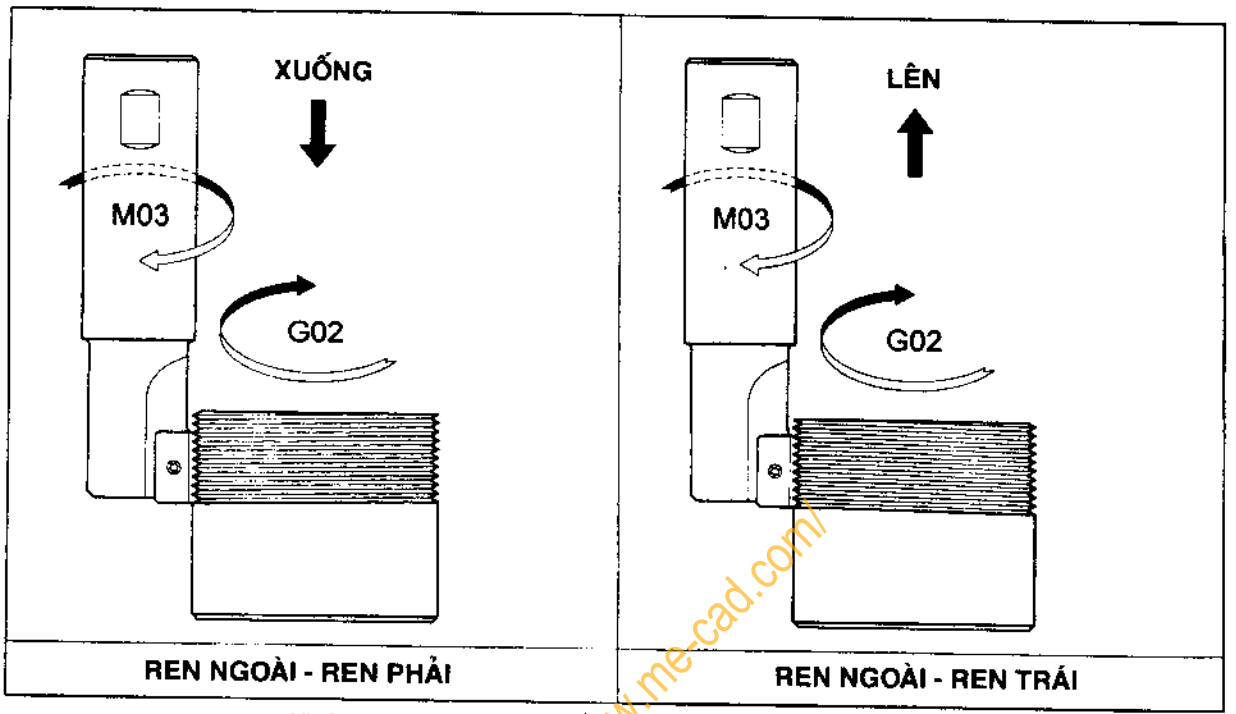
nguyên tắc quy ước, G02 tuân theo chiều CW, G03 ngược chiều CCW.

Chiều chuyển động trục Z

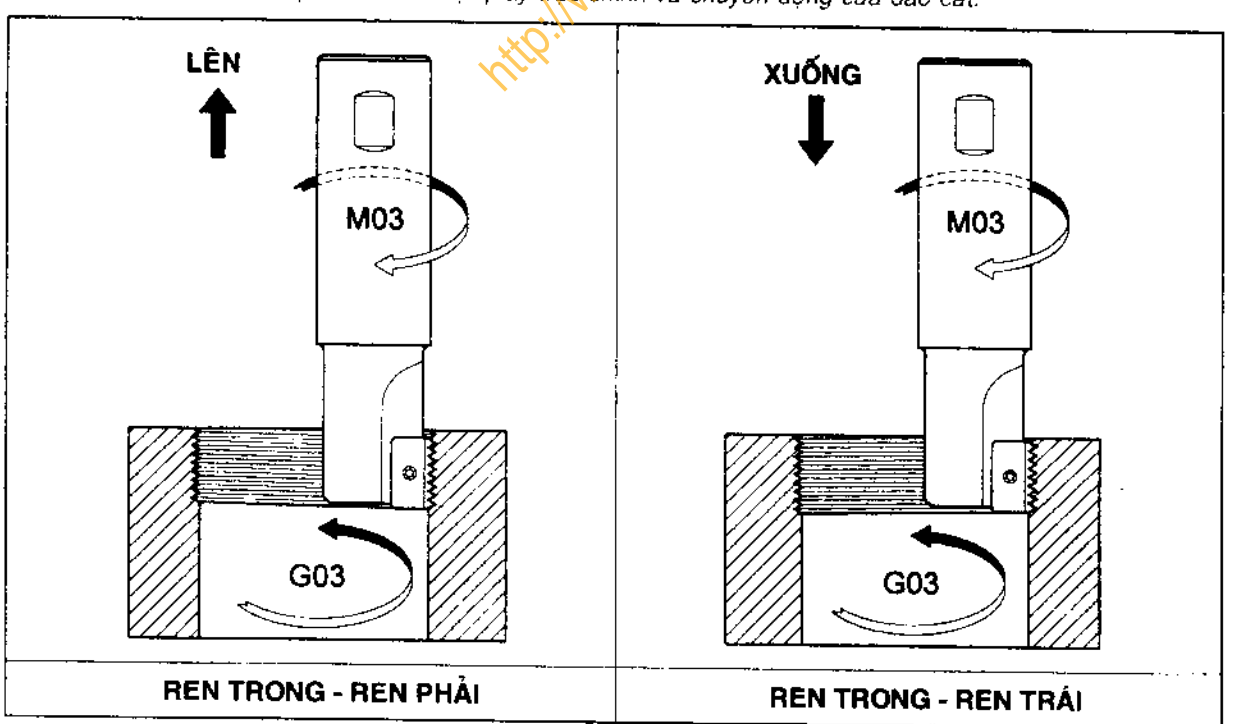
Đối với gia công đứng, chiều cắt gọt theo trục Z có thể là:

- ❑ Lên hoặc dương.
- ❑ Xuống hoặc âm.

Từng chuyển động đều quan trọng, nhưng *sự phối hợp đồng bộ các chuyển động đó* sẽ cho phép cắt gọt các ren đáp ứng yêu cầu kỹ thuật. Các chuyển động này xác định *chiều ren* (trái hoặc phải) áp dụng cho ren ngoài hay ren trong. Các Hình 44.5 và 44.6 minh họa những khả năng áp dụng chuyển động cắt ren, trong chế độ phay ngược.



Hình 44.5. *Phay ren NGOÀI sử dụng chế độ phay ngược - ren phải và trái, sự quay trục chính và chuyển động của dao cắt.*



Hình 44.6. *Phay ren TRONG sử dụng chế độ phay ngược - ren phải và trái, sự quay trục chính và các chuyển động của dao cắt.*

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trong ví dụ này, ren được phay là ren *phải* và là ren *trong*. Chiều quay trục chính là bình thường, sử dụng M03. Hình 44.6 minh họa ren được phay từ dưới lên, sử dụng chuyển động dao ngược chiều kim đồng hồ (G03).

Còn một yếu tố cần xem xét, mảnh chắp phay ren - chủ yếu là chiều cao. Chiều cao của mảnh chắp xác định số vòng quay cần thiết để cắt ren đến chiều sâu toàn phần. Ở đây sẽ sử dụng dao một mảnh chắp, với hai vòng quay là đủ để phay ren theo yêu cầu.

Để bắt đầu phay ren, dao cắt cần được định vị ở chuẩn chi tiết X0Y0 và chiều sâu Z có khoảng trống. Do sử dụng dao cắt nhiều răng và có khoảng trống khả dụng, sẽ bắt đầu từ *dưới* đáy của chi tiết, ví dụ .200, tại Z-0.95 (chiều dày tấm phôi là .750, theo bản vẽ) khoảng hở này cho phép dao dễ dàng đi vào ren. Sự khởi đầu chương trình đã đủ dữ liệu:

```
O4401 (INTERNAL RIGHT HAND THREAD MILLING)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S900 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.95 F50.0
...
```

Tương tự chương trình sử dụng nội suy tròn, bước kế tiếp cần thực hiện là xác định sự tiếp cận tuyến tính đến cung ăn vào (trong chế độ phay ngược). Đây cũng là chuyển động áp dụng bù bán kính dao.

Trong phay xoắn, bù bán kính dao chỉ áp dụng cho hai trục của mặt phẳng đã chọn.

Trong ví dụ này, bù bán kính dao được nhập vào block N6:

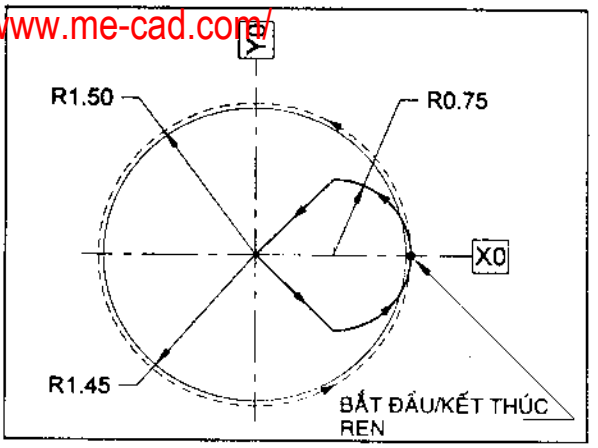
```
N6 G01 G41 X0.75 Y-0.75 D01 F10.0
```

Block kế tiếp là cung ăn vào, với bán kính tiếp cận .750. Chỉ cần các chuyển động theo trục X và Y:

```
N7 G03 Z1.5 Y0 R0.75                    (or I0 J0.75)
```

Do chỉ dùng hai trục, chuyển động này là phẳng (trên mặt phẳng). Hình 44.7 minh họa chuyển động dao. Chú ý, chuyển động này *có vẻ* tương tự chuyển động trong nội suy tròn (ứng dụng hốc tròn). Điều này là không đúng, do trong chuyển động đó còn có *trục Z*.

Tuy nhiên, phương pháp lập trình này sẽ làm cho dao phay ren ăn thẳng vào phôi. Do dao có các răng cắt ren song song, sẽ cắt một loạt các rãnh, thay vì các ren. Điều đó là không thể chấp nhận.



Hình 44.7. Các chuyển động ăn vào và ra trong chuyển động phay ren O4401

Để cắt gọt tốt hơn, hãy khởi đầu với *chuyển động xoắn* cho cung ăn vào. Điều đó có nghĩa là cần bổ sung trục Z. Sự tiếp cận xoắn cần xét đến *bước* ren và giá trị góc (tính theo *độ*) của hành trình trên chu vi đối với cung ăn vào.

Bước ren trong ví dụ này là:

1/2 = 0833333

Với góc hành trình trên chu vi là 90^0 thì X0.75Y-0.25 đến X1.5Y0. Xét dao phay ren có bước tiến 0833333 đối với từng vòng 360^0, sẽ có bước tiến là một phần tư khoảng cách đó đối với từng góc 90^0.

Có thể tính hành trình tuyến tính theo công thức:

$$L_1 = \frac{A \times P}{360}$$

Trong đó:

L_1 = Hành trình tuyến tính trong nội suy xoắn
A = Giá trị góc được nội suy (độ)
P = Bước ren (1/ TPI)

Bước tiến dao trong ví dụ này sẽ là

L_1 = 90 x .0833333/360

L_1 = .0208333 (.0208)

Chuyển động cắt sẽ theo chiều dương trục Z (đi lên), do đó giá trị tuyệt đối của vị trí đích sẽ *ở trên* vị trí khởi đầu, block N7 được chỉnh sửa như sau:

```
N7 G03 X1.5 Y0 Z-0.9292 R0.75 (or I0 J0.75)
```

Tại điểm này, dao ở vị trí khi có thể bắt đầu chuyển động xoắn 360^0 toàn phần. Bạn nên chọn khởi đầu ăn vào và ra khỏi cung tại các vị trí phần tư (0^0, 90^0, 180^0, 270^0), các tính toán sẽ dễ dàng hơn.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Tính toán theo chiều cao ren

Một số tài liệu kỹ thuật có thể dựa trên tính toán góc xoắn của mảnh chắp phay ren, nhưng vẫn có một yếu tố không thay đổi. Dao phay ren phải tiến theo khoảng cách tương đương giá trị bước trong một vòng quay (360^0). Nếu áp dụng phương pháp ăn vào cung, chỉ một phần của bước được lập trình. Giá trị hành trình tuyến tính được tính toán theo tỷ số với góc hành trình. Công thức dưới đây là phiên bản của công thức nêu trên, tính giá trị hành trình tuyến tính dựa trên số ren trong một inch (TPI):

$$L_t = \frac{A}{360 \times TPI}$$

L_t = Hành trình tuyến tính trong nội suy xoắn
A = Giá trị góc được nội suy (độ)
TPI = Số ren/inch

Phay ren

Theo cỡ dao và cỡ ren, đã chọn *hai vòng quay toàn phần* để hoàn tất ren theo yêu cầu. Đối với từng vòng quay, nghĩa là đối với từng 360^0, vị trí tuyến tính của dao phải thay đổi giá trị bước ren. Trong ví dụ này là giá trị .0833333. Chuyển động ren là phay xoắn và có thể sử dụng phương pháp lập trình số gia hoặc chế độ tuyệt đối.

Chế độ tuyệt đối được chọn trước, sau đó là chế độ số gia:

```
N8 G90 G03 X1.5 Y0 Z-0.8459 I-1.5      (TURN 1)
N9 G03 X1.5 Y0 Z-0.7626 I-1.5          (TURN 2)
```

Dữ liệu lặp lại sẽ không xuất hiện trong chương trình cuối. Để so sánh, bạn hãy thử lập trình hai chuyển động đó trong chế độ số gia:

```
N8 G91 G03 X0 Y0 Z0.0833 I-1.5         (TURN 1)
N9 G03 X0 Y0 Z0.0833 I-1.5             (TURN 2)
```

Khi hoàn tất hai chuyển động xoắn, dao đi được .1666 inch dọc theo chiều dương của trục Z và tổng cộng 720^0 (hai vòng). Phần cuối cùng của chương trình sẽ là phần kết thúc đường cắt.

Các chuyển động lùi ra

Với cùng lý do khi dao tiếp cận ren sử dụng nội suy xoắn qua cung 90^0, việc lùi ra khỏi ren cũng sẽ được xử lý tương tự. Sự lùi ra khỏi ren hoàn tất (chuyển động lùi ra) sẽ dịch chuyển dao phay ren *ra xa* ren hoàn tất, sử dụng quay một phần tư vẫn trong chế độ xoắn. Sự tính toán tương tự như phần trước. Kết quả sẽ là:

```
Lt  = 90 x .0833333 / 360
    = .0208333
```

Giá trị số gia này sẽ đưa dao lên và xa ren (lập trình trong chế độ số gia):

```
N10 G03 X0.75 Y0.75 Z-0.7418 R0.75
                                   hoặc (I-0.75 JO)
```

Tại điểm này, dao ở vị trí cách xa ren, do đó có thể thực hiện chuyển động tuyến tính và xóa bù bán kính dao, sau đó trở lại tâm lỗ, lùi dao lên phía trên chi tiết, dịch chuyển đến zero máy và kết thúc chương trình:

```
N11 G40 G01 X0 Y0
N12 G00 Z1.0 M09
N13 G28 X0 Y0 Z1.0 M05
N14 M30
%
```

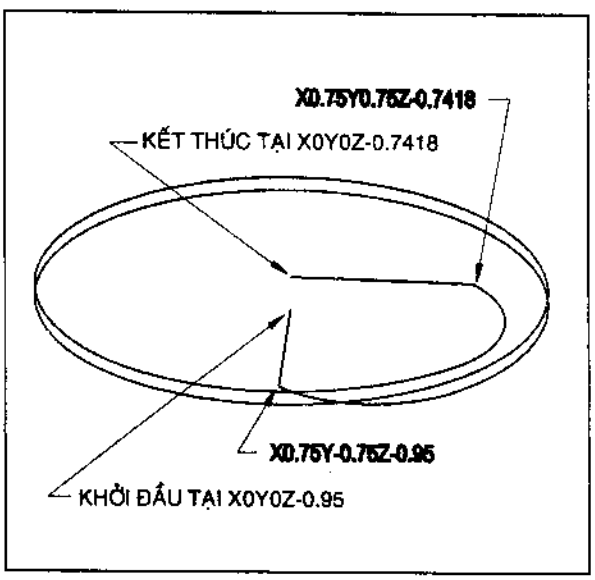
Công việc cắt ren được thực thi và có thể viết chương trình hoàn chỉnh

Chương trình hoàn chỉnh

Chương trình hoàn chỉnh dưới đây, kết hợp các tính toán riêng rẽ, gồm mọi chuyển động của dao phay ren:

```
O4401 (INTERNAL RIGHT HAND THREAD MILLING)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S900 M03
N4 G43 Z0.1 H01 M08
N5 G01 Z-0.95 F50.0
N6 G41 X0.75 Y-0.75 D01 F10.0
N7 G03 X1.5 Y0 Z-0.9292 R0.75
N8 Z-0.8459 I-1.5                      (TURN 1)
N9 Z-0.7626 I-1.5                      (TURN 2)
N10 X0.75 Y0.75 Z-0.7418 R0.75
N11 G40 G01 X0 Y0
N12 G00 Z1.0 M09
N13 G28 X0 YD Z1.0 M05
N14 M30
%
```

Chương trình này chỉ là ví dụ nhỏ về một phương pháp phay ren. Các tính toán là logic và mà chương trình là rõ ràng. Hình 44.8 minh họa chương trình phay ren O4401.



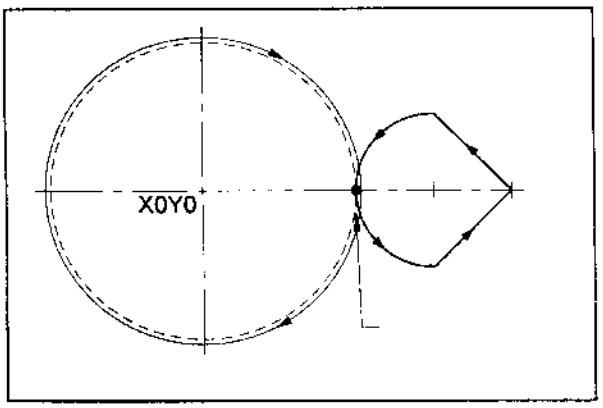
Hình 44.8. Các chuyển động dao trong ví dụ phay ren.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Phay ren ngoài

Phay ren ngoài thường dùng cho các ren lớn với mảnh chắp carbide. Các chuyển động ăn vào và ra là rất quan trọng trong tình huống đó. Các tính toán chuyển động đối với ren tuân theo các nguyên tắc như khi phay ren trong. Các chuyển động tuyến tính ăn vào và ra có thể sử dụng, như đã trình bày trong Chương 28 (*Nội suy tròn*). Nếu không, bạn hãy áp dụng các chuyển động được nêu trên Hình 44.9.



Hình 44.9. *Các chuyển động ăn vào và ra đối với phay ren ngoài.*

Phay ren côn

Hoàn toàn có thể, nhưng tương đối khó, lập trình gia công ren côn (chẳng hạn (NPT hoặc NPTF) sử dụng dao phay ren. Đối với các ren bước nhỏ, vật liệu mềm và góc côn rất hẹp, có thể dùng dao côn và lập trình như dao thẳng, trong một vòng quay. Đối với các ren lớn, phương pháp duy nhất là *mô phỏng* sự phay xoắn (trong trường hợp này cần sử dụng phần mềm chuyên dùng), sử dụng các số gia rất nhỏ chỉ trong chế độ *nội suy tuyến tính*. Cán dao và mảnh chắp phải được chọn theo kích cỡ ren danh định.

Các ren *côn* yêu cầu cán dao gia công ren trái khác với ren phải. Đây là ứng dụng đặc biệt của nội suy xoắn nhưng không thuộc lĩnh vực lập trình bằng tay.

Các khảo sát cơ bản

Có hai vấn đề cần khảo sát kỹ để có thể phay ren đến chiều sâu hợp lý. Một là áp dụng *bù bán kính dao* và thứ hai là chọn *tốc độ cắt*.

Bù bán kính dao chỉ hoạt động đối với hai trục được chọn trên mặt phẳng hoạt động (ví dụ, trong G17, đó sẽ là các trục X và Y). Bạn nên chọn phương pháp phay ngược, do đây là phương pháp được áp dụng cho hầu hết các ứng dụng phay ren.

Sự lựa chọn tốc độ cắt tương tự tốc độ cắt với cung ngoài và cung trong khi áp dụng *Nội suy tròn* (Chương 28). Do yêu cầu là ren chính xác, cần chọn tốc độ cắt chậm hơn khoảng 10-30%. Tốc độ cắt thử nghiệm có thể là .001/răng, sau đó tăng dần theo các điều kiện cụ thể.

PHƯƠNG PHÁP MÔ PHỎNG PHAY REN

Có một phương pháp phay ren *không cần* tùy chọn nội suy xoắn khả dụng trên bộ điều khiển. Đây có thể là trường hợp đối với nhiều máy CNC, hoặc trong các trường hợp xưởng cơ khí cần phay ren chỉ một lần trong khoảng thời gian nào đó và chưa có nội suy tuyến tính trên bộ điều khiển.

Để phay ren (ngoài hoặc trong) trong các điều kiện đó, cần sử dụng *sự mô phỏng* phay xoắn. Mô phỏng các chuyển động xoắn yêu cầu chuyển động cắt *tuyến tính* đồng thời cả ba trục, trong phạm vi dung sai ren cho phép. Điều đó có nghĩa là từng chuyển động sẽ là chuyển động *tuyến tính* ba trục rất nhỏ (sử dụng các trục X, Y, Z). Ren càng chính xác, chương trình càng dài. Phương pháp này hầu như không thể thực hiện bằng tay, do đòi hỏi thời gian và công sức. Điều cần thiết là phần mềm chương trình thực hiện nhiều tính toán trong thời gian ngắn. Nhiều nhà chế tạo dao phay ren cung cấp phần mềm loại này miễn phí hoặc với giá thấp.

Để minh họa chủ đề này, sẽ sử dụng ren trong chương trình O4401. Chương trình mô phỏng có thể rất dài, ít nhất là vài trăm block. Dưới đây là chương trình ví dụ, chỉ nêu vài block ở phần đầu, vài block để dao hoàn tất cung ăn vào, chỉ liên quan với đường thẳng và phần của cung ăn vào. Về thực tiễn, chương trình này là không chuẩn, do không bù bán kính dao. Bù bán kính dao cần được thực hiện trong phần mềm, không phải với G41 hoặc G42 trong sử dụng. Chương trình hoàn tất được thực hiện bằng cách dùng phần mềm CAD/CAM, và dài đến 463 block, so với chỉ 14 block của chương trình hoàn tất sử dụng nội suy xoắn.

```
G20
G17 G40 G80
G90 G54 G00 X0 Y0 S900 M03
G43 Z0.1 H01 M08
G01 Z-0.95 F10.0
X0.75 Y-0.75
X0.7846 Y-0.7492 Z-0.9494
X0.8191 Y-0.7468 Z-0.9488
X0.8536 Y-0.7428 Z-0.9482
X0.8878 Y-0.7373 Z-0.9476
X0.9216 Y-0.7301 Z-0.9470
X0.9552 Y-0.7214 Z-0.9464
X0.9883 Y-0.7112 Z-0.9457
...
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
...
...
X1.4967 Y-0.0697 Z-0.9304
X1.4992 Y-0.0350 Z-0.9298
X1.5000 Y0.0000 Z-0.9292
...
```

Chương trình sẽ xuất một chuỗi các đoạn ngắn nhỏ, theo số gia và thứ tự rất chính xác. Bạn hãy theo dõi ít nhất vài block và hình dung chuyển động thực của dao. Theo cách đó, sẽ cần ba giây để tạo ra 463 block mã trong CAD/CAM. Biết ngôn ngữ mức cao (chẳng hạn Visual Basic®, Visual C++®, và các ngôn ngữ tương tự), có thể viết phần mềm kiểu này rất hiệu quả. Nói chung, khi thực thi phần mềm tiện ích đó, người dùng sẽ nhập số vòng quay, bán kính, bước tiến ren và độ phân giải. Có thể rút ngắn chương trình, nhưng chất lượng ren sẽ giảm.

Bất kể phương pháp được dùng để tạo ra quỹ đạo khi phay ren, đây là lĩnh vực gia công và lập trình đáng được chú ý nhiều hơn trong các xưởng cơ khí.

KHOÉT XOẮN

Mặc dầu phay ren có lẽ là ứng dụng phổ biến nhất của nội suy xoắn nhưng không phải là duy nhất. Một ứng dụng rất hữu ích của tính năng điều khiển này là *khoét lỗ theo chuyển động xoắn*.

Khoét lỗ theo chuyển động xoắn được dùng chủ yếu để thay thế sự ăn dao thẳng xuống phôi nguyên khối. Chắc bạn còn nhớ, sự gia công thô trong vùng khép kín (ví dụ gia công hốc lõm), đòi hỏi dao cắt đạt đến chiều sâu Z xác định, *trước khi* thực sự cắt gọt. Chuyển động trục Z này có thể trong không gian mở, nếu vật liệu, ví dụ, đã được *khoan sẵn*. Chuyển động trục Z còn có thể là cắt gọt vào vật liệu *nguyên khối*, nếu dao cắt là kiểu *cắt tâm* (còn gọi là mũi khoan khoét). Nhưng còn có khả năng khác - *khoét xoắn* - cho phép sử dụng dao cắt phẳng, và đạt đến chiều sâu Z yêu cầu theo một chuỗi các chuyển động cắt xoắn rất nhỏ. Dao cắt có thể là *phẳng* và *không cắt tâm*, do toàn bộ tác dụng cắt được thực hiện bằng các cạnh dao, không phải đáy dao. Sau khi đạt được chiều sâu cắt yêu cầu, thường sử dụng nội suy tròn để làm sạch sau đường cắt xoắn cuối cùng. Phần mềm CAD/Cam mức cao có thể thực hiện điều này rất hiệu quả.

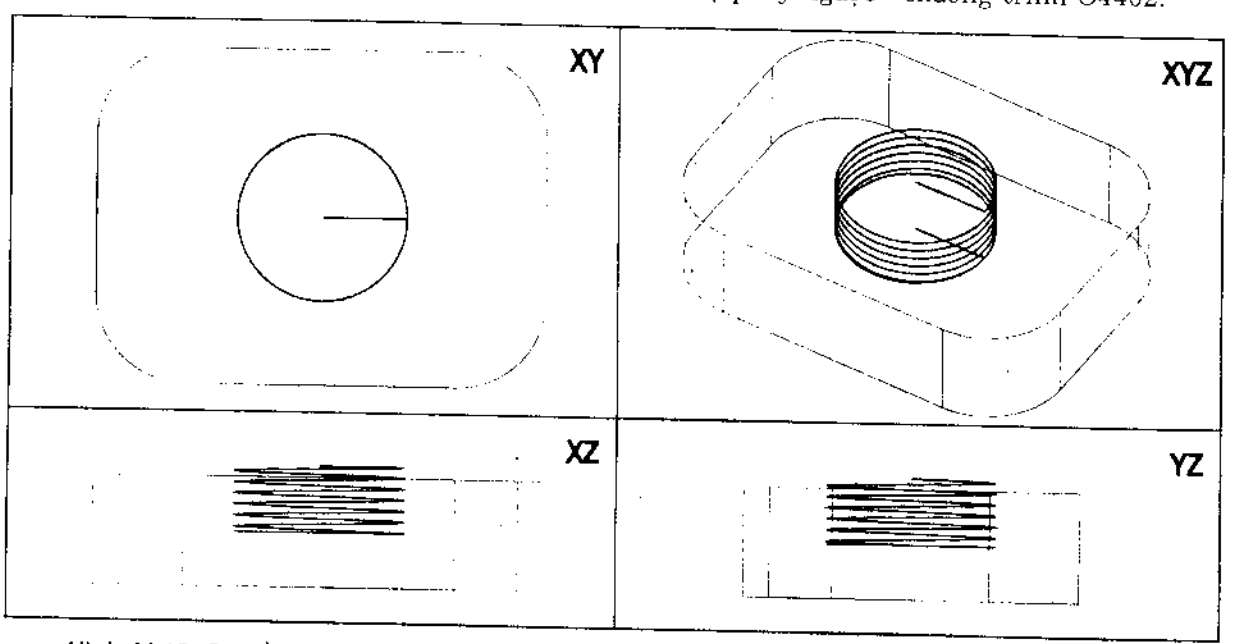
➲ **Ví dụ:**

Để minh họa kỹ thuật lập trình đối với kiểu phay này, sẽ sử dụng dao phay tiêu chuẩn, đáy phẳng ∅.500 (không cần kiểu cắt tâm), và mở lỗ ren đầu đến ∅.750. Chiều sâu hốc là .250 và trong từng chuyển động xoắn dao sẽ dịch chuyển .050. Tâm lỗ là X0Y0 và vị trí Z khởi đầu là .050 phía trên chi tiết (cũng là zero chương trình trên trục Z). Tổng số chuyển động xoắn (số vòng quay) là sáu (một phía trên mặt, cộng với năm vòng dưới mặt chi tiết).

Giá trị số gia bất kỳ có thể được chọn cho chiều sâu này, tùy theo điều khiển cắt gọt. Số gia càng nhỏ, số đường cắt xoắn càng nhiều, đòi hỏi thời gian cắt gọt càng dài.

Chương trình có thể theo chế độ số gia hoặc tuyệt đối, trong trường hợp này, chế độ số gia tương đối dễ lập trình hơn. Sự cắt gọt sẽ theo chế độ phay ngược - chương trình O4402.

http://www.me-cad.com/



Hình 44.10. Sơ đồ minh họa chuyển động xoắn được dùng trong khoét xoắn. Chương trình O4402.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
O4402 (HELICAL RAMPING)
N1 G20
N2 G17 G40 G80
N3 G90 G54 G00 X0 Y0 S700 M03
N4 G43 Z1.0 H01 M08
N5 G01 Z0.05 F50.0      (APPROACH TO Z-START)
N6 G41 X0.375 D01 F15.0 (START COMPENSATION)
N7 G91 G03 I-0.375 Z-0.05    (CUT ABOVE WORK)
N8 I-0.375 Z-0.05       (CUT 1 BELOW TOP FACE)
N9 I-0.375 Z-0.05       (CUT 2 BELOW TOP FACE)
N10 I-0.375 Z-0.05      (CUT 3 BELOW TOP FACE)
N11 1-0.375 Z-0.05      (CUT 4 BELOW TOP FACE)
N12 I-0.375 Z-0.05      (CUT 5 BELOW TOP FACE)
N13 1-0.375         (CIRCULAR BOTTOM CLEANUP)
N14 G90 G01 G40 X0        (RETURN TO XY START)
N15 G00 Z1.0 M09
N16 G28 Z1.0 M05
N17 M30
%
```

Ở đây bạn cần chú ý hai điều. Thứ nhất, do sử dụng chế độ số gia, sự khởi đầu trên trục Z, là *cực kỳ* quan trọng (block N4). Thứ hai bán kính dao được áp dụng trong chuyển động thẳng từ tâm đến phần khởi đầu chuyển động xoắn thứ nhất (Hình 44.10).

Nội suy xoắn có thể là công cụ lập trình rất mạnh, thường là không thể thay thế được bằng các phương pháp khác. Tuy chỉ là tùy chọn điều khiển, nhưng lợi ích chính là rút gọn chương trình, khả năng thay đổi nhanh chóng và dễ dàng, đáng giá với chi phí tăng thêm.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

http://www.me-cad.com/

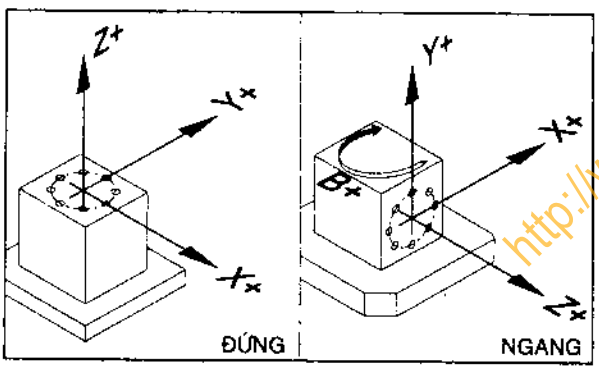
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

45 GIA CÔNG CẮT GỌT NGANG

Trong các chương của cuốn sách này đã trình bày nhiều ví dụ lập trình, chúng đều có đặc điểm chung là sử dụng trên trung tâm gia công đứng. Lý do? thứ nhất, nói chung hiện có nhiều trung tâm gia công đứng trong các xưởng cơ khí, và sự trình bày cả hai kiểu máy sẽ làm cho nội dung trở nên phức tạp hơn. Thứ hai, hầu như mọi chủ đề đã đề cập với các loại máy đứng đều có thể áp dụng cho máy ngang. Vậy, các khác biệt là gì?

Trung tâm gia công ngang khác với trung tâm đứng chủ yếu là về tính năng chung. Trung tâm gia công đứng thường dùng cho kiểu chi tiết một mặt gia công, còn trung tâm ngang được dùng để gia công chi tiết trên nhiều mặt trong một lần gá lắp. Riêng điều này cũng đủ làm cho trung tâm gia công ngang trở nên đa năng hơn, và máy đắt tiền hơn. Hình 45.1 minh họa so sánh các định hướng trục giữa hai kiểu máy.



Hình 45.1. Sự khác biệt về định hướng trục giữa máy đứng và ngang.

Từ minh họa này có thể thấy toàn bộ mặt phẳng XY được dùng làm mặt phẳng chính và trục Z được dùng để kiểm soát chiều sâu cắt. Về mặt này, không có sự khác biệt giữa hai kiểu máy, giữa lập trình và giá lắp, có ba khác biệt chính trên trung tâm gia công ngang:

- Sự có mặt của trục thứ tư, thường được gọi là trục B.
- Sự có mặt của bộ thay pallet.
- Nhiều khả năng gá lắp và xác lập bù.

Trước hết, bạn hãy xem xét trục thứ tư của trung tâm gia công ngang CNC.

CÁC TRỤC QUAY VÀ PHÂN ĐỘ

Mọi khái niệm lập trình đã đề cập cho đến phần này đều có thể áp dụng cho các máy CNC ngang. Các trục XY được sử dụng chủ yếu cho nguyên công khoan và gia công biên dạng, trục Z điều khiển chiều sâu cắt.

Trung tâm gia công ngang khác với trung tâm gia công đứng không chỉ về định hướng trục và kiểu chi tiết có thể được gia công. Một trong các khác biệt cơ bản là trục bổ sung.

Đây là trục quay hoặc phân độ, thường ký hiệu là *trục B*. Hai thuật ngữ quay và phân độ có ý nghĩa khác nhau rõ rệt.

- Bàn *phân độ* sẽ quay chi tiết lắp trên bàn đó, nhưng không thể sử dụng đồng thời với kiểu chuyển động cắt bất kỳ. Kiểu này hỗ trợ chuyển động định vị.
- Bàn máy *quay* cũng quay chi tiết lắp trên bàn, nhưng có thể có đồng thời tác động cắt. Kiểu này hỗ trợ chuyển động biên dạng.

Trục thứ tư phổ biến nhất trên trung tâm gia công ngang là kiểu phân độ, được gọi là trục B.

BÀN PHÂN ĐỘ (TRỤC B)

Trục phân độ, được dùng để phân độ bàn máy, nếu máy được trang bị tính năng này. Trung tâm gia công ngang và máy phay doa có bàn phân độ là tính năng tiêu chuẩn. Bàn máy quay toàn phần là tùy chọn trên cả hai kiểu trung tâm gia công.

Đơn vị số gia

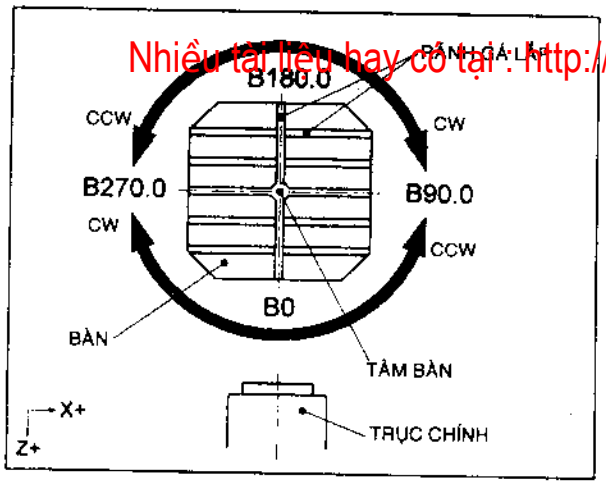
Trục phân độ được lập trình theo giá trị độ (góc) dựa trên yêu cầu gia công. Ví dụ, để phân độ bàn máy đến vị trí 45^0 bạn hãy lập trình:

```
G90 G00 B45.0
```

Số gia tối thiểu phụ thuộc vào thiết kế máy. Đối với sự phân độ, đơn vị số gia tối thiểu có thể là 1 độ hoặc thậm chí 5 độ. Tuy nhiên, để tăng tính linh hoạt, và để gia công quay, cần có đơn vị số gia nhỏ hơn. Nhiều nhà chế tạo cung cấp 0.1, 0.01, và 0.001 độ là số gia phân độ nhỏ nhất. Trong mọi trường hợp, sự lập trình các chuyển động phân độ có thể được thực hiện theo hai chiều.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 45.2. Chiều trục B (phân độ)

Chiều phân độ

Trục B có thể được lập trình để phân độ theo chiều thuận (CW) hoặc ngược (CCW) chiều đồng hồ, nhìn từ trên xuống ở bàn máy, theo mặt phẳng XZ (Hình 45.2).

Kích cỡ bàn máy, kể cả các góc, là rất quan trọng để xác định khoảng hở trước khi phân độ.

Các hàm kẹp chặt và mở kẹp bàn máy

Để duy trì sự gá lắp chắc chắn, bàn phân độ phải được kẹp chặt vào thân chính của máy trong khi cắt gọt. Đối với các chuyển động phân độ, bàn máy phải được mở kẹp chặt. Điều này là đúng đối với hầu hết các trung tâm gia công. Đối với mục đích đó, nhà chế tạo cung cấp hai hàm phụ đặc biệt, sẽ được sử dụng trong các ví dụ kế tiếp:

- ❑ Kẹp chặt bàn máy ... ví dụ M78
- ❑ Mở kẹp bàn máy ... ví dụ M79

Các chỉ số hàm này có thể khác nhau tùy theo loại máy.

Nói chung, hàm mở kẹp bàn máy được lập trình trước khi phân độ, tiếp theo là chuyển động trục B và block khác sẽ chứa hàm kẹp chặt.

```
M79                 Mở kẹp bàn máy
G00 B90.0           Phân độ bàn máy
M78                 Kẹp chặt bàn máy
```

Một số thiết kế yêu cầu mã M khác, chẳng hạn để điều khiển chốt định vị hoặc xác nhận sự sẵn sàng của bàn máy.

Trục B được lập trình về logic là hoàn toàn như các trục tuyến tính, kể cả chế độ lập kích thước. Chế độ số gia hoặc tuyệt đối đều có thể được sử dụng để phân độ, áp dụng các lệnh G90 hoặc G91 tương ứng.

Phân độ trong chế độ tuyệt đối và số gia

Tương tự các trục khác, trục B có thể được lập trình trong chế độ số gia hoặc tuyệt đối với cùng hành vi như các trục tuyến tính.

Ví dụ dưới đây là trong độ tuyệt đối, ghi theo bảng hai cột. Cột thứ nhất là chuyển động phân độ lập trình trong chế độ G90, cột thứ hai là chuyển động phân độ thực (khoảng cách dịch chuyển) và chiều chuyển động. Mọi chuyển động quay đều dựa trên sự chiếu vuông góc với mặt phẳng XZ.

➲ **Chế độ tuyệt đối - các phân độ liên tiếp:**

Chuyển động lập trình trong G90	Chuyển động phân độ thực
G90 G28 B0	Vị trí zero máy trên trục B
G00 B90.0	90 độ CW
B180.0	90 độ CW
B90.0	-90 độ CCW
B270.0	180 độ CW
B247.356	-22.644 độ CCW
B0	-247.356 độ CCW
B-37.0	-37 độ CCW
B42.0	79 độ CW
B42.0	không chuyển động (0 độ)
B-63.871	-105.871 độ CCW

Bảng kế tiếp cũng tương tự. Cột thứ nhất là chuyển động phân độ lập trình trong chế độ G91, cột thứ hai là chiều chuyển động và vị trí thực. Chiều quay đều dựa trên sự chiếu vuông góc với mặt phẳng XZ.

➲ **Chế độ số gia - các phân độ liên tiếp:**

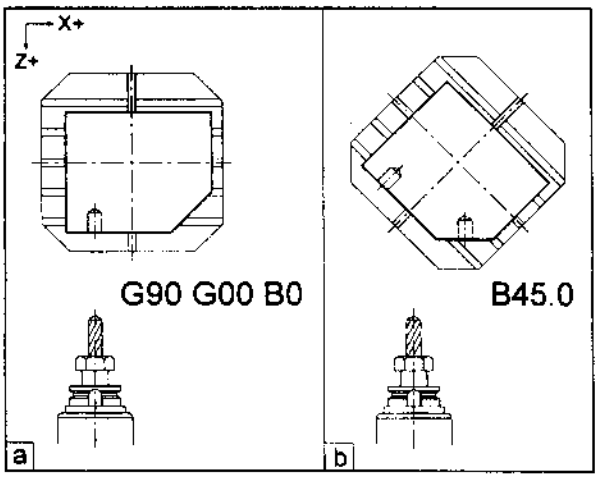
Chuyển động lập trình trong G91	Vị trí tuyệt đối thực
G90 G28 B0	Vị trí zero máy trên trục B
G91 G28 B0	Zero máy - không chuyển động
G00 B90.0	90.000 CW
B180.0	270.000 CW
B90.0	360.000 CW
B270.0	630.000 CW
B0	Không chuyển động
B125.31	755.310 CW
B-180.0	575.310 CW
B-75.31	500.000 CCW
B-75.31	424.690 CCW
B-424.69	0.000 CCW

Bạn hãy nghiên cứu hai bảng này theo từng block, với thứ tự đã nêu, để hiểu vấn đề. Chú ý, B-37.0 trong bảng thứ nhất, sẽ đạt được cùng kết quả nếu B323.0 là giá trị dương.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Trong bảng thứ hai, block thứ nhất trong giá trị tuyệt đối để bảo đảm khởi đầu tại B0. Một sự kiện thú vị - khi sự quay cùng chiều đạt đến 360^0 (đúng một vòng), số độ góc vẫn tiếp tục tăng, mà không trở về zero. Đây là điều cần lưu ý. Nếu phân độ (trong chế độ số gia) là hai vòng, vị trí tuyệt đối của bàn máy sẽ là 720.000^0. Phân độ hai lần theo chiều ngược là cần thiết để đạt đến zero tuyệt đối. Ví dụ nhỏ được minh họa trên Hình 45.3.



Hình 45.3. Chiều trục B từ B0 đến B45.0 trong chế độ tuyệt đối - O4501

Để lập trình hai vị trí nêu trên, chuỗi thứ tự block sẽ như trong ví dụ O4501

```
O4501
G90 G54 G00 X.. Y.. Z..
M79
B0
M78
...
< DRILL HOLE AT B0 >
...
G90 G55 G00 X.. Y.. Z..
M79
B45.0
M78
...
< DRILL HOLE AT B45.0 >
...
```

Đối với ví dụ này, các kích thước khoan lỗ là không quan trọng.

Luôn luôn quan sát các khoảng hở an toàn khi phân độ trục B.

TRỤC B VÀ CHẾ ĐỘ BÙ

Một trong các khác biệt quan trọng nhất giữa các trung tâm gia công đứng và ngang là cách thức lập trình và chọn hai chế độ bù chính:

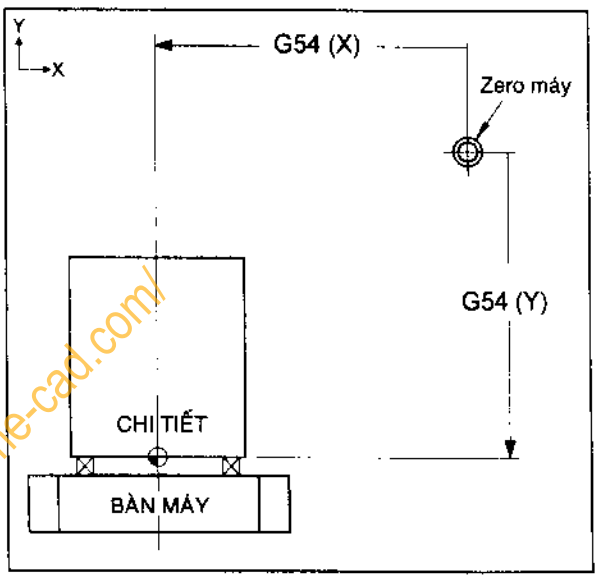
- Bù chi tiết
- Bù chiều dài dao

Bù bán kính dao cắt *không bị* tác động từ trục B và được lập trình như trong gia công đứng.

Quan hệ giữa các chế độ bù và bề mặt gia công của chi tiết là quan trọng và phức tạp hơn so với gia công đứng.

Bù chi tiết và trục B

Bù chi tiết được đo từ zero máy đến zero chương trình. Điều khác biệt là nhiều mặt được dùng để gia công thay vì chỉ một mặt. Điều đó có nghĩa là quỹ đạo dao cho từng mặt đều có zero chương trình riêng, do đó có bù chi tiết riêng. Hình 45.4 minh họa sự xác lập, chiếu chi tiết theo chiều trục chính.



Hình 45.4. Bù chi tiết đối với ứng dụng ngang - hình chiếu trước.

Mặc dù minh họa này biểu thị zero chi tiết ở tâm của bàn phân độ, zero chi tiết có thể ở trên từng mặt chi tiết hoặc thậm chí ở vị trí khác. Mỗi vị trí đều có các ưu điểm riêng, và không có vị trí "tốt nhất", nói chung tùy theo yêu cầu gia công, thiết kế đồ gá, và sở thích của nhà lập trình.

Khi thay đổi từ bề mặt này sang bề mặt khác, bạn nhớ thay đổi chế độ bù chi tiết. Ví dụ, nếu có bốn mặt cần gia công, mỗi mặt đều có bù chi tiết riêng, chẳng hạn G54, G55, G56 và G57. Trục B thường không phụ thuộc vào bù chi tiết, do đó block tốt nhất để lập trình chế độ bù mới là trong chuyển động nhanh thứ nhất. Ví dụ ngắn ở phần trước minh họa phương pháp này. Phần kế tiếp sẽ trình bày xác lập bù chi tiết đối với trục Z và bù chiều dài dao.

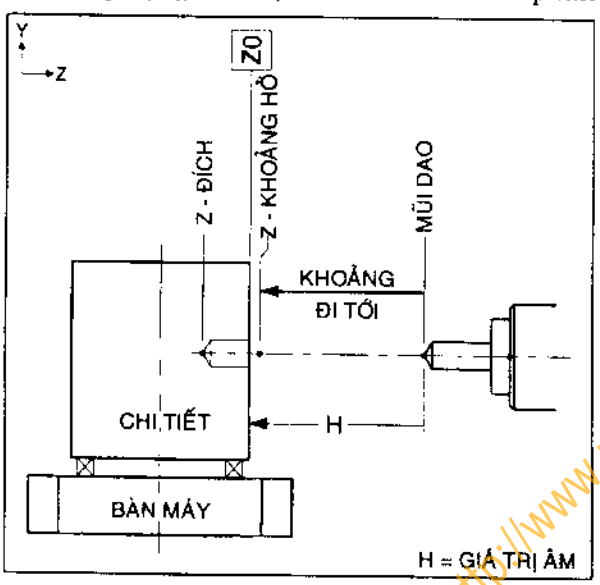
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

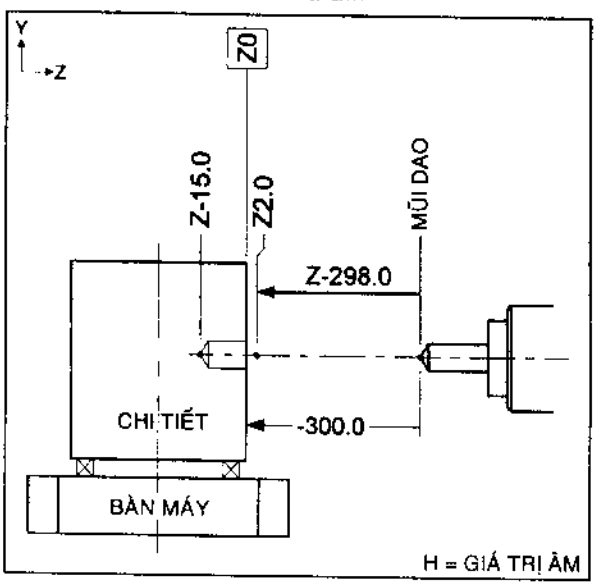
Khái niệm sử dụng nhiều chế độ bù chi tiết cho các mặt gia công là tương đối dễ hiểu. Xác lập chiều dài dao có thể phức tạp hơn, tùy thuộc vào nhiều yếu tố. Thứ nhất là phương pháp xác lập chiều dài dao. Có ít nhất hai phương pháp xác lập bù chiều dài dao đã được trình bày trong Chương 18, phần này sẽ nêu một ý nghĩa mới của chúng.

Phương pháp tiếp xúc

Một phương pháp là *tiếp xúc* Z0 của bề mặt gia công và đăng ký khoảng cách từ đỉnh dao là bù chiều dài âm. Đây là phương pháp thông dụng trên các máy đứng. Phương pháp tiếp xúc có thể áp dụng cho một số nhỏ các dao và phân



Hình 45.5. Phương pháp bù chiều dài dao tiếp xúc - với H là âm



Hình 45.6. Phương pháp bù chiều dài dao tiếp xúc - ví dụ H là giá trị âm

đó. Tuy có thể chọn tâm bàn phân độ là Z0, nhưng đây là giải pháp không thực tiễn. Hình 45.5 minh họa nguyên tắc của phương pháp tiếp xúc, và Hình 45.6 là ví dụ thực tế. Chú ý, sự gá lắp là hoàn toàn tương tự như gia công đứng. Block chương trình.

```
G43 Z20 H01
```

sẽ dịch chuyển dao Z-298.0, nếu xác lập H01 là -300.0.

Phương pháp xác lập trước

Xác lập chiều dài dao trên trung tâm gia công đứng thường là phương pháp tiếp xúc nhưng cũng có thể dùng phương pháp xác lập trước. Phương pháp *xác lập trước* sử dụng thiết bị xác lập trước chiều dài dao đặc biệt và được thực hiện bên ngoài máy. Có lý do để phương pháp xác lập trước có tính thực tiễn cao hơn đối với gia công ngang so với gia công đứng.

Nói chung, một dao thường chỉ yêu cầu một chế độ bù chiều dài dao. Xét tình huống điển hình đối với gia công ngang - một dao sẽ gia công sáu mặt, tiếp theo là bốn dao khác cũng gia công trên sáu mặt đó. Từng dao yêu cầu chiều dài dao riêng cho mỗi mặt - tổng cộng là 30 giá trị bù chiều dài khác nhau. Có vài giải pháp cho trường hợp này.

Mọi giải pháp đều sử dụng số đo chiều dài *dao xác lập trước* và *một xác lập bổ sung*. Dao lắp trên cán dao được đặt vào thiết bị xác lập trước. Thông qua bộ đọc quang, bộ xác lập trước được chuẩn hóa theo đường chuẩn của máy. Sau đó, chiều dài dao được đo một cách chính xác. Giá trị dương biểu thị chiều dài dao thực từ mũi dao đến đường chuẩn của máy. Giá trị này sẽ được nhập vào bộ đăng ký bù chiều dài dao tương ứng. Còn một vấn đề là *quan hệ giữa giá trị đo và vị trí chi tiết*. Trong phương pháp tiếp xúc, dao chạm vào chi tiết, và quan hệ này là trực tiếp. Phương pháp xác lập trước không có sự tiếp xúc. Cần có thêm *một xác lập bổ sung*.

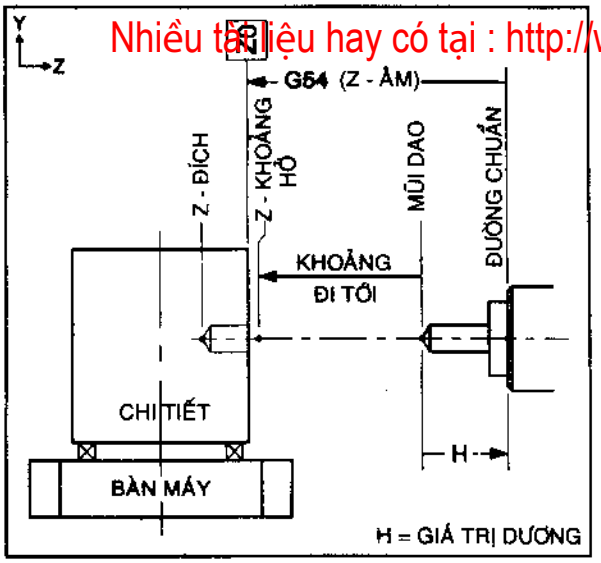
Xác lập này là mục nhập khoảng cách giữa đường chuẩn máy và Z0 của địa chỉ Z bù chi tiết hiện hành - các Hình 45.7 và 45.8.

Minh họa này biểu thị giá trị bù được nhập vào bộ đăng ký Z của bù chi tiết G54 là -500.0. Đây là khoảng cách từ đường chuẩn đến zero chi tiết. Để chứng minh tính khả dụng của phương pháp này có thể dùng H01= 200.0 và vị trí Z ở khoảng trống là G43Z2.0401. *Khoảng cách đi tới* được tính như sau:

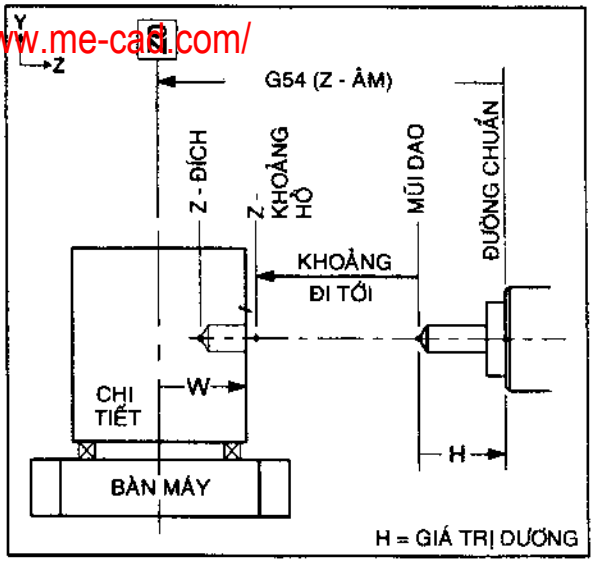
```
G54 (Z) + khoảng trống Z + H01
= -500.0 + 2.0 + 200.0
= -298.0
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

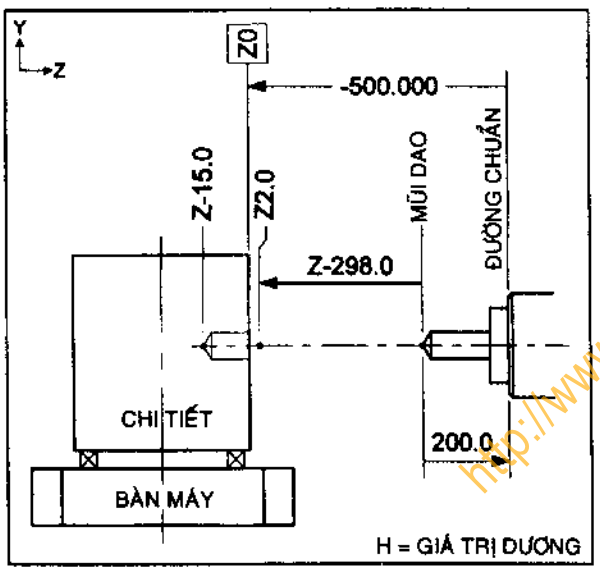
Nhiều tài liệu hay có tại : http://www.me-cad.com/



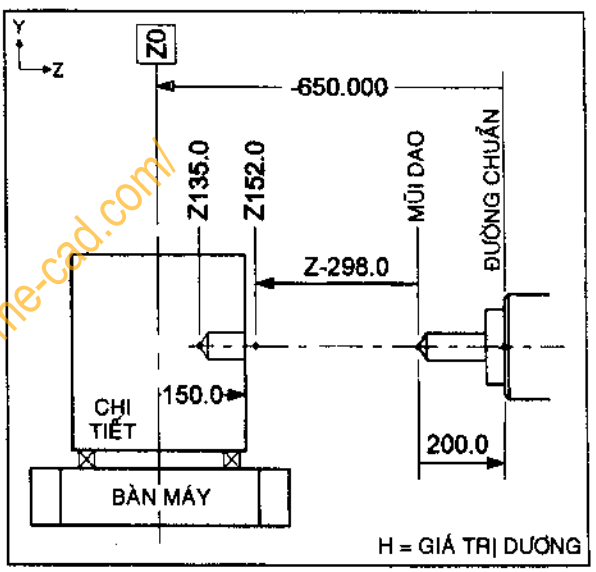
Hình 45.7. *Bù chiều dài dao xác lập trước theo Z0 = bề mặt - với H dương*



Hình 45.9. *Bù chiều dài dao xác lập trước theo Z0 = tâm - với H dương*



Hình 45.8. *Bù chiều dài dao xác lập trước theo Z0 = bề mặt- ví dụ với H có giá trị dương*



Hình 45.10. *Bù chiều dài dao xác lập trước theo Z0 = tâm - ví dụ với H có giá trị dương*

http://www.me-cad.com/

Ví dụ này đo đến vị trí Z0 và ở mặt trước của chi tiết. Còn có tùy chọn thứ hai, nếu Z0 được xác lập tại tâm bàn phân độ. Đây chẳng qua chỉ là sự thay đổi *quy chiếu*, thực tế vẫn như nhau. Hình 45.9 và 45.10 minh họa sự thay đổi biểu kiến đó.

Xác lập này chỉ thay đổi do kích thước bổ sung W, xác định khoảng cách từ zero chương trình đến bề mặt chi tiết. Các giá trị trục Z trong chương trình cũng sẽ thay đổi, do mọi kích thước đều được lấy từ Z0 ở tâm bàn phân độ, thay vì bề mặt chi tiết.

Trong chương trình, block dịch chuyển dao đến vị trí Z trống là G43Z152.0H01. Để tính *khoảng cách đi tới* trong trường hợp này, cần gộp khoảng cách W, phải được biết trong khi gá lắp (trên bản vẽ đồ gá hoặc đo đạc). Do đó giá trị W = 150.0, không thay đổi đối với bù chiều dài H01= 200.0, nhưng là thay đổi quan trọng đối với G54 - do được đo từ tâm bàn phân độ (Z0). Vị trí Z trống gộp cả chiều dài W và khoảng hở vật lý 2 mm, đúng như trong trường hợp trước. Trong ví dụ này sẽ sử dụng giá trị Z-650.0 cho G54.

```
G54 (2) + Z trống + H01 =
= -650.0 + 152.0 + 200.0
= -298.0
```

Dao sẽ tiếp tục một cách bình thường đến chiều sâu Z-135.0. Nói chung, ứng dụng này là

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

hoàn toàn tiêu tự động. Người vận hành phải biết vị trí Z0 cho từng công việc. Thông tin này luôn luôn xuất phát từ nhà lập trình dưới dạng các ghi chú chương trình hoặc bảng kê về gá lắp.

TRỞ VỀ ZERO MÁY

Đối với gia công đứng, trong hầu hết các trường hợp đều lập trình trở về zero sau từng dao cắt. Sự trở về đó chỉ theo trục Z. Lý do rất đơn giản - trên trung tâm gia công đứng, zero máy trên trục Z được đồng bộ hóa với bộ thay dao tự động. Tuy nhiên, điều này *không* được áp dụng trên trung tâm gia công ngang.

Do thiết kế, chuyển động trở về zero máy trước mỗi lần thay dao là theo *trục Y*, còn mọi khía cạnh khác, lập trình các lệnh trở về zero máy là hoàn toàn như nhau.

Dưới đây là so sánh sự kết thúc trước khi thay dao trên hai kiểu máy.

Đứng: G91 G28 Z0

Ngang: G91 G28 Y0 Z0

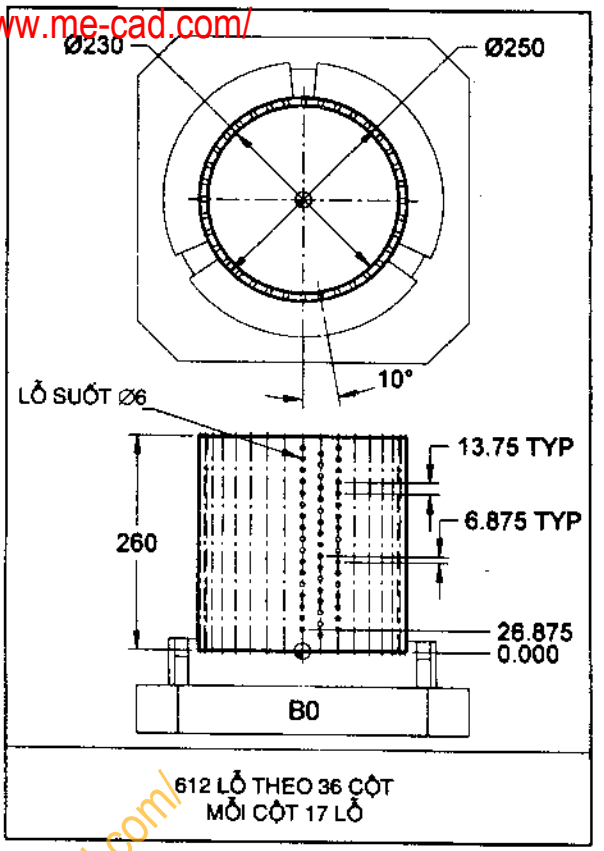
Câu hỏi đặt ra là ý nghĩa của sự trở về theo trục Z trong block khi chỉ yêu cầu trở về trục Y. Câu trả lời là *sự an toàn*. Tuy chỉ cần trục Y là đủ để thay dao tự động nhưng đồng thời dao phải *cách xa* chi tiết. Sự trở về dọc theo trục Z sẽ giúp cho quá trình đó trở nên dễ dàng hơn. Tất nhiên, lập trình khoảng hở vừa đủ trên trục Z cũng đạt được kết quả đó, nhưng điều này hơi khó thực hiện. Với bàn ở vị trí phân độ khác zero, các dao dài và ngắn, các mặt chi tiết khác nhau, đồ gá,... có thể khó biết chính xác khoảng cách rút dao theo trục Z. Đó là lý do bạn cần nhớ nguyên tắc dưới đây:

Trở về theo trục Y là do yêu cầu và theo trục Z là để an toàn.

PHÂN ĐỘ VÀ CHƯƠNG TRÌNH CON

Trình bày mọi kết hợp của các phương pháp gá lắp và ảnh hưởng của chúng đối với định dạng chương trình là điều không thể. Vấn đề gia công ngang, đặc biệt là phần gá lắp, có thể rất phức tạp và đòi hỏi kinh nghiệm. Phần này chỉ có thể cung cấp kiến thức tổng quát cùng với ví dụ lập trình đơn giản.

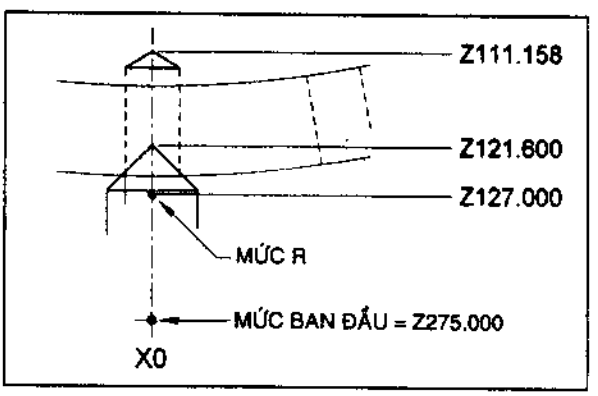
Để minh họa phương pháp áp dụng sự phân độ một cách hiệu quả, ví dụ chương trình trong phần này sẽ là khoan điểm và khoan 612 lỗ trên mặt trụ (Hình 45.11). Mũi khoan điểm còn vạt góc .400 × 45°, đo từ mặt trụ. Mọi tính toán chiều sâu đều là thực.



Hình 45.11. *Ví dụ thực tiễn về phân độ sử dụng các chương trình con - ví dụ O4502*

Bạn đừng lo ngại về số lỗ quá lớn. Sử dụng các chương trình con sẽ làm giảm rõ rệt chiều dài chương trình. Chương trình này không sử dụng chuỗi thứ tự kẹp chặt - mở kẹp chặt, điển hình đối với trục B kiểu quay. Nếu máy yêu cầu mở kẹp chặt bàn máy trước khi phân độ và kẹp chặt sau khi phân độ, bạn hãy dùng các hàm M tương ứng để kẹp chặt và mở kẹp chặt bàn phân độ.

Trước khi nghiên cứu chương trình ví dụ, cần chọn các dao cắt. Chỉ sử dụng hai dao, gồm một mũi khoan điểm 10 mm và một mũi khoan



Hình 45.12. *Dữ liệu dao được dùng trong chương trình O4502.*

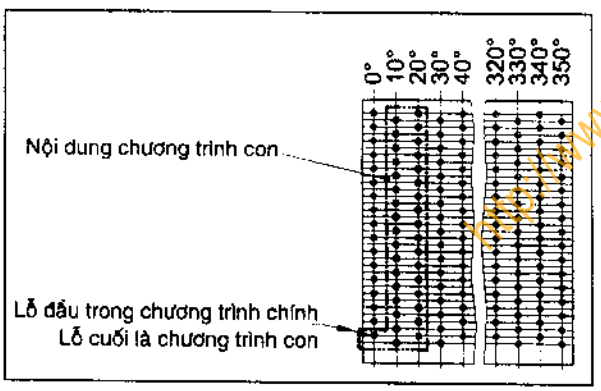
Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

6 mm. Hình 45.12 là minh họa các vị trí cơ bản của hai mũi dao đó.

Mức R là như nhau cho cả hai dao và chiều sâu đối với khoan điểm gồm cả lượng vạt góc nhỏ để loại bỏ bavia cho các lỗ. Chiều sâu khoan bảo đảm sự khoan lỗ suốt toàn phần. Ở đây các tính toán thực là không quan trọng, tuân theo cùng các nguyên tắc đã thiết lập trong các chương trước.

Sự triển khai các chương trình con đòi hỏi đôi chút công sức. Ở đây sẽ sử dụng hai chương trình con. Chúng hầu như giống hệt nhau, ngoại trừ sự lựa chọn chu kỳ cố định. Cũng có thể sử dụng một số phương pháp khác, nhưng Chương này chỉ tập trung vào bàn phân độ. Hai chương trình con sẽ bắt đầu từ dưới mặt trụ, tại vị trí B0(0^0). Lỗ này sẽ được dùng làm vị trí khởi đầu nhưng sẽ được khoan cuối cùng. Tuy chưa khoan lỗ đó, nhưng sự phân độ 10^0 được gộp vào chương trình. Đó là lý do để khởi đầu cách một cột lỗ. Hai cột là phần của từng chương trình con, giữa chúng là phân độ 10^0. Các chú thích trong chương trình con sẽ giải thích quá trình này. Chú ý vùng được đánh dấu trên Hình 45.13, biểu thị nội dung chương trình con.



Hình 45.13. *Khai triển phẳng mặt trụ - hai đầu biểu thị nội dung chương trình con.*

```
O4502 (MAIN PROGRAM)
(START FROM MACHINE ZERO - T01 IN THE SPINDLE)
(X0Y0 = FIXTURE CENTER / Z0 = BOTTOM OF PART)
(T01 - 10 MM DIA SPOT DRILL)
(T02 - 6 MM DIA DRILL THRU)

N1 G21
N2 G17 G40 G80
/N3 G91 G28 Z0
/N4 G28 X0 Y0
/N5 G28 B0
N6 G90 G54 G00 X0 Y26.875 S1000 M03 T02
N7 G43 Z275.0 H01 M08
N8 M98 P4551 L18
N9 G28 Y0 Z0
N10 G28 B0
N11 M01

N12 T02
N13 M06
N14 G90 G54 G00 X0 Y26.875 S1250 M03 T01
N15 G43 Z275.0 H02 M08
N16 M98 P4552 L18
N17 G28 X0 Y0 Z0
N18 G28 B0
N19 M06
N20 M30
%

O4551 (SUBPROGRAM FOR SPOT DRILL)
N101 G91 G80 Y-6.875      (MOVE DOWN BY PITCH)
N102 G90 Z275.0                  (CLEAR Z)
N103 G91 B10.0          (ROTATE BY 10 DEGREES)
N104 G99 G82 R-148.0 Z-5.4 P200 F120.0  (DRL)
N105 Y13.75 L16       (16 MORE HOLES IN Y PLUS)
N106 G80 G00 Y6.875         (MOVE UP BY PITCH)
N107 G90 Z275.0                  (CLEAR Z)
N108 G91 B10.0          (ROTATE BY 10 DEGREES)
N109 G99 G82 R-148.0 Z-5.4 P200       (1 HOLE)
N110 Y-13.75 L16    (16 MORE HOLES IN Y MINUS)
N111 M99            (END OF SUBPROGRAM 04651)
%

O4552 (SUBPROGRAM FOR 6MM DRILL)
N201 G91 G80 Y-6.875      (MOVE DOWN BY PITCH)
N202 G90 Z275.0                  (CLEAR Z)
N203 G91 B10.0          (ROTATE BY 10 DEGREES)
N204 G99 G83 R-148.0 Z-15.84 Q7.0 F200.0 (DRL)
N205 Y13.75 L16       (16 MORE HOLES IN Y PLUS)
N206 G80 G00 Y6.875         (MOVE UP BY PITCH)
N207 G90 Z275.0                  (CLEAR Z)
N208 G91 B10.0          (ROTATE BY 10 DEGREES)
N209 G99 G83 R-148.0 Z-15.84 Q7.0     (1 HOLE)
N210 Y-13.75 L16    (16 MORE HOLES IN Y MINUS)
N211 M99            (END OF SUBPROGRAM 04652)
%
```

Mức ban đầu Z275.0, được dùng trong cả ba chương trình, là hợp lý để phân độ an toàn. Lựa chọn khoảng hở trục Z thích hợp là rất quan trọng, ngoài ra, cần biết rõ kích cỡ bàn phân độ và kích thước các góc của bàn đó. Bàn phân độ cho chi tiết này có hình vuông 400 × 400 mm với các góc 50 × 50 mm. Gá lắp chi tiết đồng tâm với chiều quay phân độ và không có các chướng ngại.

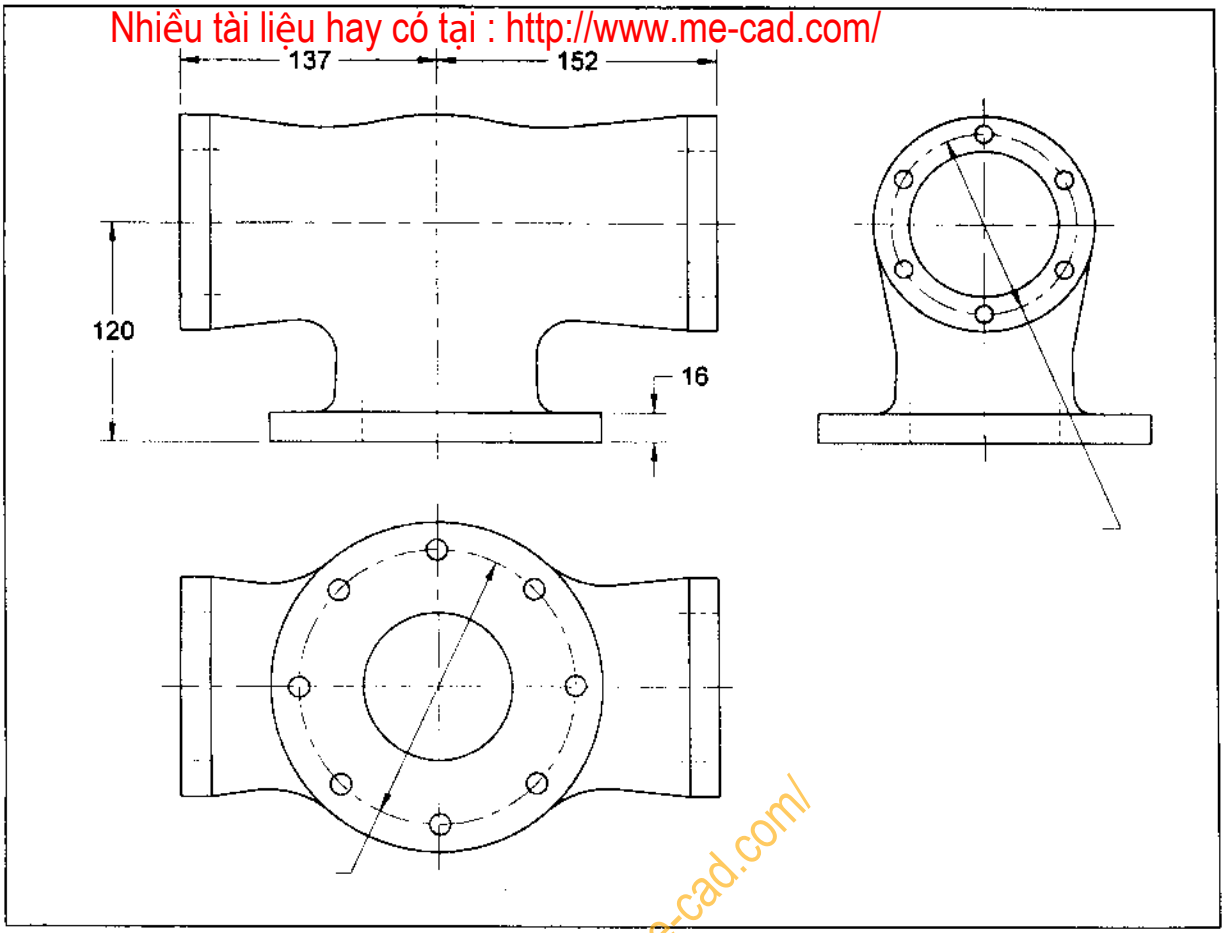
CHƯƠNG TRÌNH VÍ DỤ HOÀN CHỈNH

Chi tiết thông dụng đối với trung tâm gia công ngang yêu cầu cắt gọt trên vài bề mặt trong cùng một gá lắp. Chi tiết vỏ hộp được nêu trên Hình 45.14.

Đối với ví dụ này, chỉ gia công các lỗ trên ba bề mặt khác nhau, sử dụng mũi khoan điểm, hai mũi khoan, và một mũi ta-rô. Bước thứ nhất là chọn zero chương trình. Để dễ lập trình và gá lắp, nên chọn tâm của vòng tròn bu lông và phía trước của từng bề mặt (Z). Từng bề mặt sẽ có bù chi tiết riêng - G54 cho mặt A, G55 cho mặt B, và G56 cho mặt C. Bước thứ hai là viết hai chương trình con cho các vị trí lỗ. Mọi kích thước đều được tính toán chính xác nhưng không cần quá chi tiết. Khi khởi động, dao thứ nhất được lắp sẵn trong trục chính, chi tiết được lắp trong đồ gá trên bàn phân độ. Ví dụ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 45.14. Chi tiết nhiều mặt thích hợp cho gia công ngang chương trình O4503 (các chương trình con O4553 và O4554).

này bỏ qua sự thay đổi pallet, nhưng sẽ giải thích trong phần kế tiếp. Các chương trình con chứa các tọa độ sơ đồ lỗ bu lông.

```
O4553 (SUBPROGRAM FOR 8 HOLES AT 148 MM BCD)
N101 X74.0 Y0
N102 X52.326 Y52.326
N103 X0 Y74.0
N104 X-52.326 Y52.326
N105 X-74.0 Y0
N106 X-52.326 Y-52.326
N107 X-0 Y-74.0
N108 X52.326 Y-52.326
N109 M99
%

O4554 (SUBPROGRAM FOR 6 HOLES AT 99 MM BCD)
N201 X49.5 Y0
N202 X24.75 Y42.868
N203 X-24.75 Y42.868
N204 X-49.5 Y0
N205 X-24.75 Y-42.868
N206 X24.75 Y-42.868
N207 M99
%

O4503 (MAIN PROGRAM)
(FACE A = G54 = B0 = 8 HOLES)
(FACE B = G55 = B90.0 = 6 HOLES)
(FACE C = G56 = B270.0 = 6 HOLES)

(T01 - 15 MM DIA SPOT DRILL)
(T02 - 8.4 MM TAP DRILL)
(T03 - M10 X 1.5 TAP)
(T04 - 11 MM DIA DRILL)

(T01 - 15 MM DIA SPOT DRILL - ALL HOLES)
N1 G21
N2 G17 G40 G80
/N3 G91 G28 Z0
/N4 G28 X0 Y0
/N5 M79
/N6 G28 B0
/N7 M78
N8 G90 G54 G00 X74.0 Y0 S868 M03 T02
N9 G43 Z10.0 H01 M08
N10 G99 G82 R2.0 Z-5.8 P200 F150.0 L0
N11 M98 P4553              (SPOT DRILL FACE A)
N12 G80 Z300.0
N13 M79
N14 B90.0
N15 M78
N16 G55 X49.5 Y0 Z10.0
N17 G99 G82 R2.0 Z-5.3 P200 L0
N18 M98 P4554              (SPOT DRILL FACE B)
N19 G80 Z300.0
N20 M79
N21 B270.0
N22 M78
N23 G56 X49.5 Y0 Z10.0
N24 G99 G82 R2.0 Z-5.3 P200 L0
N25 M98 P4554              (SPOT DRILL FACE C)
N26 G80 Z300.0 M09
N27 G91 G28 Y0 Z0 M05
N28 M01
```

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
(T02 - 8.4 MM TAP DRILL)
N29 T02
N30 M06
N31 G90 G56 G00 X49.5 Y0 S1137 M03 T03
N32 G43 Z10.0 H02 M08
N33 G99 G83 R2.0 Z-24.8 Q6.0 F200.0 L0
N34 M98 P4554                    (TAP DRILL FACE C)
N35 G80 Z300.0
N36 M79
N37 B90.0
N38 M78
N39 G55 X49.5 Y0 Z10.0
N40 G99 G83 R2.0 Z-24.8 Q6.0 L0
N41 M98 P4554                    (TAP DRILL FACE B)
N42 G80 Z300.0 M09
N43 G91 G28 Y0 Z0 M05
N44 M01

(T03 - M10 X 1.5 TAP)
N45 T03
N46 M06
N47 G90 G55 G00 X49.5 Y0 S550 M03 T04
N48 G43 Z10.0 H03 M08
N49 G99 G84 R5.0 Z-23.0 F825.0 L0
N50 M98 P4554                          (TAP FACE B)
N51 G80 Z300.0
N52 M79
N53 B270.0
N54 M78
N55 G56 X49.5 Y0 Z10.0
N56 G99 G84 R5.0 Z-23.0 L0
N57 M98 P4554                          (TAP FACE C)
N58 G80 Z300.0 M09
N59 G91 G28 Y0 Z0 M05
N60 M01

(T04 - 11 MM DIA DRILL)
N61 T04
N62 M06
N63 M79
N64 B0
N65 M78
N66 G90 G54 G00 X74.0 Y0 S800 M03 T01
N67 G43 Z10.0 H04 M08
N68 G99 G81 R2.0 Z-20.3 P200 F225.0 L0
N69 M98 P4553                        (DRILL FACE A)
N70 G80 Z300.0 M09
N71 G91 G28 X0 Y0 Z0 M05
N72 M30
%
```

Trong ví dụ này chỉ vài chú thích. Cả chương trình và hai chương trình con đều rõ ràng. So sánh với ứng dụng gia công đứng, khoảng hở an toàn trục Z dường như hơi quá cao với Z300.0 được lập trình trước mỗi lần phân độ. Các khoảng hở lớn là để bảo đảm an toàn cho phép chi tiết và bàn phân độ có thể phân độ trong vùng an toàn, không có các vật cản. Sẽ không thuận tiện khi tính toán khoảng hở Z tối thiểu, nhưng điều quan trọng là phải đủ xa đối với *tất cả* các bề mặt gia công. Phần mềm CAD có thể giúp xác định khoảng hở Z.

BỘ THAY PALLET TỰ ĐỘNG - APC

Một trong các vấn đề lớn nhất trong gia công CNC là thời gian phi sản xuất cần thiết để gá lắp chi tiết ban đầu và lắp lại chi tiết trong gia công hàng loạt. Nhiều tính năng tích hợp trong hệ điều khiển hoặc trong kết cấu máy có thể giảm thời gian phi sản xuất một cách đáng kể. Chúng bao gồm bù chiều dài dao, bù chi tiết, bù bán kính dao, v.v... Tuy nhiên, các tính năng đó không thể giải quyết vấn đề thời gian gá lắp từng chi tiết trên bàn máy. Bước đột phá có lẽ là sự sử dụng bàn *pallet* trên máy CNC. Pallet không phải là ý tưởng mới trong gia công. Đối với trung tâm gia công ngang, pallet giao diện trở thành tính năng thực tiễn để giảm thời gian gá lắp. Theo truyền thống, mỗi máy có một bàn làm việc. Kiểu thiết kế máy công cụ này có một nhược điểm - trong khi máy hoạt động (còn người vận hành có thể chỉ quan sát), hầu như không thực hiện công việc khác. Điều đó có nghĩa là sự gá lắp chi tiết kế tiếp được thực hiện trong khi máy tạm dừng, đó là thời gian phi sản xuất.

Theo định nghĩa, pallet tự động là bàn làm việc có thể dịch chuyển vào và ra khỏi vị trí gia công theo lệnh chương trình. Nếu mục đích của thiết kế này là giảm thời gian phi sản xuất, cần có ít nhất hai pallet độc lập khả dụng - trong khi gia công chi tiết trên một pallet, pallet thứ hai được dùng để thay đổi gá lắp cho công việc kế tiếp hoặc tháo và lắp các chi tiết riêng rẽ. Theo cách đó, gia công và gá lắp có thể thực hiện đồng thời, rút ngắn hoặc hoàn toàn loại bỏ thời gian phi sản xuất.

Tuy hệ thống hai pallet là rất phổ biến trên trung tâm gia công ngang, nhưng thực tế có các kết cấu sử dụng đến 12 bộ pallet.

http://www.me-cad.com/

Môi trường làm việc

Đối với bộ thay pallet kép, cần phân biệt hai vùng chính:

- Vùng gia công ... trong phạm vi máy
- Vùng gá lắp ... ngoài phạm vi máy

Một pallet thường trong vùng gia công, pallet kia trong vùng gá lắp. Khi chương trình khởi động, thường bắt đầu với Pallet # 1 (có lắp chi tiết) trong vùng gia công và Pallet #2 (chưa có chi tiết) trong vùng gá lắp. Có nhiều thiết kế pallet, nhưng chúng đều có ba bộ phận chính:

- Pallet
- Bộ định vị máy
- Hệ thống truyền động

Pallet là bàn làm việc di động với bề mặt nền để lắp đồ gá và chi tiết. Bàn này có thể có các rãnh T, các lỗ côn, hoặc cả hai.

Bộ định vị máy (còn gọi là bộ nhận) là thiết bị đặc biệt bố trí bên trong máy. Mục đích là tiếp nhận và kẹp giữ pallet có lắp chi tiết sẵn

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

sàng gia công bộ này nhất đủ cứng vững và chính xác.

Hệ thống truyền động (còn gọi là bộ tải pallet) là hệ thống truyền động pallet giữa khu vực tải và khu vực dỡ tải.

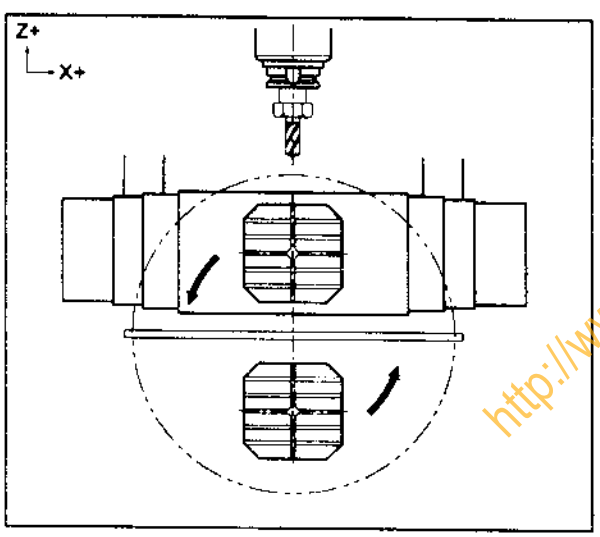
Tải có nghĩa là đưa pallet vào khu vực gia công, *dỡ tải* là đưa pallet vào khu vực gá lắp. Hệ thống truyền động xác định kiểu loại pallet.

Các kiểu pallet

Có hai kiểu pallet tổng quát, dựa trên hệ thống truyền động.

- Kiểu quay
- Kiểu đẩy

Kiểu *quay* vận hành theo nguyên lý bàn xoay, một pallet phía ngoài máy, pallet kia ở bên trong máy. Lệnh thay pallet sẽ quay các pallet 180^0 và sự lập trình rất đơn giản. Hình 45.15 minh họa pallet kiểu quay.



Hình 45.15. Kiểu quay của bộ thay pallet

Kiểu *đẩy* có hai bộ thanh ray giữa vùng tải và bộ phận bên trong máy (Hình 45.16). Sự lập trình hỏi phức tạp hơn so với kiểu quay.

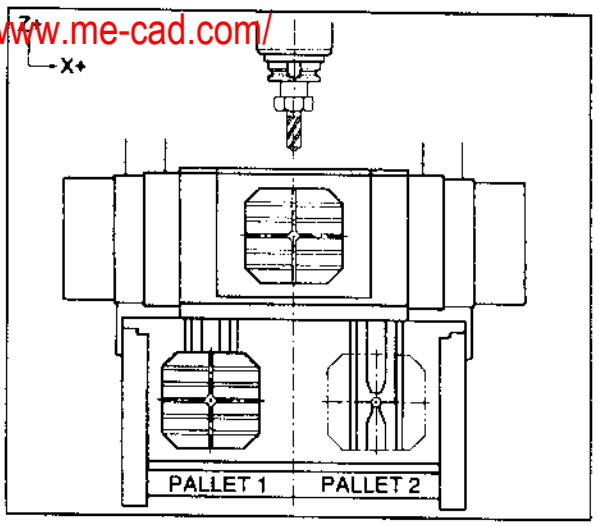
Cả hai kiểu pallet đều được tải từ vùng phía trước máy. Còn có các kiểu pallet khác chuyển dùng cho một số ứng dụng đặc biệt.

Các lệnh lập trình

Hàm M tiêu chuẩn đối với sự thay pallet tự động là M60.

M60	Thay pallet tự động (APC)

Lệnh này chỉ hoạt động chuẩn xác khi vị trí pallet ở một trong hai điểm quy chiếu máy.



Hình 45.16. Kiểu đẩy của bộ thay pallet.

G28	Trả máy về điểm quy chiếu sơ cấp
G30	Trả máy về điểm quy chiếu thứ cấp

Lệnh G28 đã được trình bày trong các chương trước. Lệnh G30 hoạt động tương tự nhưng di chuyển các trục đã chọn đến vị trí quy chiếu máy thứ cấp.

Cấu trúc chương trình thay pallet

Chương trình dưới đây tập trung vào sự thay pallet trên hệ thống pallet kiểu đẩy, có thể dễ dàng chuyển đổi cho hệ thống pallet kiểu quay. Trong cả hai trường hợp, một pallet trong vùng gia công, pallet thứ hai trong vùng gá lắp.

```
O4504
G91 G28 X0 Y0 Z0
G28 B0
M60                              (LOAD PALLET 1)
<... gia công trên pallet 1 ...>
G91 G28 X0 Y0 Z0
G28 B0
M60                            (UNLOAD PALLET 1)
G30 X0
M60                              (LOAD PALLET 2)
<... gia công trên pallet 2 ...>
G30 X0
M60                            (UNLOAD PALLET 2)
M30
%
```

PHAY DOA NGANG

Chương này sẽ không đầy đủ nếu không trình bày vài điều về máy *phay doa ngang*. Máy phay doa CNC tương tự trung tâm gia công ngang CNC, nhưng thường có kích cỡ lớn hơn, có hoặc không có bộ thay dao tự động, chuyển động trục chính được phân chia theo hai trục Z và W. Dưới đây là các xác lập điển

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

hình của máy phay doa ngang 4 trục với trục phân độ B và bộ điều khiển Fanuc:

- Sáu lệnh bù chi tiết ... G54 đến G59.
- Hai điểm quy chiếu máy ... G28 và G30

Dù được gán đến năm trục, nhưng máy phay doa ngang vẫn chỉ là máy bốn trục. Các trục đó bao gồm:

- Trục X ... dọc bàn máy
- Trục Y ... cột
- Trục Z ... đầu trục chính
- Trục W ... ngang bàn máy
- Trục B ... bàn xoay hoặc phân độ.

Các xác lập tương tự trung tâm gia công, cộng với trục W. Trong khi gá lắp, các giá bị bù chi tiết sẽ được xác lập như sau:

- G54 X = Âm
  Y = Âm
  Z = Zero
  W = Âm
  B = Zero

Do nhiều máy phay doa ngang không có bộ thay dao tự động, vị trí G30 cần được xác lập một cách thuận tiện cho người vận hành để thay dao bằng tay (các trục, X, Y, W). Vị trí này được xác lập bằng tham số hệ thống. Giá trị trục Z là chiều dài của hành trình đầu trục phía ngoài trục chính.

Định dạng lập trình dựa trên nguyên tắc mọi chuyển động theo chiều sâu đều thực hiện trên trục W, thay vì trục Z. Đầu trục chính được điều khiển theo trục Z, chỉ được kéo ra để có khoảng hở; sự kéo ra khỏi trục chính phải đủ dài để bảo đảm đủ khoảng hở cho dao ngắn nhất được dùng trong chương trình.

Định dạng lập trình dưới đây kèm theo giải thích chi tiết. Các dòng [nn] chỉ để quy chiếu theo giải thích:

```
[01] O4505 (TÊN CHƯƠNG TRÌNH)
[02] (GHI CHÚ HOẶC GIẢI THÍCH)
[03] N10 G21
[04] N20 G91 G30 W0
[05] N30 G90 S.. M03
[06] N40 G54 G00 X.. Y..
[07] N50 G43 Z..
[08] N60 G43 W.. H.
[09] N70 G01 W.. F..
[10] . . .
[11] . . .
[12] . . . <gia công>
[13] . . .
[14] . . .
[15] N350 G00 W..
[16] N360 M05
[17] N380 Z0
[18] N390 G30 G49 W0
[19] N400 G91 G30 X0 Y0
[20] N410 M06
[21] . . .
[22] . . .
[23] . . .
[24] N600 M30
[25] %
```

Các giải thích dưới đây tương ứng với số dòng [nn] trong ví dụ:

[01] Tên chương trình (đến 16 ký tự).
[02] Ghi chú cho người vận hành - trong ngoặc đơn.
[03] Chọn đơn vị đo, hệ mét hoặc Anh.
[04] Trục W chuyển động đến vị trí thay dao (theo số gia để bảo đảm an toàn).
[05] Chọn chế độ tuyệt đối và hàm trục chính.
[06] Chuyển động nhanh đến vị trí khởi đầu theo XY trong hệ tọa độ làm việc G54.
[07] Đầu trục chính đi ra theo giá trị tham số 1241 (Z).
[08] Bù chiều dài dao (xác lập từ đỉnh dao đến zero chương trình) và chuyển động đến khoảng hở.
[09] Chuyển động ăn dao đến chiều sâu theo yêu cầu.
[10] ...
[11] ...
[12] ... (gia công chi tiết) ...
[13] ...
[14] ...
[15] Chuyển động nhanh trở về khoảng hở (xem [08]).
[16] Dừng trục chính.
[17] Chuyển động nhanh, lùi đầu trục chính vào trục chính.
[18] Chuyển động nhanh đến vị trí thay dao dọc theo trục W và xóa bù chiều dài dao.
[19] Chuyển động nhanh đến vị trí thay dao dọc theo trục X và Y, theo chế độ số gia để bảo đảm an toàn.
[20] Thay dao bằng tay.
[21] ...
[22] ...(Tiếp tục gia công, theo định dạng nêu trên...)
[23] ...
[24] Kết thúc chương trình
[25] Dừng ghi (mã dừng)

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

46 VIẾT CHƯƠNG TRÌNH CNC

Viết chương trình CNC là kết quả cuối cùng của lập trình bằng tay. Chương trình gồm các lệnh riêng rẽ liên quan với sự gia công, và được sắp xếp theo chuỗi thứ tự các block. Viết chương trình không có nghĩa là chỉ sử dụng giấy bút. Các phương pháp viết hiện đại sử dụng máy tính và bộ biên tập văn bản, nhưng kết quả vẫn là văn bản của chương trình gia công được tạp ra một cách thủ công.

Nhu cầu lập trình thủ công dường như hơi lạc hậu trong thời đại máy tính, máy in, và các thiết bị công nghệ cao khác, nhưng đây là phương pháp còn được sử dụng tiếp tục trong thời gian dài. Viết chương trình một cách thủ công đòi hỏi thời gian và thường có các sai sót. Tuy viết chương trình bằng tay nhưng vẫn đòi hỏi các kỹ năng sử dụng máy tính.

Bất kể phương tiện được sử dụng, bạn cần biết phương pháp máy tính - hệ điều khiển - diễn dịch chương trình đã viết, cú pháp cần sử dụng, và loại định dạng tương ứng. Dù hoàn toàn không viết chương trình thủ công, điều quan trọng vẫn là nắm vững các kỹ thuật viết chương trình, để có thể thực hiện các thay đổi trong chương trình bất kỳ được viết hệ thống CAD/ CAM, nếu cần thiết.

Chương trình CNC phải được viết sao cho có thể dễ dàng diễn dịch

VIẾT CHƯƠNG TRÌNH

Viết mọi dữ liệu đã thu thập vào phiên bản cuối cùng của chương trình gia công CNC là một trong các bước cuối cùng của quy trình lập trình. Để đi đến giai đoạn này cần làm việc một cách nỗ lực qua các bước khác. Trong các chương trình đã nhấn mạnh tính logic trong khi viết chương trình. Phần này sẽ tập trung vào phương pháp viết chương trình CNC, tuân theo quá trình logic.

Viết chương trình dựa trên hai yếu tố ban đầu.

- ❑ Các tiêu chuẩn kỹ thuật ... do công ty quyết định
- ❑ Phong cách cá nhân ... do bạn quyết định

Cả hai yếu tố này đều có thể được phối hợp đồng thời trong một chương trình chúng hoàn toàn tương thích với nhau. Sẽ không hợp lý khi mong muốn có *các tiêu chuẩn toàn cầu hoặc tiêu chuẩn công nghiệp* liên quan đến các kỹ thuật lập trình khác nhau, trừ khi có tập hợp các nguyên tắc chung và các khuyến nghị về lập trình.

Yếu tố thứ hai là phong cách cá nhân. Người lập trình không thể tự cô lập, luôn luôn phải làm việc theo nhóm. Chương trình kết quả sẽ có ít nhất một người sử dụng, người vận hành CNC.

Các dạng tập trình

Vào thời kỳ đầu của điều khiển số, dạng lập trình đặc biệt ưa chuộng là dùng các cột in sẵn cho từng địa chỉ trong block. Khi đó chỉ nhập các giá trị số vào cột tương ứng, vị trí cột xác định ý nghĩa của địa chỉ. Các dụng chương trình này thường do nhà chế tạo máy và bộ điều khiển cung cấp, hỗ trợ cho việc viết chương trình. Ngày nay, chỉ cần viết chương trình trên giấy bình thường. Các chương trình hiện đại sử dụng hệ thống chữ và số cùng với các ký hiệu đặc biệt.

Các ký tự dễ bị nhầm lẫn

Tính rõ ràng, dễ đọc của chương trình viết tay là rất quan trọng. Bạn cần chú ý khi viết một số ký tự (chữ hoặc số) có thể bị hiểu theo nhiều cách. Tùy theo các nét chữ cá nhân, một số ký tự có thể gây nhầm lẫn cho người đọc. Ví dụ, chữ O và số 0 viết tay có thể rất giống nhau; chữ I, số 1, và chữ l đôi khi hơi khó phân biệt. Bạn nên có thói quen viết chữ ổn định để phân biệt rõ các ký tự có thể bị nhầm lẫn.

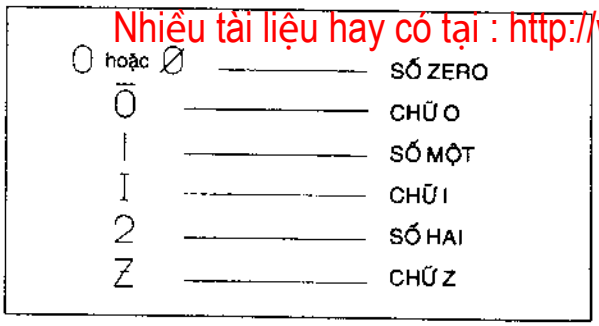
Ví dụ, mọi máy tính và máy in (kể cả hệ thống cũ sử dụng băng từ) đều có phương pháp đặc biệt để phân biệt các ký tự riêng rẽ trên màn hình và trên bản in. Trong cuốn sách này có sự khác biệt rõ ràng giữa chữ O rộng (như trong `O1111`) và số 0 hẹp (như trong `O0001`).

Kỹ thuật này cũng được áp dụng cho bản viết tay. Trên nhiều bộ điều khiển, hầu như không sử dụng chữ O, trừ phần chỉ số chương trình và trong các chú thích, trong đó sự in sai cũng không gây ra vấn đề. Nếu cần, bạn nên dùng ký hiệu đặc biệt cho chữ O, và phần còn lại là số 0 theo mặc định.

Hình 46.1 minh họa một số phương pháp *phân biệt* các ký tự khi viết tay. Bạn nên chọn

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/



Hình 46.1. Dạng đặc biệt của các ký tự viết để tránh nhầm lẫn.

phương pháp riêng của mình để cải thiện tính rõ ràng và dễ đọc của bản viết tay. Điều quan trọng là suy trì phương pháp đó một cách liên tục và ổn định.

Bạn hoàn toàn có thể bỏ qua phương pháp viết tay bằng cách dùng bàn phím và máy tính. Chương trình sau đó có thể tối ưu hóa, chi tiết được gia công và chương trình kiểm chứng có thể được gởi hoặc lưu một cách dễ dàng. Cách tốt nhất và nhanh nhất là chuẩn bị chương trình trong bộ biên tập văn bản trên máy tính và gởi chương trình đó vào máy CNC một cách trực tiếp thông qua nối kết cáp.

ĐỊNH DẠNG XUẤT CHƯƠNG TRÌNH

Nếu bạn đọc cuốn sách này từ đầu, lần lượt từng chương, bạn đã làm quen với nhiều kỹ thuật lập trình. Phần này sẽ trình bày định dạng chương trình thực tế, không đề cập đến *nội dung*, chỉ nêu *hình thức* chương trình xuất hiện trên giấy sau khi in hoặc trên màn hình máy tính. Bạn hãy đánh giá bốn phiên bản của *cùng một* chương trình. Chúng hoàn toàn như nhau về nội dung, chỉ khác về hình thức. Bạn không cần chú ý đến nội dung chương trình, chỉ cần đánh sự thay đổi hình thức qua từng phiên bản.

➲ Phiên bản 1

```
G20
G17G40G80G49
T01M06
G90G54G00X-32500Y0S900M03T02
G43Z10000H01M08
G99G82X-32500Y0R1000Z-3900P0500F80
X32500Y32500
X0
X-32500
Y0
Y-32500
X0
X32500
G80G00Z10000M09
G28Z10000M05
M01
T02M06
G90G54G00X-32500Y0S750M03T03
G43Z10000H02M08
G99G81X-32500Y0R1000Z-22563F120
X32500Y32500
X0
X-32500
Y0
Y-32500
X0
X32500
G80G00Z10000M09
G28Z10000M05
M01
T03M06
G90G54G00X-32500Y0S600M03T01
G43Z10000H03M08
G99G84X-32500Y0R5000Z-13000F375
X32500Y32500
X0
X-32500
Y0
Y-32500
X0
X32500
G80G00Z10000M09
G28X32500Y-32500Z10000M05
M30
%
```

Đây là hình thức thô sơ nhất để viết chương trình. Người vận hành CNC sẽ gặp nhiều khó khăn khi đọc chương trình này.

➲ Phiên bản 2

```
N1G20
N2G17G40G80G49
N3T01M06
N4G90G54G00X-3.25Y0S900M03T02
N5G43Z1.0H01M08
N6G99G82X-3.25Y0R0.1Z-0.39P0500F8.0
N7X3.25Y3.25
N8X0
N9X-3.25
N10Y0
N11Y-3.25
N12X0
N13X3.25
N14G80G00Z1.0M09
N15G28Z1.0M05
N16M01
N17T02M06
N18G90G54G00X-3.25Y0S750M03T03
N19G43Z1.0H02M08
N20G99G81X-3.25Y0R0.1Z-2.2563F12.0
N21X3.25Y3.25
N22X0
N23X-3.25
N24Y0
N25Y-3.25
N26X0
N27X3.25
N28G80G00Z1.0M09
N29G28Z1.0M05
N30M01
N31T03M06
N32G90G54G00X-3.25Y0S600M03T01
N33G43Z1.0H03M08
N34G99G84X-3.25Y0R0.5Z-1.3F37.5
N35X3.25Y3.25
N36X0
N37X-3.25
N38Y0
N39Y-3.25
N40X0
N41X3.25
N42G80G00Z1.0M09
```


Nhiều tài liệu hay có tại : http://www.me-cad.com/

```
N43G28X3.25Y-3.25Z1.0M05
N44M30
%
```

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Đây là phiên bản cải tiến, các block đều được đánh số, ở các giá trị đều có dấu thập phân, do đó chương trình dễ đọc hơn.

➲ Phiên bản 3

```
N1 G20
N2 G17 G40 G80 G49
N3 T01 M06
N4 G90 G54 G00 X-3.25 Y0 S900 M03 T02
N5 G43 Z1.0 H01 M08
N6 G99 G82 X-3.25 Y0 R0.1 Z-0.39
N7 X3.25 Y3.25-
N8 X0
N9 X-3.25
N10 Y0
N11 Y-3.25
N12 X0
N13 X3.25
N14 G80 G00 Z1.0 M09
N15 G28 Z1.0 M05
N16 M01
N17 T02 M06
N18 G90 G54 G00 X-3.25 Y0 S750 M03 T03
N19 G43 Z1.0 H02 M08
N20 G99 G81 X-3.25 Y0 R0.1 Z-2.2563 F12.0
N21 X3.25 Y3.25
N22 X0
N23 X-3.25
N24 Y0
N25 Y-3.25
N26 X0
N27 X3.25
N28 G80 G00 Z1.0 M09
N29 G28 Z1.0 M05
N30 M01
N31 T03 M06
N32 G90 G54 G00 X-3.25 Y0 S600 M03 T01
N33 G43 Z1.0 H03 M08
N34 G99 G84 X-3.25 Y0 R0.5 Z-1.3 F37.5
N35 X3.25 Y3.25
N36 X0
N37 X-3.25
N38 Y0
N39 Y-3.25
N40 X0
N41 X3.25
N42 G80 G00 Z1.0 M09
N43 G28 X3.25 Y-3.25 Z1.0 M05
N44 M30
%
```

Phiên bản này được cải tiến rõ rệt, sử dụng *khoảng trắng giữa các từ ngữ*, nhưng vẫn hơi khó xác định sự khởi đầu của dụng cụ cắt. Phiên bản kế tiếp sẽ bổ sung sự cách dòng giữa các dao cắt. Các khoảng trắng không làm tăng thêm dung lượng nhớ cần thiết, chỉ giúp cho chương trình dễ đọc hơn.

➲ Phiên bản 4

```
(DRILL-04.NC)
(PETER SMID - 07-DEC-01 - 19:43)

(T01 - 1.0 DIA - 90DEG SPOT DRILL)
(T02 - 11/16 TAPDRILL - THROUGH)
(T03 - 3/4-16 TPI PLUG TAP)

(T01 - 1.0 DIA - 90DEG SPOT DRILL)
N1 G20
N2 G17 G40 G80 G49
N3 T01 M06
N4 G90 G54 G00 X-3.25 Y0 S900 M03 T02
N5 G43 Z1.0 H01 M08              (INITIAL LEVEL)
N6 G99 G82 X-3.25 Y0 R0.1 Z-0.39 P0500 F8.0
                                 (HOLE 1)
N7 X3.25 Y3.25                   (HOLE 2)
N8 X0                            (HOLE 3)
N9 X-3.25                        (HOLE 4)
N10 Y0                           (HOLE 5)
N11 Y-3.25                       (HOLE 6)
N12 X0                           (HOLE 7)
N13 X3.25                        (HOLE 8)
N14 G80 G00 Z1.0 M09
N15 G28 Z1.0 M05
N16 M01

(T02 - 11/16 TAP DRILL - THROUGH)
N17 T02 M06
N18 G90 G54 G00 X-3.25 Y0 S750 M03 T03
N19 G43 Z1.0 H02 M08
N20 G99 G81 X-3.25 Y0 R0.1 Z-2.2563 F12.0
                                 (HOLE 1)
N21 X3.25 Y3.25                  (HOLE 2)
N22 X0                           (HOLE 3)
N23 X-3.25                       (HOLE 4)
N24 Y0                           (HOLE 5)
N25 Y-3.25                       (HOLE 6)
N26 X0                           (HOLE 7)
N27 X3.25                        (HOLE 8)
N28 G80 G00 Z1.0 M09
N29 G28 Z1.0 M05
N30 M01

(T03 - 3/4-16 PLUG TAP)
N31 T03 M06
N32 G90 G54 G00 X-3.25 Y0 S600 M03 T01
N33 G43 Z1.0 H03 M08
N34 G99 G84 X-3.25 Y0 R0.5 Z-1.3 F37.5(HOLE 1)
N35 X3.25 Y3.25                  (HOLE 2)
N36 X0                           (HOLE 3)
N37 X-3.25                       (HOLE 4)
N38 Y0                           (HOLE 5)
N39 Y-3.25                       (HOLE 6)
N40 X0                           (HOLE 7)
N41 X3.25                        (HOLE 8)
N42 G80 G00 Z1.0 M09
N43 G28 X3.25 Y-3.25 Z1.0 M05
N44 M30
%
```

http://www.me-cad.com/

Đây là phiên bản rõ ràng và đầy đủ nhất, có tên nhà lập trình, thời gian cập nhật gần nhất, các chú thích cho người vận hàng CNC, mô tả các dụng cụ cắt ở đầu chương trình và ở từng phần tương ứng.

CHƯƠNG TRÌNH DÀI

Các bộ điều khiển cũ sử dụng đục lỗ để lưu chương trình. Chiều dài chương trình tối đa bằng chiều dài của băng lắp trong cuộn băng, khoảng 900 feet, tương đương 275 mét, có thể chứa đến 108000 ký tự, với thiết bị hiện đại ngày nay, hoàn toàn không sử dụng băng đục lỗ, hầu hết các chương trình gia công đều chạy từ độ nhớ của hệ thống CNC. Thật không may, dung lượng nhớ bị giới hạn, thậm chí thấp hơn băng đục lỗ. Điều đó có thể gây ra vấn đề, khi bộ nhớ không đủ dung lượng chứa chương trình dài. Ngoài việc thu gọn thư mục nhớ, còn có hai

Nhiều tài liệu hay có tại : http://www.me-cad.com/

khả năng cho phép giải quyết vấn đề thiếu dung lượng nhớ.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Rút gọn chương trình

Phương pháp đơn giản để giảm chiều dài chương trình là bỏ mọi ký tự *không cần thiết* ra khỏi chương trình. Do vấn đề liên quan với *chương trình dài*, sự giảm chiều dài thực sẽ lớn hơn so với minh họa ở đây. Có nhiều lĩnh vực cần khảo sát trước khi bắt đầu các thay đổi:

- Loại bỏ mọi chữ số 0 đứng trước hoặc đứng sau không cần thiết (G00 = G0, x 0.0100 = X.01....)
- Loại bỏ mọi chữ số 0 lập trình để tăng tính thuận tiện (Ví dụ X 2.0 = X2.)
- Loại bỏ toàn bộ hoặc hầu hết các số block.
- Nếu sử dụng các số block, sự tăng theo đơn vị (1) sẽ rút ngắn chương trình.
- Kết hợp vài chương trình dao đơn vào chuyển động dao nhiều trục, nếu điều kiện an toàn cho phép.
- Sử dụng các xác lập điều khiển mặc định, nhưng cần kiểm tra trước.
- Không đưa các giải thích vào chương trình.
- Sử dụng các chú thích, giải thích và mô tả trên tờ giấy riêng.

Tổ chức các bước lập trình sẽ cho phép rút gọn chương trình chẳng hạn gộp tối đa số lệnh có thể vào một block, thay vì chia chúng vào nhiều block riêng rẽ. Sử dụng các chương trình con, nếu có thể, giảm số lần thay dao, thậm chí sử dụng ít dao hơn, nếu điều kiện gia công cho phép. Đồng thời, cần xem xét các hậu quả khi loại bỏ hoặc điều chỉnh định dạng chương trình đã được thiết lập.

Các biện pháp nêu trên đòi hỏi sự thỏa hiệp giữa sự thuận tiện và tính cần thiết. Khi suy nghĩ cẩn thận và tổ chức công việc một cách hợp lý, kết quả sẽ xứng đáng với nỗ lực đó.

Chỉ dùng các biện pháp này khi thực sự cần thiết, không nên coi đó là kỹ thuật lập trình tiêu chuẩn.

Để minh họa các phương pháp rút gọn chương trình, bạn hãy so sánh hai ví dụ dưới đây, cả hai đều có cùng kết quả - *hầu như* cùng kết quả.

```
O4601 (TYPICAL PROGRAM)
N10 G21 G17 G40 G80 G90
N20 G54 G00 X120.0 Y35.0
N30 G43 Z25.0 H01
N40 S500 M03
N50 M08
N60 G99 G81 X120.0 Y35.0 R3.0 Z-10.0 F100.0
N70 X150.0
N80 Y55.0
N90 G80 G00 Z25.0
N100 M09
N110 G28 X150.0 Y55.0 Z25.0
N120 M30
%
```

Tổng cộng đã lập trình 194 ký tự. Phiên bản thu gọn của chương trình này chỉ cần 89 ký tự. Chương trình rút gọn sử dụng dung lượng nhớ ít hơn, do đó hiệu quả hơn, nhưng khó đọc hơn.

```
O4602
G90 G0 X120. Y35.
G43 Z25. H1 S500 M3
M8
G99 G81 R3. Z-10. F100.
X150.
Y55.
G80 Z25. M9
G91 G28 X0 Y0 Z0
M30
%
```

Sự rút gọn chương trình có thể rất hữu ích trong một số trường hợp, các phương pháp đã áp dụng cho ví dụ nêu trên bao gồm:

- Loại bỏ sự mô tả chương trình.
- Loại bỏ các số block.
- Loại bỏ G21, G17, và G54 (xác lập đúng dựa vào bộ điều khiển nhưng cần rất cẩn thận).
- Các số zreo sau dấu thập phân đều bị xóa.
- Kết hợp một số block với nhau.
- G80 và G00 được thay bằng G80 (G00 là dư, nhưng vẫn được sử dụng thường xuyên)
- Các số 0 đứng trước trong G00, M08, M09, H01 và M03 bị loại bỏ.
- Trả về zero máy được đổi từ chế độ tuyệt đối sang chế độ số gia.

Mặc dầu cả hai chương trình này đều gia công chi tiết theo các yêu cầu kỹ thuật trên bản vẽ, nhưng một số lệnh lập trình sẽ được xử lý khác nhau. Có thể đạt được sự thay đổi quan trọng trong sự tiếp cận dao hướng đến chi tiết. Trong ví dụ thứ nhất (phiên bản tiêu chuẩn), lệnh chuyển động định vị các trục X và Y trước, sau đó là chuyển động trục Z trong block riêng. Trong ví dụ thứ hai (rút gọn), vì các lý do an toàn, thứ tự các chuyển động không thay đổi. Nếu các điều kiện gia công cho phép, có thể kết hợp hai chuyển động này thành một. Các lệnh G43 và G54 có thể làm việc với nhau trong một block mà không gây ra vấn đề:

```
G90 G0 G43 G54 X120. Y35. Z25. H1 S500 M3
```

Luôn luôn cẩn thận, xem xét sự gá lắp trước, và bảo đảm sự chuyển động nhanh an toàn tiến đến hoặc ra xa chi tiết. Nếu có các vật cản trên đường chuyển động do rút gọn chương trình, bạn cần xử lý chính xác và hợp lý.

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Sự chuẩn bị và viết chương trình sẽ dần dần trở thành thói quen sau khi thiết lập phong cách lập trình riêng. Nếu sử dụng máy tính, bạn nên tập cách viết chương trình trực tiếp từ bàn phím sẽ lãng phí thời gian nếu bạn viết bằng tay trước.

Chế độ nhớ và chế độ băng

Hầu hết các hệ thống CNC đều có bộ chọn *chuyển đổi chế độ* (Mode Switch) đặc biệt để chọn từ ít nhất hai tùy chọn, chế độ MEMORY (nhớ) và chế độ TAPE (băng). Chế độ Memory được dùng thường xuyên - chương trình được tải vào bộ nhớ CNC, được biên tập và chạy từ bộ nhớ. Chế độ Tape, chạy chương trình từ băng, nhưng nhiều người thường bỏ qua các khả năng của chế độ này. Ngay cả khi hoàn toàn không sử dụng băng đột lỗ trong xưởng cơ khí, chế độ Tape vẫn có thể được dùng để *mô phỏng* băng với nhiều khả năng bổ sung. Bạn không nên hiểu chế độ Tape theo nghĩa đen. Bạn hãy coi đây là chế độ *ngoài*, không nên theo ý nghĩa gốc.

Để sử dụng chế độ ngoài cần có thêm phần cứng và phần mềm. Về phần cứng, chỉ cần máy tính để bàn với dung lượng ổ cứng vừa phải (khoảng 10GB), cáp với cấu hình thích hợp nối máy CNC với máy tính. Về phần mềm, chỉ cần phần mềm truyền thống để gửi chương trình từ máy tính đến máy CNC và ngược lại.

Sau khi kết nối, bạn hãy lưu các chương trình CNC trên ổ cứng máy tính, tải phần mềm tương ứng và làm việc với hệ thống CNC một cách bình thường. Sự khác biệt chính là sự biên tập. Do chương trình trong ổ cứng máy tính, bạn hãy dùng máy tính và bộ biên tập văn bản (chẳng hạn Microsoft Office) để biên tập chương trình CNC, thay vì hệ điều khiển. Dung lượng nhớ của ổ cứng hiện nay rất cao (40 GB hoặc cao hơn). Các công ty khuôn mẫu, các xưởng cơ khí lớn,... đòi hỏi chương trình rất dài, đều sử dụng phương pháp này một cách thành công.

Ngoài ra, bạn hãy xét phương pháp này để áp dụng các chương trình *Gia công Tốc độ cao*. Công nghệ tương đối mới này sử dụng tốc độ trục chính và tốc độ cắt rất cao nhưng với chiều sâu cắt nhỏ. Sự phối hợp này đòi hỏi chương trình rất dài, trong khi dung lượng nhớ của hệ thống CNC hầu như không thể đáp ứng. Sử dụng máy tính với tốc độ truyền dữ liệu đủ cao sẽ giải quyết tốt vấn đề đó.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

47 GIAO DIỆN VÀ CÁC THIẾT BỊ

Chương trình CNC hoàn chỉnh, đã được xử lý lỗi và tối ưu hóa để đạt hiệu suất cao nhất, cần được lưu lại để *sử dụng* hoặc *tham khảo*. Trước khi lưu, chương trình đó phải được tải vào bộ nhớ CNC, kiểu nghiệm và tối ưu hóa. Có nhiều phương pháp tải chương trình gia công hoàn chỉnh vào bộ nhớ CNC. Phương pháp cơ bản nhất, và tốn thời gian nhất là gõ phím chương trình trực tiếp vào máy, sử dụng bảng điều khiển và bàn phím. Đây còn là phương pháp có hiệu quả thấp nhất, dễ dàng dẫn đến các lỗi. Các bộ điều khiển Fanuc có tính năng *Biên tập nền*. Đây là tính năng tiêu chuẩn của hầu hết các bộ điều khiển, cho phép người vận hành nhấn phím (và/hoặc biên tập) trong một chương trình gia công, trong khi bộ điều khiển chạy các thao tác gia công đối với chương trình khác. Tuy nhiên, trong thực tế nhiều người vận hành chưa tận dụng ưu thế của tính năng này, vì nhiều lý do khác nhau.

Để tải chương trình gia công vào bộ nhớ CNC hoặc lấy chương trình ra khỏi bộ nhớ CNC, cần có nối kết phần cứng được gọi là *giao diện dữ liệu*. Giao diện thường là thiết bị điện tử, được thiết kế để truyền thông với máy tính của hệ thống CNC.

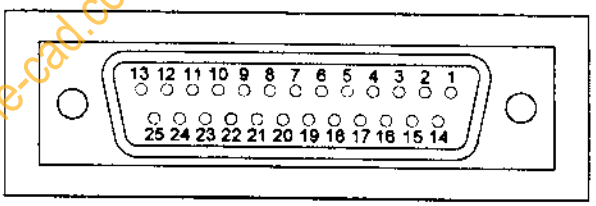
Các giao diện và phương tiện lưu trữ bao gồm:

- Bộ đọc băng và bộ đục lỗ băng
- Băng cassette dữ liệu
- Thẻ dữ liệu
- Băng cassette bọt
- Đĩa mềm
- Đĩa cứng
- Thiết bị tháo được
- Thiết bị ROM (bộ nhớ chỉ đọc)
- v.v...

Nhiều thiết bị loại này có tính đặc thù, một số không chỉ cần sự nối cáp đặc biệt mà còn cả phần mềm (driver) để có thể hoạt động. Chương này sẽ tập trung vào các nối kết có thể dễ dàng lắp đặt và sử dụng các cấu hình tiêu chuẩn. Hiện có một tiêu chuẩn công nghiệp áp dụng chung có hầu hết các thiết bị đó - giao diện RS-232C. Có một số biến thể của tiêu chuẩn này nhưng vẫn áp dụng cùng một nguyên tắc. Phần này không trình bày sâu về các nối kết CNC, chỉ khái quát các tiêu chuẩn có tính hướng dẫn chung.

GIAO DIỆN RS-232C

Truyền dữ liệu giữa hai thiết bị điện tử (máy tính và bộ điều khiển) đòi hỏi một số xác lập sử dụng cùng các nguyên tắc đối với từng thiết bị. Do từng thiết bị có thể được chế tạo từ các công ty khác nhau, do đó phải có tiêu chuẩn độc lập áp dụng chung cho các nhà chế tạo. RS-232C là loại tiêu chuẩn đó - RS là viết tắt của "*Recommended Standard*" (Tiêu chuẩn nên dùng). Hầu như mọi hệ thống CNC, máy tính, bộ đột lỗ băng và bộ đọc băng, đều có bộ nối (được gọi là *cổng*) với ký hiệu RS-232C hoặc tương tự. Cổng này có hai dạng, một với cấu hình 25 *chân*, và một có cấu hình 25 *lỗ*. Dạng có chân được gọi là đầu nối DB-25P, dạng có lỗ được gọi là ổ cắm DB-25S (Hình 47.1).



Hình 47.1. Cổng RS-232C với 25 chân - kiểu DB

Cổng RS-232C trên hệ thống CNC thường là tính năng tiêu chuẩn và sử dụng kiểu DB-25S (với S là viết tắt của Socket - ổ cắm). Máy tính bên ngoài, loại để bàn hoặc laptop (xách tay), cùng với cáp thích hợp và phần mềm truyền thông là cần thiết để truyền các chương trình CNC. Thiết bị bên ngoài chủ yếu sử dụng kiểu đầu nối DB-25P (P là viết tắt của Pin-chân). Chương trình CNC được gởi đến bộ nhớ hệ thống và được lưu ở đây đủ để thực thi công việc. Người vận hành CNC thường thực hiện vài thay đổi và khi công việc hoàn tất, mọi thay đổi vẫn được giữ nguyên và được gởi trở lại máy tính để lưu trên đĩa cứng. Phương pháp này thường áp dụng cho một hoặc nhiều máy CNC.

Mặc dầu các thuật ngữ *Transmit* (hoặc *Send*) và *Receive* rất phổ biến trong công nghệ thông tin, nhưng ngay cả các hệ thống CNC mới nhất vẫn sử dụng thuật ngữ *Punch* (tương đương với *Send*) và *Read* (tương đương với

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Receive). Chúng xuất hiện từ thời kỳ đầu khi hệ thống CNC đều sử dụng băng đục lỗ để lưu chương trình.

Để phương pháp truyền thông này có thể hoạt động, chỉ cần cáp thích hợp nối giữa cổng máy tính và cổng hệ thống CNC. Tải và lập cấu hình phần mềm truyền thông cần được thực hiện trước. Ngoài ra, cả hai thiết bị đều phải xác lập sao cho chúng có thể "nói chuyện" với nhau.

Cuối chương này sẽ nêu vài vấn đề liên quan đến các nguyên tắc cơ bản, về sử dụng máy tính cá nhân và giao diện với hệ thống CNC. Trước hết sẽ xem xét thiết bị giao diện gốc - băng đục lỗ - phương tiện đã được sử dụng nhiều năm trước, nhưng ngày nay hầu như đã được thay thế hoàn toàn bằng máy tính.

BĂNG ĐỤC LỖ

Từ thời kỳ đầu của công nghệ điều khiển số, băng đục lỗ đã là phương tiện chính để chuyển các lệnh chương trình cho hệ thống điều khiển. Cuối thập niên 1980, băng đục lỗ hầu như đã được máy tính cá nhân thay thế.

Băng đục lỗ dễ bị hư và thường khá cồng kềnh dễ bị nhiễm bẩn, nhưng đã từng rất thông dụng, khá kinh tế và vẫn còn khả dụng. Hầu hết các máy CNC ngày nay đều không còn sử dụng bộ đọc băng. Nhiều bộ điều khiển cũ chỉ chấp nhận băng là thiết bị nhập nhưng không chạy chương trình từ băng đục lỗ. Băng này chỉ tải bộ nhớ CNC. Các thay đổi chương trình có thể thực hiện thông qua CNC và sau đó đục lỗ băng để hiệu chỉnh.

Bộ đọc và bộ đục lỗ băng

Một trong các thiết bị ban đầu để truyền dữ liệu là bộ đọc băng được lắp trong các máy NC và CNC cũ. Chức năng của bộ đọc trên máy CNC hoàn toàn khác với trên thiết bị không có CNC. Thay vì sử dụng bộ đọc băng làm nguồn để chạy chương trình, bộ đọc băng trên máy CNC được dùng để *tải* chương trình ghi trên băng giấy vào bộ nhớ hệ thống. Sau khi tải, chương trình sẽ thực thi từ bộ nhớ, trong xác lập *chế độ Memory*, và không cần dùng băng giấy. Phương pháp này có một nhược điểm. Làm việc trên máy CNC thường gặp các thay đổi bắt buộc đối với chương trình, *sau khi* tải vào bộ nhớ. Do các thay đổi này không thể phản ánh trên băng, sau này có thể bị nhầm lẫn, đặc biệt khi lặp lại công việc. Đây là vấn đề tổ chức và có thể dễ dàng giải quyết.

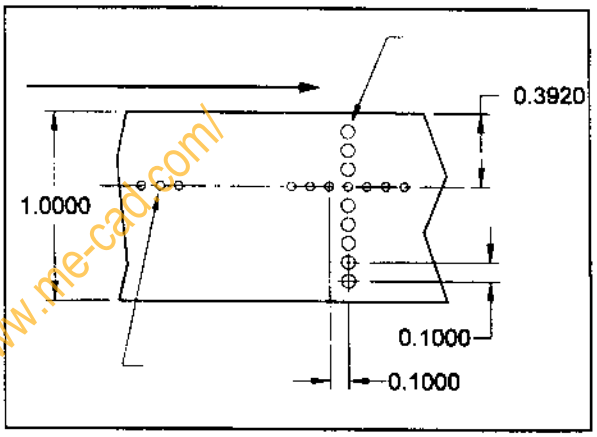
Một tùy chọn là thực hiện mọi thay đổi cần thiết trên hệ thống CNC, sau đó đục lỗ băng mới, sử dụng cổng RS-232C. Khó khăn của phương pháp này là thường có sẵn bộ đọc băng trong máy nhưng không có bộ đục lỗ.

Các xưởng cơ khí hiện đại không còn sử dụng băng, bộ đục lỗ, và bộ đọc băng. Các công cụ này đã được thay bằng máy tính và phần mềm truyền thông.

Tuy công nghệ băng đục lỗ đã lạc hậu so với tiêu chuẩn hiện đại, nhưng vẫn có thể hữu ích đối với những người còn sử dụng công nghệ này hoặc quan tâm đến "lịch sử" của điều khiển số.

Băng đục lỗ

Băng đục lỗ là phương tiện cũ nhất để lưu chương trình. Băng được làm từ giấy chất lượng cao có tăng bền. Băng đục lỗ rộng 1 inch (25.4 mm) và dài 900 inch (274 m) xếp theo cuộn, được sản xuất theo các tiêu chuẩn chính xác (Hình 47.2).



***Hình 47.2.** Băng đục lỗ - các tiêu chuẩn kích thước cơ bản*

Băng giấy thường có màu đen, để có độ mờ đục 100% theo yêu cầu của hầu hết các bộ đọc băng. Băng cũng có thể từ vật liệu khác, chẳng hạn Mylar®, là loại băng giấy ở giữa hai lớp chất dẻo. Chất dẻo làm cho băng bền hơn, đây là yếu tố quan trọng khi băng được sử dụng thường xuyên. Ngoài ra còn có băng bằng nhôm hoặc kim loại.

Băng đục lỗ thường ở dạng cuộn, đôi khi ở dạng các dải gấp chồng lên nhau. Băng có hai công dụng chính:

- Lưu dữ liệu chương trình để sử dụng
- Phương tiện truyền dữ liệu chương trình vào hệ điều khiển thông qua bộ đọc băng

Mã hóa băng

Băng đục lỗ chứa các chuỗi lỗ, xếp theo chiều rộng băng, mỗi hàng lỗ hiển thị một ký

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tự của chương trình - *ký tự là đơn vị nhận nhỏ nhất*. Các ký tự được lỗ được truyền qua bộ đọc băng đến hệ thống điều khiển dưới dạng tín hiệu điện. Từng ký tự có thể gồm đến 8 tín hiệu, hiển thị bằng sự phối hợp đặc thù các lỗ đục qua chiều rộng băng theo số 0.1000 inch (2.54 mm), ký tự có thể là chữ in hoa, chữ số, cùng với các ký hiệu, chẳng hạn dấu thập phân, dấu trừ, dấu vạch, v.v....

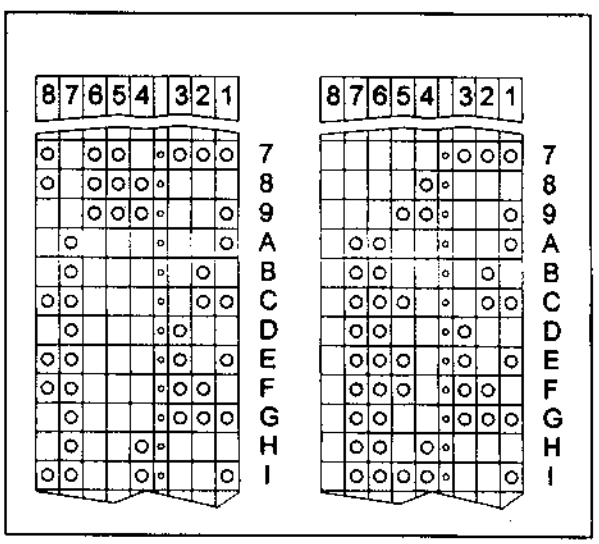
Định dạng băng theo ISO và EIA

Khi chuẩn bị băng, bạn cần biết hai phương pháp mã hóa băng tiêu chuẩn - thứ nhất sử dụng số *chẵn* các lỗ, thứ hai dùng số *lẻ* các lỗ. Thuật ngữ kỹ thuật đối với hai hệ thống này là *Cặp chẵn*, khi ký tự gồm 2, 4, 6, 8 lỗ, và *Cặp lẻ khi ký tự gồm 1, 3, 5, 7 lỗ. Ngoài ra còn có hệ thống mã hóa sử dụng phối hợp cả hai nhưng không áp dụng cho máy công cụ. Hình 47.3 minh họa một phần băng đã mã hóa.*

Cặp chẵn của băng đục lỗ theo tiêu chuẩn quốc tế ISO, cũng được áp dụng cho mã ASCII (*American Standard Code for Information Interchange - Mã tiêu chuẩn Mỹ trao đổi thông tin*). Cặp lẻ là tiêu chuẩn của Electronic Industries Association (Hiệp hội công nghiệp điện tử), viết tắt là EIA, mức độ sử dụng giảm dần, chủ yếu do số lượng ký tự bị giới hạn.

Định dạng cặp chẵn còn được dùng trong các tiêu chuẩn DIN 66024 (ISO), RS-358 (EIA), và mã ISO R-840. Định dạng EIA cặp lẻ có số tiêu chuẩn là RS-244-A.

Hầu hết các điều khiển số hiện đại, nếu có giao diện đọc băng, đều tự động chấp nhận mã hóa băng, dựa trên tính chẵn lẻ của ký tự *thứ nhất kết thúc block* được đục lỗ trên băng đó.



Hình 47.3. Các tiêu chuẩn mã hóa băng. Cặp chẵn (ISO) bên trái, cặp lẻ (EIA) bên phải

Kiểm tra chẵn lẻ

Khi đục lỗ băng, bạn cần bảo đảm quá trình này là nhất quán trên toàn bộ chiều dài băng chương trình. Sự phối hợp các mã ISO và EIA trên một băng bất kỳ sẽ làm cho bộ đọc băng loại bỏ băng đó. Điều này thường được gọi là *lỗi chẵn lẻ*. Bộ điều khiển sẽ tự động kiểm tra tính chẵn lẻ khi băng đục lỗ được vào bộ nhớ CNC hoặc khi được xử lý theo cuộn băng. Bộ điều khiển kiểm tra sự xuất hiện các ký tự *lẻ* trên băng ISO và ký tự *chẵn* trên băng EIA. Mục đích của kiểm tra này là phát hiện lỗi từ thiết bị đục lỗ hoặc đọc băng để tránh các sự cố khi xử lý chương trình.

Điều khiển vào và ra

Trên các băng ISO (định dạng chẵn), cặp mã đục biểu thị *dấu ngoặc đơn* là phần hệ thống điều khiển *không* xử lý. Thông tin bất kỳ trong dấu ngoặc đơn sẽ được bộ điều khiển bỏ qua. Đây là phần chứa các ghi chú chương trình, được ghi trên bản in nhưng sẽ không xử lý khi đọc băng.

Băng trắng

Băng trắng là băng hoàn toàn chưa có lỗ. Nói chung, trên mặt băng thường có các mũi tên để chỉ chiều nạp vào bộ đọc hoặc mặt trên của băng.

Băng trắng cũng có thể là loại chỉ có các lỗ phân chia, nhưng chưa có các lỗ biểu thị các ký tự chương trình. Lỗ phân chia là các lỗ nhỏ giữa kênh thứ ba và thứ tư của băng. Phần trắng của băng được dùng ở đầu và cuối băng để dễ xử lý. Phần trắng còn được dùng để bảo vệ phần mã hóa khi bảo quản băng theo dạng cuộn.

Phần ý nghĩa

Phần băng đục lỗ chứa dữ liệu chương trình thường được gọi là *phần dữ liệu ý nghĩa*. Thuật ngữ được dùng liên quan đến phần dữ liệu ý nghĩa là *hàm bỏ qua nhãn*. Hàm này có nghĩa là mọi thứ cho đến ký tự EOB (kết thúc block) thứ nhất, được đục lỗ trên băng, sẽ bị bỏ qua, nghĩa là phần dữ liệu ý nghĩa của băng là phần sau ký tự EOB thứ nhất.

Lần xảy ra thứ nhất của sự trả về đầu dòng (tương tự nhấn phím Enter trên bàn phía máy tính) là lần xảy ra thứ nhất của ký tự EOB. Tín hiệu này xác định sự bắt đầu phần *dữ liệu ý nghĩa* - phần lưu chương trình thực. Phần dữ liệu ý nghĩa kết thúc bằng mã dừng, thường là dấu phần trăm (%), tác động như ký tự *kết thúc tập tin*. Khi bộ đọc gặp mã dừng, sự đọc băng

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

sẽ hoàn tất. Đó là lý do không ghi thông tin sau dấu phần trăm (%).

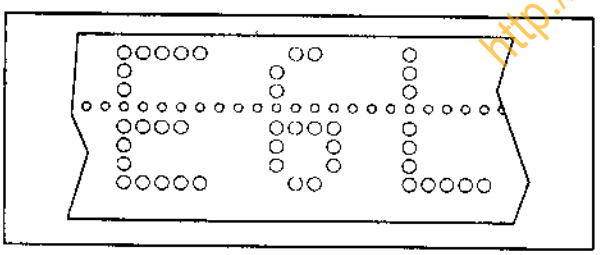
Đầu và cuối băng

Phần *trắng* của băng đục lỗ được sử dụng ở đầu và cuối băng. Phần trắng ở trước dữ liệu chương trình mã hóa (phần dữ liệu chương trình ý nghĩa) được gọi là *đầu băng*, phần tiếp sau phần dữ liệu được gọi là *cuối băng*. Chiều dài thích hợp của phần đầu băng hoặc cuối băng thường khoảng 10 inch (250 mm) đối với chức năng nhớ (không có cuộn), nhưng đến 60 inch (1500 mm) khi băng trên cuộn băng. Đối với các cuộn đường kính nhỏ, các phần đầu và cuối băng có thể ngắn hơn. Đôi khi phải tăng chiều dài phần đầu để ghi ký hiệu băng bằng bút màu.

Nhận biết băng

Từng băng đục lỗ đều phải được nhận biết theo nội dung. Dữ liệu viết tay, các nhãn dán hoặc các ký tự dễ đọc đều có thể được sử dụng trong phần đầu của băng đục lỗ. Các nhãn dán có thể không phải là lựa chọn tốt do có xu hướng bị tróc và bong ra ngoài. Các ghi chú viết tay có thể hơi khó khi viết trên nền đen. Sự nhận biết thường gồm số chương trình hoặc chỉ số băng, số bản vẽ và tên chi tiết, và thông tin cần thiết khác.

Ký tự dễ đọc (Hình 47.4) dường như là giải pháp tốt nhất, do có thể tạo ra các chữ trên hầu hết các trang thiết bị chuẩn bị băng.



Hình 47.4. Ví dụ về ký tự dễ đọc trên băng đục lỗ.

Các ký tự này là các lỗ biểu diễn các ký tự thực, chữ cái, chữ số, và ký hiệu, thay vì các mã số hóa. Ký tự EOB (kết thúc block) hoặc mã dừng có thể không được sử dụng trong phần đầu này, nếu phần đó sẽ đi qua bộ đọc băng.

Các ký tự không in

Hầu hết các ký tự được lưu trên băng đục lỗ sẽ in một cách bình thường. Chúng được gọi là ký tự *in được*, gồm mọi chữ hoa từ A đến Z, chữ số 0 đến 9, và hầu hết các ký hiệu. Mặc dầu các ký tự số và chữ đều in được, nhưng không thể in các ký hiệu dưới đây:

- Mã dừng theo định dạng EIA
- Ký tự xóa
- Trả về đầu dòng (phím Enter)
- Chuyển dòng
- Các mã nhãn

Một ký tự xuất hiện trên màn hình dưới dạng dấu chấm phẩy (;). Đây là ký hiệu đối với ký tự kết thúc block và không được viết. Đây là sự *biểu diễn* của hệ thống điều khiển về sự trả về đầu dòng trong chương trình gia công.

Bảo quản và vận chuyển

Băng giấy được đục lỗ trong bộ đục lỗ băng. Một số các bộ đục lỗ chỉ có tính năng cơ bản, số khác có các tính năng cao cấp, chẳng hạn bàn phím, máy in, bộ đọc băng, các công tắc xác lập, cổng Nhập/Xuất (I/O), v.v...

Băng thường được bảo quản trong hộp nhựa, đủ nhỏ để đặt trong các ngăn tủ được thiết kế đặc biệt. Các băng có thể được chuyển thành tập tin trên máy tính để bảo quản lâu dài.

Nếu vẫn sử dụng băng giấy, cần phải xử lý chúng một cách cẩn thận, đặc biệt khi chúng được cuộn lại hoặc mở các cuộn dài. Để tránh bị hư, không nên quấn băng thành cuộn nhỏ và chặt. Nhiệt, ánh nắng, nước,... là các kẻ thù của băng đục lỗ.

Các băng có thể bị hư hỏng nếu lắp không chuẩn vào bộ đọc băng. Các băng dài đòi hỏi sự chú ý cao hơn so với băng ngắn. Dầu mỡ và bụi bặm là các kẻ thù nguy hiểm của băng giấy. Băng được sử dụng nhiều lần cần có vài bản sao hoặc tập tin trên máy tính.

ĐIỀU KHIỂN SỐ PHÂN PHỐI

Cổng nhập xuất (I/O) của RS-252C trên máy CNC được dùng để gởi và nhận dữ liệu. Các nguồn bên ngoài thường là đĩa cứng hoặc băng giấy. Trong nhiều xưởng, chương trình được truyền thông qua DNC, viết tắt của *Distributed Numenical Control* (điều khiển dữ liệu số phân phối). Bộ điều khiển này có các tính năng để truyền dữ liệu.

Để truyền thông giữa máy CNC và máy tính sử dụng cổng RS-232C, toàn bộ thiết bị cần là cáp nối giữa hai thiết bị và phần mềm. Để truyền thông giữa hai hoặc nhiều máy, sử dụng một cổng RS-232C, từng máy phải được nối cáp với hộp chia. Hộp chia có hai hoặc nhiều cửa ra, có thể lựa chọn bằng công tắc. Đây là dạng DNC đơn giản nhất, đòi hỏi các quy trình được

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

tổ chức tốt để hoạt động một cách hiệu quả. DNC không phải là phần của bộ điều khiển.

Một số phần mềm DNC còn cho phép một tính năng mạnh, được gọi là "truyền từng phần", có thể sử dụng khi chương trình quá lớn so với dung lượng nhớ CNC.

THUẬT NGỮ TRUYỀN THÔNG

Truyền thông có các thuật ngữ riêng, rất phong phú, nhưng phần này chỉ đề cập năm thuật ngữ thông dụng trong CNC:

- Tốc độ baud
- Tính chẵn lẻ
- Các bit dữ liệu
- Bit khởi đầu
- Bit dừng

Tốc độ baud

Tốc độ baud là tốc độ truyền dữ liệu, được đo theo số bit dữ liệu trong một giây, *bps*. Tốc độ baud chỉ khả dụng với các giá trị cố định. Các giá trị đối với bộ điều khiển Fanuc đời cũ gồm 50, 100, 110, 200, 600, 1200, 2400, 4800, và 9600 bps. Các bộ điều khiển hiện đại có tốc độ 2400, 4800, 19200, 38400, 57600, và 76800 bps. Tốc độ càng cao, truyền dữ liệu càng nhanh. Tốc độ truyền bit dữ liệu đơn là kết quả của một (1) chia cho tốc độ baud.

$$S_b = \frac{1}{B}$$

Trong đó:

S_b = Thời gian cần thiết để truyền bit tính theo giây

B = Tốc độ baud tính theo giây.

Một bit được truyền với tốc độ 300 bps sẽ cần 0.03333 giây, tốc độ 24004 bps cần 0.00042 giây. Trong thực tế, cần 10 bit để truyền một ký tự (xem phần Bit dừng), do đó với xác lập 2400 bps, tốc độ truyền sẽ khoảng 240 cps (ký tự /giây). Tốc độ 4800 bps là thông dụng nhất hiện nay.

Tính chẵn lẻ

Tính chẵn lẻ là phương pháp kiểm tra tính chính xác khi truyền dữ liệu. Bạn hãy hình dung điều sẽ xảy ra nếu một số ký tự hoặc chữ số được truyền không chính xác hoặc thậm chí không được truyền. Tính chẵn lẻ có thể là *chẵn*, *lẻ* hoặc *không*, và *chẵn* là lựa chọn thông dụng nhất trong truyền thông CNC.

Các bit dữ liệu

Bit là viết tắt của *Binary digit* (chữ số nhị phân), là đơn vị nhỏ nhất có thể lưu thông tin trong máy tính. Chữ số nhị phân có thể có giá trị một (1) hoặc zero (0). Một và zero biểu diễn trạng thái ON và OFF tương ứng, do đó *bit* thực chất là công tắc hai trạng thái có thể chuyển đổi ON và OFF theo yêu cầu. Trong máy tính, chữ cái, chữ số, ký hiệu dùng trong chương trình CNC được biểu thị bằng chuỗi tám bit, tạo thành đơn vị được gọi là *byte*.

Bit khởi đầu và bit dừng

Để tránh tổn thất dữ liệu trong khi truyền thông, trước mỗi byte đều có một bit đặc biệt được gọi là bit *khởi đầu*, là tín hiệu mức điện áp thấp. Tín hiệu này được gởi đến bộ nhận dữ liệu và thông báo sẽ có byte dữ liệu kế tiếp.

Bit tương tự bit khởi đầu, nhưng ở *cuối* byte, có ý nghĩa ngược lại, gởi tín hiệu đến bộ nhận, cho biết byte đó kết thúc hoặc dừng. Bit ở cuối byte được gọi là bit *dừng*. Do các bit khởi động và dừng đi cùng với nhau, chúng thường được gọi chung là các bit dừng và xác lập các thiết bị theo *hai bit dừng*.

XÁC LẬP DỮ LIỆU

Dữ liệu được dùng để truyền thông phải được xác lập một cách thích hợp trước khi bắt đầu truyền thông. Xác lập tại một đầu (máy tính hoặc hệ thống CNC) phải tương hợp với xác lập ở đầu kia. Đối với tốc độ baud, bạn hãy xem sổ tay hướng dẫn sử dụng máy - giá trị khởi đầu thường là 2400 bps. Các bộ điều khiển mới hơn có mặc định cao hơn. Xác lập phần mềm được thực hiện thông qua cấu hình trên máy tính và các tham số hệ thống ở đầu CNC. Các xác lập ở hai đầu phải tương hợp với nhau. Các xác lập Fanuc phổ biến bao gồm:

- Tốc độ baud 4800 bps
- Cặp chẵn
- 7 bit dữ liệu
- 2 bit dừng

Nối kết phụ thuộc chủ yếu vào cấu hình của các cáp nối dữ liệu.

CÁP NỐI

Cáp phổ biến nhất để truyền thông giữa máy CNC và máy tính là cáp có bảo vệ và có nối mát, chứa nhiều dây nhỏ (ít nhất là 8) từng dây được bao kín trong ống chất dẻo mầu. Cáp truyền thông nối cổng CNC (thường là 25 lỗ

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

cắm) với cổng máy tính (25 chân) sử dụng cấu hình thích hợp. Cáp có bảo vệ thường cho phép nối các khoảng cách xa và cho phép tránh sự giao thoa trong khi truyền dữ liệu. Các dây được ghi rõ giá trị cỡ chuẩn, dây cỡ -22 hoặc cỡ -24 là lựa chọn tốt nhất truyền thông.

Cổng 25 chân có từng chân hoặc lỗ cắm được đánh số và các dây riêng rẽ của các phải được nối vào các số tương ứng giữa hai đầu.

Null Modem

Sự nối cáp rất phổ biến được dùng trong truyền thông tổng quát là *null modem*. Nối kết giữa hai đầu tuân theo tiêu chuẩn xác định, được nêu trên Hình 47.5. Từng số biểu thị chân hoặc lỗ cắm trên đầu nối DB.25. Chú ý có sự nối kết giữa 6 và 8 ở cả hai đầu. Hình 47.6 minh họa cấu hình null modem theo đồ thị. Đây là phương pháp rất phổ biến biểu thị các cấu hình cáp.

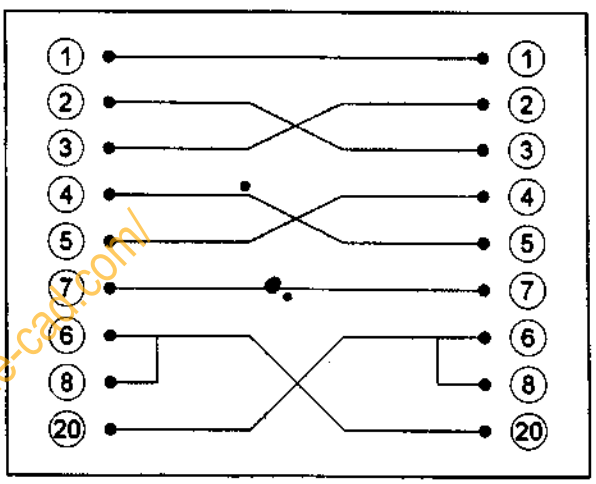
PIN DB-25P	SOCKET DB-25S
1	1
2	3
3	2
4	5
5	4
7	7
6 và 8	20
20	6 và 8

Hình 47.5. Nối kết chân null modem

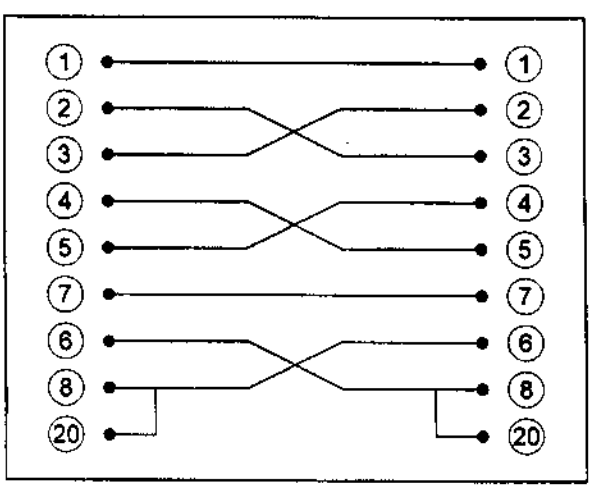
Nối cáp giữa Fanuc và máy tính cá nhân

Cấu hình cáp (Hình 47.7) là phương tiện truyền thông phổ biến giữa bộ điều khiển Fanuc và máy tính xách tay hoặc để bàn. Bạn hãy chú ý sự tương tự với cấu hình null modem.

Bất kể cấu hình cáp được sử dụng, cần có phần mềm truyền thống để xử lý quá trình truyền dữ liệu. Một số công ty sử dụng phần mềm được thiết kế đặc biệt cho hệ thống CNC, số khác mua phần mềm đa dụng trong các cửa hàng máy tính.



Hình 47.6. Biểu diễn đồ thị các nối kết null modem



Hình 47.7. Cấu hình cáp đối với các bộ điều khiển Fanuc.

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

48 CNC và CAD/CAM

Chương này sẽ trình bày lĩnh vực lập trình bằng tay được thay thế bằng máy tính, phần mềm thích hợp, một số kỹ năng bổ sung. Lập trình với máy tính luôn luôn là điều mong muốn, nhưng biết các kỹ năng lập trình cơ bản là điều kiện tiên quyết, hiểu rõ về quá trình lập trình bằng tay, để có thể áp dụng các kỹ thuật lập trình CAD/CAM

Lập trình cao cấp sử dụng phần mềm CAD/CAM yêu cầu kiến thức vững chắc về các phương pháp lập trình bằng tay

LẬP TRÌNH BẰNG TAY?

Trong lĩnh vực các kỹ thuật ứng dụng lập trình CNC, tính ở mọi mức độ, từ máy tính cá nhân đến trạm làm việc, đều có khả năng tạo ra hầu hết các chương trình gia công CNC với thời gian ngắn hơn nhiều so với phương pháp lập trình bằng tay bất kỳ. Tuy nhiên, trong thực tế luôn luôn nhấn mạnh tầm quan trọng của lập trình bằng tay, coi đây là phương pháp chủ yếu và sẽ còn tồn tại lâu dài. Ít nhất có hai lý do quan trọng để giải thích điều đó.

Lý do thứ nhất là trong lập trình bằng tay, nhà lập trình có khả năng thực hiện điều máy tính chưa thể, và có lẽ không bao giờ có thể, đó là *nhà lập trình có thể suy nghĩ*. Lập trình bằng tay dạy các bài học vô giá về tính kỷ luật, đức tính rất quan trọng của nhà lập trình CNC chuyên nghiệp. Kỷ luật có nghĩa là tập trung, thường xuyên đánh giá, đưa ra các quyết định và liên tục suy nghĩ. Trong lập trình bằng tay, luôn luôn có sự kiểm soát tuyệt đối, rõ ràng, và toàn bộ đối với sản phẩm cuối cùng – chương trình gia công. Chỉ nhà lập trình mới có thể đánh giá tình huống, phân tích vấn đề và thích ứng với các trường hợp bất ngờ. Chỉ con người mới có thể sử dụng các công cụ như quá trình tư duy, trí thông minh, trực giác, kinh nghiệm,... lập trình CNC tương tự công việc của người nghệ sĩ, không thể hoàn toàn tự động hóa. Về bản chất, máy tính và phần mềm cũng chỉ là sản phẩm và công cụ của con người.

Phần mềm CAM

Phần mềm CAM hiện hành với nhiều tính năng có thể chuyển tải vào chương trình CNC, phù hợp các ý tưởng về cách thức viết chương trình gia công. Phần mềm có thể tạo ra chương trình gần sát với hướng suy nghĩ, gần sát phong cách lập trình cụ thể, nhưng *gần sát* không có nghĩa là hoàn toàn ăn khớp. Đến đây xuất hiện lý do thứ hai.

Lý do thứ hai là khi lập trình bằng tay, nhà lập trình *hiểu* rõ quá trình lập trình và kết quả xuất. Chương trình do máy tính tạo ra phải theo định dạng tương thích với máy CNC và hệ điều khiển. Nếu chương trình hoạt động tốt, sẽ không có vấn đề gì. Mặt khác, nếu chương trình có vấn đề? Trở lại máy tính và lập trình lại có thể giải quyết được vấn đề đó. Câu hỏi là giá bao nhiêu. Khả năng đọc và hiểu rõ mã chương trình CNC, cũng có nghĩa là khả năng thay đổi, chỉnh sửa chương trình. Tiêu tốn thời gian máy tính chỉ để bổ sung hàm chất làm nguội bị bỏ sót có lẽ là quá lãng phí. Tuy ví dụ này rất đơn giản nhưng cũng cho thấy thực sự *hiểu* quá trình lập trình là rất quan trọng. Cách tốt nhất để hiểu quá trình này là không dùng máy tính nhưng vẫn đạt kết quả, điều đó chỉ có thể đạt được với lập trình bằng tay.

Không nên so sánh giữa hai dạng lập trình máy tính và lập trình bằng tay. Điều cần nhấn mạnh là kiến thức và hiểu biết các nguyên tắc lập trình bằng tay. Không có kiến thức này không thể trở thành nhà lập trình CNC giỏi.

Hầu hết công việc lập trình CNC đều có thể thực hiện trên máy tính cá nhân. Công nghệ hiện nay phát triển rất nhanh, nhiều ứng dụng lập trình 2D và 3D chỉ có giá bằng một phần nhỏ so với giá cả cách đây vài năm. Xu hướng này sẽ còn tiếp tục trong tương lai.

Lập trình trên máy tính để bàn

Hệ thống máy tính hoàn chỉnh – gồm phần cứng, phần mềm, và các thiết vị ngoại vi – thích hợp cho lập trình CNC, phát triển rất nhanh đến mức sự trình bày sâu về phần cứng sẽ trở nên lạc hậu chỉ trong vài tuần, điều này cũng đúng đối với phần mềm. Các tính năng mới, công cụ mới, liên tục xuất hiện trên thị trường. Do phần cứng và phần mềm thường được xét chung với nhau, câu hỏi thứ nhất là chọn phần cứng hay phần mềm trước?

Quyết định phải dựa trên ứng dụng mong muốn. Máy tính được dùng cho công việc gì? Loại công việc nào cần máy tính hóa, tự động

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

hóa? Kết quả? Đó là các xem xét *chính* khi chọn phải là loại màn hình, máy in, hoặc dung lượng đĩa cứng. Chúng cũng rất quan trọng, nhưng chỉ *sau khi* thiết lập các nhu cầu ứng dụng.

Một số ứng dụng lập trình là chung cho mọi xưởng cơ khí, số khác là đặc thù đối với kiểu chế tạo cụ thể và dạng công việc hoặc sản phẩm được chế tạo. Danh mục dưới đây sẽ tóm tắt các nhóm chính có thể thực hiện với hệ thống lập trình CNC dựa trên máy tính:

- ❑ Môi trường tạo ra dạng hình học quỹ đạo dao
- ❑ Tạo ra quỹ đạo dao
- ❑ Môi trường lập trình hoàn chỉnh
- ❑ Xử lý hậu kỳ
- ❑ Hỗ trợ đào tạo và kỹ thuật

Điều quan trọng là hiểu *tầm quan trọng* của các tính năng đó. Ngoài ra, bạn cần biết *các công cụ* của phần mềm và *phương pháp* sử dụng chúng trong lập trình hàng ngày.

PHÁT TRIỂN HÌNH HỌC QUỸ ĐẠO DAO

Hầu hết các hệ thống lập trình CNC đều đòi hỏi tạo ra dạng hình học quỹ đạo dao trước khi có thể thiết kế quỹ đạo thực của dụng cụ cắt. Điều cốt lõi ở đây là *dạng hình học quỹ đạo dao*. Một số nhà lập trình hiểu điều này là tái tạo mọi thứ trong bản vẽ gốc, nhưng điều đó là hoàn toàn sai.

Bạn sẽ gặp hai tình huống khi giải quyết dạng hình học quỹ đạo dao. Thứ nhất là dạng công việc trên bản vẽ giấy, thứ hai là từ bản vẽ CAD lưu trong máy tính. Tuy có các khác biệt, nhưng điều cơ bản là tạo ra dạng hình học mới hoặc chỉnh sửa dạng hình học hiện có. Các hệ thống CAD hiện đại cho phép vẽ trong CAD – sử dụng các tính năng biên tập, chẳng hạn cắt, xén, sao chép, quay với đối xứng gương, lập tỷ lệ...

Nói chung, nhà lập trình sẽ xác định những điều thường chưa có trên bản vẽ, ít nhất là trên bản vẽ hai chiều. Bổ sung chiều sâu, phân biệt các vật thể bằng màu sắc, chia các lớp, bổ sung khoảng hở, chuyển động dao vào và ra, v.v...

TẠO QUỸ ĐẠO DAO

Yêu cầu cơ bản của phần mềm CNC là tạo ra chương trình với quỹ đạo dao chính xác đối với máy CNC cụ thể. Sự tạo quỹ đạo dao, với mọi tính toán liên quan, là công việc tốn nhiều thời gian nhất trong lập trình bằng tay. Do đó đây cũng là vấn đề quan trọng nhất cần xem xét khi lập kế hoạch tự động hóa quá trình lập trình CNC. Chỉ phần mềm CNC cao cấp mới có thể hỗ trợ nhiều dạng quỹ đạo dao. Ví dụ, phay xoắn hoặc gia công 3D toàn phần không phải lúc nào cũng là tính năng tiêu chuẩn trong phần mềm.

Một nhầm lẫn trong lựa chọn phần mềm là chỉ chú ý các máy CNC *hiện hữu* và các phương pháp gia công hiện có. Phương pháp này không phải lúc nào cũng thành công. Bạn hãy khảo sát các kế hoạch tương lai theo cả hai phần (máy và phương pháp gia công) và vốn đầu tư. Chủng loại sản phẩm, sự thay đổi sản phẩm trong 5 năm tới, hướng phát triển của thị trường ...

Công nghệ máy tính phát triển rất nhanh, không ai có thể dự đoán chính xác các tính năng tương lai của lập trình và gia công CNC. Nếu xác định đúng các nhu cầu hiện tại và tương lai *trước khi* đặt mua hệ thống lập trình, bạn sẽ có cơ hội tránh được sự lạc hậu trong thời gian dài. Các nhà viết phần mềm CNC thường xuyên cập nhật sản phẩm của họ, với nhiều tính năng bổ sung khi năng lực tính toán tăng lên. Các cập nhật (phiên bản mới của phần mềm), thường phản ánh sự phát triển của công nghệ cả phần cứng và phần mềm.

MÔI TRƯỜNG HOÀN CHỈNH

Phần mềm lập trình CNC chất lượng cao cho phép lập trình và mọi nhiệm vụ liên quan đều được thực hiện từ menu cấu trúc, sử dụng mouse hoặc thiết bị trỏ tương tự. Điều quan trọng là sau khi tải phần mềm, có thể hoàn tất mọi tác vụ vừa không cần trở về mức hệ điều hành. Một số hệ thống lập trình dựa trên các module và tập tin không thể truy cập từ menu, hoặc không bao quát mọi bước lập trình.

Danh mục dưới đây chỉ là hướng dẫn tóm tắt một số tính năng chính áp dụng cho lập trình CNC trên máy tính cá nhân. Đây là các tính năng chung từ phần mềm CAM bất kỳ:

- ❑ Hỗ trợ nhiều máy (trung tâm gia công, máy tiện, EDM)
- ❑ Các thao tác liên quan để biên tập linh hoạt.
- ❑ Gá lắp và xác định phôi gia công
- ❑ Danh mục dụng cụ và gá lắp
- ❑ Nối kết giữa các máy tính (tính năng truyền thông)
- ❑ Bộ biên tập văn bản chương trình (với các tính năng phù hợp với CNC)
- ❑ Khả năng in (văn bản và đồ họa)
- ❑ Bút vẽ (máy vẽ)
- ❑ Giao diện với phần mềm CAD (DXF, IGES, CADL, STL)
- ❑ Hỗ trợ lập mô hình khối

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Các tính năng và đặc tính kỹ thuật phần mềm (kể cả xử lý hậu kỳ chuyên biệt hóa)
- Hỗ trợ phần cứng khả dụng
- Các tiện ích và tính năng đặc biệt, cấu trúc mở

Tuy các tính năng nêu trên đều rất hữu dụng trong hệ thống lập trình, nhưng không có nghĩa là tất cả đều luôn luôn cần thiết. Một số tính năng đòi hỏi thêm thiết bị phần cứng, chẳng hạn máy in, máy vẽ, nối cáp, các thiết bị ngoại vi nhỏ, v.v..

Hỗ trợ nhiều máy

Khi hỗ trợ nhiều kiểu máy khác nhau, phần mềm CNC có thể chia thành hai nhóm

- Phần mềm chuyên dùng
- Phần mềm tích hợp

Phần mềm *chuyên dùng* chỉ hỗ trợ một loại máy. Ví dụ, phần mềm được thiết kế đặc biệt để tạo ra các chương trình đối với trang thiết bị chế tạo CNC, không thể dùng cho máy tiện, trung tâm gia công,...

Phần mềm chuyên dùng thường được viết cho phạm vi ứng dụng hẹp, chuyên sâu, hoặc chỉ áp dụng cho máy cụ thể, chẳng hạn máy đột dập CNC, máy dập khuôn, ...

Phần mềm *tích hợp* cho phép nhà lập trình lựa chọn vài kiểu máy công cụ. Sự lựa chọn thường gồm phay, tiện, và EDM (gia công bằng tia lửa điện) Nhóm lựa chọn thứ hai gồm máy cắt laser, cắt bằng tia nước, cắt biên dạng, ...

Phần mềm tích hợp có *giao diện* linh hoạt, có thể sử dụng màn hình để hiển thị các nguyên công tiện, phay, hoặc EDM. Các menu phần mềm có hình thức như nhau, các thao tác tìm kiếm có cùng các mục menu, sự chuyên biệt hóa phần mềm (kể cả xử lý hậu kỳ) đơn giản hơn.

Các nguyên công liên quan

Khi khai triển quỹ đạo dao, quỹ đạo đó sẽ gắn chặt với dạng hình học quỹ đạo dao đã xác định trước. Vì nhiều lý do, dạng hình học quỹ đạo dao có thể thay đổi. Phương pháp truyền thống (nhiều công ty phần mềm vẫn còn áp dụng) là tái tạo dạng hình học, sau đó tái tạo quỹ đạo dao.

Nguyên công liên quan tránh việc tạo ra quỹ đạo dao mới, chỉ cập nhật một cách tự động, do đó nhanh và chính xác hơn. Nhiều tham số dao có thể thay đổi hoặc cập nhật nhanh chóng theo yêu cầu.

Gá lắp

Gá lắp là quá trình xác định phôi gia công chi tiết, hình dạng, kích thước, chuẩn định vị và các yếu tố liên quan. Dao, chế độ cắt, tốc độ, lượng ăn dao,... và các tham số chương trình cũng được lựa chọn trong khi gá lắp. Thư mục lưu dữ liệu về dụng cụ cắt, vật liệu và các nguyên công là tính năng chuẩn của phần mềm CNC.

Danh sách dao và các chú thích gia công

Lập trình CNC là quy trình gồm nhiều bước. Dù lập trình bằng tay hay với máy tính, vẫn phải chọn dụng cụ cắt một cách thủ công. Sau khi chọn, từng dao được gán ký hiệu, tốc độ và lượng ăn dao. Nhiều dao có thể được chia nhóm trong tập tin thư mục dụng cụ cắt và lưu vào ổ cứng. Sau đó sẽ chọn thứ tự sử dụng dao trong chương trình. Một số chi tiết đòi hỏi vài nguyên công cắt gọt. Các gá lắp phức tạp đòi hỏi sự hướng dẫn hoặc giải thích cho người vận hành máy, nêu rõ các ý đồ của nhà lập trình. Mọi quyết định lập trình phải được ghi lại và tài liệu phải gởi đến xưởng cơ khí. Yêu cầu cơ bản là phần mềm lập trình CNC bất kỳ phải hỗ trợ danh mục dụng cụ cắt, ở dạng tập tin thư mục dao và danh mục xử lý. Tập tin thư mục vật liệu cũng rất hữu ích, có thể gồm cơ sở dữ liệu về tốc độ bề mặt đối với nhiều vật liệu, phần mềm lập trình sẽ tính toán chính xác tốc độ trục chính và tốc độ cắt, dựa trên dao được chọn, bảo đảm sự tương tác giữa thư mục dụng cụ cắt và thư mục vật liệu.

Nối kết giữa các máy tính

Hệ thống lập trình phải có nối kết (tùy chọn truyền thông) giữa máy tính cá nhân và máy CNC. Tính năng này cho phép trao đổi dữ liệu chương trình thông qua cáp nối. Các chương trình có thể gởi từ máy tính đến bộ nhớ trên máy CNC và ngược lại.

Điều quan trọng là không phải mọi máy CNC đều có cổng (kết nối) và khả năng tận dụng ưu thế của kết nối trực tiếp. Ngay cả khi tất cả các máy trong xưởng đều có khả năng này, vẫn cần có phần cứng bổ sung và tính kỹ luật để bảo đảm mọi bộ phận hoạt động hài hòa và đồng bộ với nhau. Phần mềm lập trình phải có tính năng nối kết trực tiếp để bảo đảm sự truyền thông theo yêu cầu.

Bộ biên tập văn bản chương trình

Chương trình CNC được tạo ra bằng phần mềm phải hoàn chỉnh 100% và sẵn sàng cho

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

máy sẽ không thể biết có nghĩa là chương trình đó phải hoàn hảo, không cần biên tập thêm, đây là điều lý tưởng. Nếu cần thay đổi trong chương trình, nên thực hiện *trong phạm vi* thiết kế hình dạng chi tiết và thông qua phần mềm CNC – *không phải bên ngoài phần mềm*. Lý do là sự thay đổi bằng tay bất kỳ đối với chương trình đã tạo ra sẽ không tương ứng dữ liệu chương trình như khi được tạo ra bằng máy tính. Trong môi trường dữ liệu được chia sẻ cho nhiều người dùng, phương pháp đó sẽ gây ra nhiều vấn đề.

Điều này đưa đến câu hỏi, tại sao phần mềm CNC phải có bộ biên tập văn bản được thiết lập bên trong? Có hai lý do. Thứ nhất, bộ biên tập có thể được dùng để viết hoặc chỉnh sửa các tập tin văn bản, chẳng hạn các bản gá lắp, dữ liệu vận hành, mẫu xử lý hậu kỳ, tập tin cấu hình, các hướng dẫn đặc biệt, ... Các tập tin đó có thể cập nhật và chỉnh sửa theo yêu cầu, *mà không* ảnh hưởng đến cơ sở dữ liệu chương trình. Thứ hai, trong một số trường hợp *đặc biệt*, chương trình CNC có thể được biên tập bên ngoài máy tính, nhưng thay đổi đó phải không tác động đến dữ liệu. Ví dụ, bổ sung hàm *chất làm nguội* M08 bị thiếu vào chương trình được thực hiện trong bộ biên tập văn bản sẽ nhanh hơn so với sự lặp lại quá trình tạo chương trình máy tính. Tuy đây không phải là phương pháp chuẩn để sử dụng bộ biên tập văn bản nhưng ít nhất các dữ liệu quan trọng (vị trí dụng cụ cắt) không bị ảnh hưởng và cơ sở dữ liệu vẫn chính xác.

Nhiều nhà lập trình sử dụng các bộ biên tập văn bản bên ngoài kể cả bộ xử lý từ ngữ trong chế độ văn bản. Các kiểu bộ biên tập này không được định hướng cho lập trình CNC, do chúng thiếu một số tính năng cần thiết để triển khai chương trình CNC. Chỉ các bộ biên tập văn bản định hướng CNC mới có thể xử lý tự động chuỗi số thứ tự block, loại bỏ các số block, bổ sung các khoảng trắng thẩm mỹ trong chương trình và các chức năng khác. Bộ biên tập này phải được truy cập từ menu chính hoặc bên trong phần mềm.

Khả năng in ấn

Văn bản bất kỳ lưu theo tập tin (file), kể cả các chương trình CNC, đều có thể in sử dụng máy in tiêu chuẩn. Bản in trên giấy là tài liệu tham khảo cần thiết của người vận hành, lưu giữ tài liệu, hoặc chỉ để thuận tiện. Máy in chỉ cần in được khổ giấy A4. Một số phần mềm lập trình hỗ trợ tùy chọn chuyển hình ảnh đồ họa từ màn hình sang máy in với chất lượng cao. Hình ảnh đồ họa này rất có ích trong giai đoạn viết chương trình. Sự hỗ trợ máy in được môi trường Windows cung cấp, hầu như mọi phần mềm CAM dựa trên máy tính cá nhân (PC) đều có thể hoạt động với hệ điều hành Windows.

Máy vẽ đồ họa

Máy vẽ đồ họa thường có chất lượng hình ảnh cao hơn máy in nhưng đối với lập trình CAM điều này chưa hẳn là cần thiết, chỉ dùng khi yêu cầu khổ giấy lớn hơn máy in tiêu chuẩn, các bản vẽ màu,... để đáp ứng các yêu cầu đặc biệt khách hàng. Trước khi phần mềm đồ họa xuất hiện trên thị trường, máy vẽ được sử dụng rộng rãi để kiểm tra quỹ đạo dao. Ngày nay, quỹ đạo dao được kiểm tra trực tiếp trên màn hình máy tính, trong quá trình lập trình tương tác, gồm cả các hình chiếu và độ phóng đại khác nhau.

Hầu hết các máy vẽ đều tương thích với HPGL (viết tắt của *Hewlett Packard Graphics Language* – Ngôn ngữ đồ họa Hewlett - Packard), định dạng trao đổi tập tin đồ họa thông dụng hiện nay.

Truy cập phần mềm CAD

Nếu bản vẽ kỹ thuật được vẽ bằng phần mềm CAD, toàn bộ thông tin bản vẽ được lưu trong cơ sở dữ liệu máy tính. Cơ sở dữ liệu này có thể được truy cập bằng nhiều gói phần mềm lập trình, thông qua tiện ích diễn dịch dạng tập tin. Khi phần mềm CNC tiếp nhận và xử lý dữ liệu từ hệ thống CAD, nhà lập trình có thể tập trung thiết kế quỹ đạo dao, thay vì xác định dạng hình học quỹ đạo dao từ đầu. Ưu điểm lớn nhất của phần mềm CAD/CAM chất lượng là tránh sự sao đúp. Không có hệ thống CAD, nhà lập trình có thêm việc phải làm, hầu hết là sao đúp.

Phần mềm CNC chất lượng cao còn cho phép tập tin chương trình hiện hữu chuyển thành tập tin hệ thống CAD có thể chấp nhận. Tùy chọn này được gọi là *xử lý ngược*, rất có ích khi cần chuyển các chương trình hiện hữu được viết bằng tay thành dạng điện tử.

Phần mềm CNC cao cấp là loại *độc lập*, nghĩa là không cần truy cập hệ thống CAD – dạng hình học quỹ đạo dao và quỹ đạo dao có thể thiết kế ngay trong phần mềm CAM, độc lập với phần mềm khác.

Hỗ trợ hình khối

Lập mô hình khối cho các ứng dụng 3D từ lâu đã là lĩnh vực của các hệ thống máy tính lớn. Với sự phát triển của máy vi tính mạnh,

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

lập mô hình khối trở thành một phần trong phần mềm CNC cao cấp.

Với các mô hình khối, quá trình gia công các bề mặt phức tạp trở nên trôi chảy hơn. Ngoài ra, mô hình khối còn giúp cung cấp dữ liệu kỹ thuật, quan sát, xử lý các vật thể dễ dàng hơn, và nhiều tính năng khác.

Đặc tính kỹ thuật phần mềm

Phần mềm CNC cao cấp cung cấp nhiều tính năng rất hữu ích. Điều làm cho từng hệ thống có sắc thái riêng, thường là phương pháp thực thi quá trình lập trình. Trong thời kỳ đầu, lập trình được thực hiện bằng cách dùng các ngôn ngữ lập trình đặc biệt, chẳng hạn APT™ hoặc Compact II™. Tuy vài ngôn ngữ vẫn còn khả dụng nhưng ngày càng ít sử dụng. Lập trình đồ họa tương tác hiện đại hầu như loại bỏ nhu cầu về ngôn ngữ trong hầu hết các lĩnh vực chế tạo. Dạng lập trình chủ yếu hiện nay dựa trên *tương tác đồ họa*. Nhà lập trình xác định các dạng hình học, thường là dạng hình học quỹ đạo dao. Lỗi bất kỳ trong quá trình này sẽ thể hiện hầu như tức thời trên màn hình đồ họa và có thể chỉnh sửa trước khi tiếp tục công việc.

Đặc tính kỹ thuật phần cứng

Đặc tính kỹ thuật phần mềm sẽ xác định sự lựa chọn phần cứng. Phần cứng là thuật ngữ chung để biểu thị máy tính, màn hình, bàn phím, máy in, modem, máy vẽ, mouse, máy quét, ổ đĩa... Phần cứng trong chương này dựa trên hệ điều hành Windows™. Các hệ điều hành hiện đại dựa trên *giao diện người dùng đồ họa* (GUI). Nói chung, nên sử dụng phiên bản hệ điều hành mới nhất, chẳng hạn Windows XP và phần mềm CAM trên máy tính

Khi xét phần cứng máy tính, bạn cần biết ít nhất ba tiêu chuẩn chính:

- ❑ Hiệu suất ... tốc độ máy tính
- ❑ Lưu dữ liệu ... kiểu loại và dung lượng
- ❑ Nhập / xuất ... các cổng

Tốc độ máy tính

Hiệu suất của hệ thống máy tính thường được đo bằng tốc độ tương đối của bộ xử lý chính. Tốc độ này càng cao, máy tính xử lý dữ liệu càng nhanh. Đối với CAD/CAM, bạn nên sử dụng bộ xử lý mới nhất, tốc độ xử lý càng lớn, hiệu suất lập trình CNC càng cao.

RAM và lưu dữ liệu

Khi khởi động ứng dụng, chẳng hạn lập trình CNC, phần mềm CAM được tải vào bộ nhớ máy tính. Phần mềm ứng dụng càng mạnh, đòi hỏi dung lượng nhớ càng cao. Mỗi phần mềm đều đưa ra yêu cầu RAM tối thiểu. RAM trên máy tính ngày nay có thể có dung lượng vào cỡ GB. Nói chung, tối thiểu là 512 MB. Để lưu dữ liệu từ RAM vào các tập tin đĩa, bạn cần có ổ cứng, và các phương tiện nhớ chẳng hạn CD-RW, DVD-R,... riêng đĩa mềm thường không thích hợp, do dung lượng nhớ quá nhỏ.

Nhập và xuất

Các tính năng máy tính *nhập* và *xuất* (I/O) được thực hiện thông qua màn hình, thẻ mạch đồ họa, bàn phím, bộ số hóa, máy in, máy vẽ, ... Màn hình thích hợp cho CAD / CAM là loại màn hình màu cỡ lớn với độ phân giải rất cao và thẻ mạch đồ họa tương ứng. Tốc độ xuất video cũng rất quan trọng.

Mọi thiết bị ngoại vi đều có giao diện với máy tính sử dụng các cáp cấu hình đặc biệt nối với các *cổng* nhập/xuất (I/O). Modem, có lẽ là không cần thiết đối với lập trình CNC, ngoại trừ trao đổi dữ liệu với máy tính xa hoặc truy cập Internet. Máy in phun mực hoặc in laser thường sử dụng cổng song song, nhưng nhiều thiết bị khác sử dụng cổng nối tiếp. Ngoài ra, còn có các tùy chọn I/O khác, chẳng hạn cổng USB.

http://www.me-cad.com/

Yêu cầu phần cứng

Hiện nay, phần cứng phổ biến đối với lập trình CNC là hệ thống máy tính dựa trên Windows. Dưới đây là một số quy tắc có thể áp dụng cho hệ thống bất kỳ, có lẽ chưa bị lạc hậu trong tương lai gần.

- ❑ Phần cứng tương thích với IBM – không phải Macintosh
- ❑ Phiên bản hệ điều hành mới nhất (phải hỗ trợ phần mềm CAM)
- ❑ Tốc độ xử lý cao nhất có thể (GHz)
- ❑ Cache nhớ nhanh (L2)
- ❑ Bộ đồng xử lý số (còn gọi là đồng xử lý toán–thường được thiết lập sẵn.)
- ❑ RAM với dung lượng nhớ cao (tối thiểu là 512MB)
- ❑ Đủ dung lượng ổ cứng để lưu dữ liệu và chương trình (trên 20GB)
- ❑ Hệ thống dự phòng để bảo vệ dữ liệu (ổ đĩa tháo được, ổ CD-RW, DVD)
- ❑ Thẻ mạch đồ họa có độ phân giải cao
- ❑ Màn hình màu cỡ lớn (không dưới 19 inch) có độ phân giải cao
- ❑ Thiết bị trỏ
- ❑ Máy vẽ, nếu thực sự cần thiết

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

- Đồng hồ lịch thời gian thực (ghi rõ thời gian cho mọi tập tin)
- Máy in chất lượng cao với cổng song song hoặc USB
- CD-ROM, DVD và các tính năng đa phương tiện (kể cả thẻ mạch âm thanh)
- Modem
- Ít nhất hai cổng nối tiếp.

Các trình tiện ích và tính năng đặc biệt

Ngay cả phiên bản hệ điều hành mới nhất cũng không đủ mạnh và linh hoạt như người dùng mong muốn. Vì lý do đó, nhiều công ty phần mềm cung cấp hàng ngàn trình tiện ích bổ sung cho các tính năng bên trong máy tính. Nhiều trình tiện ích này có thể tải miễn phí từ Internet. Truy cập Internet và World Wide Web cung cấp nguồn CNC rất lớn, thông tin và nhiều vấn đề khác liên quan đến xưởng cơ khí.

XỬ LÝ HẬU KỲ

Phần mềm CNC phải có khả năng tạo ra chương trình CNC theo định dạng riêng cho từng bộ điều khiển. Phần quan trọng nhất để tạo ra quỹ đạo dụng cụ cắt là tính toàn vẹn dữ liệu. Chương trình do máy tính tạo ra phải chính xác và sẵn sàng cho máy CNC, nghĩa là chương trình hoàn hảo không yêu cầu biên tập, tối ưu hóa... Mục đích này chỉ có thể đạt được bằng phong cách lập trình tốt và xử lý hậu kỳ cho từng máy CNC.

Xử lý hậu kỳ chất lượng có lẽ là tính năng quan trọng nhất của phần mềm CNC. Khi nhập dữ liệu vào phần mềm này, chủ yếu là nhập các giá trị về hình dạng chi tiết, chế độ cắt, tốc độ trục chính... Phần mềm phân tích dữ liệu, phân loại chúng và tạo ra cơ sở dữ liệu. Cơ sở dữ liệu đó biểu thị dạng hình học chi tiết, chuyển động dao và các đặc tính khác. Hệ thống CNC không thể *hiểu* dữ liệu, bất kể độ chính xác của chúng. Hệ thống CNC còn yêu cầu các ký tự, các ký hiệu theo định dạng riêng. Một số mã chương trình là đặc thù đối với máy, số khác là chung cho nhiều loại máy. Mục đích của xử lý hậu kỳ là sử dụng cơ sở dữ liệu từ các giá trị số, dựa trên sự nhập của nhà lập trình, chuyển chúng thành mã máy cho từng hệ điều khiển.

Chuyên biệt hóa xử lý hậu kỳ

Bộ xử lý hậu kỳ ít nhiều đều có tính tổng quát và cần chuyên biệt hóa. Quá trình này tùy thuộc vào kiểu của bộ xử lý hậu kỳ và định dạng của bộ đó.

CÁC TÍNH NĂNG QUAN TRỌNG

Có một số tính năng ảnh hưởng đến chất lượng của chương trình ở mức độ máy CNC, chúng rất quan trọng và cần xem xét cẩn thận.

Nhập từ người dùng

Một trong các tính năng quan trọng của phần mềm lập trình CAM là khả năng xử lý phần nhập từ người dùng. Phần nhập này có thể là chuỗi thứ tự các lệnh không chỉ xử lý bằng bộ xử lý hậu kỳ, hoặc đòi hỏi nhiều nỗ lực. Các lệnh này thường chỉ gồm vài ký tự, có thể được gọi lên và sử dụng trong chế độ đồ họa mỗi khi có yêu cầu. Các ứng dụng đó có thể là chuỗi thứ tự cấp phôi thanh trên máy tiện hoặc thay pallet trên trung tâm gia công ngang. Nếu phần mềm hỗ trợ kiểu lệnh người dùng, sẽ tăng thêm tính linh hoạt và năng lực cho hệ thống.

Các chu kỳ gia công

Đặc tính rất quan trọng của phần mềm CAM là khả năng tạo ra các chu kỳ cố định hoặc lặp lại, hỗ trợ từ các bộ điều khiển hiện đại. Các chu kỳ này giúp cho sự lập trình bằng tay trở nên nhanh hơn và dễ dàng hơn. Hệ thống CNC hiện đại tận dụng ưu thế của các chu kỳ đó nhưng chỉ có dung lượng nhớ hạn chế. Vì lý do đó, sự hỗ trợ tính năng này là rất quan trọng trong phần mềm CNC, cho phép dễ dàng biên tập ngay tại máy.

Giao diện người dùng

Chuyên biệt hóa sự hiển thị cũng là tính năng rất hữu ích. Tuy không quá quan trọng như các tính năng khác, nhưng khả năng chuyên biệt hóa kiểu chữ, màu sắc, thanh công cụ, kể cả các menu, sẽ tăng thêm sức mạnh cho phần mềm này. Màu sắc là rất quan trọng trong CAD/CAM. Các xác lập màu sắc phải dễ thay đổi để có sự phân biệt tốt hơn. Dáng vẻ màn hình cũng có thể thay đổi bằng sự phối hợp các màu sắc phông nền, cận cảnh, và văn bản. Kết quả sẽ là sự nhấn mạnh thị giác vào những phần quan trọng.

Tính năng giao diện người dùng cuối cùng là sự lựa chọn các tùy chọn kiểm tra trong phần mềm. Khi sự mô phỏng quỹ đạo dao được nêu trên màn hình, vòng tròn biểu thị đường kính dao trong các ứng dụng phay và hình dạng dao tiện trên máy tiện CNC. Hình ảnh dao này biểu thị vị trí dao hiện hành, có giá trị đối với phần chương trình đang được xử lý. Nói chung, hình ảnh đó dịch chuyển theo biên dạng hoàn toàn không để lại các dấu vết. Một biến

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

thể là dao dừng ở các điểm thay đổi biên dạng. Điều này được gọi là hiển thị tĩnh, rất quan trọng đối với một số nguyên công cắt gọt. Phần mềm CAM cao cấp còn cho phép thiết kế hình dạng dao chuyên biệt, kể cả cán dao, và sử dụng trên màn hình để mô phỏng quỹ đạo dao thực. Các dao 3D làm tăng thêm tính hiện thực cho sự mô phỏng chương trình.

Điều quan trọng là biểu thị quỹ đạo các dao tiện. Nhiều dao trên máy tiện CNC có góc thoát sau. Phần mềm chất lượng cao có thể đánh giá góc thoát sau trên dao cắt trong các tính toán và trên màn hình.

Giao diện CAD

Hệ thống lập trình CNC độc lập không cần phần mềm CAD để xác định các dạng hình học, hệ thống CNC có thể tạo ra dạng hình học riêng. Trong hệ thống CAD/CAM bất kỳ điều quan trọng là có tùy chọn nhập dạng hình học của chi tiết từ hệ thống CAD.

Nếu không có phần mềm CAD, máy tính không thể tiếp nhận các tập tin bản vẽ do phần mềm đó tạo ra. Các tập tin này có tính độc quyền, với cấu trúc chỉ có thể truy cập một cách đặc biệt. Do đó, phải có cách khác để trao đổi các tập tin bản vẽ, đó là tính năng *sử dụng định dạng tập tin khác*.

Các định dạng trao đổi tập tin

Nhu cầu trao đổi tập tin thiết kế giữa các phần mềm khác nhau luôn luôn là yêu cầu hàng đầu. Hiện có nhiều định dạng tập tin. Cũ nhất trong số đó là IGES (*Initial Graphics Exchange Specification* – Đặc tính kỹ thuật trao đổi đồ họa ban đầu), được thiết kế để chuyển các tập tin thiết kế phức tạp từ phần mềm này sang phần mềm khác. Định dạng kế tiếp là DXF của Autodeck™.

DXF (*Drawing eXchange Format* - định dạng trao đổi bản vẽ, hoặc *Data eXchange Format* - định dạng trao đổi dữ liệu) được nhiều người đánh giá là tiêu chuẩn trao đổi tập tin bản vẽ giữa các máy tính. Định dạng này là của Autodesk™, cũng là tác giả của AutoCAD™, CAD dựa trên máy tính được sử dụng rộng rãi trên thế giới. Định dạng DXF chỉ thích hợp với các dạng hình học cơ bản, chẳng hạn điểm, đường, cung, v.v...

KẾT LUẬN

Rất khó dự báo tương lai của công nghệ CNC. Có nhiều dấu hiệu cho thấy công nghệ này sẽ tiếp tục phát triển. Bộ điều khiển sẽ có năng lực tính toán cao hơn, tăng cường tiêu chuẩn hóa phương pháp lập trình, lập mô hình 3D phức tạp hơn, dung lượng nhớ cao hơn,...

Các máy CNC độc lập vẫn rất cần thiết. Trên các trung tâm gia công sẽ tập trung tăng tốc độ gia công. Trên các máy tiện CNC, sự phát triển sẽ là sử dụng kỹ thuật phân độ dụng cụ cắt, cho phép sử dụng nhiều dao hơn và giữa các dao chưa dùng cách xa khu vực gia công.

Dự báo về máy tính có thể khó khăn hơn, ngoại trừ năng lực của máy sẽ tăng rõ rệt. Phần cứng dường như phát triển nhanh hơn phần mềm. Phần mềm CNC cũng không phải là ngoại lệ. Người chiến thắng trong cuộc cạnh tranh sẽ là người có thể kết hợp phần cứng, phần mềm và người dùng, sản phẩm dễ sử dụng và có giá cả hợp lý. Kinh tế bảo hộ sẽ ngày càng suy giảm, thương mại sẽ là một phần trong nền kinh tế toàn cầu. Điều quan trọng nhất là "*học, ứng dụng trong công việc, và tiếp tục học*"

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Đương lượng thập phân

Inch thập phân	Phân số	Số / Mẫu tự	Hệ mét (mm)
.0059		97	0.15
.0063		96	0.16
.0067		95	0.17
.0071		94	0.18
.0075		93	0.19
.0079		92	0.20
.0083		91	0.21
.0087		90	0.22
.0091		89	0.23
.0095		88	0.24
.0100		87	0.25
.0105		86	
.0110		85	0.28
.0115		84	
.0118			0.30
.0120		83	
.0125		82	
.0126			0.32
.0130		81	
.0135		80	
.0138			0.35
.0145		79	
.0150			0.38
.0158	1/64		
.0157			0.40
.0160		78	
.0177			0.45
.0180		77	
.0197			0.50
.0200		76	
.0210		75	
.0217			0.55
.0225		74	
.0236			0.60
.0240		73	
.0250		72	
.0256			0.65
.0260		71	
.0276			0.70
.0280		70	
.0292		69	
.0295			0.75
.0310		68	
.0313	1/32		
.0315			0.80
.0320		67	
.0330		66	
.0335			0.85
.0350		65	
.0354			0.90
.0360		64	
.0370		63	
.0374			0.95
.0380		62	
.0390		61	
.0394			1.00
.0400		60	
.0410		59	
.0413			1.05
.0420		58	
.0430		57	
.0433			1.10
.0453			1.15
.0465		56	
.0469	3/64		
.0472			1.20
.0492			1.25
.0512			1.30
.0520		55	
.0531			1.35
.0550		54	
.0551			1.40
.0571			1.45
.0591			1.50
.0595		53	
.0610			1.55
.0625	1/16		
.0630			1.60
.0635		52	
.0650			1.65
.0669			1.70
.0670		51	
.0689			1.75
.0700		50	
.0709			1.80
.0728			1.85
.0730		49	
.0748			1.90
.0760		48	
.0768			1.95
.0781	5/64		
.0785		47	
.0787			2.00
.0807			2.05
.0810		46	
.0820		45	
.0827			2.10
.0846			2.15
.0860		44	
.0866			2.20
.0886			2.25
.0890		43	
.0906			2.30
.0925			2.35
.0935		42	
.0938	3/32		
.0945			2.40
.0960		41	

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Inch thập phân	Phân số	Số / Mẫu tự	Hệ mét (mm)
.0965			2.45
.0980		40	
.0984			2.50
.0995		39	
.1015		38	
.1024			2.60
.1040		37	
.1063			2.70
.1065		36	
.1083			2.75
.1094	7/64		
.1100		35	
.1102			2.80
.1110		34	
.1130		33	
.1142			2.90
.1160		32	
.1181			3.00
.1200		31	
.1220			3.10
.1250	1/8		
.1260			3.20
.1280			3.25
.1285		30	
.1299			3.30
.1339			3.40
.1360		29	
.1378			3.50
.1405		28	
.1406	9/64		
.1417			3.60
.1440		27	
.1457			3.70
.1470		26	
.1476			3.75
.1495		25	
.1496			3.80
.1520		24	
.1535			3.90
.1540		23	
.1562	5/32		
.1570		22	
.1575			4.00
.1590		21	
.1610		20	
.1614			4.10
.1654			4.20
.1660		19	
.1673			4.25
.1693			4.30
.1695		18	
.1719	11/64		
.1730		17	
.1732			4.40
.1770		16	
.1772			4.50
.1800		15	
.1811			4.60
.1820		14	
.1850		13	4.70
.1870			4.75

Inch thập phân	Phân số	Số / Mẫu tự	Hệ mét (mm)
.1875	3/16		
.1890		12	4.80
.1910		11	
.1929			4.90
.1935		10	
.1960		9	
.1969			5.00
.1990		8	
.2008			5.10
.2010		7	
.2031	13/64		
.2040		6	
.2047			5.20
.2055		5	
.2067			5.25
.2087			5.30
.2090		4	
.2126			5.40
.2130		3	
.2165			5.50
.2188	7/32		
.2205			5.60
.2210		2	
.2244			5.70
.2264			5.75
.2280		1	
.2283			5.80
.2323			5.90
.2340		A	
.2344	15/64		
.2362			6.00
.2380		B	
.2402			6.10
.2420		C	
.2441			6.20
.2460		D	
.2461			6.25
.2480			6.30
.2500	1/4	E	
.2520			6.40
.2559			6.50
.2570		F	
.2598			6.60
.2610		G	
.2638			6.70
.2556	17/64		
.2657			6.75
.2660		H	
.2677			6.80
.2717			6.90
.2720		I	
.2756			7.00
.2770		J	
.2795			7.10
.2810		K	
.2812	9/32		
.2835			7.20
.2854			7.25
.2874			7.30
.2900		L	
.2913			7.40

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Inch thập phân	Phân số	Số / Mẫu tự	Hệ mét (mm)
.2950			
.2953			7.50
.2969	19/64		
.2992			7.60
.3020		N	
.3031			7.70
.3051			7.75
.3071			7.80
.3110			7.90
.3125	5/16		
.3150			8.00
.3160		O	
.3189			8.10
.3228			8.20
.3230		P	
.3248			8.25
.3268			8.30
.3281	21/64		
.3307			8.40
.3320		Q	
.3346			8.50
.3386			8.60
.3390		R	
.3425			8.70
.3438	11/32		
.3445			8.75
.3465			8.80
.3480		S	
.3504			8.90
.3543			9.00
.3580		T	
.3583			9.10
.3594	23/64		
.3622			9.20
.3642			9.25
.3661			9.30
.3680		U	
.3701			9.40
.3740			9.50
.3750	3/8		
.3770		V	
.3780			9.60
.3819			9.70
.3839			9.75
.3858			9.80
.3860		W	
.3898			9.90
.3906	25/64		
.3937			10.00
.3970		X	
.4040		Y	
.4062	13/32		
.4130		Z	
.4134			10.50
.4219	27/64		
.4331			11.00
.4375	7/16		
.4528			11.50
.4531	29/64		
.4688	15/32		

Inch thập phân	Phân số	Số / Mẫu tự	Hệ mét (mm)
.4724			12.00
.4844	31/64		
.4921			12.50
.5000	1/2		12.70
.5118			13.00
.5156	33/64		
.5312	17/32		
.5315			13.50
.5469	35/64		
.5512			14.00
.5625	9/16		
.5709			14.50
.5781	37/64		
.5906			15.00
.5938	19/32		
.6094	39/64		
.6102			15.50
.6250	5/8		
.6299			16.00
.6406	41/64		
.6496			16.50
.6562	21/32		
.6693			17.00
.6719	43/64		
.6875	11/16		
.6890			17.50
.7031	45/64		
.7087			18.00
.7188	23/32		
.7283			18.50
.7344	47/64		
.7480			19.00
.7500	3/4		
.7656	49/64		
.7677			19.50
.7812	25/32		
.7874			20.00
.7969	51/64		
.8071			20.50
.8125	13/16		
.8268			21.00
.8281	53/64		
.8438	27/32		
.8465			21.50
.8594	55/64		
.8661			22.00
.8750	7/8		
.8858			22.50
.8906	57/64		
.9055			23.00
.9062	29/32		
.9219	59/64		
.9252			23.50
.9375	15/16		
.9449			24.00
.9531	61/64		
.9646			24.50
.9688	31/32		
.9843			25.00
.9844	63/64		
1.0000	1		25.40

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ren hệ Anh - UNC/UNF

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ren-TPI	Kích cỡ mũi khoan ta-rô	Inch	Mét
#0-80	3/64	.0469	
1/16-64	3/64	.0469	
#1-64	#53	.0595	
#1-72	#53	.0595	
#2-56	#50	.0700	
#2-64		.0709	1.80
3/32-48	#49	.0730	
#3-48	#47	.0785	
#3-56	#45	.0820	
#4-32	#45	.0820	
#4-36	#44	.0860	
#4-40	#43	.0890	
#4-48	#42	.0935	
#5-40	#39	.0995	
#5-44	#37	.1040	
1/8-40	#38	.1015	
#6-32	#36	.1065	
#6-36	#34	.1110	
#6-40	#33	.1130	
5/32-32	1/8	.1250	
5/32-36	#30	.1285	
#8-32	#29	.1360	
#8-36	#29	.1360	
#8-40	#28	.1405	
3/16-24	#26	.1470	
3/16-32	#22	.1570	
#10-24	#25	.1495	
#10-28	#23	.1540	
#10-30	#22	.1570	
#10-32	#21	.1590	
#12-24	#16	.1770	
#12-28	#14	.1820	
#12-32	#13	.1850	4.70
7/32-24	#16	.1770	
7/32-32	#12	.1890	4.80
#14-20	#10	.1935	
#14-24	#7	.2010	5.10
1/4-20	#7	.2010	5.10
1/4-28	#4	.2090	
1/4-32	7/32	.2188	5.50
5/16-18	F	.2570	6.50
5/16-20	17/64	.2656	
5/16-24	I	.2720	6.90
5/16-32	9/32	.2813	7.10
3/8-16	5/16	.3125	8.00
3/8-20	21/64	.3281	
3/8-24	Q	.3320	18.50
3/8-32	11/32	.3438	
7/16-14	U	.3680	9.40
7/16-20	25/64	.3906	9.90
7/16-24	X	.3970	10.00
7/16-28	Y	.4040	
1/2-13	27/64	.4219	
1/2-20	29/64	.4531	11.50
1/2-28	15/32	.4688	
9/16-12	31/64	.4844	
9/16-18	33/64	.5156	13.00
9/16-24	33/64	.5156	13.00
5/8-11	17/32	.5313	13.50
5/8-12	35/64	.5469	
5/8-18	37/64	.5781	
5/8-24	37/64	.5781	
11/16-12	39/64	.6094	
11/16-16	5/8	.6250	
11/16-24	41/64	.6406	
3/4-10	21/32	.6563	16.50
3/4-12	43/64	.6719	17.00
3/4-16	11/16	.6875	17.50
3/4-20	45/64	.7031	17.50
3/4-28	23/32	.7188	
13/16-12	47/64	.7344	
13/16-16	3/4	.7500	
7/8-9	49/64	.7656	19.50
7/8-12	51/64	.7969	20.00
7/8-14	13/16	.8125	
7/8-16	13/16	.8125	
7/8-20	53/64	.8281	
15/16-12	55/64	.8594	
15/16-16	7/8	.8750	
15/16-20	57/64	.8906	
1–8	7/8	.8750	
1–12	59/64	.9219	
1–14	15/16	.9375	
1–20	61/64	.9531	
1-1/16–12	63/64	.9844	
1-1/16–16	1.0	1.0000	

http://www.me-cad.com/

Ta-rô ren ống thẳng NPS

Kích cỡ mũi ta-rô	Ø mũi khoan ta-rô (in)	Mét
1/8–27	S	8.80
1/4–18	29/64	11.50
3/8–18	19/32	15.00
1/2–14	47/64	18.50
3/4–14	15/16	23.75
1–11-1/2	1-3/16	30.25
1-1/4–11-1/2	1-33/64	38.50
1-1/2–11-1/2	1-3/4	44.50
2–11-1/2	2-7/32	56.00

Kích cỡ ống	TPI	Mũi khoan ta-rô	Kích cỡ thập phân
1/16	27	1/4	.2500
1/8	27	11/32	.3438
1/4	18	7/16	.4375
3/8	18	37/64	.5781
1/2	14	23/32	.7188
3/4	14	59/64	.9219
1.0	11-1/2	1-5/32	1.1563
1-1/4	11-1/2	1-1/2	1.5000
1-1/2	11-1/2	1-3/4	1.7500
2.0	11-1/2	2-7/32	2.2188

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ta-rô ren ống côn NPT

Kích cỡ mũi ta-rô	Ø mũi khoan ta-rô (in)	Mét
1/16–27	D	6.30
1/8–27	R	8.70
1/4–18	7/16	11.10
3/8–18	37/64	14.50
1/2–14	45/64	18.00
3/4–14	59/64	23.25
1–11-1/2	1-5/32	29.00
1-1/4–11-1/2	1-1/2	38.00
1-1/2–11-1/2	1-47/64	44.00
2–11-1/2	2-7/32	56.00
2-1/2–8	2-5/8	67.00
3–8	3-1/4	82.50

		Chỉ khoan		Doa côn	
Kích cỡ ống	TPI	Mũi khoan ta-rô	Kích cỡ thập phân	Mũi khoan ta-rô	Kích cỡ thập phân
1/16	27	D	.2460	15/64	.2344
1/8	27	Q	.3320	21/64	.3281
1/4	18	7/16	.4375	27/64	.4219
3/8	18	37/64	.5781	9/16	.5625
1/2	14	45/64	.7031	11/16	.6875
3/4	14	29/32	.9062	57/64	.8906
1.0	11-1/2	1-9/64	1.1406	1-1/8	1.1250
1-1/4	11-1/2	1-31/64	1.4844	1-15/32	1.4688
1-1/2	11-1/2	1-47/64	1.7344	1-23/32	1.7188
2.0	11-1/2	2-13/64	2.2031	2-3/16	2.1875

Ren thô hệ mét

Ø danh định x bước (mm)	Ø mũi khoan ta-rô (mm)	Inch
M1 x 0.25	0.75	.0295
M1.2 x 0.25	0.95	.0374
M1.4 x 0.3	1.10	.0433
M1.5 x 0.35	1.15	.0453
M1.6 x 0.35	1.25	.0492
M1.8 x 0.35	1.45	.0571
M2 x 0.4	1.60	.0630
M2.2 x 0.45	1.75	.0689
M2.5 x 0.45	2.05	.0807
M3 x 0.5	2.50	.0984
M3.5 x 0.6	2.90	.1142
M4 x 0.7	3.30	.1299
M4.5 x 0.75	3.75	.1476
M5 x 0.8	4.20	.1654
M6 x 1	5.00	.1969
M7 x 1	6.00	.2362
M8 x 1.25	6.75	.2657
M9 x 1.25	7.75	.3051
M10 x 1.5	8.50	.3346
M11 x 1.5	9.50	.3740
M12 x 1.75	10.20	.3937
M14 x 2	12.00	.4724
M16 x2	14.00	.5512
M18 x 2.5	15.50	.6102
M20 x 2.5	17.50	.6890
M22 x 2.5	19.50	.7677
M24 x 3	21.00	.8268
M27 x 3	24.00	.9449
M30 x 3.5	26.50	1.0433

Ren mịn hệ mét

Ø danh định x bước (mm)	Ø mũi khoan ta-rô (mm)	Inch
M3 x 0.35	2.65	.1043
M3.5 x 0.35	3.15	.2283
M4 x 0.5	3.50	.1378
M4.5 x 0.5	4.00	.1575
M5 x 0.5	4.50	.1772
M5.5 x 0.5	5.00	.1969
M6 x 0.75	5.25	.2067
M7 x 0.75	6.25	.2461
M8 x 1	7.00	.2756
M9 x 1	8.00	.3150
M10 x 0.75	9.25	.3642
M10 x 1	9.00	.3543
M10 x 1.25	8.75	.3445
M11 x 1	10.00	.3937
M12 x 1	11.00	.4331
M12 x 1.25	10.75	.4232
M12 x 1.5	10.50	.4134
M13 x 1.5	11.50	.4528
M13 x 1.75	11.25	.4429
M14 x 1.25	12.75	.5020
M14 x 1.5	12.50	.4921
M15 x 1.5	13.50	.5315
M16 x 1	15.00	.5906
M16 x 1.5	14.50	.5709
M17 x 1.5	15.50	.6102
M18 x 1	17.00	.6693
M18 x 1.5	16.50	.6496
M18 x 2	16.00	.6299
M20 x 1	19.00	.7480
M20 x 1.5	18.50	.7283
M20 x 2	18.00	.7087
M22 x 1	21.00	.8268
M22 x 1.5	20.50	.8071
M22 x 2	20.00	.7874
M24 x 1	23.00	.9055
M24 x 1.5	22.50	.8858
M24 x 2	22.00	.8661
M25 x 1.5	23.50	.9252
M27 x 2	25.00	.9843
M28 x 2	26.00	1.0236
M30 x 2	28.00	1.1024
M30 x 3	27.00	1.0630

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

MỤC LỤC

Chương 1. ĐIỀU KHIỂN SỐ

Định nghĩa điều khiển số 7
Công nghệ NC và CNC 7
Gia công CNC và gia công cổ điển 8
Ưu thế của điều khiển số 8
Các kiểu máy công cụ CN C10
Máy phay và trung tâm gia công . . . 11
Máy tiện và trung tâm tiện 11
Nhân lực sử dụng CNC 11
Nhà lập trình CNC 12
Người vận hành máy CNC 12
An toàn với máy CNC 12
Hình học máy 13

Chương 2. PHAY CNC

Máy CNC - Phay 15
Các kiểu máy phay 15
Trung tâm gia công đứng 16
Trung tâm gia công ngang 17
Máy phay doa ngang 17
Các đặc tính kỹ thuật thông dụng . . . 18

Chương 3. TIỆN CNC

Máy CNC - Tiện 19
Các kiểu máy tiện CNC 19
Ký hiệu trục 19
Máy tiện hai trục 20
Máy tiện ba trục 20
Máy tiện bốn trục 20
Máy tiện sáu trục 21
Tính năng và đặc tính kỹ thuật 21
Đặc tính kỹ thuật máy tiện CNC . . . 21
Các tính năng điều khiển 22

Chương 4. HỆ THỐNG ĐIỀU KHIỂN

Khái quát 23
Tính năng hệ thống 25
Các xác lập tham số 26
Các mặc định hệ thống 27
Dung lượng nhớ 28
Dừng chương trình bằng tay 28
Thao tác single block 29
Feedhold 29
Emergency Stop 29
Nhập dữ liệu bằng tay - MDI 29
Vượt qua dữ liệu chương trình 29
Rapid Motion Override 30
Spindle Speed Override 30
Feedrate Override 31
Sự vận hành Dry Run 31
Z Axis Neglect 32
Manual Absolute Setting 32
Sequence Return 32
Khóa các hàm phụ 32
Machine Lock 32
Ứng dụng thực tế 33
Các tùy chọn hệ thống 33
Hiển thị đồ họa 33
Đo trong khi gia công 34
Giới hạn hành trình được lưu giữ . . . 34
Nhập kích thước vẽ 34
Chu kỳ gia công 34
Tạo hoạt hình dụng cụ cắt 34
Nối kết với thiết bị bên ngoài 35

Chương 5. QUY HOẠCH CHƯƠNG TRÌNH

Các bước quy hoạch chương trình 36
Lập trình bằng tay 36
CAD/CAM và CNC 36
Sự tích hợp 37
Các bước lập trình điển hình 37
Bản vẽ chi tiết 38
Bảng công nghệ 39
Đặc tính kỹ thuật của vật liệu 39
Định mức tính gia công 39
Chuỗi thứ tự gia công 40
Lựa chọn dụng cụ cắt 40
Gá lắp chi tiết 41
Quyết định công nghệ 41
Quỹ đạo dao cắt 41
Định mức công suất máy 42
Chất làm nguội và bôi trơn 42
Phác thảo chi tiết và tính toán 42

Chương 6. CẤU TRÚC CHƯƠNG TRÌNH GIA CÔNG

Các thuật ngữ lập trình cơ bản 44
Định dạng chương trình 45
Word Address Format 45
Giải thích định dạng 46
Dạng rút gọn 46
Định dạng hệ thống phay 46
Định dạng hệ thống tiện 47
Địa chỉ nhiều từ ngữ 48
Ký hiệu trong lập trình 48
Dấu cộng và dấu trừ 48
Tiêu đề chương trình 48
Cấu trúc chương trình 49

Chương 7. CÁC LỆNH CHUẨN BỊ

Ý nghĩa và công dụng 50
Ứng dụng trên hệ thống phay 50
Ứng dụng trên hệ thống tiện 51
Mã G trong block chương trình 52
Tính chế độ của các lệnh G 52
Va chạm các lệnh trong block 53
Thứ tự từ ngữ trong block 53

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chia nhóm các lệnh 54
Các kiểu mã G 55
Mã G và dấu thập phân 55

Chương 8. CÁC HÀM CHUNG

Giải thích và công dụng 56
Các hàm liên quan với máy 56
Các hàm liên quan với chương trình . 56
Các ứng dụng phổ biến 57
Các hàm MDI đặc biệt 57
Các nhóm ứng dụng 57
Hàm M trong block 58
Khởi động các hàm M 58
Thời hạn của hàm M 59
Các hàm chương trình 59
Dừng chương trình 59
Dừng chương trình tùy chọn 60
Kết thúc chương trình 60
Kết thúc chương trình con 61
Các hàm máy 61
Các hàm điều khiển chất làm nguội . 61
Các hàm trục chính 62
Phụ tùng máy 63

Chương 9. BLOCK CHUỖI THỨ TỰ

Cấu trúc block 64
Nhận biết chương trình 65
Số chuỗi thứ tự 66
Lệnh số chuỗi thứ tự 66
Định dạng block thứ tự 66
Đánh số theo số gia. 67
Chương trình dài và số block 67
Ký tự kết thúc block 67
STARTUP block hoặc SAFE block . . . 68
Sự va chạm từ ngữ trong block 69
Các giá trị lập trình chế độ 70
Tính ưu tiên thực thi 71

Chương 10. NHẬP KÍCH THƯỚC

Đơn vị hệ anh và hệ mét 72
Các chế độ tuyệt đối và số gia 73
Các lệnh chuẩn bị G90 và G91 74
Nhập dữ liệu tuyệt đối G90 75
Nhập dữ liệu số gia G91 75
Phối hợp trong block 75
Lập trình đường kính 76
Số gia chuyển động tối thiểu 76
Định dạng nhập kích thước 76
Nhập kiểu CALCULATOR 79

Chương 11. ĐIỀU KHIỂN TRỤC CHÍNH

Hàm trục chính 80
Chiều quay trục chính 80
Chiều phay 80
Chiều tiện 81
Các đặc tính chiều quay. 81
Khởi động trục chính 82
Dừng trục chính 83
Sự định hướng trục chính 83
Tốc độ trục chính - r/min 84
Tốc độ trục chính - bề mặt 84
Tốc độ bề mặt không đổi 85
Xác lập tốc độ trục chính cực đại . . . 87
Tính toán đường kính chi tiết trong chế độ CSS. 88

Chương 12. ĐIỀU KHIỂN TỐC ĐỘ CẮT

Điều khiển tốc độ cắt 89
Hàm tốc độ cắt 89
Lựa chọn lượng ăn dao 90
Tăng tốc và giảm tốc 90
Lệnh dừng chính xác 91
Lệnh chế độ dừng chính xác 91
Sự vượt qua góc một cách tự động. . . 91
Chế độ taro ren 91
Chế độ cắt 92
Sự ăn dao không đổi 92
Tốc độ cắt theo chế độ tròn 92
Tốc độ ăn dao cực đại 93
Các yếu tố tác động đến tốc độ cực đại 93
FEEDHOLD và OVERRIDE 93
Công tắc Feedhold 93
Công tắc Feedrate Override. 93
Các tính năng vượt qua tốc độ cắt. . . 94
Địa chỉ E trong tiện ren 94

Chương 13. HÀM DỤNG CỤ CẮT

Hàm T trên trung tâm gia công 96
Hộp giữ dao cắt. 96
Sự chọn dao cố định 96
Lựa chọn dao cắt nhớ ngẫu nhiên. . . 97
Đăng ký chỉ số dao cắt 97
Định dạng lập trình 97
Ổ dao rỗng 98
Hàm thay dao - M06 98
Bộ thay dao tự động - ATC 99
Hệ thống ATC thông dụng 99
Chu kỳ ATC. 100
Sự vận hành MDI 101
Lập trình ATC 101
Lập trình một dao cắt 101
Lập trình nhiều dao cắt 101
Dao bất kỳ trong trục chính - không phải dao thứ nhất. 102
Dao thứ nhất trong trục chính 103
Không có dao trong trục chính . . . 103
Dao thứ nhất trong trục chính với sự thay dao bằng tay 104
Không có dao trong trục chính với sự thay dao bằng tay. 105
Dao thứ nhất trong trục chính và dao quá khổ 105
Không có dao trong trục chính và dao quá khổ 105

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Hàm T đối với máy tiện . . . 106
Đăng ký bù dao cắt . . . 107
Geometry Offset . . . 107
Wear Offset . . . 108
Điều chỉnh wear offset . . . 108
Các xác lập R và T . . . 109

Chương 14. CÁC ĐIỂM QUY CHIẾU

Các nhóm điểm quy chiếu . . . 110
Điểm quy chiếu máy . . . 111
Điểm quy chiếu chi tiết . . . 112
Điểm quy chiếu dụng cụ cắt . . . 115

Chương 15. CÁC LỆNH ĐĂNG KÝ

Lệnh POSITION REGISTER . . . 116
Định nghĩa đăng ký vị trí . . . 116
Định dạng lập trình . . . 116
Xác lập vị trí dao cắt . . . 116
Ứng dụng các trung tâm gia công . . . 117
Xác lập dao ở zero máy . . . 117
Xác lập dao cách xa zero máy . . . 117
Đăng ký vị trí theo trục Z . . . 118
Ví dụ lập trình . . . 118
Ứng dụng trên máy tiện . . . 118
Gá lắp dao . . . 119
Các nhóm xác lập ba dao cắt . . . 119
Xác lập dao cắt đường tâm . . . 119
Xác lập dao tiện ngoài . . . 119
Xác lập dao tiện trong . . . 120
Chi tiết đỉnh góc . . . 120
Ví dụ lập trình . . . 120

Chương 16. BÙ VỊ TRÍ

Mô Tả . . . 122

Chương 17. BÙ CHI TIẾT

Vùng làm việc khả dụng . . . 125
Khởi đầu và mặc định bù chi tiết . . . 126
Thay đổi giá trị bù chi tiết . . . 127
Ứng dụng trục Z . . . 128
Áp dụng trên máy ngang . . . 129
Bù chi tiết ngoài . . . 129
Các ứng dụng tiện . . . 130
Các kiểu bù . . . 130
Bù hình học . . . 130
Bù mòn dao . . . 130
Chỉ số bù và dao . . . 130
Gá lắp dao . . . 131
Dao cắt gọt theo đường tâm . . . 131
Dao tiện . . . 131
Dao doa . . . 132
Điểm lệnh và bù dụng cụ cắt . . . 132

Chương 18. BÙ CHIỀU DÀI DAO

Nguyên lý chung . . . 133
Lệnh bù chiều dài dao . . . 134
Khoảng cách dịch chuyển theo trục Z . . . 134
Xác lập chiều dài dao . . . 135
Các quan hệ trên trục Z . . . 136
Xác lập trước chiều dài dao . . . 136
Sử dụng chiều dài dao chính . . . 137
Sự khác biệt giữa G43 và G44 . . . 138
Định dạng lập trình . . . 139
Chưa có bù chiều dài dao . . . 139
Bù chiều dài dao và G92 . . . 140
Bù chiều dài dao và G54 - G59 . . . 140
Bù chiều dài dao và nhiều dao . . . 141
Thay đổi bù chiều dài dao . . . 141
Áp dụng trên trung tâm gia công ngang . . . 142
Xóa bù chiều dài dao . . . 143

Chương 19. ĐỊNH VỊ NHANH

Chuyển động chạy dao nhanh . . . 144
Lệnh G00 . . . 144
Quỹ đạo dao chuyển động nhanh . . . 145
Kiểu chuyển động và so sánh thời gian . . . 147
Giảm tốc độ chạy dao nhanh . . . 147
Công thức chuyển động nhanh . . . 148
Tiến đến chi tiết . . . 148

Chương 20. TRẢ VỀ ZERO MÁY

Vị trí quy chiếu máy . . . 149
Trả về zero máy sơ cấp . . . 151
Điểm trung gian . . . 151
Chế độ tuyệt đối và chế độ số gia . . . 152
Trở về vị trí chiều sâu Z . . . 153
Trả về các trục cần thiết đối với ATC . . . 154
Trở về zero đối với máy tiện CNC . . . 155
Lệnh kiểm tra vị trí trả về . . . 155
Trở về từ điểm zero máy . . . 156
Trả về zero máy thứ cấp . . . 157

Chương 21. NỘI SUY TUYẾN TÍNH

Lệnh tuyến tính . . . 158
Định dạng lập trình . . . 159
Tốc độ ăn dao tuyến tính . . . 160
Ví dụ lập trình . . . 161

Chương 22. HÀM BỎ QUA BLOCK

Các ứng dụng cơ bản . . . 162
Ký hiệu bỏ qua block . . . 162
Xác lập bộ điều khiển . . . 162
Bỏ qua block và các lệnh chế độ . . . 163
Ví dụ lập trình . . . 164
Gia công cắt gọt thô . . . 164
Thay đổi biên dạng gia công . . . 165
Cắt gọt thử để đo . . . 167
Kiểm chứng chương trình . . . 168
Ứng dụng cấp phôi thanh . . . 169
Bỏ qua block theo số . . . 169

Chương 23. LỆNH DWELL

Ứng dụng lập trình . . . 170
Lệnh dwell . . . 170

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Lựa chọn thời gian tạm dừng . . . 171
Xác lập chế độ và dwell . . . 172
Dwell tối thiểu . . . 172
Số vòng quay . . . 172
Thời gian dwell dài . . . 173
Chu kỳ cố định và dwell . . . 174

Chương 24. CHU KỲ CỐ ĐỊNH

Gia công điểm - điểm . . . 176
Chuyển động dao và chu kỳ cố định . 177
Lựa chọn chu kỳ cố định . . . 177
Định dạng lập trình . . . 178
Các nguyên tắc chung . . . 178
Giá trị tuyệt đối và giá trị số gia . . . 179
Chọn mức ban đầu . . . 179
Chọn mức R . . . 180
Tính toán chiều sâu Z . . . 181
Các chu kỳ cố định . . . 181
G81 - Chu kỳ khoan . . . 182
G82 - Chu kỳ khoan điểm . . . 182
G83 - Chu kỳ khoan lỗ sâu - Tiêu chuẩn . . . 182
G73 - Chu kỳ khoan lỗ sâu - Tốc độ cao . . . 183
G84 - Chu kỳ tarô ren - Tiêu chuẩn . 184
G74 - Chu kỳ tarô ren - Ngược . . . 184
G85 - Chu kỳ doa . . . 185
G86 - Chu kỳ doa . . . 185
G87 - Chu kỳ doa ngược . . . 186
G88 - Chu kỳ doa . . . 186
G89 - Chu kỳ doa . . . 186
G76 - Chu kỳ doa chính xác . . . 187
Xóa chu kỳ cố định . . . 187
Lặp lại chu kỳ cố định . . . 187

Chương 25. GIA CÔNG CÁC LỖ

Đánh giá một lỗ . . . 189
Các nguyên công khoan . . . 192
Khoan nhiều đoạn trong một lỗ . . . 196
Ứng dụng khoan nhiều đoạn . . . 197
Tính toán số đoạn khoan . . . 197
Chọn số đoạn . . . 197
Điều khiển chiều sâu xuyên qua . . . 198
Chuốt . . . 198
Thiết kế dao chuốt . . . 198
Tốc độ trục chính khi chuốt lỗ . . . 199
Tốc độ cắt khi chuốt . . . 199
Lượng dư gia công . . . 199
Các vấn đề chuốt . . . 199
Doa một điểm . . . 199
Dao doa một điểm . . . 200
Hướng trục chính . . . 200
Dụng cụ khối . . . 201
Doa với dịch chuyển dao . . . 201
Chu kỳ doa chính xác G76 . . . 201
Chu kỳ doa ngược G87 . . . 201
Ví dụ lập trình . . . 202
Các chú ý trong lập trình và gá lắp . 202
Mở rộng các lỗ . . . 202
Khoan nhiều mức . . . 204
Khoan qua nhiều bậc . . . 205
Tarô ren . . . 206
Gia công lỗ trên máy tiện . . . 209
Chuyển động tiếp cận của dụng cụ cắt 210
Chuyển động dao trở về . . . 210
Khoan và chuốt trên máy tiện . . . 211
Chu kỳ khoan gián đoạn - G74 . . . 211
Tarô ren trên máy tiện . . . 212
Các nguyên công khác . . . 213

Chương 26. SƠ ĐỒ LỖ

Các kiểu sơ đồ lỗ . . . 214
Sơ đồ lỗ ngẫu nhiên . . . 214
Sơ đồ lỗ hàng ngang . . . 215
Sơ đồ lỗ hàng chéo . . . 215
Sơ đồ góc . . . 217
Sơ đồ lưới . . . 217
Sơ đồ lỗ hình cung tròn . . . 218
Sơ đồ lỗ theo đường tròn . . . 219
Hệ tọa độ cực . . . 221

Chương 27. PHAY BỀ MẶT

Lựa chọn dao cắt . . . 223
Các yếu tố cắt gọt . . . 224
Các kỹ thuật lập trình . . . 226
Phay bề mặt với một đường cắt . . . 226
Phay nhiều đường cắt . . . 227
Sử dụng bù vị trí . . . 229

Chương 28. NỘI SUY ĐƯỜNG TRÒN

Định dạng lập trình . . . 231
Lập trình bán kính . . . 234
Bán kính hòa hợp . . . 234
Bán kính riêng . . . 234
Lập trình đường tròn . . . 234
Lập trình cắt gọt cung tròn . . . 238
Tốc độ cắt đối với chuyển động tròn . . . 239

Chương 29. BÙ BÁN KÍNH DAO CẮT

Các tính toán bằng tay . . . 241
Quỹ đạo các tâm điểm dụng cụ cắt . . 242
Bán kính dao cắt . . . 243
Tính toán các điểm tâm . . . 243
Bù quỹ đạo dao . . . 244
Kỹ thuật lập trình . . . 244
Chiều chuyển động cắt . . . 244
Trái hoặc phải - không phải CW hoặc CCW . . . 245
Các lệnh bù . . . 245
Bán kính dao . . . 246
Lịch sử các kiểu bù . . . 246
Định dạng lập trình . . . 246
Địa chỉ H hoặc D . . . 247
Bù hình học và bù mòn dao . . . 247

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Áp dụng chương trình mẫu 246
Phương pháp khởi động 248
Xóa lệnh bù 249
Thay đổi chiều dao cắt 250
Cơ chế hoạt động của bù bán kính 250
Kiểu bù dự đoán trước 251
Nguyên tắc đối với bù bán kính kiểu dự đoán trước 251
Bán kính dao cắt 252
Sự cản trở bù bán kính 253
Khởi động đơn trục và đa trục 254
Tóm tắt các nguyên tắc chung 255
Ví dụ thực tiễn - phay 256
Bù bán kính mũi dao 260
Đỉnh mũi dao 260
Lệnh bù bán kính 260
Định hướng đỉnh dao 260
Tác dụng của bù bán kính mũi dao . . 261
Chương trình mẫu 261
Khoảng hở cần thiết tối thiểu 262
Thay đổi chiều chuyển động 262

Chương 30. LỰA CHỌN MẶT PHẲNG

Gia công trong các mặt phẳng 263
Lệnh lập trình xác định mặt phẳng . 264
Trạng thái điều khiển mặc định . . . 264
Chuyển động thẳng trong mặt phẳng . . 264
Nội suy tròn trong mặt phẳng 265
G17-G18-G19 các lệnh chế độ 266
Không có dữ liệu trục trong block . . 266
Bù bán kính dao trong mặt phẳng . . 267
Ví dụ thực tiễn 267
Chu kỳ cố định trong mặt phẳng . . . 268

Chương 31. PHAY CHU VI

Dao phay mặt đầu 269
Tốc độ trục chính và tốc độ cắt 270
Cắt gọt phôi 271
Ăn dao thẳng xuống 271
Ăn dao nghiêng vào và ra 272
Chiều cắt gọt 272
Chiều rộng và chiều sâu cắt 272

Chương 32. GIA CÔNG RÃNH VÀ HỐC

Biên hở và kín 273
Lập trình các rãnh 273
Phay hốc 276
Hốc chữ nhật 277
Lượng dư gia công 277
Chiều rộng cắt 278
Chiều dài cắt 278
Chuyển động bán tinh 278
Quỹ đạo dao gia công tinh 279
Chương trình phay hốc chữ nhật . . . 279
Hốc tròn 280
Đường kính dao tối thiểu 280
Phương pháp ăn vào 281
Tiếp cận tuyến tính 281
Tiếp cận tuyến tính - tròn 281
Gia công thô hốc tròn 282
Chu kỳ gia công hốc tròn 283

Chương 33. TIỆN VÀ DOA

Hàm dụng cụ cắt - dao tiện 284
Địa chỉ T284
Các chế độ bù trên máy tiện 285
Lập trình nhiều giá trị bù 286
Phương pháp chung 287
Dung sai đường kính 287
Dung sai trên bậc trụ 287
Dung sai đường kính và bậc trụ . . . 288
Xác lập bù 289
Các hàm khoảng tốc độ 289
Vạt góc tự động 289
Điều kiện lập trình 292
Ví dụ lập trình 292
Gia công thô và tinh 292
Lập trình mặt lõm 294
Tốc độ trục chính trong chế độ CSS . . 295
Định dạng lập trình tiện 296
Định dạng chương trình - Mẫu chuẩn 296
Định dạng chương trình chung 296
Tiếp cận chi tiết gia công 296

Chương 34. CHU KỲ TIỆN

Sự cắt gọt trên máy tiện 298
Nguyên lý của các chu kỳ tiện 298
G90 - Chu kỳ cắt gọt thẳng 298
G94 - Chu kỳ gia công mặt đầu 302
Chu kỳ lặp nhiều lần 303
Khái quát 303
Các kiểu định dạng chu kỳ 303
Các chu kỳ cắt gọt và biên dạng chi tiết 303
Các chu kỳ bẻ gãy phoi 304
Các chu kỳ cắt gọt biên dạng 304
Định nghĩa đường bao 304
Điểm khởi đầu và các điểm P và Q . 304
Chu kỳ kiểu I và kiểu II 305
Lập trình các chu kỳ Kiểu I và Kiểu II 305
Định dạng chu kỳ 305
G71 - Chu kỳ tiện 306
Định dạng chu kỳ G71 - 10T/11T/15T 306
Định dạng chu kỳ G71 - 0T/16T/18T/20T/21T 306
G71 tiện thô mặt ngoài 306
G71 tiện thô mặt trong 307
Chiều cắt trong G71 307
G72 - Gia công mặt đầu 307
Định dạng chu kỳ G72 - 10T/11T/15T 307
Định dạng chu kỳ G72 - 0T/16T/18T/20T/21T 308
G73 - Chu kỳ lặp gia công theo mẫu chuẩn 308

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Định dạng chu kỳ G73 - 10T/11T/15T 308
Định dạng chu kỳ G73 - 0T/16T/18T/20T/21T 309
G70 - Chu kỳ gia công tinh biên dạng . . 310
Định dạng chu kỳ G70 - tất cả các bộ điều khiển 310
Nguyên tắc cơ bản của các chu kỳ G70-G73 311
G74 - Chu kỳ khoan gián đoạn 311
Định dạng chu kỳ G74 - 10T/11T/15T 311
Định dạng chu kỳ G74 - 0T/16T/20T/21T 311
Chu kỳ cắt rãnh - G75. 312
Định dạng chu kỳ G75 - 10T/11T/15T 312
Định dạng chu kỳ G75 - 0T/16T/18T/20T/21T 312
Các nguyên tắc cơ bản của G74 và G75 . 312

Chương 35. CẮT RÃNH TRÊN MÁY TIỆN

Các nguyên công cắt rãnh 313
Hình dạng rãnh 313
Vị trí rãnh 314
Kích thước rãnh 314
Lập trình rãnh đơn giản 315
Kỹ thuật gia công rãnh chính xác 316
Nhiều rãnh 320
Rãnh ở mặt đầu 320
Rãnh góc/rãnh bậc 321
Chu kỳ gia công rãnh 322
Ứng dụng chu kỳ G75 322
Cắt rãnh đơn với G75 322
Gia công nhiều rãnh với G75 323
Các rãnh đặc biệt 323
Rãnh và chương trình con 323

Chương 36. CẮT ĐỨT

Quy trình cắt đứt 325
Dao cắt đứt 325
Lượng dư gia công 326
Chuyển động trả dao 327
Cắt đứt có vạt góc 327
Tránh hư hại chi tiết 328

Chương 37. GIA CÔNG REN MỘT LƯỠI CẮT

Gia công ren trên máy tiện CNC 329
Thuật ngữ về ren 330
Quy trình gia công ren 330
Tốc độ cắt ren và tốc độ trục chính . . . 334
Điểm quy chiếu dao 337
Lập trình tiện ren theo block 337
Chu kỳ tiện ren cơ bản - G92 338
Chu kỳ lặp nhiều lần - G76 339
Định dạng chu kỳ G72 - 10T/11T/15T 340
Định dạng chu kỳ G76 - 0T/16T/18T . 340
Ví dụ lập trình 341
Tính toán ren thứ nhất 341
Phương pháp ăn dao 342
Ăn dao hướng tâm 342
Ăn dao kết hợp 343
Góc dao cắt ren - Tham số A. 343
Kiểu cắt gọt ren - Tham số P 343
Các tính toán trong phương pháp một block 344
Các xem xét ban đầu 344
Tính toán vị trí khởi đầu trục Z . . . 345
Chuyển động lùi dao 346
Các hàm lùi dao tiện ren 346
Chiều ren 347
Tiện ren gần vai trục 347
Các dạng ren khác 349
Ren côn 350
Chiều sâu và các khoảng hở 350
Tính toán độ côn 350
Ren côn lập trình theo block. 351
Ren côn sử dụng chu kỳ đơn giản . . 352
Ren côn và chu kỳ lặp 352
Ren nhiều đầu mối 353
Cắt lại ren 355

Chương 38. CHƯƠNG TRÌNH CON

Chương trình chính và các chương trình con 356
Công dụng của chương trình con . . . 356
Nhận biết chương trình con 357
Các hàm chương trình con 357
Hàm gọi chương trình con 357
Hàm kết thúc chương trình con . . . 357
Số block trở về 358
Số lần lặp lại chương trình con . . . 358
LO/KO trong lệnh gọi chương trình con 359
Đánh số chương trình con 360
Phát triển chương trình con 363
Sắp xếp chương trình con đa mức 365
Gia công biên dạng với chương trình con 367
Chương trình con thay dao 368
Lưới 100 000 000 lỗ 368

Chương 39. ĐỊCH CHUYỂN MỐC CHUẨN

Dịch chuyển mốc chuẩn với G92 hoặc G50 370
Hệ tọa độ cục bộ 372
Lệnh G52 372
Hệ tọa độ máy 373
Xác lập dữ liệu 374
Bù chi tiết 375
Bù chiều dài dao 376
Bù bán kính dao 376
Bù khi tiện 377
Xác lập dữ liệu MDI 377
Nhập tham số lập trình 377

Chương 40. HÌNH ẢNH GƯƠNG

Nguyên tắc cơ bản của hình ảnh gương . 381
Chiếu gương bằng xác lập 383
Hình ảnh gương lập trình 384
Hình ảnh gương trên máy tiện CNC . . . 386

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Chương 41. QUAY TOẠ ĐỘ
Các lệnh quay 387
Ứng dụng thực tiễn 390

Chương 42. HÀM LẬP TỶ LỆ
Mô tả 393
Định dạng lập trình 393
Ví dụ lập trình 395

Chương 43. PHỤ TÙNG MÁY TIỆN CNC
Điều khiển mâm cặp 397
Ụ động và chuôi ụ động 398
Phân độ ổ dao hai chiều 399
Đồ gá cấp phôi thanh 401
Các tùy chọn bổ sung 402
Ví dụ lập trình 403

Chương 44. PHAY XOẮN
Nguyên công phay xoắn 405
Phay ren 406
Đường xoắn 407
Ví dụ phay ren 408
Ren thẳng 408
Phay ren 412
Các chuyển động lùi ra 412
Chương trình hoàn chỉnh 412
Phay ren ngoài 413
Phay ren côn 413
Các khảo sát cơ bản 413
Phương pháp mô phỏng phay ren 413
Khoét xoắn 414

Chương 45. GIA CÔNG CẮT GỌT NGANG
Các trục quay và phân độ 416
Bàn phân độ (trục B) 416
Trục B và chế độ bù 418
Bù chi tiết và trục B 418
Bù chiều dài và trục B 419
Trở về zero máy 421
Phân độ và chương trình con 421
Chương trình ví dụ hoàn chỉnh 422
Bộ thay pallet tự động - APC 424
Phay doa ngang425

Chương 46. VIẾT CHƯƠNG TRÌNH CNC
Viết chương trình 427
Các dạng lập trình 427
Các ký tự dễ bị nhầm lẫn 427
Định dạng xuất chương trình 428
Chương trình dài 429
Rút gọn chương trình 430
Chế độ nhớ và chế độ băng 431

Chương 47. GIAO DIỆN VÀ CÁC THIẾT BỊ
Giao diện RS-232C 432
Băng đục lỗ 433
Bộ đọc băng đục lỗ băng 433
Đầu và cuối băng 435
Nhận biết băng 435
Các ký tự không in 435
Bảo quản và vận chuyển 435
Điều khiển số phân phối 435
Thuật ngữ truyền thông 436
Tốc độ baud 436
Tính chẵn lẻ 436
Các bit dữ liệu 436
Bit khởi đầu và bit dừng 436
Xác lập dữ liệu 436
Cáp nối 436
Null Modem 437
Nối cáp giữa Fanuc và máy tính cá nhân 437

Chương 48. CNC và CAD/CAM
Lập trình bằng tay? 438
Phần mềm CAM 438
Lập trình trên máy tính để bàn . . . 438
Phát triển hình học quỹ đạo dao 439
Tạo quỹ đạo dao 439
Môi trường hoàn chỉnh 439
Hỗ trợ nhiều máy 440
Các nguyên công liên quan 440
Gá lắp 440
Danh sách dao và các chú thích gia công 440
Nối kết giữa các máy tính 440
Bộ biên tập văn bản chương trình . . 440
Khả năng in ấn 441
Máy vẽ đồ họa 441
Truy cập phần mềm CAD 441
Hỗ trợ hình khối 441
Đặc tính kỹ thuật phần mềm 442
Đặc tính kỹ thuật phần cứng 442
Yêu cầu phần cứng 442
Các trình tiện ích và tính năng đặc biệt 443
Xử lý hậu kỳ 443
Chuyên biệt hóa xử lý hậu kỳ 443
Các tính năng quan trọng 443
Nhập từ người dùng 443
Các chu kỳ gia công 443
Giao diện người dùng 443
Giao diện CAD 444
Kết luận 444

CÁC BẢNG THAM CHIẾU
Đương lượng thập phân 445
Ren hệ Anh - UNC/UNF 448
Ta-rô ren ống thẳng NPS 448
Ta-rô ren ống côn NPT 449
Ren thô hệ mét 449
Ren mịn hệ mét 449

http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Nhiều tài liệu hay có tại : http://www.me-cad.com/

Ấn Bản MỚI

SỔ TAY LẬP TRÌNH CNC

THỰC HÀNH - LẬP TRÌNH GIA CÔNG TRÊN MÁY CNC



http://www.me-cad.com/

Nơi Phát Hành

HIỆU SÁCH NGUYỄN TRÃI

4A - Ng Trãi - Q5 - Tp. HCM

ĐT: 8383669

Nhiều tài liệu hay có tại : http://www.me-cad.com/