COMPUTER NUMERICAL CONTROL:

SME

CONCEPTS AND PROGRAMMING

Second Edition

WARREN S. SEAMES

SOCIETY OF MANUFACTURING ENGINEERS & DELMAR PUBLISHERS INC.

A PARTNERSHIP IN EDUCATIONAL EXCELLENCE

SME and DELMAR have proudly joined forces to form a partnership dedicated to educational excellence. We believe that quality manufacturing education is the key to keeping America competitive in the years ahead.

The Society of Manufacturing Engineers is an international technical society dedicated to advancing scientific knowledge in the field of manufacturing. SME has more than 80,000 members in 70 countries and serves as a forum for engineers and managers to share ideas, information, and accomplishments.

To be successful, today's engineers and technicians must keep pace with the torrent of information that appears each day. To meet this need, SME provides, in addition to the publication of books, many opportunities in continuing education for its members. These opportunities include: monthly meetings through five associations and more than 300 chapters; educational programs including seminars, clinics, and videotapes, as well as conferences and expositions.

Today's manufacturing technology students represent our future. Our goal is to provide these students with the finest manufacturing technology educational products. By pooling our many resources, SME and DELMAR are going to help teachers get the job done.

Together SME and DELMAR will provide outstanding educational materials to prepare students to enter the real world of manufacturing.

Thomas J. Drozda
Director of Publications
Society of Manufacturing Engineers

Gregory C. Spatz
President
Delmar Publishers Inc.

COMPUTER NUMERICAL CONTROL

CONCEPTS AND PROGRAMMING

2nd Edition

WARREN S. SEAMES

DELMAR PUBLISHERS INC.®

NOTICE TO THE READER

Dedication

This book is dedicated to my wife, Dolores, for her love, understanding, and patience throughout this project.

Cover Photos: Keyboard photo courtesy of Cincinnati Milacron Marketing Co.
Industrial equipment photo courtesy of KGK International Corp.

Delmar Staff
Executive Editor: David C. Gordon
Associate Editor: Marjorie A. Bruce
Project Editor: Eleanor Isenhart
Production Coordinator: Larry Main
Design Coordinator: Susan C. Mathews

For information, address Delmar Publishers Inc.
2 Computer Drive West, Box 15-015
Albany, New York 12212

10 9 8 7 6 5 4 3 2 1

Printed in the United States of America
Published simultaneously in Canada
by Nelson Canada,
A Division of the Thomson Corporation

Library of Congress Cataloging in Publication Data

Seames, Warren S.
Computer numerical control: concepts and programming/Warren S. Seames.—2nd ed.
p. cm.
Includes index.
ISBN 0-8273-3782-5.—ISBN 0-8273-3783-3 (instructor's guide)
1. Machine-tools—Numerical control. I. Title.
TJ1189.S36 1990 89-11700
621.9'023—dc20 CIP

CONTENTS

PREFACE x

CHAPTER 1 ■ An Introduction to Numerical Control Machinery 1
The History of NC / 2
CNC Machines / 5
Input Media / 5
Binary Numbers / 7
Tape Formats / 8
Objectives of Numerical Control / 11
Applications in Industry / 14

CHAPTER 2 ■ Numerical Control Systems 18
Components / 18
Types of Control Systems / 18
Servomechanisms / 22
Loop Systems / 22
The Cartesian Coordinate System / 25
Positive and Negative Movement / 25
Positioning Systems / 29
Setting the Machine Origin / 31
Dimensioning / 37

CHAPTER 3 ■ Process Planning and Tool Selection 40
Process Planning / 40
Tooling for Numerical Control / 42
Tooling for Hole Operations / 45
Milling Cutters / 49
Special Inserted Cutters / 52
A Processing Example / 54
Speeds and Feeds / 59

CHAPTER 4 ■ Tool Changing and Tool Registers 75
Tool Changes / 75
Automatic Tool Changers / 81
Tool Storage / 92
Tool Length and Offset / 95

CHAPTER 5 ■ Programming Coordinates 99
Hole Operations / 99
Milling Operations / 101
Mixing Absolute and Incremental Positioning / 105
Metric Coordinates / 105

CHAPTER 6 ■ Two-Axis Programming 111
Machinist Shop Language / 111
Drilling in Machinist Shop Language / 113
Word Address Format / 120
Drilling in Word Address Format / 121
Milling in Machinist Shop Language / 126
Milling in Word Address Format / 136
Fanuc Controller Applications / 142

CHAPTER 7 ■ Three-Axis Programming 149
A Three-Axis Programming Task / 149
Machinist Shop Language / 154
Word Address Format / 166
Other G Codes Used in CNC Programming / 177
Special Fanuc Section / 181

CHAPTER 8 ■ Math for Numerical Control Programming 187
Basic Trigonometry / 187
Using Trigonometry for Cutter Offsets / 191
A Milling Example / 194
A Lathe Example / 198

CHAPTER 9 ■ Linear and Circular Interpolation 203
Linear Interpolation / 203
Circular Interpolation / 209

CHAPTER 10 ■ Cutter Diameter Compensation 219
Definitions and Codes / 219
Machinist Shop Language / 223
Word Address Format / 229
Special Considerations / 232
Fine Tuning with Cutter Diameter Compensation / 240
Special Fanuc Section / 241

CHAPTER 11 ■ Do Loops and Subroutines 247
Do Loops / 247
Subroutines / 255
Nested Loops / 264
Special Fanuc Section / 269

CHAPTER 12 ■ Advanced CNC Features 276
Mirror Imaging / 276
Polar Rotation / 285
Helical Interpolation / 293

CHAPTER 13 ■ The Numerical Control Lathe 302
Lathe Bed Design / 302
Axis Movement / 305
Toolholders and Tool Changing / 306
Spindle Speeds / 311
Feedrates / 313
Machine Origin and Work Coordinate Systems / 313
Quicksetters / 314

CHAPTER 14 ■ Programming CNC Turning Machines 316
Machine Reference Point / 316
Diameter vs. Radius Programming / 317
Turning and Facing / 317
Taper Turning / 323
Circular Interpolation / 328
Drilling / 331
Threading / 334
A Complete Lathe Example / 343
Canned Cycles / 350

CHAPTER 15 ■ Use of Computers in Numerical Control Programming 356
Offline Programming Terminals / 356
Computer-Aided Programming / 357
Computer Graphics Programming / 386

CHAPTER 16 ■ The Future of Numerical Control 392
NC in Prototype and Job Shops / 392
CNC in Manufacturing / 393
Employment Opportunities in NC / 396

APPENDIX 1 ■ EIA Codes 399
Preparatory Functions / 399
Miscellaneous Functions / 400
Other Addresses / 401

APPENDIX 2 ■ Machinist Shop Language Commands 402
Commands / 402
Preparatory Functions (G Codes) / 403
Variable (V) Codes / 404
Auxiliary (AUX) Codes / 404

APPENDIX 3 ■ Word Address Codes Used in Text Examples 406
Preparatory Functions (G Codes) Used in Milling / 406
Miscellaneous (M) Functions Used in Milling and Turning / 407
Preparatory Functions (G Codes) Used in Turning / 408

APPENDIX 4 ■ Codes in Common Use with Tape Machinery 410
Preparatory Functions (g Codes) / 410
Miscellaneous (m) Functions / 410

APPENDIX 5 ■ Safety Rules for Numerical Control 412
Safety Rules for Operating Machines / 412
Safety Rules for Programmers / 413

APPENDIX 6 ■ Useful Machining Formulas and Data 414
Machining Formulas / 414
Cutting Speed Data / 415
Feedrate Data / 416

APPENDIX 7 ■ Lathe Canned Cycle Example 422
Tooling Used / 422
Sequence of Operations / 422
Program Notes / 423
Canned Cycle Notes / 425

APPENDIX 8 ■ Sample Programs 427
Sample Part Program / 427
Sample Milling Program / 435
Sample Lathe Program / 439

GLOSSARY 444

RELATED SME TITLES 473

INDEX 474

PREFACE

The second edition of *Computer Numerical Control: Concepts and Programming* offers an updated and improved introduction to the basic principles of computer numerical control, plus expanded programming examples and part problems for programming.

The programs presented in this text are not as complex as those found in industry. The examples were carefully selected to demonstrate the *basic* concepts of CNC programming. It was the author's goal to eliminate the learner's confusion that often accompanies the introduction of CNC programming. The learner will achieve an understanding of the programming principles through the programming applications and problems presented in the text. With a firm understanding of the basic principles, the learner can then move confidently ahead to apply the principles to actual industrial situations. Programming is an art, and like art in its generally visualized form, the programming practices used by programmers or NC instructors may vary. However, the basic principles of CNC programming are exhibited in each programmer's effort.

This new edition of *Computer Numerical Control* reflects the maturing of the industry with a complementary trend to standardization in codes and controllers. It has moved from a generic approach to one which reflects more programming for specific controllers to provide learners with real world practice.

FEATURES

The following features of this text aid the learner in understanding the principles of computer numerical control and mastering programming using the word address format.

- Learning objectives to identify the essential knowledge gained from a study of the chapter.
- Numerous illustrations to clarify and augment text discussions and highlight applications. Photos to identify equipment commonly found in industry and expand learner awareness of the real world of CNC.
- Solved programming examples with explanations of each step in the program to familiarize learners with the logical progression of commands and the use of the standard G and M codes.
- Emphasis on the word address format in programming examples and problems to mirror industrial usage.

- Emphasis on the specific algebra and trigonometry functions needed to solve cutter path geometry problems. Solved examples, clear explanations, and diagrams to clarify these math concepts.
- Introduction to numerical control principles to serve as a review or to make learners aware of the development of the industry to its current state.
- Clear, concise explanations of essential programming concepts including Cartesian coordinate system, part and machine coordinate systems, setting the machine origin and dimensioning (absolute and incremental).
- Explanation of the special requirements for tooling for NC, including tooling for hole operations and milling. Covers cutting tool materials (including carbide inserts) and calculations of speeds and feeds with solved problems.
- Coverage of programming functions, such as linear interpolation, circular interpolation, cutter diameter compensation, do loops, subroutines, nested loops, mirror imaging, polar rotation and helical interpolation.
- Discussion of the future of numerical control and job opportunities in this expanding industry.
- Sample part programs from industry for learner analysis.
- Illustrated glossary to reinforce understanding of essential terminology.

The major changes for the second edition include the following items.

- In Chapter 2, added content on setting machine origin and the difference between the part coordinate system and the machine coordinate system, with methods of transferring from one to the other.
- A new Chapter 3 on tooling now includes an introduction to process planning with an example showing the steps and documents utilized in planning the machining of a part prior to the actual programming of the machining process. The chapter also includes new and updated content on tools specifically designed for CNC, tooling materials and greater emphasis on the calculation of speeds and feeds.
- In Chapters 6, 7, 10, 11 and 15, added "Special Fanuc Section" consisting of a part drawing and instructions to program the cutting on a specific machine using a specific Fanuc controller. All special codes for the specific controller are defined.
- In Chapter 8, added a milling example and a lathe example requiring the computation of multiple cutter locations. Solutions are provided to show learners the steps to be taken.
- In Chapter 10, expanded the discussion of cutter diameter compensation to fine tune (compensate) for the difference between the programmed cutter diameter and the actual cutter diameter.
- Updated many programs and program explanations to reflect current usage of G codes in word address format.
- Chapters 13 and 14 on NC lathes, expanded with discussion of canned cycles and threading cycles.

- In Chapter 15, added a sample APT programming problem, including part drawing, part geometry figure, and postprocessor output.
- Added section on "Computer Graphics Programming" to stress the growing use of CAM software in manufacturing; also discusses CAD/CAM programming and computer integrated manufacturing (CIM).
- Added programming examples and problems in word address format.
- Added new Appendix 7, Lathe Canned Cycle Example, with part drawing, program, explanation of tooling used, sequence of operations, and program notes.

SOCIETY OF MANUFACTURING ENGINEERS (SME)

Delmar Publishers Inc. and the Society of Manufacturing Engineers have joined together to provide exceptional educational materials for the preparation of students to enter the manufacturing industries equipped with essential skills, and to further the education of those already employed in manufacturing. A list of complementary texts and other educational materials available from the Society of Manufacturing Engineers can be found following the Glossary.

ABOUT THE AUTHOR

Warren S. Seames is an experienced programmer with many years experience in industry and in teaching NC programming at the community college level. Mr. Seames is a former instrument maker and journeyman machinist. Presently he is an NC programmer, computer systems analyst, and resident applications programmer in industry. Mr. Seames is a member of the Society of Manufacturing Engineers.

ACKNOWLEDGMENTS

The author is indebted to the following individuals who reviewed the first edition and supplied many recommendations for the revised edition.

Alva B. Powell, Macomb Community College, Warren, MI
Jim Newman, Jefferson College, Hillsboro, MO
T. W. Sorrell, Northeast Missouri State University, Kirksville, MO

Lorin V. Waitkus, Cuyahoga Community College, Cleveland, OH
George T. Richardson, Mt. San Antonio College, Walnut, CA
Gary Dauer, Staples Technical Institute, Staples, MN
James H. Morris, Joliet Junior College, Joliet, IL
Melvin B. Gage, Linn Technical College, Linn, MO
David Dickie, Fresno City College, Fresno, CA
Ray Greb, Mesa State College, Grand Junction, CO
Robert G. Dixon, Chemeketa Community College, Salem, OR
Stanley Smolinsky, ITT Technical Institute, West Covina, CA

The following CNC instructors provided detailed critical reviews of the manuscript for the second edition of the text. Their contributions to the development of the new edition are gratefully acknowledged.

Mark E. Meyer, College of DuPage, Glen Ellyn, IL 60137
David Dickie, Fresno City College, Fresno, CA 93741
Ken S. Friend, Indiana Vocational Technical College, Richmond, IN 47274
Richard Colacino, National Tooling & Machining Association Training Center, Santa Clara, CA 95054
Melvin B. Gage, Linn Technical College, Linn, MO 65051
Ray Greb, Jr., Mesa State College, Grand Junction, CO 81501

CHAPTER 1

An Introduction to Numerical Control Machinery

OBJECTIVES Upon completion of this chapter, you will be able to:

- Describe the difference between direct and distributive numerical control.
- Describe the difference between a numerical control tape machine and a computer numerical control machine.
- Describe four ways that programs can be entered into a computer numerical controller.
- Explain two tape code formats in use with computer numerical control (CNC) machinery.
- Give the major objectives of numerical control.

Welcome to the world of numerical control. Numerical control (NC) has become popular in shops and factories because it helps solve the problem of making manufacturing systems more flexible. In simple terms, a *numerical control machine* is a machine positioned automatically along a preprogrammed path by means of coded instructions. The key words here are "preprogrammed" and "coded." Someone has to determine what operations the machine is to perform and put that information into a coded form that the NC control unit understands before the machine can do anything. In other words, someone has to program the machine.

Machines may be programmed manually or with the aid of a computer. Manual programming is called *manual part programming;* programming done by a computer is called *computer aided programming (CAP).* Sometimes a manual program is entered into the machine's controller via its own keypad. This is known as *manual data input (MDI).* This text will focus on manual part programming.

Advances in microelectronics and microcomputers have allowed the computer to be used as the control unit on modern numerical control machinery. This computer takes the place of the tape reader found on earlier NC machines. In other words, instead of reading and executing the program directly from punched tape, the program is loaded into and executed from the machine's computer. These machines, known as *computer numerical control* (CNC) machines, are the NC machines being manufactured today. The primary focus of this text is the MDI programming of *computer numerical control* (CNC) machinery.

THE HISTORY OF NC

In 1947, John Parsons of the Parsons Corporation, began experimenting with the idea of using three-axis curvature data to control machine tool motion for the production of aircraft components. In 1949, Parsons was awarded a U.S. Air Force contract to build what was to become the first numerical control machine. In 1951, the project was assumed by the Massachusetts Institute of Technology. In 1952, numerical control arrived when MIT demonstrated that simultaneous three-axis movements were possible using a laboratory-built controller and a Cincinnati Hydrotel vertical spindle. By 1955, after further refinements, numerical control became available to industry.

Early NC machines ran off punched cards and tape, with tape becoming the more common medium. Due to the time and effort required to change or edit tape, computers were later introduced as aids in programming. Computer involvement came in two forms: computer aided programming languages and direct numerical control (DNC). *Computer aided programming languages* allowed a part programmer to develop an NC program using a set of universal "pidgin English" commands, which the computer then translated into machine codes and punched into the tape. *Direct numerical control* involved using a computer as a partial or complete controller of one or more numerical control machines (see Figure 1–1). Although some companies have been reasonably successful at implementing DNC, the expense of computer capability and software and problems associated with coordinating a DNC system renders such systems economically unfeasable for all but the largest companies.

Recently a new type of DNC system called *distributive numerical control* has been developed (Figure 1–2). It employs a network of computers to coordinate the operation of a number of CNC machines. Ultimately, it may be possible to coordinate an entire factory in this manner. Distributive numerical control solves some of the problems that exist in coordinating a direct numerical control system. There is another type of distributive numerical control that is a spin-off of the system previously explained. In this system, the NC program is transferred in its entirety from a host computer directly to the machine's controller. Alternately, the program can be transferred from a mainframe host computer to a personal computer (PC) on the shop floor where it will be stored until it is needed. The program will then be transferred from the PC to the machine controller.

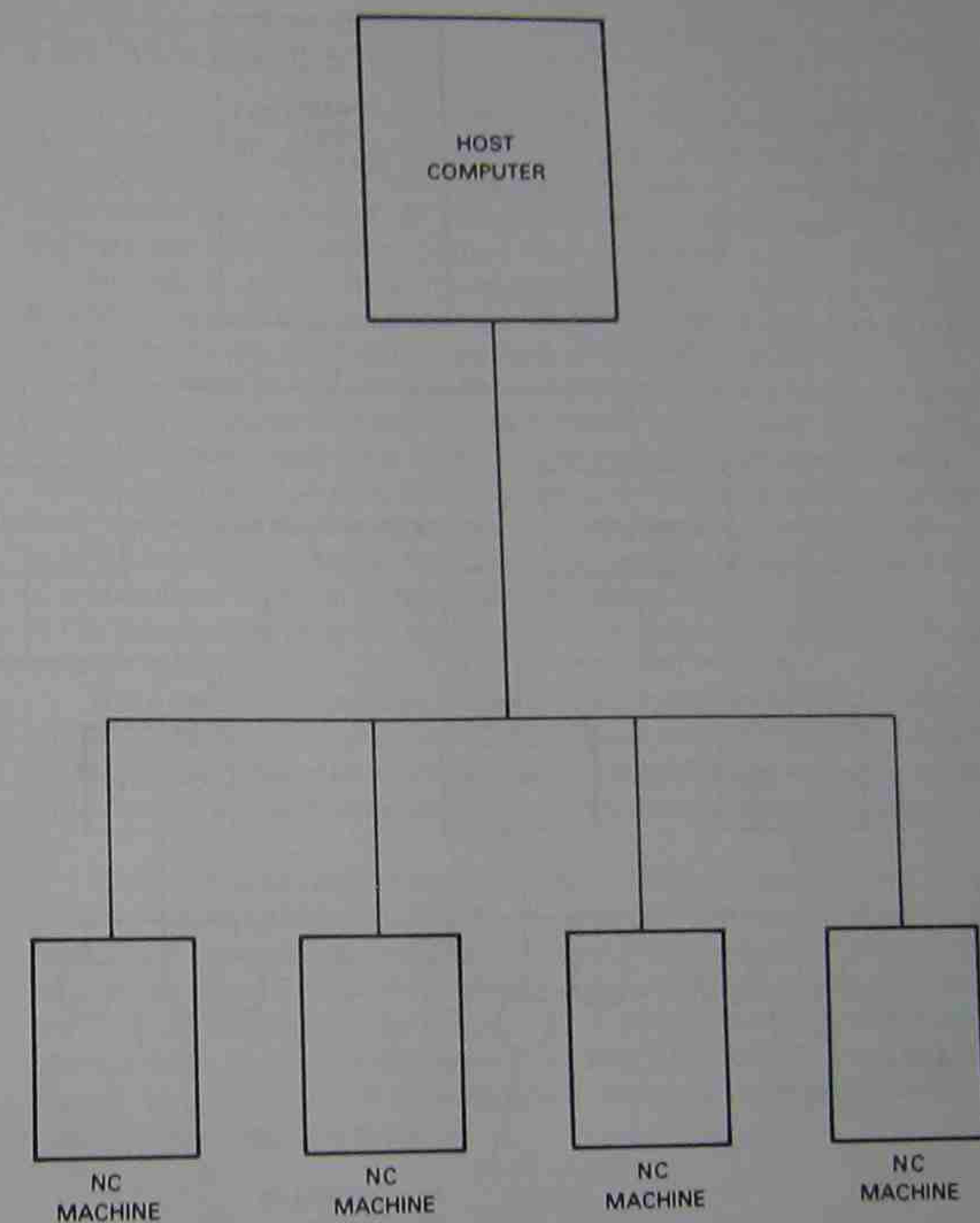


FIGURE 1–1
Direct numerical control

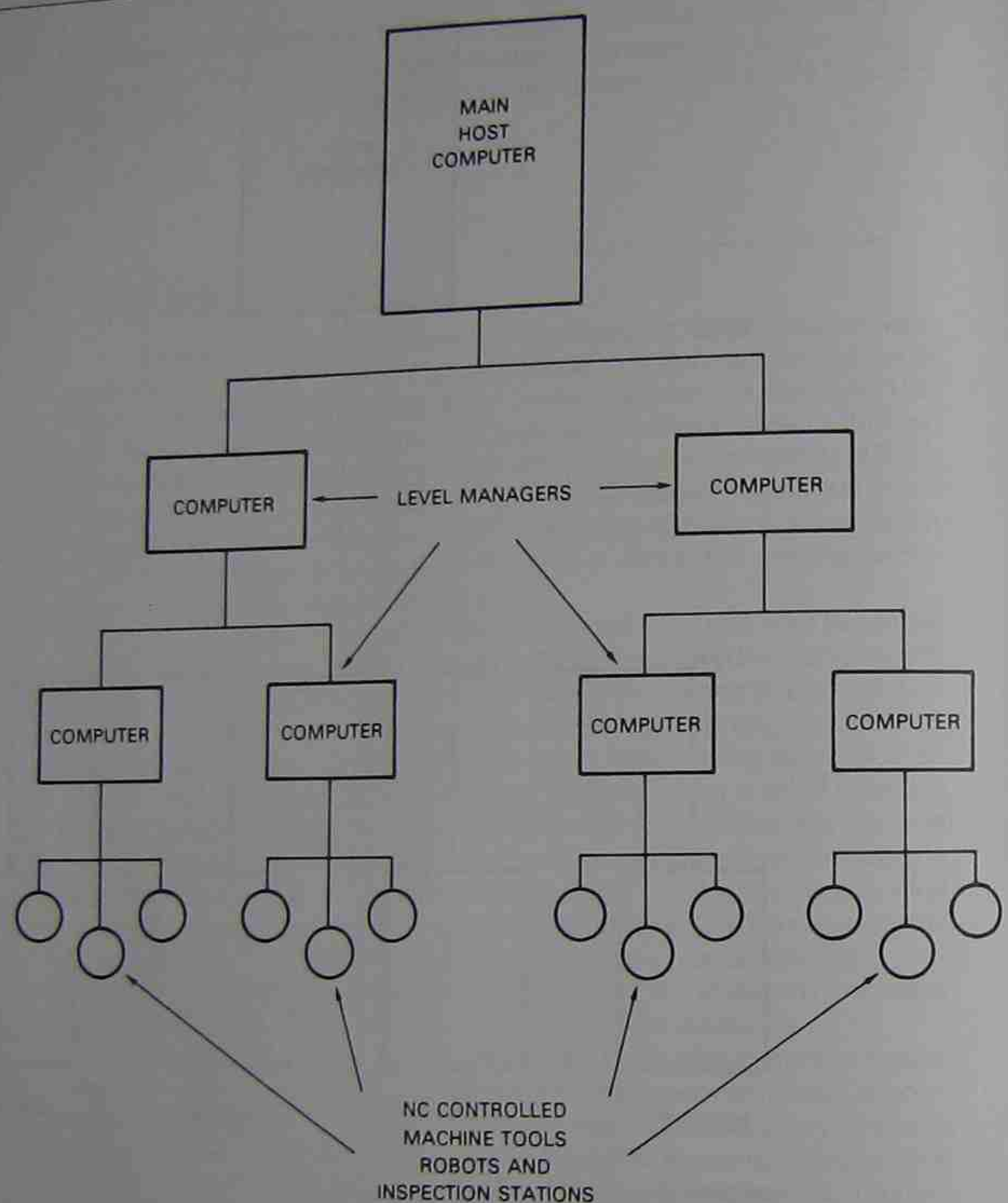


FIGURE 1–2
Distributive numerical control

CNC MACHINES

Figures 1–3 to 1–10 show modern CNC machines. CNC machines have more programmable features than older NC tape machinery and may be used as stand-alone units, or in a network of machines such as a flexible machining center (described in Chapter 16). They are easier to program, and most CNC machines may be programmed by more than one method.

All machines can be programmed via an on-board computer keyboard. In addition to the keyboard, there is a tape reader or electronic connector to allow the transfer of a program written elsewhere to the CNC machine.

A CNC machine is a *soft-wired controller*; that is, once the NC program is loaded into the computer's memory, no hardware is necessary to transfer the numerical control codes to the controller. The controller uses a permanent resident program, called an *executive program*, to process the codes into the electrical pulses that control the machine. The executive program is often referred to as the executive "software," but technically speaking, software is a misnomer. The executive program is more appropriately called *firmware*. In any CNC machine, the executive program resides in ROM memory, and the NC code resides in RAM memory.

ROM stands for *read only memory*. The information in ROM memory is written into the electronic chip and cannot be erased without special equipment. ROM can be accessed by the computer, but cannot be altered. This is why the executive program cannot be erased and is always active when the machine is on.

RAM stands for *random access memory*. RAM can be accessed and altered (written to) by the computer. The NC code is written into RAM by either the keyboard or an outside source. The contents of RAM are lost when the controller is turned off. Many CNC controllers utilize a battery backup system that powers the computer long enough for the program to be transferred (saved) to some storage media in the event of a power loss; other CNC controllers use a special type of RAM called *CMOS memory*, which retains its contents even when the power to the computer is turned off.

INPUT MEDIA

Input media are used to electronically or mechanically store the NC programs until they are needed. A program is simply read from the input medium when it is loaded into the machine. As mentioned previously, there are different methods of inputting the NC code into a controller. Whereas old NC machinery could only read programs from punched tape or a direct numerical control system, CNC machines may possess multiple means of program input.

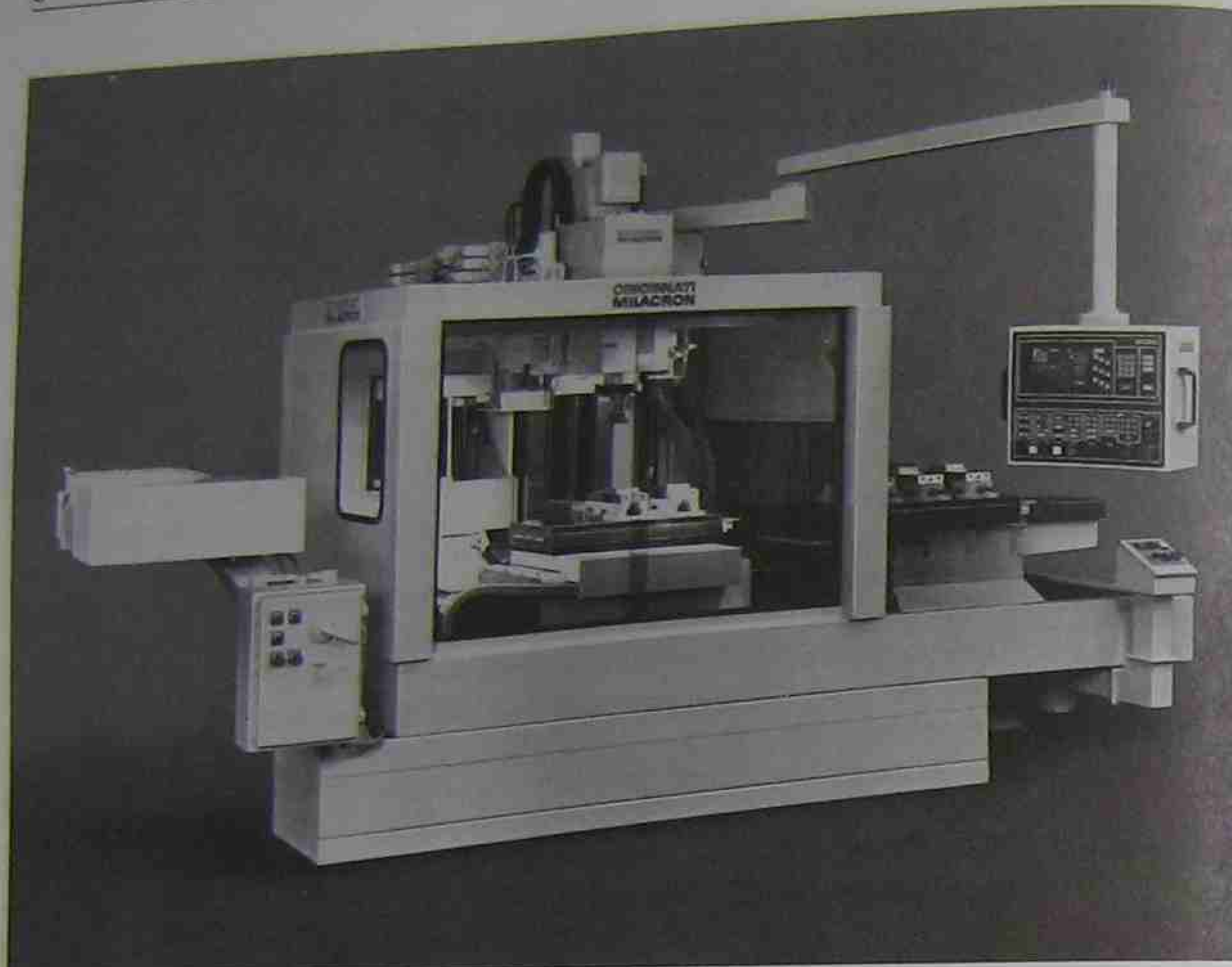


FIGURE 1–3
A vertical spindle machining center, featuring twin pallets *(Photo courtesy of Cincinnati Milacron)*

The most popular medium for program storage is still *punched tape* made of paper or mylar plastic (mylar is most commonly used as it is stronger than paper and less likely to tear). The NC program code is entered into the tape by use of a *tape puncher* which punches a series of holes that represent the NC codes. A tape reader employing electrical, optical, or mechanical means senses the holes in the tape and transfers the coded information into the machine computer.

A tape puncher may be attached to a teletype machine or a Flexowriter (a typewriterlike machine). With either of these two pieces of equipment, a code character is punched into the tape as it is being typed onto a sheet of paper. It is more common, however, to see a tape puncher attached to a microcomputer. The NC code is entered into either a CAM (computer-aided manufacturing) or wordprocessor type of program and punched into the tape after all editing of the program is completed.

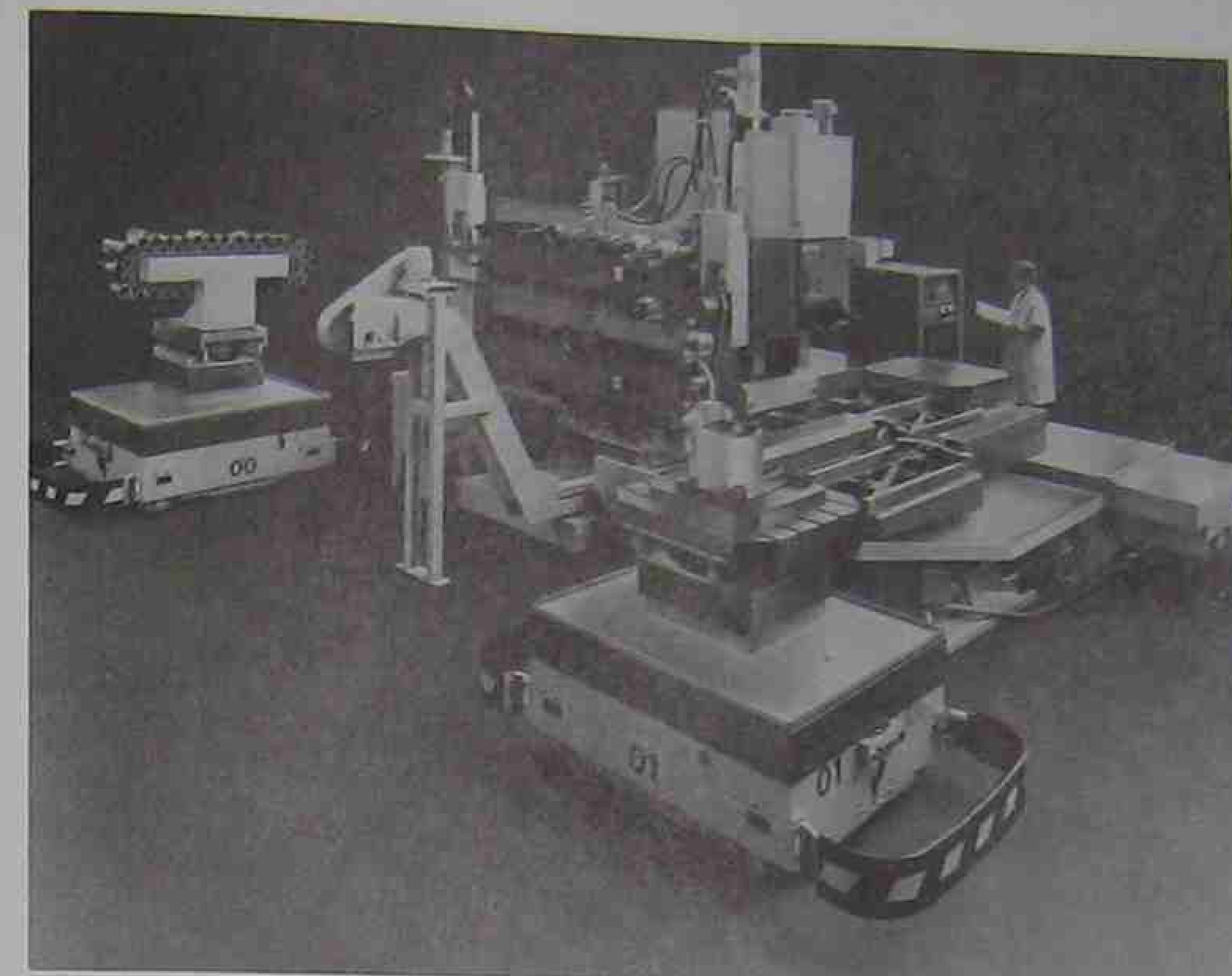


FIGURE 1–4
A horizontal machining center utilizing twin matrix tool storage magazines. Note the workpiece and tool delivery systems. Safety guards are removed to show clarity. *(Photo courtesy of Cincinnati Milacron)*

Magnetic tape is another popular storage medium. Early experiments with magnetic tape for program storage were not very successful, because the shop environment was not conducive to the delicate tapes of yesteryear. Today's high-quality tapes can survive the rigors of the shop environment with reasonable care in their handling. The most commonly used style of magnetic tape is ¼-inch computer grade cassette tape. The cassette case affords good protection and the small size is convenient for storage. Standards for tape format and tape coding have been developed by the Electronics Industries Association (EIA).

BINARY NUMBERS

An understanding of how the controller processes information is helpful in learning to program computer numerical control machinery. Computers and computer-controlled machinery do not deal in Arabic symbols or numbers. All of

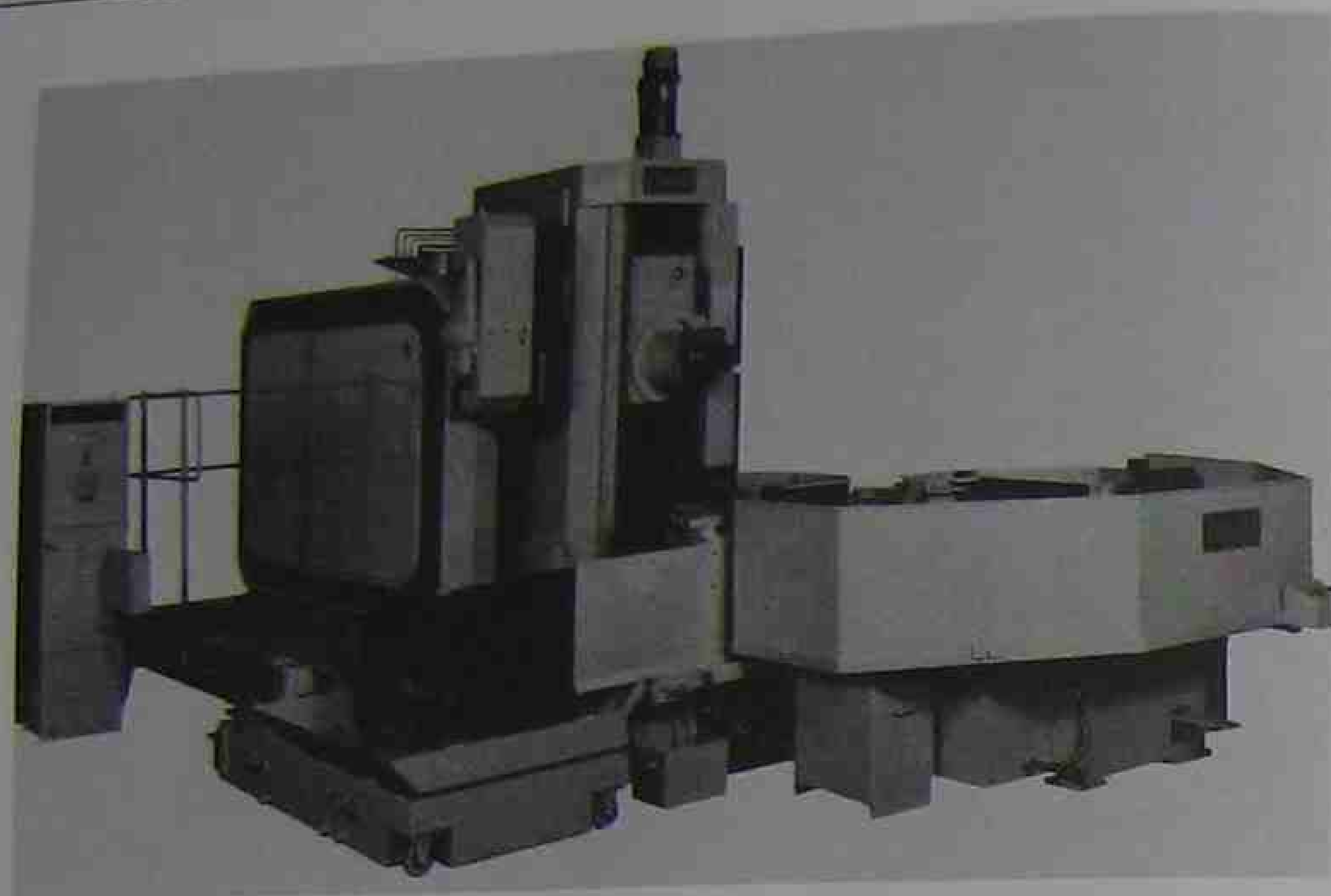


FIGURE 1–5
A horizontal machining center featuring a Fanuc controller *(Photo courtesy of Bendix Corp.)*

the internal processing is done by calculating or comparing binary numbers. Binary numbers contain only two digits, zero and one, as illustrated in Figure 1–11.

Within the CNC controller, each binary digit "one" may represent a positive charge and a "zero" a negative charge, or a "one" may be the presence of charge and a "zero" the absence of charge. The method used depends on the particular controller. In either case, the CNC program code in binary form must be loaded into the computer. Programming formats and languages allow the NC code to be written using alphabetic characters and base-ten decimal numbers. When the NC program is punched or recorded on tape, the information is translated into binary form.

TAPE FORMATS

Various tape coding formats have been developed and tried since the beginning of numerical control. Today, tapes are primarily made using the EIA standard RS-274 format, also called *word address format.* Program information is contained in program lines, called *blocks,* which are punched into the tape in one of two tape code standards. RS-274 is a *variable* block coding for-

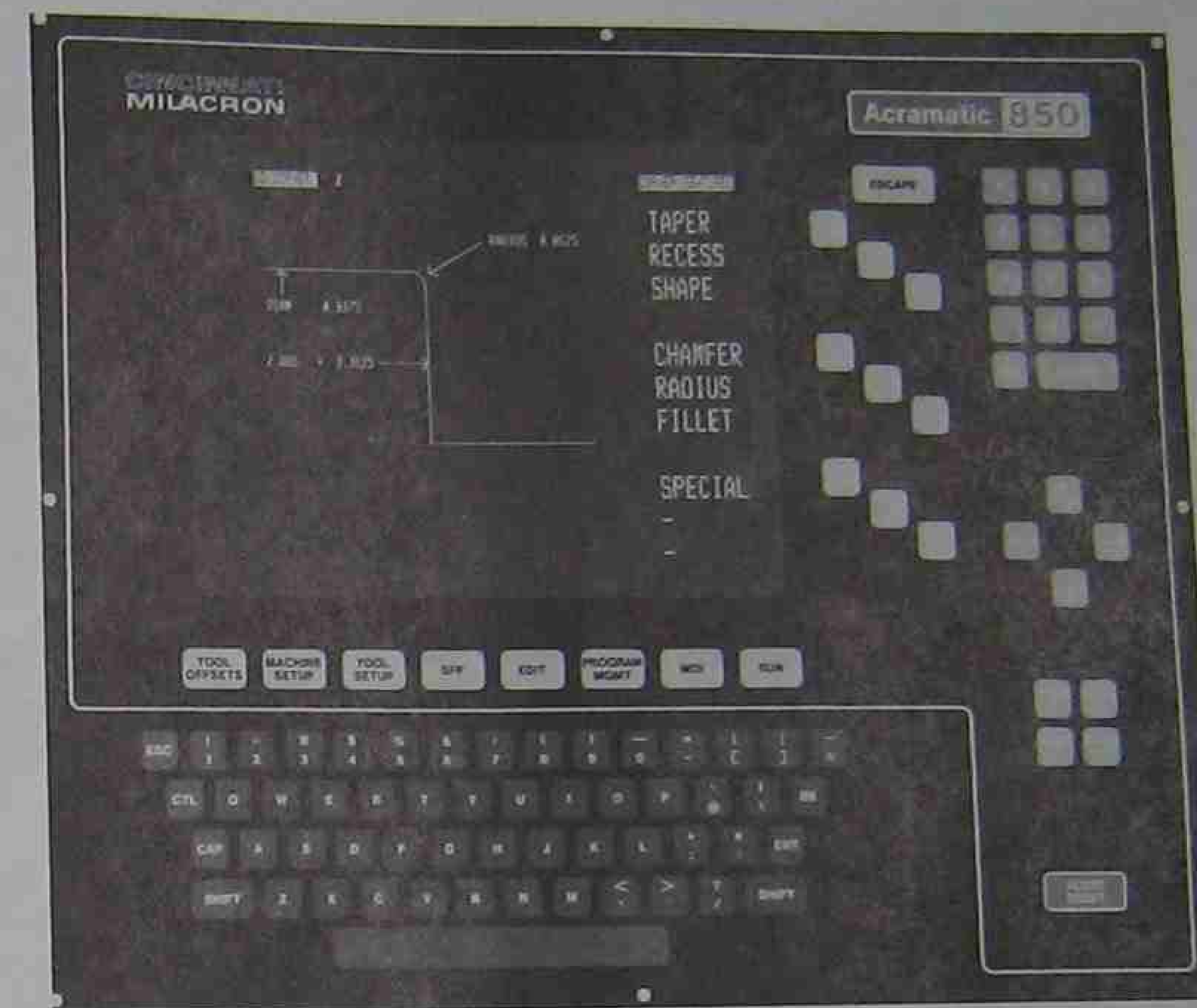


FIGURE 1–6
A CNC controller featuring interactive graphics *(Photo courtesy of Cincinnati Milacron)*

mat, meaning that the information contained in a block may be arranged in any order. Discussion of MDI programming using word address format begins in Chapter 6.

RS-244 Binary Coded Decimal

The EIA RS-244 standard, illustrated in Figure 1–12, is one of two tape codes used for NC tapes. It became a standard early in the development of numerical control. Notice that the code utilizes lowercase letters and limited punctuation. Each hole punched into the tape represents the binary digit "one," while a blank space represents the digit "zero." The tape code allows alphabetic characters and base-ten numbers to be translated into the binary code that the controller requires. For this reason, RS-244 is also known as *binary coded decimal* (BCD).

FIGURE 1–7
A modern vertical spindle machining center *(Photo courtesy of Bridgeport Machines Division of Textron Inc.)*

RS-358

At the time that NC was being implemented in industry, other industries were also using punched tape. Government, telephone, and computer industries all required a tape code that contained both upper and lowercase letters and more punctuation than the limited NC tape code used. What was adequate for numerical control use was not sufficient for other applications. The standard that was adopted for tape coding in these industries was the American Standard Code for Information Interchange (ASCII). To expand the role of computers in NC programming and strive for one standard tape code, EIA RS-358 was adopted for use. This code, known as both ISO (International Standards Organization) and ASCII, is a subset of the ASCII code used in other applications. It is illustrated in Figure 1–13. Both codes are used to prepare tape for use with CNC machines today. Many CNC controllers can detect which of the two formats is being sent and accept either one.

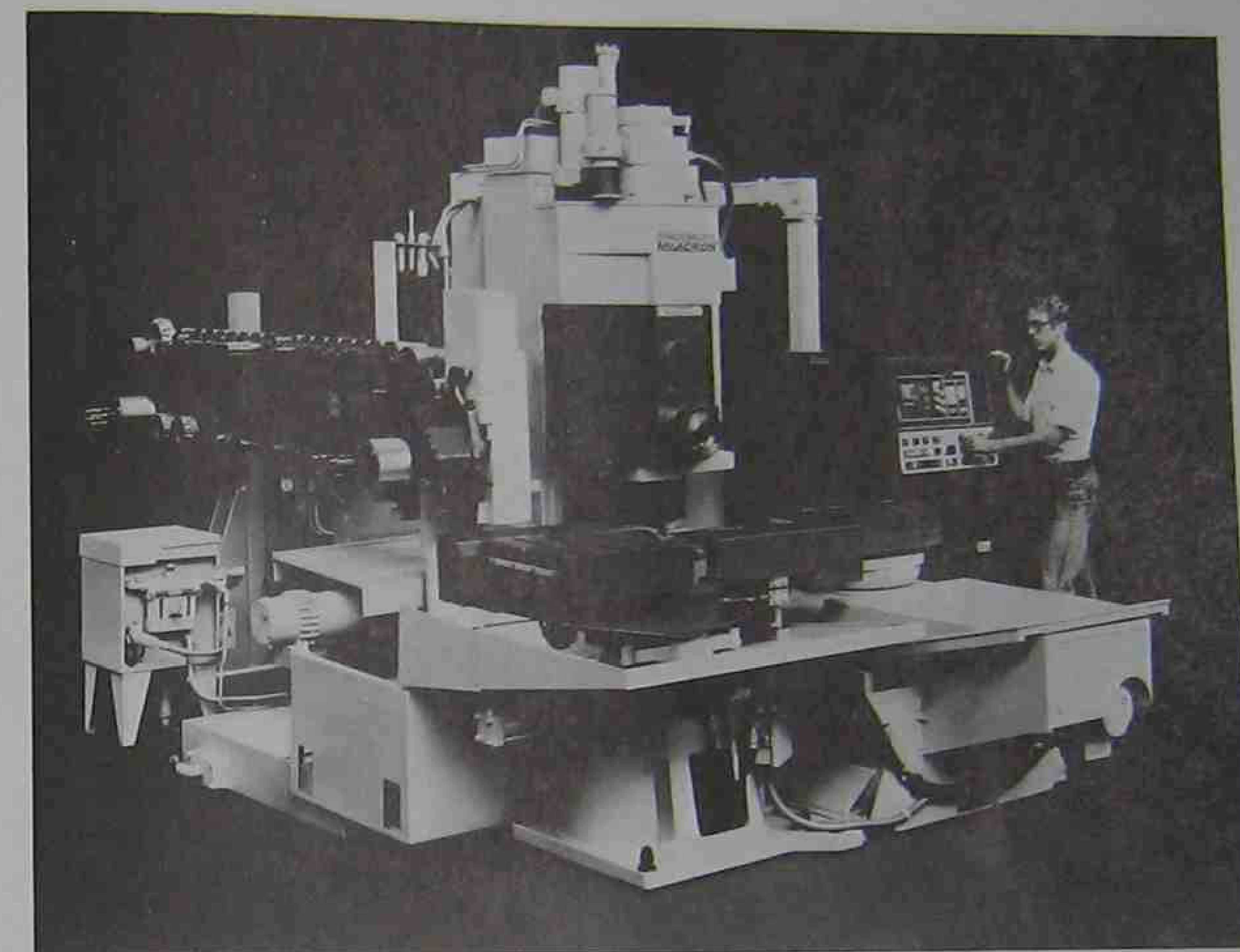


FIGURE 1–8
A modern horizontal spindle machining center *(Photo courtesy of Cincinnati Milacron)*

OBJECTIVES OF NUMERICAL CONTROL

Numerical control (NC) was developed with these goals in mind:

1. To increase production
2. To reduce labor costs
3. To make production more economical
4. To do jobs that would be impossible or impractical without NC
5. To increase the accuracy of duplicate parts

Before deciding (in light of NC objectives) to utilize an NC or CNC machine for a particular job, the requirements and economics of the job must be weighed against the following advantages and disadvantages of the machinery. Such an

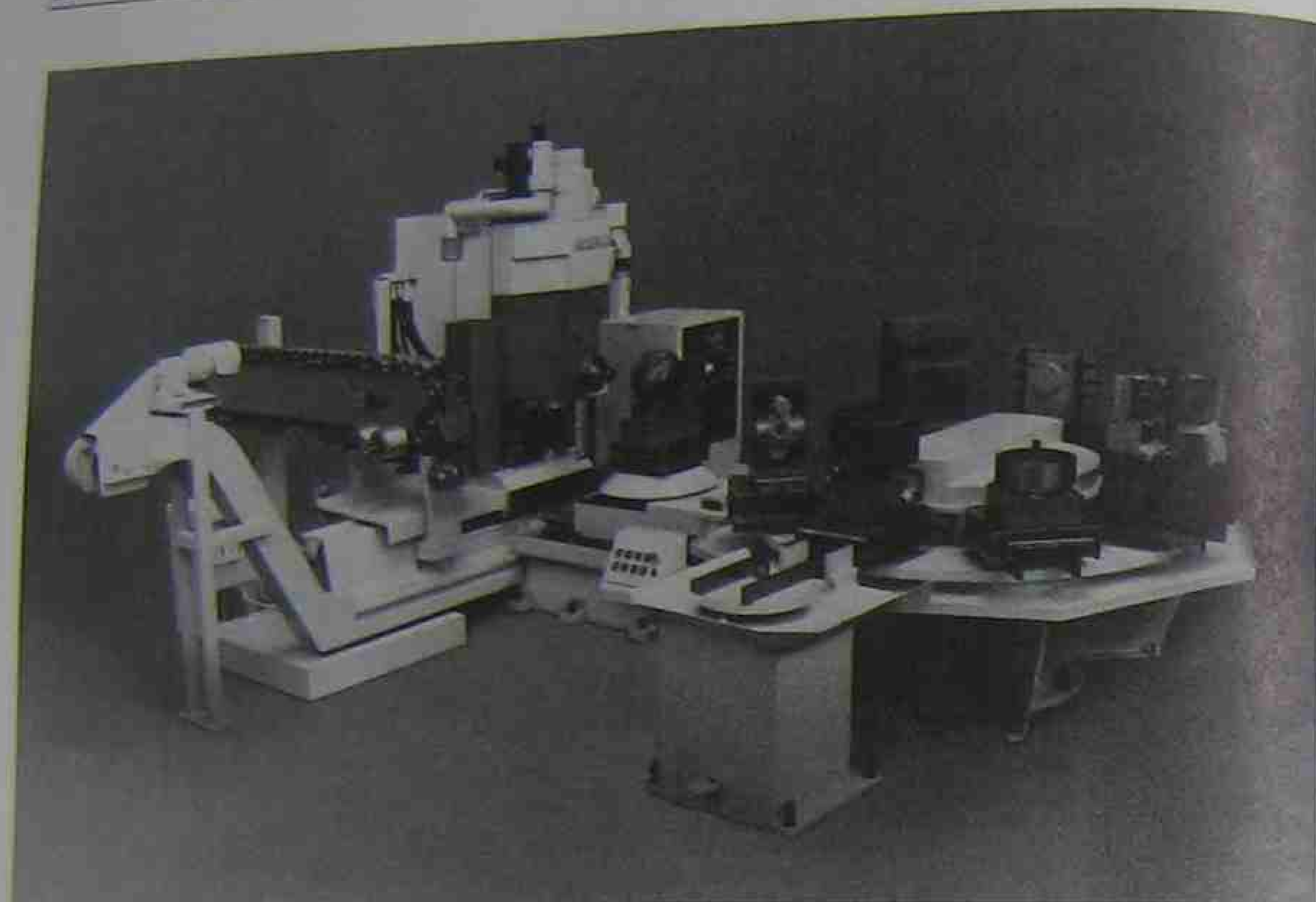


FIGURE 1–9
A horizontal machining center equipped with an eight-pallet automatic workchanger. Safety guards have been removed for clarity. *(Photo courtesy of Cincinnati Milacron)*

evaluation is necessary to determine if such a machine is practical for the particular job. (Note: NC is a general term used for numerical control. It is also used to describe numerical control machinery that runs directly off of tape. CNC refers specifically to computer numerical control. CNC machines are all NC machines, but not all NC machines are CNC machines.)

Advantages

1. Increased productivity
2. Reduced tool/fixture storage and cost
3. Faster set-up time
4. Reduced parts inventory
5. Flexibility that speeds changes in design
6. Better accuracy of parts
7. Reduction in parts handling
8. Better uniformity of parts
9. Better quality control
10. Improvement in manufacturing control

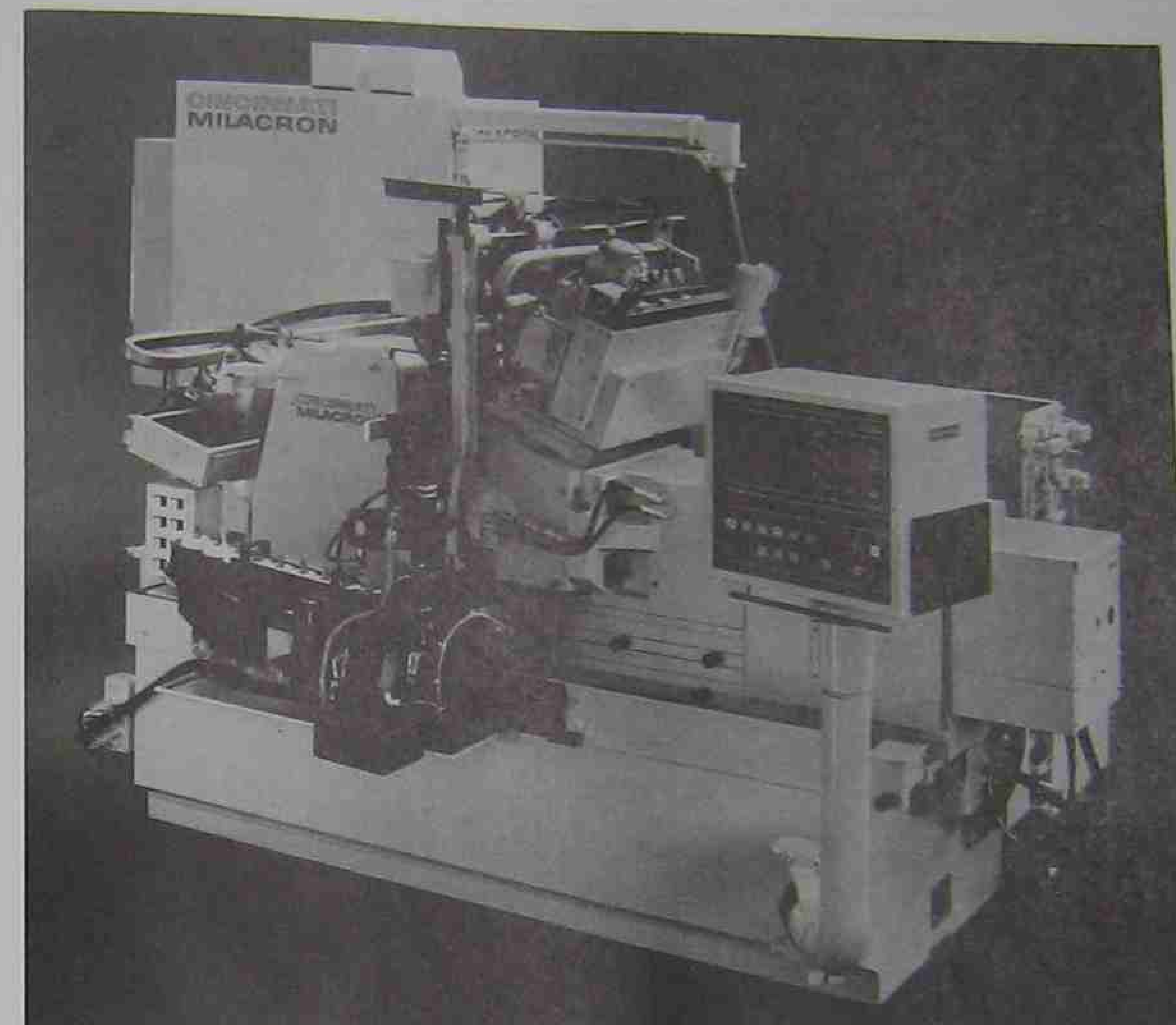


FIGURE 1–10
A CNC centerless grinding machine. This machine features an epoxy granite bed. Safety guards have been removed for clarity. *(Photo courtesy of Cincinnati Milacron)*

Disadvantages

1. Increase in electrical maintenance
2. High initial investment
3. Higher per-hour operating cost than traditional machine tools
4. Retraining of existing personnel

This is not a complete listing of the various advantages and disadvantages of numerical control machines; however, it should give a general idea of the types of jobs for which NC machines are suited.

ARABIC	BINARY	ARABIC	BINARY
0	0	18	10010
1	1	19	10011
2	10	20	10100
3	11	21	10101
4	100	22	10110
5	101	23	10111
6	110	24	11000
7	111	25	11001
8	1000	26	11010
9	1001	27	11011
10	1010	28	11100
11	1011	29	11101
12	1100	30	11110
13	1101	31	11111
14	1110	32	100000
15	1111	64	1000000
16	10000	128	10000000
17	10001		

FIGURE 1–11
Binary numbers compared to Arabic numbers

APPLICATIONS IN INDUSTRY

Developed originally for use in aerospace industries, NC is enjoying widespread acceptance in manufacturing. The use of CNC machines continues to increase, becoming visible in most metalworking and manufacturing industries. Aerospace, defense contract, automotive, electronic, appliance, and tooling industries all employ numerical control machinery. Advances in micro-electronics have lowered the cost of acquiring CNC equipment. It is not unusual to find CNC machinery in contract tool, die, and moldmaking shops. With the advent of low cost OEM (original equipment manufacturer) and retrofit CNC vertical milling machines, even shops specializing in one-of-a-kind prototype work are using CNCs.

Although numerical control machines traditionally have been machine tools, bending, forming, stamping, and inspection machines have also been produced as numerical control systems. Since this text is written with the student machinist in mind, only CNC machines will be considered.

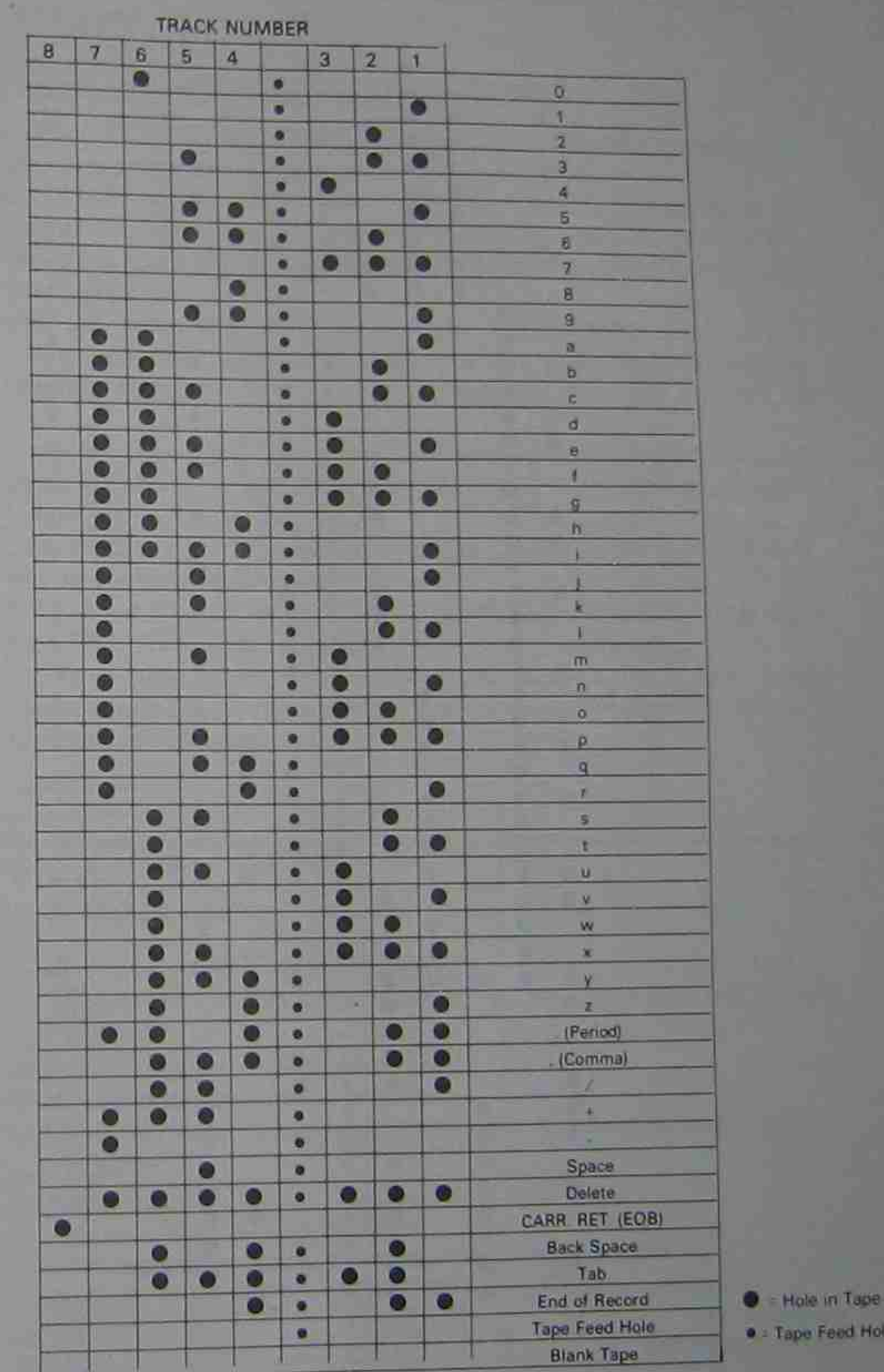


FIGURE 1–12
EIA RS-244 tape code

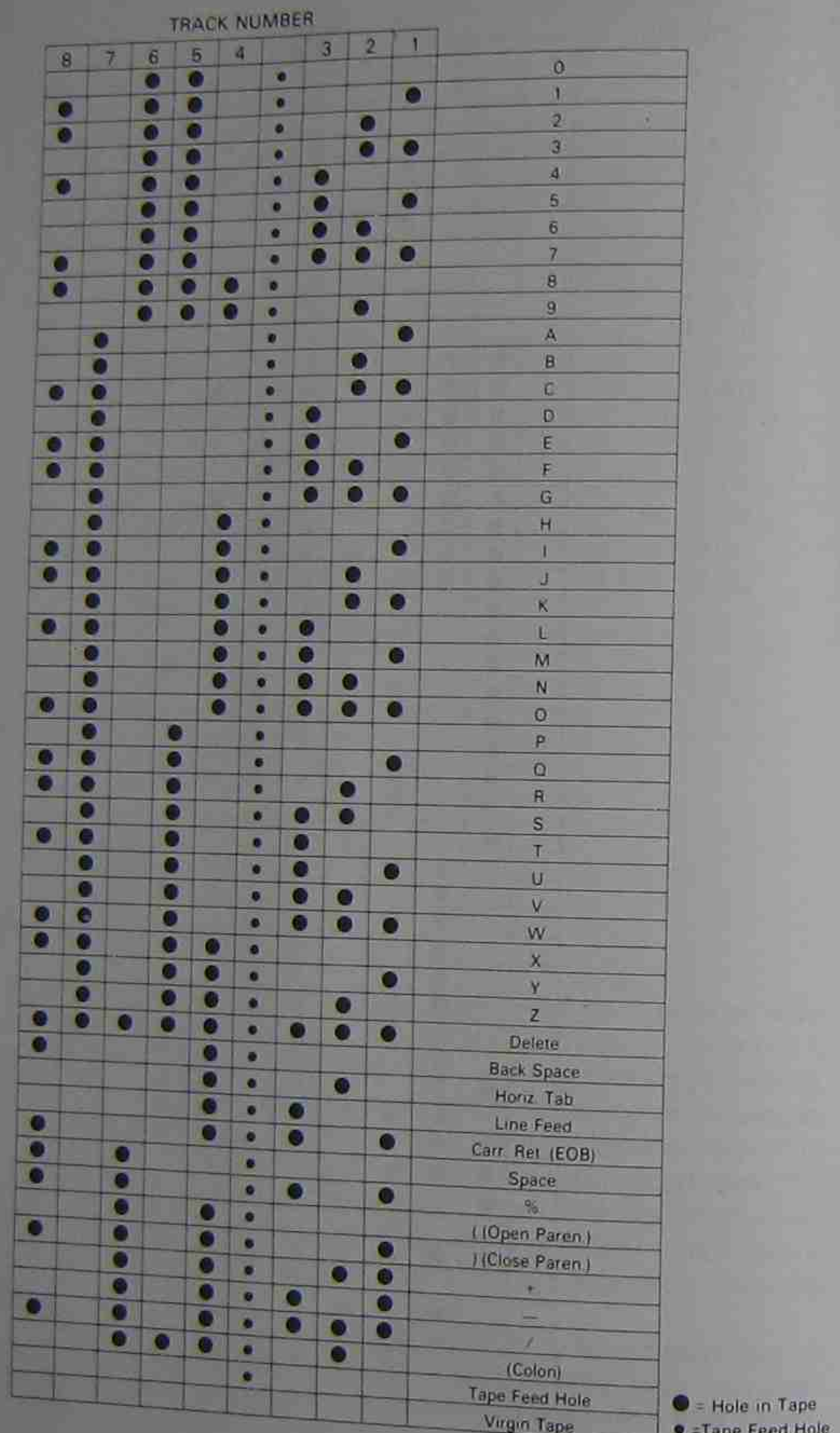


FIGURE 1–13
EIA RS-358 tape code

SUMMARY

The important concepts presented in this chapter are:

- A numerical control machine is a machine that is positioned automatically along a preprogrammed path by way of coded instructions.
- Direct numerical control involves a computer that acts as a partial or full controller to one or more NC machines. Distributive numerical control is a network of computers and numerical control machinery coordinated to perform some task.
- CNC machines use an on-board computer as a controller.
- Offline programming is the programming of a part away from the computer keyboard (hence the term *offline*). This is usually done with a microcomputer.
- There are four ways to input programs into CNC machinery: MDI (manual data input), punched tape, magnetic tape, and DNC (direct numerical control/distributive numerical control).
- Computers work with binary numbers. The CNC program must be loaded into the controller in binary form.
- RS-244 and RS-358 are tape codes used to place information on punched tape. The information is punched into the tape in binary form.
- Before deciding on a numerical control machine for a specific job, the advantages and disadvantages of NC must be weighed in view of the primary objectives of numerical control.

REVIEW QUESTIONS

1. What is a numerical control machine?
2. What is the difference between an NC tape machine and a CNC machine?
3. What is direct numerical control?
4. What is distributive numerical control?
5. Name four ways to enter a program into a computer numerical control machine.
6. What does CNC mean? What does MDI mean?
7. What are RS-244 and RS-358?
8. What are the five major objectives of numerical control?
9. What are the advantages of NC? What are the disadvantages?

CHAPTER 2

Numerical Control Systems

OBJECTIVES Upon completion of this chapter, you will be able to:

- Describe the two types of control systems in use on NC equipment.
- Name the four types of drive motors used on NC machinery.
- Describe the two types of loop systems used.
- Describe the Cartesian coordinate system.
- Define a machine axis.
- Describe the motion directions on a three-axis milling machine.
- Describe the difference between absolute and incremental positioning.
- Describe the difference between datum and delta dimensioning.

COMPONENTS

A CNC machine consists of two major components: the *machine tool* and the *controller,* or *machine control unit* (MCU), which is an on-board computer. These components may or may not be manufactured by the same company. General Numeric, Fanuc, General Electric, Bendix, Cincinnati Milacron, and G & L Electronics are among those manufacturers of CNC controllers that supply units to makers of machine tools. Figure 2–1 shows a typical controller. Each controller is manufactured with a standard set of built-in codes. Other codes are added by the machine tool builders. For this reason, program codes vary somewhat from machine to machine. Every CNC machine, regardless of manufacturer, is a collection of systems coordinated by the controller.

TYPES OF CONTROL SYSTEMS

There are two types of control systems used on NC machines: *point-to-point systems* and *continuous-path systems. Point-to-point* machines move only in straight lines. They are limited in a practical sense to hole operations

FIGURE 2–1
A modern CNC controller *(Photo courtesy of Giddings & Lewis/Davis Corp.)*

(drilling, reaming, boring, etc.) and straight milling cuts parallel to a machine axis. When making an axis move, all affected drive motors run at the same speed. When one axis motor has moved the instructed amount, it stops while the other motor continues until its axis has reached its programmed location. This makes the cutting of 45-degree angles possible, but not arcs or angles other than 45 degrees. Angles and arc segments must be programmed as a series of straight line cuts (see Figure 2–2).

A *continuous-path* machine (or *contouring system*) has the ability to move its drive motors at varying rates of speed while positioning the machine; the

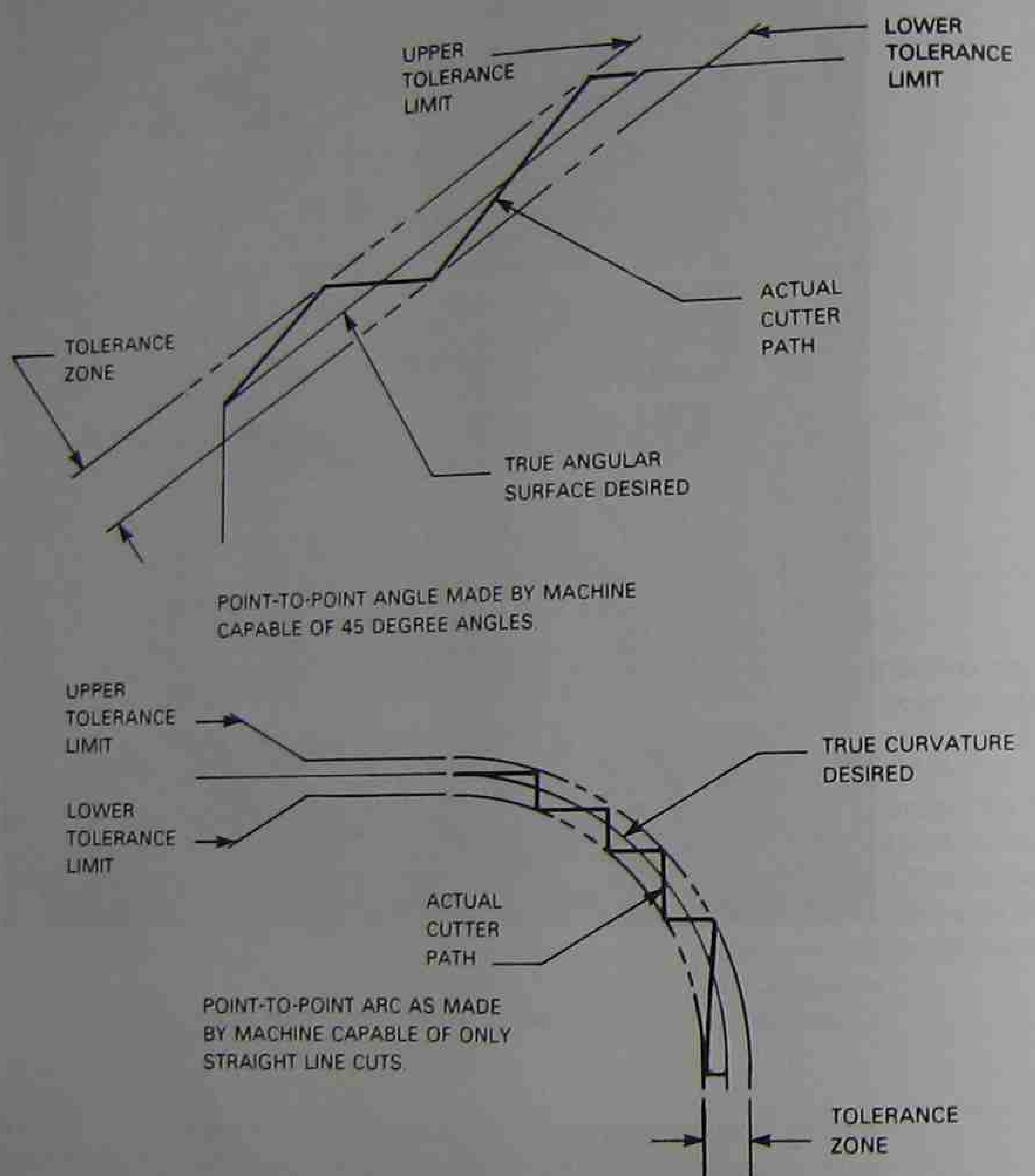


FIGURE 2–2
Point-to-point angles and arcs

cutting of arc segments and any angle may be easily accomplished (see Figure 2–3). At one time, point-to-point machines were common; their electronics were less expensive to produce and they were, therefore, less expensive to acquire. Technological advancements, however, have narrowed the cost difference between point-to-point and continuous-path machines to where most CNC machines now manufactured are of the continuous-path type.

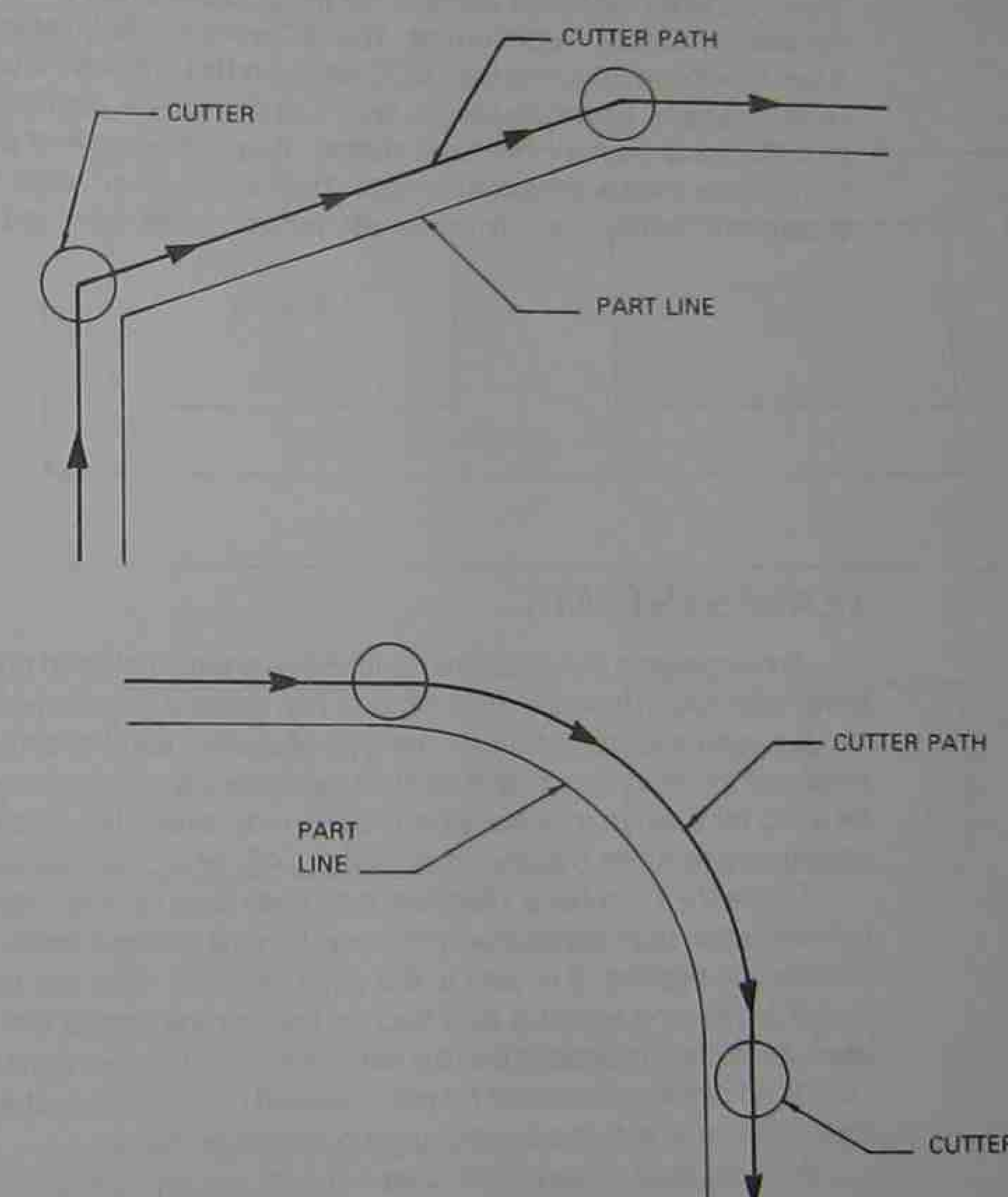


FIGURE 2–3
Continuous-path angles and arcs

SERVOMECHANISMS

It is helpful to understand the drive systems used on NC machinery. The *drive motors* on a particular machine will be one of four types: stepper motors, DC servos, AC servos, or hydraulic servos. Stepper motors move a set amount of rotation (a step) every time the motor receives an electronic pulse. DC and AC servos are widely used variable speed motors found on small and medium continuous-path machines. Unlike a stepper motor, a servo does not move a set distance; when current is applied, the motor starts to turn; when the current is removed, the motor stops turning. The AC servo is a fairly recent development. It can develop more power than a DC servo and is commonly found on CNC machining centers. Hydraulic servos, like AC or DC servos, are variable speed motors. Because they are hydraulic motors, they are capable of producing much more power than an electrical motor. They are used on large NC machinery, usually with an electronic or pneumatic control system attached.

LOOP SYSTEMS

Loop systems are electronic feedback systems that send and receive electronic information from the drive motors. Two types of loop systems are currently in use: *open* and *closed loop*. The type of system used affects the overall accuracy of the machine. This is valuable to know before selecting a machine to be used for a close tolerance part. Open loop systems use stepper motors; closed loop systems usually use hydraulic, AC, or DC servos.

Figure 2–4 is a block diagram of an open loop system. The machine gets its information from the reader and stores it in the storage device. When the information is needed, it is sent to the drive motor(s). After the motor has completed its move, a signal is sent back to the storage device that the move has been completed, indicating that the next instruction may be received. Notice that there is no process to correct for error induced by the drive system. (There is no such thing as a perfect positioning drive system or motor.)

A closed loop system block diagram is shown in Figure 2–5. As in an open loop system, the machine gets its information from the reading device and stores it in the storage device. When the information is sent to the drive motor(s), the position of the motor is monitored by the system and compared to what was sent. If an error is detected, the necessary correction is sent to the drive system.

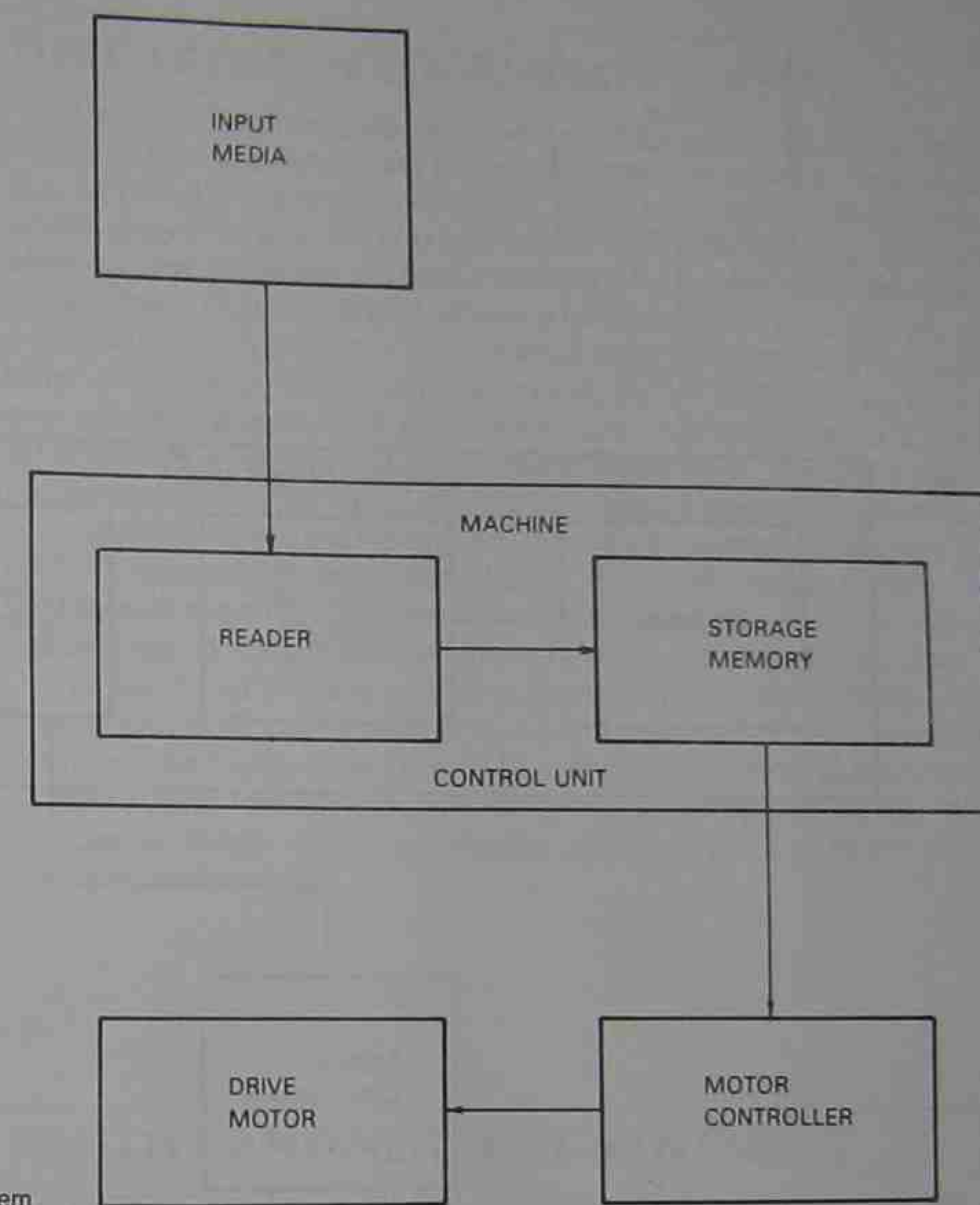


FIGURE 2–4
An open loop system

If the error is large, the machine may simply stop executing the program until the inaccuracy is corrected. This type of system eliminates most errors in position produced by the drive motors.

Recent advances in stepper motor technology have made the manufacture of extremely accurate open loop systems possible. These systems also eliminate the extra hardware and electronics required for closed loop systems.

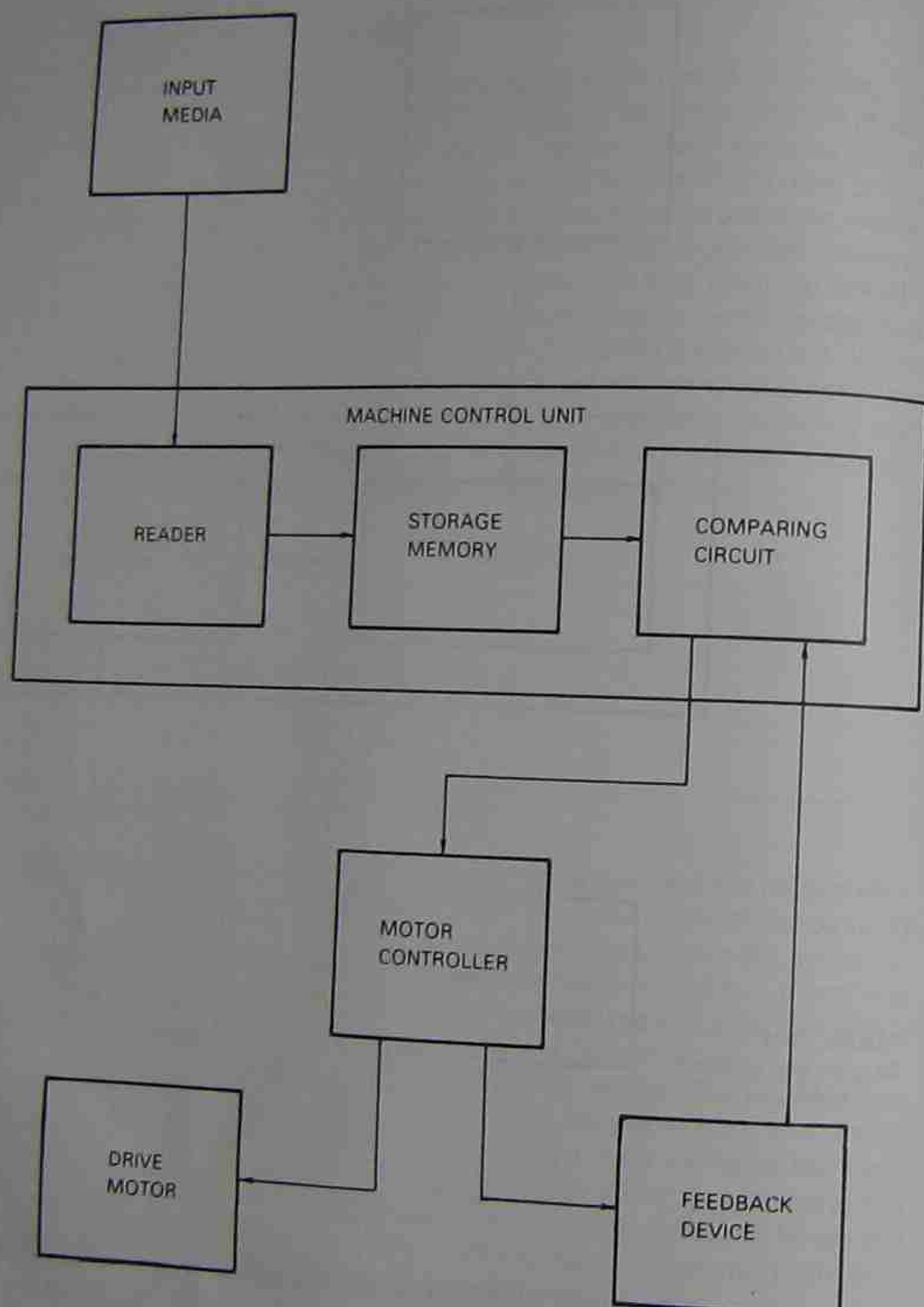


FIGURE 2–5
A closed loop system

THE CARTESIAN COORDINATE SYSTEM

The basis for all machine movement is the Cartesian coordinate system. Figure 2–6 illustrates two- and three-axis coordinate systems. On a machine tool, an *axis* is a direction of movement. The X and Y axes on the coordinate system shown in Figure 2–6(a) can be likened to a two-axis milling machine, where X is the direction of the table travel, and Y is the direction of the cross (or saddle) travel. Figure 2–6(b) illustrates a three-axis coordinate system. Using a vertical mill for example, X would be the table travel, Y the cross (saddle) travel, and Z the spindle travel (up and down). Figure 2–7 illustrates the three-axis system on a vertical mill. Machines are also available in four- and five-axis arrangements. A six-axis layout is shown in Figure 2–8. The milling machines programmed in this text will all use this EIA standard axis arrangement.

Cartesian coordinate systems are divided into quarters (quadrants). In Figure 2–9, the quadrants have been labeled I, II, III, and IV, respectively, in a counterclockwise direction. This is the universal way of labeling axis quadrants. Note that the signs of X and Y change when moving from quadrant to quadrant.

Figure 2–10 shows a number of points on a two-axis Cartesian system. Each of the points can be defined by a set of coordinates. The X-axis value is given first; the Y-axis value second. In mathematics this set of points is called an *ordered pair.* In numerical control programming, the points are referred to as *coordinates.* In later chapters, Cartesian coordinates will be used in writing numerical control programs.

POSITIVE AND NEGATIVE MOVEMENT

Machine axis direction is defined in terms of *spindle movement.* On some axes, the machine slides actually move; on other axes, the spindle travels. For purposes of standardization, the positive and negative direction of each axis is always defined as if the spindle did the traveling. The arrows in Figure 2–7 show the positive and negative direction of spindle movement along each axis. On a vertical mill, the table would move in the direction opposite to the sign indicated. For example, to make a move in the +X direction (spindle right), the table would move to the left. To make a move in the +Y direction (spindle toward the column), the saddle would move away from the column. The Z axis movement is always positive when the spindle moves toward the machine head and negative when it moves toward the workpiece.

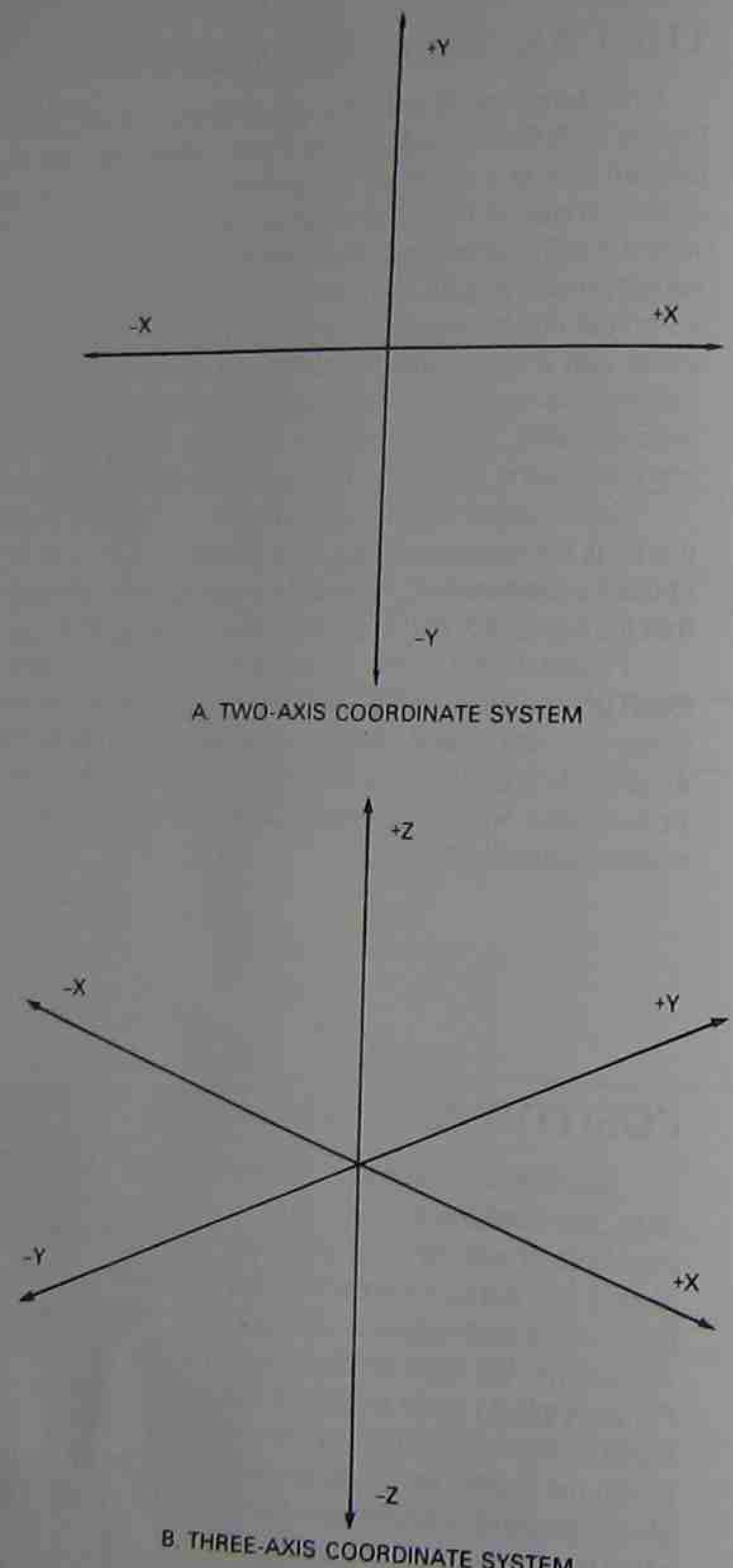


FIGURE 2-6
Cartesian coordinate system

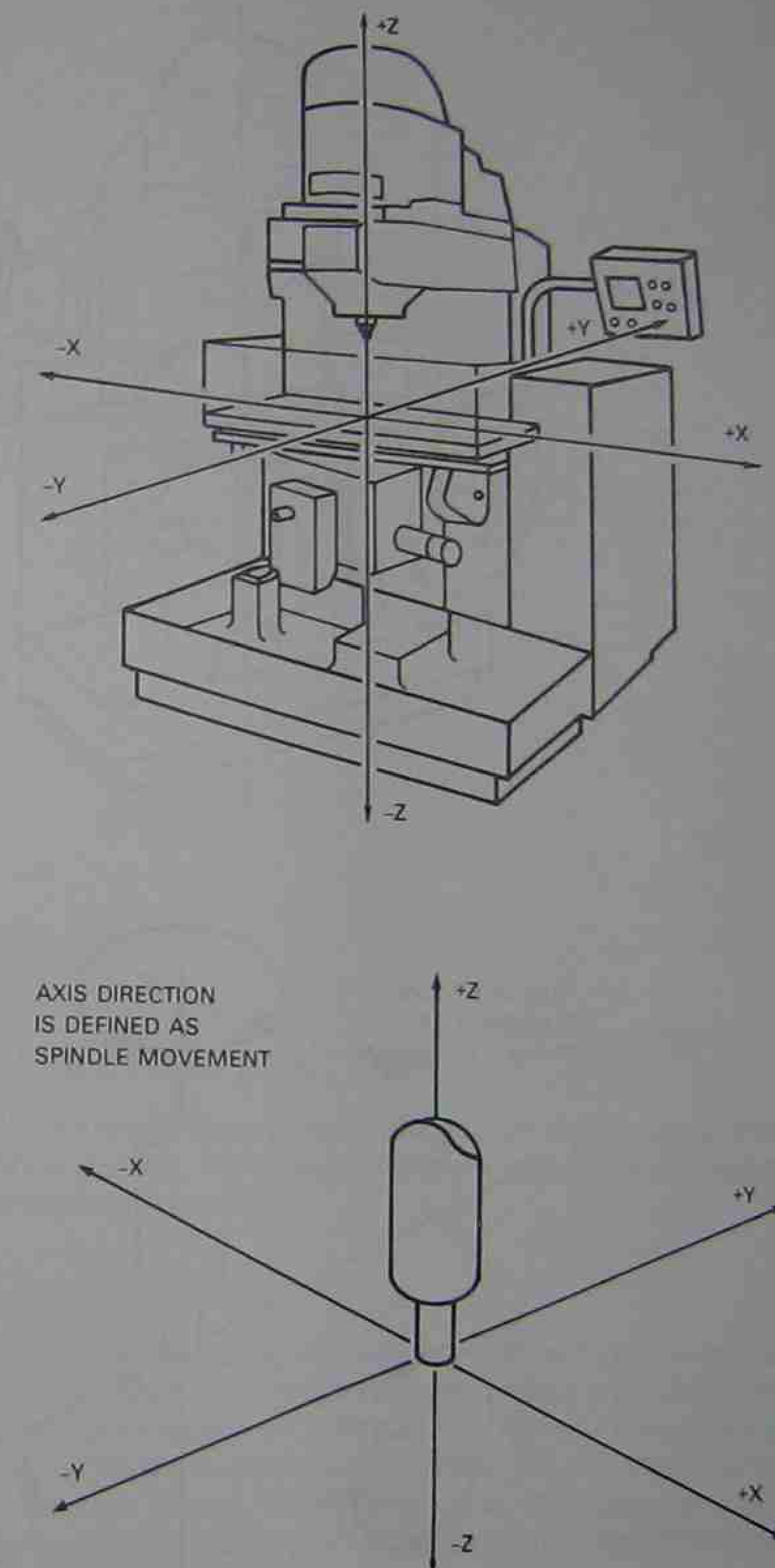


FIGURE 2-7
Three-axis vertical mill

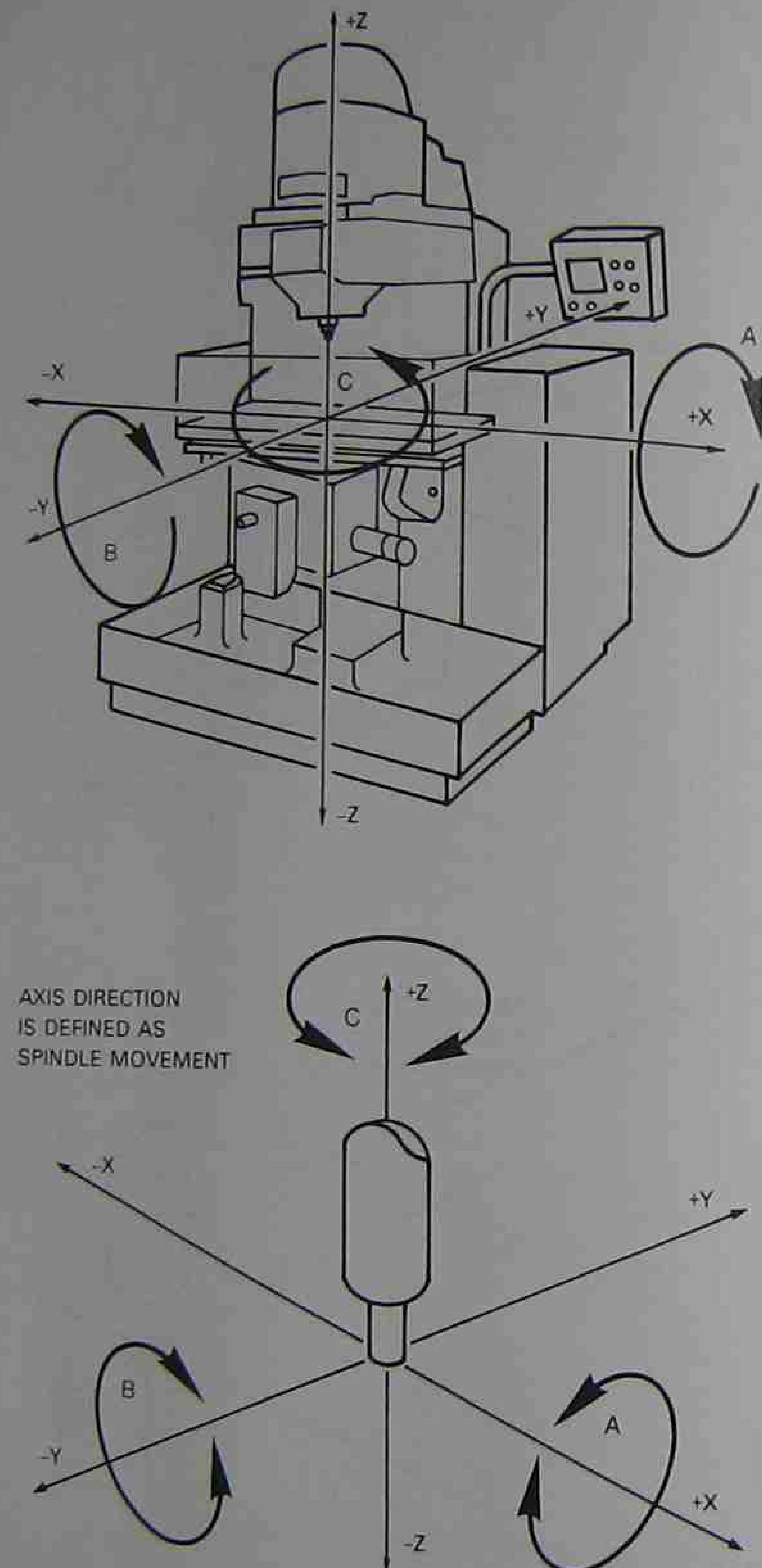


FIGURE 2–8
Six-axis machine layout

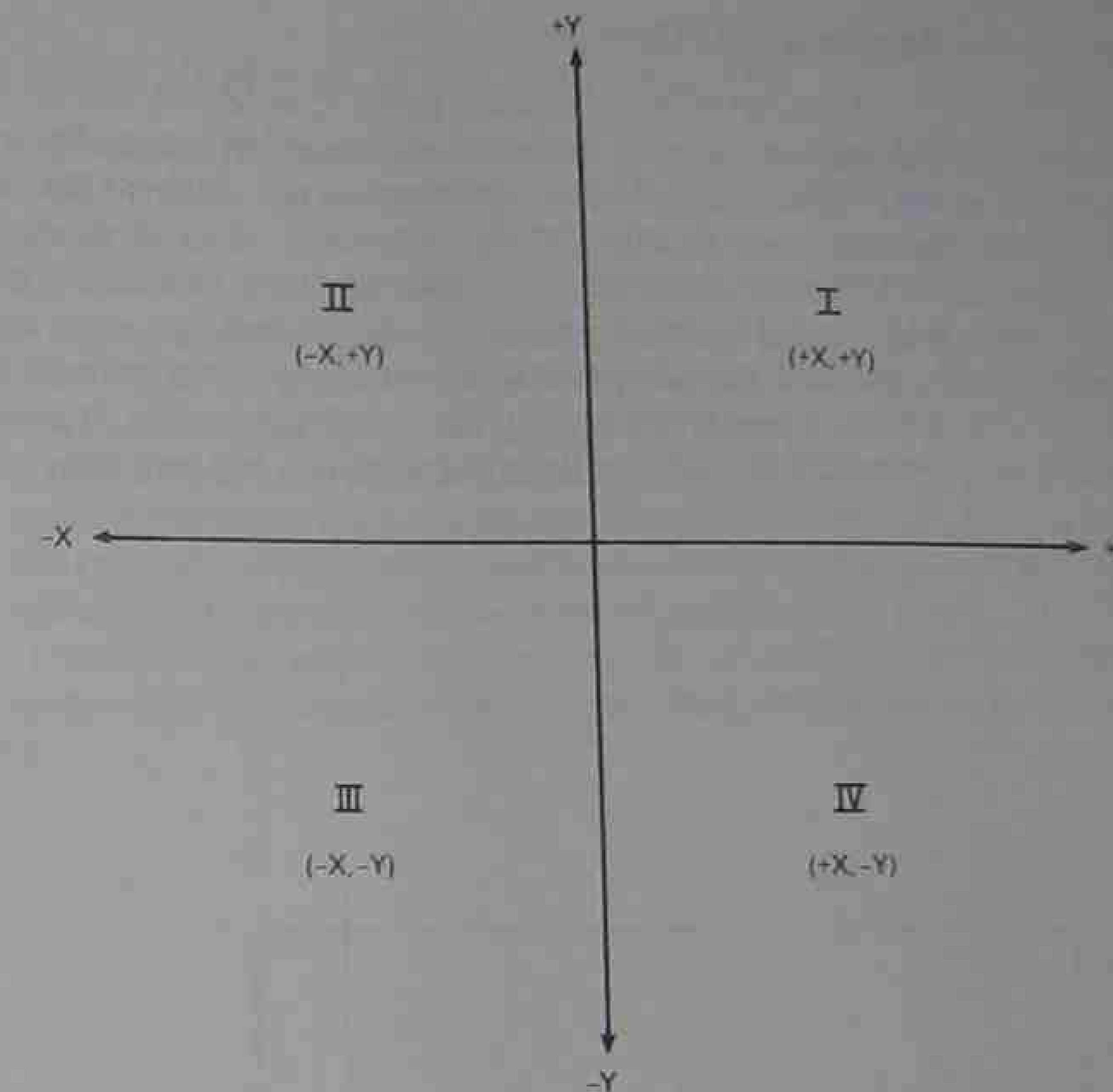


FIGURE 2–9
Cartesian coordinate quadrants

POSITIONING SYSTEMS

There are two ways that machines position themselves with respect to their coordinate systems. These two systems are called *absolute positioning* and *incremental positioning*.

Absolute Positioning

In absolute positioning (Figure 2–11), all machine locations are taken from one fixed zero point. Note that all positions on the part are taken from the X0/Y0 point at the lower left corner of the part. The first hole would have coordinates of X1.000, Y1.000; the second hole coordinates are X2.000, Y1.000; the third hole coordinates are X3.000, Y1.000. Every time the machine moves, the controller references the original zero point at the lower left corner of the part.

Incremental Positioning

In incremental positioning (see Figure 2–12) the X0/Y0 point moves with the machine spindle. Note that each position is specified in relation to the previous one. The first hole coordinates are X1.000, Y1.000; the second hole coordinates are X1.000, Y0.000. The third hole coordinates are again X1.000, Y0.000. After each machine move, the current location is reset to X0/Y0 for the next move. Figures 2–13 and 2–14 illustrate absolute and incremental positioning and their relationship to the Cartesian coordinate system. Notice that with incremental positioning, the coordinate system "moves" with the location. The machine controller does not reference any common zero point.

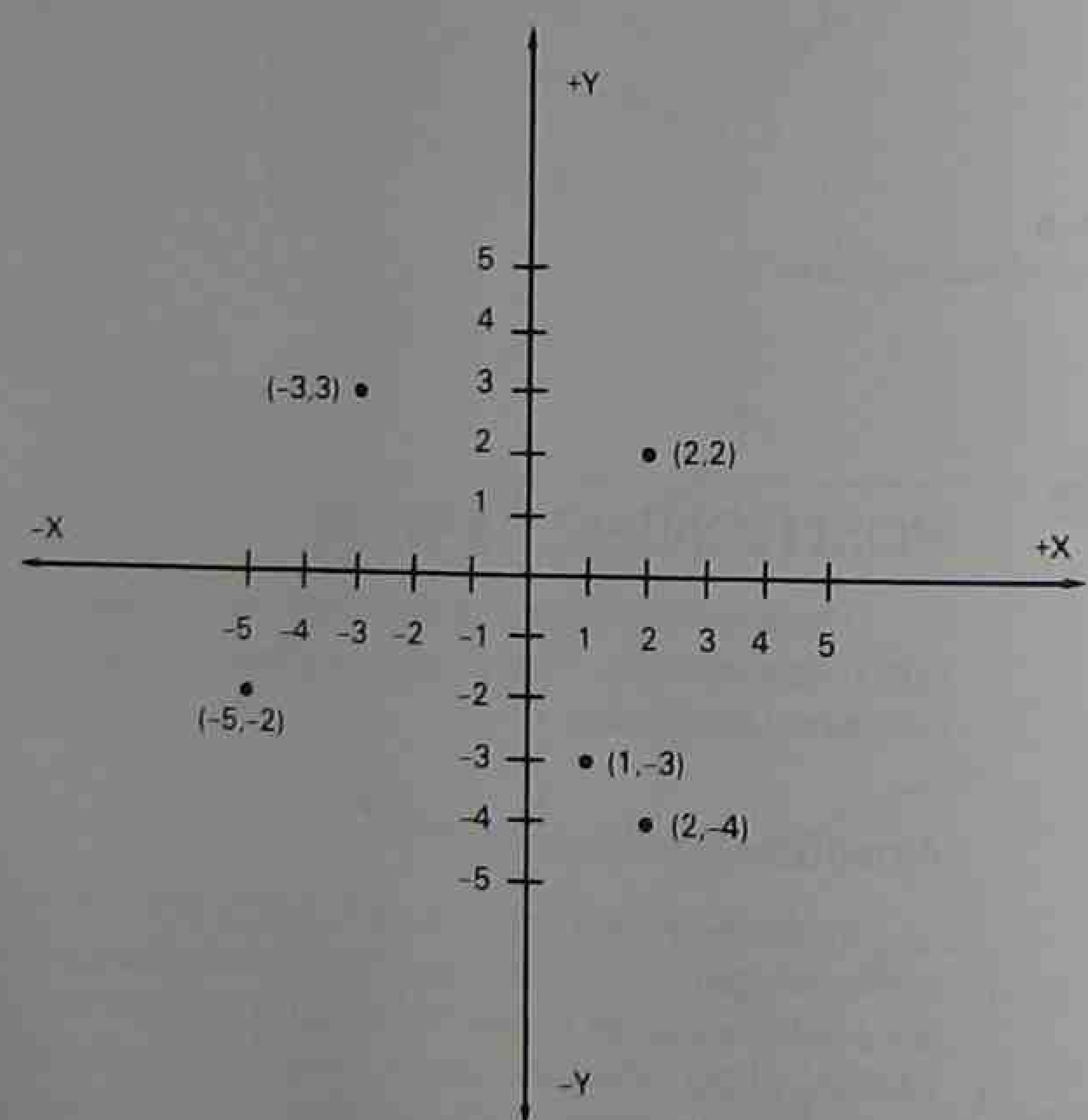


FIGURE 2–10
Cartesian coordinates

SETTING THE MACHINE ORIGIN

Most CNC machinery has a default coordinate system the machine assumes upon power-up, known as the *machine coordinate system*. The origin of this system is called the *machine origin* or *home zero location*. Home zero is usually, but not always, located at the tool change position of a machining center. A part is programmed independent of the machine coordinate system. The programmer will pick a location on the part or fixture. This location becomes the origin of the coordinate system for that part. The programmer's coordinate system is called the *local* or *part coordinate system*. The machine coordinate system and the part coordinate system will almost never coincide. Prior to running the part program, the coordinate system must be transferred from the machine system to the part system. This is known as *setting a zero point*.

There are three ways a zero point can be set on CNC machines: manually by the operator, by a programmed absolute zero shift, or by using work coordinates.

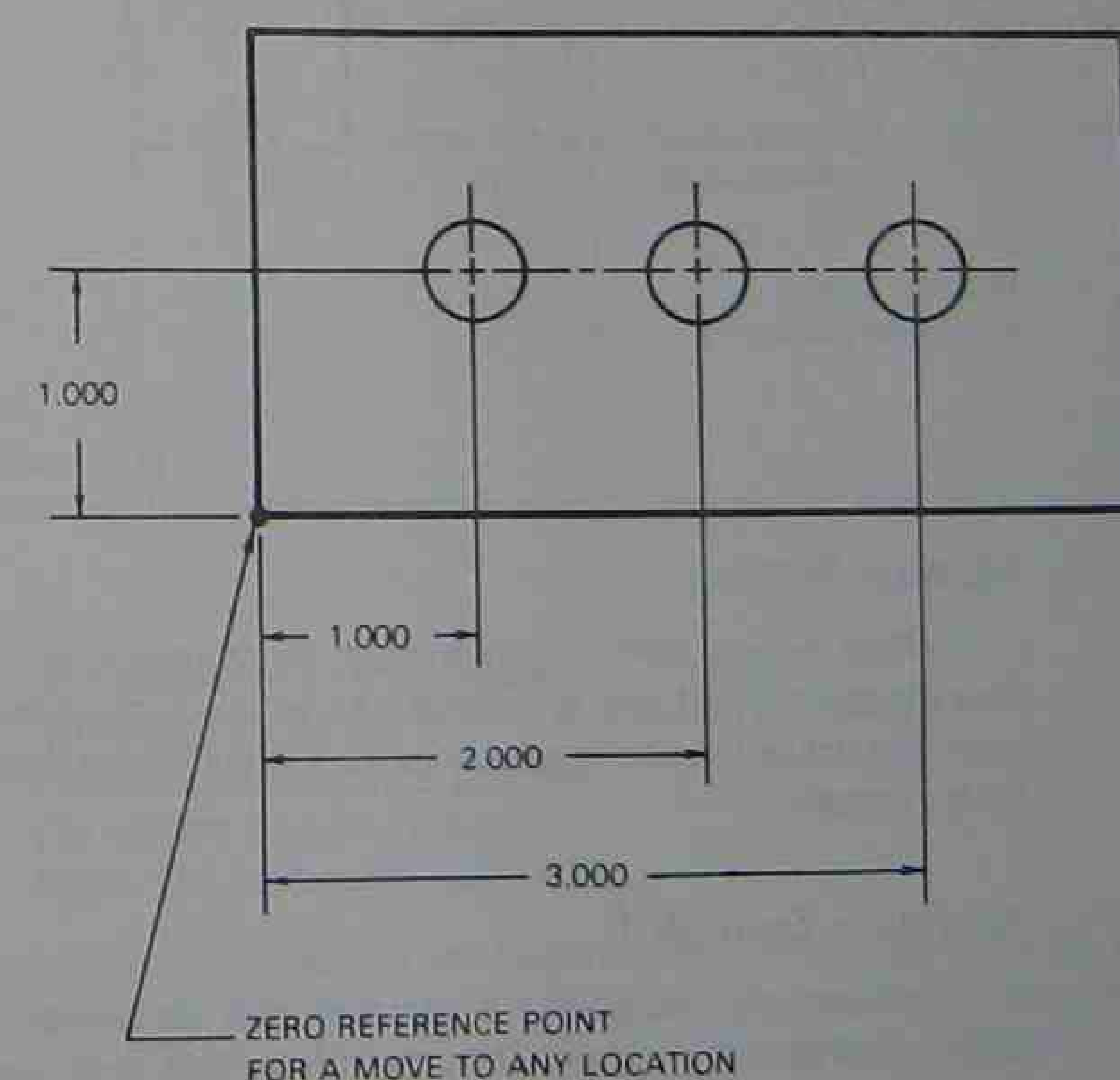


FIGURE 2–11
Absolute positioning

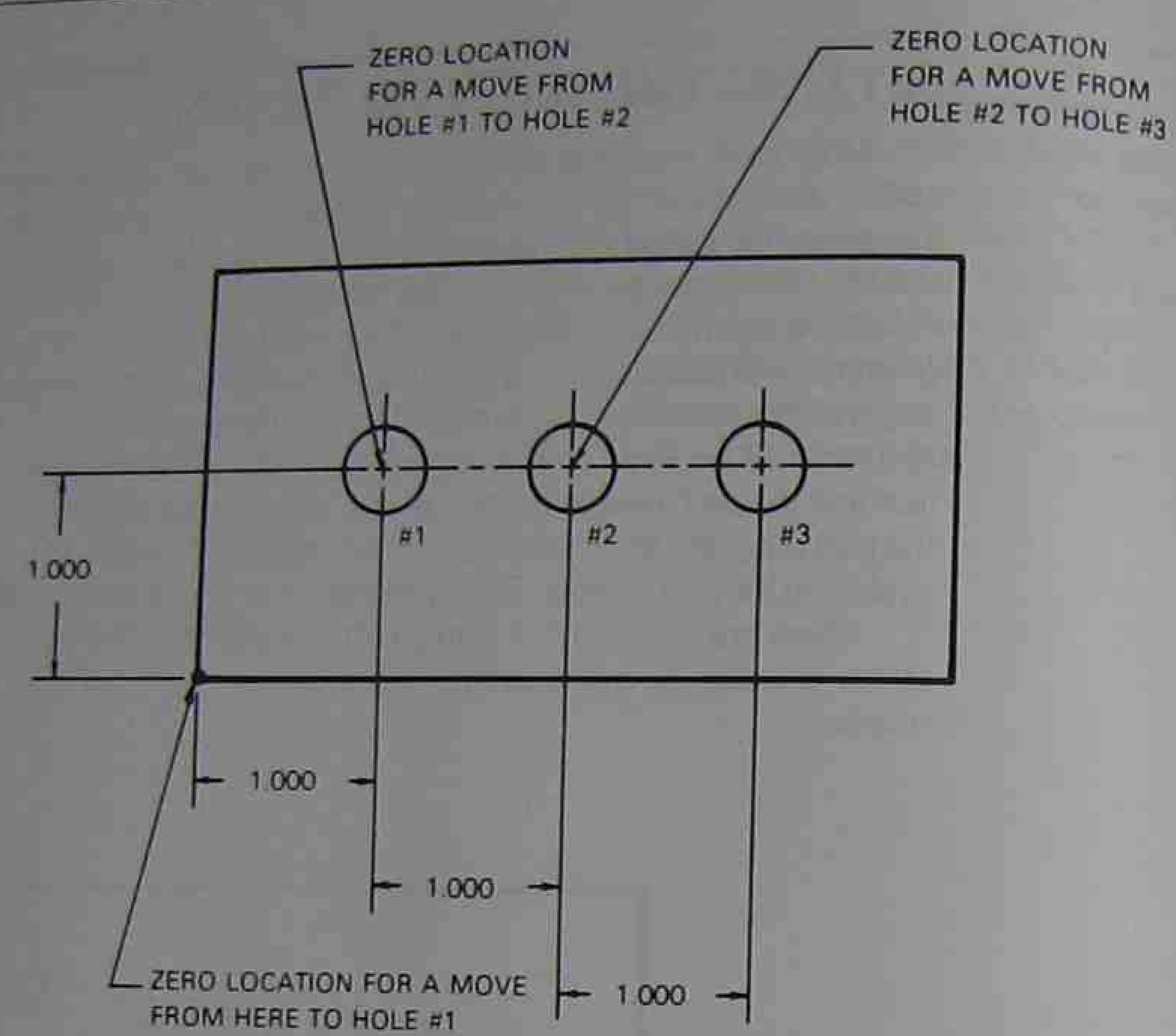


FIGURE 2–12
Incremental positioning

Manual Setting

When *manual zero setting* is used, the setup person positions the spindle over the desired part zero and zeros out the coordinate system on the MCU console. The actual keystroke sequence for accomplishing this varies from controller to controller.

Absolute Zero Shift

An *absolute zero shift* is a transferring of the coordinate system which is done inside the NC program. The programmer first commands the spindle to the home zero location. Next, a command is given that tells the MCU how far from the home zero location the coordinate system origin is to be located. An absolute zero shift is given as follows:

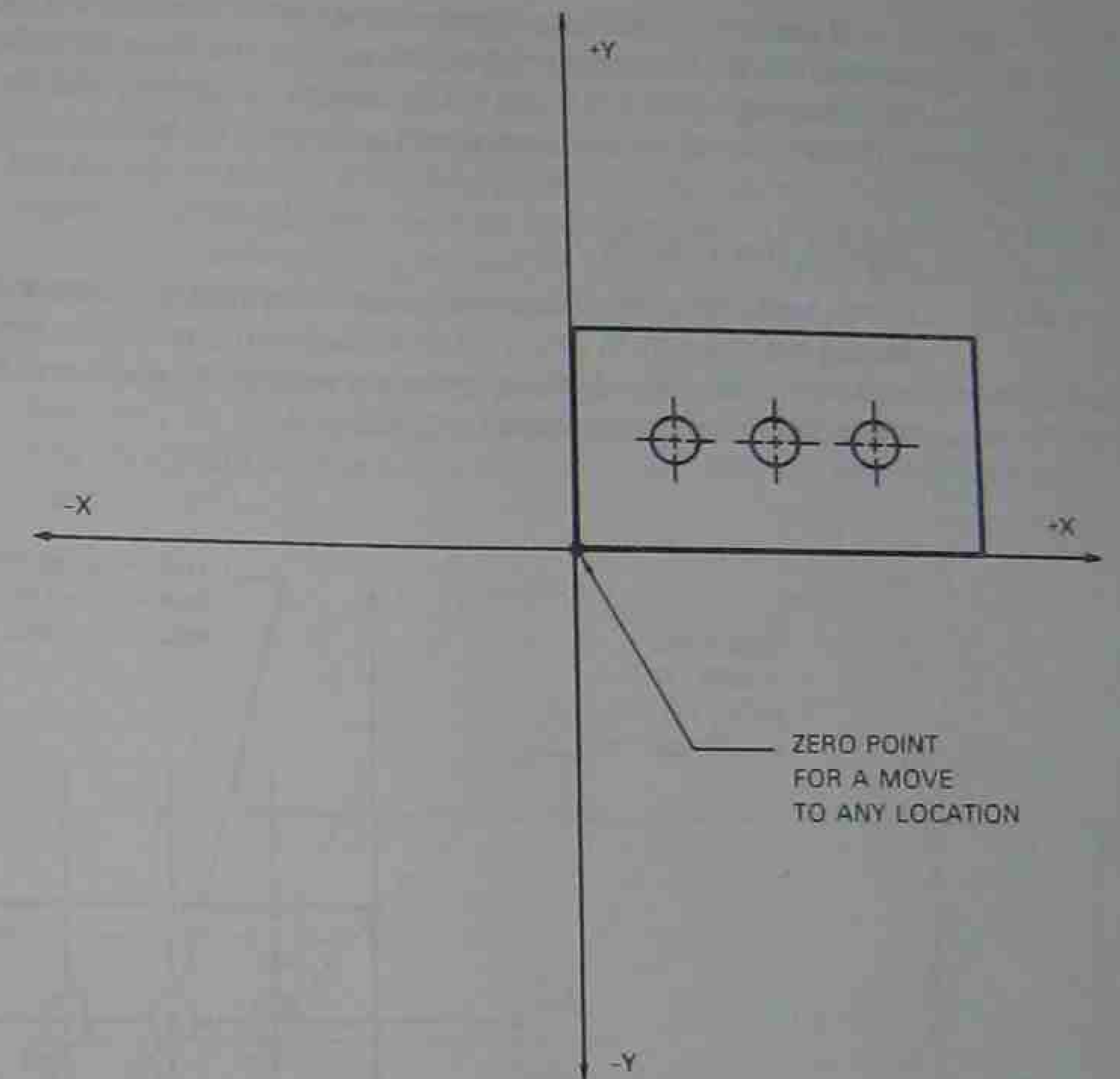


FIGURE 2–13
Relationship of the Cartesian coordinate system to the part when using absolute positioning

```
(Send the spindle to home zero)
N010 G28 X0 Y0 Z0
(Set the current spindle position)
(To X5.000 Y6.000 Z7.000)
N020 G92 X5.000 Y6.000 Z7.000
```

In line N010 the spindle moves to home zero. Following line N020, even though the spindle did not physically move from home zero, the location of the spindle became X5.0 Y6.0 Z7.0 as far as the MCU is concerned. The machine will now reference the part coordinate system. G92 is a fairly standard command for an absolute zero shift. The term G92 line is often used to describe an absolute zero shift.

If more than one fixture is to be used on a machine, the programmer will want to use more than one part coordinate system. By sending the spindle back to home zero using a G28 X0 Y0 Z0 command, another G92 line can be used in the program to set the second part coordinate system.

Work Coordinates

A *work coordinate* is a modification of the absolute zero shift. Work coordinates are registers in which the distance from home zero to the part zero can be stored. The part coordinate system does not take effect until the work coordinate is commanded in the NC program.

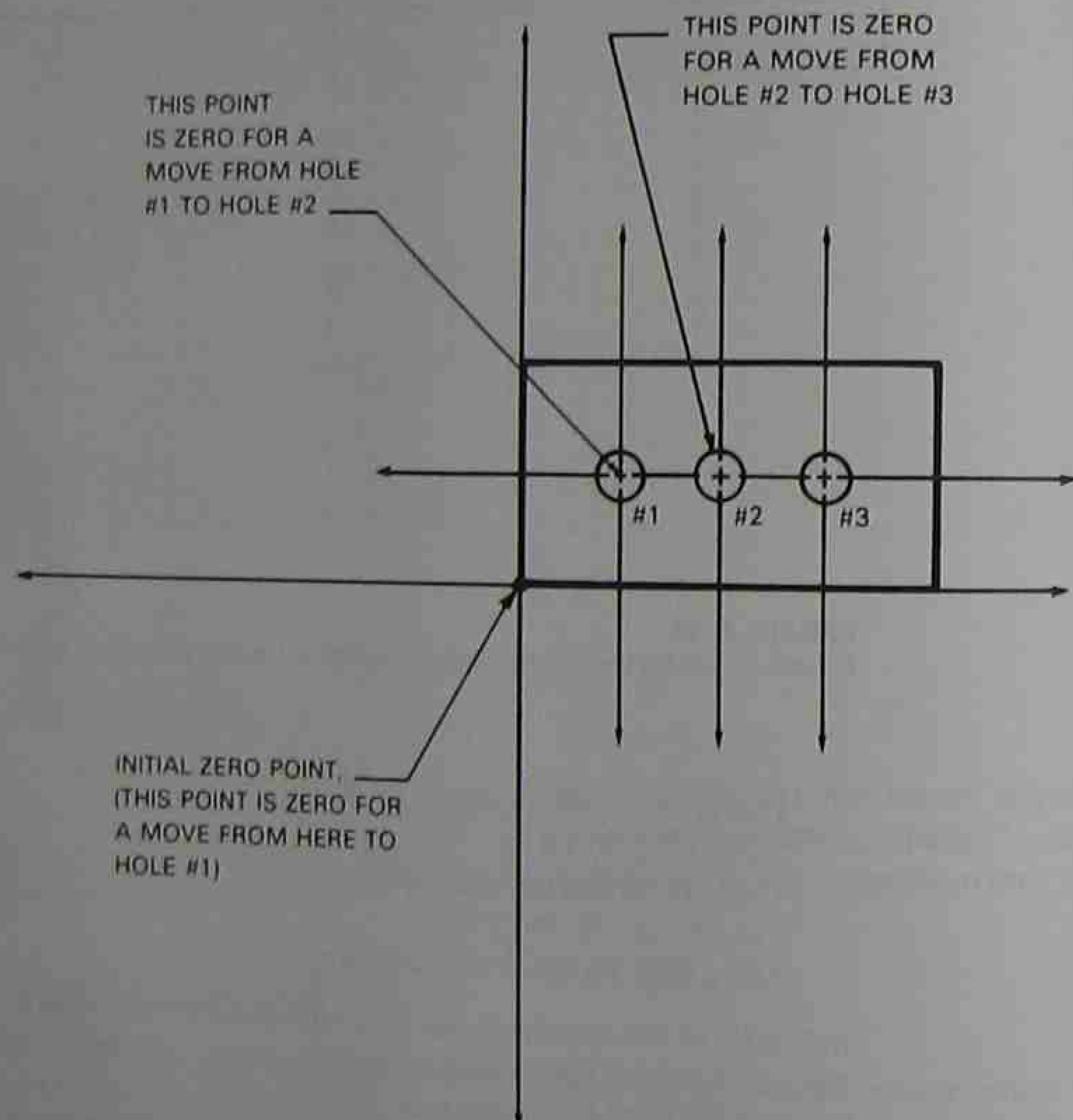


FIGURE 2–14
Relationship of the Cartesian coordinate system to the part when using incremental positioning

When using G92 zero shifts, the coordinate system was changed to the part coordinate system when the G92 line was issued. When using work coordinates, a register can be set at one place in the program and called at another. If more than one fixture is used on a machine, a second part zero can be entered in a second work coordinate, and called up when needed. The work coordinate registers can be set either manually by the operator, or in the program by the NC programmer, without having to send the spindle to the home zero location. This saves program cycle time by eliminating the moves to home zero in the program.

Work coordinates are set and called up in a program by commands called *G-codes*. G54, G55, and G56 would be examples of G-codes to call up different work coordinate registers. The following is an example of using work coordinates:

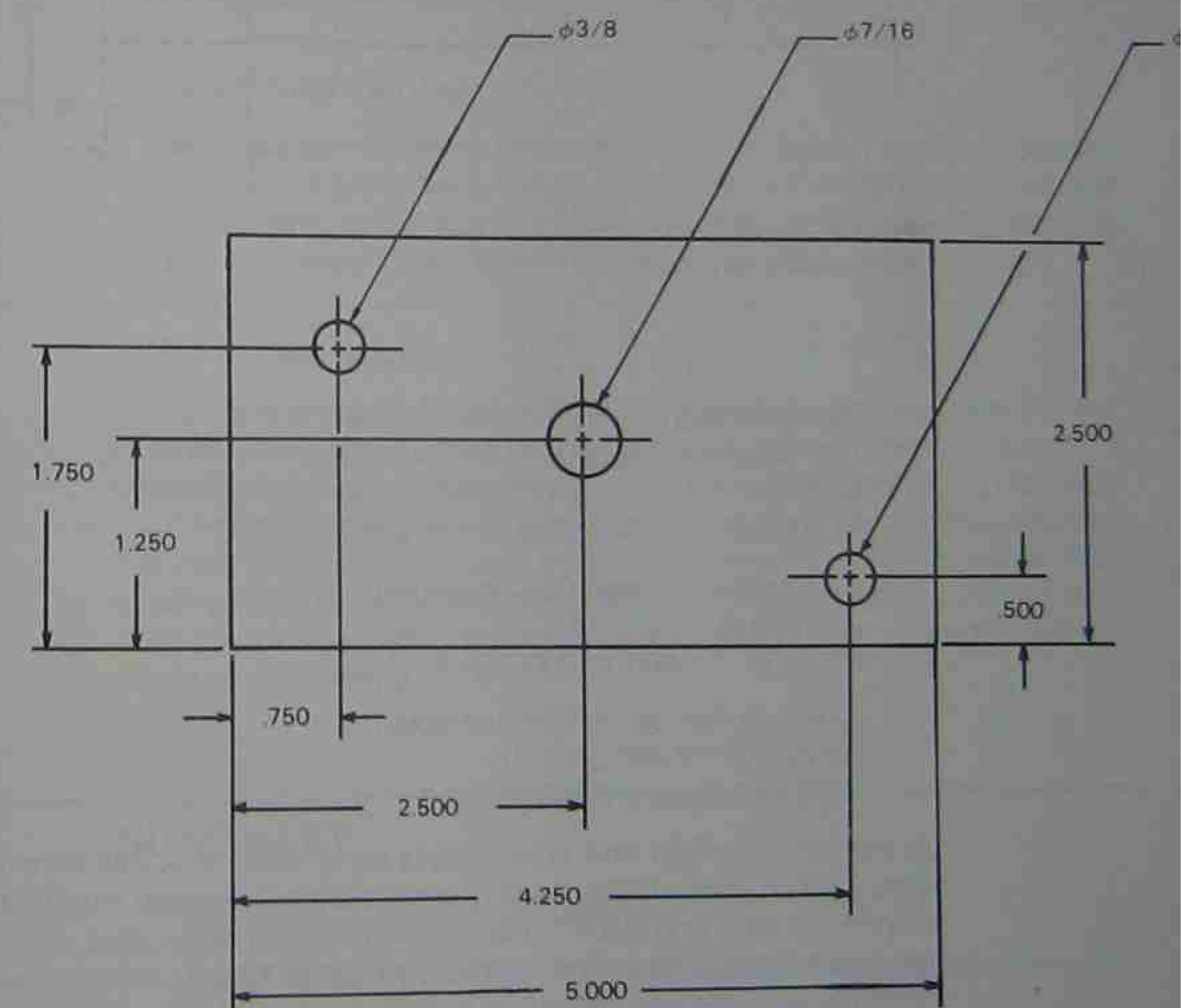


FIGURE 2–15
A datum dimensioned drawing

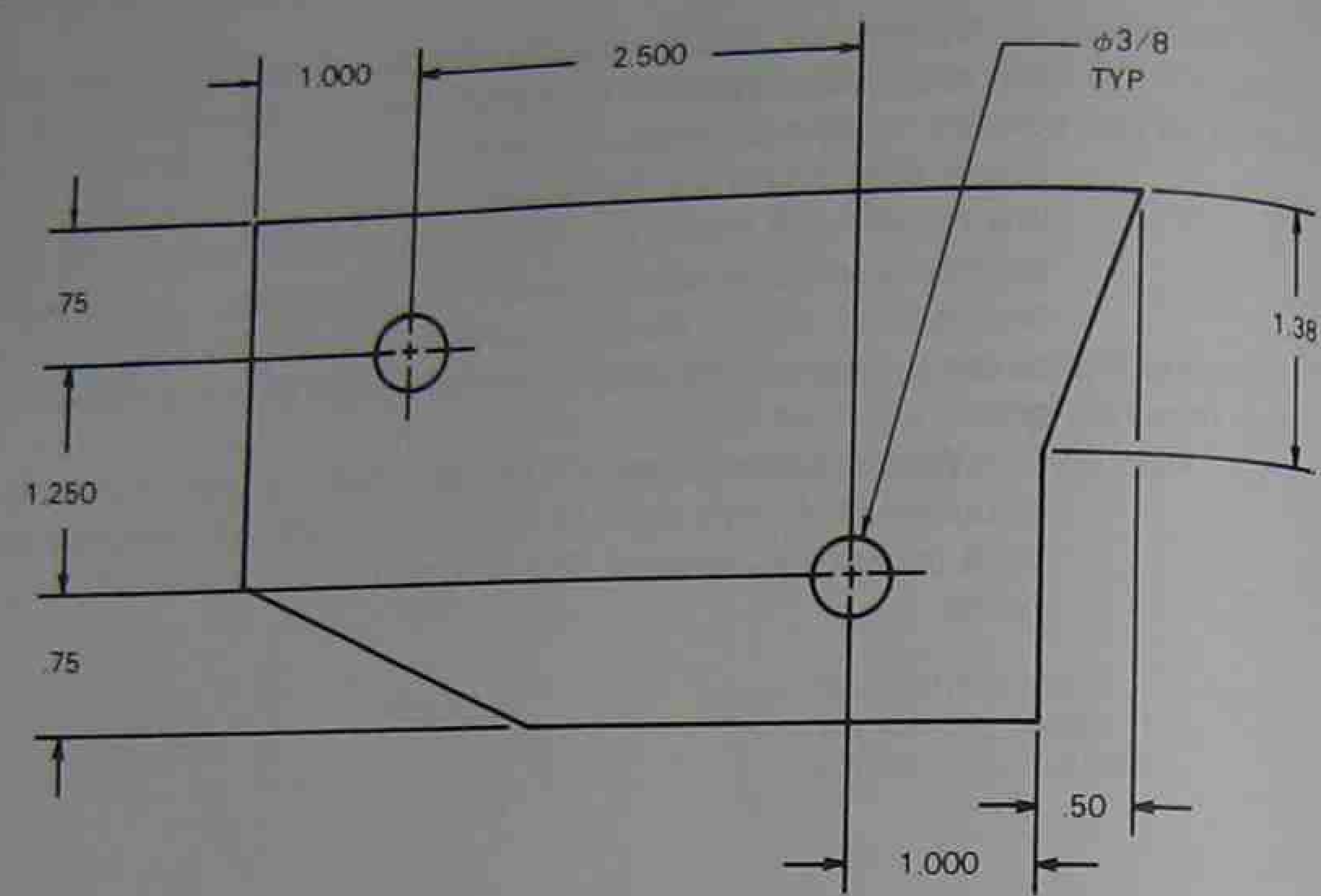


FIGURE 2–16
A delta dimensioned drawing

(Set work coordinate P1—which is G54)
(and work coordinate P2—which is G55)
N010 G10 L2 P1 X5.000 Y6.000 Z7.000
N020 G10 L2 P2 X10.000 Y3.000 Z15.000

(Call work coordinate G54 and move)
(To X1.000 Y1.000 Z.500)
N100 G54 X1.000 Y1.000 Z.500

(Call work coordinate G55 and move)
(To X2.000 Y2.000 Z3.000)
N110 G55 X2.000 Y2.000 Z3.000

In line N010, the G54 work coordinate is set to X5.0, Y6.0, Z7.0 from the home zero location. In line N020, the G55 work coordinate is set to X10.0 Y3.0 Z15.0 from home zero. In line N100, the G54 work coordinate is called, activating the part coordinate system. The spindle is moved to X1.0 Y1.0 Z.5 as referenced from the activated part coordinate system. In line N110, the G55 work coordinate is called, activating the second part coordinate system. The spindle is moved to X2.0 Y2.0 Z3.0 as referenced from the second part coordinate system.

Work coordinates remain active once called until cancelled by another work coordinate. They may be called on a line by themselves, or in a line with motion commands as in the example.

DIMENSIONING

In conjunction with NC (or N/C) machinery, there are two types of dimensioning practices used on part blueprints: *datum* and *delta*. These two dimensioning methods are related to absolute and incremental positioning. (Note: although this text uses the NC abbreviation, N/C is equally accepted and is beginning to become the more prominent form.)

Datum Dimensioning

In datum dimensioning, all dimensions on a drawing are placed in reference to one *fixed* zero point. Datum dimensioning is ideally suited to absolute positioning equipment. Figure 2–15 shows a datum dimensioned drawing; notice how all dimensions are taken from the corner of the part.

Delta Dimensioning

Dimensions placed on a delta dimensioned drawing are "chain-linked." Each location is dimensioned from the *previous* one, as shown in Figure 2–16. Delta drawings are suited for programming incremental positioning machines.

In many cases, the drafting practice does not suit the available machines. It is often necessary to calculate program coordinates from print dimensions because a delta dimensioned drawing is being used to program an absolute positioning machine, and vice versa. It is not uncommon to find the two methods mixed on one drawing.

SUMMARY

The important concepts presented in this chapter are:

- There are two types of NC control systems: point-to-point and continuous-path.
- There are four types of drive motors used on NC equipment: stepper motors, AC servos, DC servos, and hydraulic servos.

- Loop systems are electronic feedback systems used to help control machine positioning. There are two types of loop systems: open and closed. Closed loop systems can correct errors induced by the drive system; open loop systems cannot.
- The basis of machine movement is the Cartesian coordinate system. Any point on the Cartesian coordinate system may be defined by X/Y or X/Y/Z coordinates.
- An absolute positioning system locates machine coordinates relative to a fixed datum reference point.
- In an incremental positioning system, each coordinate location is referenced to the previous one.
- The machine coordinate system can be transferred to the part coordinate system manually, by an absolute zero shift, or by use of work coordinates.
- The positive or negative direction of an axis movement is always thought of as spindle movement.
- Machine movements occur along axes which correspond to the direction of travel of the various machine slides. On a vertical mill, the Z axis of a machine is always the spindle axis. The X and Y axes of a machine are perpendicular to the Z axis, with X being the axis of longer travel.
- There are two dimensioning systems used on part drawings intended for numerical control: datum and delta. Datum dimensioning references each dimension to a fixed set of reference points; delta dimensioning references each dimension to the previous one.

REVIEW QUESTIONS

1. What is the difference between point-to-point and continuous-path systems?
2. What are the four types of drive motors used on numerical control machines?
3. What is a loop system?
4. What is the difference between an open and closed loop system? Why is the choice of loop system important?
5. What is a machine axis?
6. What machine feature determines the positive or negative direction of an axis?
7. What are the two types of positioning systems? What are the differences between them?
8. What three methods are used to origin a part on a CNC machine?
9. What two types of dimensioning systems are used on NC part prints?
10. Give the coordinates of the points shown in Figure 2–17.

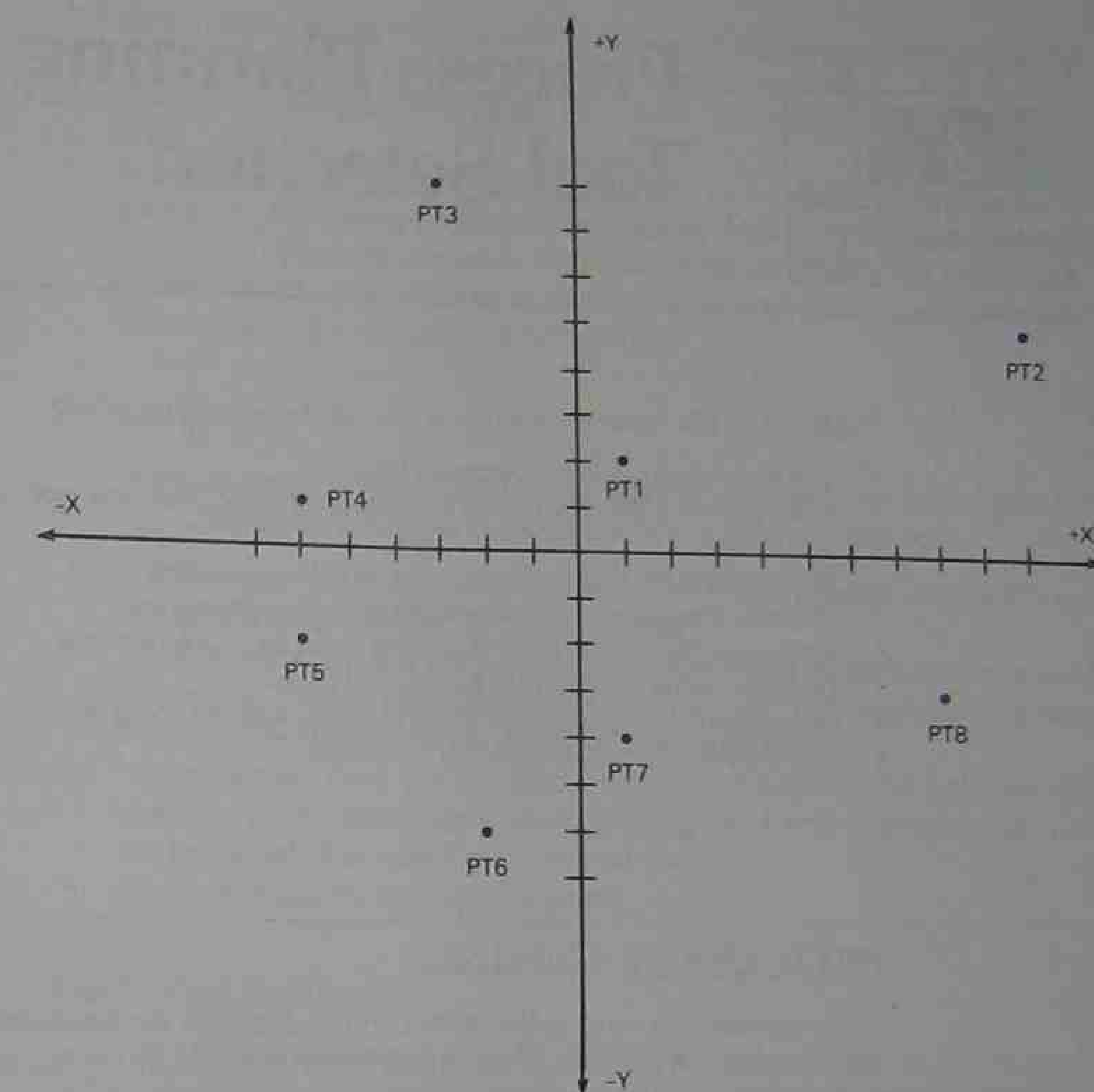


FIGURE 2–17
Coordinate system for review question #10

CHAPTER 3

Process Planning and Tool Selection

OBJECTIVES Upon completion of this chapter, you will be able to:

- List the steps involved in process planning.
- List the factors that influence the selection of an NC machine, workholding devices, and tooling.
- Describe the types of tools available for hole operations.
- Describe the types of tools available for milling operations.
- Determine the proper grade of carbide insert for a given material.
- Describe some common NC turning tool types.
- Determine the proper spindle RPM to obtain a given cutting speed.
- Explain the importance of proper feedrates.

PROCESS PLANNING

Process planning is the term used to describe the development of an NC part program. A number of decisions must be made by the NC programmer to successfully program a part.

- Which NC machine should be used?
- How will the part be held in the machine?
- What machining operations and strategy will be used?
- What cutting tools will be used?

This process is known as *methodizing*—developing the entire method of producing the part.

Machine Selection

A programmer must first decide which machine will be used. This decision is based on a number of factors.

- What is the programmer's experience?
- What machines are available?

- How many parts are in the order? Are there enough to justify the setup time and higher per hour run cost on a more complex machine?
- Is the particular part best suited for a lathe or a milling machine application?
- Is a vertical or horizontal spindle preferred? Vertical spindles are advantageous for hole drilling and boring operations. Horizontal spindles are best for heavy milling operations. The horizontal orientation of the spindle causes the chips to fall away from the tool, whereas vertical spindles tend to keep the chips packed around the tool.

Fixturing

The next decision to be made is how will the workpiece be held? Again this decision is based on a number of factors, many of them economic.

- Will standard holding devices (clamps, mill vises, chucks, etc.) suffice, or will special fixturing need to be developed?
- What quantity of parts will be run? A large number of parts means special fixturing to shorten the machining cycle may be feasible, even if conventional workholding methods would otherwise be used.
- How elaborate does the fixturing need to be? If many part runs are foreseen, a more durable fixture must be designed. If only one or two part runs are projected, a simpler fixture can be used.
- What will make the best quality part?

Machining Strategy

The machining strategy must be developed before the NC program can be written. Machining sequences used in a part program are determined by the following decisions.

- What is the programmer's experience?
- What is the shape of the part and the blueprint tolerance?
- What tooling is available?
- How many parts are in the order?

Tool Selection

Tool selection is the final important step in process planning. The selection is based on the following decisions.

- What tools are available?
- What machining strategy is to be used?
- How many parts are in the order? If a large number of parts are in the order, special timesaving tools can be made or purchased.

- What are the blueprint tolerances?
- What machine is being used?

The NC Setup Sheet

Once the process planning is finished and the program written, the programmer must communicate to the setup personnel in the shop what tools and fixtures are to be used in the NC program. This information is often placed on setup sheets such as those shown in Figures 3–1 and 3–2. The setup sheet should contain all necessary information to prepare for the job. Any special instructions to the setup personnel or machine operators should be communicated. Any special notes regarding tooling should also be included.

TOOLING FOR NUMERICAL CONTROL

Tooling is a vital consideration in efficient NC programming. This section is intended to show the prospective NC programmer various standard tooling options available. It is assumed the student has had exposure to cutting tools commonly used in a machine shop. A brief look through any beginning machining text will supply any necessary review of cutting tools.

Cutting Tool Materials

Cutting tools are available in three basic material types: high speed steel, tungsten carbide, and ceramic. The type of tool should be carefully chosen.

High speed steel (HSS) is one type of tool material. It has the following advantages over carbide:

- HSS costs less than carbide or ceramic tooling.
- HSS is less brittle and not as likely to break during interrupted cuts.
- The tools can be resharpened easily.

High speed steel tools have the following disadvantages:

- HSS does not hold up as well as carbide or ceramic at the high temperatures generated during machining.
- HSS does not cut hard materials well.

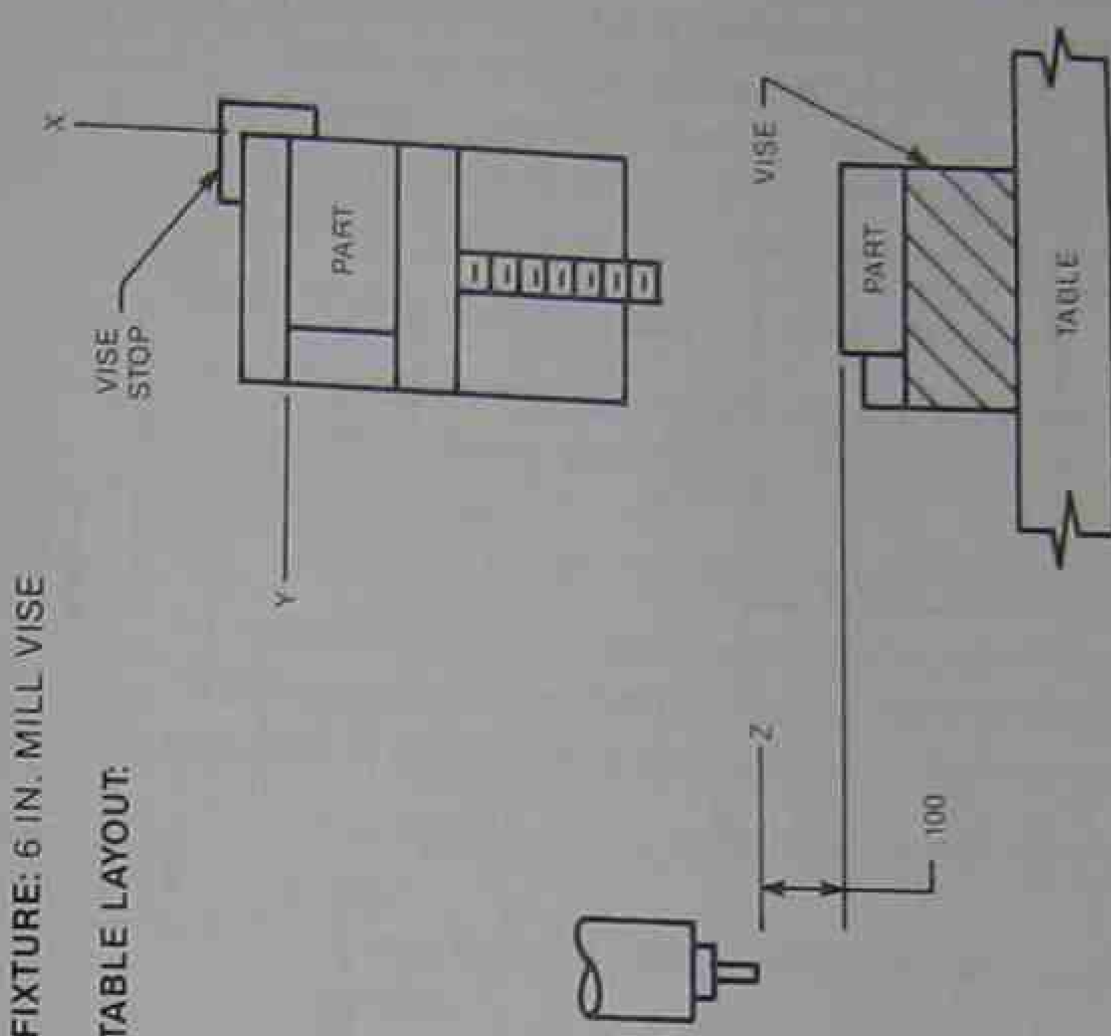
Tungsten carbide (known simply as carbide) is another material often used for cutting tools. Carbide tools come in one of three basic types. Solid carbide tools are made from a solid piece of carbide. Brazed carbide tools use a carbide cutting tip brazed on a steel shank. Inserted carbide tooling utilizes indexable inserts made of carbide which are held in steel tool holders. Tungsten carbide has the following advantages over high speed steel:

STA. NO.	CRO REG.	TOOL DESCRIPTION
1	—	3.0 DIA. INSERTED FACE MILL
		W/ .015 R GRADE 883 INSERTS
2	D12	.500 DIA. 4-FLUTE SOLID CARB. END MILL
3	—	NO. 4 × 90° C'DRILL
4	—	1/4 DRILL (.250 DIA.)
5	—	.262 DIA. BORING BAR

TAPE NUMBER: 1053

FIXTURE: 6 IN. MILL VISE

TABLE LAYOUT:



NOTES: DRILL POINT ANGLES TO BE 118° INCL.

DRWN: WSS

PROG: WSS

DATE: 1-10-89

B/P REV: C

NC SETUP SHEET FOR:

MACHINE: VERTICAL MACHINING CENTER

FIGURE 3–1
NC setup sheet for a CNC machining center

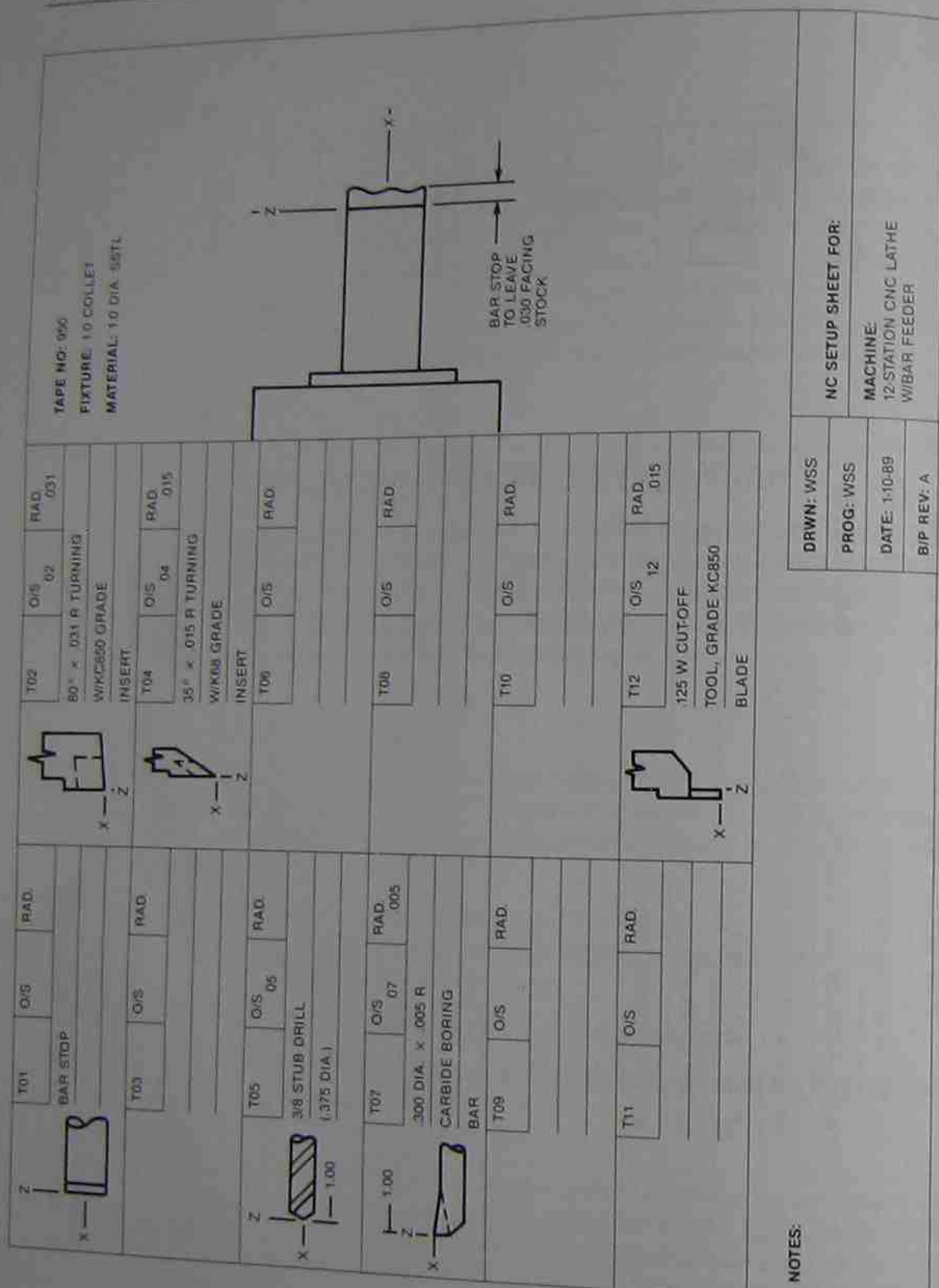


FIGURE 3-2
NC setup sheet for a CNC lathe

- Carbide holds up well at elevated temperatures.
- Carbide can cut hard materials well.
- Solid carbide tools absorb workpiece vibration and reduce the amount of "chatter" generated during machining.
- When inserted cutters are used, the inserts can be easily changed or indexed, rather than replacing the whole tool.

Carbide also has the following disadvantages:

- Carbide costs more than high speed steel.
- Carbide is more brittle than HSS, and has a tendency to chip during interrupted cuts.
- Carbide is harder to resharpen and requires diamond grinding wheels.

Ceramic tooling has made great advances in the past several years. While once very expensive, some ceramic inserts can now be purchased for less than the cost of carbide. Ceramic has the following advantages:

- Ceramic is sometimes less expensive than carbide when used in insert tooling.
- Ceramic will cut harder materials at a faster rate and has superior heat hardness.

Ceramic has the following disadvantages:

- Ceramic is more brittle than HSS or carbide.
- Ceramic must run within its given surface speed parameters. If run too slowly, the insert will break down quickly. Many machines do not have the spindle RPM range needed to use ceramics.

High speed steel is generally used on aluminum and other nonferrous alloys, while carbide is used on high silicon aluminums, steels, stainless steels, and exotic metals. Ceramic inserts are used on hard steels and exotic metals. Inserted carbide tooling is becoming the preferred tooling for many NC applications.

Some carbide inserts are coated with special substances, such as titanium nitride to improve the insert life. These coatings can increase tool life by up to 20 times when used in accordance with the manufacturer's recommended cutting speeds and feedrates.

TOOLING FOR HOLE OPERATIONS

There are four basic hole operations which are performed on NC machinery: drilling, reaming, boring, and tapping.

Drilling

Drills are available in different styles for different materials. Figure 3–3 shows a standard twist drill. Even with all the new tooling technology, twist drills remain one of the most common tools for making holes. Drills have a tendency to walk as they drill, resulting in a hole that is not truly straight. Centerdrills such as shown in Figure 3–4 are often used to predrill a pilot hole to help twist drills start straight. Drills also produce triangular shaped holes.

If a hole tolerance is closer than .003 inch, a secondary hole operation should be used to size the hole, such as boring or reaming. Large holes are sometimes produced by spade drills (Figure 3–5). The flat blades allow good chip flow and economical replacement of the drill tip.

Drill point angle must be considered when selecting a drill. The harder the material to be cut, the greater the drill point angle needs to be to maintain satisfactory tool life. Mild steel is usually cut with a 118-degree included angle drill point. Stainless steels often use a 135-degree drill point.

Drills are available in different types. HSS drills are the most common, but brazed carbide and solid carbide are also used. Carbide drills have a tendency to chip when drilling holes. When drilling hard materials cobalt drills (HSS with cobalt added to the alloy) are used. Cobalt drills have greater heat hardness than HSS drills.

Special drills utilizing carbide inserts have been developed for NC applications (Figure 3–6). The economics of using these tools should be considered by the programmer when hard materials or high run quantities are involved.

FIGURE 3–3
Tapered shank twist drill *(Photo courtesy of Morse Cutting Tools Division)*

FIGURE 3–4
Center drill *(Photo courtesy of DoALL Manufacturing)*

Reaming

Reaming is used to remove a small amount of metal from an existing hole as a finishing operation. Reaming is a precision operation which will hold a tolerance of + or − .0002 easily.

Reamers are made with two basic flute designs: straight fluted (Figure 3–7) and spiral fluted (Figure 3–8). Spiral fluted reamers produce better surface finishes than straight flutes, but are more difficult to resharpen. Reamers are available in three basic tool materials: high speed steel, brazed carbide, and solid carbide.

Boring

Boring removes metal from an existing hole with a single point boring bar. Boring heads are available in two designs: offset boring heads, in which the boring bar is a separate tool inserted into the head, and cartridge type. Cartridge boring heads use an adjustable insert in place of a boring bar.

Boring bars are available in the four material types: high speed steel, solid carbide, brazed carbide, and inserted carbide. Inserted carbide bars are used for large holes, whereas brazed and solid carbide bars are usually supplied in smaller sizes (up to ½–inch diameter).

Tapping

Tapping is used to produce internally threaded holes. They are available in several flute designs. Standard machine screw taps (Figure 3–9) are widely used, especially when tapping blind holes. Spiral pointed taps (known as gun taps) are preferred for thru hole operations. These taps shoot the chips forward and out the bottom of the hole. High spiral taps (Figure 3–10) are used for soft stringy material such as aluminum.

FIGURE 3–5
Spade drill *(Photo courtesy of DoALL Manufacturing)*

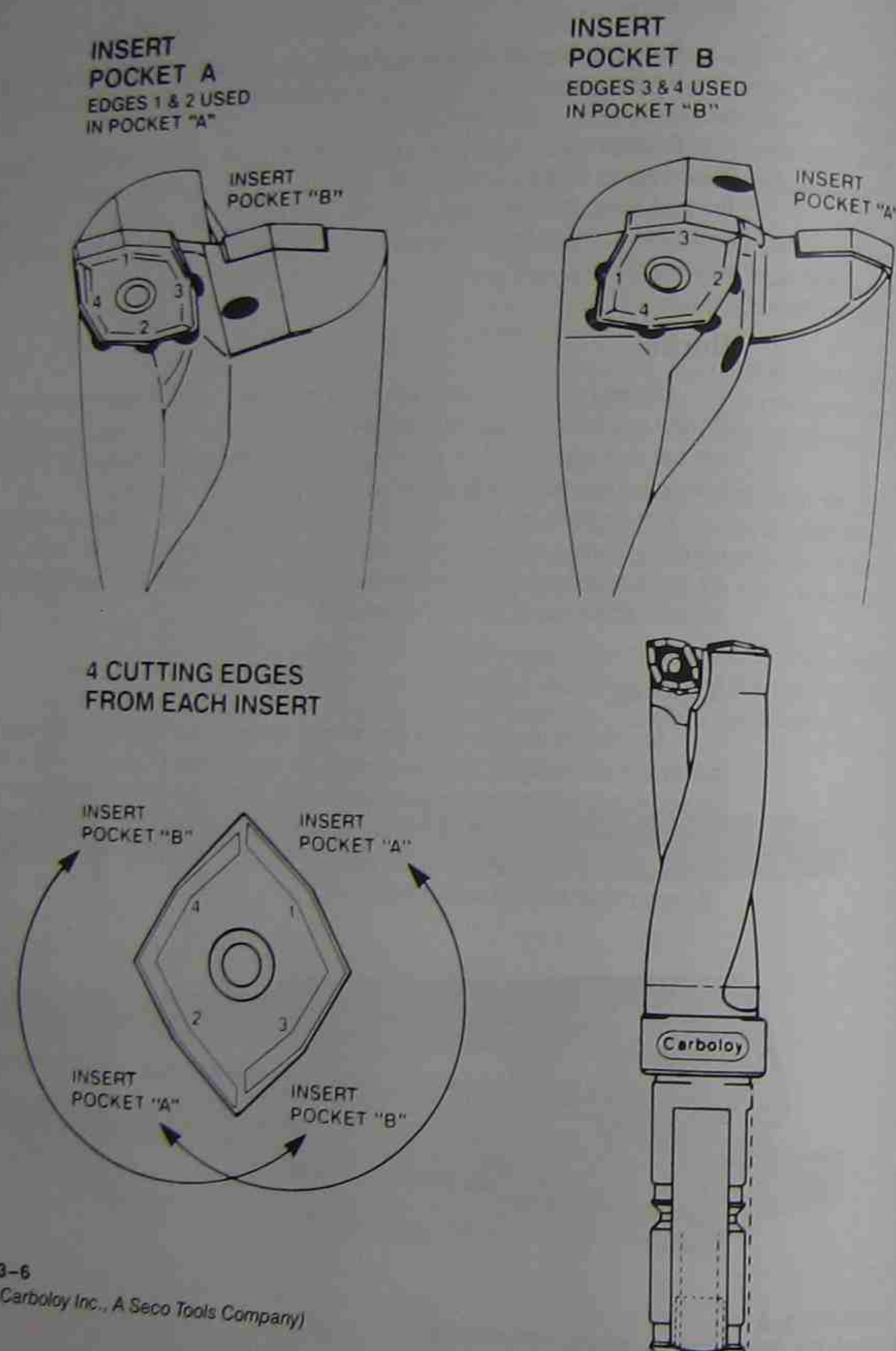


FIGURE 3–6
(Courtesy Carboloy Inc., A Seco Tools Company)

FIGURE 3–7
Straight flute chucking reamer *(Photo courtesy of Cleveland Twist Drill Company)*

FIGURE 3–8
Spiral flute chucking reamer *(Photo courtesy of DoALL Manufacturing)*

A special milling cutter called a thread hob (Figure 3–11) is sometimes used to mill a thread in a workpiece. Thread hobs make use of an NC machine's helical interpolation capabilities. Helical interpolation is presented in Chapter 12.

MILLING CUTTERS

The greatest advances in tooling for NC have taken place in the area of inserted milling cutters. *Milling* allows the contouring capabilities of the NC machine to be used to efficiently perform operations that would require special tooling if done manually. Milling cutters can be placed in two basic categories: solid milling cutters and inserted milling cutters. They can be further classified as end mills and face mills.

End Mills

End mills are available in HSS and solid carbide from .032 inch to 2 inches in diameter in two or four flute. Inserted end mills are available from .500 inch to 3 inch diameters. Figure 3–12 shows a four flute HSS end mill. Figure 3–13 shows a two flute solid carbide end mill. Two flute cutters with their deeper gullets are well suited for roughing operations. Four flute end mills, however, are more rigid because of their thicker core. The programmer's experience will determine when to use a two or four flute cutter.

Figures 3–14 and 3–15 illustrate two different types of inserted end mills. Inserted cutters are preferred for NC applications. Inserts are less expensive to replace than an entire tool. By indexing the inserts, four or six cutting edges can be used on one insert. When the insert is used up, it is thrown away rather than resharpened. Inserted cutters may also be used on many different types of workpiece materials by simply changing the inserts from one designed for aluminum, for example, to one designed for stainless steel.

Figures 3–16 and 3–17 show two different styles of inserted ball end mills. Ball end mills are also available in HSS and solid carbide. Ball mills are used for three-, four-, or five-axis contouring work, where the Z-axis will be used. They are also used to produce a given radius on a part.

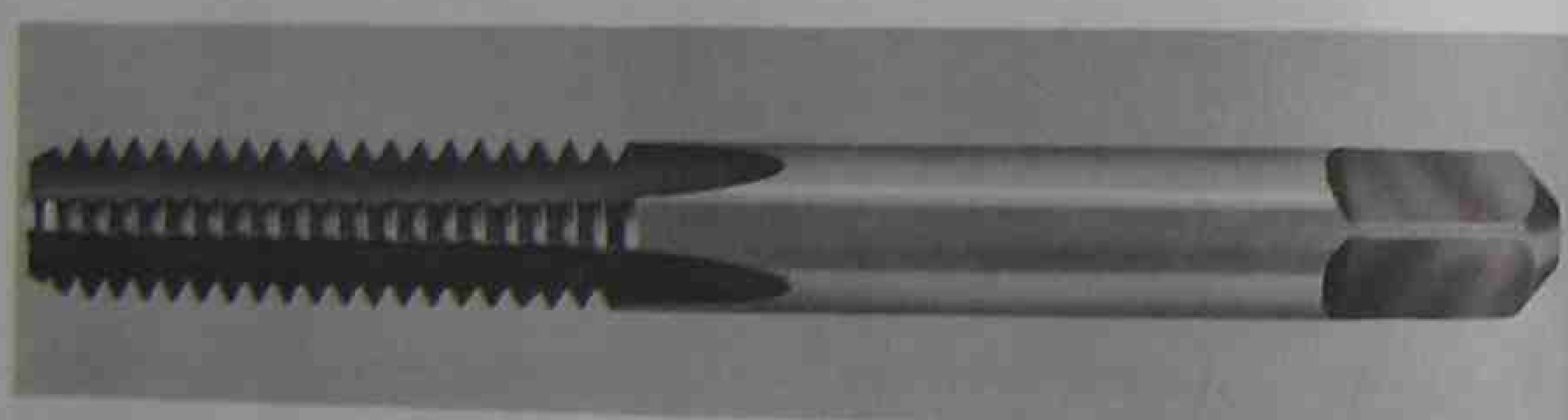


FIGURE 3–9
Machine screw tap *(Photo courtesy of Morse Cutting Tools Division)*

FIGURE 3–10
High spiral coated tap *(Photo courtesy of DoALL Company)*

Figure 3–18 shows a special type of inserted end mill called a cyclo mill, designed by Valenite GTE. It uses a series of round inserts staggered in a helical pattern. This mill can remove large amounts of material at fairly high speeds. It is just one example of inserted tooling that is being developed for NC use.

Face Mills

Face mills differ from end mills in their major application. Face mills are designed to remove large amounts of material from the face of a workpiece. They are manufactured in HSS, brazed carbide, and inserted carbide types.

Face mills are available in sizes from 2 inches to over 8 inches in diameter. Inserted carbide is the most common type of facing tool. The costs of large brazed carbide and HSS mills limit their application to special situations.

Figure 3–19 shows a common type of inserted face mill. Figure 3–20 shows a large diameter face mill. Note the number of inserts used. In Figure 3–

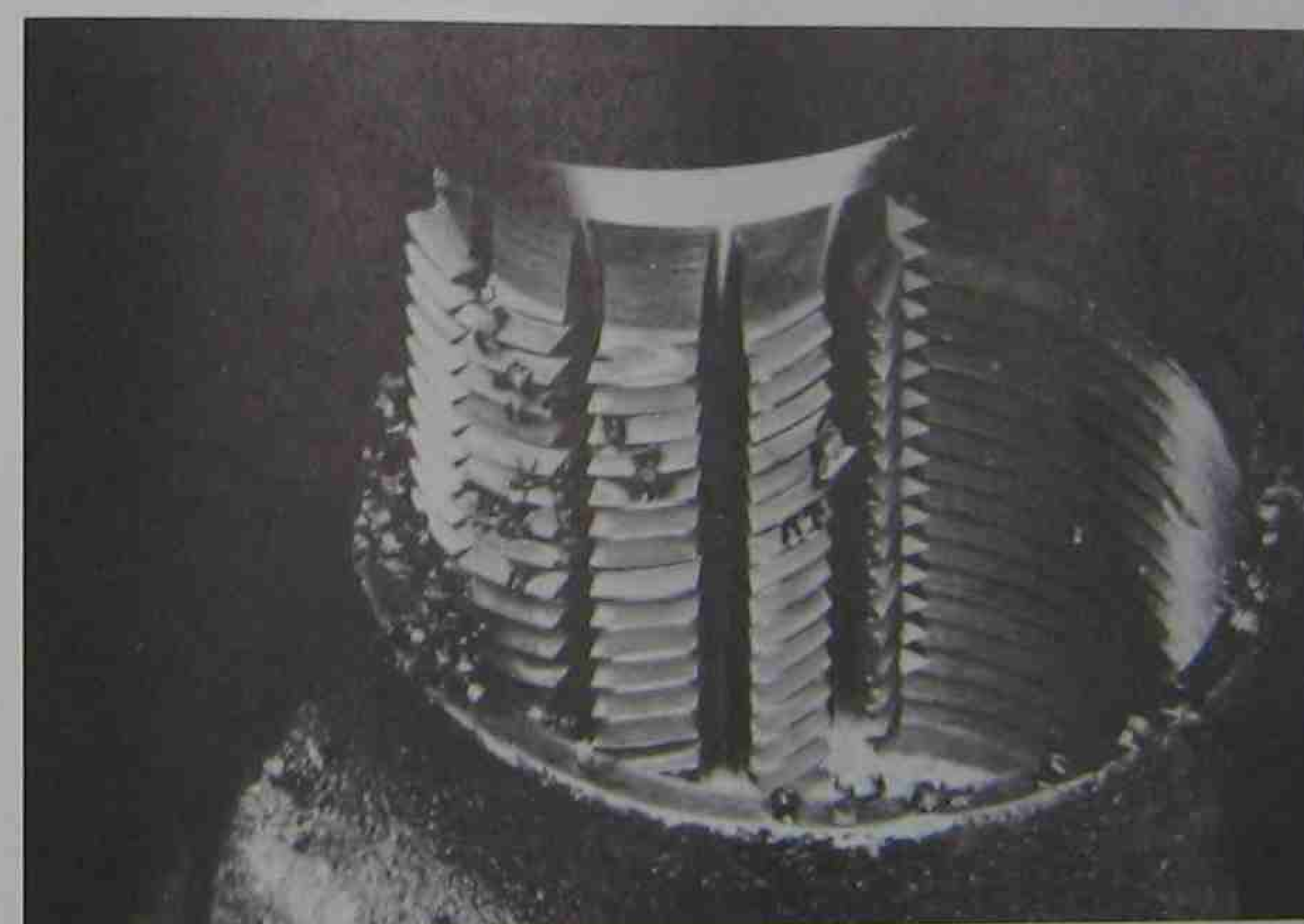


FIGURE 3–11
Thread hob *(Photo courtesy of GTE Valenite)*

FIGURE 3–12
Single end, multiple flute end mill, standard length flutes (Photo courtesy of Sharpaloy Division, Precision Industries, Inc.)

FIGURE 3–13
Solid carbide, two-flute, end mill (Photo courtesy of DoALL Company)

21, a special type of mill cutter is shown. This cutter is called a plunge and profile cutter. It is designed to plunge into the material first and then begin the cutting path. This design is a cross between an end mill and a face mill.

SPECIAL INSERTED CUTTERS

A number of special tools have been developed for uses with NC. The NC programmer is always confronted with new ideas to improve productivity. Prospective and experienced programmers should spend time looking at tooling catalogs to become acquainted with current tooling developments.

Figures 3–22 through 3–24 illustrate some of the current tooling ideas developed specifically for NC applications.

Carbide Inserts and Their Selection

Carbide inserts are manufactured in a variety of types and grades. The type of insert describes the shape of the insert. Some common shapes include

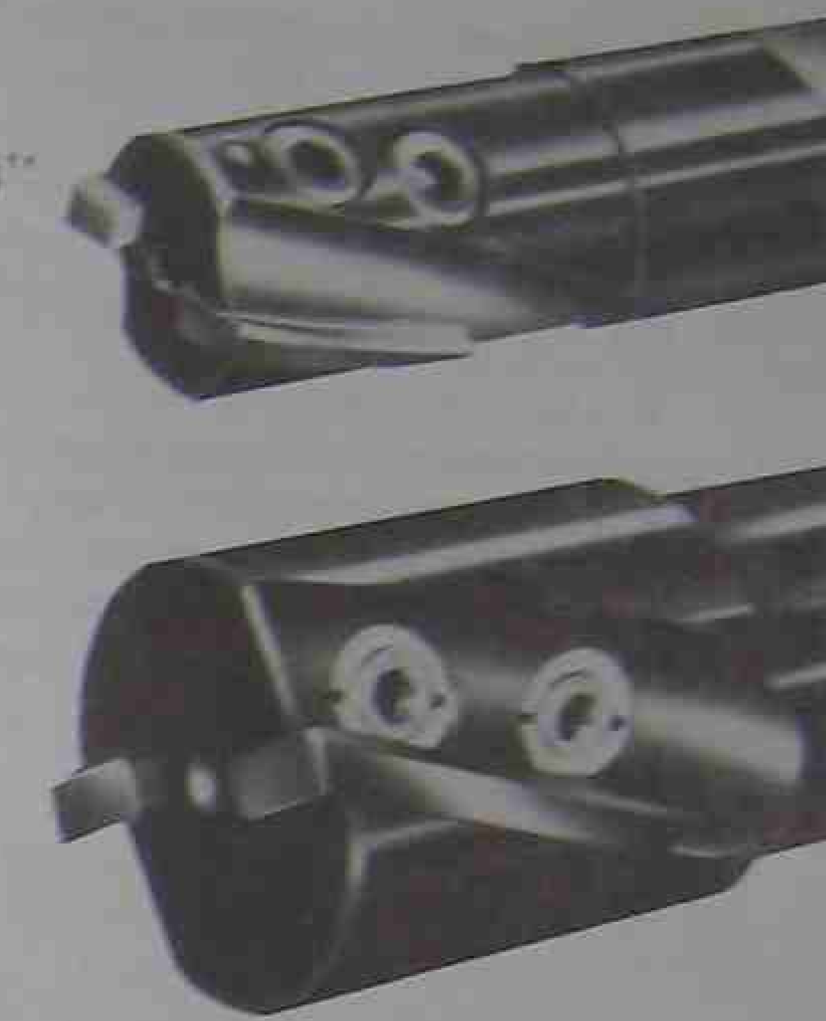


FIGURE 3–14
Inserted carbide end mills (Photo courtesy of GTE Valenite)

triangular, 80-degree diamond, 55-degree diamond, and round. The grade of insert refers to the hardness of the insert and application for which it was developed. The NC programmer must be aware of the type and grade of insert available when making tooling selections. Each type of insert is identified by a designation code. The identification system used on an insert will vary depending on the manufacturer. Figure 3–25 shows one such system.

Figure 3–26 illustrates some of the carbide grades available and their applications. Each grade of carbide is designated by an ANSI "C" number designation from C–1 to C–8. In addition, each grade of carbide has also been classified by the ISO. The ISO designation uses a "K" or "P" number, depending on insert hardness. In the United States, the ANSI system is generally used. In other countries, the ISO system is followed. Manufacturers develop their own grade system based on the ANSI or ISO rating. A C–2 ANSI grade insert would be called CQ2 by RWT corporation, K1 by Kennametal Inc., and VC-1 by Valenite GTE Inc. It is necessary therefore, for the programmer to consult the individual manufacturer's catalog to arrive at the proper grade number.

Lathe Tooling

Carbide inserted tools dominate tool selection for CNC turning applications. One of the more popular insert styles is the diamond insert. Figure 3–27 illustrates a series of inserted turning tools. Figure 3 – 28 shows a number of boring bars and toolholders.

A PROCESSING EXAMPLE

In a large company, the formal processing (determining the machine routing through the shop) is done by a processing engineer. The process is then sent to the programming department where the tooling concepts and machining strategy is done. In a small company, the NC programmer does both the processing and programming. It is important that the department processing a job work closely with the programming department to efficiently and economically produce a manufacturing process.

FIGURE 3–15
"Centerdex" two-flute inserted end mills *(Photo courtesy of GTE Valenite)*

Figure 3 – 29 is a part that is to be machined from an aluminum casting. The casting has .250 dia. of stock to be removed from the 4.000 and 3.000 diameters. The center of the casting was cored to 1.000 inch, and the 1.00 height was cast at 1.250. After consultation with the NC programming department, the process illustrated in Figure 3 – 30 was developed. The 4.000 inch diameter and .38 dimension are to be done on a conventional lathe. The part will then be routed to a vertical spindle CNC machining center where the balance of the part is to be completed.

The fixture concept to hold the part was developed by the NC programmer. The concept drawing is shown in Figure 3 – 31. The part will be nested in the 4.0015 diameter fixture bore. It will be clamped with four swiveling clamps, available as a purchased item from a tooling component supplier. This fixture design was based on the following factors.

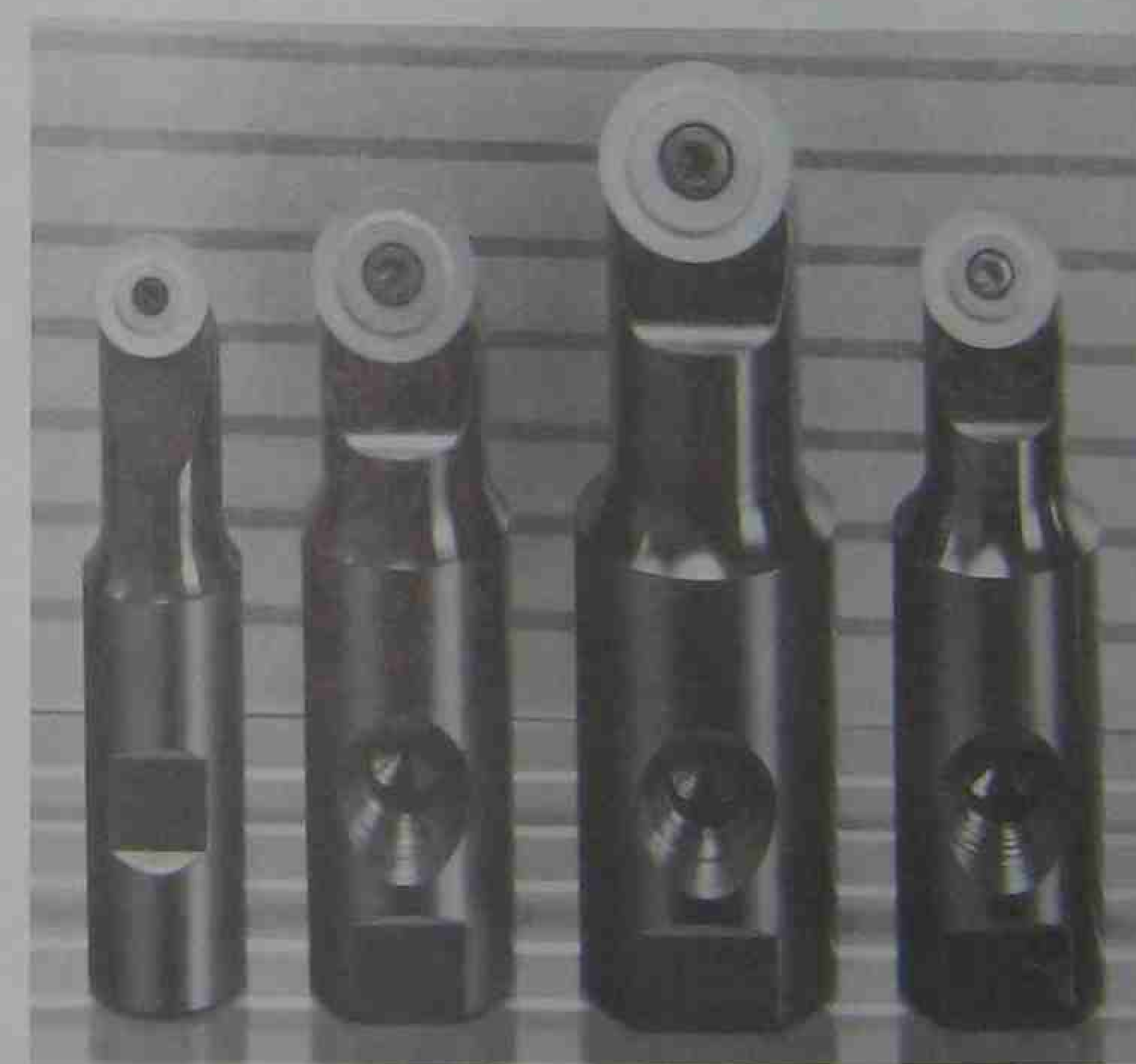


FIGURE 3–16
Ball nose end mills featuring round inserts *(Photo courtesy of GTE Valenite)*

FIGURE 3–17
Ball nose end mills featuring triangular inserts *(Photo courtesy of GTE Valenite)*

- The 4.000 diameter and .38 dimensions were completed in the previous operation, making this feature the logical choice for locating the part.
- The run quantity is only 200 parts. The fixture design is simple, making it economical to build.
- The design is easy to load.

The sequence of the machining operation at the machining center was planned as follows.

FIGURE 3–18
"Cyclo Mill" special multi-inserted milling cutter *(Photo courtesy of GTE Valenite)*

- Face the 1.000 and .25 dimensions using a 3¼ carbide inserted face mill.
- Center drill the .188 and .250 diameter holes. A 90-degree center drill was chosen. The 90 degree-chamfer will provide an edge break at the drilled hole, reducing the amount of deburr time.

- Drill the .188 diameter holes using a $\frac{3}{16}$ drill. Since drills almost always drill .001 or more oversize, the hole will be comfortably within tolerance.
- Drill the .250 diameter hole using a $\frac{1}{4}$ drill.
- Mill the 3.000 diameter using a $1\frac{1}{4}$ diameter inserted helical end mill. The end mill has inserts up the sides of the insert, allowing side cutting up to 2.00 deep.
- Using the same end mill, mill the 1.500 diameter bore.

The setup sheet for the NC operation is shown in Figure 3–32.

FIGURE 3–19
Carbide inserted face mill *(Photo courtesy of GTE Valenite)*

SPEEDS AND FEEDS

The efficiency and life of a cutting tool depend upon the cutting speed and the feedrate at which it is run.

Cutting Speed

The *cutting speed* is the edge or circumferential speed of a tool. In a machining center or milling machine application, the cutting speed refers to the edge speed of the rotating cutter. In a turning center or lathe application, the cutting speed refers to the edge speed of the rotating workpiece. Cutting speed

FIGURE 3–20
Large inserted face mill—note number of inserts on cutter *(Photo courtesy of GTE Valenite)*

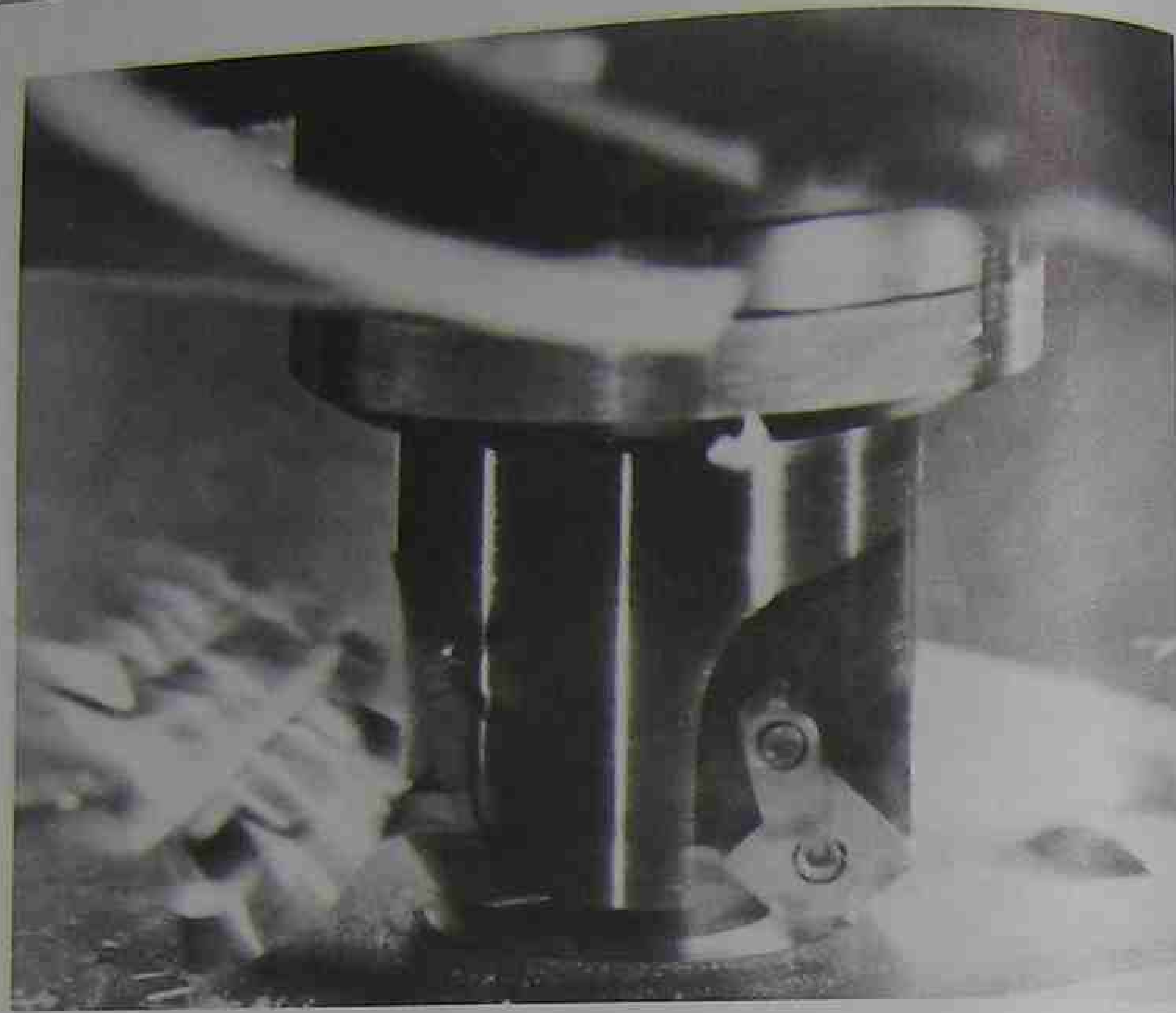


FIGURE 3–21
Plunge and profile inserted milling cutter *(Photo courtesy of GTE Valenite)*

(CS) is expressed in surface feet per minute (SFM). It is the number of feet a given point on a rotating part or cutter moves in one minute.

Proper cutting speed varies from material to material. Generally, the softer the material, the higher the cutting speed. Recommended cutting speeds for various materials can be found in tables contained in machinists' handbooks, and tooling manufacturers' catalogs. Appendix 6 of this text contains one such chart.

It should be understood that cutting speed and spindle RPM are two different things. A .250-inch diameter drill turning at 1200 RPM has a cutting speed of approximately 75 surface feet per minute. A .500-inch diameter drill turning 1200 RPM has a cutting speed of approximately 150 SFM. The spindle RPM necessary to achieve a given cutting speed can be calculated by the formula:

$$RPM = \frac{CS \times 12}{D \times \pi}$$

Where: CS = cutting speed in surface feet per minute
D = diameter in inches of the tool (workpiece diameter for lathes)
π = 3.1416

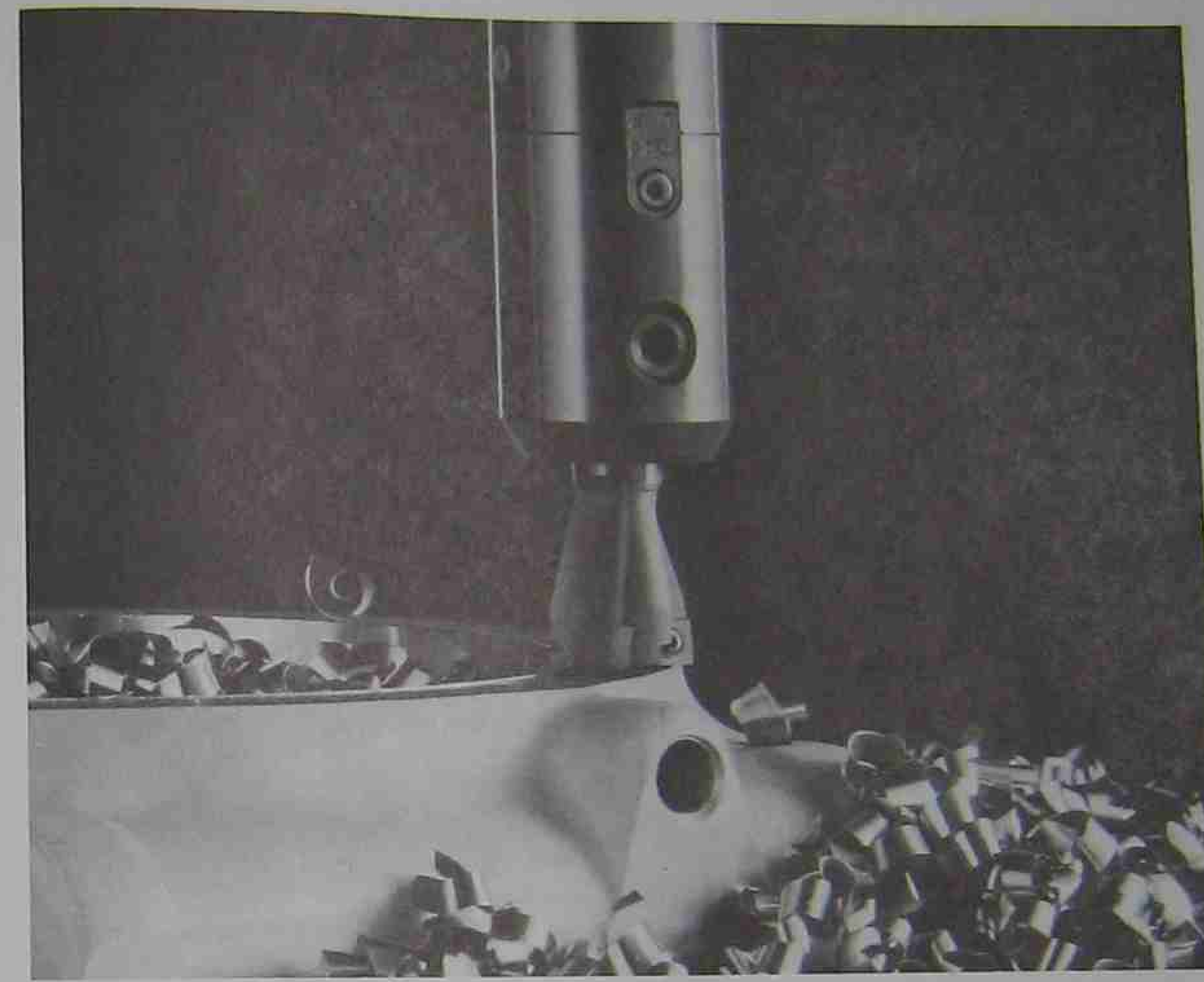


FIGURE 3–22
Special small diameter inserted end mill *(Photo courtesy of GTE Valenite)*

The cutting speed of a particular tool can be determined from the RPM, using the formula:

$$CS = \frac{D \times \pi \times RPM}{12}$$

On the shop floor, the formulas are often simplified. The following formulas will yield results similar to the formulas just given.

$$RPM = \frac{CS \times 4}{D}$$

$$CS = \frac{RPM \times D}{4}$$

For turning applications, the diameter of the workpiece, rather than the tool diameter, is used to determine the cutting speed and spindle speed. For milling applications, the diameter of the tool is used.

Feedrates

Feedrate is the velocity at which a tool is fed into a workpiece. Feedrates are expressed two ways: inches per minute of spindle travel and inches per revolution of the spindle. For milling applications, feedrates are generally given in inches per minute (IPM). For turning they are expressed most often in inches per revolution (IPR).

Feedrates are critical to the effectiveness of a job. Too heavy a feedrate will result in premature dulling and burning of tools. Feedrates which are too light will result in tools chipping. This chipping will rapidly lead to tool burning and breakage.

Turning Feedrates

The vast majority of turning tools used with NC are inserted tools. The feedrates used vary with material type and insert type. Tables found in manu-

FIGURE 3-23
Special inserted tooling for use with NC. From top to bottom: an inserted milling cutter with interchangeable tooling extensions, a machine tap in a tap holder with interchangeable tooling extensions, and an inserted drill mounted in a holder with interchangeable extensions *(Photo courtesy of GTE Valenite)*

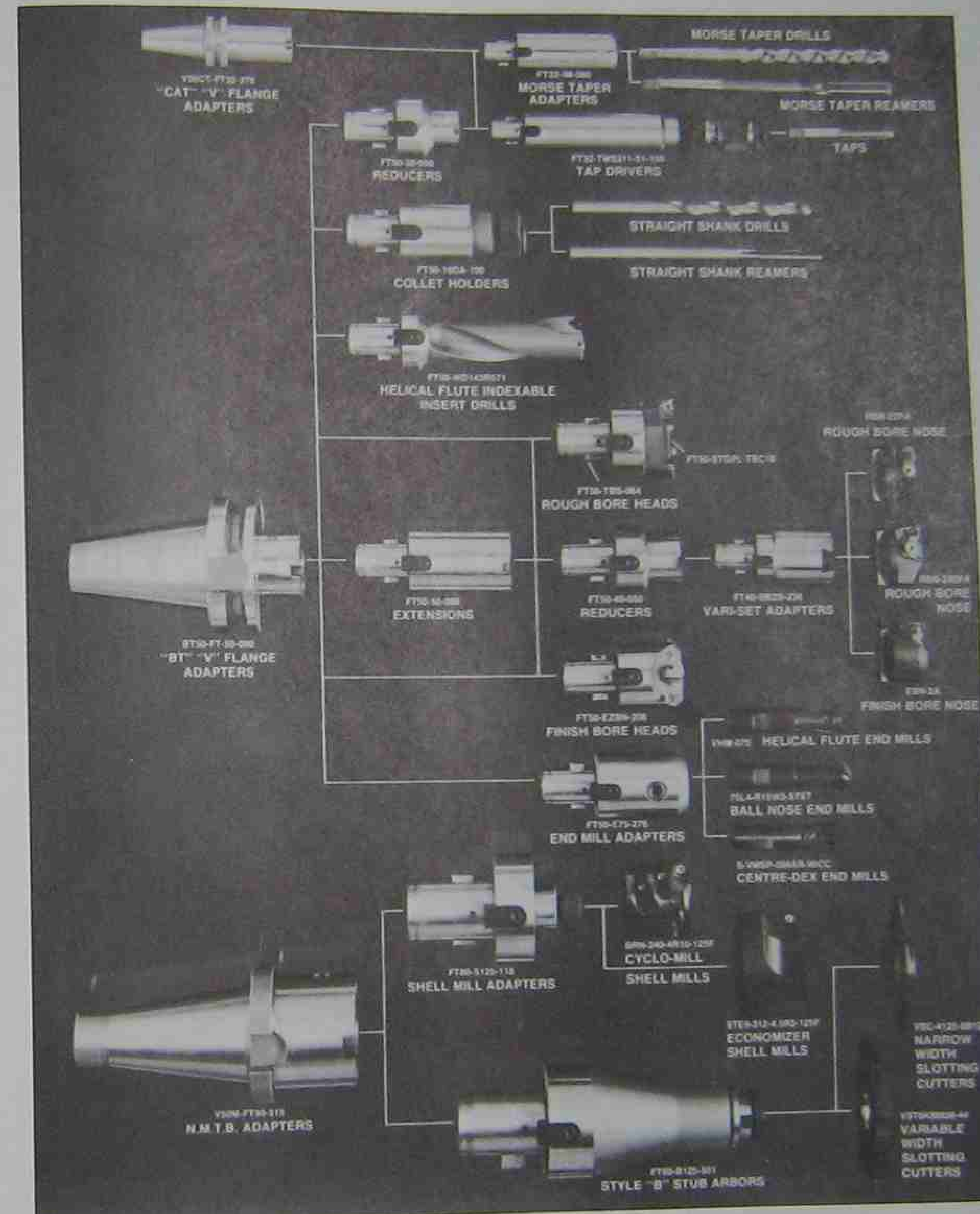


FIGURE 3-24
An NC tooling system featuring tool adapters, interchangeable extensions, tool bodies, boring heads, and arbors *(Photo courtesy of GTE Valenite)*

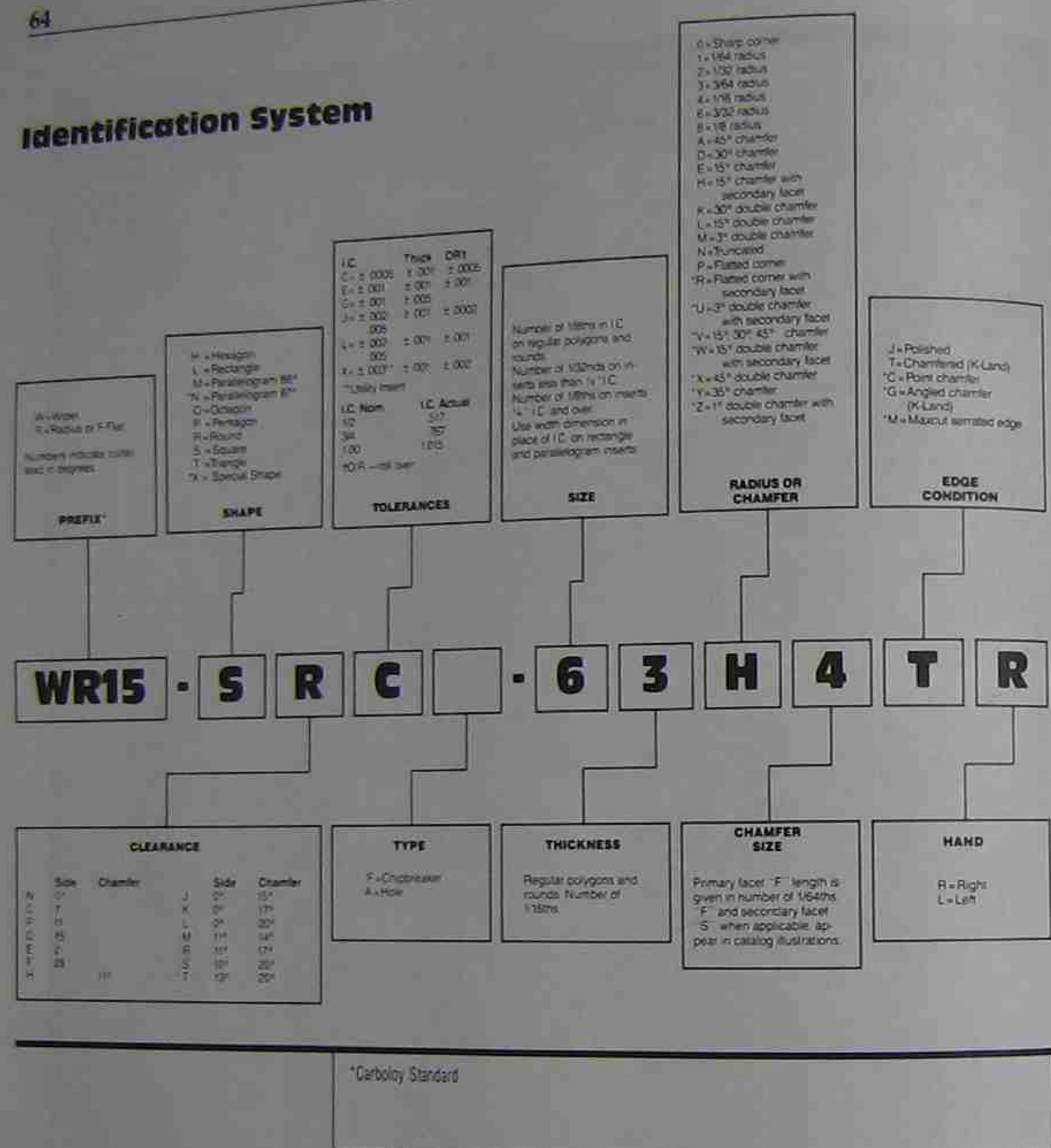


FIGURE 3–25
Carbide insert identification system *(Courtesy of Carboloy Inc., A Seco Tools Company)*

facturers' catalogs and machining data handbooks are the best sources for turning feedrates. It must be noted that conditions such as part geometry, machine rigidity, and rigidity of the setup will affect both speeds and feedrates. The values given in the tables are starting points. The actual speed and feedrate used during the run will ultimately be determined during the job setup, when the first piece is run.

INSERT GRADE APPLICATION CHART

Cast iron and nonferrous materials	Alloy and tool steels Stainless steels
C-1: Roughing	C-5: Roughing
C-2: General Purpose	C-6: General Purpose
C-3: Finishing	C-7: Finishing
C-4: Precision Finishing	C-8: Precision Finishing

MANUFACTURER'S GRADE DESIGNATION

ANSI Class	ISO Class	Carboloy	Iscar	Kennametal	Sandvik	Valenite
C-8	P-01 P-05	210	IC-80t	K7H	F02	VC-8
C-7	P-10 P-25	350	IC-70	K45	S1P	VC-7
C-6	P-25 P-35	370	IC-50	KC850	S4	VC-55
C-5	P-40 P-50	518	IC-54	—	S35	VC-5
C-4	K-01 K-05	999	IC-4	K11	—	VC-4
C-3	K-10 K-15	905	IC-20	K68	H10	VC-3
C-2	K-20 K-25	883	IC-2	K6	H20	VC-2
C-1	K-30 K-20	820	IC-28	K1	H	VC-1

Note: Most manufacturers produce more than one grade per insert class. Consult the manufacturer's catalog for a complete listing.

FIGURE 3–26
Carbide insert grades

Drilling Feedrates

Drilling feedrates are dependent on the drill diameter. Tables in machinists' handbooks will list recommended feedrates in IPR for given diameters of a given tool material. For example, HSS drills from 1/8 to 1/4 inch use feedrates of .002 to .004 IPR. Drills from 1/4 to 1/2 inch use feedrates of .004 to .007 IPR. Drills from 1/2 to 1 inch use feedrates from .007 to .015 IPR. The final feedrate used will depend upon these factors.

For machining center use, the feedrates given in the tables will have to be converted to IPM values. To accomplish this, the following formula is used:

$$IPM = RPM \times IPR$$

Where: IPM = the required feedrate expressed in inches per minute
RPM = the programmed spindle speed in revolutions per minute
IPR = the drill feedrate to be used expressed in inches per revolution

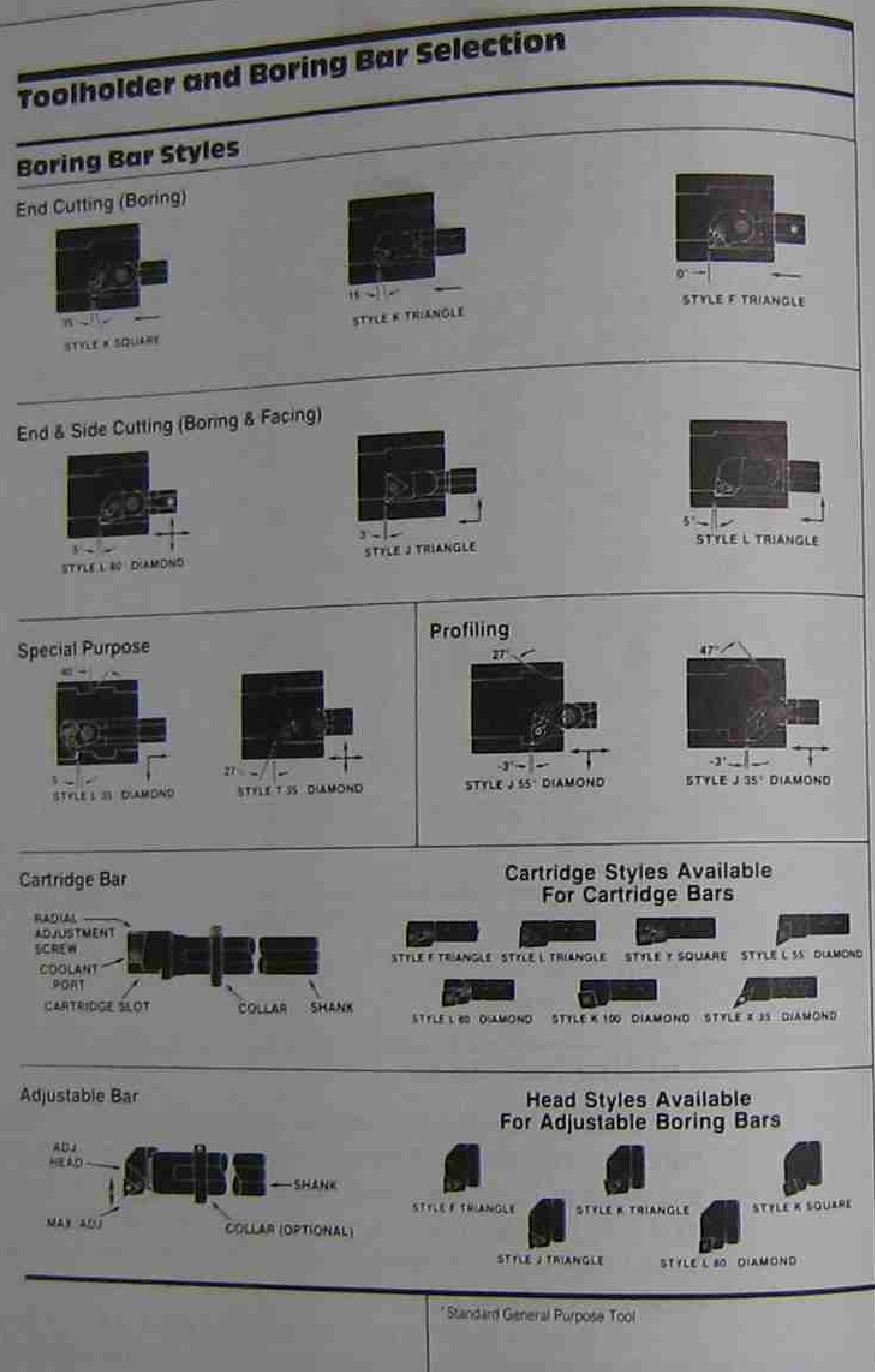


FIGURE 3–27
Toolholder and boring bar sections—boring bar styles *(Courtesy of Carboloy Inc., A Seco Tools Company)*

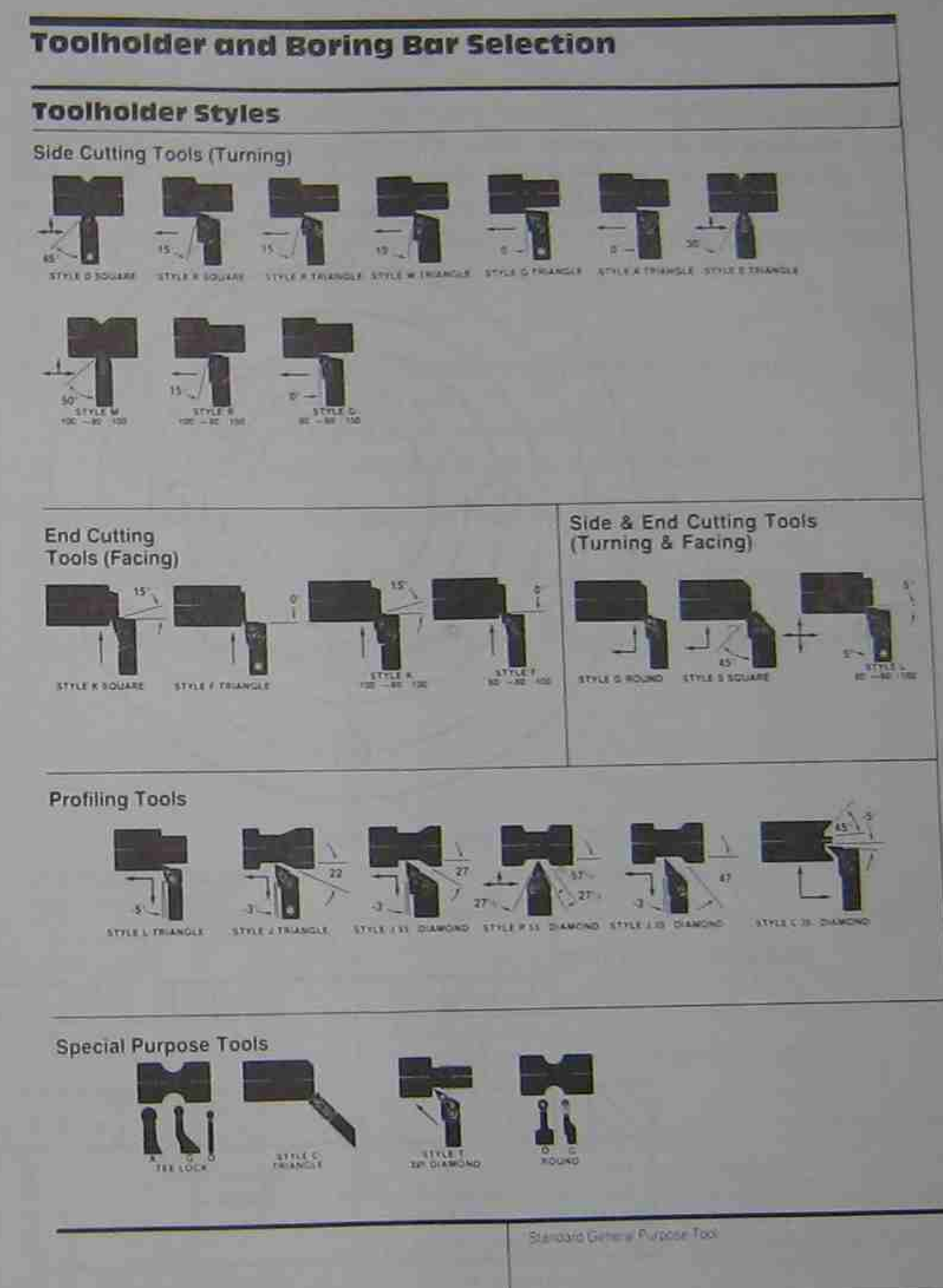


FIGURE 3–28
Toolholder and boring bar selection—toolholder styles *(Courtesy of Carboloy Inc., A Seco Tools Company)*

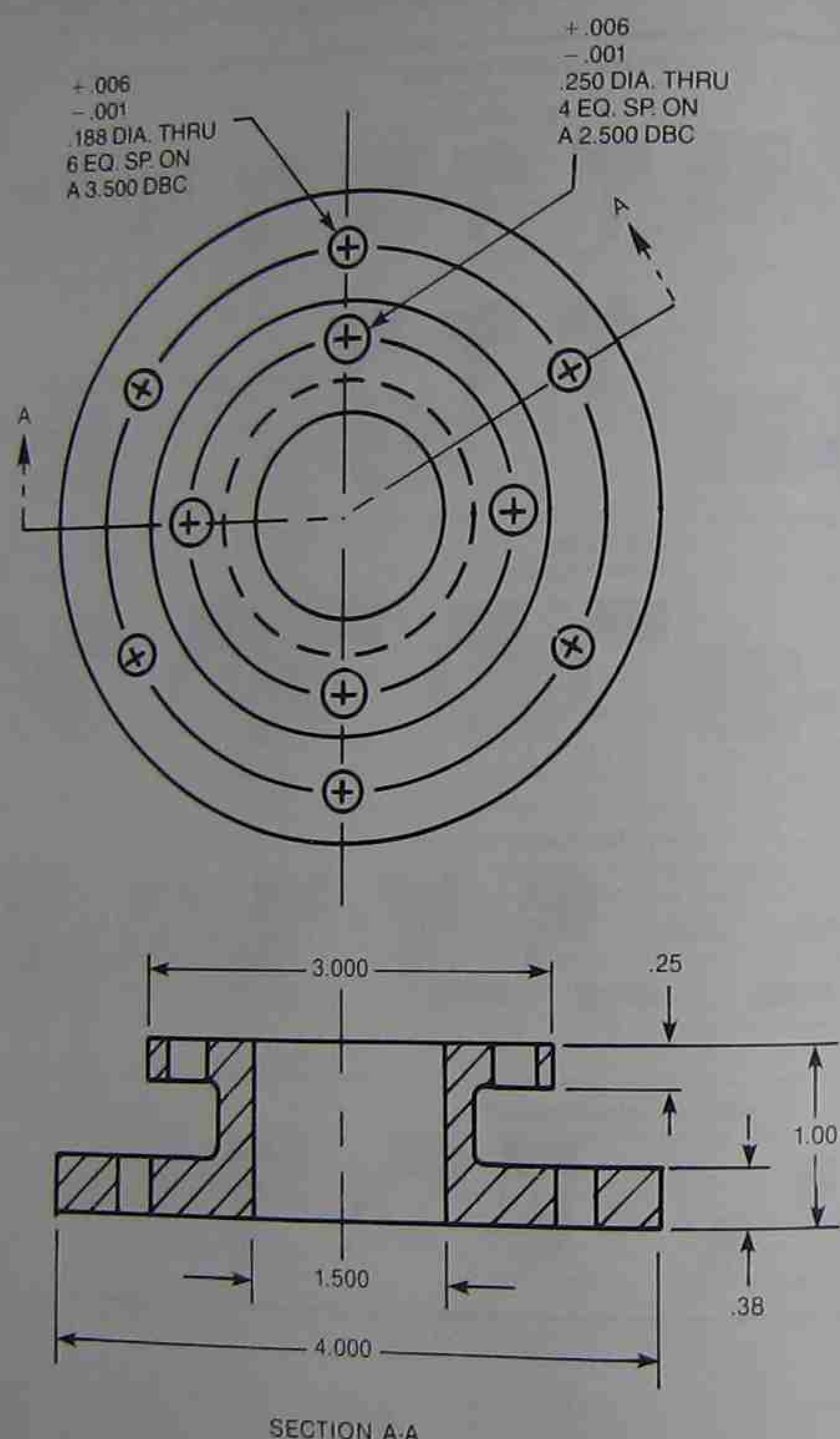


FIGURE 3–29
Part drawing

Milling Feedrates

Feeds used in milling depend not only on the spindle RPM, but also on the number of teeth on the cutter. The milling feedrate is calculated to produce a desired chip load on each tooth of the cutter. In end milling, for example, chip load should be .002 to .006 inch per tooth. The recommended chip loads for various mill cutters are given in machinists' handbooks. For inserted cutters, the insert manufacturer's catalog will list recommended chip loads for a given insert. To calculate the feedrate for a mill cut, the following formula is used:

$$F = R \times T \times RPM$$

Where: F = the milling feedrate expressed in inches per minute
R = the chip load per tooth
T = the number of teeth on the cutter
RPM = the spindle speed in revolutions per minute.

Milling feedrates are also affected by machine and setup rigidity, and by part geometry.

In the case of inserted milling cutters, there is another factor which affects feedrates: chip thickness. This is not the chip load on the tooth, but the actual thickness of the chip produced at a given feedrate. Chip thickness will vary

MANUFACTURING PROCESS

Part Number: Adapter
Run Quantity: 200
Job Number: 000-000-001
Material: Alum. Casting.

OPERATION NUMBER	OPERATION CODE	DESCRIPTION OF OPERATION
010	issue	Issue 356 alum. castings
020	manual lathes	Chuck on 3.250 as cast dia. • Turn 4.000 ± .010 b/p dim to 4.000 ± .001 dia. (tooling dimension). • Face .38 b/p dim.
030	vert. mach. center	Locate parts in fixture NCF-000-100 • Drill .188 +.006 −.001 dia. thru 6 plcs. • Drill .250 +.006 −.001 dia. thru 4 plcs. • Bore 1.500 ±.010 dia. thru 1 plc. • Mill the 3.000 ±.010 dia., hole the 1.000 and .25 dims.
040	burr	• Deburr parts as required.
050	insp	• Inspect parts for b/p conformance.

FIGURE 3–30
Manufacturing process for part shown in Figure 3–29

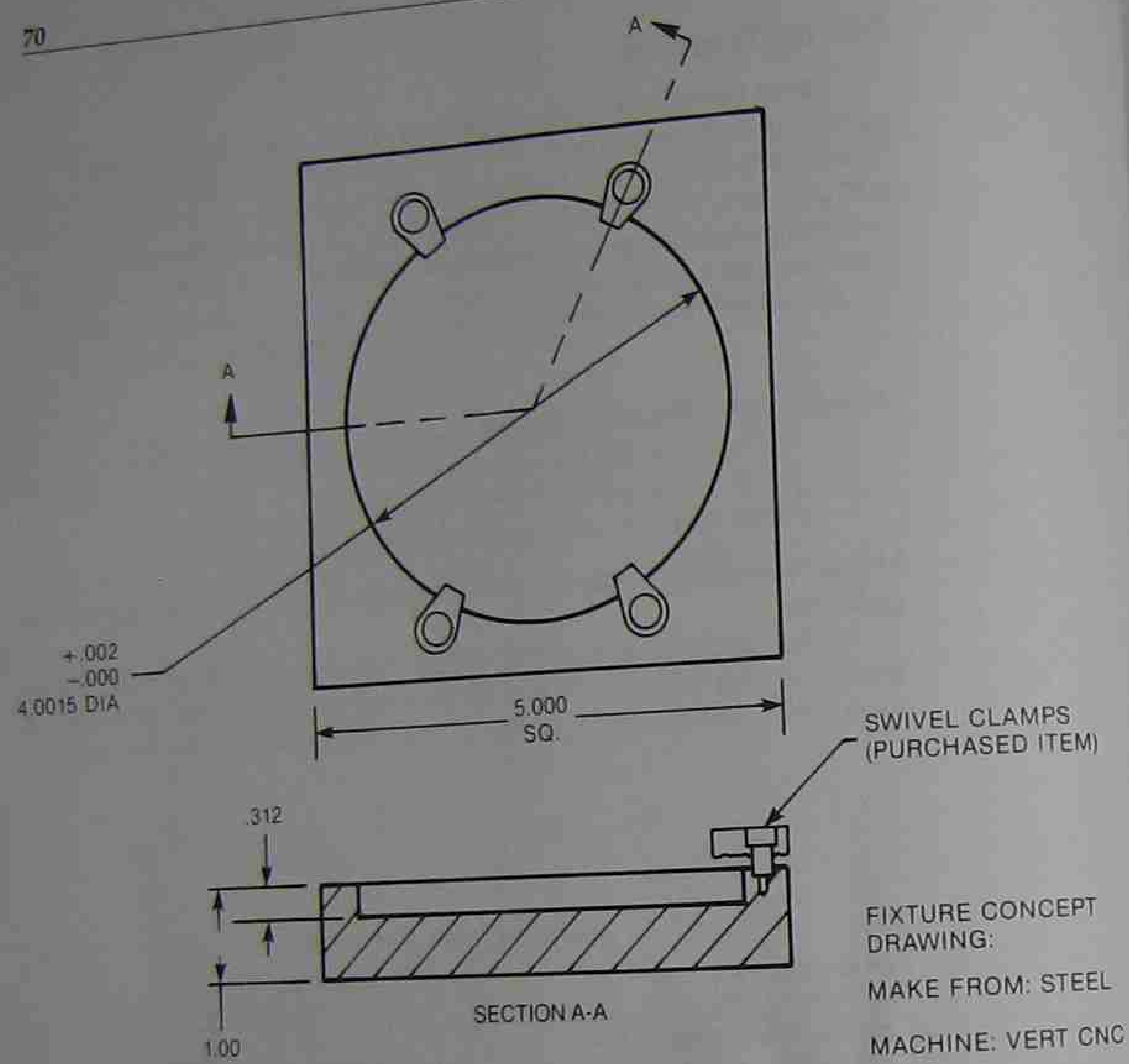
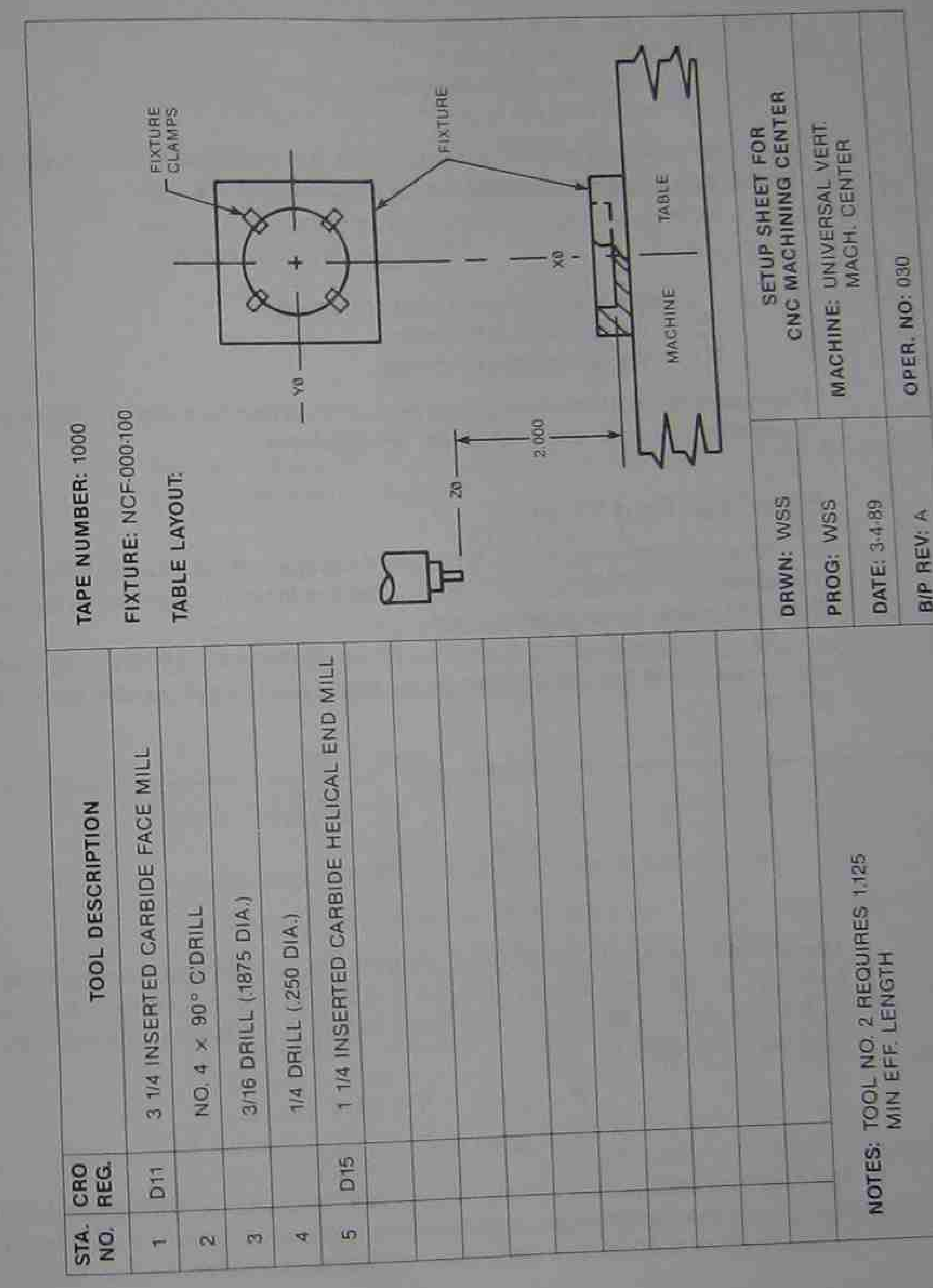


FIGURE 3–31
Fixture concept

some with the geometry of the cutter (positive rake, negative rake, neutral rake), but should be maintained in the range of .004 to .008 inch. Chip thickness less than or greater than these values will place either too little or too great a pressure on the insert for efficient machining. Once a feedrate has been calculated, the chip thickness it produces should be derived. If chip thickness is out of the recommended range, the feedrate should be adjusted to bring it to acceptable limits.

Chip thickness can be calculated by the following formula:

$$CT = \sqrt{\frac{W}{D}} \times R$$



STA. NO.	CRO REG.	TOOL DESCRIPTION
1	D11	3 1/4 INSERTED CARBIDE FACE MILL
2		NO. 4 × 90° C'DRILL
3		3/16 DRILL (.1875 DIA.)
4		1/4 DRILL (.250 DIA.)
5	D15	1 1/4 INSERTED CARBIDE HELICAL END MILL

NOTES: TOOL NO. 2 REQUIRES 1.125 MIN EFF. LENGTH

TAPE NUMBER: 1000
FIXTURE: NCF-000-100
TABLE LAYOUT:

DRWN: WSS
PROG: WSS
DATE: 3-4-89
B/P REV: A

SETUP SHEET FOR CNC MACHINING CENTER
MACHINE: UNIVERSAL VERT. MACH. CENTER
OPER. NO: 030

FIGURE 3–32
NC setup sheet for CNC machining center

Where: CT = the chip thickness
W = the width of the cut
D = the diameter of the cutter
R = the feed per tooth

If the chip thickness is found to be too small, this modification of the preceding formula can be used to determine an acceptable feedrate:

$$f = \sqrt{\frac{D}{W}} \times CT$$

Where: f = the feed per tooth being calculated
D = the diameter of the cutter
CT = the desired chip thickness

This new calculated value of the feed per tooth can then be substituted back into the feedrate formula, and a new feedrate calculated.

Speed and Feed Example

An aluminum workpiece is to be milled using a carbide inserted mill cutter. The cutter is 1.750 diameter × 4 flute. What would be the appropriate spindle RPM and milling feedrate for the workpiece?

An appropriate cutting speed (SFM) for aluminum is 1000 surface feet per minute. Using this value in the spindle speed formula with a cutter diameter of 1.75:

$$RPM = \frac{1000 \times 12}{1.75 \times 3.1416}$$

$$RPM = \frac{12000}{5.4978}$$

$$RPM = 2{,}183$$

The feedrate can now be determined using the feedrate formula. The machinist handbook's tables give a recommended chip load of .002 to .006 inch. A value of .004 per tooth is selected. Using these values in the feedrate formula, the feedrate is calculated.

$$F = 2{,}183 \times 4 \times .004$$

$$F = 34.91 \text{ inches per minute}$$

The chip thickness is then calculated to insure the inserts will not break down prematurely. For this example, it will be assumed the width of the cut to be taken is 1.000 inch wide. Using this value, the chip thickness is determined as follows:

$$CT = \sqrt{\frac{1.000}{1.750}} \times .004$$

$$CT = .755 \times .004$$

$$CT = .00302$$

The chip thickness is less than the recommended minimum of .004. The feed per tooth is therefore calculated as using the feed per tooth formula. A chip thickness of .008 is used.

$$f = \sqrt{\frac{1.75}{1.000}} \times .008$$

$$f = 1.3229 \times .008$$

$$f = .010$$

The new value for the chip load per tooth is substituted in the feedrate formula, and the feedrate recalculated.

$$F = 2183 \times 4 \times .010$$

$$F = 87.32 \text{ inches per minute}$$

The 2183 RPM spindle speed and 87.32 inches per minute feedrate are "book value" rates. They will have to be adjusted up or down depending on the machine, fixture, tool, and workpiece rigidity.

SUMMARY

The important concepts presented in this chapter are:

- Process planning is the term used to describe the steps the programmer uses to develop and implement a part programming.
- The steps in process planning are: determine the machine, determine the workholding, determine the machining strategy, select the tools to be used.
- Tool selection is important to the efficiency of the NC program.
- Cutting tools for NC are made in high speed steel, tungsten carbide, and ceramic.
- Inserted cutters are the preferred tools for NC use.
- Inserts are manufactured in different grades with different applications intended.
- Cutting speed is the edge speed of the tool; it is a function of the spindle RPM and the tool diameter.

- Feedrates that are too heavy will result in excess tool wear and premature tool failure.
- Feedrates that are too light will result in chipping of tools and premature tool failure.
- When calculating milling feedrates, chip thickness must be considered.

REVIEW QUESTIONS

1. What is process planning?
2. List three factors that influence NC machine selection.
3. List three factors used by the programmer in deciding how to hold a workpiece.
4. List two factors that influence the development of machining strategy.
5. List three factors that determine tool selection.
6. What is an NC setup sheet?
7. What are the three basic cutting tool materials?
8. Give one advantage of HSS tools? Give one disadvantage.
9. Give one advantage of carbide tools. Give one disadvantage.
10. On what types of materials are ceramic inserts used?
11. What types of tools are preferred for NC use? Why?
12. What are the four common hole operations performed on NC machines?
13. Why are drilled holes not perfectly straight?
14. Is drill point angle important? Why?
15. Which reamer flute design produces better finishes: straight or spiral?
16. What are the two types of boring head?
17. What is the difference between a tap and a hob?
18. What are the two categories of milling cutters?
19. Into what further classifications can the categories in #18 be put?
20. Why should a programmer be familiar with insert grades?
21. What is cutting speed? Is it the same as the spindle RPM?
22. What happens when a feedrate is too light? too heavy?
23. What two ways are feedrates expressed?
24. How is a milling feedrate calculated?
25. Why should chip thickness be checked after a feedrate is determined?

CHAPTER 4

Tool Changing and Tool Registers

OBJECTIVES Upon completion of this chapter, you will be able to:

- Explain why the speed, repeatability, and accuracy of tool changing are important factors in numerical control.
- Name the two types of tool changes.
- Explain why quick-change tooling is used on NC mills.
- Explain how tooling is used in automatic tool change functions.
- Name the five types of automatic tool changers and briefly describe the operation of each.
- Describe the two basic methods of tool storage.
- Explain what tool registers are and what they are used for.
- Describe what tool offset length is and how it is determined.
- Explain how tool offsets may be entered by the operator during setup and how the programmer allows for this.

This chapter deals with CNC tool changing and tool registers. A good general understanding of these subjects is required for three-axis CNC programming.

TOOL CHANGES

There are two types of tool changes: *manual* and *automatic*. When referring to CNC mills, tool changing is understood to be manual unless otherwise stated. A machining center, on the other hand, incorporates automatic tool change (ATC). It is the tool-changing capability that separates the CNC machining center from CNC milling machines. Machining centers, like milling machines, have the capability to do numerous machining operations (drilling, tapping, spotfacing, and milling, among others). This is opposed to a machine capable of a single function only, such as an NC drilling machine. Figure 4–1 illustrates a CNC milling machine; Figures 4–2 and 4–3 illustrate CNC ma-

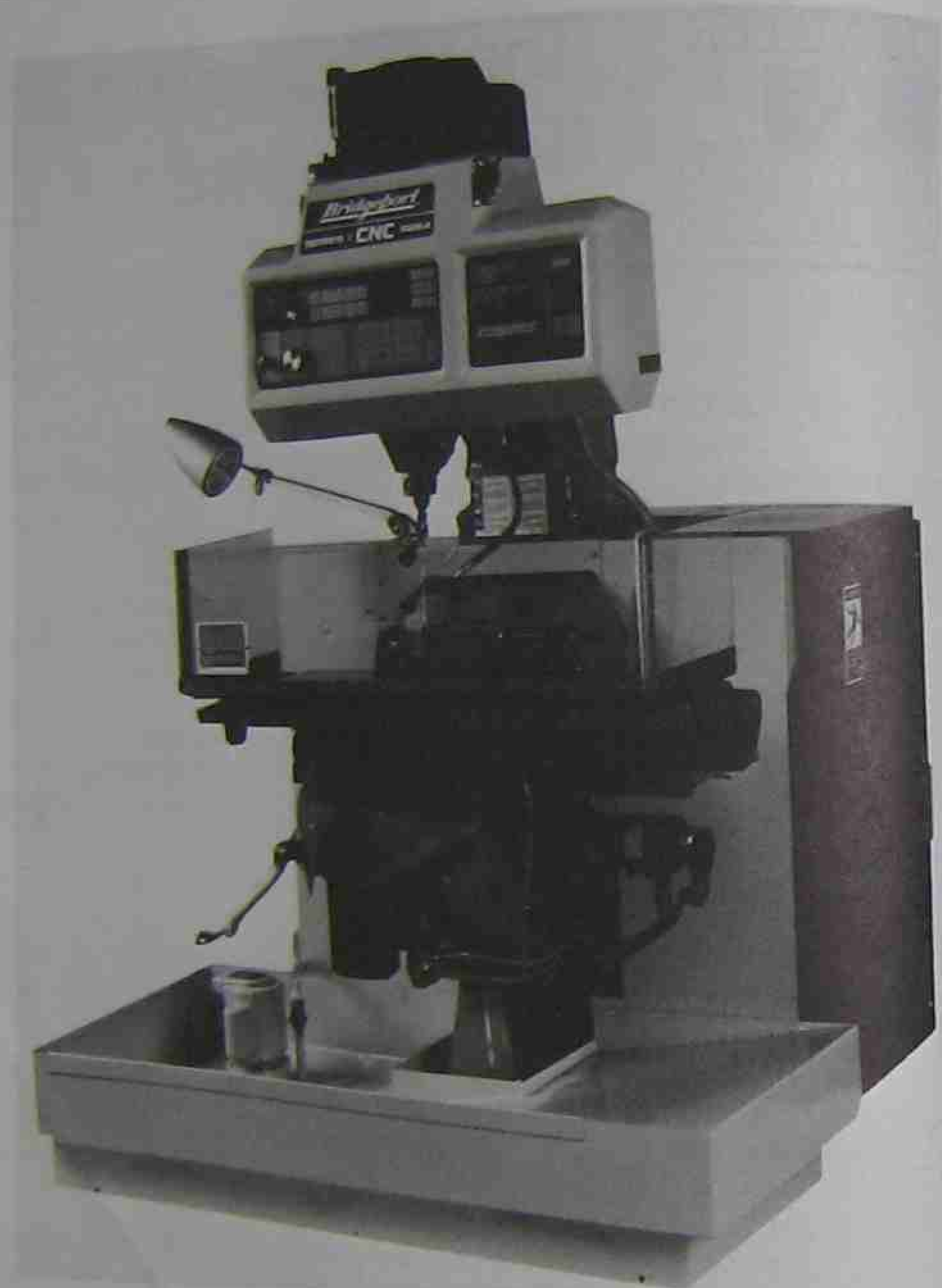


FIGURE 4–1
A vertical spindle CNC milling machine. Note the quick-change tooling system installed in the spindle. *(Photo courtesy of Bridgeport Machines Division of Textron Inc.)*

chining centers. Notice the presence of the tool changer on the machining centers. Also note that a machining center may have either a horizontal or vertical spindle just like a milling machine.

Tooling for Manual Tool Change

What is to be gained by the speed with which a CNC machine can position itself for hole drilling if the tool changes are so lengthy as to cancel the time and accuracy gained by using numerical control? Tool changing greatly influences the efficiency of numerical control, so tool changes should take place as quickly as is safely possible. The tool must not only be (1) accurately located in the spindle to assure proper machining of the workpiece, but (2) the tool must be located as accurately as possible in the same location and (3) in the same relationship to the workpiece each time it is inserted in the spindle. This is known as *repeatability* of a tool—the ability to locate, or repeat, its position in the spindle each time it is used.

Numerical control mills (manual tool change) usually are supplied with or have had added to them some type of quick-change tooling system to accomplish this task.

Most small vertical turret mills are manufactured with what is known as an R-8 spindle taper, which will accept R-8 collets. Figure 4–4 depicts an R-8 spindle and collet. The CNC milling machine in Figure 4–1 has an R-8 spindle employing a quick-change tool-changing system. The R-8 collet is a standard

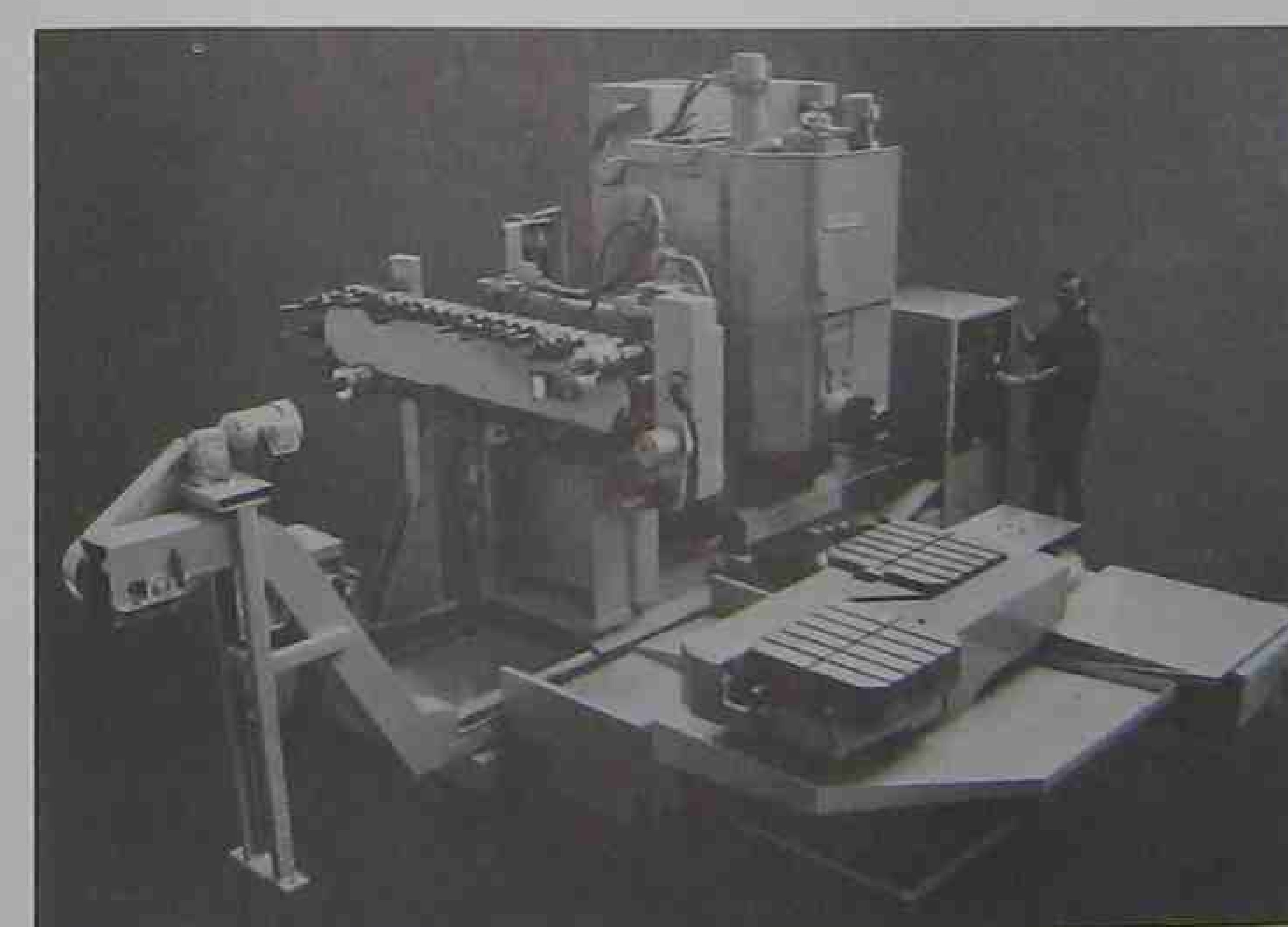


FIGURE 4–2
A horizontal CNC machining center employing automatic tool change. Note the pivot insertion tool changer on the side. Tools are stored in a matrix magazine. Safety guards have been removed for clarity. *(Photo courtesy of Cincinnati Milacron)*

collet on Bridgeport vertical mills. Since most vertical turret mills are spin-offs of this design, the R-8 spindle has become psuedo-standard on these machines.

R-8 collets and R-8 tool holders require the use of a drawbar. For CNC use, either an automatically tightening drawbar is suppied with the machine, or a quick-change tool system is added.

The quick-change tool system consists of a quick release chuck (which is held in the machine spindle) and a set of tool holders that hold the individual tools needed for a particular part program. The chuck is a separate tool-holding mechanism that stays in the spindle. During a tool change, the tool holder is removed from the chuck (sometimes called the tool changer), and a tool holder containing the next required tool is installed in its place. The tools placed in the tool holders are securely held by means of set screws. Many varieties of these quick-change tool systems are available on the market. Figure 4–5 illustrates a quick-change tooling system in action.

Larger vertical mills and most horizontal mills use another type of spindle taper, called the American Standard Milling Machine Taper (Figure 4–6). Like the R-8 taper, this taper requires the use of a drawbar. If no automatic drawbar is supplied with a machine, a quick-change tooling system is added to the machine to improve tool changing.

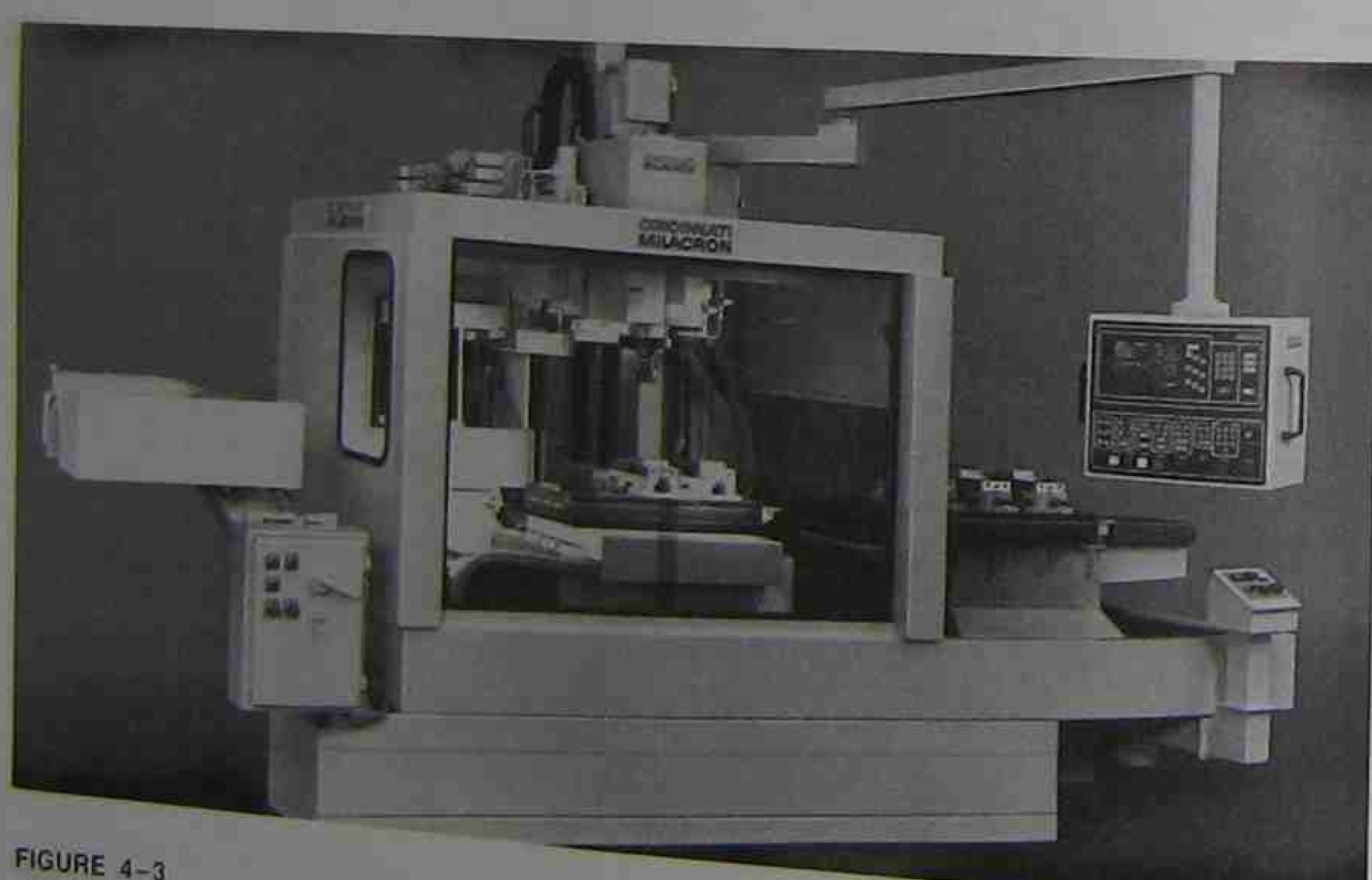


FIGURE 4–3
A vertical spindle CNC machining center *(Photo courtesy of Cincinnati Milacron)*

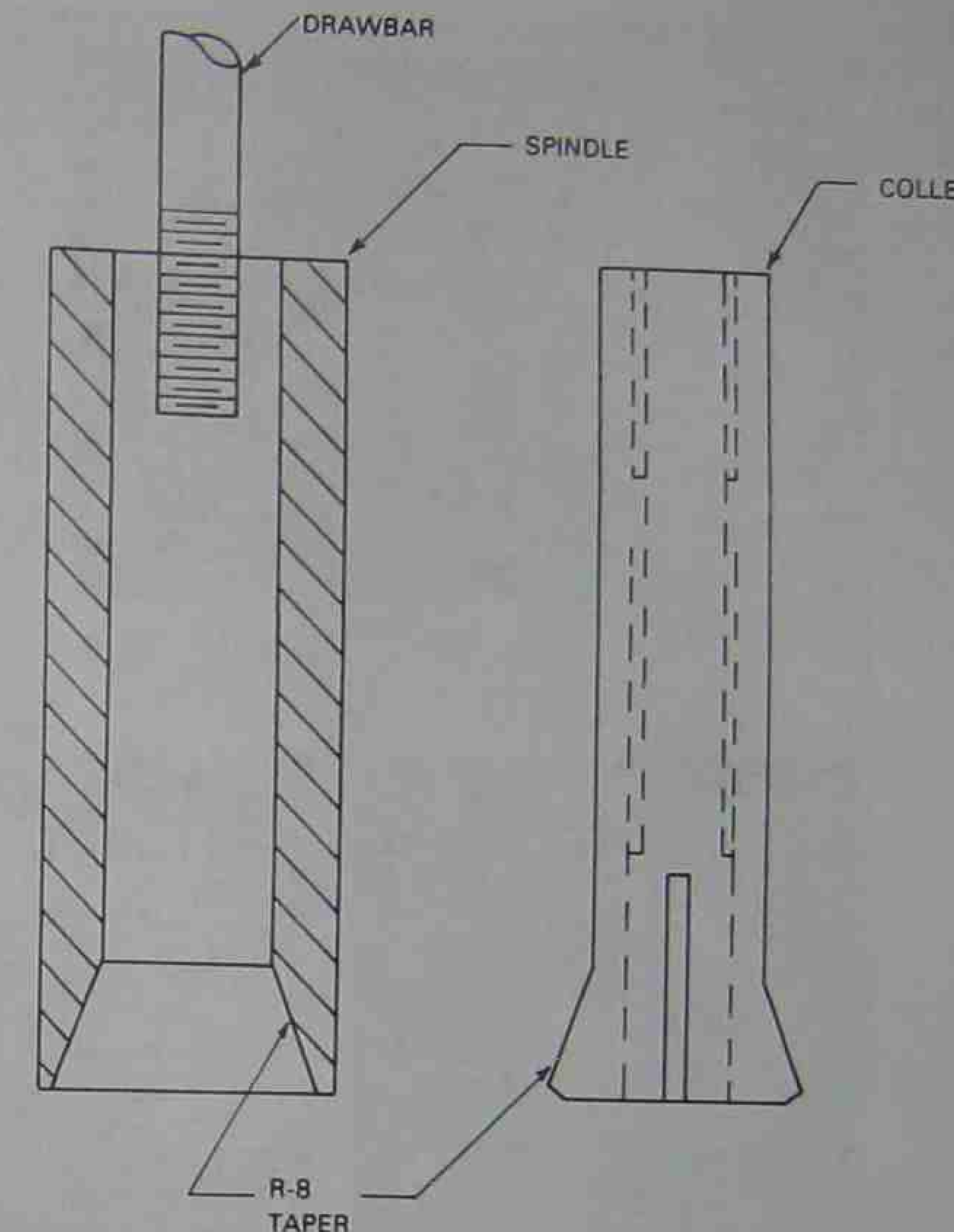


FIGURE 4–4
R-8 spindle and collet

Tooling for Automatic Tool Change

When automatic tool change is used, the requirements for speed and repeatability are even more critical. The machine's tool changer cannot think and correct for misalignment of tooling or tool setup errors like a human being. The tool changer will faithfully carry out its tool-changing cycle and nothing else (that's all it was programmed to do). Tooling used with a tool changer, therefore, must be (1) easy to center in the spindle; (2) easy for the tool changer to grab; and (3) have some means providing for the safe disengagement of the tool changer from the tool once secured in the spindle. Figure 4–7 depicts a common type of tool holder used with ATC (automatic tool change).

The tool changer grips the tool at point A in Figure 4–7 and places the tool in position, aligned with the spindle. The tool changer will then insert the tool

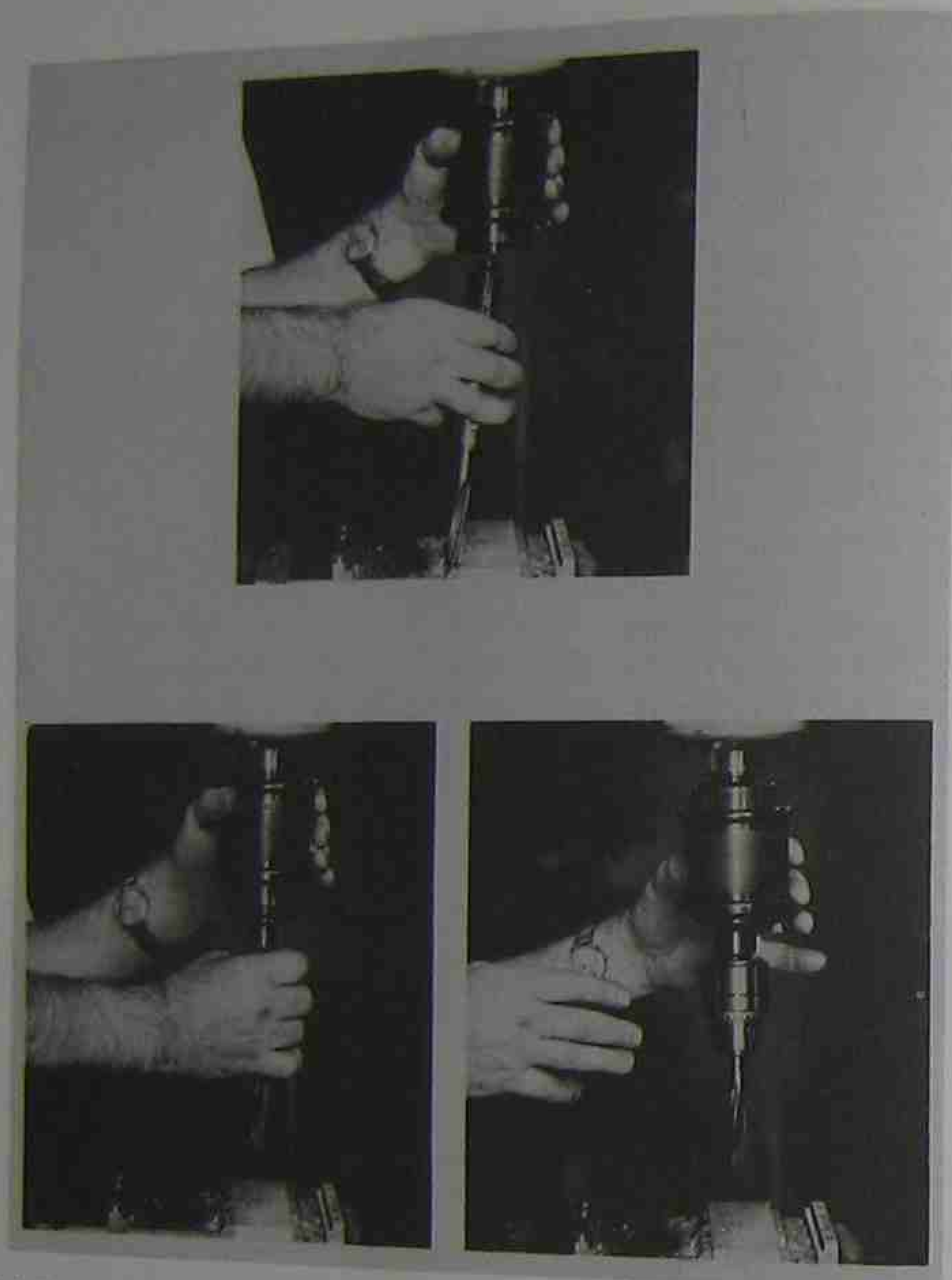


FIGURE 4–5
A quick-change tooling system used for manual tool change *(Photo courtesy of Immotion Quick Change Tool Systems)*

into the spindle. In some cases, insertion of the tool is accomplished by the spindle descending over the tool. As the tool engages the spindle, a split bushing in the spindle will close on the *tool retention knob* (point B in Figure 4–7). This split bushing holds the tool so that the tool changer can release its grip on the tool. The tool is then drawn completely up into the spindle and tightened. Using this procedure insures proper alignment of the tool with the spindle and prevents damage from occurring to the spindle or tool holder taper. Figure 4–8 shows tool insertion using a split bushing.

Another insertion method can be used with a different type of tool holder (Figure 4–9). Here, the tool changer grips the tool in slot A. After the tool is inserted into the spindle, the tool changer moves toward the spindle as the tool is drawn up into the spindle. When the tool is secured in the spindle, the tool changer slides off the tool holder from the side.

AUTOMATIC TOOL CHANGERS

Automatic tool changers, while varied, are made in five basic types: turret head, 180-degree rotation, pivot insertion, multi-axis, and spindle direct. Tools used in automatic tool change are secured in tool holders designed for that pur-

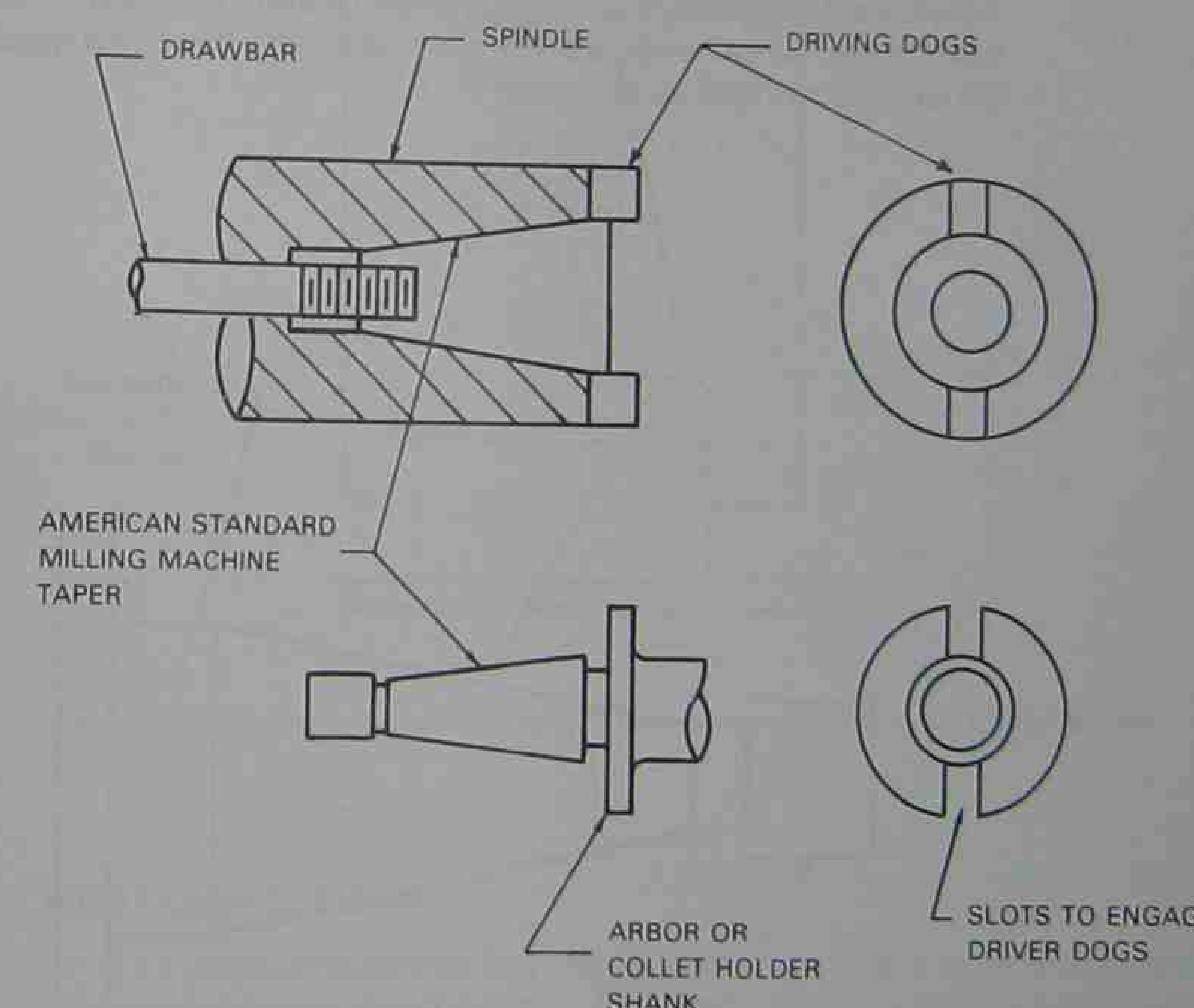


FIGURE 4–6
American Standard Milling Machine Taper used on spindle and arbor (or collet holder shank)

pose. These tool holders are installed directly in the spindle at each tool change by the tool changer. An assortment of tools and tool holders used with CNC machining centers is shown in Figure 4–10.

Turret Head

Tool changing accomplished through the use of a turret head is perhaps the oldest form of automatic tool change. A *turret head* is a number of spindles linked to the same milling machine head, as depicted in Figure 4–11. The tools are placed in the spindles prior to running the program. When another tool is needed, the head *indexes* (moves) to the desired position.

The main disadvantage of this system is the limited number of tool spindles available. In order to use a greater number of tools than available spindles, the operator must remove tools that have already been used and insert those called for later in the program. While other tool-changing methods require less machine operator attention once the program is running, no tool removal is actually performed during the tool change. This results in a very quick tool change. Turret heads are still being used today on certain types of NC machinery such as drilling machines.

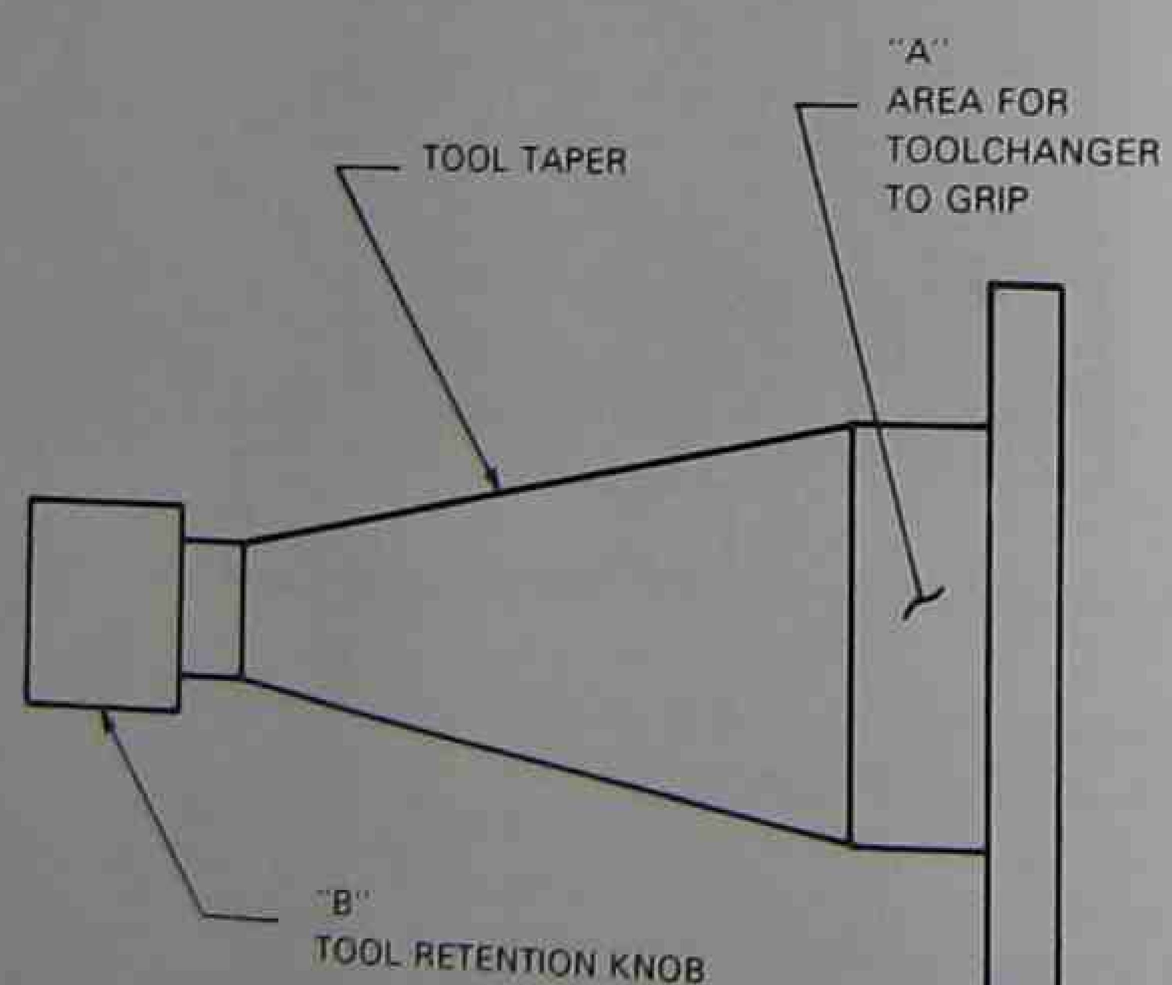


FIGURE 4–7
Typical toolholder used with ATC

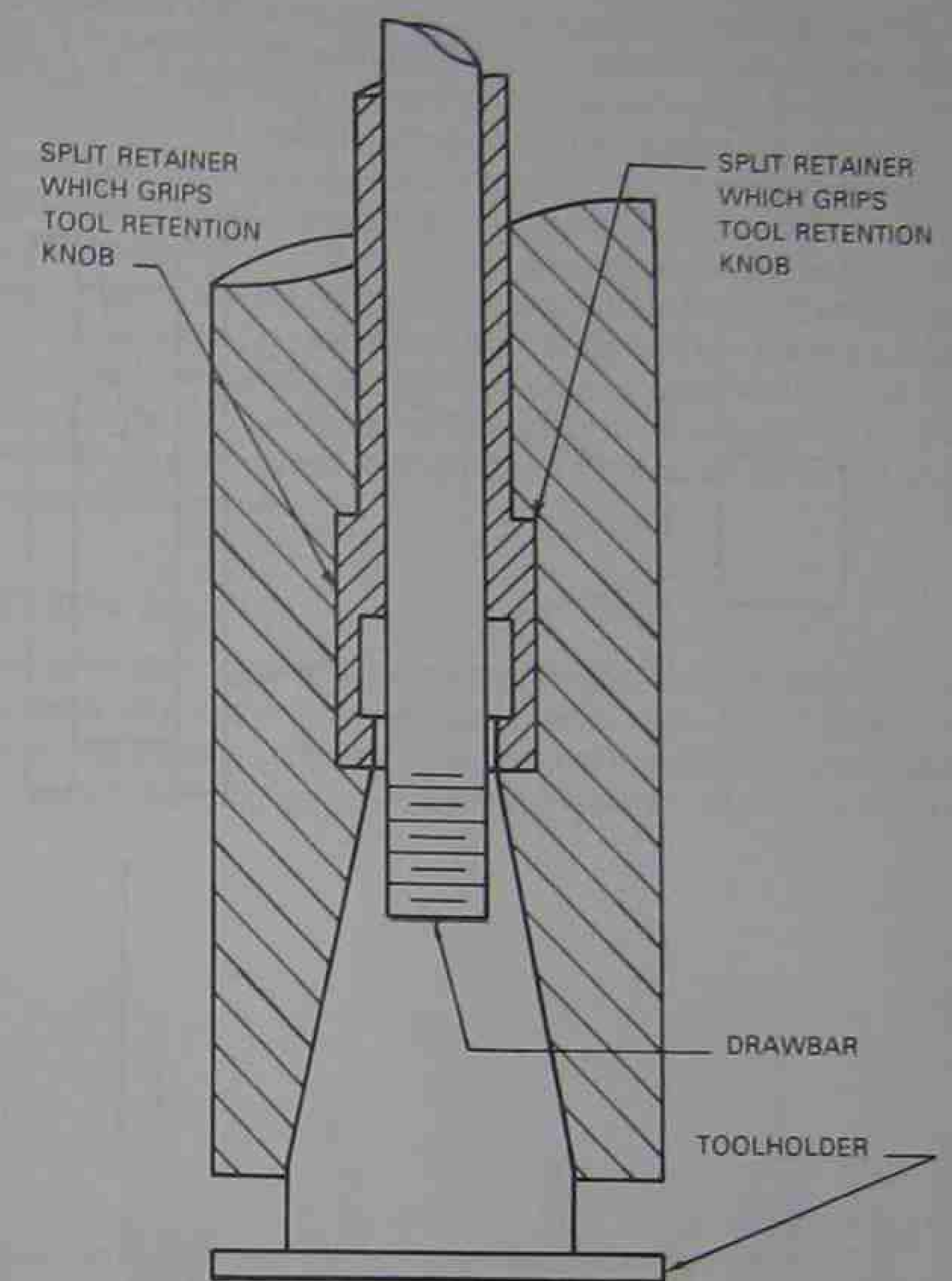


FIGURE 4–8
Split bushing closes over the retention knob to secure the tool as it is drawn into the spindle

180-Degree Rotation

The simplest of the true tool-changing mechanisms is the *180-degree rotation* tool changer (see Figure 4–12). Upon receiving a tool change command, the machine control unit sends the spindle to its fixed tool change coordinates. At the same time, the tool magazine is indexed to the proper position. The tool changer then rotates and engages both the tool in the spin-

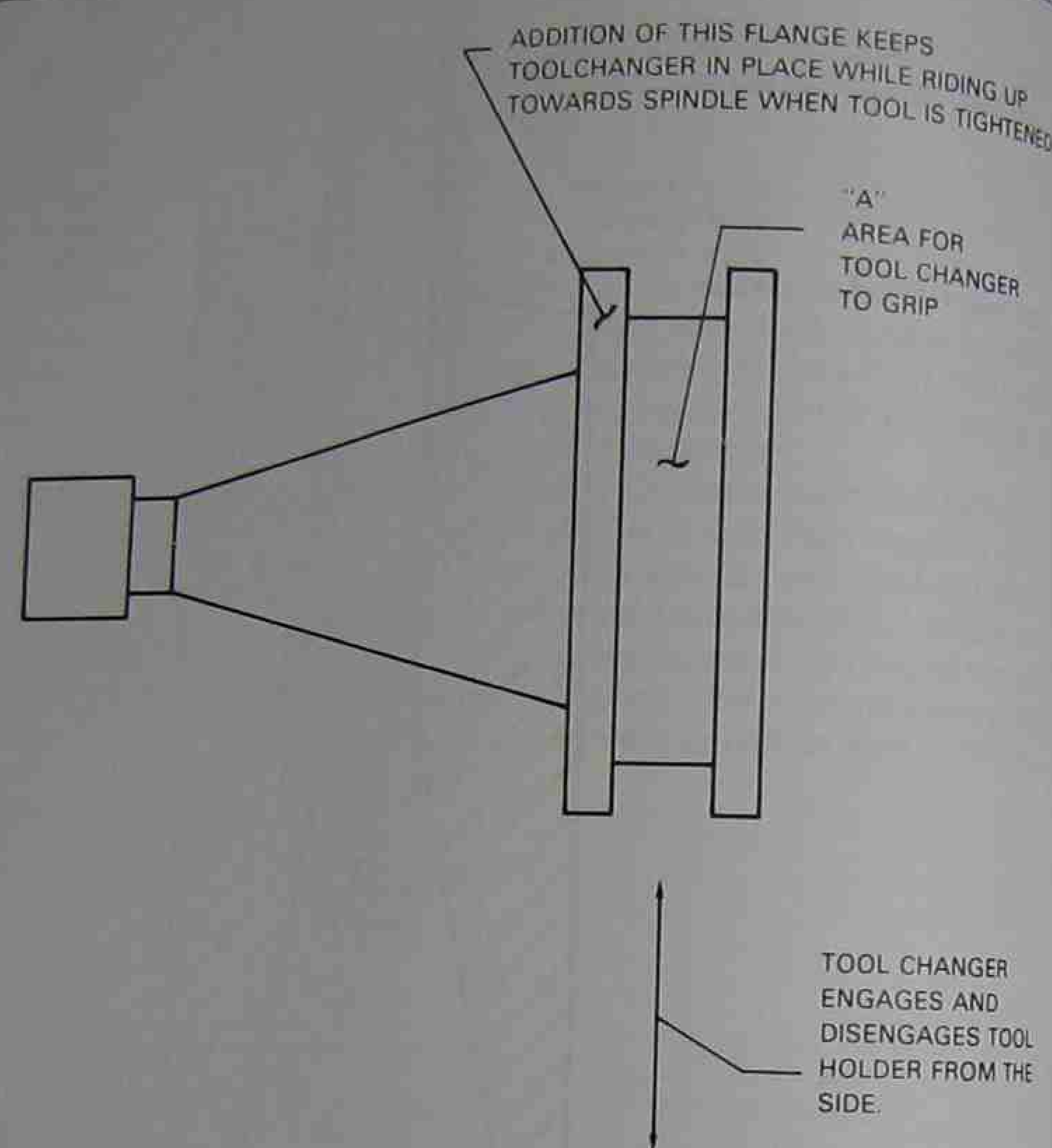


FIGURE 4–9
Tool changer moves in from the side to grip the toolholder in area A while the tool is secured in the spindle.

dle and the tool in the magazine at the same time. The drawbar is removed from the tool in the spindle, and the tool changer removes both tools from their respective places. The tool changer then rotates 180 degrees and swaps the tool that was in the spindle with the one that was in the magazine. While the tool changer is rotating, the magazine repositions itself to accept the old tool that was removed from the spindle. The tool changer then installs the new tool in the spindle and the old tool in the magazine. Finally, the tool changer rotates back to its "parked" position where it remains until needed. The tool change is thus complete and the program continues.

The principal advantage of this type of changer is its simplicity. The amount of motion involved is minimal and tool changes are fast. The principal disadvantage is that the tools must be stored in a plane parallel to the spindle. The chances of chips and coolant getting on the tool holders are greatly increased compared to those in *side-* or *back-mounted magazines*. Extra protection for the tools must, therefore, be provided. Chips on the tool holder taper will also cause an inaccurate tool change, possibly damaging both the tool holder and the spindle. Some machining centers employ a transfer arm that allows the tool magazine to be stored on the side of the machine. When the tool change command is issued, the transfer arm removes the tool from the magazine and pivots to the front of the machine, positioning the tool to be engaged by the tool changer. The 180-degree rotation tool changer may be used on either horizontal or vertical spindle machines.

Pivot Insertion

An adaptation of the 180-degree rotation tool changer is the *pivot insertion* tool changer (one of the most popular types in use). A pivot insertion system combines the functions of the tool changer and transfer arm. The operation of a pivot insertion tool changer is depicted in Figure 4–13. Figure 4–14 shows a

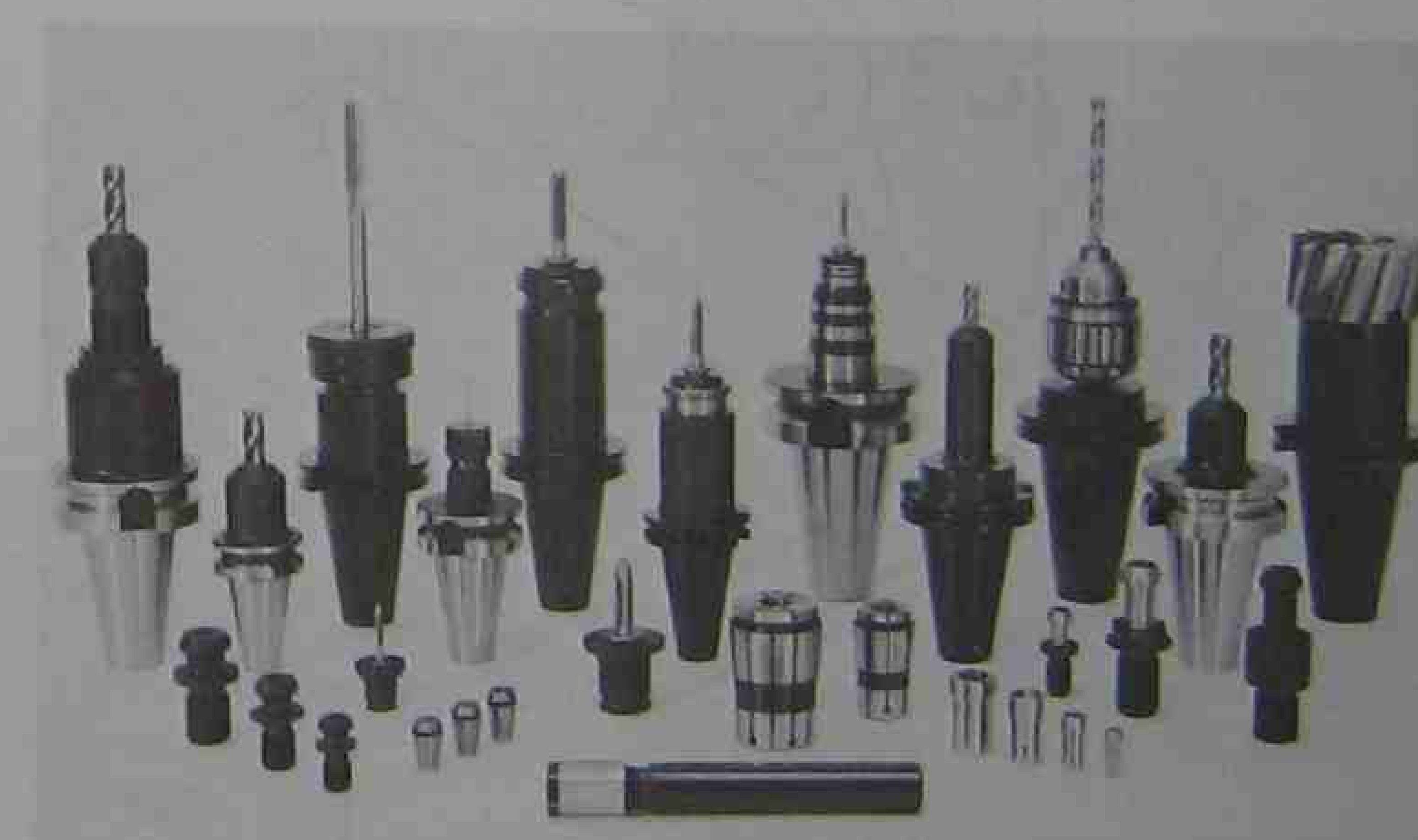


FIGURE 4–10
An assortment of tools and toolholders used with CNC machining center *(Photo courtesy of Command Corporation International)*

pivot insertion tool changer on a horizontal machining center. This tool changer has the same physical design as that of the 180-degree rotation tool changer.

When a tool change command is given, the spindle is sent to the tool change location, and the tool magazine is rotated to the proper location for the tool changer to remove the new tool from its slot. The tool changer rotates and removes the new tool from the magazine, which is located on the side of the machine. The tool changer then pivots around to the front of the machine where it engages and removes the tool from the spindle, rotates 180 degrees, and inserts the new tool in the spindle. During this time, the tool magazine has indexed to the proper position to receive the old tool. The tool changer then

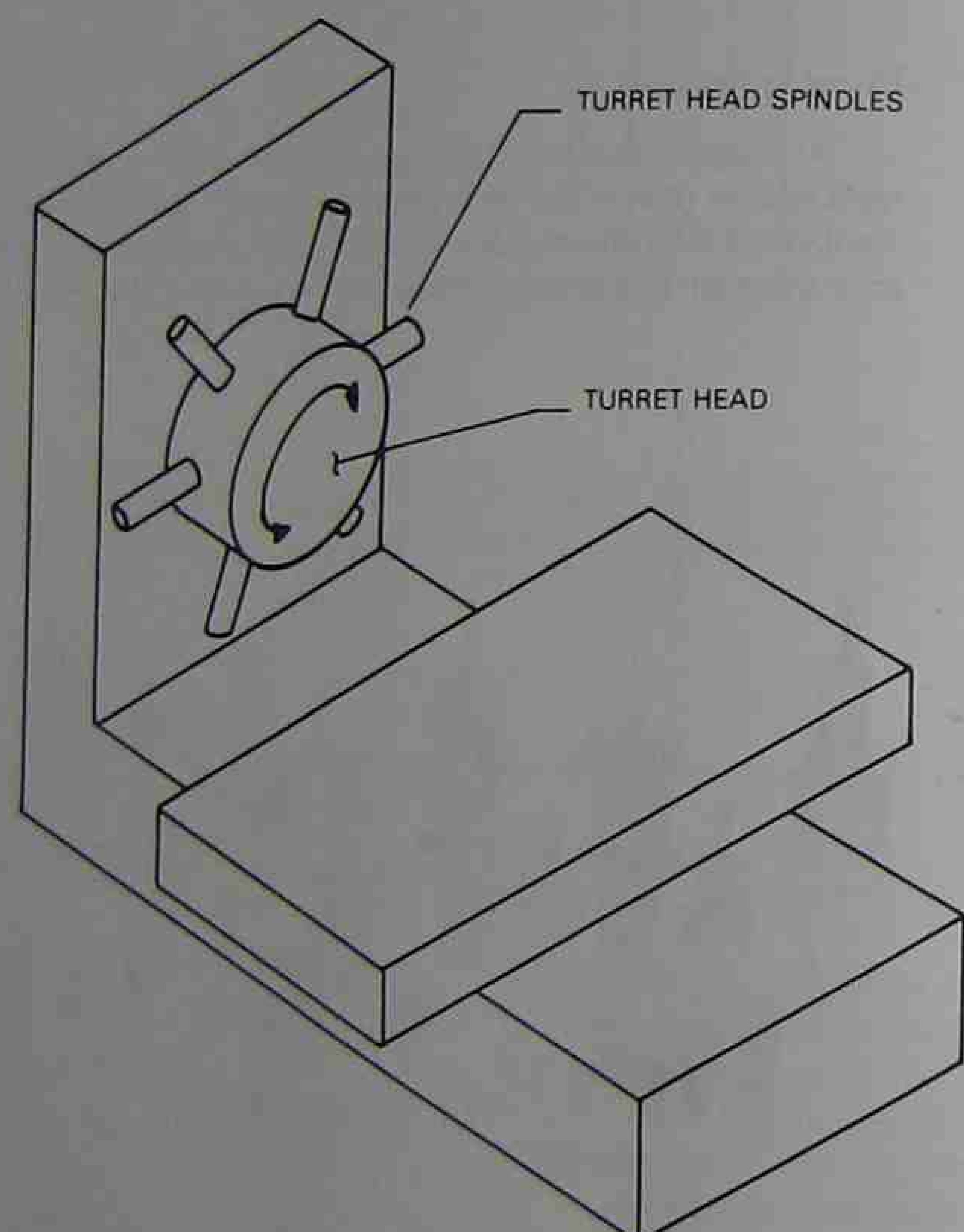


FIGURE 4–11
Turret head tool changer

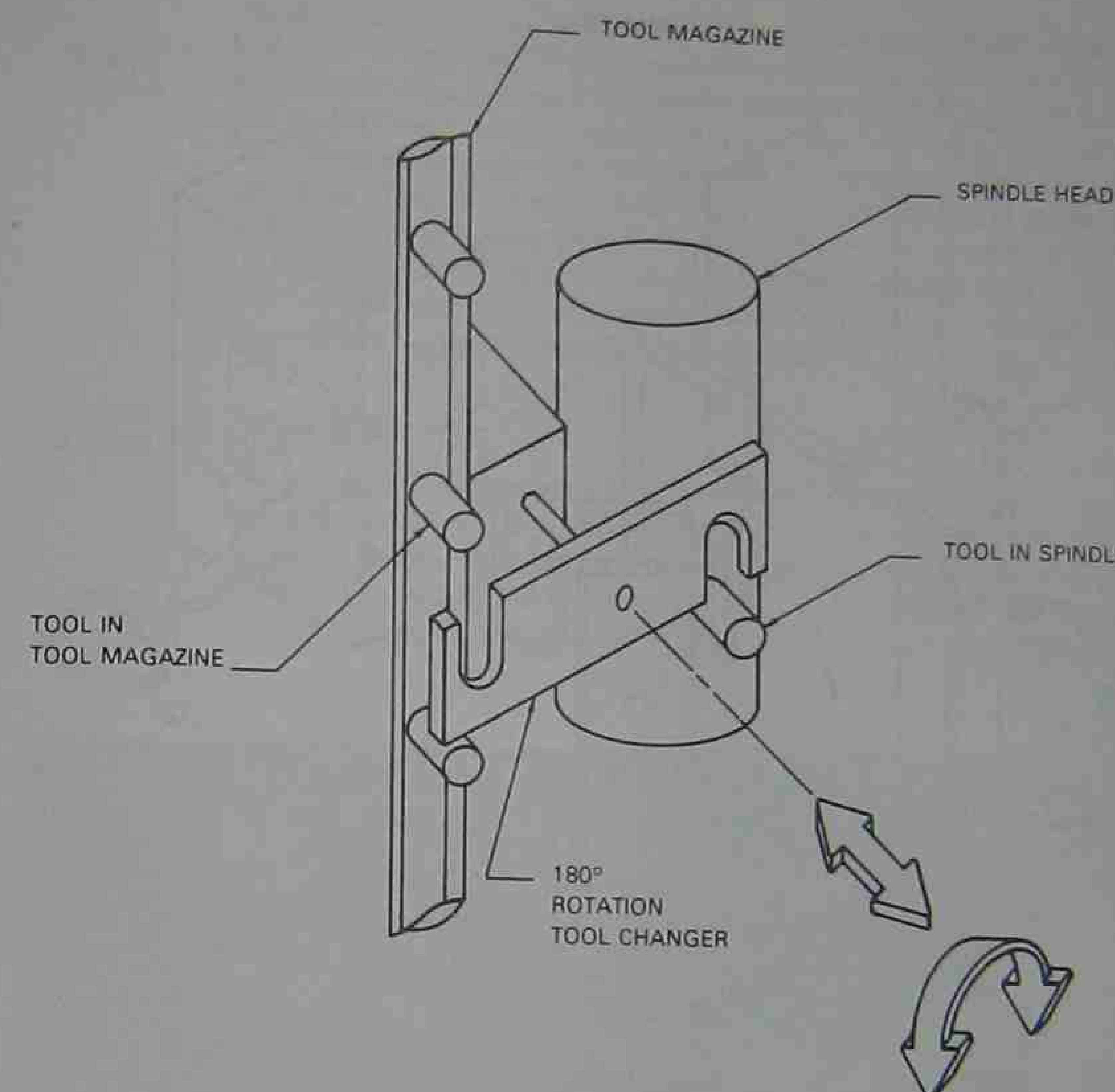


FIGURE 4–12
180-degree rotation tool changer

pivots around to the side of the machine and places the old tool in its slot in the tool magazine. Finally, the tool changer "parks," and the NC program continues.

The main advantage of this system is that the tools may be stored on the side of the machine away from potentially damaging chips. Its disadvantage as compared to the 180-degree rotation tool changer is that pivot insertion requires more motion and therefore results in a more time-consuming tool change.

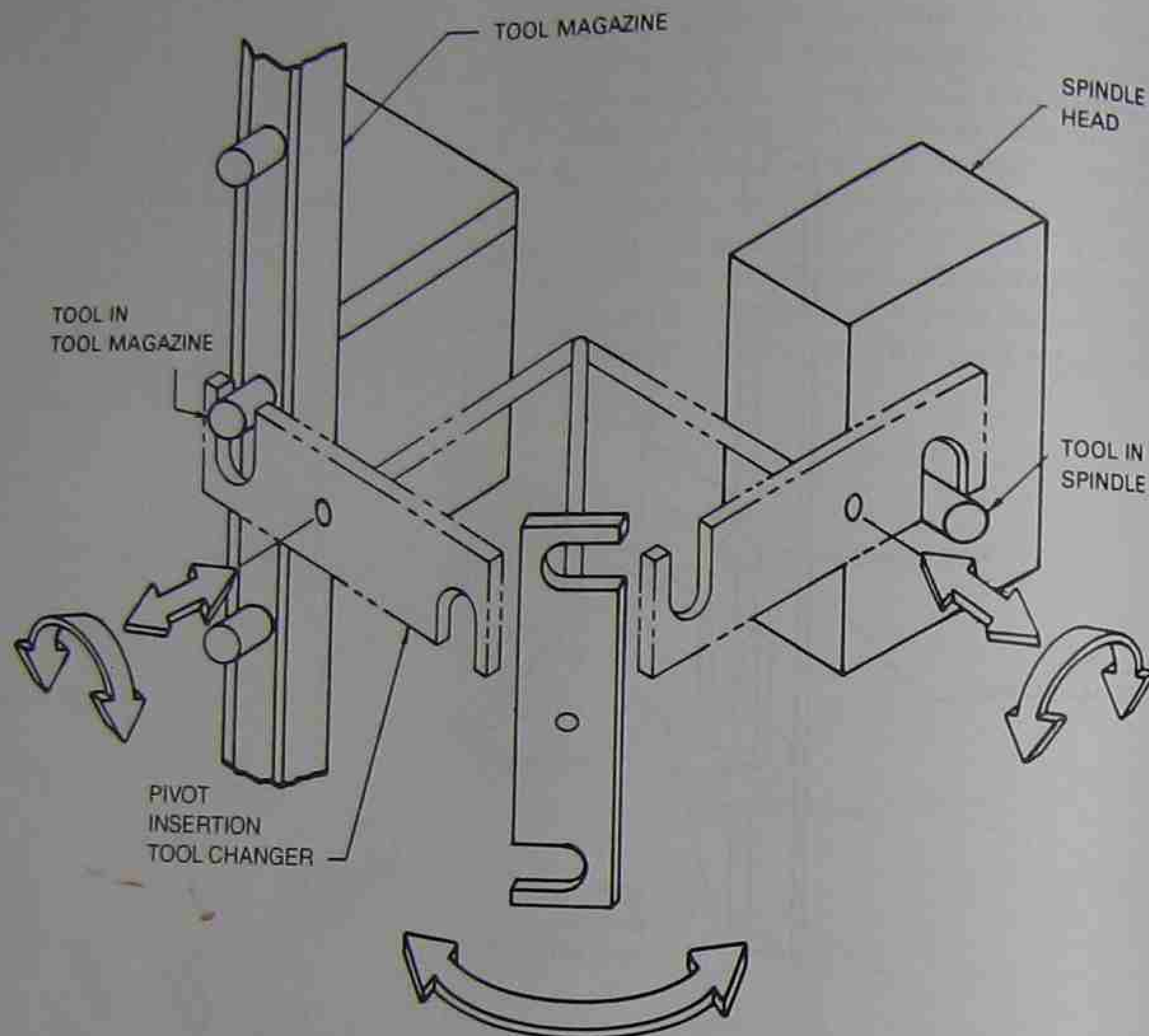


FIGURE 4–13
Pivot insertion tool changer

Multi-Axis

Multi-axis tool change operation is depicted in Figure 4–15. This type of tool changer can be used with either side-mounted or back-mounted tool magazines. Its design lends itself very well to use with vertical spindle machining centers. When given a tool change command, the tool changer moves from its "parked" position, grabs the tool that is in the spindle, and removes it. The tool

changer then swings (or sweeps) back to the tool magazine and places the old tool into the magazine. The changer then removes the desired tool from the magazine, swings around to the spindle again, and installs the tool in the spindle. Finally, the tool changer returns to "park," and the tool change is completed.

The main advantage of this system is the placement of the tool magazine on the back or side of the machine, where maximum protection can be afforded to the tools. Its disadvantage is the amount of tool handling and motion that must be employed. Today, multi-axis tool changers are giving way to other tool-changing mechanisms such as the 180-degree rotation, and, on vertical spindle machining centers, the spindle direct tool changer.

Spindle Direct

Spindle direct tool changing differs from other types of tool changing in that the tool magazine (carousel) moves directly to the machine spindle or vice versa. Figure 4–16 depicts the operation of a spindle direct tool change. Figures 4–17 and 4–18 illustrate vertical machining centers employing spindle direct tool changers. When a tool change is initiated, the spindle is directed to

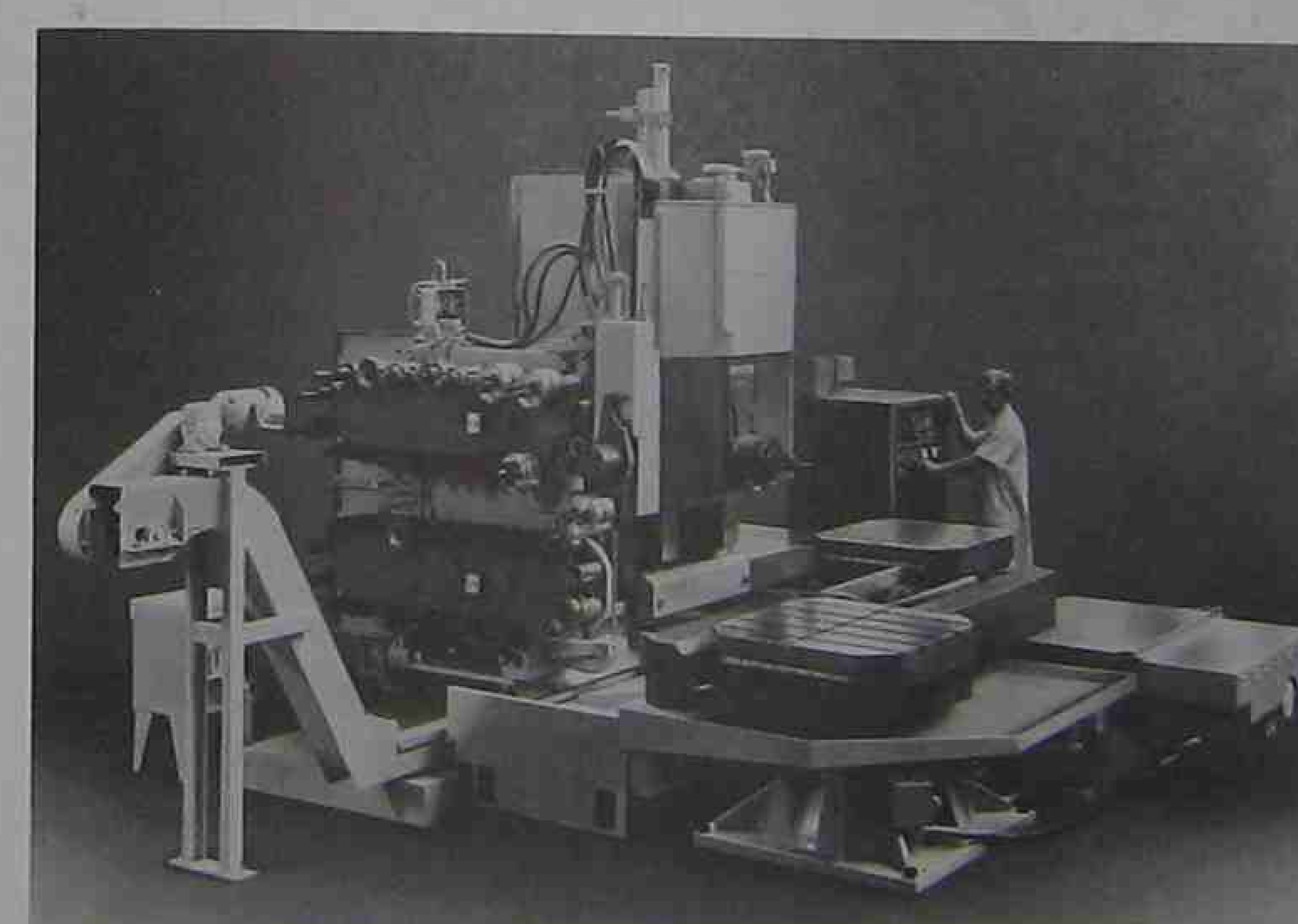


FIGURE 4–14
A pivot insertion tool changer on a horizontal machining center using twin matrix tool storage magazines. Guards have been removed for clarity. *(Photo courtesy of Cincinnati Milacron)*

the tool change location. The tool carousel indexes to the required tool slot, moves out of its "parked" position to the tooling position, and engages the tool-holder that is in the spindle. The drawbar is then removed from the toolholder,

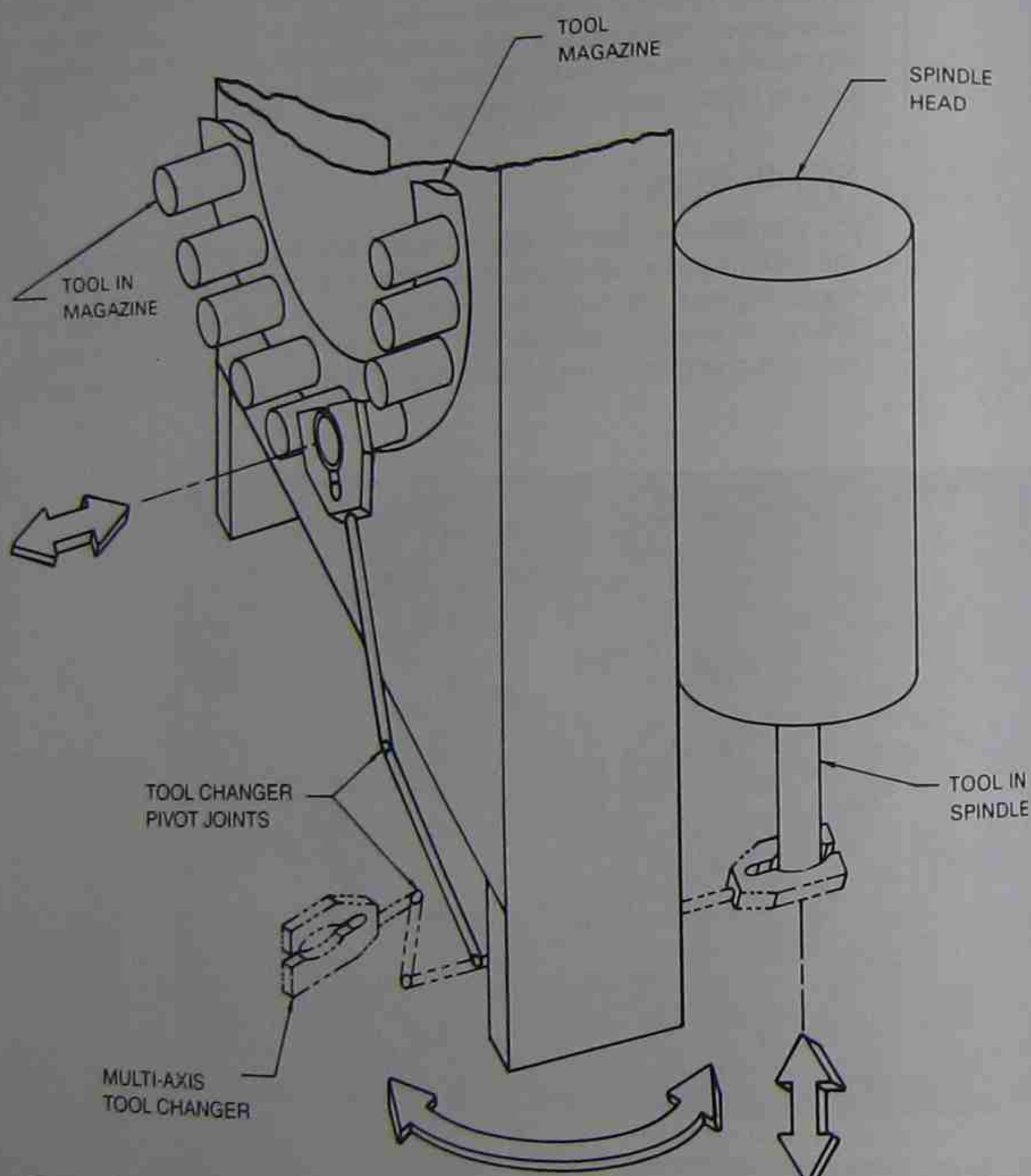


FIGURE 4-15
Multi-axis tool changer

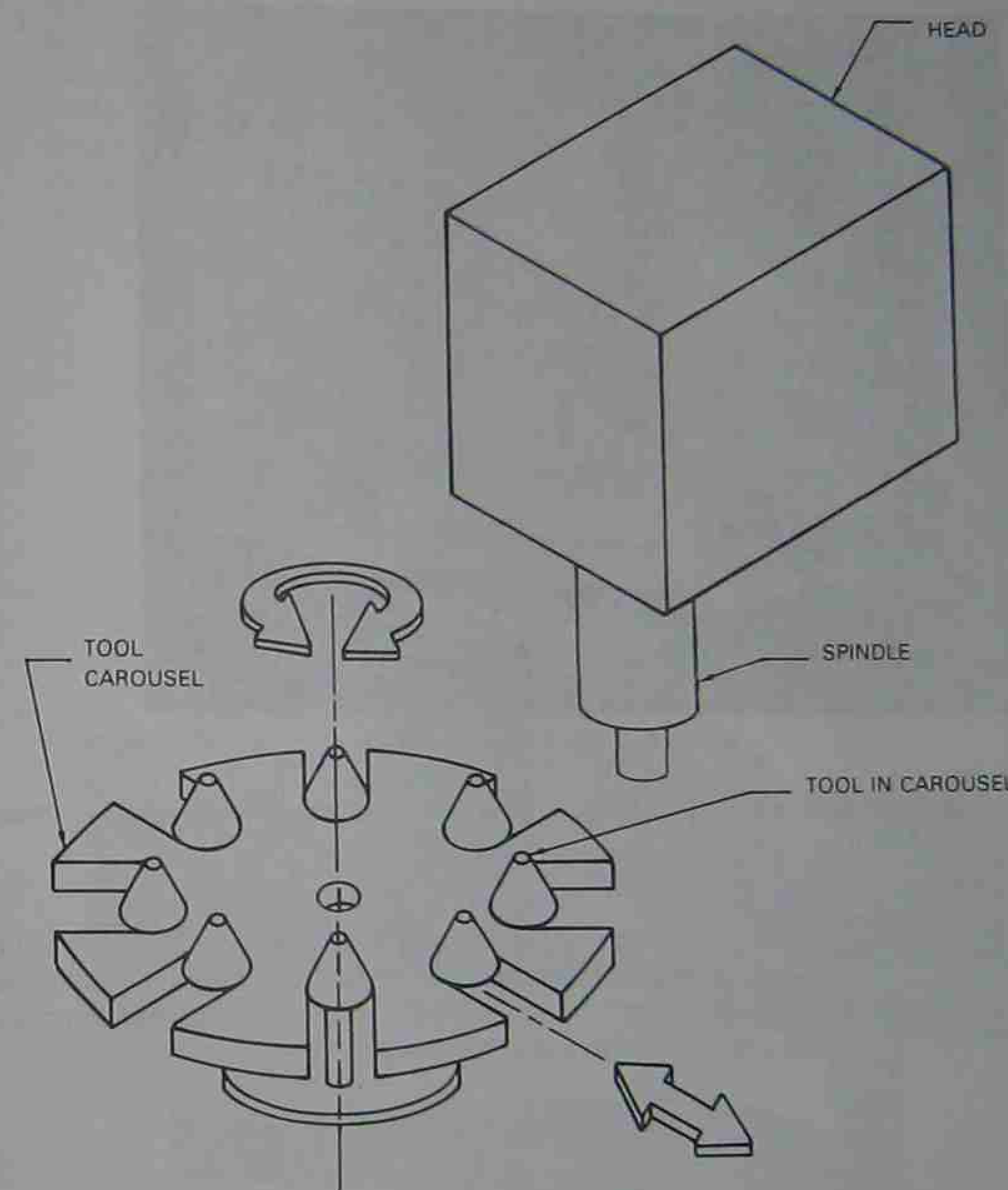


FIGURE 4-16
Spindle direct tool change

and the tool carousel moves downward, removing the tool. The carousel then indexes to align the required tool with the spindle, and moves upward, inserting the tool into the spindle where the tool is secured. Finally, the carousel moves sideways away from the spindle, thus disengaging itself from the tool holder, and returns to its "parked" position. The tool change is now complete.

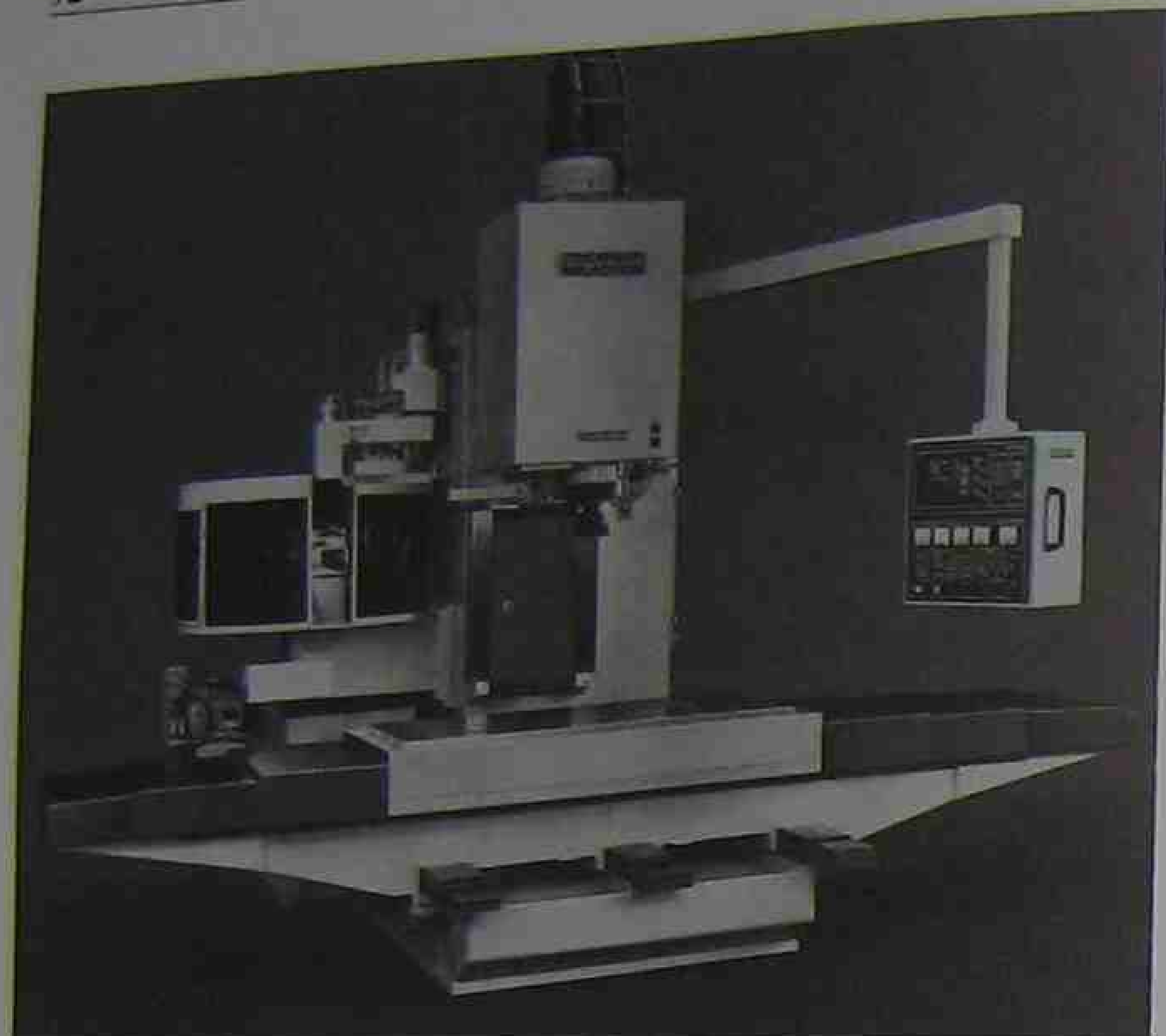


FIGURE 4–17
A vertical spindle machining center. Note the tool changer and carousel tool storage magazine. *(Photo courtesy of Cincinnati Milacron)*

On some large vertical spindle machinery the procedure varies from the one just described. Tool carousels on very large machinery are too large to manipulate easily. Rather than move the carousel, the spindle is moved to the carousel and lowered over it to remove and insert the tools.

TOOL STORAGE

As with tool changers, there are as many tool storage systems as there are manufacturers. However, tool storage systems may be loosely grouped into two types: *carousel* and *matrix*.

Carousel Magazine

A carousel magazine stores the tools in a circular fashion. The machining centers pictured in Figures 4–3, 4–17, and 4–18 employ a tool carousel for

tool storage. When a particular tool is called up, the carousel indexes to position the correct tool in the proper location for the tool changer to grab it. In addition to their use for spindle direct tool change on vertical spindle machining centers, carousels may be mounted on carts and moved to the proper spot as needed, such as when spindle direct tool change is employed on large equipment. They may also be mounted on the sides or backs of machines, depending on the type of tool changer used.

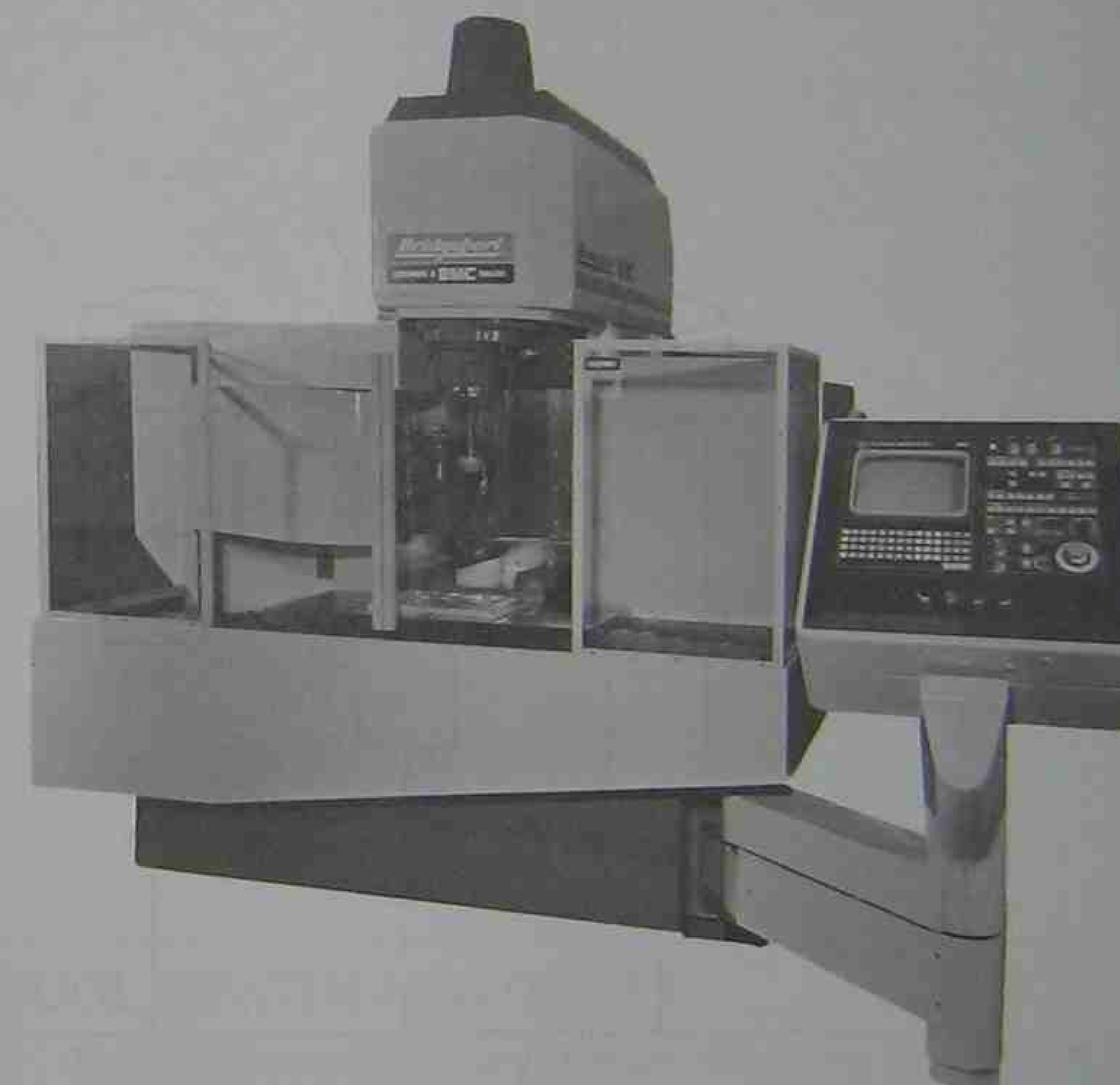


FIGURE 4–18
A vertical spindle machining center using carousel tool storage *(Photo courtesy of Bridgeport Machines Division of Textron Inc.)*

Matrix Magazine

Figures 4–2 and 4–14 picture machining centers employing matrix tool magazines. Figure 4–2 shows a single matrix magazine; Figure 4–14 shows a double. In either case, tool holder sockets are incorporated into long chains. When a tool is needed, the chain of sockets moves to position the correct tool socket in line with the tool changer. The advantage of the matrix magazine is that it is not limited to a circular configuration. In an oval configuration, for example, a matrix magazine can store a large number of tools in a limited amount of space.

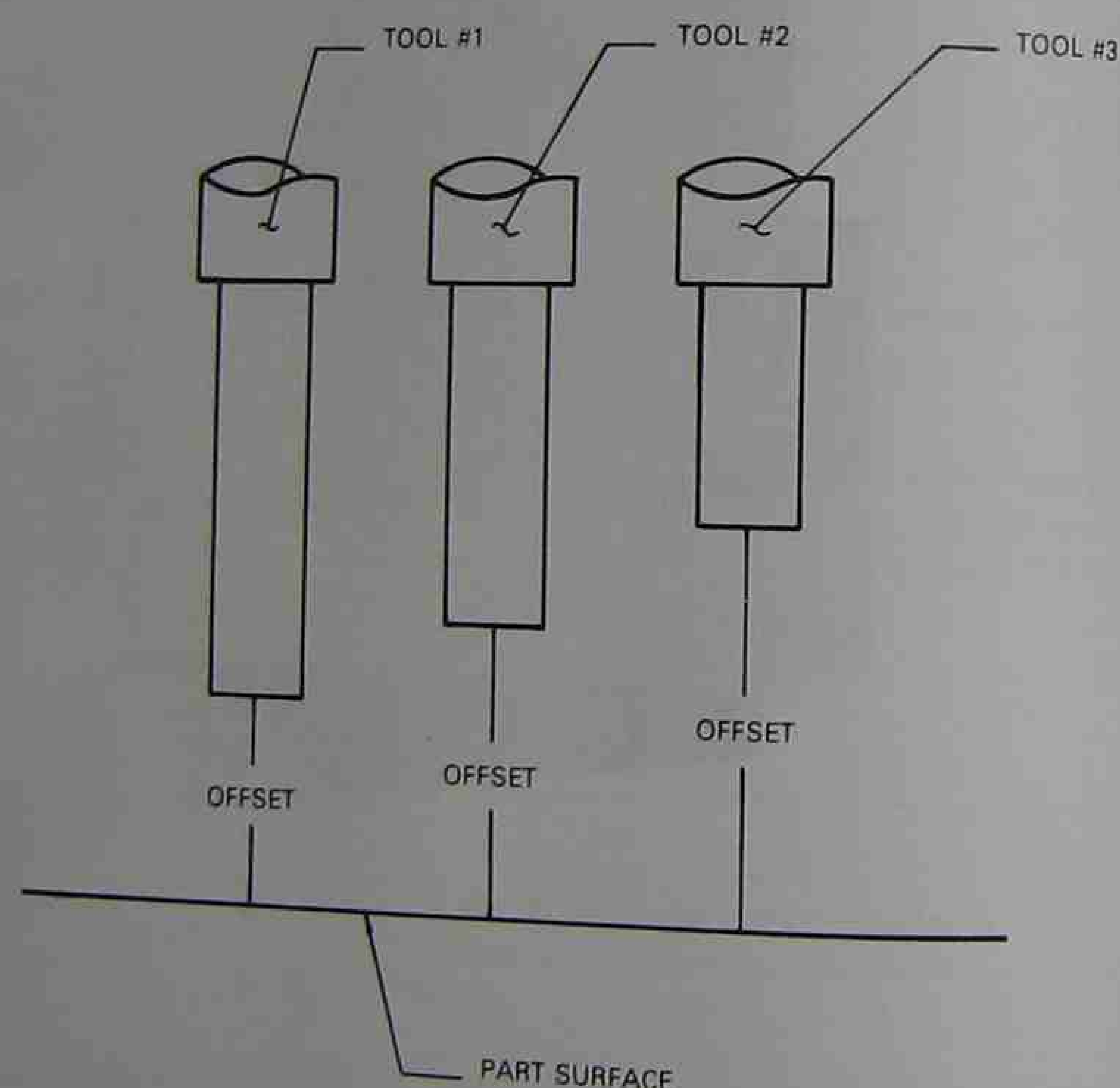


FIGURE 4–19
Tool length offset

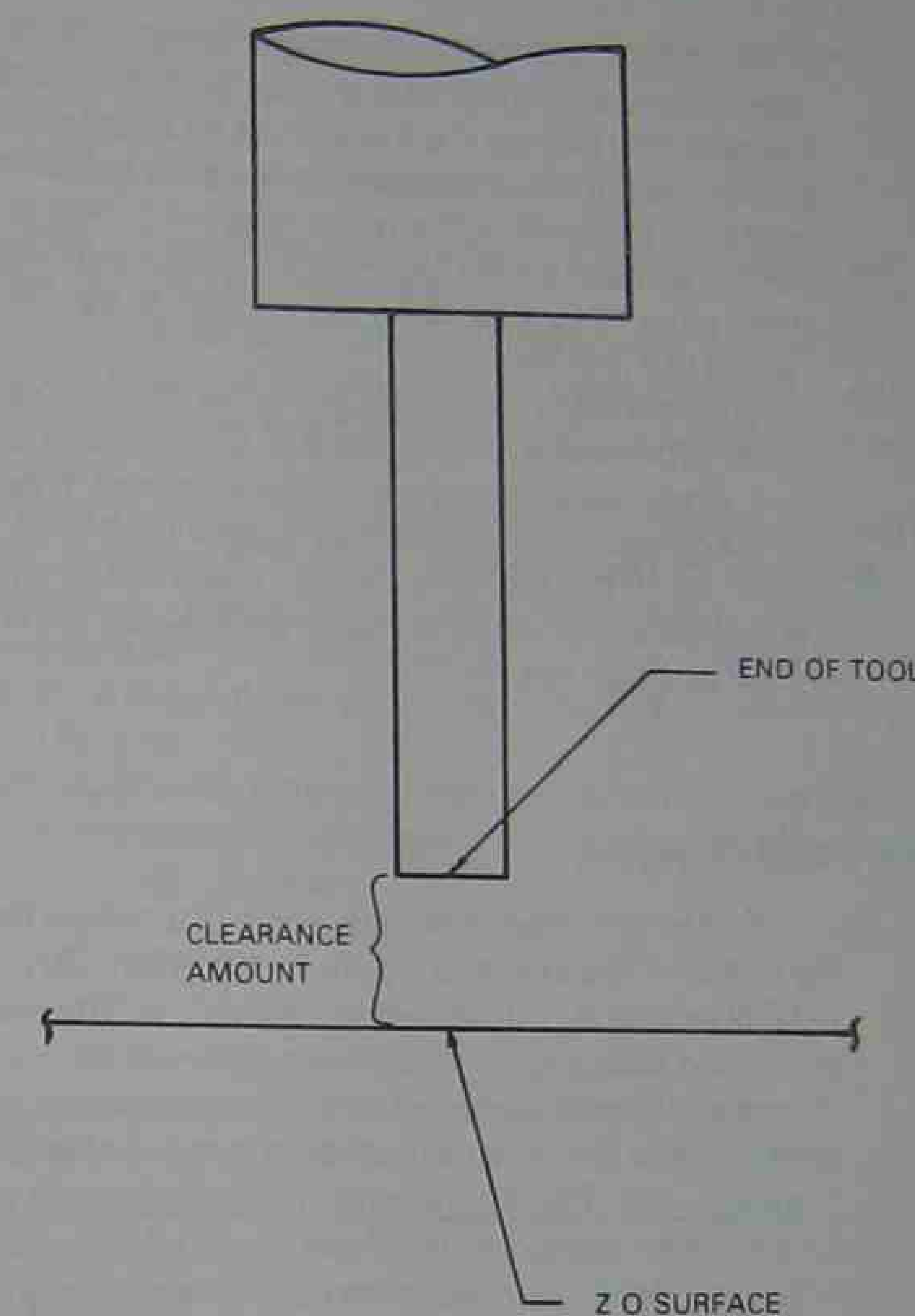


FIGURE 4–20
Tool clearance

TOOL LENGTH AND OFFSET

Tools used for machining vary in length. When using three-axis NC machinery, some means to compensate for the differing tool lengths must be employed. One method of dealing with this problem is to measure the tools prior to writing the program, so the programmed coordinates on any given tool movement will not interfere with the part, a clamp, or the machine table. Typically, the tool lengths are specified in an instruction sheet developed by the programmer that is sent to the shop floor for use in setting up the machine for a particular run.

Sometimes a tool setup drawing is used. Special tool setting equipment is needed to measure the tools accurately. The cost of this equipment, and the labor necessary to set the tools, must be included in the cost of any numerical control system utilizing premeasured tooling. This method of tool length compensation also makes the replacement of broken or dull tools complicated, as such tools must be set to a specific length to function properly. With tape machinery, however, measuring tools is usually the only way to accommodate the various tool lengths.

The advent of CNC machinery has revolutionized tool setting by introducing the *programmable tool register.* A tool register is a memory spot in the computer where the length of a tool may be stored. When a particular tool is called up, the computer checks the tool register to see how much *offset* has been programmed for that tool (see the discussion on *tool offset* that follows). These offset figures are usually entered by the operator at the time the machine is set up for the program run.

Tool Offset

Tool length offset is not the length of a tool but the distance from the part to the bottom of the tool (see Figure 4–19). After a Z0 point has been set, the longest tool to be used is installed in the machine. The table or machine head is then positioned with a specific distance between the tool and the workpiece. This distance is determined by either the programmer or setup man and must be sufficient to clear any clamps or other projections when the spindle is retracted (see Figure 4–20). The programmer may have to leave empty lines in the program for the setup operator to enter tool length offsets. To determine a particular tool offset, the tool is installed in the spindle and the spindle lowered until the tool is at the desired Z0 point on the part. The amount of offset for that tool will be displayed in the axis readout on the MCU. This offset amount, the distance from the tool to the part, is then entered in the MCU. The spindle can then be raised back to Z0, the tool removed, and the procedure repeated for the next tool.

Each time a tool is called up by the program, the offset value for that tool is used to shift the original Z0 point to the position on the part that the programmer desires as the Z0 point for that tool. To fully retract the spindle, the tool offset is cancelled, shifting the Z0 point back to its original position.

There are two basic types of tool offset methods being used on CNC machinery. Some controllers separate the offset from the tool; that is, when a particular tool is called up, the offset to be used with that tool must be called up separately within the CNC program. On other controllers, the offset is associated with a particular tool when it is entered in the machine control unit (MCU). When that particular tool is called up, the offset is automatically included.

SUMMARY

The important concepts presented in this chapter are:

- The speed, repeatability, and accuracy of a tool change greatly influence the efficiency of numerical control.
- There are two types of tool changes: manual and automatic.
- Machinery utilizing manual tool change generally incorporates some type of quick-change tooling system to facilitate the speed and accuracy of tool changes.
- Automatic tool changers are grouped into five categories: turret head, 180-degree rotation, pivot insertion, multi-axis, and spindle direct.
- Tool storage magazines are grouped into two types: carousel or matrix.
- Tool registers are places in the computer's memory to program tool offsets.
- A tool offset is the distance from the bottom of the tool to the desired Z0 point on the part.
- Tool offsets may be entered during setup. In this case the programmer leaves empty blocks in the program in which the tool offsets are placed by the setup man (or operator).

REVIEW QUESTIONS

1. Why is tool changing so important in numerical control?
2. What are the two types of tool changes?
3. On what type of machinery are R-8 spindle tapers found?
4. On what type of machinery is an American Standard Machine Taper used?
5. What type of device is used on manual tool change machines to increase the speed of the tool change?
6. What are the five basic types of tool changers?
7. How does a 180-degree rotation tool changer work? How does a pivot insertion tool changer work?
8. What type of machinery is a spindle direct tool-changing system best suited for?
9. What are the two types of tool storage magazines?
10. What is a tool register?

11. What is a tool length offset?
12. Why are tool registers an improvement over other types of tool length solutions?
13. How does a programmer allow for tool length offsets in a part program?
14. What procedure is used by the operator to determine the tool length offsets?

CHAPTER 5 Programming Coordinates

OBJECTIVES Upon completion of this chapter, you will be able to:

- Explain what a hole operation is.
- Program hole operation coordinates using absolute and incremental positioning.
- Program milling coordinates using absolute and incremental positioning.

HOLE OPERATIONS

To understand how to program coordinates for hole operations, such as drilling, reaming, boring, and tapping, assume that the holes shown on the part drawing in Figure 5–1 are to be drilled using an absolute positioning machine. For hole #1, the coordinates are X0.7500, Y1.7500; for hole #2, the coordinates are X2.0000, Y0.2500; for hole #3, the coordinates are X3.0000, Y1.0000. Note that no plus or minus signs are given with any of these coordinates. If a coordinate is positive, no sign need be given; the machine will assume a positive coordinate unless otherwise indicated. Looking at Figure

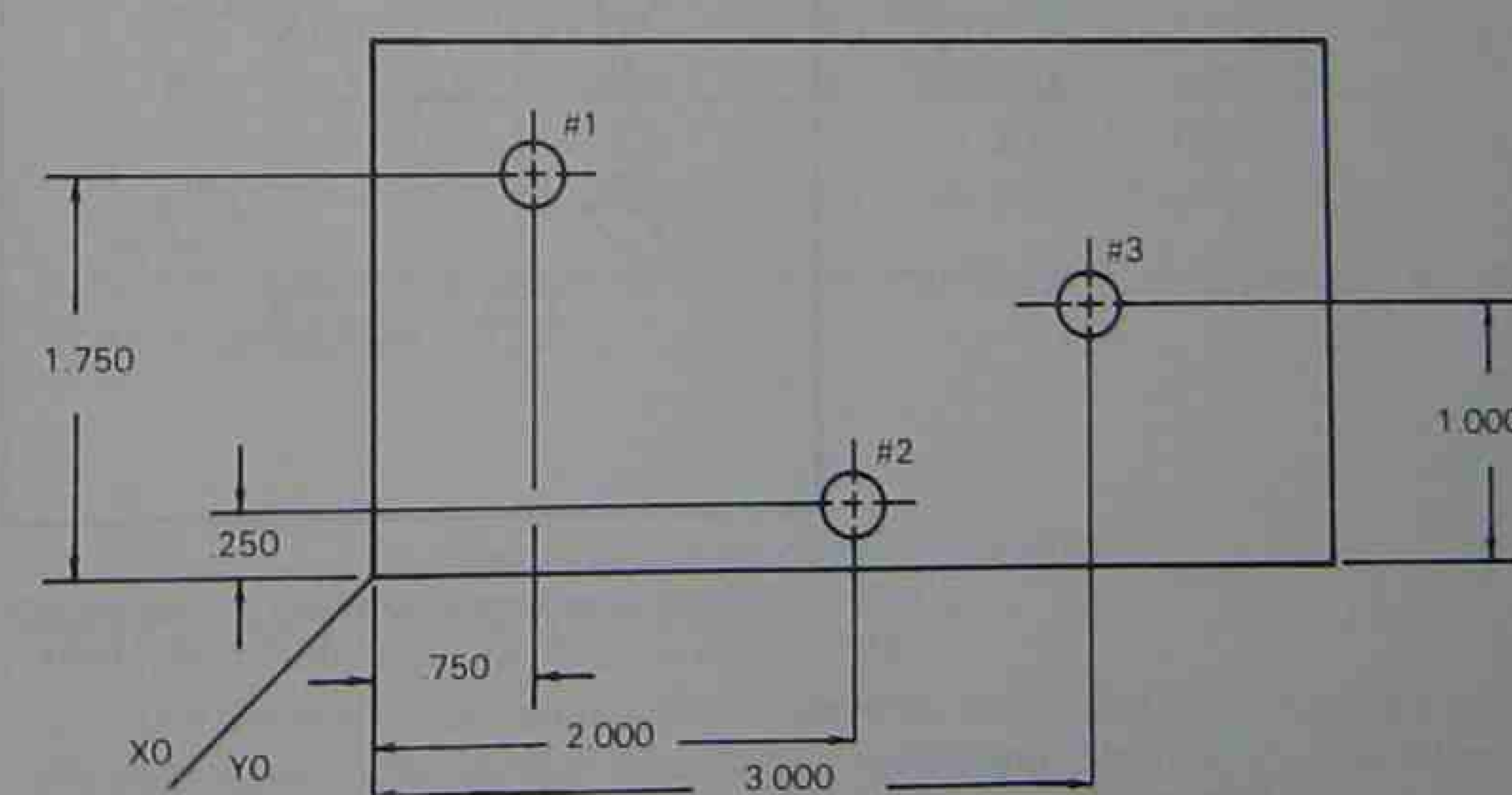


FIGURE 5–1

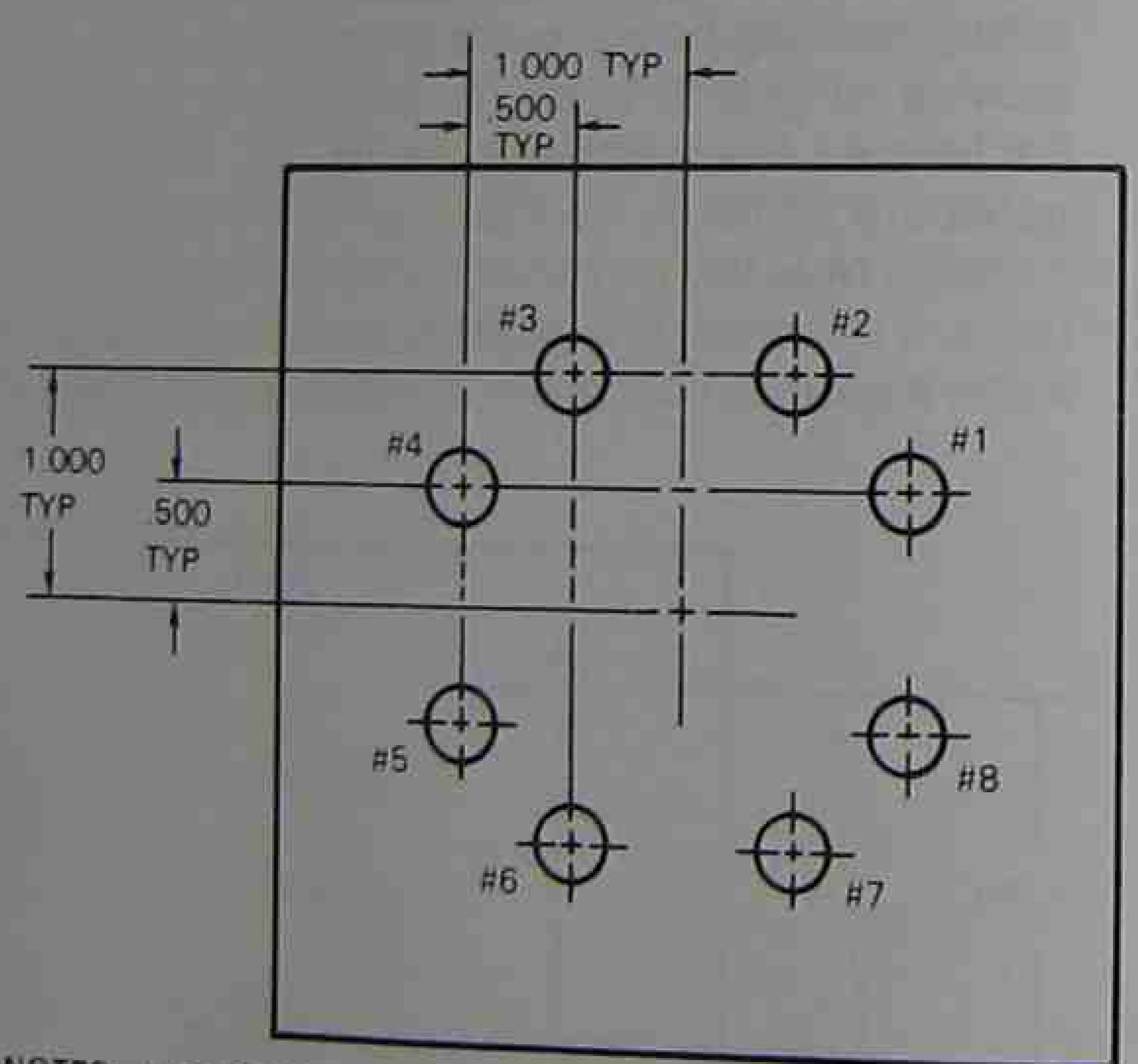
5–2, try to arrive at the coordinates to drill this part on an absolute positioning machine. The proper coordinates are as follows:

#1	X1.0000, Y0.5000	#5	X – 1.0000, Y – 0.5000
#2	X0.5000, Y1.0000	#6	X – 0.5000, Y – 1.0000
#3	X – 0.5000, Y1.0000	#7	X0.5000, Y – 1.0000
#4	X – 1.0000, Y0.5000	#8	X1.0000, Y – 0.5000

The same principles apply to the parts in Figures 5–1 and 5–2. The difference is that X0/Y0 is located at the center of the part in Figure 5–2. Notice that the signs of X and Y change as the coordinate locations move from quadrant to quadrant.

Figure 5–3 shows the same part as that in Figure 5–1 but delta dimensioned rather than datum dimensioned. Try to derive the proper coordinates to drill the holes in Figure 5–3, using an incremental positioning machine. The coordinates for the holes are as follows:

#1 X0.7500, Y1.7500
#2 X1.2500, Y – 1.5000
#3 X1.0000, Y0.7500



NOTES: 1) PART X0/Y0 IS CENTER OF PART
2) FOR INCREMENTAL MOVES, THE SPINDLE IS ASSUMED CENTERED OVER X0/Y0 AT THE START OF PROGRAM SEQUENCE.

FIGURE 5–2

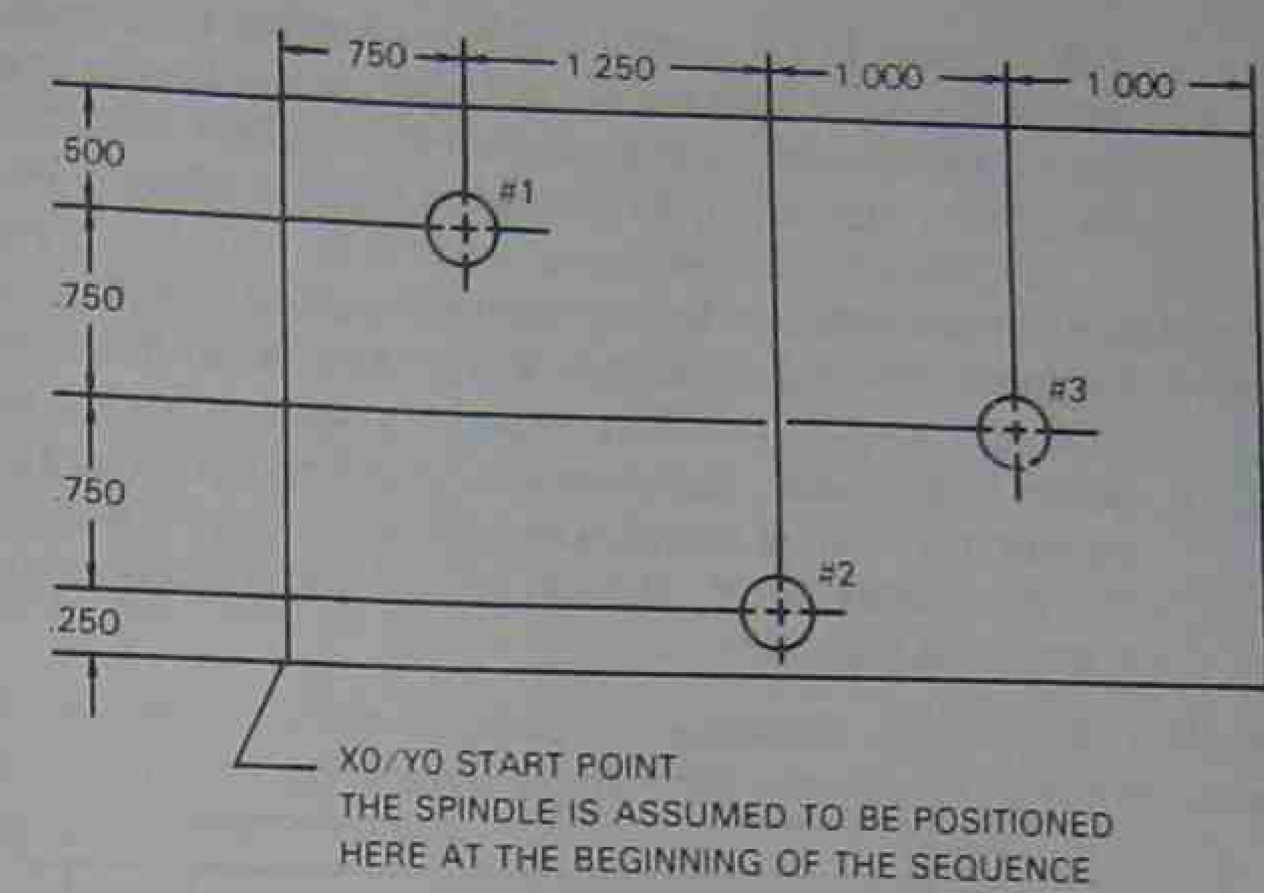


FIGURE 5–3

Notice that the sign of Y was negative when moving to hole #2. Since incremental positioning was being used, hole #1 became the X0/Y0 point for the movement to hole #2. With incremental drawings, it is necessary to add and subtract dimensions in order to correctly program the part, even when using delta dimensioned drawings.

Referring again to Figure 5–2, assume that an incremental positioning machine is to be used. Determine the coordinates necessary to drill the part. The correct coordinates are:

#1	X1.0000, Y0.5000	#5	X0.0000, Y – 1.0000
#2	X – 0.5000, Y0.5000	#6	X0.5000, Y – 0.5000
#3	X – 1.0000, Y0.0000	#7	X1.0000, Y0.0000
#4	X – 0.5000, Y – 0.5000	#8	X0.5000, Y0.5000

Even though this is a datum dimensioned drawing, it is often possible to program incrementally from it.

MILLING OPERATIONS

The system of coordinates presented thus far is used for centering a spindle over a particular location specified on a drawing. This means that when a coordinate location is given to the machine, the center of the spindle is sent to that location. In the case of milling cutters, this technique would cause a problem in that more than the correct amount of stock would be removed from the part (an amount equal to the radius of the cutter). When positioning the spindle for a milling operation, an allowance must be made for the radius of the cutter.

A .500-inch-diameter end mill is to be used to mill the part in Figure 5–4, and an absolute positioning mill will be used. Sending the cutter to X0/Y0 to begin a milling pass from location #1 to location #2 will remove an additional .250 inch of metal from the part that is called out in the drawing. To allow for the radius of the cutter, calculate the cutter coordinate by subtracting half the diameter of the cutter from the coordinate location in each axis. For location #1 the coordinates are X−0.2500, Y−0.2500. The coordinates for all four locations are as follows:

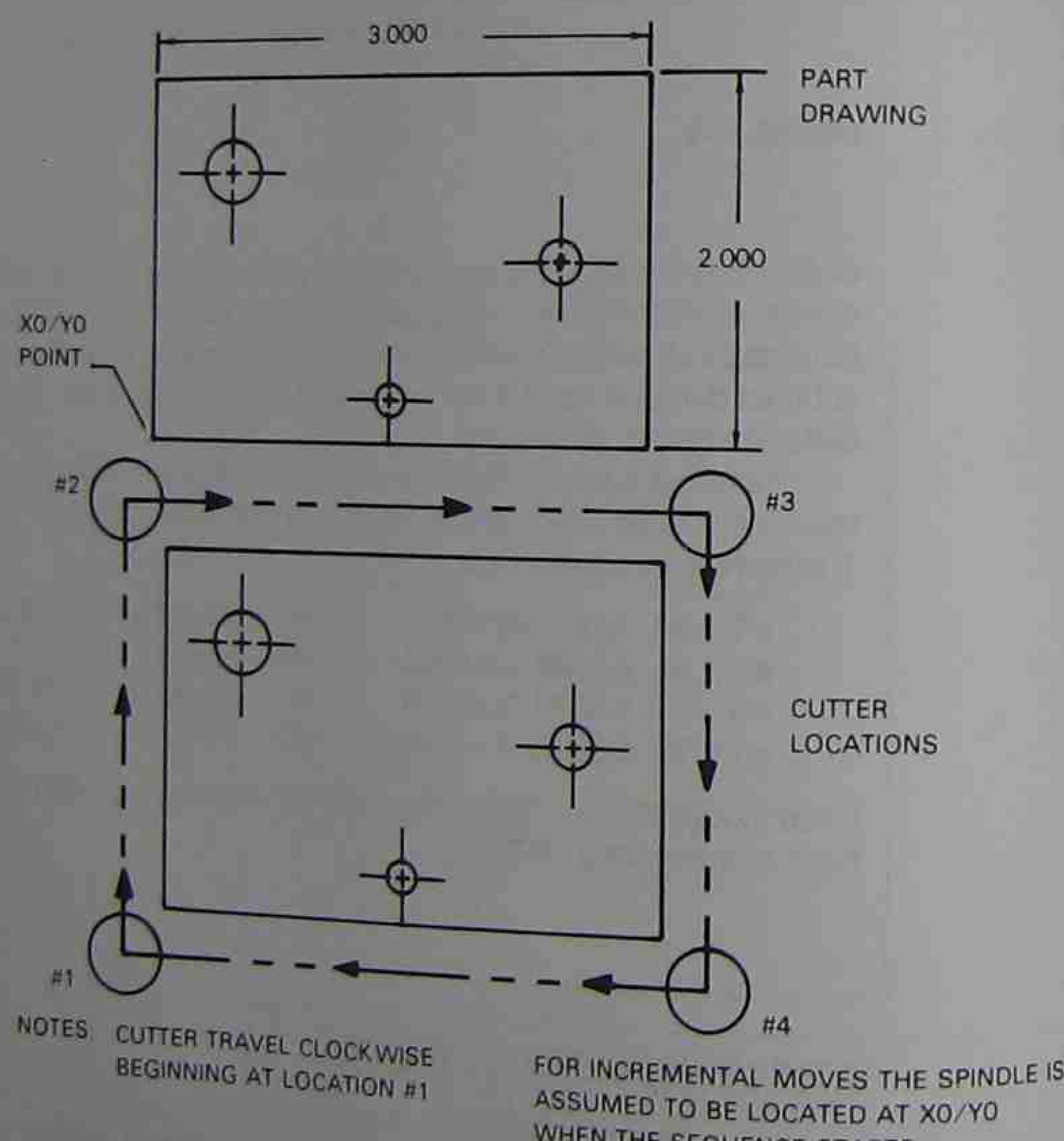


FIGURE 5–4

#1 X−0.2500, Y−0.2500
#2 X−0.2500, Y2.2500
#3 X3.2500, Y2.2500
#4 X3.2500, Y−0.2500

Assume that an absolute positioning machine is to be used to mill points indicated on the part drawing in Figure 5–5. The coordinates for this part are:

#1 X0.7500, Y0.7500
#2 X0.7500, Y1.2500
#3 X2.2500, Y1.2500
#4 X2.2500, Y0.7500

Try to determine the coordinates required to mill the parts in Figures 5–4 and 5–5, using an incremental positioning machine. The correct coordinates for Figure 5–4 are as follows:

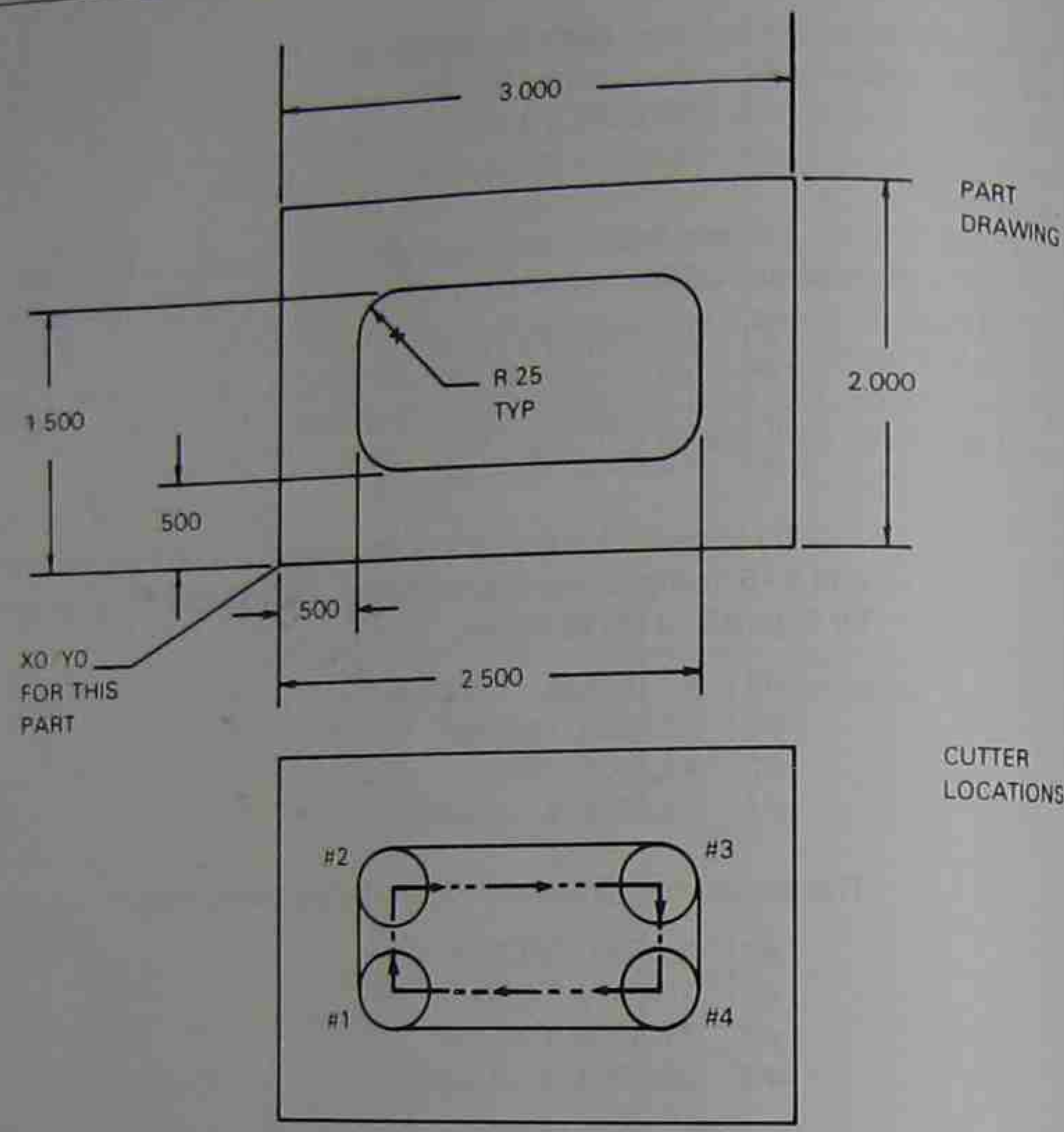
#1 X−0.2500, Y−0.2500
#2 X0.0000, Y2.5000
#3 X3.5000, Y0.0000
#4 X0.0000, Y−2.5000

The correct coordinates for Figure 5–5 are as follows:

#1 X.07500, Y0.7500
#2 X0.0000, Y0.5000
#3 X1.5000, Y0.0000
#4 X0.0000, Y−0.5000

Movement of the Z axis is easier than that of the X or Y axis. To drill any of the parts examined in this chapter, all that is required is to give the Z axis a coordinate that would place the end of the tool thru the part.

Assume that the zero point for the Z axis is the top of a .250-inch-thick part. A ¼-inch-diameter hole is to be used to drill a hole in the part. The coordinate for the Z axis would be the thickness of the part plus the length of the drill point. For a ¼-inch drill the coordinate would be Z−0.3250. The length of a drill point is calculated by multiplying the diameter of the drill by .3. In this case, the drill point is .075 inch long. This length added to the part depth (.250 inch) results in the .3250 length. In practice, it is wise to allow a small amount of additional movement to compensate for differences in drill point and part thickness tolerances. A movement toward the machine table would be a −Z movement. Movement toward the head of the machine would be a +Z movement. Chapter 7 covers three-axis milling and use of the Z axis in more detail. At this point, an understanding of the X and Y movements necessary to program coordinates will suffice.



NOTES: CUTTER TRAVEL - CLOCKWISE BEGINNING AT LOCATION #1

FOR INCREMENTAL MOVES, THE SPINDLE IS ASSUMED TO BE LOCATED AT X0/Y0 WHEN THE SEQUENCE STARTS

FIGURE 5–5

MIXING ABSOLUTE AND INCREMENTAL POSITIONING

CNC machines are capable of both incremental and absolute positioning. This gives the programmer a great deal of flexibility in programming parts. Assume that the part in Figure 5–6 is to be drilled using both absolute and incremental positioning. Hole #1 is to be drilled first, using absolute positioning; holes #2, #3, and #4 are to be drilled next, using incremental positioning; hole #5 is to be programmed next, using absolute positioning; and holes #6, #7, and #8 will be drilled using incremental positioning. Notice that the method of programming these coordinates is similar to the dimensioning used on the part print. Determine the coordinates to program the hole locations before looking at the following correct coordinates.

#1	X0.5000, Y−0.5000	#5	X2.7500, Y−2.0000
#2	X0.0000, Y−0.7500	#6	X0.0000, Y−0.7500
#3	X1.0000, Y0.0000	#7	X0.7500, Y0.0000
#4	X0.0000, Y0.7500	#8	X0.0000, Y0.7500

METRIC COORDINATES

Some industries have converted all or part of their operations to metric units of measure. Most countries outside of the United States use metric measurement. It is advantageous, therefore, for companies with worldwide markets to use this system in manufacturing their products. Automobile manufacturers are but one example of a number of industries now converting to the metric system. It appears that both the inch and metric systems will be used for quite some time in the United States. Many experts agree that the United States will never fully convert to the metric system, but the numerical control programmer will have to deal with metric measures and should become familiar with their use in the shop.

The metric system in use today is called the *Système International d'Unites*, or the *SI* metric system. There are seven base units used in the metric system. Length is based on the *meter* (m), mass on the *kilogram* (kg), time on the *second* (s), electric current on the *ampere* (A), temperature on the *kelvin* (K), amount of substance on the *mole* (mol), and luminous intensity on the *candela* (cd). All metric units are built on a base-ten system. In the machine shop, measurement is based on the meter, which can be broken down into smaller units. A decimeter is 0.1 meter; a centimeter is 0.01 meter (0.1 decimeter); a millimeter is 0.001 meter (0.1 centimeter).

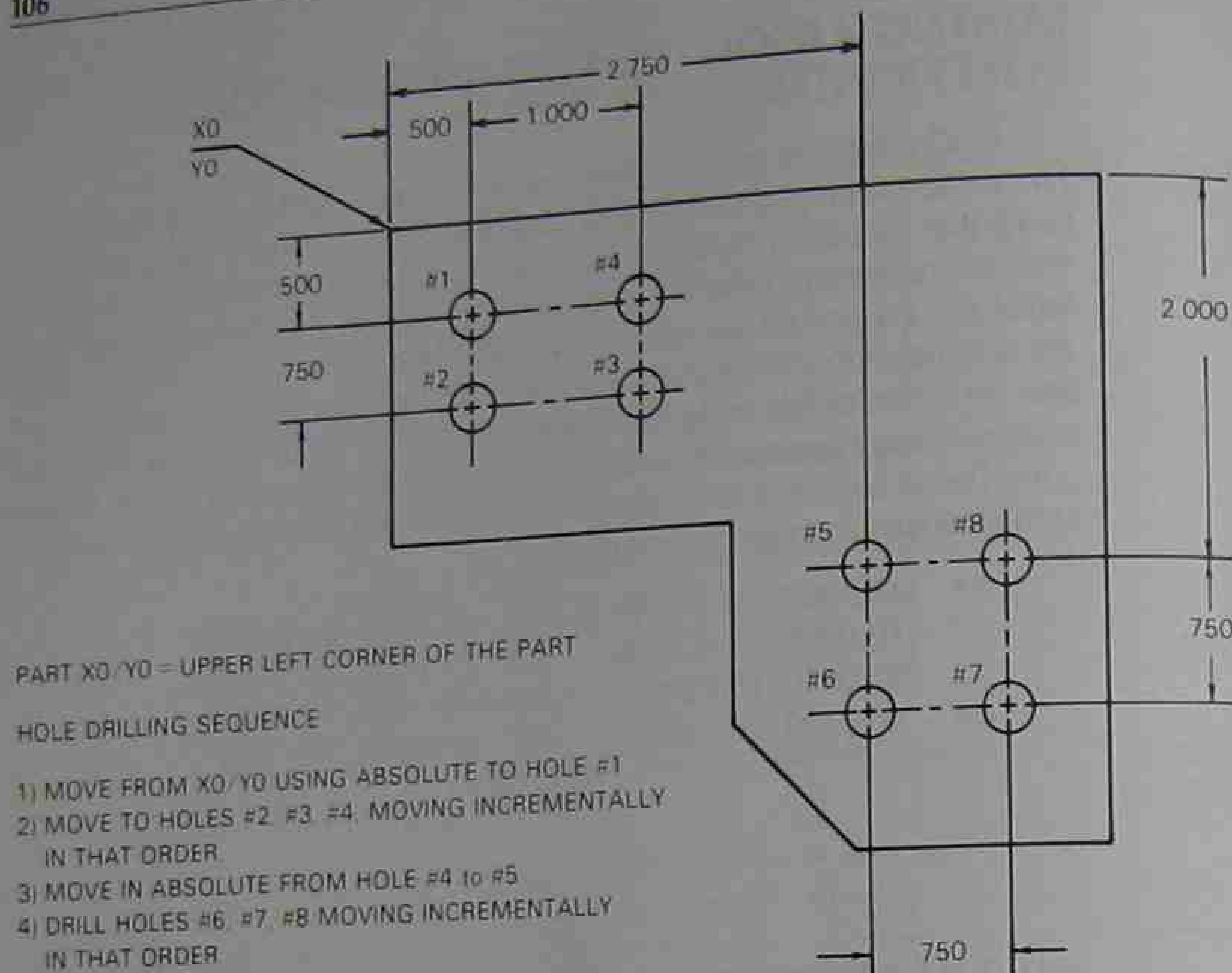


FIGURE 5–6

In the inch system, length measurement is based on the yard. Units smaller than a yard are built on fractions of a yard, foot, or inch, whichever is most compatible (one-half of a yard = 1½ feet or 18 inches). In the machine shop, however, measurement is referenced to thousandths of inches. In the shop, 1" would be one inch; .500" however, is not thought of as five-tenths of an inch but, rather, as five-hundred thousandths of an inch. Therefore, .0005" is not usually called five ten-thousands of an inch, but five tenths, meaning five tenths of one-thousandth of an inch. When dealing with metric measurement in the shop, measurement is referenced to millimeters. One centimeter (1 cm) is not spoken of as one centimeter but is called ten millimeters (10 mm); one millimeter is approximately .0394 inch. Units smaller than one millimeter are also referenced in terms of millimeters; 0.01 is one-hundredth of a millimeter; 0.001 is one-thousandth of a millimeter; 0.001 inch is approximately .0254 millimeter; and .0001 inch is approximately 0.00254 millimeter. Many times a

metric print tolerance will call for a two-place decimal to be held to + or −0.02 mm. This roughly corresponds to holding an inch dimension to ± .001 inch.

Metric units are easy to work with as long as a company's commitment to metric conversion is carried all the way through from drafting room to tool crib. If metric cutters are available, working with metric dimensions is no problem. Modern CNC machinery has the capability to accept either metric or inch dimensions. The only difference in writing a program in metric versus inch measurements is that the coordinates are expressed differently. If inch tooling is used, it is necessary to convert the cutter sizes to metric units, so that proper milling coordinates can be programmed. To convert an inch dimension to a metric one, multiply the inch dimension by 25.4. To convert a metric dimension to one of inches, multiply the metric dimension by .03937 (or divide the metric dimension by 25.4).

Having learned the use of absolute and incremental positioning, and understanding how the Cartesian coordinate system works, a numerical control program may now be written.

SUMMARY

The important concepts presented in this chapter are:

- To program a hole location coordinate, the center line for the hole is used.
- To program a coordinate for milling operations, the coordinate for the location must include an appropriate allowance for the radius of the cutter.
- For absolute positioning, the datum reference plane remains the X0, Y0 point for all programmed moves.
- For incremental positioning, the current coordinate location is the X0, Y0 point for the next move.
- CNC machines are capable of mixing absolute and incremental positioning. This allows for flexibility in programming.
- Metric measurement in the machine shop is based on the millimeter, where .02 mm is roughly equivalent to .001 inch.
- To convert an inch dimension to millimeters, multiply the inch dimension by 25.4. To convert a metric dimension to inches, multiply the metric dimension by .03937, or divide the metric dimension by 25.4.

REVIEW QUESTIONS

1. What is a hole operation?
2. Where does the spindle centerline have to be programmed for a hole operation? For a milling operation?
   (Questions #3, #4 and #5 refer to Figure 5–7.)
3. What would the absolute coordinates for holes #2, #3, and #4 be?
4. What would the incremental coordinates be for holes #1, #3, and #4, moving to the holes in that order and starting at the lower left corner of the part?
5. Assume the spindle is positioned at hole #3. What would the incremental coordinates be to move from there to holes #2, #1, and #4, in that order? What would the absolute coordinates be?
6. Using a .625-inch-diameter end mill, what would the four absolute coordinates necessary to mill the part periphery be? What would the incremental coordinates be?
7. Assume that the hole patterns in Figure 5–8 are to be drilled using a CNC machine capable of both incremental and absolute positioning. Give the absolute coordinates to drill hole #1. Give the incremental coordinates to then drill holes a, b, c, and d, respectively. Give the absolute coordinate to drill hole #2, and the incremental coordinates to then drill holes e, f, g, and h, respectively.
8. Convert the following inch measurements to metric measurements.
   a. .500
   b. .4375
   c. .3125
   d. .125
9. Convert the following metric measurements to inch measurements.
   a. 0.02
   b. 0.005
   c. 2.5
   d. 8.0

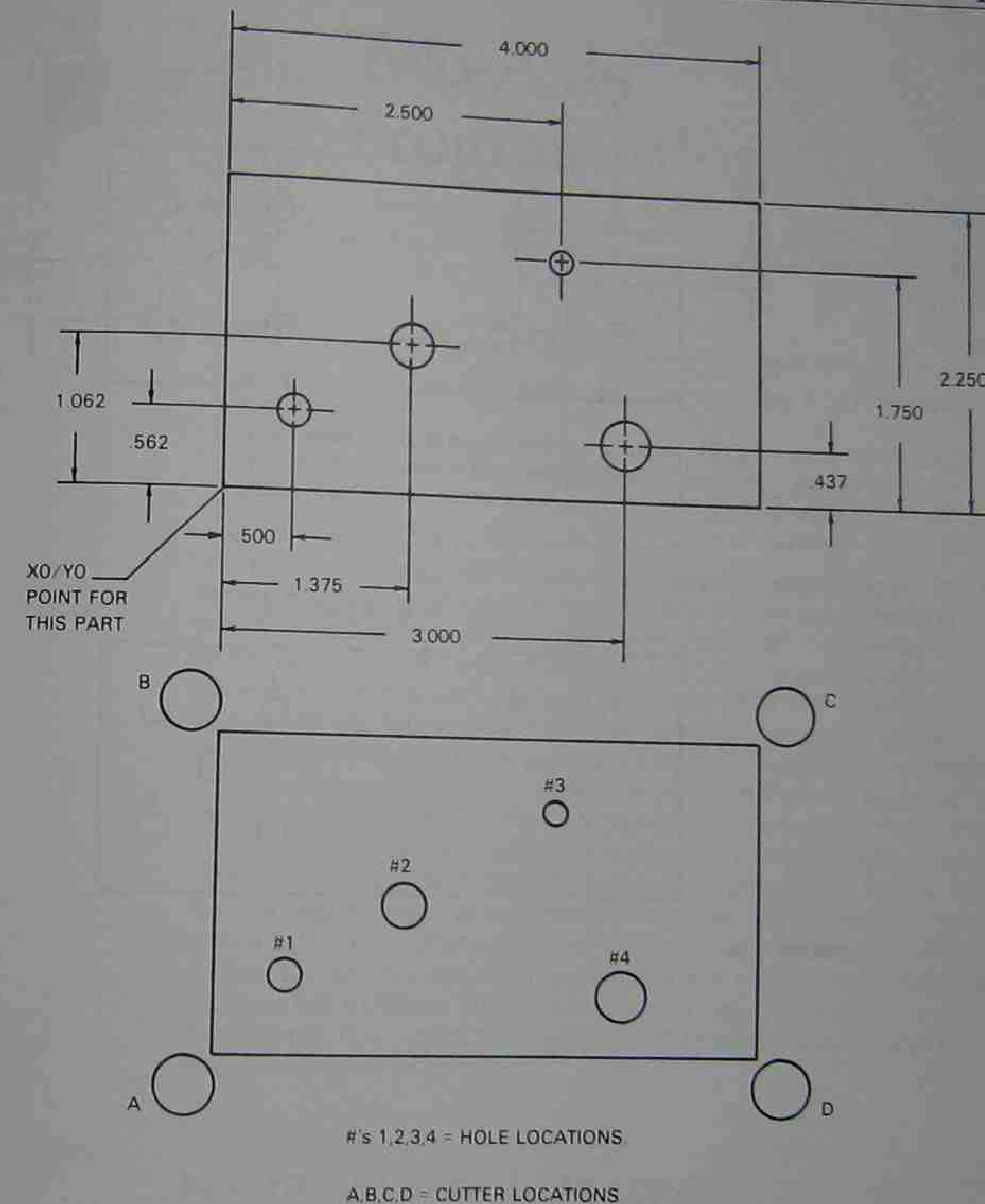


FIGURE 5–7

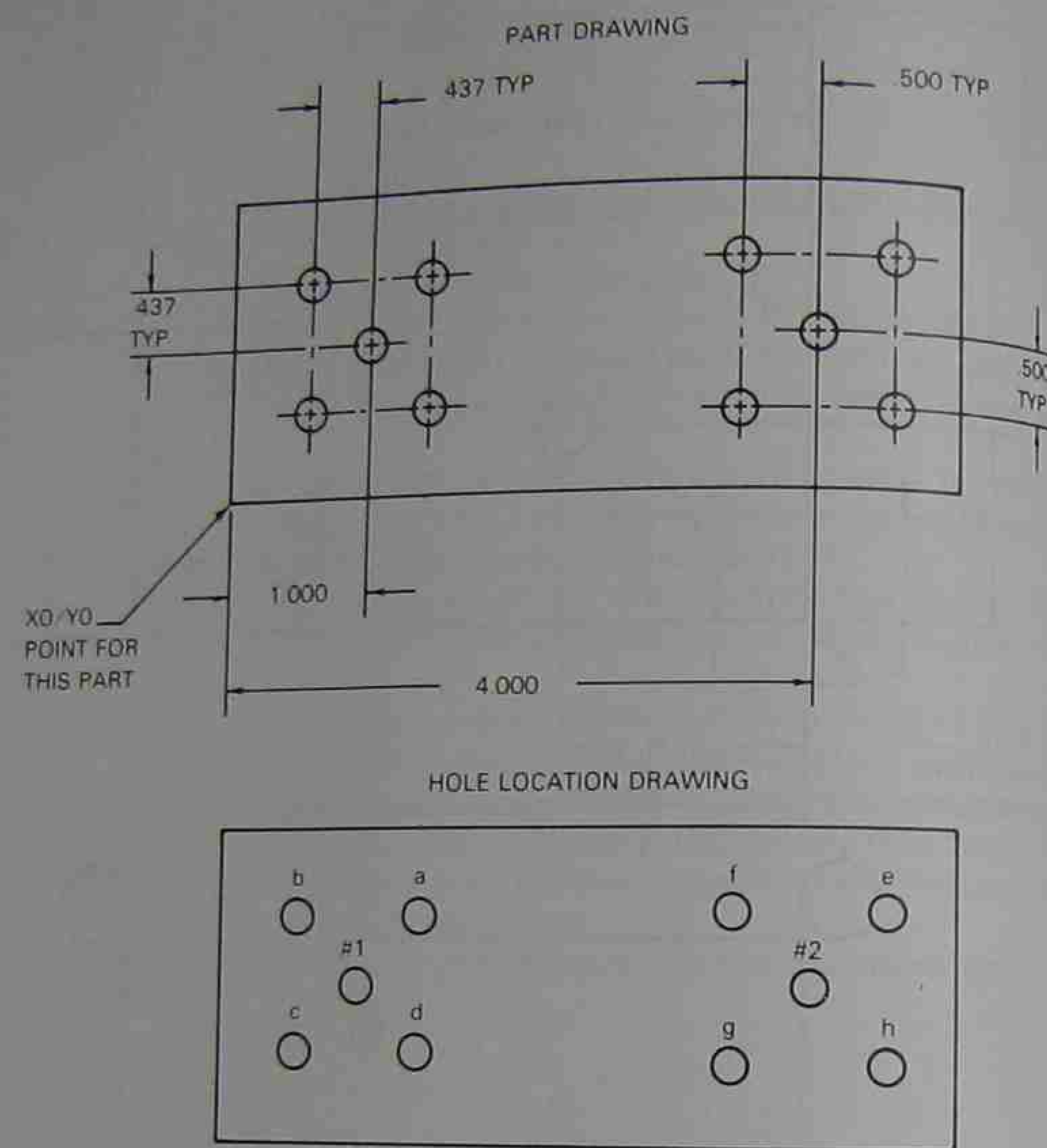


FIGURE 5–8

CHAPTER 6 Two-Axis Programming

OBJECTIVES Upon completion of this chapter, you will be able to:

- Write simple two-axis programs in Machinist Shop Language format to perform hole operations.
- Write simple two-axis programs in word address format to perform hole operations.
- Write simple two-axis milling programs using Machinist Shop Language.
- Write simple two-axis milling programs using word address programming format.

This text is concerned primarily with manual programming of CNC machinery. Each successive chapter will introduce a more advanced level of numerical control programming. For purposes of continuity, two basic machines will be used for the next several chapters. The following point cannot be overemphasised: *no two CNC machines program exactly alike*. There are, however, similarities between them. By learning to write programs for these machines, only minimal effort will be required to program other CNC machines.

Programming in this text is done primarily in two formats throughout: *Machinist Shop Language* and *word address format*. The instructional examples used in the next several chapters are milling and drilling examples. Chapters 13 and 14 deal with CNC lathes. The first machine programmed in this chapter is a CNC mill equipped with an Analam Crusader II controller. This machine uses a conversational programming language called *Machinist Shop Language*. The second machine programmed is a vertical machining center equipped with a General Numerics controller; it is programmed using word address format. This chapter deals with two-axis programming. Chapter 7 will introduce three-axis programming.

MACHINIST SHOP LANGUAGE

Machinist Shop Language, as mentioned above, is a conversational language. The commands used in programming with this language are common shop words rather than NC codes. Conversational languages are easy to learn since they use English commands. The MCU on a conversational language machine converts the English commands into the codes required for the program.

In Machinist Shop Language, the machine is given its instructions by means of commands placed in program lines called *events*. There are two types of events used in a Machinist Shop Language program: events *with motion* and events *without motion*. Events with motion position the machine to a desired coordinate location; events without motion perform such tasks as assigning feedrates or initiating planned program dwell. Following are some Machinist Shop Language commands.

Commands

X—When used before a number, X designates an X-axis coordinate. An X-axis coordinate is entered in the format XXX.XXXX for an inch coordinate. The format is XX.XXX for metric coordinates.

Y—When used before a number, Y designates a Y-axis coordinate. A Y-axis coordinate is entered in the format YYY.YYYY for an inch coordinate. The format is YY.YYY for metric coordinates.

Z—When used before a number, Z designates a Z-axis coordinate. A Z-axis coordinate is entered in the format ZZZ.ZZZZ for an inch coordinate. The format is ZZ.ZZZ for metric coordinates.

F—Initiates a move at the programmed feedrate.

A—Specifies that absolute positioning mode is to be used.

I—Specifies that incremental positioning mode is to be used.

FEED—Assigns a feedrate to be used as needed in the program. The format for the FEED command in inch mode is FEED ##.##. For metric mode the format is FEED ###.#.

DWELL—Causes the machine to halt execution of the program. A dwell may be entered with a time element specified, so that the program will recommence after the specified time interval. If no time element is specified, the machine will halt execution of the program until the start button is pushed. When entering a timed DWELL, the time interval is entered as an X-axis value.

TOOL—Acts like an untimed dwell, halting program execution until the start button is depressed. TOOL also assigns length and cutter diameter values to a particular tool. The format for a TOOL command is TOOL # where # is the tool desired (for example, TOOL 1, TOOL 2, etc.). The format for assigning tool lengths and diameters is TOOL 10## where ## is the number of the tool assigned (for example, TOOL 1001, TOOL 1025, etc.). The use of TOOL 10## will be further discussed in Chapter 7.

END—Signals the end of a program. It is also used to mark the end of a do loop or a subroutine. Do loops and subroutines will be covered in Chapter 11.

This is not a complete listing of all Machinist Shop Language commands (see Appendix 2). Other commands will be introduced as more operations are presented.

DRILLING IN MACHINIST SHOP LANGUAGE

The CNC mill to be programmed using Machinist Shop Language is a vertical turret milling machine with a computer numerical control unit. It is a continuous-path machine, with the ability to use both absolute and incremental positioning. The milling machine uses manual tool change. This means that the operator must change the tools. Since the machine has no *set* tool change location, the location must be specified in the program.

The part in Figure 6–1 is to be drilled on the CNC mill. Notice the X0/Y0 point for the part in the lower left corner. A tool change position has been selected at X – 2.0, Y – 1.5. A separate tool change location positions the spindle out of the way of the workpiece during tool changes. It also aids in removing the part from the vise or fixture. In Figure 6–2, a metric part is shown.

The first program for this part is shown in Figure 6–3. This program was written using absolute positioning. Figure 6–4 is the same program but written for the metric part pictured in Figure 6–2. The programming logic for the two parts is identical. The only difference between the two programs is the coordinates. When entering a program in the MCU of this machine, a button is first pushed to select inch or metric input.

Leading and Trailing Zeros

Before presenting program explanations, a discussion of leading and trailing zeros is in order. In the shop and on part drawings, dimensions often contain trailing zeros. For example, the dimension .500 contains two trailing zeros. On part dimensions, the trailing zeros are necessary to communicate the significance of a particular dimension (that is, three-place versus two-place decimals). Sometimes, for the sake of clarity, a leading zero is used, as in 0.500. Early NC equipment required the use of leading and/or trailing zeros in specifying any coordinate. Many CNC controllers do not require the use of either leading or trailing zeros; thus, .500 may be entered as .5 on these machines. The CNC machine will locate all programmed coordinates within the resolution of the machine. The programs in Figures 6–3 and 6–4 reflect this practice of omitting leading and trailing zeros. Not all controllers allow this practice, but for purposes of standardization in teaching, it is assumed that all controllers in this text do.

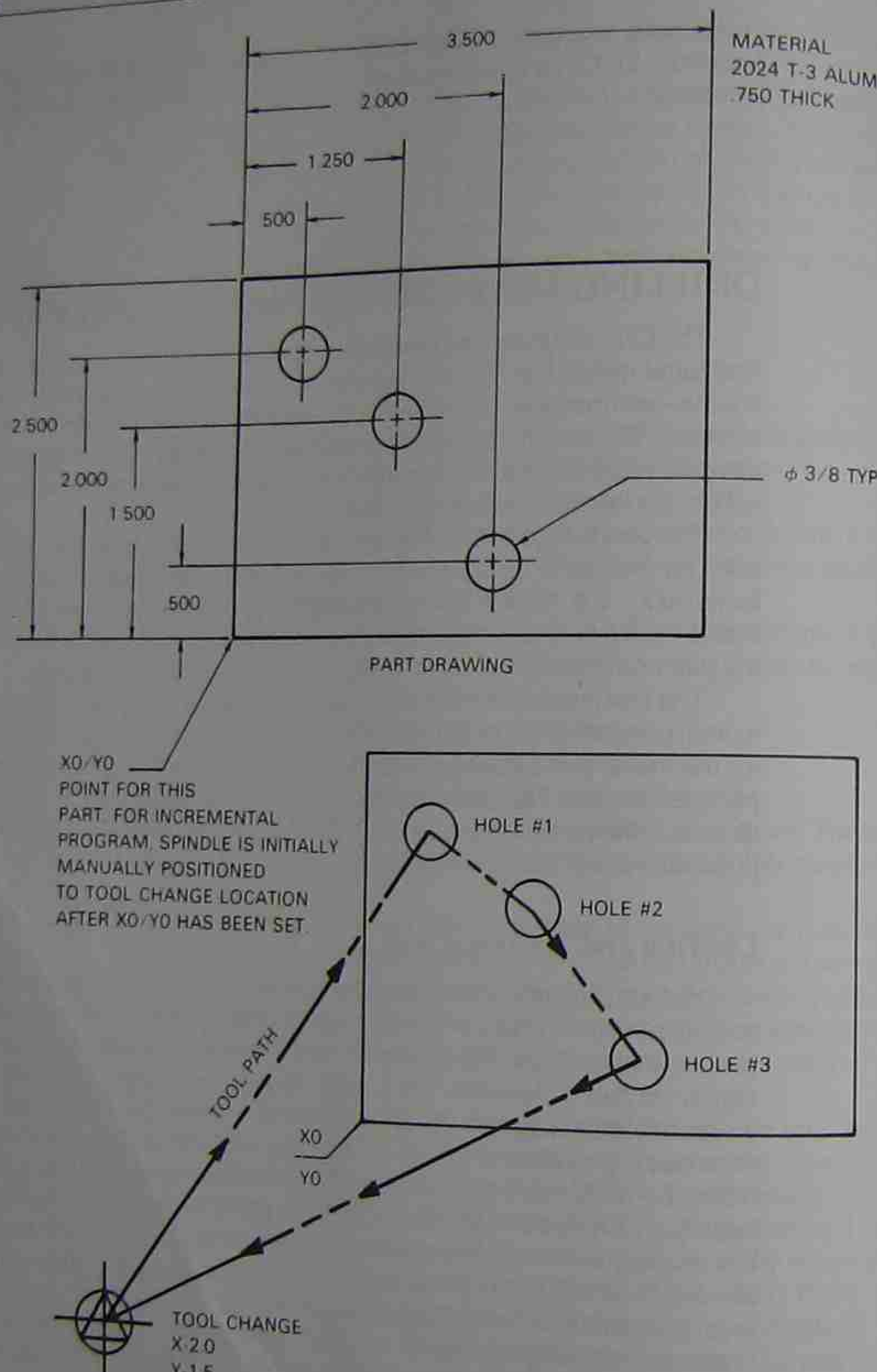


FIGURE 6–1
Hole operations part drawing, nonmetric

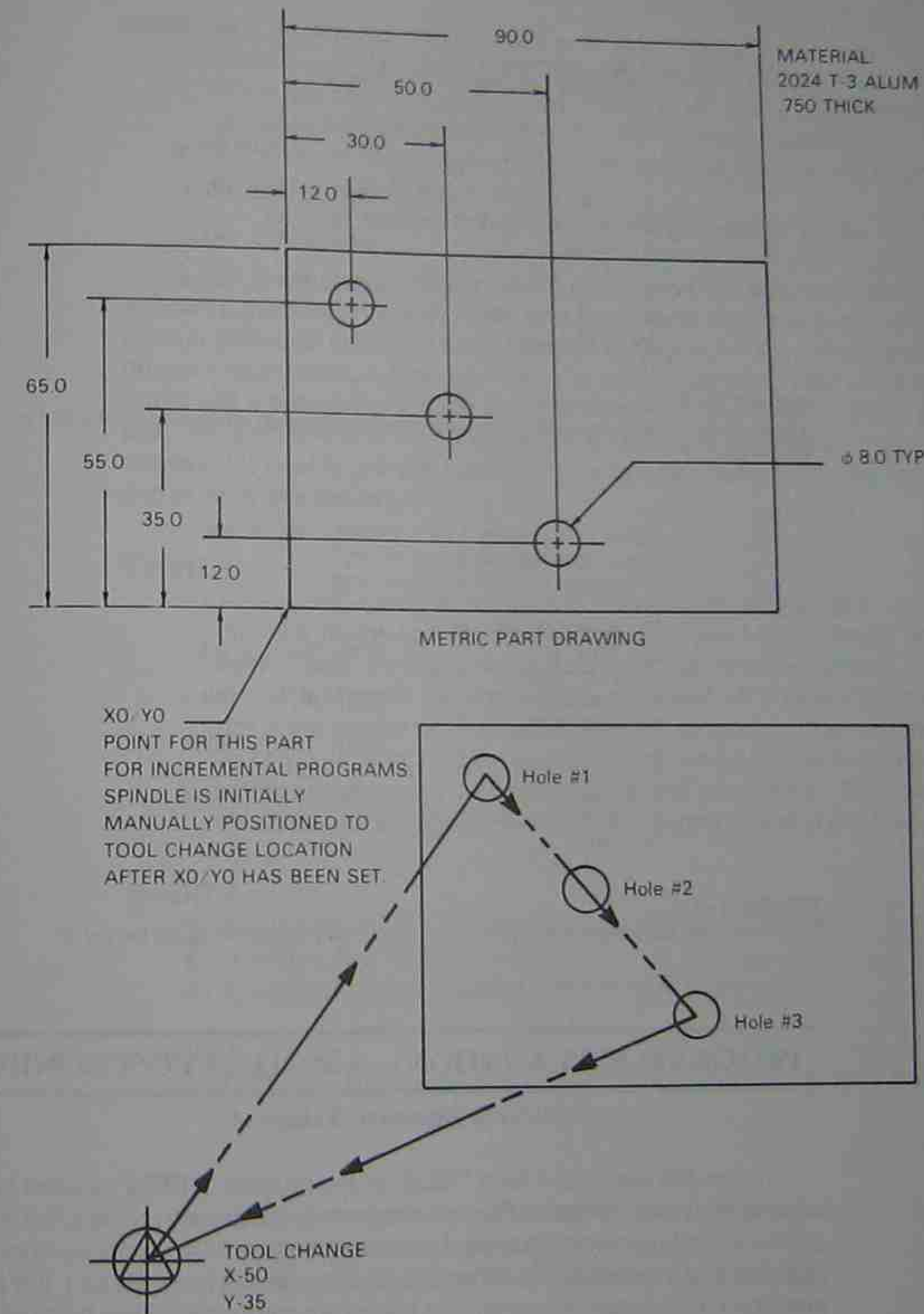


FIGURE 6–2
Hole operations part drawing, metric

```
XO/YO = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-2 Y-1.5
SPINDLE SPEED = 2500 RPM

1  X-2 Y-1.5    R A
2  TOOL 1              REM: 3/8 DRILL
3  X.5 Y2       R A
4  DWELL               REM: DRILL HOLE
5  X1.25 Y1.5 R A
6  DWELL               REM: DRILL HOLE
7  X2 Y.5       R A
8  DWELL               REM: DRILL HOLE
9  X-2 Y-1.5    R A
10 END
```

FIGURE 6–3
Machinist Shop Language drilling program, nonmetric absolute positioning, for the part in Figure 6–1

```
XO/YO = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-50 Y-35
SPINDLE SPEED = 2500 RPM

1  X-50 Y-35    R A
2  TOOL 1              REM: 8mm DRILL
3  X12 Y55      R A
4  DWELL               REM: DRILL HOLE
5  X30 Y35      R A
6  DWELL               REM: DRILL HOLE
7  X50 Y12      R A
8  DWELL               REM: DRILL HOLE
9  X-50 Y-35    R A
10 END
```

FIGURE 6–4
Machinist Shop Language drilling program, metric absolute positioning, for the part in Figure 6–2

PROGRAM EXPLANATION—ABSOLUTE POSITIONING

(Refer to Figures 6–3 and 6–4.)

Notice the use of the term "REM" in the program. "REM" is used in this case for the word "remark." Remark statements are usually provided for by the controller manufacturer. They are ignored by the controller; some symbol or G code is used to precede the statement (in this case the term "REM"). It is good practice to use remark statements in a program manuscript. They help not only the operator to determine what is happening in the program but also aid in debugging the program prior to running the first part.

EVENT 1

- X/Y tool change coordinates. These coordinates position the spindle to the tool change location. The operator will install a 3/8-inch self-centering drill in the spindle (8 mm drill in the metric program).
- R—Instructs the machine to make the move at rapid traverse, which on this machine is 100 in./min.
- A—Tells the machine to use the absolute positioning mode.

A and R are programmed in, using a button on the MCU console. Once activated, rapid/feedrate movement and absolute/incremental positioning remain in force until cancelled by its complementary command. When writing a program manuscript, it is good practice to enter a rapid/feedrate mode command and a positioning system command on every event with motion. If the program is entered in the MCU via the MCU keyboard, the programmer will be reminded to double check the positioning and feed modes for the correct setting at each affected event.

EVENT 2

- TOOL 1—Causes the machine to stop executing the program until the operator depresses the start button. This gives the operator time to safely install the tool without danger of machine movement. The remark (REM) notes that the operator will install a 3/8-inch self-centering drill in the spindle (8-mm drill in the metric program). This machine has no computer control of the spindle motor except the panic stop button, which kills power to the drive motors and the spindle. The operator would turn on the spindle motor to the correct speed at this time.

EVENT 3

- X/Y coordinates—To move from tool change to hole #1.
- R—Specifies rapid traverse.
- A—Specifies that absolute positioning is being used.

EVENT 4

- DWELL—Halts execution of the program until the start button is pushed. The operator then drills hole #1.

EVENT 5

- X/Y coordinates—To move from hole #1 to hole #2.
- R—Specifies rapid traverse.
- A—Specifies that absolute positioning is being used.

EVENT 6

- DWELL—Halts program execution. Hole #2 is drilled.

EVENT 7

- X/Y coordinates—To move from hole #2 to hole #3.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 8

- DWELL—As in events 4 and 6, the program is halted and hole #3 is drilled.

EVENT 9

- X/Y coordinates—To move from hole #3 to tool change. It is good practice to send the spindle back to tool change at the end of a program, even if machining only one part. Aside from forming good habits for multipiece programming, this practice safely positions the tool out of the way of the part.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 10

- END—This instructs the machine that this is the end of the program.

All the moves made in the absolute and incremental programs are in the rapid traverse mode. When drilling holes, the faster the speed of the axis movements between holes, the quicker and more economical is the machining. No feedrate need be entered in the program for movement to hole locations as rapid movement is the most efficient mode. Had this been a three-axis machine, it would have been necessary to assign a feedrate to control the rate at which the drills went thru the part.

PROGRAM EXPLANATION—INCREMENTAL POSITIONING

(Refer to Figures 6–5 and 6–6.)

Figure 6–5 is an incremental program for the part in Figure 6–1. Figure 6–6 is an incremental program for the part in Figure 6–2. The program sequence is identical to the two absolute positioning programs just discussed. Only the coordinates for the three hole locations and tool change location differ. Note that it is mandatory for the spindle to begin and finish at the same location for the program to cycle correctly on to the second part. For the programs in Figures 6–5 and 6–6, the corner of the part was established as the zero point at setup. The machine was then manually positioned to the tool change location. No move to tool change is possible in the first event because the spindle is already in position.


```
PROGRAM STARTING X0/Y0 = TOOL CHANGE
SET TO TOOL CHANGE = X-2 Y-1.5  PRIOR  TO
RUNNING FIRST CYCLE.
TOOL = 3/8 DRILL
SPINDLE SPEED = 2500 RPM

1  TOOL 1              REM: 3/8 DRILL
2  X2.5 Y3.5   R I
3  DWELL               REM: DRILL HOLE
4  X.75 Y-.5   R I
5  DWELL               REM: DRILL HOLE
6  X.75 Y-1    R I
7  DWELL               REM: DRILL HOLE
8  X-4 Y-2     R I
9  END
```

FIGURE 6–5
Machinist Shop Language drilling program, nonmetric incremental positioning, for the part in Figure 6–1

```
PROGRAM STARTING X0/Y0 = TOOL CHANGE
SET TO TOOL CHANGE = X-50  Y-35 PRIOR  TO
RUNNING FIRST CYCLE.
TOOL= 8mm DRILL

1  TOOL 1               REM: 8mm DRILL
2  X62  Y90   R I
3  DWELL                REM: DRILL HOLE
4  X18  Y-20  R I
5  DWELL                REM: DRILL HOLE
6  X20  Y-23  R I
7  DWELL                REM: DRILL HOLE
8  X-100 Y-47  R I
9  END
```

FIGURE 6–6
Machinist Shop Language drilling program, metric incremental positioning, for the part in Figure 6–2

EVENT 1

- TOOL 1—Causes the machine to stop executing the program until the operator depresses the start button. This gives the operator time to safely install the tool without danger of machine movement. Since the spindle was manually positioned to tool change at setup, no movement to tool change is needed.

EVENT 2

- X/Y coordinates—To move from tool change to hole #1.
- R—Specifies rapid traverse.
- I—Specifies that incremental positioning is being used.

EVENT 3
- DWELL—Halts execution of the program until the start button is pushed. The operator then drills hole #1.

EVENT 4
- X/Y coordinates—To move from hole #1 to hole #2.
- R—Specifies rapid traverse.
- I—Specifies that incremental positioning is being used.

EVENT 5
- DWELL—Halts program execution. Hole #2 is drilled.

EVENT 6
- X/Y coordinates—To move from hole #2 to hole #3.
- R—Specifies rapid traverse.
- I—Specifies incremental positioning.

EVENT 7
- DWELL—As in events 3 and 5, the program is halted and hole #3 is drilled.

EVENT 8
- X/Y incremental coordinates—To move from hole #3 to tool change.
- R—Specifies rapid traverse.
- I—Specifies incremental positioning.

EVENT 9
- END—Instructs the machine that this is the end of the program.

WORD ADDRESS FORMAT

The next machine to be programmed is a CNC mill using a General Numerics controller. This machine is a continuous-path machine that uses a programming format called *word address*. Word address was developed as a tape programming format. Another name for word address is *variable block* format, so named because the program lines (blocks) may vary in length according to the information contained in them. Earlier tape formats required an entry for all possible machine registers. In these earlier formats, a zero was programmed as a null input if the register values were to be unaffected, but in word address, the blocks need only contain necessary information. Although developed as a tape format, word address is used as the format for manual data input on many CNC machines.

Addresses

The block format for word address is as follows:

N...G..X....Y....Z....I....J....K....F....S....T..M..

Only the information needed on a line need be given. Each of the letters is called an address (or word). The various words are as follows:

N— Designates the start of a block. Program lines or blocks are sometimes also called *sequence lines*. On some machinery the address "O" may also be used to start a block of information.
G— Initiates a preparatory function. Preparatory functions change the control mode of the machine. Examples of preparatory functions are rapid/feedrate mode, drilling mode, tapping mode, boring mode, and circular interpolation. Preparatory functions are called *prep functions*, or more commonly, *G codes*.
X— Designates an X-axis coordinate. X is also used to enter a time interval for a timed dwell.
Y— Designates a Y-axis coordinate.
Z— Designates a Z-axis coordinate.
I— Identifies the X-axis location of an arc centerpoint.
J— Identifies the Y-axis location of an arc centerpoint.
K— Identifies the Z-axis location of an arc centerpoint.
S— Sets the spindle RPM.
F— Assigns a feedrate.
T— Specifies the tool to be used in a tool change.
M— Initiates miscellaneous functions (*M functions*). M functions control auxilliary functions such as the turning on and off of the spindle and coolant, initiating tool changes, and signaling the end of a program.

Other words used in word address will be explained as they are used. A list of EIA codes for word address is contained in Appendix 1.

DRILLING IN WORD ADDRESS FORMAT

Figure 6–7 contains a program written in the word address format to drill the part in Figure 6–1. Figure 6–8 contains the program to drill the part in Figure 6–2. The program sequence is identical to that used in the Machinist Shop Language example. The specific codes used in the programs are:

G00—Puts the machine in rapid traverse mode. All moves made with G00 active are made in rapid traverse.
G01—Linear interpolation; puts the machine in feedrate mode. All moves made with G01 active are made in a straight line at the programmed feedrate.

```
XO/YO = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-2 Y-1.5
TOOL 3/8 DRILL SPINDLE SPEED 2500 RPM

N010 G00 G70 G90 X-2 Y-1.5  M06   REM:3/8
DRILL
N020 X.5 Y2
N030 G04
N040 X1.25 Y1.5
N050 G04
N060 X2 Y.5
N070 G04
N080 X-2 Y-1.5
N090  M30
```

FIGURE 6–7
Word address format drilling program, nonmetric absolute positioning, for the part in Figure 6–1

```
XO/YO = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-50 Y-35
TOOL 8mm DRILL
SPINDLE SPEED 2500 RPM

N010 G00 G71 G90 X-50 Y-35 M06  REM:8  mm
DRILL
N020 X12 Y55
N030 G04
N040 X30 Y35
N050 G04
N060 X50 Y12
N070 G04
N080 X-50 Y-35
N090  M30
```

FIGURE 6–8
Word address format drilling program, metric absolute positioning, for the part in Figure 6–2

G04—Dwell command. Causes a halt in the program execution until the cycle start button is depressed. Some controllers require the use of M00 rather than G04. However, G04 will be used throughout this particular text as the dwell code.
G70—Selects inch input.
G71—Selects metric input.
G90—Selects absolute positioning.
G91—Selects incremental positioning.
M06—Institutes a tool change. In two-axis operation, this command functions as a dwell.
M30—Signals the end of the program and resets the computer to the start of the program.

PROGRAM EXPLANATION—ABSOLUTE POSITIONING

(Refer to Figures 6–7 and 6–8.)

N010
- N010—The sequence number. The word is ignored by the controller. It is used only to identify a block.
- G00—Puts the machine in rapid traverse mode. Moves will be made at rapid traverse speed until the mode is cancelled with a G01, G02, or G03.
- G70—Selects inch input. All numbers entered will be inch coordinates.
- G71—Selects metric input for the metric program.
- G90—Selects absolute positioning.
- X/Y coordinates of the tool change location—These coordinates are absolute dimensions.
- M06—Tool change command. In a two-axis program, this command acts like a dwell. The machine moves to the X/Y tool change coordinates and halts for a tool change. Note that on some controllers, the M06 command may have to be placed on a program line by itself in order to function as explained here. Also note that some controllers using word address format will not recognize an M06 command when two axes are supplied on a machine. In these cases a G04 (or M00) dwell command will be used.

N020
- N020—The sequence number.
- X/Y coordinates—To move from tool change to hole #1.

N030
- N030—The sequence number.
- G04—The dwell command. The program halts its execution, allowing the operator to drill the holes.

N040
- N040—The sequence number.
- X/Y coordinates—To move from hole #1 to hole #2.

N050
- N050—The sequence number.
- G04—The dwell command. This halts the program so that hole #2 can be drilled.

N060
- N060—The sequence number.
- X/Y coordinates—To move from hole #2 to hole #3.

N070

- N070—The sequence number.
- G04—The dwell command. Hole #3 is drilled.

N080

- N080—The sequence number.
- X/Y coordinates—To move from hole #3 to tool change.

N090

- N090—The sequence number.
- M30—Signals that the program has ended and resets the computer's memory to the start of the sequence.

PROGRAM EXPLANATION—INCREMENTAL POSITIONING

(Refer to Figures 6–9 and 6–10.)

The program sequence used in these examples is identical to that used in the Machinist Shop Language programs in Figures 6–5 and 6–6.)

N010

- N010—The sequence number. The word is ignored by the controller as it is used only to identify a block.
- G00—Puts the machine in rapid traverse mode. Moves will be made at rapid traverse speed until the mode is cancelled with a G01, G02, or G03.
- G70/G71—Selects inch or metric input.
- G91—Specifies that incremental positioning is to be used.
- M06—Tool change command. This halts the program to allow insertion of the drill. Note that the spindle had been manually positioned at the tool change location prior to the start of the first program cycle, just as was done with the Machinist Shop Language programs.

N020

- N020—The sequence number.
- X/Y—The incremental coordinates required to move from tool change to hole #1.

N030

- N030—The sequence number.
- G04—The dwell command. The program halts its execution, allowing the operator to drill hole #1.


```
PROGRAM X0/Y0 = TOOL CHANGE
SET TO TOOL CHANGE = X-2 Y-1.5  PRIOR  TO
STARTING FIRST CYCLE
TOOL 3/8 DRILL SPINDLE SPEED 2500 RPM

N010 G00 G70 G91 M06  REM:3/8 DRILL
N020 X2.5 Y3.5
N030 G04
N040 X.75 Y-.5
N050 G04
N060 X.75 Y-1
N070 G04
N080 X-4 Y-2
N090  M30
```

FIGURE 6–9
Word address format drilling program, nonmetric incremental positioning, for the part in Figure 6–1

```
PROGRAM START X0/Y0 = TOOL CHANGE
SET TO TOOL CHANGE = X-50 Y-35  PRIOR  TO
STARTING FIRST CYCLE
TOOL 8mm DRILL SPINDLE SPEED 2500 RPM

N010 G00 G71 G91 M06  REM:8mm DRILL
N020 X62 Y90
N030 G04              REM:DRILL HOLE
N040 X18 Y-20
N050 G04              REM:DRILL HOLE
N060 X20 Y-23
N070 G04              REM:DRILL HOLE
N080 X-100 Y-47
N090  M30
```

FIGURE 6–10
Word address format drilling program, metric incremental positioning, for the part in Figure 6–2

N040

- N040—The sequence number.
- X/Y—The incremental coordinates required to move from hole #1 to hole #2.

N050

- N050—The sequence number.
- G04—The dwell command. This halts the program so that hole #2 can be drilled.

N060

- N060—The sequence number.
- X/Y—The incremental coordinates required to move from hole #2 to hole #3.

N070

- N070—The sequence number.
- G04—The dwell command. Hole #3 is drilled.

N080

- N080—The sequence number.
- X/Y—The incremental coordinates required to move from hole #3 to tool change.

N090

- N090—The sequence number.
- M30—Signals that the program has ended and resets the computer's memory to the start of the sequence.

MILLING IN MACHINIST SHOP LANGUAGE

Assume that the part in Figure 6–11 is to be milled. The part is an aluminum casting which requires that only the length and width be machined. Figure 6–12 is a metric part. The part setup drawing is Figure 6–13. Clamping will be done through the center hole. Two passes around the part will be made, a roughing pass and a finishing pass. Left for the finish pass will be .010 inch of stock (0.25 mm metric version).

Two programs, one nonmetric and one metric, written using absolute positioning will be presented first. (The programs are contained in Figures 6–15 and 6–16.) A .500-inch-diameter end mill will be used in the nonmetric program and a 5-mm-diameter end mill in the metric version. Notice that only an X or Y coordinate, rather than an X/Y pair of coordinates, is used in some lines. If the machine is already positioned in one of its axes, a coordinate for that axis need not be given. No movement is to take place; therefore the second coordinate is not required.

Up Milling and Down Milling

When milling cuts are programmed, it is important to understand the difference between *up* and *down milling*. Figure 6–14 illustrates these two machining practices. Notice that in up milling (also called *conventional milling*), the cutter forces acting on the part try to lift the part up off of the table, hence the name

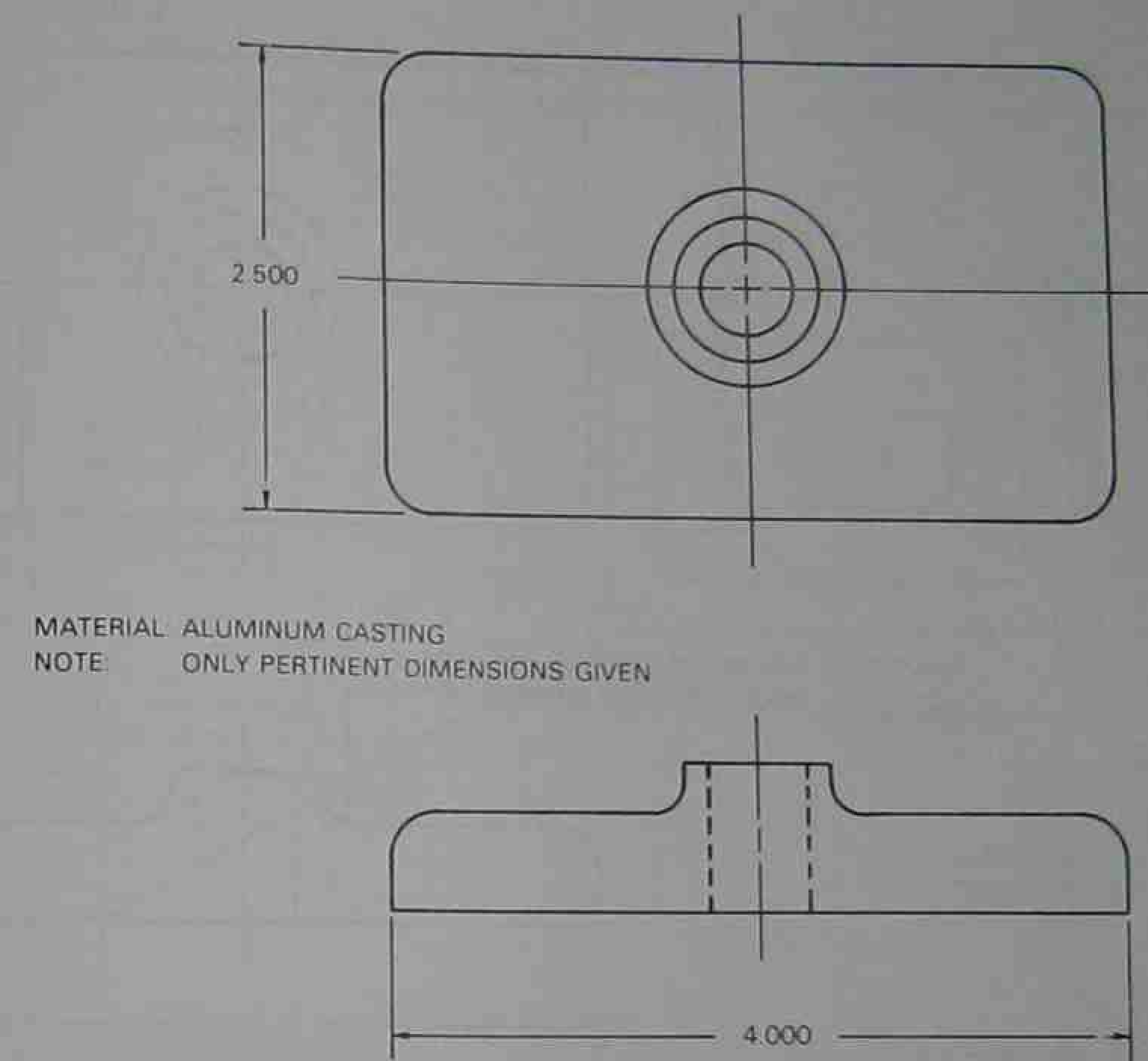


FIGURE 6–11
Milling part drawing, nonmetric

up milling. In down milling the force of the cutter tries to push the part downward onto the table, thus the name down milling. Down milling is also referred to as *climb cutting*, because the cutter is trying to "climb up" on top of the part.

Up milling is used for cutting most ferrous materials, brass and bronze, and for roughing cuts on aluminum and aluminum alloys. Down milling is used for finishing cuts on aluminum and aluminum alloys. It is also occasionally used for finishing cuts on other metals, if conditions warrant it. Down milling requires less power for a particular cut but places more stress on the machine slides and ball screws than does up milling. Exactly when to use up and down milling is something that must be learned through experience as it depends not only on the machine available but also on the cutting tools, coolant, and workpiece materials.

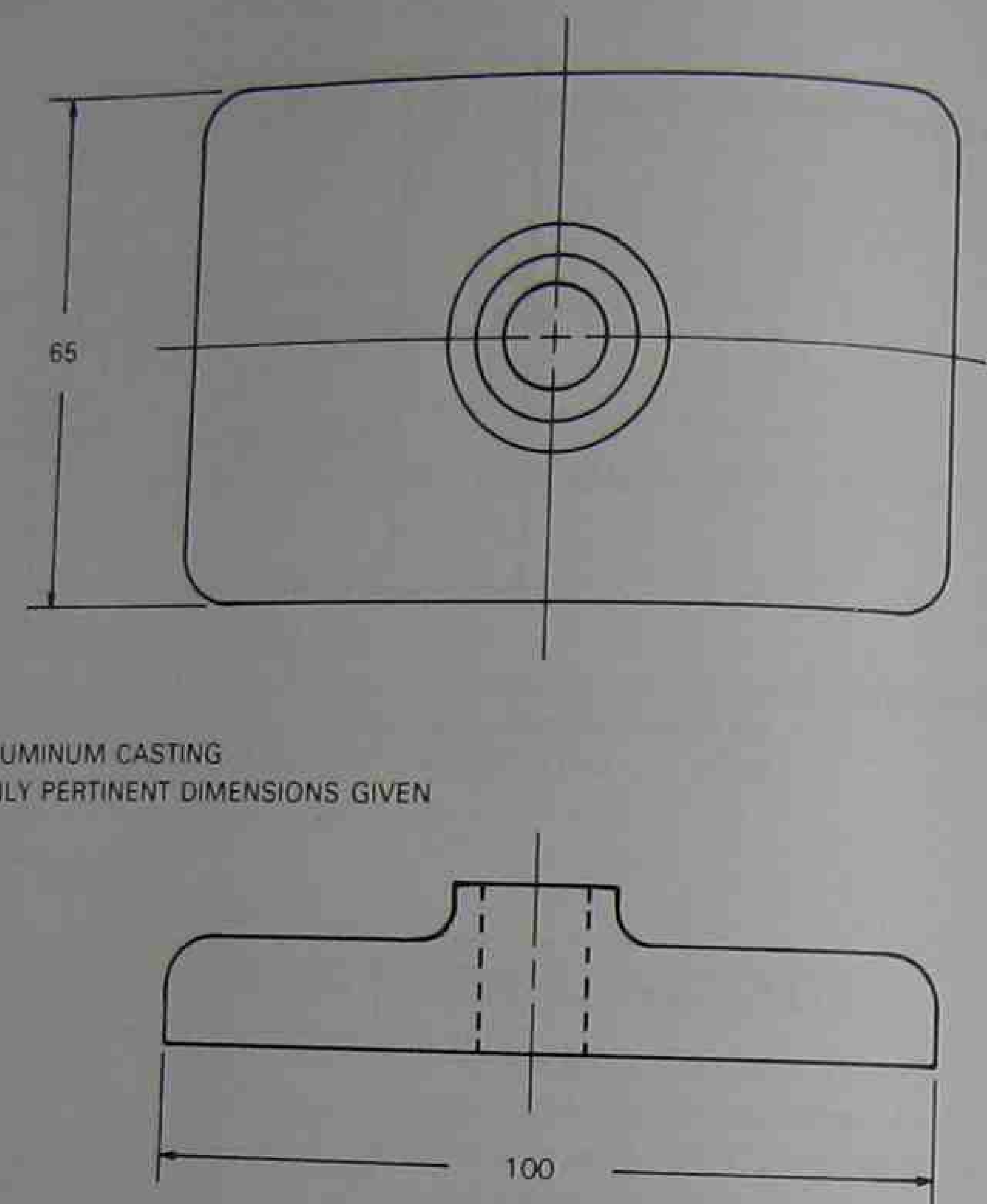


FIGURE 6-12
Milling part drawing, metric

PROGRAM EXPLANATION—ABSOLUTE POSITIONING

(Refer to Figures 6-15 and 6-16.)

EVENT 1

- X/Y coordinates of the tool change location—This move to tool change allows the end mill to be inserted at a safe location.
- R—Specifies a move at rapid traverse.
- A—Specifies absolute positioning.

EVENT 2

- TOOL 1—Halts the program execution to allow the operator to install the end mill in the spindle.

EVENT 3

- FEED 20—Assigns a feedrate of 20 in./min (500 mm/min metric) to be used when needed. A feedrate may be assigned at any point before it is needed.

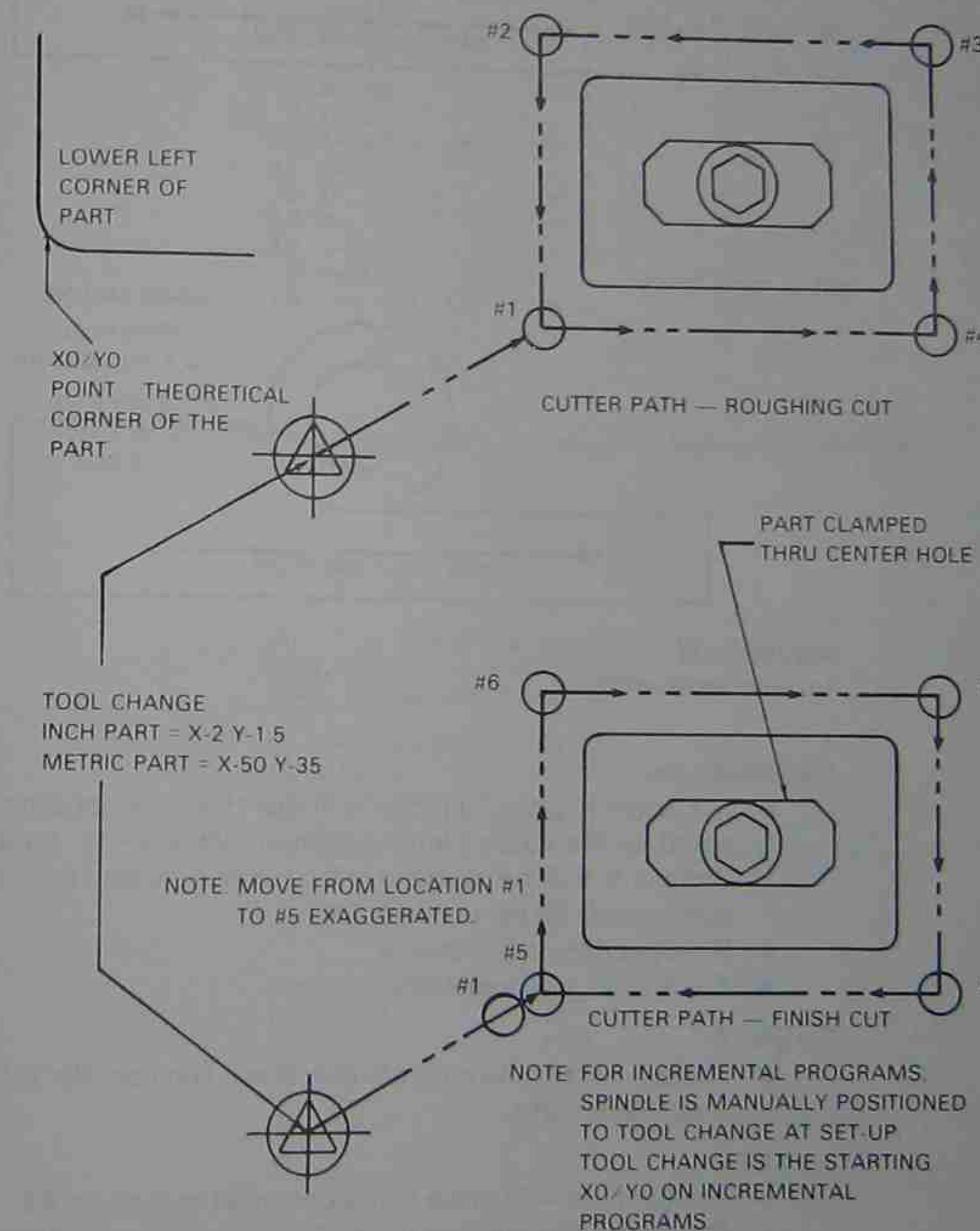


FIGURE 6-13
Setup drawing for the part in Figures 6-11 and 6-12

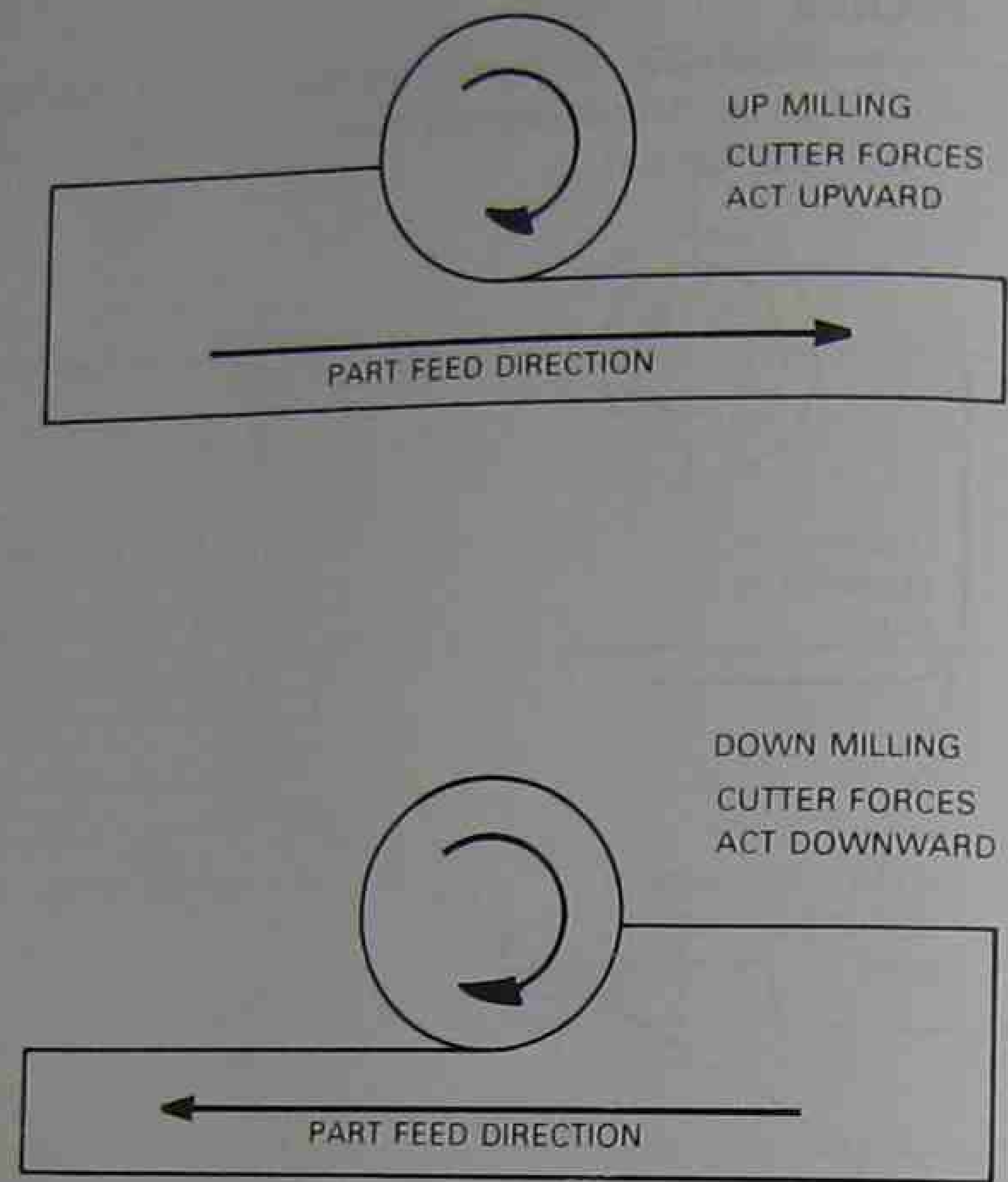


FIGURE 6-14
Up milling and down milling

EVENT 4

- X/Y coordinates—To move from tool change to location #1 as indicated on the cutter path diagram in Figure 6–13. As explained in Chapter 5, half the diameter of the cutter is allowed for in both axes to compensate for the cutter radius.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 5

- DWELL—Halts the program execution. The operator is instructed to lower the spindle.

EVENT 6

- X coordinate—To move from location #1 to location #2.
- F—Specifies a feedrate move. This means that a milling cut feedrate movement is required.
- A—Specifies absolute positioning.


```
XO/YO = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-2 Y-1.5
TOOL = .500 END MILL

 1 X-2 Y-1.5         R A
 2 TOOL 1                  REM:2500 RPM
 3 FEED 20
 4 X-.26 Y-.26       R A
 5 DWELL                   REM:LOWER SPNDL
 6 X4.26             F A
 7 Y2.76             F A
 8 X-.26             F A
 9 Y-.26             F A
10 X-.25 Y-.25       F A
11 Y2.75             F A
12 X4.25             F A
13 Y-.25             F A
14 X-.25             F A
15 DWELL                   REM:RAISE SPNDL
16 X-2 Y-1.5         R A
17 END
```

FIGURE 6-15
Machinist Shop Language milling program, nonmetric absolute positioning, for the part in Figure 6–11

```
XO/YO = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-50 Y-35
TOOL = 5mm END MILL

 1 X-50 Y-35         R A
 2 TOOL 1                  REM:2500 RPM
 3 FEED 500
 4 X-2.75 Y-2.75     R A
 5 DWELL                   REM:LOWER SPNDL
 6 X102.75           F A
 7 Y67.75            F A
 8 X-2.75            F A
 9 Y-2.75            F A
10 X-2.5 Y-2.5       F A
11 Y67.5             F A
12 X102.5            F A
13 Y-2.5             F A
14 X-2.5             F A
15 DWELL                   REM:RAISE SPNDL
16 X-50 Y-35         R A
17 END
```

FIGURE 6-16
Machinist Shop Language milling program, metric absolute positioning, for the part in Figure 6–12

EVENT 7
- Y coordinate—To move from location #2 to location #3.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 8
- X coordinate—To move from location #3 to location #4.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 9
- Y coordinate—To move the machine from location #4 to location #1, to complete the milling of the part.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 10
- X/Y coordinates—To move from location #1 to location #5. This move positions the cutter for the finish pass.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 11
- Y coordinate—To move from location #5 to location #6. Notice that .010 inch of stock (0.25 mm metric) material is removed from the side of the part during this move.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 12
- X coordinate—To move from location #6 to location #7.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 13
- Y coordinate—To move from location #7 to #8.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 14
- X coordinate—To move from location #8 to location #5, thus completing the finish milling cut.

EVENT 15
- DWELL—Halts the program execution. The operator is instructed to raise the spindle.

EVENT 16
- X/Y coordinates—To move from location #5 to tool change.
- R—Specifies rapid traverse. The milling cut is complete. Therefore, a feedrate move is no longer required.
- A—Specifies absolute positioning.

EVENT 17
- END—Signals the end of program.

PROGRAM EXPLANATION—INCREMENTAL POSITIONING

Incremental versions of these programs are featured in Figures 6–17 and 6–18. As with the drilling programs, the machine is positioned manually to the tool change position after the edge of the part has been located. The programs begin and end at the tool change location.

EVENT 1
- TOOL 1—Halts the program execution to allow the operator to install the end mill in the spindle if not done previously.

EVENT 2
- FEED 20—Assigns a feedrate of 20 in./min (500 mm/min metric) to be used when needed.

EVENT 3
- X/Y—Incremental coordinates required to move from tool change to location #1, as indicated on the cutter path diagram in Figure 6–13.
- R—Specifies rapid traverse.
- I—Specifies incremental positioning.

EVENT 4
- DWELL—Halts the program execution. The operator is instructed to lower the spindle.

EVENT 5
- X—Incremental coordinate required to move from location #1 to location #2.
- F—Specifies a feedrate move. This means that a milling cut feedrate movement is required.
- I—Specifies incremental positioning.

```
PROGRAM START X0/Y0 = TOOL CHANGE
SET TO TOOL CHANGE = X-2 Y-1.5 PRIOR TO
STARTING FIRST CYCLE
TOOL = .500 END MILL

                                  REM:2500 RPM
 1 TOOL 1
 2 FEED 20
 3 X1.74 Y1.24       R A
 4 DWELL                          REM:LOWER SPNDL
 5 X4.52             F A
 6 Y3.02             F A
 7 X-4.52            F A
 8 Y-3.02            F A
 9 X.01 Y.01         F A
10 Y3.0              F A
11 X4.5              F A
12 Y-3               F A
13 X-4.5             F A
14 DWELL                          REM:RAISE SPNDL
15 X-1.75 Y-1.25     R A
16 END
```

FIGURE 6–17
Machinist Shop Language milling program, nonmetric incremental positioning, for the part in Figure 6–11

```
PROGRAM START X0/Y0 = TOOL CHANGE
SET TO TOOL CHANGE = X-50 Y-35 PRIOR TO
STARTING FIRST CYCLE
TOOL = 5mm END MILL

 1 TOOL 1                         REM:2500 RPM
 2 FEED 500
3 X47.75 Y32.25  R A
 4 DWELL                          REM:LOWER SPNDL
 5 X105.5            F A
 6 Y70.5             F A
 7 X-105.5           F A
 8 Y-70.5            F A
 9 X.25 Y.25         F A
10 Y70               F A
11 X105              F A
12 Y-70              F A
13 X-105             F A
14 DWELL                          REM:RAISE SPNDL
15 X-47.5 Y-32.5     R A
16 END
```

FIGURE 6–18
Machinist Shop Language milling program, metric incremental positioning, for the part in Figure 6–12

EVENT 6

- Y—Incremental coordinate required to move from location #2 to location #3.
- F—Specifies a feedrate move.
- I—Specifies incremental positioning.

EVENT 7

- X—Incremental coordinate required to move from location #3 to location #4.
- F—Specifies a feedrate move.
- I—Specifies incremental positioning.

EVENT 8

- Y—Incremental coordinate required to move the machine from location #4 to location #1 to complete the roughing cut.
- F—Specifies a feedrate move.
- I—Specifies incremental positioning.

EVENT 9

- X/Y—Incremental coordinates required to move from location #1 to location #5. This move positions the spindle for the finish pass.

EVENT 10

- Y—Incremental coordinate required to move from location #5 to location #6. Left for finish cut is .010 inch of stock (0.25 mm metric).
- F—Specifies a feedrate move.
- I—Specifies incremental positioning.

EVENT 11

- X—Incremental coordinate required to move from location #6 to location #7.
- F—Specifies a feedrate move.
- I—Specifies incremental positioning.

EVENT 12

- Y—Incremental coordinate required to move from location #7 to location #8.
- F—Specifies a feedrate move.
- I—Specifies incremental positioning.

EVENT 13

- X—Incremental coordinate required to move from location #8 to location #5 to complete the finish pass.
- F—Specifies a feedrate move.
- I—Specifies incremental positioning.

EVENT 14

- DWELL—Halts the program execution. The operator is instructed to raise the spindle.

EVENT 15

- X/Y—Incremental coordinates required to move from location #5 to tool change.
- R—Specifies rapid traverse (the milling cut is complete).
- I—Specifies incremental positioning.

EVENT 16

- END—Signals that the program has ended.

MILLING IN WORD ADDRESS FORMAT

The part pictured in Figure 6–11 will now be milled using word address format. Figure 6–19 is the word address program in absolute positioning. Figure 6–20 is the metric version to mill the part in Figure 6–12. Figure 6–21 is an incremental program for the part in Figure 6–11, and Figure 6–22 is the metric version to mill the part in Figure 6–12. The programming logic for these programs is identical to that of the Machinist Shop Language programs.

```
X0/Y0 = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-2 Y-1.5
TOOL = .500 END MILL

N010 G00 G70 G90 X-2 Y-1.5 M06    REM:2500 RPM.
N020 X-.26 Y-.26
N030 G04                          REM:LOWER SPNDL
N040 G01 X4.26 F20
N050 Y2.76
N060 X-.26
N070 Y-.26
N080 X-.25 Y-.25
N090 Y2.75
N100 X4.25
N110 Y-.25
N120 X-.25
N130 G04
N140 G00 X-2 Y-1.5                REM:RAISE SPNDL
N150 M30
```

FIGURE 6–19
Word address format milling program, nonmetric absolute positioning, for the part in Figure 6–11


```
X0/Y0 = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-50 Y-35
TOOL = 5mm END MILL

N010 G00 G71 G90 X-50 Y-35 M06    REM:2500
RPM.
N020 X-2.75 Y-2.75
N030 G04
REM:LOWER SPNDL
N040 G01 X102.75 F500
N050 Y67.75
N060 X-2.75
N070 Y-2.75
N080 X-2.5 Y-2.5
N090 Y67.5
N100 X102.5
N110 Y-2.5
N120 X-2.5
N130 G04
REM:RAISE SPNDL
N140 G00 X-50 Y-35
N150 M30
```

FIGURE 6–20
Word address format milling program, metric absolute positioning, for the part in Figure 6–12

```
PROGRAM START X0/Y0 = TOOL CHANGE
MANUALLY SET TO TOOL CHANGE PRIOR TO
RUNNING FIRST CYCLE
TOOL = .500 END MILL

N010 G00 G70 G91 M06        REM:2500 RPM.
N020 X1.74 Y1.24
N030 G04                    REM:LOWER SPNDL
N040 G01 X4.52 F500
N050 Y3.02
N060 X-4.52
N070 Y-3.02
N080 X.01 Y.01
N090 Y3
N100 X4.5
N110 Y-3
N120 X-4.5
N130 G04                    REM:RAISE SPNDL
N140 G00 X-1.75 Y-1.25
N150 M30
```

FIGURE 6–21
Word address format milling program, nonmetric incremental positioning, for the part in Figure 6–11

```
PROGRAM START X0/Y0 = TOOL CHANGE
MANUALY SET TO TOOL CHANGE PRIOR TO
RUNNING FIRST CYCLE
TOOL = 5mm END MILL

N010 G00 G71 G91 M06     REM:2500 RPM.
N020 X47.25 Y32.25
N030 G04                 REM:LOWER SPNDL
N040 G01 X105.5 F500
N050 Y70.5
N060 X-105.5
N070 Y-70.5
N080 X.25 Y.25
N090 Y70
N100 X105
N110 Y-70
N120 X-105
N130 G04                 REM:RAISE SPNDL
N140 G00 X-47.5 Y-32.5
N150 M30
```

FIGURE 6-22
Word address format milling program, metric incremental positioning, for the part in Figure 6-12

PROGRAM EXPLANATION—ABSOLUTE POSITIONING

(Refer to Figures 6-19 and 6-20.)

N010

- N010—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- G70/G71—Specifies inch/metric input.
- G90—Specifies absolute positioning.
- X/Y coordinates—To move to tool change.
- M06—Initiates a tool change. The .500-inch-diameter end mill (5 mm metric) is installed in the spindle. As previously mentioned, some controllers will not use this command on a manual tool change machine. In those cases, a G04 (dwell) or an M00 (program stop) is used for manual tool changes.

N020

- N020—The sequence number.
- X/Y coordinates—To move from tool change to location #1 (see Figure 6-13).

N030

- N030—The sequence number.
- G04—A dwell command. The operator is instructed to lower the spindle.

N040

- N040—The sequence number.
- G01—Puts the machine in feedrate mode.
- X coordinate—Required to move from location #1 to location #2 (see Figure 6-13).
- F20—Assigns a feedrate of 20 in./min (500 mm/min metric).

N050

- N050—The sequence number.
- Y coordinate—Required to move from location #2 to location #3.

N060

- N060—The sequence number.
- X coordinate—Required to move from location #3 to location #4.

N070

- N070—The sequence number.
- Y coordinate—Required to move from location #4 to location #1. This move completes the milling roughing pass.

N080

- N080—The sequence number.
- X/Y coordinates—Required to move from location #1 to location #5. This is a feed move to position the cutter for the finish pass.

N090

- N090—The sequence number.
- Y coordinate—To move from location #5 to location #6. Notice that this is a down milling cut. Down milling gives a nice surface finish on aluminum alloys such as the aluminum casting used for this example.

N100

- N100—The sequence number.
- X coordinate—To move from location #6 to location #7.

N110

- N110—The sequence number.
- Y coordinate—To move from location #7 to location #8.

N120

- N120—The sequence number.
- X coordinate—To move from location #8 to location #5. This completes the finish milling pass.

N130

- N130—The sequence number.
- G04—A dwell command. The operator is instructed to raise the spindle.

N140

- N140—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- X/Y coordinates—To move from location #5 to tool change.

N150

- N150—The sequence number.
- M30—Signals that the program has ended. The computer's memory is reset to the start of the program.

PROGRAM EXPLANATION—INCREMENTAL POSITIONING

(Refer to Figures 6–21 and 6–22.)

N010

- N010—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- G70/G71—Specifies inch/metric input.
- G91—Specifies incremental positioning.
- M06—Initiates a tool change. The .500-inch-diameter end mill (5 mm metric) is installed in the spindle. For controllers that will not use this command on a manual tool change machine, a G04 (dwell) or an M00 (program stop) is used. To insure that the program started and stopped in the same location, the spindle was positioned at the tool change location prior to starting the program.

N020

- N020—The sequence number.
- X/Y incremental coordinates—To move from tool change to location #1 (see Figure 6–13).

N030

- N030—The sequence number.
- G04—A dwell command. The operator is instructed to lower the spindle.

N040

- N040—The sequence number.
- G01—Puts the machine in feedrate mode.
- X incremental coordinate—Required to move from location #1 to location #2 (see Figure 6–13).
- F20—Assigns a feedrate of 20 in./min (500 mm/min metric).

N050

- N050—The sequence number.
- Y incremental coordinate—Required to move from location #2 to location #3.

N060

- N060—The sequence number.
- X incremental coordinate—Required to move from location #3 to location #4.

N070

- N070—The sequence number.
- Y incremental coordinate—Required to move from location #4 to location #1, completing the milling roughing pass.

N080

- N080—The sequence number.
- X/Y incremental coordinates—Required to move from location #1 to location #5.

N090

- N090—The sequence number.
- Y incremental coordinate—To move from location #5 to location #6.

N100

- N100—The sequence number.
- X incremental coordinate—To move from location #6 to location #7.

N110

- N110—The sequence number.
- Y incremental coordinate—To move from location #7 to location #8.

N120

- N120—The sequence number.
- X incremental coordinate—Required to move from location #8 to location #5, completing the finish milling pass.

N130

- N130—The sequence number.
- G04—A dwell command. The operator is instructed to raise the spindle.

N140

- N140—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- X/Y incremental coordinates — To move from location #5 to tool change.

N150

- N150—The sequence number.
- M30—Signals that the program has ended. The computer's memory is reset to the start of the program.

Note that the programming logic was identical for the absolute and incremental positioning programs. Incremental positioning is primarily used within an absolute program rather than as a program in and of itself. From this point on, incremental coordinates will be used within the body of a general absolute program.

FANUC CONTROLLER APPLICATIONS

Fanuc CNC controllers have become a common choice of CNC manufacturers. In some areas of the country Fanuc controls dominate. This chapter is one of several in which a special section containing solutions to the chapter examples in Fanuc format is presented to assist those schools and industries utilizing Fanuc controllers. There are a few important differences in Fanuc format from those of the generic examples presented in this chapter.

Each part program begins with an O number (the letter "O"). This number identifies the program to the controller. Through the use of "O" numbers, multiple programs can be stored in the MCU memory, and called up as required.

The program manuscript does not contain spaces between command words. While some Fanuc models would ignore the spaces, in some older models, they would cause the controller to go into alarm and halt the reading of the tape. Third, comment lines are placed between parentheses. The open parenthesis "(" is known as a control-out character. When the MCU encounters this character, it interprets all information following as a comment until a closed parenthesis ")" is encountered. The closed parenthesis is called a control-in character.

The program stop command used by Fanuc is M00, as opposed to the G04 command used in the text examples. Also, Fanuc controls do not require the use of trailing zeros. If a whole number is programmed (i.e., 1 inch, 3 inches, etc.), the decimal point must be programmed, but the trailing zeros are omitted. 1.000 would be written 1., 3.000 would be 3..

Figure 6–23 is a Fanuc program written in to drill the part in Figure 6–1 using absolute positioning. Figure 6–24 is a program for the same part using incremental positioning. Figures 6–25 (absolute positioning) and Figure 6–26 (incremental positioning) are mill programs for the part in Figure 6–11. Figure 6–27 shows a Fanuc series 15–T control.

```
%
O0601
(---------------------------------------------------------)
(THIS PROGRAM USES ABSOLUTE POSITIONING)
(X/Y ORIGIN IS LOWER LEFT CORNER OF PART)
(PLACE 3/8 DRILL IN SPINDLE PRIOR TO START OF CYCLE)
(---------------------------------------------------------)
(SET PARAMETERS TO RAPID - INCH INPUT - ABS. POS.)
N010G00G70G90
(MOVE TO 1ST HOLE AND PROG. STOP TO DRILL)
N020X.5Y2.
N030M00
(MOVE TO 2ND HOLE AND PROG. STOP TO DRILL)
N040X1.25Y1.5
N050M00
(MOVE TO 3RD HOLE AND PROG. STOP TO DRILL)
N060X2.Y.5
N070M00
(RETURN TO TOOL CHANGE AND END CYCLE)
N080X-2.Y-1.5
N090M30
%
```

FIGURE 6–23

```
%
O0601
(---------------------------------------------------------)
(THIS PROGRAM USES INCREMENTEAL POSITIONING)
(CYCLE STARTS THE TOOL CHANGE LOCATION)
(PLACE 3/8 DRILL IN SPINDLE PRIOR TO START OF CYCLE)
(---------------------------------------------------------)
(SET PARAMETERS TO RAPID - INCH INPUT - INCR. POS.)
N010G00G70G91
(MOVE TO 1ST HOLE AND PROG. STOP TO DRILL)
N020X2.5Y3.5
N030M00
(MOVE TO 2ND HOLE AND PROG. STOP TO DRILL)
N040X.75Y-.5
N050M00
(MOVE TO 3RD HOLE AND PROG. STOP TO DRILL)
N060X.75Y-1.
N070M00
(RETURN TO TOOL CHANGE AND END CYCLE)
N080X-4.Y-2.
N090M30
%
```

FIGURE 6–24

```
%
O0611
(------------------------------------------------------------)
(THIS PROGRAM USES ABSOLUTE POSITIONING)
(X/Y ORIGIN IS LOWER LEFT CORNER OF PART)
(PLACE 1/2 END MILL IN SPINDLE PRIOR TO START OF CYCLE)
(------------------------------------------------------------)
(SET PARAMETERS TO RAPID - INCH INPUT - ABS. POS.)
N010G00G70G90
(AT PROG. STOP - LOWER SPINDLE AND CLAMP)
N020X-.26Y-.26
N030M00
(BEGIN ROUGH MILL CUT AT FEEDRATE)
N040G01X4.26F20.0
N050Y2.76
N060X-.26
N070Y-.26
(BEGIN FINISH MILL CUT)
N080X-.25Y-.25
N090Y2.75
N100X4.25
N110Y-.25
N120X-.25
(AT PROG. STOP UNCLAMP AND RAISE SPINDLE)
N130M00
(RETURN TO TOOL CHANGE LOCATION AND END CYCLE)
N140G00X-2.Y-1.5
N150M30
%
```

FIGURE 6–25

```
%
O0611
(------------------------------------------------------------)
(THIS PROGRAM USES INCREMENTAL POSITIONING)
(CYCLE STARTS FROM TOOL CHANGE LOCATION)
(PLACE 1/2 END MILL IN SPINDLE PRIOR TO START OF CYCLE)
(------------------------------------------------------------)
(SET PARAMETERS TO RAPID - INCH INPUT - INCR. POS.)
N010G00G70G91
(AT PROG. STOP - LOWER SPINDLE AND CLAMP)
N020X1.74Y1.24
N030M00
(BEGIN ROUGH MILL CUT AT FEEDRATE)
N040G01X4.52F20.0
N050Y3.02
N060X-4.52
N070Y-3.02
(BEGIN FINISH MILL CUT)
N080X.01Y.01
N090Y3.
N100X4.5
N110Y-3.
N120X-4.5
(AT PROG. STOP UNCLAMP AND RAISE SPINDLE)
N130M00
(RETURN TO TOOL CHANGE LOCATION AND END CYCLE)
N140G00X-1.75Y1.25
N150M30
%
```

FIGURE 6–26

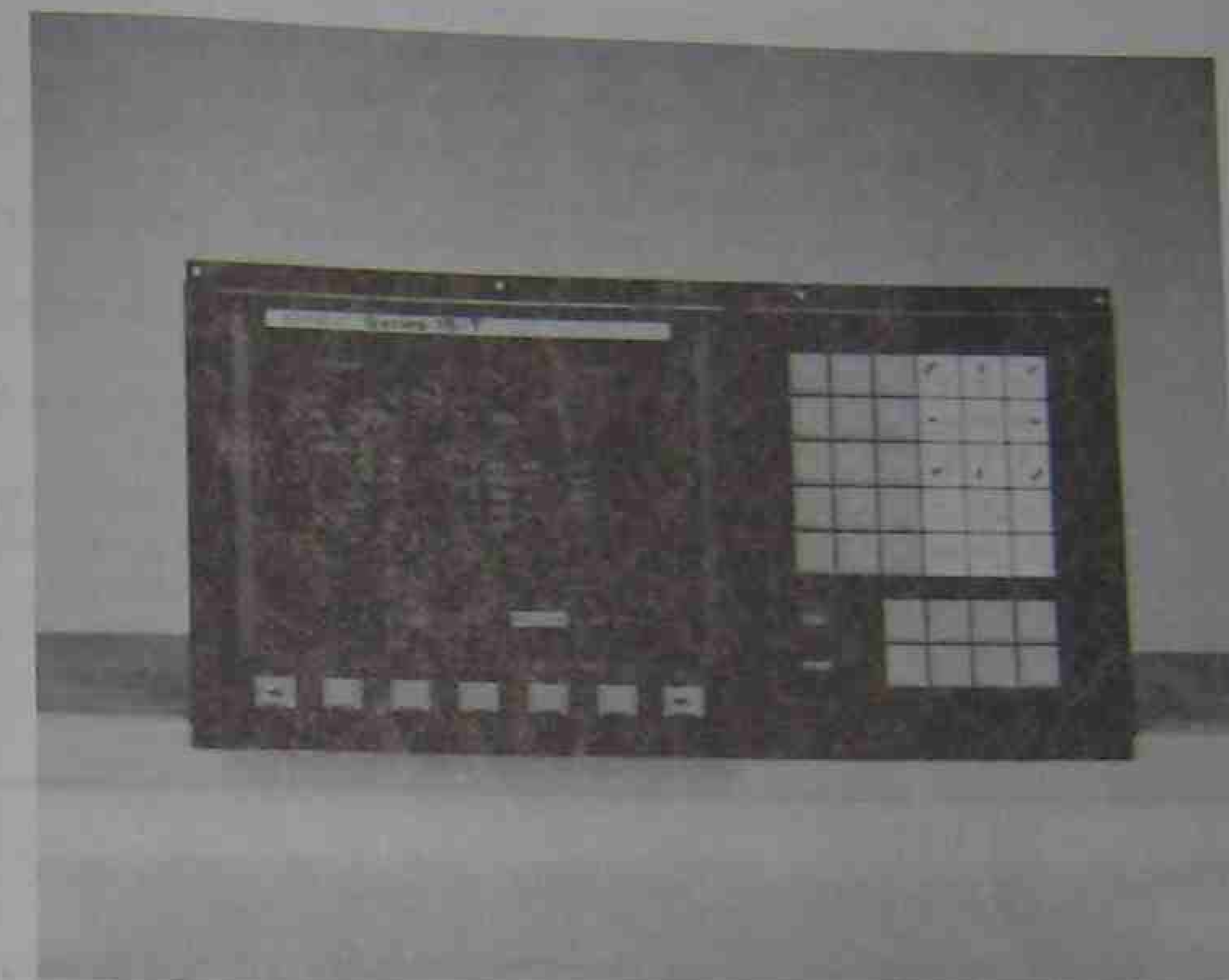


FIGURE 6–27
Fanuc series 15-T control

SUMMARY

The important concepts presented in this chapter are:

- Some procedure for tool change must be included in a program. For a manual tool change mill, a tool change location is used to safely position the spindle away from the part. The program must then be halted to allow the safe insertion of the tool. In Machinist Shop Language, the TOOL command is used to perform this function at the hole location; in word address format, an M06 is used.
- The spindle must be positioned safely out of the way at the end of the program to allow safe loading and unloading of the workpiece. This is accomplished in both the milling and drilling examples by sending the spindle back to its tool change location at the end of the program.

- Incremental programs differ from absolute programs only in the coordinates used. Programs in absolute and incremental positioning use the same programming logic. In incremental positioning, it is imperative that the machine start and stop in the same location. Failure to program for this will result in incorrect positioning for the second cycle.
- To perform hole operations, it is necessary to position the spindle over the centerline of the hole.
- A dwell command is used at hole locations to halt the program and enable the operator to drill the hole.
- When programming coordinates for milling, an allowance must be made for the size of the cutter.
- F is used to specify a feedrate move in Machinist Shop Language.
- R is used to specify a rapid move in Machinist Shop Language.
- G00 is used in word address format to specify a rapid move.
- G01 is used in word address format to specify a feedrate move.

REVIEW QUESTIONS

1. What do each of the following Machinist Shop Language commands mean: X, Y, Z, R, F, A, FEED, DWELL, END?
2. What do the following addresses stand for in word address format: X, Y, G, M, S, F, N?
3. What is a preparatory function?
4. What are miscellaneous functions?

   (Questions #5 – #8 refer to the part in Figure 6 – 28. The cutter path drawing is given in Figure 6 – 29.)
5. Write a program in Machinist Shop Language to mill and drill the part using absolute positioning.
6. Write a program in Machinist Shop Language to mill and drill the part using incremental positioning.
7. Write a program in word address format to mill and drill the part using absolute positioning.
8. Write a program in word address format to mill and drill the part using incremental positioning.

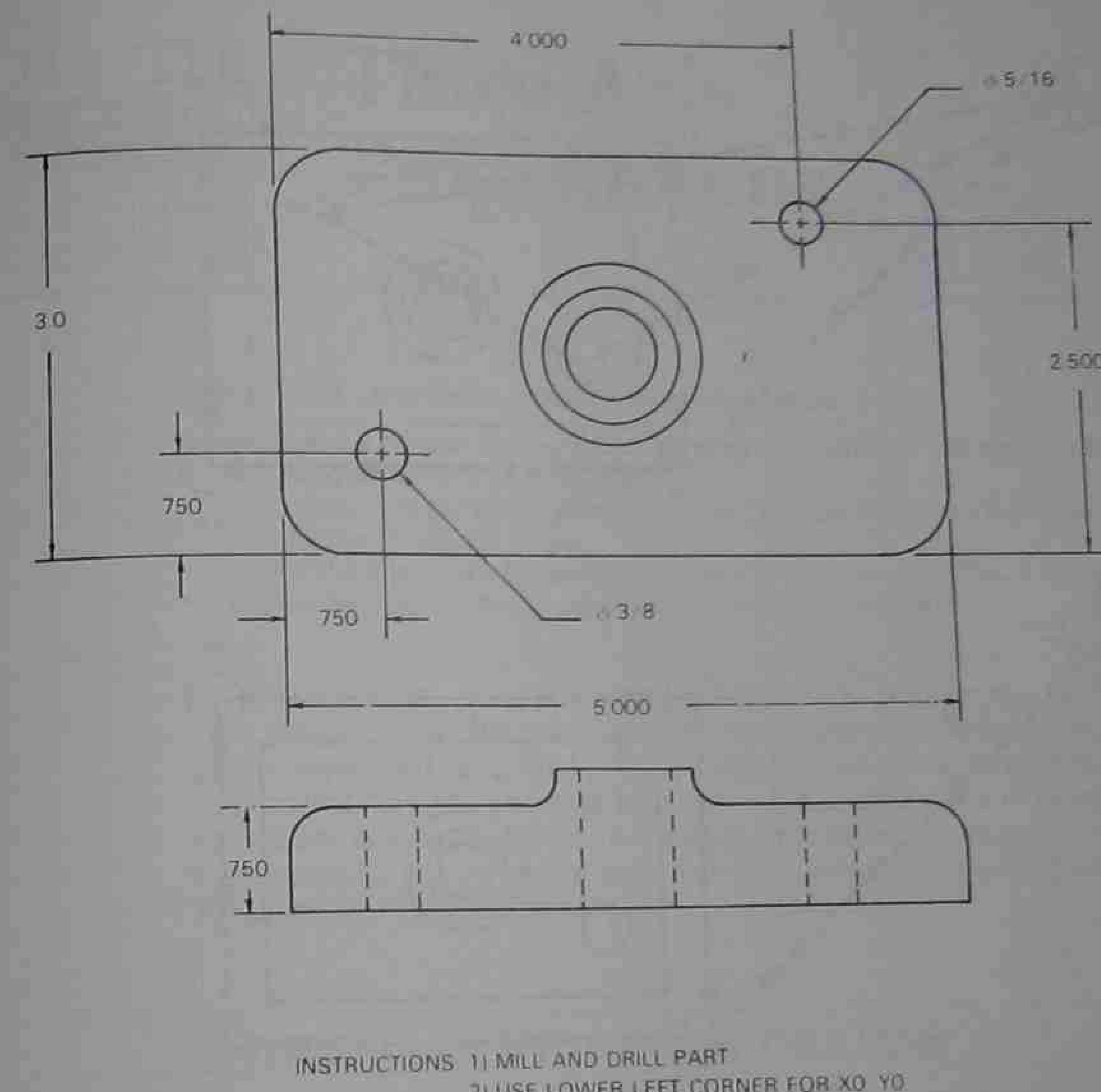


FIGURE 6–28
Part drawing for review questions #5–8

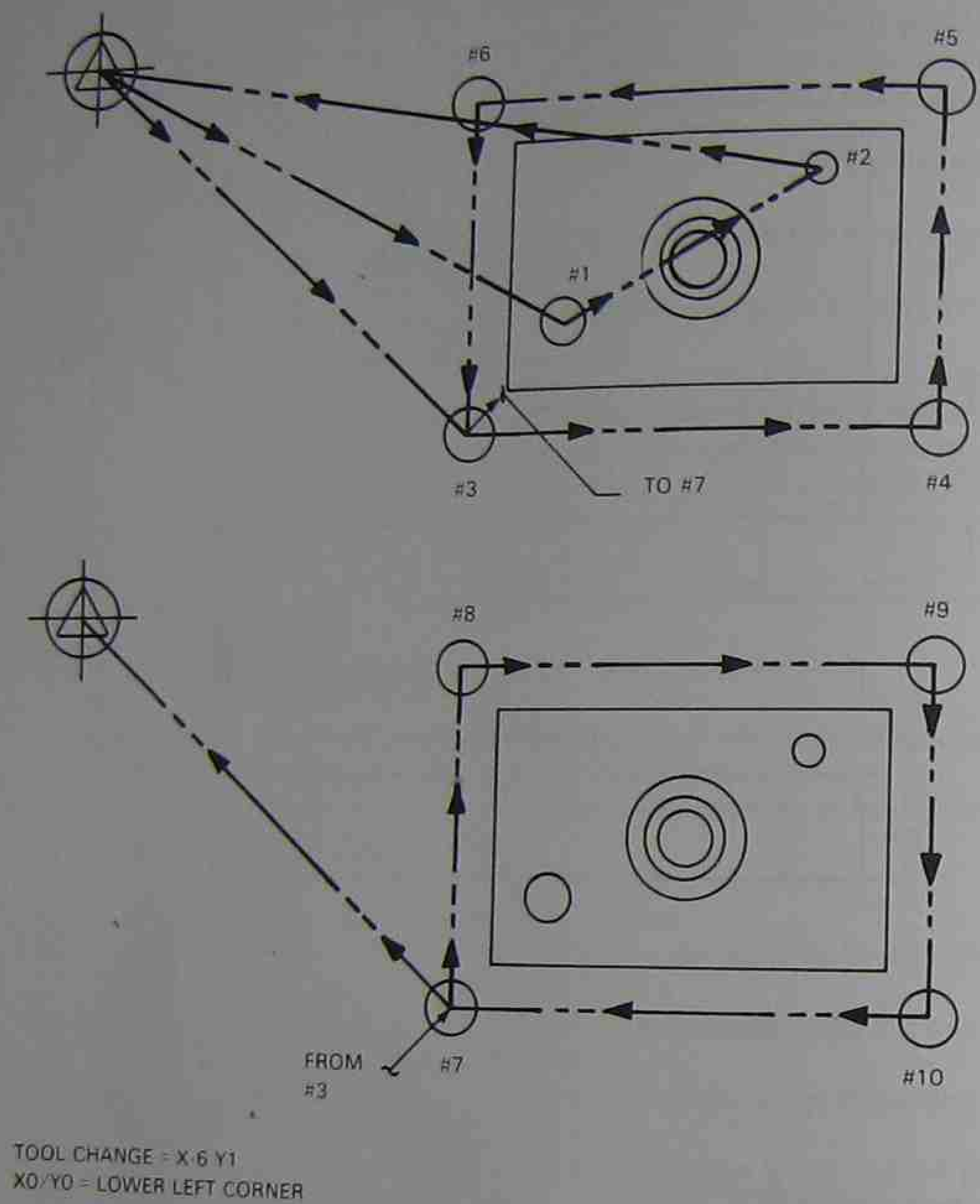


FIGURE 6–29
Cutter path for Figure 6–28

CHAPTER 7

Three-Axis Programming

OBJECTIVES Upon completion of this chapter, you will be able to:

- Write simple programs to perform three-axis hole operations and simple milling cuts using Machinist Shop Language.
- Write simple programs to perform three-axis hole operations and simple milling cuts using word address format.
- Explain the difference between initial level and reference level on CNC machinery.
- Explain the difference between a modal and nonmodal command.

In this chapter, drilling and milling operations are programmed using all three machine axes. The first program is written using Machinist Shop Language; the second using word address format. When writing three-axis programs using the CNC, one must either know the lengths of the tools in their toolholders, or leave empty lines in the program to allow the operator to enter the tool lengths. The concept of tool length offset was discussed in Chapter 4. The programs in this chapter will put the concept to use.

A THREE-AXIS PROGRAMMING TASK

The part in Figure 7–1 is to be milled. The program is to be written in Machinist Shop Language, using incremental positioning. Figure 7–2 depicts the cutter paths necessary to machine the part. In accordance with the cutter path drawing, the following sequence of events is to be performed:

1. At the tool change location, place a drill into the spindle. Move to location #1 and turn the spindle and coolant on.
2. Drill hole #1.
3. Drill hole #2.
4. Drill hole #3.
5. Drill hole #4.
6. Drill hole #5.
7. Turn off the spindle and coolant and return to tool change for a 1-inch-diameter end mill.

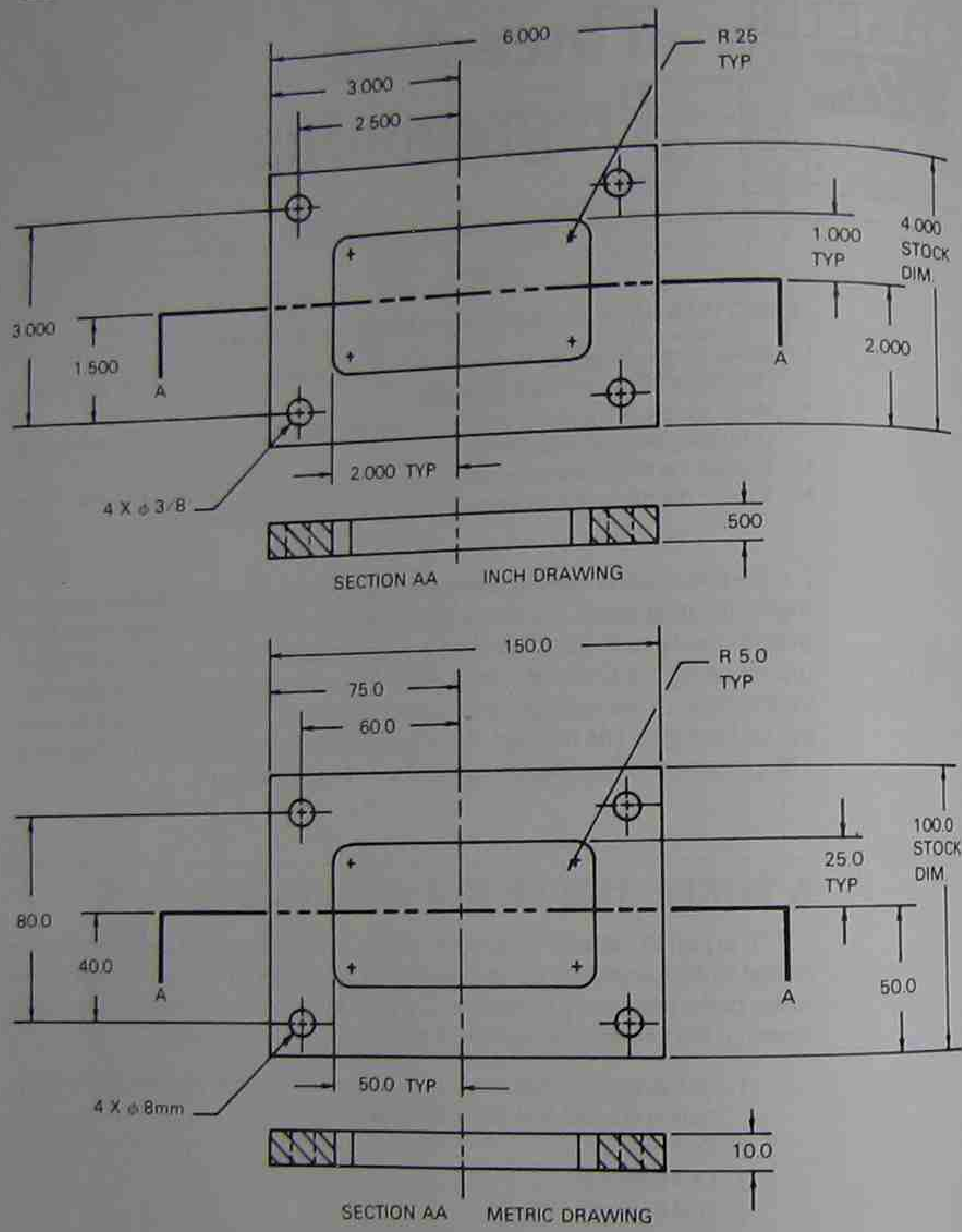


FIGURE 7–1
Part drawing for three-axis programming task

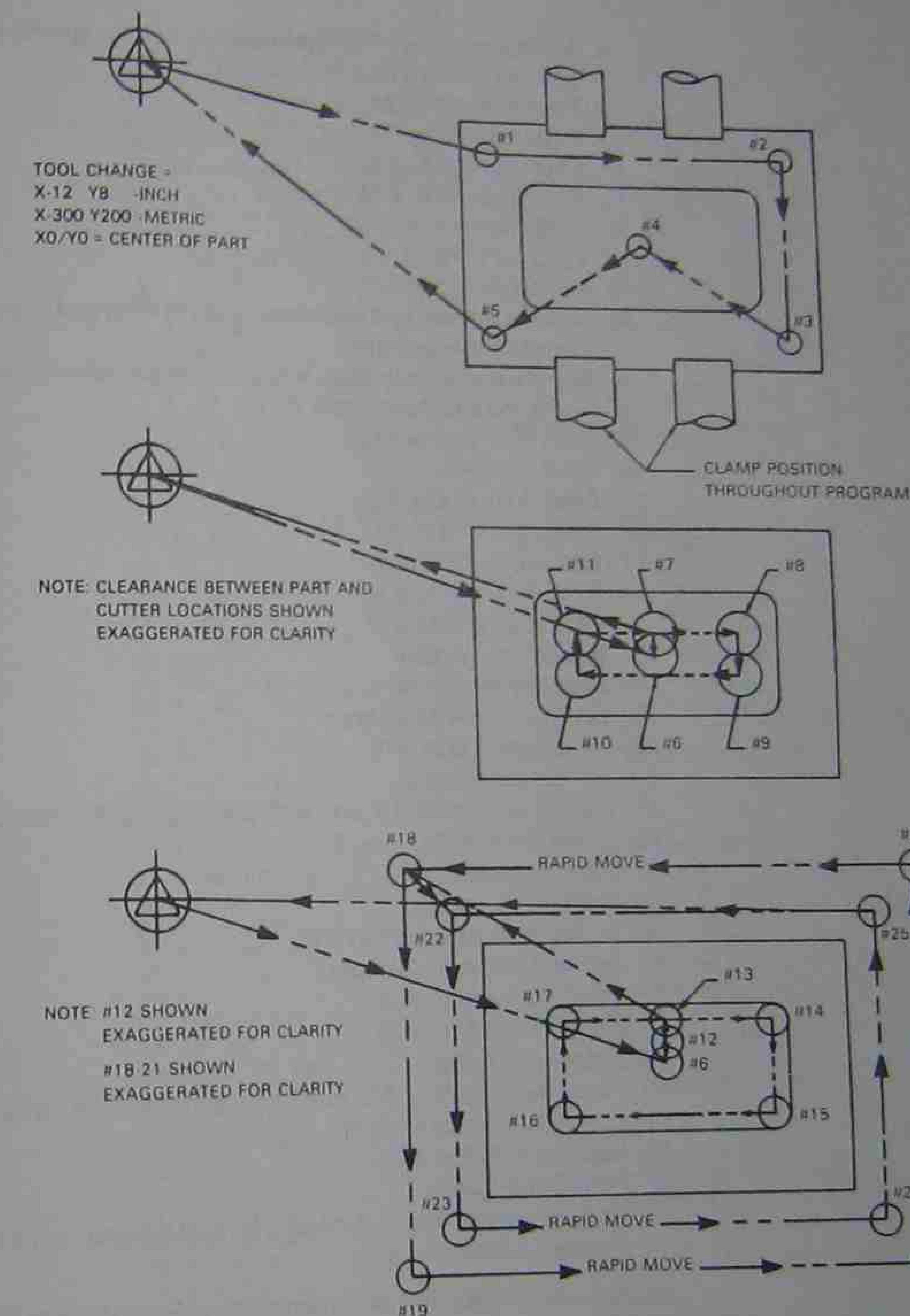


FIGURE 7–2
Cutter paths for part drawing in Figure 7–1

8. Move to location #6 at rapid traverse, turn the spindle and coolant on, and plunge cut a hole thru the part.
9. Feed from #6 to #7.
10. Feed from #7 to #8.
11. Feed from #8 to #9.
12. Feed from #9 to #10.
13. Feed from #10 to #11.
14. Feed from #11 to #7.
15. Retract the spindle.
16. Turn the spindle and coolant off and return to tool change for a .500-inch-diameter end mill.
17. Rapid traverse to location #12, and turn the spindle and coolant on.
18. Lower the spindle to depth.
19. Feed from #12 to #13.
20. Feed from #13 to #14.
21. Feed from #14 to #15.
22. Feed from #15 to #16.
23. Feed from #16 to #17.
24. Feed from #17 to #13.
25. Feed from #13 to #12.
26. Retract the spindle.
27. Rapid traverse from #12 to #18.
28. Lower the spindle to depth.
29. Feed from #18 to #19.
30. Retract the spindle.
31. Rapid traverse from #19 to #20, jumping over the clamps.
32. Lower the spindle to depth.
33. Feed from #20 to #21.
34. Retract the spindle.
35. Rapid move from #21 to #18.
36. Lower the spindle to depth.
37. Feed from #18 to #22.
38. Feed from #22 to #23.
39. Retract the spindle.
40. Rapid traverse from #23 to #24, jumping over the clamps.
41. Lower the spindle to depth.
42. Feed from #24 to #25.
43. Retract the spindle.
44. Turn the spindle and coolant off and rapid traverse to the spindle park position, ending the task.

Figure 7–3 is a Machinist Shop Language program written for inch specifications; Figure 7–4 is an identical program written for metric. Both a roughing and finishing milling cut are taken on the surfaces to be milled. The part is clamped to the table along the surfaces that do not require machining.


```
XO/YO = CENTER OF PART
TOOL CHANGE = X-12 Y8
TOOLS LIST:
TOOL 1 = 3/8 COMB. DRILL
TOOL 2 = 1.000 4 FLUTE END MILL
TOOL 3 = .500 4 FLUTE END MILL
BUFFER HEIGHT:  TOP OF PART FOR TOOL 1,  .100 INCH
FOR TOOLS 2 AND 3
CLEARANCE OVER CLAMPS:  3.000 IN.

 1 TOOL 1001
 2 Z4   A
 3 TOOL 1002
 4 Z3   A
 5 TOOL 1003
 6 Z3.5  A
 7 X-12 Y8 RA
 8 TOOL 1                          REM:3/8 DRILL 1066 RPM
 9 V20 12.8
10 V21 .1
11 G81
12 X-2.5 Y1.5 Z-.62 RA            REM:DRILL #1
13 Z3 RA                          REM:RETRACT SPNDL
14 X2.5 RA                        REM:DRILL #2
15 Y-1.5 RA                       REM:DRILL #3
16 XO YO RA                       REM:DRILL #4
17 X-2.5 Y-1.5 RA                 REM:DRILL #5
18 G80
19 TOOL 0                         REM:CANCEL OFFSET
20 ZO RA                          REM:RETRACT SPNDL
21 X-12 Y8 RA                     REM:TCH
22 TOOL 2                         REM:1.000 E/M 425 RPM
23 FEED 6.8
24 XO YO RA                       REM:RAPID TO #6
25 ZO RA
26 Z-.62 FA                       REM:FEED TO DEPTH
27 Y.48 FA                        REM:FEED TO #7
28 X1.48 FA                       REM:FEED TO #8
29 Y-.48 FA                       REM:FEED TO #9
30 X-1.48 FA                      REM:FEED TO #10
31 Y.48 FA                        REM:FEED TO #11
32 XO FA                          REM:FEED TO #7
33 TOOL 0                         REM:CANCEL OFFSET
34 ZO RA                          REM:RETRACT SPNDL
35 X-12 Y8 RA                     REM:TCH
36 TOOL 3                         REM:.500 E/M 800 RPM
37 FEED 12,8
38 XO YO RA                       REM:RAPID TO #6
39 ZO RA                          REM:RAPID TO BUFFER
40 Z-.62 FA                       REM:FEED TO DEPTH
41 Y.75 FA                        REM:FEED TO #13
42 X1.75 FA                       REM:FEED TO #14
43 Y-.75 FA                       REM:FEED TO #15
44 X-1.75 FA                      REM:FEED TO #16
45 Y.75 FA                        REM:FEED TO #17
46 XO FA                          REM:FEED TO #13
47 Y.74 FA
```

```
48 Z3 RA
49 X-3.26 Y2.26 RA            REM:RAPID TO #18
50 Z0 RA                      REM:RAPID TO BUFFER
51 Z-.62 FA                   REM:FEED TO DEPTH
52 Y-2.26 FA                  REM:FEED TO #19
53 Z3 RA
54 X3.26 RA                   REM:RAPID TO #20
55 Z0 RA                      REM:RAPID TO BUFFER
56 Z-.62 FA                   REM:FEED TO DEPTH
57 Y2.26 FA                   REM:FEED TO #21
58 Z3 RA
59 X-3.26 RA                  REM:RAPID TO #18
60 Z0 RA                      REM:RAPID TO BUFFER
61 Z-.62 FA                   REM:FEED TO DEPTH
62 X-3.25 Y2.25 FA            REM:FEED TO #22
63 Y-2.25 FA                  REM:FEED TO #23
64 Z3 RA
65 X3.25 RA                   REM:RAPID TO #24
66 Z0 RA                      REM:RAPID TO BUFFER
67 Z-.62 FA                   REM:FEED TO DEPTH
68 Y2.25 FA                   REM:FEED TO #25
69 TOOL 0                     REM:CANCEL OFFSET
70 Z0 RA                      REM:RETRACT QUILL
71 X-12 Y8 RA                 REM:TCH
72 END
```

FIGURE 7–3
Machinist Shop Language three-axis program, nonmetric, for the part in Figure 7–1

MACHINIST SHOP LANGUAGE

To use all three axes on the CNC machine, it is necessary to introduce some new commands.

TOOL—Used in Chapter 6 to call up a tool at tool change, TOOL is also used to assign the length of a tool into an offset register.

G—This is a preparatory function (G code). With three-axis operation, many different cycles may be used, each called up by a G code in both Machinist Shop Language and word address format. The G code G04 was used in the last chapter to cause a dwell to occur in word address format.

V—This means *variable*, and allows the programmer to assign values to specific things. V20, for example, specifies the feedrate for the Z axis when using G codes. V21 specifies the height off the surface of the workpiece at which the tool begins and ends its feedrate movements when using G codes. A listing of Machinist Shop Language G code and V code commands is contained in Appendix 2.


```
X0/Y0 = CENTER OF PART
TOOL CHANGE = X-300 Y200
TOOLS LIST:
TOOL 1 = 8mm COMB. DRILL
TOOL 2 = 25mm 4 FLUTE END MILL
TOOL 3 = 10mm 4 FLUTE END MILL
BUFFER HEIGHT:  TOP OF PART FOR TOOL 1,2.54mm FOR
TOOLS 2 AND 3
CLEARANCE OVER CLAMPS:  75mm

 1 TOOL 1001
 2 Z100 A
 3 TOOL 1002
 4 Z75  A
 5 TOOL 1003
 6 Z87.5 A
 7 X-300 Y200 RA
 8 TOOL 1                     REM:8mm DRILL 1066 RPM
 9 V20 216
10 V21 2.54
11 G81
12 X-60 Y40 Z-13 RA           REM:DRILL #1
13 Z75 RA                     REM:RETRACT SPNDL
14 X60 RA                     REM:DRILL #2
15 Y-40 RA                    REM:DRILL #3
16 X0 Y0 RA                   REM:DRILL #4
17 X-60 Y-40 RA               REM:DRILL #5
18 G80
19 TOOL 0                     REM:CANCEL OFFSET
20 Z0 RA                      REM:RETRACT SPNDL
21 X-300 Y200 RA              REM:TCH
22 TOOL 2                     REM:25mm E/M 425 RPM
23 FEED 172.5
24 X0 Y0 RA                   REM:RAPID TO #6
25 Z0 RA
26 Z-13 FA                    REM:FEED TO DEPTH
27 Y12.25 FA                  REM:FEED TO #7
28 X37.25 FA                  REM:FEED TO #8
29 Y-12.25 FA                 REM:FEED TO #9
30 X-37.25 FA                 REM:FEED TO #10
31 Y12.25 FA                  REM:FEED TO #11
32 X0 FA                      REM:FEED TO #7
33 TOOL 0                     REM:CANCEL OFFSET
34 Z0 RA                      REM:RETRACT SPNDL
35 X-300 Y200 RA              REM:TCH
36 TOOL 3                     REM:10.0 E/M 800 RPM
37 FEED 325.1
38 X0 Y0 RA                   REM:RAPID TO #6
39 Z0 RA                      REM:RAPID TO BUFFER
40 Z-13 FA                    REM:FEED TO DEPTH
41 Y20 FA                     REM:FEED TO #13
42 X45 FA                     REM:FEED TO #14
43 Y-20 FA                    REM:FEED TO #15
44 X-45 FA                    REM:FEED TO #16
45 Y20 FA                     REM:FEED TO #17
46 X0 FA                      REM:FEED TO #13
47 Y44.75 FA
48 Z75 RA
```

```
49 X-80.25 Y55.25 RA      REM:RAPID TO #18
50 Z0 RA                  REM:RAPID TO BUFFER
51 Z-13 FA                REM:FEED TO DEPTH
52 Y-55.25 FA             REM:FEED TO #19
53 Z75 RA
54 X80.25 RA              REM:RAPID TO #20
55 Z0 RA                  REM:RAPID TO BUFFER
56 Z-13 FA                REM:FEED TO DEPTH
57 Y55.25 FA              REM:FEED TO #21
58 Z75 RA
59 X-80.25 RA             REM:RAPID TO #18
60 Z0 RA                  REM:RAPID TO BUFFER
61 Z-13 FA                REM:FEED TO DEPTH
62 X-80 Y55 FA            REM:FEED TO #22
63 Y-55 FA                REM:FEED TO #23
64 Z75 RA
65 X80 RA                 REM:RAPID TO #24
66 Z0 RA                  REM:RAPID TO BUFFER
67 Z-13 FA                REM:FEED TO DEPTH
68 Y55 FA                 REM:FEED TO #25
69 TOOL 0                 REM:CANCEL OFFSET
70 Z0 RA                  REM:RETRACT QUILL
71 X-300 Y200 RA          REM:TCH
72 END
```

FIGURE 7–4
Machinist Shop Language three-axis program, metric, for the part in Figure 7–2

Tool Length Offsets

As noted, tool length offsets are assigned using the TOOL command. If the tool lengths are not known by the programmer (and they usually are not), a blank event is created in the program at the proper spot to allow the setup man to enter tool length values. In this chapter it is assumed that the tools have been measured and the lengths are known: a 3/8-inch combination center drill/drill; 1-inch-diameter, four-flute end mill; and 1/2-inch-diameter, four-flute end mill. Their lengths are as follows:

- Drill—1.000 inch long
- 1-inch-diameter end mill—2.000 inches long
- 1/2-inch-diameter end mill—1.500 inches long

A buffer area of .100 inch is to be used between the top of the part and the tool when the tool offset is active for the two end mills. Because of the way a Machinist Shop Language preparatory function cycle operates, the buffer for the drill will be programmed within the drilling cycle. The buffer is established by setting the tool height using a .100-inch gaging block on top of the part as explained in Chapter 4. A total of 3 inches of clearance is desired between the start of the buffer level and the longest tool when the spindle is retracted.

The format for assigning tool lengths is

TOOL 10##

Where the first two digits, "10" (which always remain the same) tell the controller that tool information will be defined in the following event, and the second two digits (##) are the tool number of the tool being defined. Although they may be given at any time before they are used, tool statements are generally placed first in the program for the convenience of the setup man.

The tool statements for the tools used in the following part program are:

1. TOOL 1001	3. TOOL 1002	5. TOOL 1004
2. Z4 A	4. Z3 A	6. Z3.5 A

For the first tool, the tool command specifies that tool information for tool 01 is being assigned. The second event sets the first tool offset at 4 inches. With the longest tool (2 inches) in the spindle, the clearance zone is 3 inches. With an inch-long tool, the clearance zone increases to 4 inches. That is, 4 inches is the distance necessary to move the end of the tool from the spindle-retracted position to the start of the buffer zone. The offset is entered as an absolute Z coordinate. The remaining tools are entered in like manner.

Remark Statements

Notice the use of the term "REM" in the program. REM, as used in this case, stands for the word *remark* (or reminder). Remark statements are usually provided for by the controller manufacturer and are ignored by the controller. Their inclusion in the program has no effect on the actual program; their main purpose in the program listing is to remind the operator what is happening in the program or to tell someone else what the program intends to accomplish in each part. They also aid in debugging the program prior to milling the first part.

In word address systems, a comment (remark) is often a statement placed between parentheses on a line of its own. While displayed on the machine MCU monitor, the comment is ignored by the controller. For consistency however, this text will use the term REM throughout the examples to indicate remarks to the student.

PROGRAM EXPLANATION

(Refer to Figures 7–3 and 7–4.)

EVENT 1

- TOOL 1001—Signals the MCU that a tool length is to be assigned for tool #1. The first two digits (10) specify that tool information is contained in the event that follows. The last two digits (01) tell the MCU that this offset is to be assigned to tool #1. The offset will become active when the command TOOL 1 is given.

EVENT 2

- Z coordinate that equals the tool offset value—This coordinate is always entered in absolute mode.

EVENT 3

- TOOL 1002—Assigns the offset in Event 4 to tool #2.

EVENT 4

- Z coordinate—For tool #2 offset.

EVENT 5

- TOOL 1003—Assigns the offset in Event 6 to tool #3.

EVENT 6

- Z coordinate—For tool #3 offset.

EVENT 7

- X/Y coordinates—For the tool change location.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 8

- TOOL 1—In three-axis operation, this causes two things to happen: dwell is automatically assigned to the controller, thereby allowing the operator to install the first tool in the spindle; and the offset that was entered using TOOL 1001 is activated.

EVENT 9

- V20—A variable register code that is unique to Machinist Shop Language. V20 is used to assign a feedrate to be used by the Z axis whenever an 80 series G code cycle is called up. G code cycles are often called *canned cycles* because they are built into the executive program. In this case, a feedrate of 12.8 in./min (172.5 mm/min) is used.

EVENT 10

- V21—Sets the amount of buffer to be established between the Z0 point (offset active) and the tool. A buffer gives a safety cushion for the tool to begin the feedrate move. Leaving a cushion allows for deceleration of the tool, and insures that the feedrate will be active when the tool cutting edge contacts the metal. V21.1 sets a .100-inch (2.54-mm) buffer zone. With this feature, a buffer need not be built into the drill, as was done with the end mills at setup. Appendix 2 lists V codes used in Machinist Shop Language.

EVENT 11

- G81—Calls up the canned drilling cycle. When a G81 is issued, the spindle rapids to the X/Y coordinate, rapids to the start of the buffer zone, feeds to the indicated Z axis coordinate, and rapids back out to the start of the buffer zone. At the end of this chapter is a brief summary of the more common G codes. Appendix 2 lists all the G codes used in Machinist Shop Language.

EVENT 12

- X/Y coordinates—To move from tool change to hole #1 and drill the hole.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 13

- Z-axis coordinate—To raise the spindle. Since a 3-inch clearance was allowed with the longest tool, there are at least three inches of upward movement possible for the spindle. With the G81 active, the spindle retracted to the buffer zone height after drilling hole #1. A clamp is in the way of the move to hole #2. The spindle was therefore raised to allow the tool to clear the clamp. Another technique that could be employed here is to cancel the G code, cancel the tool offset, raise the spindle to Z0 (the fully retracted position), call up the tool offset, and reinstitute the G code. By using the practice chosen, the tool offset remains active the entire time the tool is used.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 14

- X-axis coordinate—To move from hole #1 to #2. G81 is still active; therefore hole #2 is drilled.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 15

- Y-axis coordinate—To drill hole # 3.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 16

- X/Y coordinates—To drill hole #4.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 17
- X/Y coordinates—To drill hole #5.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 18
- G80—Cancels the drilling cycle.

EVENT 19
- TOOL 0—Cancels the tool length offset. Z0 is now the fully retracted spindle position.

EVENT 20
- Z0—Retracts the spindle.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 21
- X/Y coordinates—To move from hole #5 to tool change.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 22
- TOOL 2—Calls up the offset for tool #2 and halts the program so that the operator can install the end mill.

EVENT 23
- FEED 6.8 (172.5 in the metric version)—Assigns a feedrate to be used for feedrate moves.

EVENT 24
- X/Y coordinates—To move from tool change to location #6.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 25
- Z0 coordinate—Rapids the spindle to the start of the .100 buffer zone that was built into the tool offset at setup per the programmer's instruction.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 26
- Z coordinate—Feeds the end mill to depth. The coordinate is derived by adding the thickness of the part, the height of the buffer zone, and the additional space below the part that the programmer desires the tool to feed.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 27
- Y coordinate—To feed from #6 to #7.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 28
- X coordinate—To feed from #7 to #8.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 29
- Y coordinate—To feed from #8 to #9.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 30
- X coordinate—To feed from #9 to #10.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 31
- Y coordinate—To feed from #10 to #11.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 32
- X coordinate—To feed from #11 to #7.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 33
- TOOL 0—Cancels the active tool offset. Z0 becomes the fully retracted spindle position.

EVENT 34
- Z0—Positions the spindle at the fully retracted location.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 35
- X/Y coordinates—To move from #7 to tool change.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 36
- TOOL 3—Calls up the offset for tool #3. The .500-inch-diameter (10-mm metric) end mill is installed in the spindle.

EVENT 37
- FEED 12.8 (325.1 in the metric version)—Assigns a feedrate to be used for feedrate moves.

EVENT 38
- X0 Y0 coordinates—To move from tool change to location #6.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 39
- Z0—Moves the spindle to the start of the buffer zone. Z0 is .100 above the top of the part when the tool offset is active.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 40
- Z coordinate—To feed the end mill to depth.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 41
- Y coordinate—To feed from #12 to #13.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 42
- X coordinate—To feed from #13 to #14.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 43
- Y coordinate—To feed from #14 to #15.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 44
- X coordinate—To feed from #15 to #16.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 45
- Y coordinate—To feed from #16 to #17.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 46
- X axis coordinate—To feed from #17 to #13.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 47
- Y coordinate—To feed from #13 to #12.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 48
- Z coordinate—To retract the spindle to clear the clamps.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 49
- X/Y coordinates—To move from #12 to #18.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 50
- Z0—Rapids the spindle to the start of the buffer zone.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 51
- Z coordinate—To feed the end mill to depth.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 52
- Y coordinate—To feed from #18 to #19.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 53
- Z coordinate—To raise the spindle to clear the clamps.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 54
- X coordinate—To move from #19 to #20.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 55
- Z0—Positions the spindle at the start of the buffer zone.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 56
- Z coordinate—To feed the end mill to depth.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 57
- Y coordinate—To feed from #20 to #21.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 58
- Z coordinate—To raise the spindle to clear the clamps.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 59
- X coordinate—To move from #21 to #18.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 60
- Z0—Positions the spindle at the start of the buffer zone.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 61
- Z coordinate—To feed the end mill to depth.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 62
- X/Y coordinates—To feed from #18 to #22. This move positions the cutter for the finish cut on the outside surfaces of the part.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 63
- Y coordinate—To feed from #22 to #23.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 64
- Z coordinate—To raise the spindle to clear the clamps.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 65
- X coordinate—To move from #23 to #24.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 66
- Z0—Positions the spindle at the start of the buffer zone.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 67
- Z coordinate—To feed the end mill to depth.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 68
- Y coordinate—To feed from #24 to #25.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 69
- TOOL 0—Cancels the tool offset, making Z0 the fully retracted spindle position.

EVENT 70
- Z0—Fully retracts the spindle.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 71
- X/Y coordinates—Specifies the tool change location.
- R—Specifies rapid traverse.
- A—Specifies absolute positioning.

EVENT 72
- END—Signals the end of the program. The computer's memory resets to the start of the program.

Modal/Nonmodal Commands

Notice that in this program, and in those in the previous chapter, certain commands remained active until canceled by another code. Codes that are active for more than the line in which they are issued are called *modal* commands. In this program, rapid traverse, feedrate moves, and the G81 canned cycle were examples of modal commands. A *nonmodal* command is one which is active only in the program line in which it is issued. TOOL and DWELL are examples of nonmodal commands.

WORD ADDRESS FORMAT

The part program using word address format follows the same basic programming logic as that just given in Machinist Shop Language. Although the sequence of operations is the same, the codes used to carry out the operations vary somewhat from the Machinist Shop Language commands. They are:

G00—Selects rapid traverse mode.
G01—Selects feedrate mode.
G70—As in two-axis operation, selects inch input.
G90/G91—As in two-axis programming, selects absolute or incremental positioning.
G10—Used when assigning tool length offsets, G10 fulfills the same function as the first two digits in the Machinist Shop Language command TOOL 1001; namely, to tell the MCU that tool length information is to be assigned.
H—Used to assign a tool register (just as the last two digits of a TOOL 10## in Machinist Shop Language did). H01 would assign the information given to offset register #1. H02 would assign the information to offset register #2. H is used in conjunction with G10. A tool assignment statement would be: G10 H##, where ## is the register number.
G45—Calls up the tool length offset. G45 accomplishes a Z0 shift toward the workpiece. The coding used for the programs in Figures 7–5 and 7–6 is in the General Numerics format. The controller actually uses G43 and G44 for tool length offsets, but these codes conflict with EIA standards; therefore, G45 will be used in this text (G45 is normally an unassigned code). As has been pointed out, codes vary from controller to controller and machine to machine. The coding used in these programs is also similar to that used on Fanuc controllers. The only way to know which code to use is to check the programming manual for a particular machine. Always remember: *when in doubt, check the manual!*
G49—This is the tool length offset cancel code.
G81—As in Machinist Shop Language, G81 is the canned drill cycle. It functions in the same manner explained in the previous example.
G80—As used previously, G80 cancels an 80 series canned cycle.
R—This address stands for reference level. The *reference level* is the spot where the programmer desires the canned cycle to start feeding into the workpiece. The reference level is also called the *rapid* or *gage level.* The reference level is usually the same height as the buffer zone, but it may not be.
G98/G99—G98 is the return to initial level command. G99 is the return to rapid (reference) level command. When an 80 series canned cycle is active in word address format, the spindle may be directed to return to the rapid level with a G99. G99 is modal and will remain active until a G80 cancel code is issued, or until canceled with a G98. If a clamp is in the path of movement, or if the spindle is at the last location in a series, the spindle may be retracted to the initial starting point in the cycle by using a G98 command.
M03—M functions, as briefly explained in Chapter 6, control a number of auxiliary functions. M03 is the code for turning the spindle on in the clockwise direction. In Appendix 3 is a list of common M functions used on numerical control machinery.
M05—Turns the spindle off.
M06—Tool change code.
M08—Turns the coolant on.
M09—Turns the coolant off.
T—Selects the tool to be put in the spindle by the tool changer.
F—Assigns feedrates, as in two-axis programming.
S—Designates the spindle speed.

PROGRAM EXPLANATION

(Refer to Figures 7–5 and 7–6.)

The machine used for the programs in Figures 7–5 and 7–6 is a vertical machining center using automatic tool change such as that shown in Figure 7–7. Figure 7–5 is a word address format program for the inch-dimensioned part in Figure 7–1. Figure 7–6 is the metric version of the program.

N010

- N010—The sequence number.
- G70—Selects inch input.
- G90—Selects absolute positioning.
- G10 H01 Z4—Assigns a 4-inch value to tool register #1 (100 mm in the metric program). G10 tells the controller that tool information is to be assigned. H01 tells the controller to place the information in register #1.

N020

- N020—The sequence number.
- G10 H02 Z3—Assigns a 3-inch value to tool register #2 (in the metric program, a value of 75 mm is assigned).

N030

- N030—The sequence number.
- G10 H03 Z3.5—Assigns a 3.5-inch value to tool register #3 (in the metric program, a value of 87.5 mm is assigned).

```
X0/Y0 = CENTER OF PART
TOOL CHANGE = X-12 Y8
TOOLS LIST:
TOOL 1 = 3/8 COMB. DRILL
TOOL 2 = 1.000 4 FLUTE END MILL
TOOL 3 = .500 4 FLUTE END MILL
BUFFER HEIGHT:
.100 ABOVE PART SURFACE

N010 G70 G90 G10 H01 Z4
N020 G10 H02 Z3
N030 G10 H03 Z3.5
N040 G00 M06 T1                        REM:3/8 DRILL #1
N050 G45 H01 S1066 M03                 REM:SPNDL ON
N060 G81 G98 X-2.5 Y1.5 Z-.62 R0 F12.8 M08  REM:DRILL #1
N070 G99 X2.5                          REM:DRILL #2
N080 Y-1.5                             REM:DRILL #3
N090 X0 Y0                             REM:DRILL #4
N100 X-2.5  Y-1.5                      REM:DRILL #5
N110 G80 G49 Z0                        REM:CANCEL DRILL & OFFSET
N120 M06 T2                            REM:1.000 E/M
N130 X0 Y0 S425 M03                    REM:RAPID TO #6, SPNDL ON
N140 G45 H02 Z0
N150 G01 Z-.62  F6.8 M08               REM:FEED TO DEPTH
N160 Y.48                              REM:FEED TO #7
N170 X1.48                             REM:FEED TO #8
N180 Y-.48                             REM:FEED TO #9
N190 X-1.48                            REM:FEED TO #10
N200 Y.48                              REM:FEED TO #11
N210 X0                                REM:FEED TO #7
N220 G00 G49 Z0 M09                    REM:RETRACT SPNDL, COOL.
OFF
N230 M06 T3                            REM:.500 E/M
N240 G45 H03
N250 X0 Y0 S800 F12.8 M03              REM:RAPID TO #6, SPNDL ON
N260 Z0                                REM:RAPID TO BUFFER
N270 G01 Z-.62                         REM:FEED TO DEPTH
N280 Y.75                              REM:FEED TO #13
N290 X1.75                             REM:FEED TO #14
N300 Y-.75                             REM:FEED TO #15
N310 X-1.75                            REM:FEED TO #16
N320 Y.75                              REM:FEED TO #17
N330 X0                                REM:FEED TO #13
N340 Y.74                              REM:FEED TO #12
N350 G00 Z3                            REM:RETRACT SPNDL
N360 X-3.26 Y2.26                      REM:RAPID TO #18
N370 Z0                                REM:RAPID TO BUFFER
N380 G01 Z-.62                         REM:FEED TO DEPTH
N390 Y-2.26                            REM:FEED TO #19
N400 G00 Z3                            REM:RETRACT SPNDL
N410 X3.26                             REM:RAPID TO#20
N420 Z0                                REM:RAPID TO BUFFER
N430 G01 Z-.62                         REM:FEED TO DEPTH
N440 Y2.26                             REM:FEED TO #21
N450 G00 Z3                            REM:RETRACT  SPNDL
N460 X-3.26                            REM:RAPID TO #18
N470 Z0                                REM:RAPID TO BUFFER
```



```
N480 G01 Z-.62                         REM:FEED TO DEPTH
N490 X-3.25 Y2.25                      REM:FEED TO #22
N500 Y-2.25                            REM:FEED TO #23
N510 G00 Z3                            REM:RETRACT SPNDL
N520 X3.25                             REM:RAPID TO #24
N530 Z0                                REM:RAPID TO BUFFER
N540 G01 Z-.62                         REM:FEED TO DEPTH
N550 Y2.25                             REM:FEED TO #25
N560 G00 G49 Z0 M09                    REM:CANCEL OFFSET, COOL.
OFF
N570 X-12 Y8 M05                       REM:RAPID TO PARK, SPNDL
OFF
N580 M30
```

FIGURE 7-5

Word address format three-axis program, nonmetric, for the part in Figure 7–1

```
X0/Y0 = CENTER OF PART
TOOL CHANGE = X-300 Y200
TOOLS LIST:
TOOL 1 = 8mm COMB. DRILL
TOOL 2 = 25mm 4 FLUTE END MILL
TOOL 3 = 10mm 4 FLUTE END MILL
BUFFER HEIGHT:
.25mm ABOVE PART SURFACE

N010 G71 G90 G10 H01 Z100
N020 G10 H02 Z75
N030 G10 H03 Z87.5
N040 G00 M06 T1                        REM:8mm DRILL #1
N050 G45 H01 S1066 M03                 REM:SPNDL ON
N060 G81 G98 X-60 Y40 Z-13 R0 F216 M08 REM:DRILL #1
N070 G99 X60                           REM:DRILL #2
N080 Y-40                              REM:DRILL #3
N090 X0 Y0                             REM:DRILL #4
N100 X-60 Y-40                         REM:DRILL #5
N110 G80 G49 Z0                        REM:CANCEL DRILL, OFFSET
N120 M06 T2                            REM:25mm E/M
N130 X0 Y0 S425 M03                    REM:RAPID TO #6, SPNDL ON
N140 G45 H02 Z0
N150 G01 Z-13  F172.5 M08              REM:FEED TO DEPTH
N160 Y12.25                            REM:FEED TO #7
N170 X37.25                            REM:FEED TO #8
N180 Y-12.25                           REM:FEED TO #9
N190 X-37.25                           REM:FEED TO #10
N200 Y12.25                            REM:FEED TO #11
N210 X0                                REM:FEED TO #7
N220 G00 G49 Z0 M09                    REM:RETRACT SPNDL, COOL.
OFF
N230 M06 T3                            REM:.500 E/M
N240 G45 H03
N250 X0 Y0 S800 F325.1 M03             REM:RAPID TO #6, SPNDL ON
N260 Z0                                REM:RAPID TO BUFFER
```

```
N270 G01 Z-13               REM:FEED TO DEPTH
N280 Y20                    REM:FEED TO #13
N290 X45                    REM:FEED TO #14
N300 Y-20                   REM:FEED TO #15
N310 X-45                   REM:FEED TO #16
N320 Y20                    REM:FEED TO #17
N330 X0                     REM:FEED TO #13
N340 Y44.75                 REM:FEED TO #12
N350 G00 Z75                REM:RETRACT SPNDL
N360 X-80.25 Y55.25         REM:RAPID TO #18
N370 Z0                     REM:RAPID TO BUFFER
N380 G01 Z-13               REM:FEED TO DEPTH
N390 Y-55.25                REM:FEED TO #19
N400 G00 Z75                REM:RETRACT SPNDL
N410 X80.25                 REM:RAPID TO#20
N420 Z0                     REM:RAPID TO BUFFER
N430 G01 Z-13               REM:FEED TO DEPTH
N440 Y55.25                 REM:FEED TO #21
N450 G00 Z75                REM:RETRACT  SPNDL
N460 X-80.25                REM:RAPID TO #18
N470 Z0                     REM:RAPID TO BUFFER
N480 G01 Z-13               REM:FEED TO DEPTH
N490 X-80 Y55               REM:FEED TO #22
N500 Y-55                   REM:FEED TO #23
N510 G00 Z75                REM:RETRACT SPNDL
N520 X80                    REM:RAPID TO #24
N530 Z0                     REM:RAPID TO BUFFER
N540 G01 Z-13               REM:FEED TO DEPTH
N550 Y55                    REM:FEED TO #25
N560 G00 G49 Z0 M09         REM:CANCEL OFFSET, COOL.
OFF
N570 X-300 Y200 M05         REM:RAPID TO PARK, SPNDL
OFF
N580 M30
```

FIGURE 7-6
Word address format three-axis program, metric, for the part in Figure 7–2

N040

- N040—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- M06—Initiates a tool change.
- T1—Selects tool #1 to be loaded into the spindle by the tool changer.

N050

- N050—The sequence number.
- G45 H01—Calls up an offset for tool #1. Notice that the offset is not automatically assigned to the tool. With this system, any offset can be called up for any tool. (Note: It is wise to use the offset register number that corresponds to the tool number being used to avoid confusion.)
- S1066—Specifies that a spindle speed of 1066 RPM is to be used.
- M03—Turns the spindle on in clockwise rotation.

FIGURE 7-7
A light-duty vertical machining center *(Photo courtesy of Bridgeport Machines Division of Textron Inc.)*

N060

- N060—The sequence number.
- G81—Calls up the canned drilling cycle.
- G98—Instructs the MCU to return the spindle to its initial level at the Z-axis position when the 80 series G code was instituted. It is impor-

tant to know beforehand the position of the spindle relative to fixtures and clamps when using G98 to clear parts, fixtures, and clamps (as is done in this program). The tool change that was performed in block N050 left the spindle in its fully retracted position.

- X/Y/Z coordinates—Give the location of hole #1. With an 80 series G code, the machine positions to the X/Y coordinate before moving to the Z axis. The Z-axis coordinate is calculated to feed the drill through the part.
- R0—Sets the rapid level. The tool offset activated in block N050 shifted the Z0 point to the start of the buffer zone. The buffer zone height is also the desired rapid (or reference) level; zero as the Z-axis coordinate specifies this.
- F—Assigns the feedrate to be used for Z-axis feedrate moves with the active G81.
- M08—Turns on the coolant.

N070

- N070—The sequence number.
- G99—Instructs the machine to return the spindle to the rapid (reference) level (the start of the buffer zone).
- X coordinate—Gives the location of hole #2.

N080

- N080—The sequence number.
- Y coordinate—To move from hole #2 to hole #3.

N090

- N090—The sequence number.
- X0/Y0—To move from hole #3 to hole #4. Since the G81 is still active, a hole is drilled at this location.

N100

- N100—The sequence number.
- X/Y coordinates—To move from hole #4 to hole #5.

N110

- N110—The sequence number.
- G80—Cancels the G81.
- G49—Cancels the tool offset, thereby making Z0 the fully retracted position.
- Z0—Retracts the spindle.

N120

- N120—The sequence number.
- M06—Initiates an automatic tool change.
- T2—Specifies that tool #2 is to be used.

N130

- N130—The sequence number.
- X0/Y0—Positions the machine to location #6 in Figure 7–2. G00 is still active from the first block, so this is a rapid move.
- S425—Specifies a spindle speed of 425 RPM.
- M03—Turns the spindle on in clockwise rotation.

N140

- N140—The sequence number.
- G46 H02—Calls up tool offset #2, which is used here with tool #2.
- Z0—Rapids the spindle to the Z0 point when the tool offset is active, which is the start of the buffer zone.

N150

- N150—The sequence number.
- G01—Puts the machine in feedrate mode.
- Z coordinate—Feeds the end mill to proper milling depth.
- F—Assigns a feedrate to be used with feedrate moves, in this case, the Z-axis movement to milling depth.
- M08—Turns on the coolant.

N160

- N160—The sequence number.
- Y coordinate—To feed from #6 to #7.

N170

- N170—The sequence number.
- X coordinate—To feed from #7 to #8.

N180

- N180—The sequence number.
- Y coordinate—To feed from #8 to #9.

N190

- N190—The sequence number.
- X coordinate—To feed from #9 to #10.

N200

- N200—The sequence number.
- Y coordinate—To feed from #10 to #11.

N210

- N210—The sequence number.
- X coordinate—To feed from #11 to #7.

N220

- N220—The sequence number.
- G00—Puts the machine in rapid traverse mode.

- G49—Cancels the tool length offset.
- Z0—Rapids the spindle to the fully retracted position.
- M09—Turns off the coolant.

N230
- N230—The sequence number.
- M06—Initiates an automatic tool change.
- T3—Specifies that tool #3 is to be used.

N240
- N240—The sequence number.
- G46 H03—Calls up tool offset #3, which in this case is to be used with tool #3.

N250
- N250—The sequence number.
- X/Y coordinates—Position the machine to location #6. G00 is active and the move is therefore at rapid traverse.
- S800—Sets the spindle speed to 800 RPM.
- F—Assigns a feedrate to be used with feedrate moves.
- M03—Turns the spindle on in clockwise rotation.

N260
- N260—The sequence number.
- Z0—Rapids the spindle to the start of the buffer zone.

N270
- N270—The sequence number.
- G01—Puts the machine in feedrate mode.
- Z coordinate—Feeds the end mill to depth.

N280
- N280—The sequence number.
- Y coordinate—To feed from #6 to #13.

N290
- N290—The sequence number.
- X coordinate—To feed from #13 to #14.

N300
- N300—The sequence number.
- Y coordinate—To feed from #14 to #15.

N310
- N310—The sequence number.
- X coordinate—To feed from #15 to #16.

N320
- N320—The sequence number.
- Y coordinate—To feed from #16 to #17.

N330
- N330—The sequence number.
- X coordinate—To feed from #17 to #13.

N340
- N340—The sequence number.
- Y coordinate—To feed from #13 to #12. This move is used to pull the tool away from the finished machined part surface. This will prevent tool marks from being left on the part when the spindle is retracted.

N350
- N350—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- Z coordinate—Retracts the spindle a sufficient amount to clear the part and clamps (the tool offset is still active in this case).

N360
- N360—The sequence number.
- X/Y coordinates—To move from #12 to #18.

N370
- N370—The sequence number.
- Z0—Rapids the spindle to the start of the buffer zone.

N380
- N380—The sequence number.
- G01—Puts the machine in feedrate mode.
- Z coordinate—To feed the end mill to depth.

N390
- N390—The sequence number.
- Y coordinate—To feed from #18 to #19.

N400
- N400—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- Z coordinate—Raises the spindle to clear the clamps that will be encountered on the next move.

N410
- N410—The sequence number.
- X coordinate—To move from #19 to #20. The move is in rapid traverse mode.

N420
- N420—The sequence number.
- Z0—Rapids the spindle to the start of the buffer zone.

N430
- N430—The sequence number.
- G01—Puts the machine in feedrate mode.
- Z coordinate—Feeds the end mill to depth.

N440
- N440—The sequence number.
- Y coordinate—To feed from #20 to #21.

N450
- N450—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- Z coordinate—Raises the spindle to clear clamps.

N460
- N460—The sequence number.
- X coordinate—To move from #21 to #18. The move is in rapid traverse mode.

N470
- N470—The sequence number.
- Z0—Rapids the spindle to the start of the buffer zone.

N480
- N480—The sequence number.
- G01—Puts the machine in feedrate mode.
- Z coordinate—Feeds the spindle to depth.

N490
- N490—The sequence number.
- X/Y coordinates—Feed the tool from #18 to #22. This feedrate move positions the tool for the finish milling cut.

N500
- N500—The sequence number.
- Y coordinate—To feed from #22 to #23.

N510
- N510—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- Z coordinate—Raises the spindle to clear clamps.

N520
- N520—The sequence number.
- X coordinate—To move from #23 to #24.

N530
- N530—The sequence number.
- Z0—Rapids the spindle to the start of the buffer height.

N540
- N540—The sequence number.
- G01—Puts the machine in feedrate mode.
- Z coordinate—To feed the tool to depth.

N550
- N550—The sequence number.
- Y coordinate—To feed from #24 to #25.

N560
- N560—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- G49—Cancels the tool length offset.
- Z0—Retracts the spindle.
- M09—Turns off the coolant.

N570
- N570—The sequence number.
- X/Y coordinates—The location of the park position. This is a location, safely out of the way, at which to position the machine at the end of the program. In this case it is assumed that the park position is roughly the same place where tool changes occur. Upon a rerunning of the program, block N040 would initiate a tool change to select tool #1. The park position is then used to protect the operator and part during loading and unloading from the fixture.
- M05—Turns off the spindle.

N580
- N580—The sequence number.
- M30—Signals the end of the program. It also resets the computer memory to the start of the program.

OTHER G CODES USED IN CNC PROGRAMMING

In the examples presented in this chapter, the basic drill cycle G81 was used. There are a number of other G codes that can be used in CNC programs. A list of these is contained in Appendix 3. Some of the more common codes are explained below. These codes are diagrammed in Figures 7–8, 7–9, and 7–10.

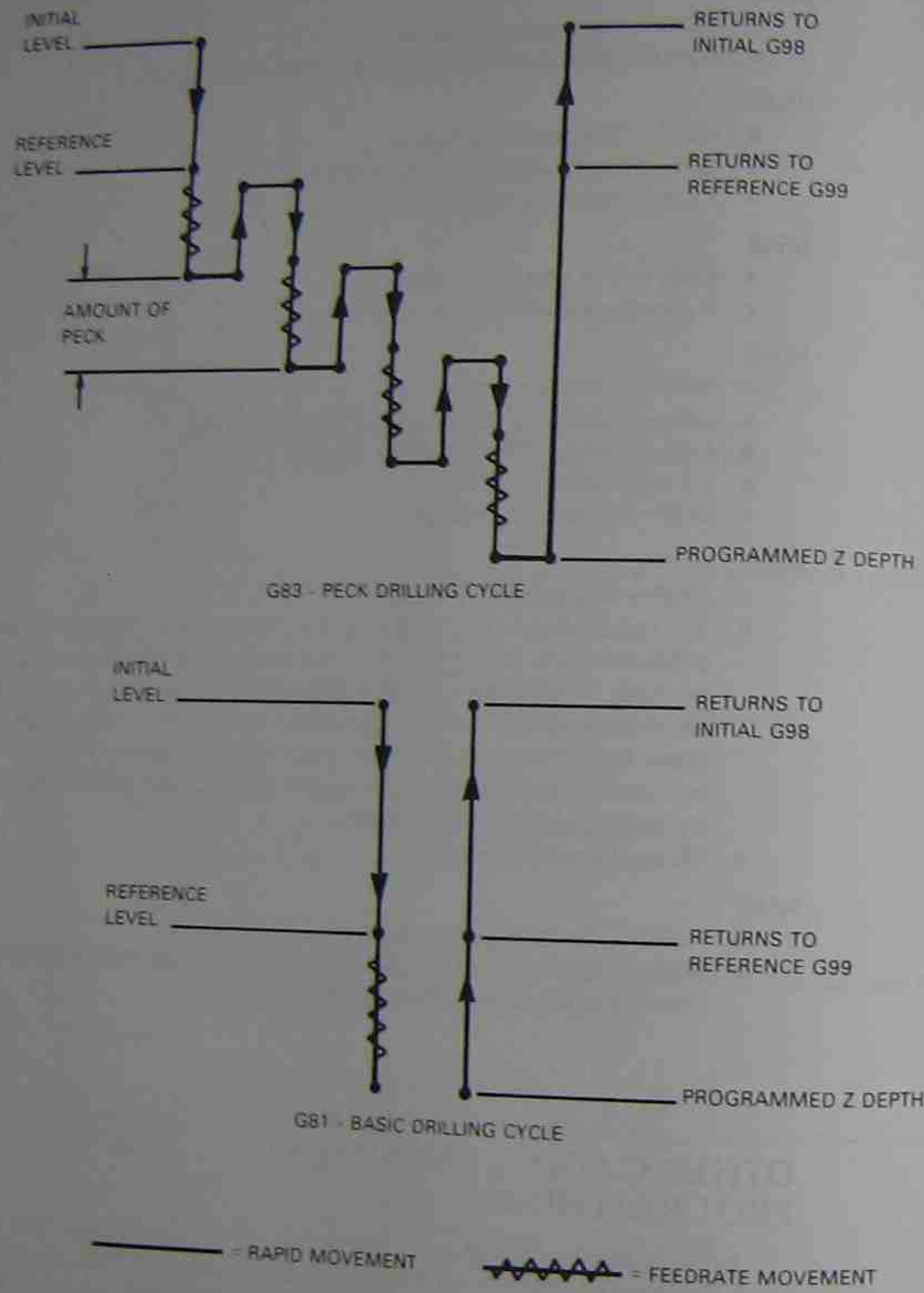


FIGURE 7–8
G codes for peck drilling and basic drilling cycles

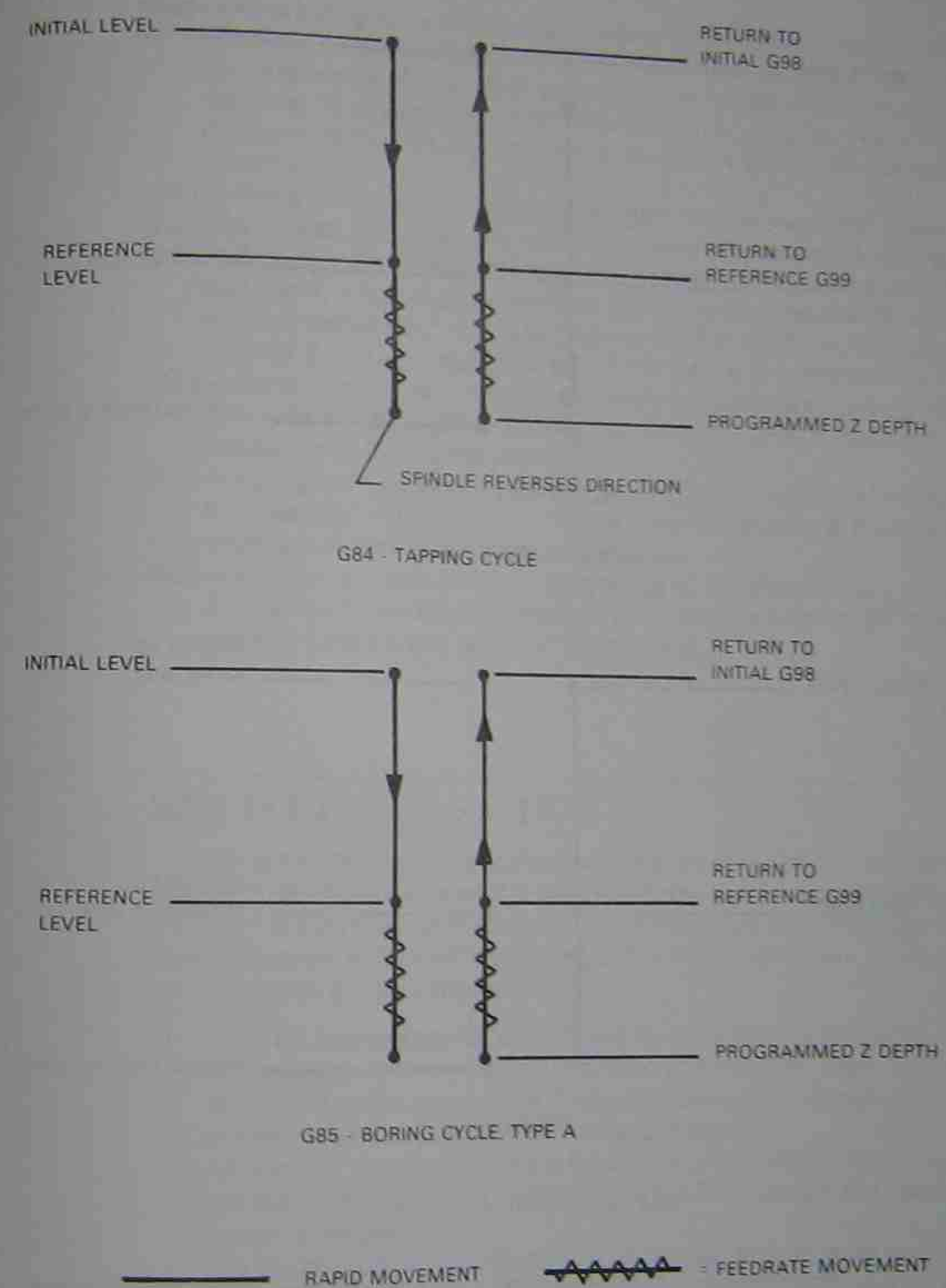


FIGURE 7–9
G codes for tapping and boring cycles

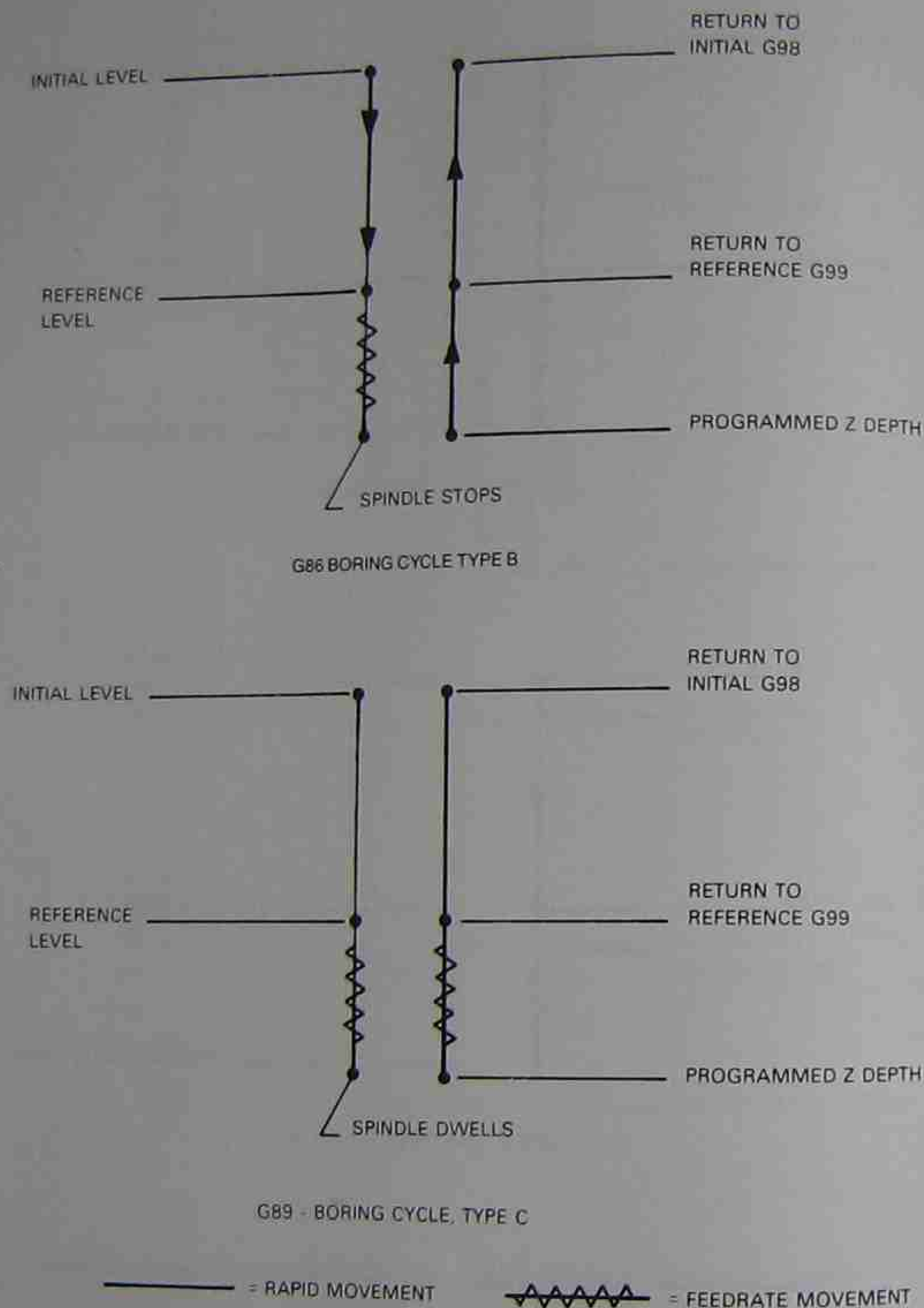


FIGURE 7–10
G codes for boring cycles

G83—Peck drilling cycle. The spindle rapids to reference level, feeds in .050, rapids out, feeds in .050 additional, and rapids out. This cycle repeats until the programmed Z-axis depth is reached. The spindle then rapids out to either the reference or initial level, depending on the program instructions. On some controllers, the amount of the peck is programmable using G codes.

G81—Basic drilling cycle. The spindle rapids to reference level, feeds to Z-axis depth, and rapids out, returning to either reference or initial level.

G84—Tapping cycle. The spindle rapids to the reference level and feeds to Z-axis depth; it then reverses direction, feeds to the reference level, and reverses direction again. If G98 is programmed, the spindle rapids to the initial level; if G99 is programmed, the spindle returns to reference level.

G85—Boring cycle, type A. The spindle rapids to the reference level, feeds to Z-axis depth, and feeds to the reference level. If G98 is programmed, the spindle rapids to the initial level.

G86—Boring cycle, type B. The spindle rapids to the reference level and feeds to Z-axis depth; it then stops and rapids to the reference level. If G98 is programmed, the spindle rapids to the initial level.

G89—Boring cycle, type C. The spindle rapids to the reference level, feeds to Z-axis depth, dwells, and feeds to the reference level. If G98 is programmed, the spindle rapids to the initial level.

SPECIAL FANUC SECTION

Because of the growing popularity of Fanuc controls, this section has been included to aid those with access to CNC machinery with Fanuc controllers. Figure 7–11 is a program to drill and mill the part in Figure 7–1. It is written for a Bayer Industries ACROLOC vertical mill with a Fanuc 6M control. There are a few commands that should be noted.

- O0701 is the program "O" number that identifies this program to the MCU.
- N.. sequence numbers start at N1 and increment by one. There is no hard-fast rule for sequence numbers. The control does not need them. Each programmer chooses their own numbering convention.
- An absolute zero shift is used in lines N2 and N3. G28X0.Y0.Z0. returns the spindle to home zero. G92X10.625Y7.5Z6. transfers the coordinate system to the part.
- The machine utilizes spindle direct tool change. An M06 is not necessary to change tools. It is used where it is the tool number (T01 for tool 1, T02 for tool 2, etc.).
- G44 is the command used for calling up the offsets.

```
%
O0701
(----------------------------------------)
(COORDINATE SYSTEM ORIGIN)
(X0 - CENTERLINE OF PART)
(Y0 - CENTERLINE OF PART)
(Z0 - .100 ABOVE TOP OF PART)
(----------------------------------------)
(TOOL LIST)
(TOOL 1 - 3/8 STUB DRILL)
(TOOL 2 - 1.0 END MILL 4-FLT. CENTER CUTTING)
(TOOL 3 - 1/2 END MILL 4-FLT.)
()
(----------------------------------------)
(ABSOLUTE ZERO SHIFT TO PART SYSTEM)
N1G80G90G70
N2G28X0Y0Z0
N3G92X10.625Y7.5Z6.
()
(----------------------------------------)
(3/8 STUBB DRILL - DRILL HOLES)
N4G00X-2.5Y1.5T01
N5S1066M03
N6G44Z0.H01M08
N7G81G99X-2.5Y1.5Z-.62R0.F12.8
N8X2.5
N9Y-1.5
N10X0.Y0.
N11X-2.5Y-1.5
N12G80G00Z0.
N13G49
()
(----------------------------------------)
(1.0 DIA.4-FLT. END MILL)
(ROUGH MILL INSIDE SLOT)
N14G0090X0.Y0.T02
N15S425M03
N16G44Z0.H02M08
(FEED TO DEPTH)
N17G01Z-.62F6.8
(ROUGH MILL INSIDE)
N18Y.48
N19X1.48
N20Y-.48
N21X-1.48
N22Y.48
N23X0.
N24G80G00Z0.M09
N25G49
()
(----------------------------------------)
(1/2 DIA. 4-FLT. END MILL)
(FINISH INSIDE SLOT)
```



```
(ROUGH/FINISH OUTSIDE)
N26G00G90X0.Y0.T03
N27S800M03
N28G44Z0.H03M08
(FEED TO DEPTH)
N29G01Z-.62F12.8
(FINISH MILL INSIDE SLOT)
N30Y.75
N31X1.75
N32Y-.75
N33X-1.75
N34Y.75
N35X0.
(PULL AWAY FROM PART AND RETRACT SPINDLE)
N36Y.74
N37G00Z3.
(POSITION TO START OF OUTSIDE MILL CUT)
N38X-3.26Y2.26
N39Z0.
(FEED TO DEPTH AND ROUGH MILL 1ST SIDE)
N40G01Z-.62F12.8
N41Y-2.26
(RETRACT SPINDLE AND JUMP OVER CLAMP)
(POSITION FOR ROUGH CUT ON 2ND SIDE)
N42G00Z3.
N43X3.26
N44Z0.
(FEED TO DEPTH AND ROUGH MILL 2ND SIDE)
N45G01Z-.62F12.8
N46Y2.26
(RETRACT SPINDLE AND JUMP OVER CLAMP)
(POSITION FOR FINISH CUT ON 1ST SIDE)
N47G00Z3.
N48X-3.26Y2.26
N49Z0.
(FEED TO DEPTH - MOVE TO PART SURFACE)
(AND FINISH MILL 1ST SIDE)
N50G01Z-.62F12.8
N51X-3.25Y2.25
N52Y-2.25
(RETRACT SPINDLE AND JUMP OVER CLAMP)
(POSITION FOR FINISH CUT ON 2ND SIDE)
N53G00Z3.
N54X3.25
N55Z0.
(FEED TO DEPTH AND FINISH MILL 2ND SIDE)
N56G01Z-.62F12.8
N57Y2.25
(HOME SPINDLE - CANCEL TOOL OFFSETS AND END PROGRAM)
N58G80G00Z0
N59G49
N60X0Y6.T15M09
N61M30
%
```

FIGURE 7-11

- G80G00Z0. is the command sequence necessary to retract the spindle to the Z home position prior to issuing a G49 offset cancel.
- In block 60 the spindle is sent to X0 Y6 to allow easy access to the part. The tool carousel is also indexed to the last station. This is done so the tools only index one station at the start of the cycle, rather than all the way around.

SUMMARY

The important concepts presented in this chapter are:

- Three-axis hole operations in Machinist Shop Language are accomplished through the use of G codes.
- Tool lengths in three-axis machines must be preset by the operator or set in the program; in Machinist Shop Language, this is accomplished by using the TOOL command. The format for assigning offsets is TOOL 10##, where the first two digits (10) tell the MCU that tool information is to be assigned, and ## is the number of the tool being programmed.
- Tool lengths in word address format are programmed by using G10 H##, where ## is the tool register number.
- A buffer zone is the distance between the top of the part and the feed engagement point of the tool. The amount of buffer is determined by the programmer and is built into the tool lengths at setup.
- Feedrates and buffer zones for use with canned cycles are set by using V codes in Machinist Shop Language.
- On word address CNC machinery, the initial level is the Z-axis spindle position when an 80 series G code commences. A reference (or rapid) level is the Z-axis feedrate engagement point selected by the programmer. G98 selects a return to initial level, and G99 selects a return to reference level when using 80 series G codes.

REVIEW QUESTIONS

1. How are tool lengths called up in Machinist Shop Language? How are they canceled? How are they defined?
2. What is a buffer zone?
3. How are buffer zones set for canned cycles in Machinist Shop Language? For milling?

4. How are canned cycle feedrates set in word address format? In Machinist Shop Language?
5. How are tool lengths entered into a word address CNC? How are they called up? How are they canceled?
6. What is a modal command? What is a nonmodal command?
7. How are straight line feedrate moves initiated in word address?
8. What is meant by the terms initial level and reference level?
9. What command is used for initial level? For reference level?
10. How is absolute positioning specified on word address CNC machinery? How is incremental positioning specified?
11. Write a program to mill and drill the part in Figure 7–12:
    a. Using Machinist Shop Language.
    b. Using a word address CNC mill.

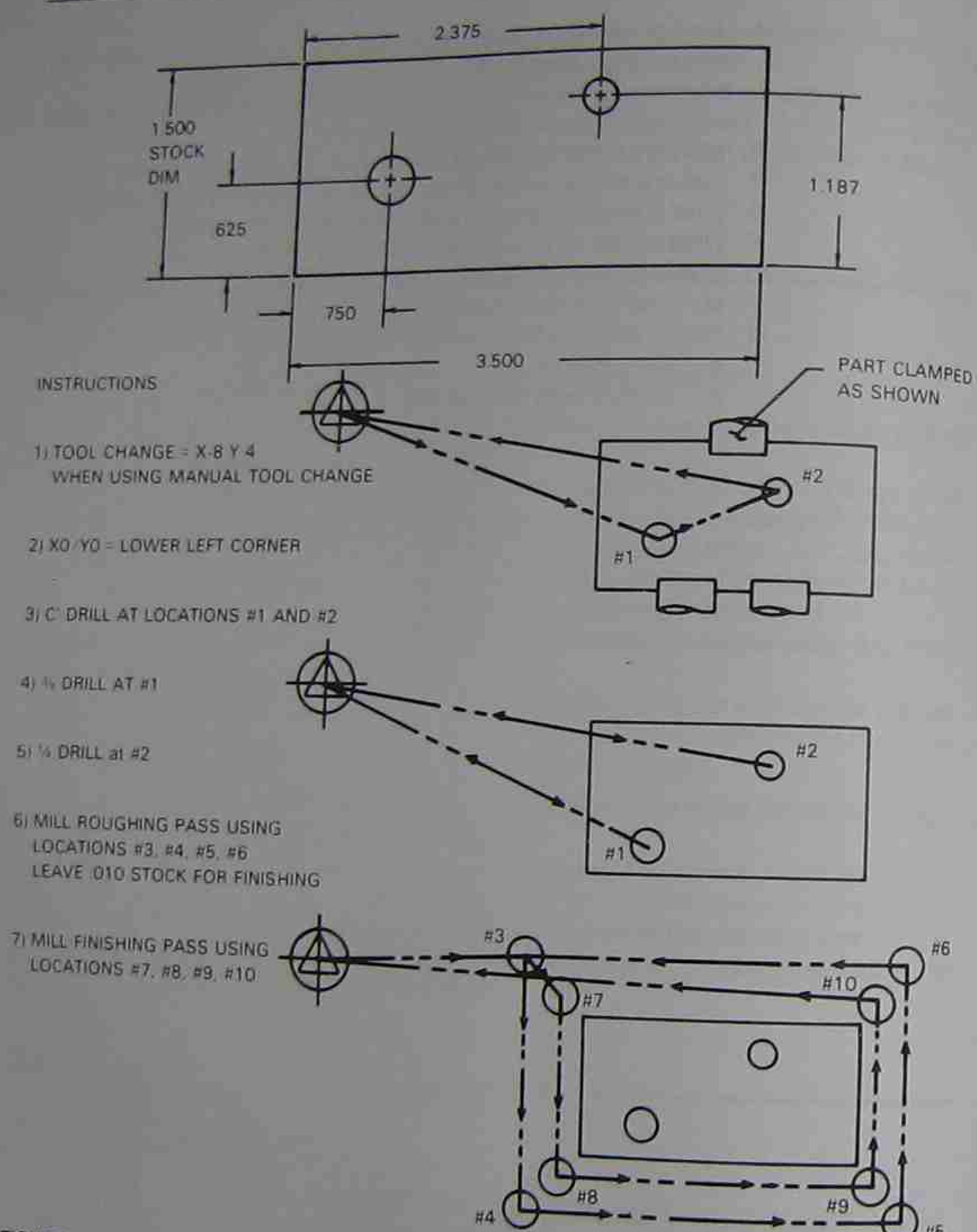


FIGURE 7–12
Part drawing for review question #11

CHAPTER 8

Math for Numerical Control Programming

OBJECTIVE Upon completion of this chapter, you will be able to:

- Use right-angle trigonometry to determine programming coordinates from part drawings.

In the following chapters, the machining of arcs and angles will be discussed. For students already possessing a good working knowledge of trigonometry, this chapter will serve as a review. It is included here for students who have either not taken a course in shop math or who feel a review is in order.

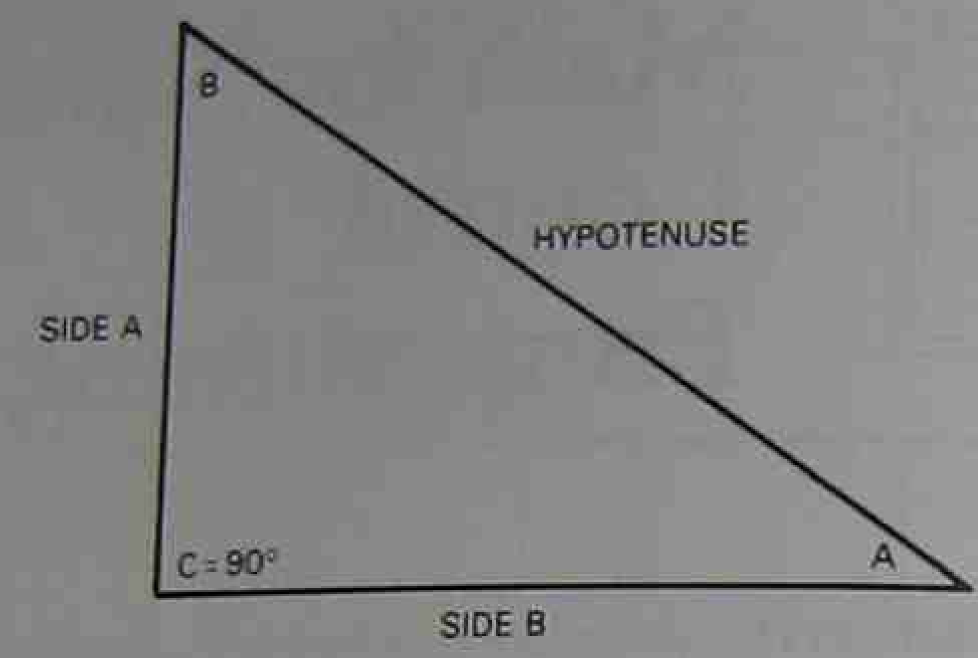
What the machinist is able to do by blending arcs and angles through skill and feel for the craft, the NC part programmer must put into numeric coordinates. It is necessary for the programmer to become proficient at trigonometry to accomplish this task. Trigonometry has applications not only in NC programming but also in other types of machining situations. It is easily mastered with a little practical experience.

BASIC TRIGONOMETRY

Trigonometry is the mathematical science dealing with the solution of triangles. For example, knowing one side plus one other angle or side of a right triangle, all other information concerning the triangle can be derived using trigonometry. For machine shop use, the types of triangles usually dealt with are right triangles (see Figure 8–1). Note that one of the angles in the triangle is 90 degrees. A 90-degree angle is called a right angle: hence the name right triangle. The following formulas are also given in Figure 8–1:

$$\text{SINE} = \frac{\text{OPPOSITE SIDE}}{\text{HYPOTENUSE}}$$

$$\text{COSINE} = \frac{\text{SIDE ADJACENT}}{\text{HYPOTENUSE}}$$



$$\text{SINE} = \frac{\text{OPPOSITE SIDE}}{\text{HYPOTENUSE}} \qquad \text{COSECANT} = \frac{\text{HYPOTENUSE}}{\text{SIDE OPPOSITE}}$$

$$\text{COSINE} = \frac{\text{SIDE ADJACENT}}{\text{HYPOTENUSE}} \qquad \text{SECANT} = \frac{\text{HYPOTENUSE}}{\text{SIDE ADJACENT}}$$

$$\text{TANGENT} = \frac{\text{SIDE OPPOSITE}}{\text{SIDE ADJACENT}} \qquad \text{COTANGENT} = \frac{\text{SIDE ADJACENT}}{\text{SIDE OPPOSITE}}$$

FIGURE 8–1
Right triangle

$$\text{TANGENT} = \frac{\text{SIDE OPPOSITE}}{\text{SIDE ADJACENT}}$$

The other formulas are the inverses of these three. In the machine shop these three will cover most situations.

Figure 8–2 will help to demonstrate the value of triangles in shop mathematics. If the part in Figure 8–2 is to be drilled without using a rotary table, it will be necessary to specify coordinates as dimensioned in Figure 8–2. These are known as *jig borer coordinates*, because they are a common way of locating hole patterns on jig borers. They are also commonly used with milling machines. They are especially important in CNC programming.

The immediate problem in looking at Figure 8–2 is that only the dimensions for holes #1 and #4 are known (they are located on the radius of the bolt circle). However, dimensions a and b can be determined by using trigonometry.

Note that a triangle has been constructed in the first quadrant of the part. If this triangle is solved for the length of its sides, it will supply the information

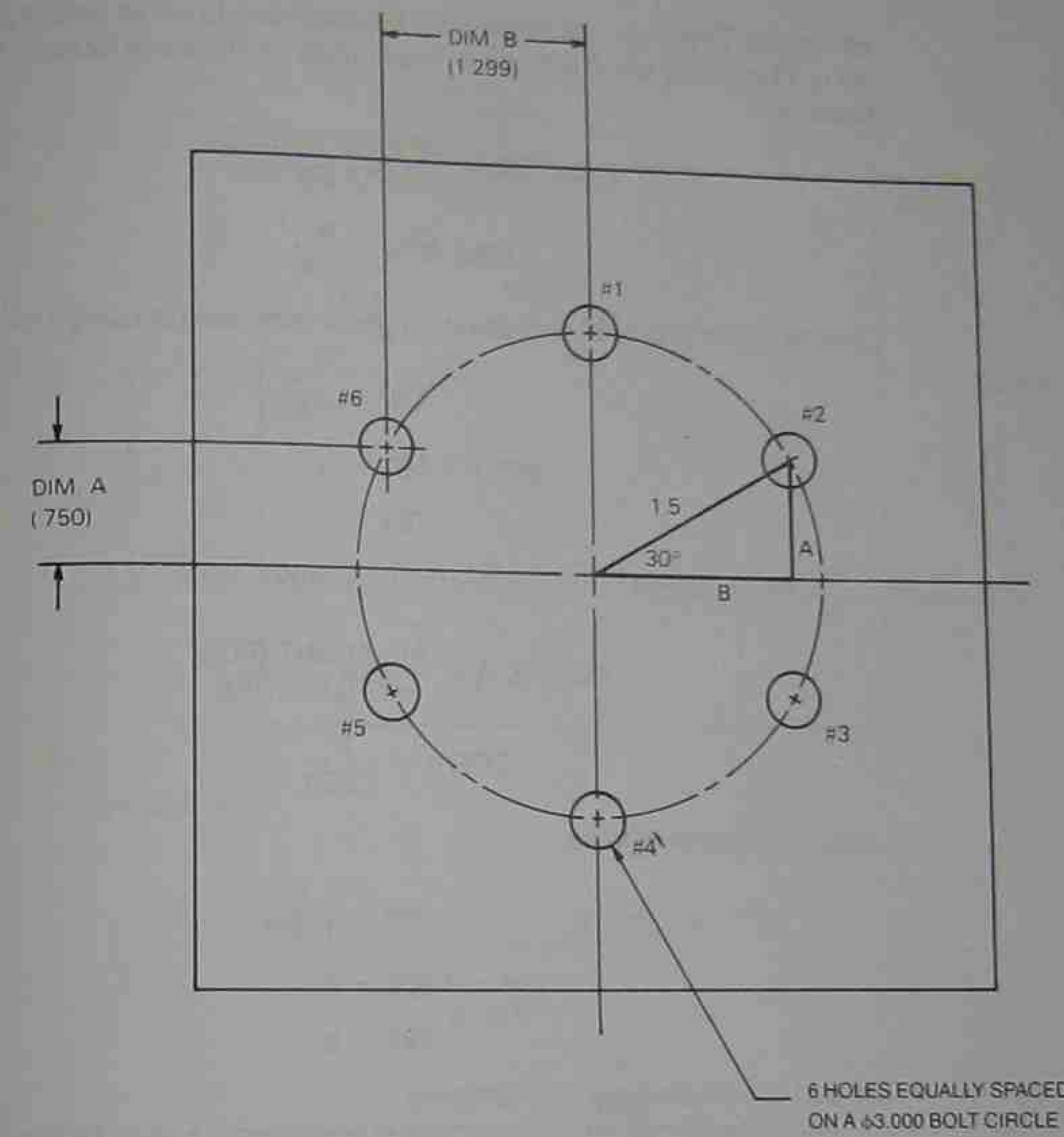


FIGURE 8–2

needed for the missing coordinates. The sides of this triangle are labeled a, b, and 1.5 (which is the hypotenuse of the triangle).

What is known about this triangle? The length of the hypotenuse is half the diameter of the bolt circle, or 1.500 inches. Angle A is also known; since the bolt circle (of 360 degrees) is divided into six equal spaces, the angle between each hole is 60 degrees. Half the distance to each hole lies on each side of the centerline. Therefore, the angle from the centerline to either hole #2 or hole #3 is 30 degrees, which is angle A. The formula for the sine is found to be most practical.

$$\text{SINE A} = \frac{\text{OPPOSITE SIDE}}{\text{HYPOTENUSE}}$$

$$\text{SINE } 30 = \frac{a}{1.500}$$

Looking up the sine of 30 degrees in a trigonometric table or using a calculator:

$$.500 = \frac{a}{1.500}$$

$$.500 \times 1.500 = a$$

$$.750 = a$$

To solve for side b, the formula for the cosine is used:

$$\text{COSINE A} = \frac{\text{ADJACENT SIDE}}{\text{HYPOTENUSE}}$$

$$\text{COS A} = \frac{b}{1.500}$$

Using a calculator or table:

$$.866 = \frac{b}{1.500}$$

$$.866 \times 1.500 = b$$

$$1.299 = b$$

The part coordinates are now complete.

Another application of trigonometry is presented in Figure 8–3, where the value of the X dimension is needed. By constructing a triangle as shown in the figure, the length of side b can be determined. If this length is subtracted from the overall length of 3.000 inches, the X dimension is obtained. Two things are known about this triangle: the length of side a and angle A. By using the formula for the tangent, the triangle can be solved as follows:

$$\text{TAN } 40 = \frac{\text{OPPOSITE SIDE}}{\text{ADJACENT SIDE}}$$

$$\text{TAN } 40 = \frac{1.000}{b}$$

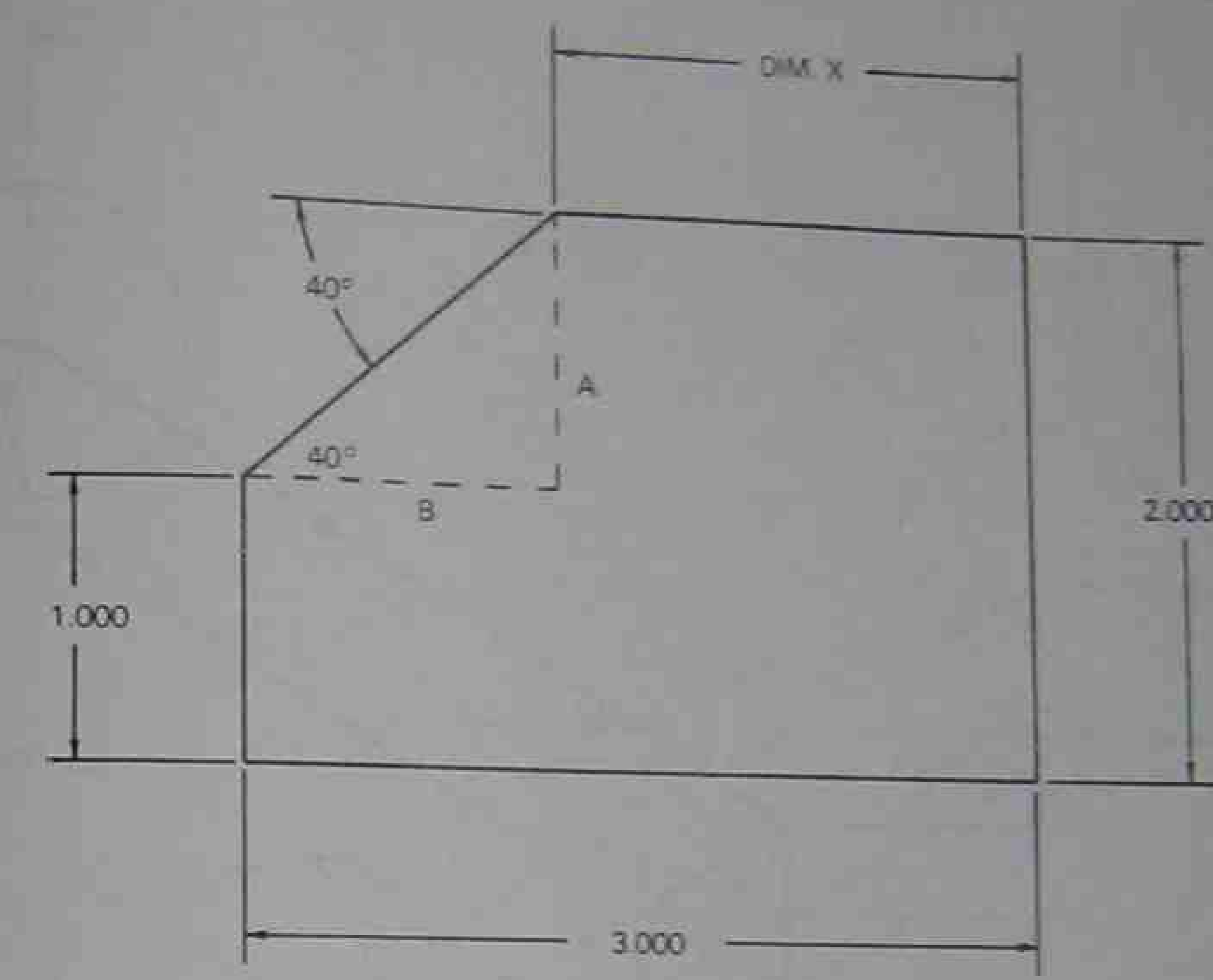


FIGURE 8–3

$$.839 = \frac{1.000}{b}$$

$$.839 \times b = 1.000$$

$$b = \frac{1.000}{.839}$$

$$b = 1.191$$

Dimension X equals 3.000 − 1.191, or 1.809.

USING TRIGONOMETRY FOR CUTTER OFFSETS

A common use for trigonometry in NC programming is calculating cutter offsets for use with linear or circular interpolation (discussed further in Chapter 9). Assume the angle in Figure 8–3 is to be milled. The coordinates of the cutter will have to be determined mathematically because, as Figure 8–4 illus-

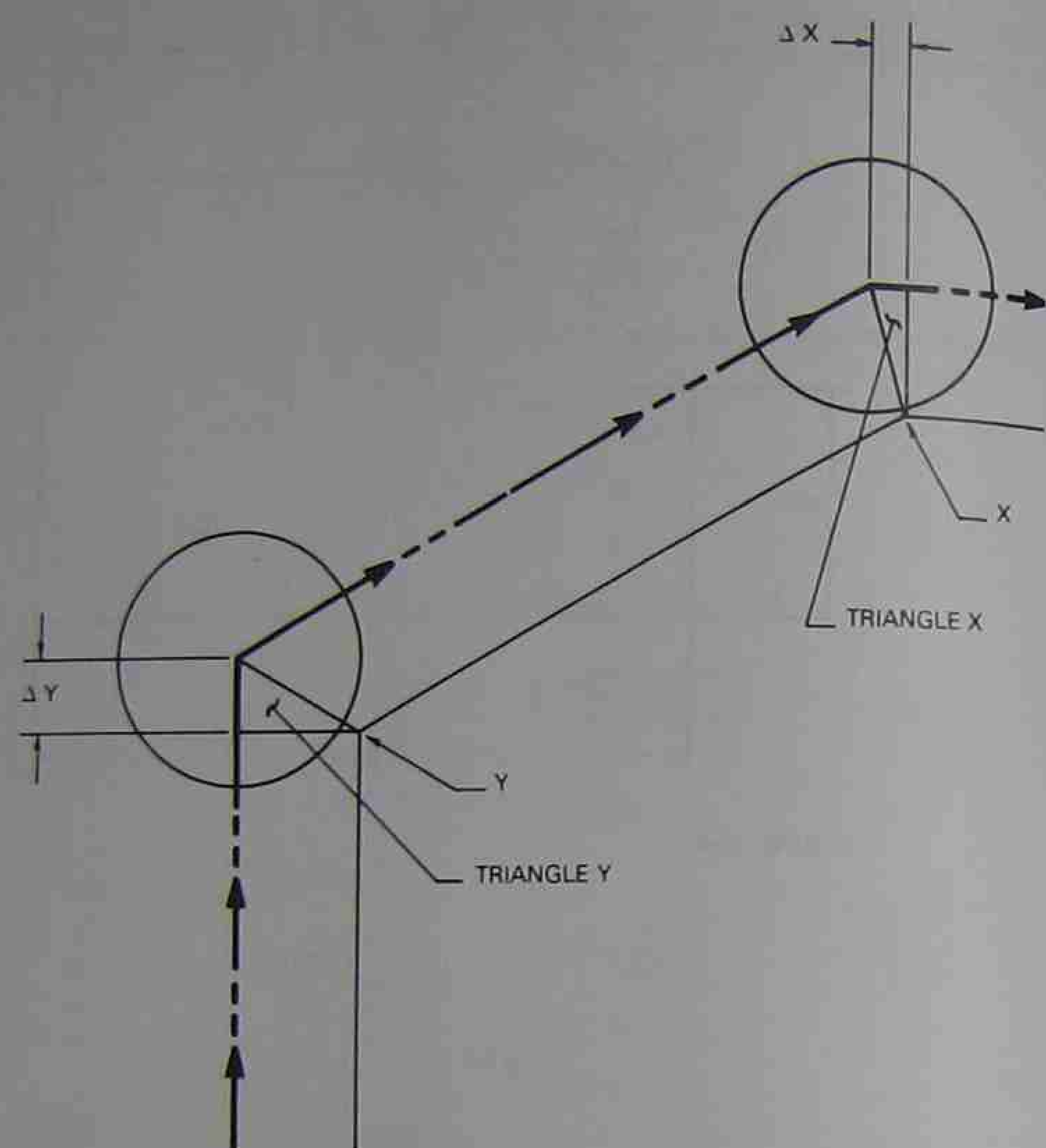


FIGURE 8–4

trates, the cutter cannot be positioned at point Y but must be positioned some unknown distance away. Similarly, the cutter cannot be moved to point X, but some unknown distance short of point X. By solving triangles Y and X, the proper coordinates can be determined.

The angles shown in Figure 8–5 can be found with little effort by looking at the angles formed around points Y and X. When determining angles by this method, three rules must be remembered:

1. The total number of degrees in a circle is 360.
2. The sum of the angles in a triangle is 180 degrees.
3. The complement of an angle is 90 minus the angle.

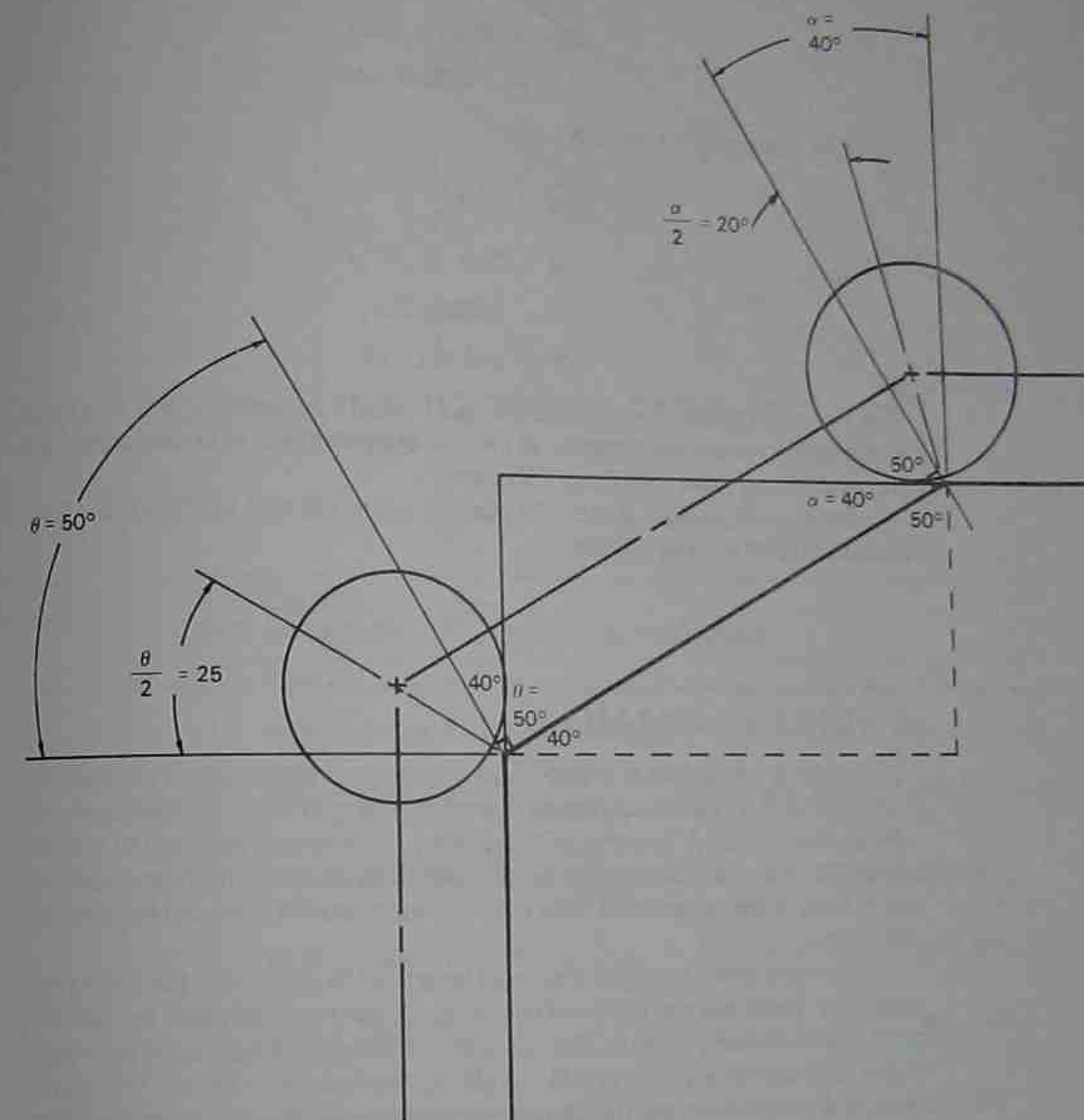


FIGURE 8–5

The angles that will be used for the calculation are 25 degrees for triangle Y and 20 degrees for triangle X, as shown in the figure.

Solving triangle Y for $\triangle Y$:

$$\frac{\triangle Y}{.250} = \text{TAN } 25$$

$$\triangle Y = \text{TAN } 25\,(.250)$$
$$\triangle Y = .46631\,(.250)$$
$$\triangle Y = .11658 \text{ or } .117$$

Solving triangle X for $\triangle X$:

$$\frac{\triangle X}{.250} = \text{TAN } 20$$
$$\triangle X = \text{TAN } 20\,(.250)$$
$$\triangle X = .36397(.250)$$
$$\triangle X = .09099 \text{ or } .091$$

The amount of offset can be added to or subtracted from points Y and X to arrive at the correct cutter coordinates. In the next chapter this sort of calculation will be performed for use with linear interpolation.

Refer to Figure 8–6 for right triangle solutions and to Figure 8–7 for oblique-angled triangle solutions.

A MILLING EXAMPLE

Figure 8–8 shows a goblet-shaped casting. A ledge .250 inch deep by 1.000 radius is to be milled in three places, blending to the .120 thick cast web with a .125 radius. A ¼ inch (.250 diameter) end mill can be used to mill the ledge. Six cutter locations must be calculated. Since three of these locations are mirrors of the other three, only the three locations P1, P2, and P3 need be calculated.

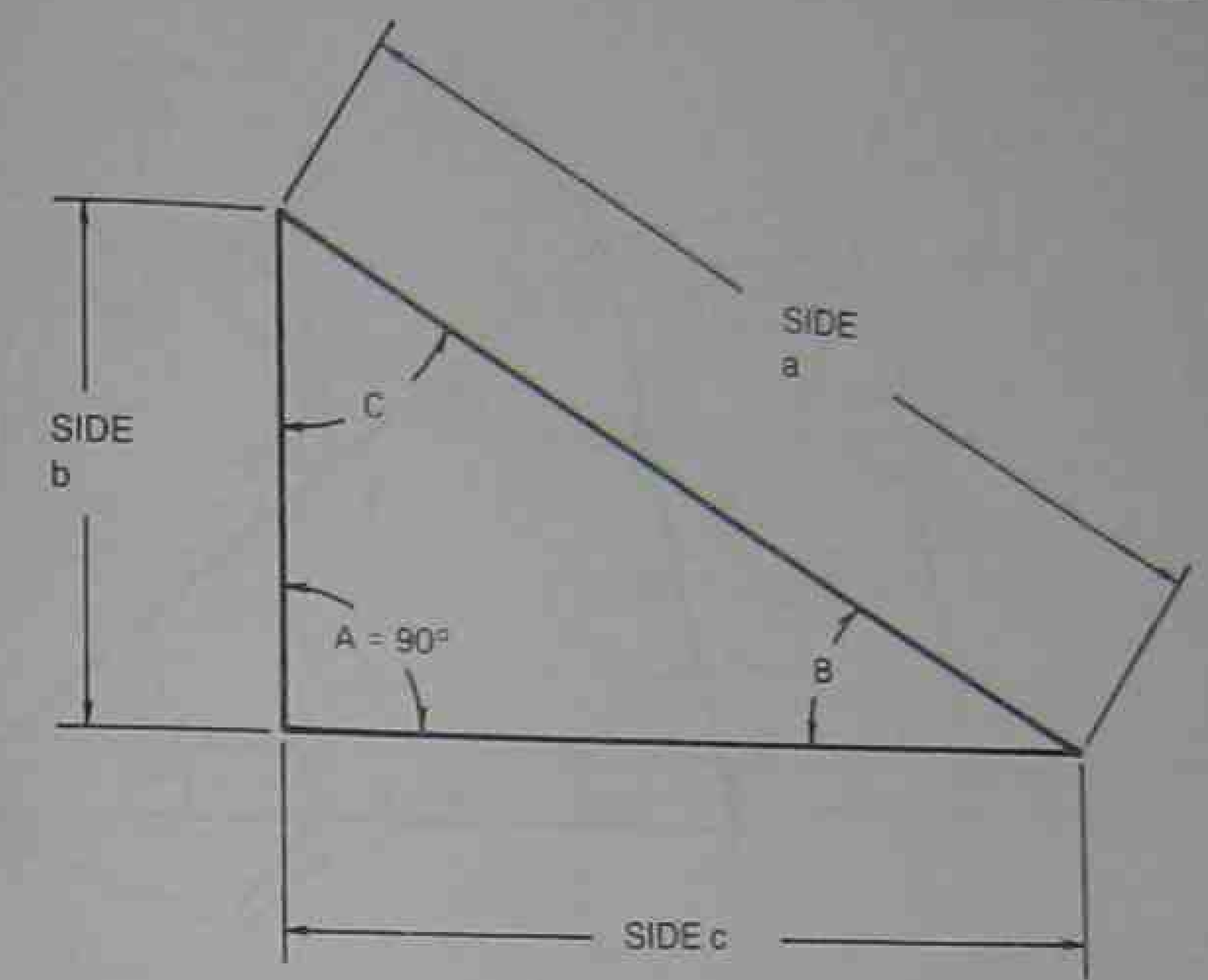
There are three triangles that must be solved to determine the proper coordinates. There are two pieces of information known that will allow the solution of P1, and determine angle θ: The radius from the center of the part to the center of the .250 diameter cutter (1.0 R − .125 R) and the .185 (.06 + .125) leg of triangle A. Angle θ can also be determined from this same information as shown in the figure. To determine angle θ, the SIN function is used:

$$\text{SIN } \theta = \frac{.185}{.875} \text{ or } .21143$$

$$\theta = 12.206 \text{ degrees}$$

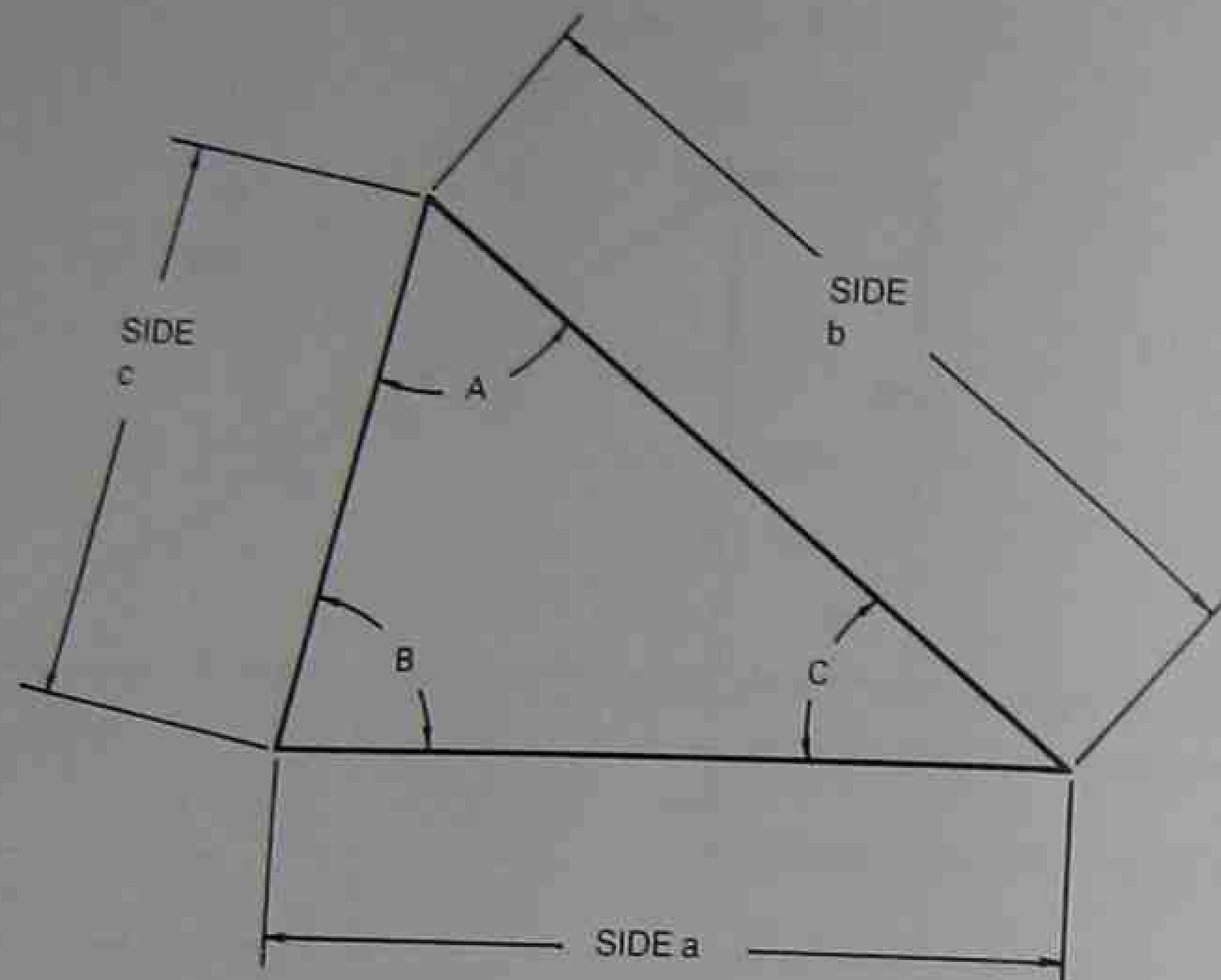
Once angle θ is known, angle α and angle β can be determined:

$$\alpha = 30^\circ - \theta \quad \text{and} \quad \beta = 60^\circ - \theta$$



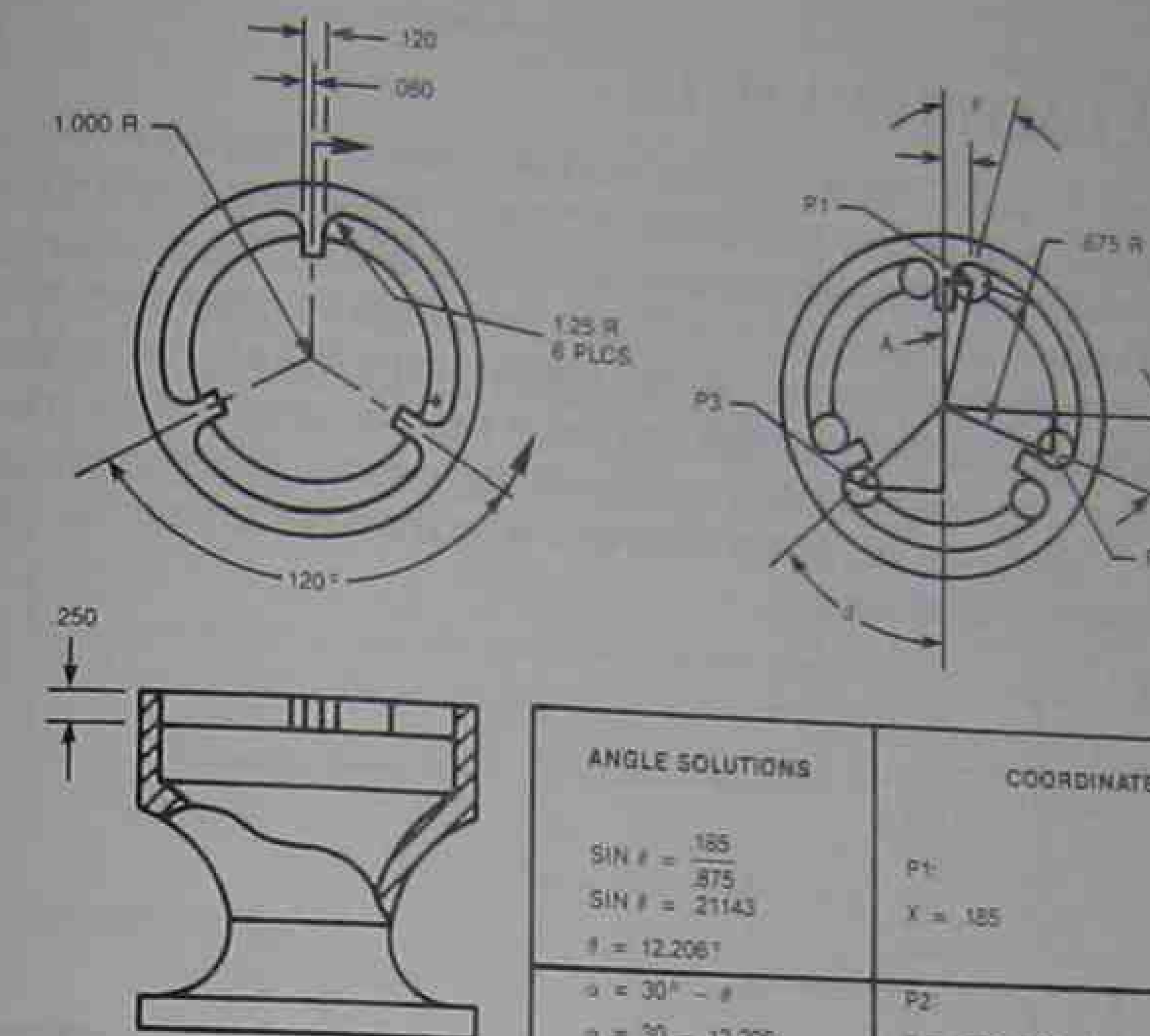
KNOWN VARIABLES	SOLUTION FORMULAS		
SIDE a, ANGLE B	$b = a \times \text{SIN B}$	$c = a \times \text{COS B}$	$C = 90° - B$
SIDE a, ANGLE C	$b = a \times \text{COS C}$	$c = a \times \text{SIN C}$	$B = 90° - C$
SIDE b, ANGLE B	$a = \frac{b}{\text{SIN B}}$	$c = b \times \text{COT B}$	$C = 90° - B$
SIDE b, ANGLE C	$a = \frac{b}{\text{COS C}}$	$c = b \times \text{TAN C}$	$B = 90° - C$
SIDE c, ANGLE B	$a = \frac{c}{\text{COS B}}$	$b = c \times \text{TAN B}$	$C = 90° - B$
SIDE c, ANGLE C	$a = \frac{c}{\text{SIN C}}$	$b = c \times \text{COT C}$	$B = 90° - C$
SIDES a AND b	$c = \sqrt{a^2 - b^2}$	$\text{SIN B} = \frac{b}{a}$	$C = 90° - B$
SIDES a AND c	$b = \sqrt{a^2 - c^2}$	$\text{SIN C} = \frac{c}{a}$	$B = 90° - C$
SIDES b AND c	$a = \sqrt{b^2 + c^2}$	$\text{TAN B} = \frac{b}{c}$	$C = 90° - B$

FIGURE 8–6
Solutions of right triangles



ONE SIDE AND TWO ANGLES KNOWN: GIVEN: SIDE a, OPPOSITE ANGLE A, AND OTHER ANGLE B		
$C = 180^\circ - (A + B)$	$b = \dfrac{a \times \text{SIN } B}{\text{SIN } A}$	$c = \dfrac{a \times \text{SIN } C}{\text{SIN } A}$
TWO SIDES AND THE ANGLE BETWEEN THEM KNOWN: GIVEN: SIDES a, b, AND ANGLE C		
$\text{TAN } A = \dfrac{a \times \text{SIN } C}{b - (a \times \text{COS } C)}$	$B = 180^\circ - (A + C)$	$c = \dfrac{a \times \text{SIN } C}{\text{SIN } A}$
$c = \sqrt{a^2 + b^2 - (2ab \times \text{COS } C)}$		
TWO SIDES AND ANGLE OPPOSITE ONE SIDE KNOWN: GIVEN: SIDE a, OPPOSITE ANGLE A, AND SIDE B		
$\text{SIN } B = \dfrac{b \times \text{SIN } A}{a}$	$C = 180^\circ - (A + B)$	$c = \dfrac{a \times \text{SIN } C}{\text{SIN } A}$
ALL THREE SIDES KNOWN:		
$\text{COS } A = \dfrac{b^2 + c^2 - a^2}{2bc}$	$\text{SIN } B = \dfrac{b \times \text{SIN } A}{a}$	$C = 180^\circ - (A + B)$

FIGURE 8–7
Solutions of oblique-angled triangles



ANGLE SOLUTIONS	COORDINATE SOLUTIONS	
$\text{SIN } \theta = \dfrac{.185}{.875}$ $\text{SIN } \theta = .21143$ $\theta = 12.206^\circ$	P1: $X = .185$	$Y = \sqrt{.875^2 - .185^2}$ $Y = .85522$
$\alpha = 30^\circ - \theta$ $\alpha = 30 - 12.206$ $\alpha = 17.794^\circ$	P2: $X = .875(\text{COS } \alpha)$ $X = .83314$	$Y = -[.875(\text{SIN } \alpha)]$ $Y = -.2674$
$\beta = 60^\circ - \theta$ $\beta = 60 - 12.206$ $\beta = 47.794^\circ$	P3: $X = -[.875(\text{COS } \beta)]$ $X = -.5878$	$Y = -[.875(\text{SIN } \beta)]$ $Y = -.6481$

FIGURE 8–8

The Pythagorean formula is used to determine the Y coordinate of P1:

$X = .185$ (known information)

$Y = \sqrt{.875^2 - .185^2}$ or .85522

The coordinates of P2 are calculated by the SIN and COS of angle α.

P2:

$X = .875(\text{COS } \alpha)$ OR .83314

$Y = -[.875(\text{SIN } \alpha)]$ or $-.2674$

The coordinates of P3 are calculated by the SIN and COS of angle β.

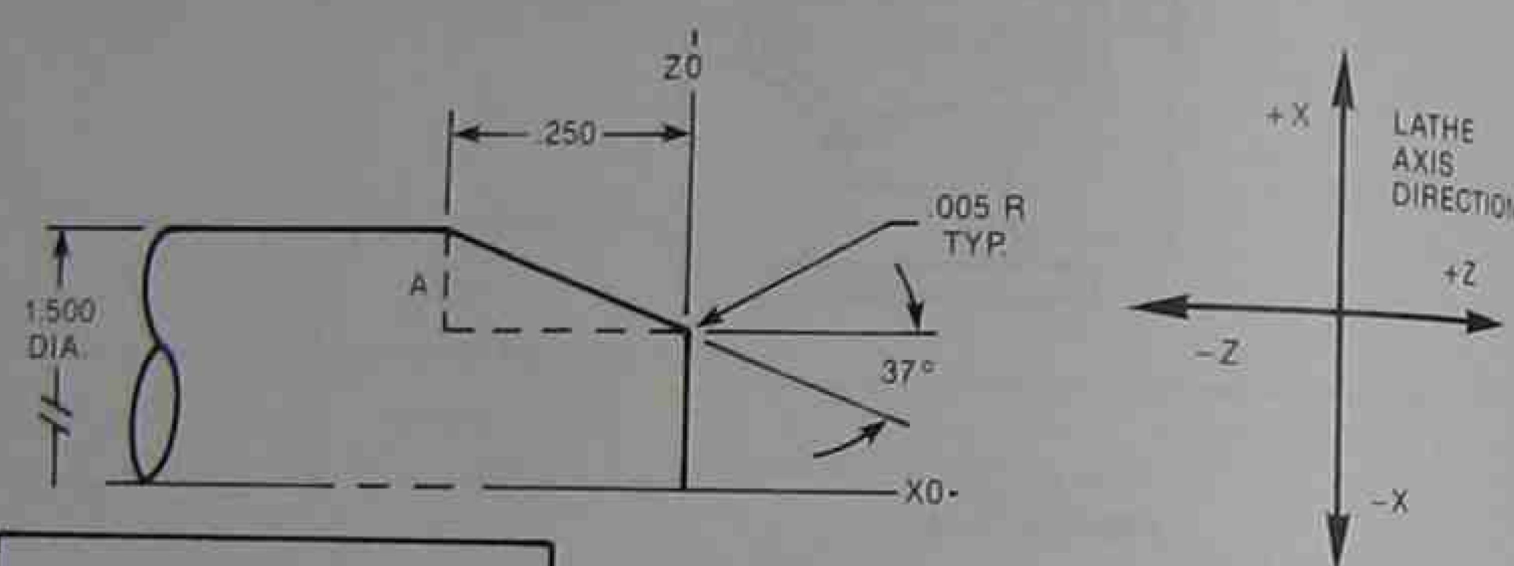
P3:

$X = -[.875(\text{COS } \beta)]$ or $-.5878$

$Y = -[.875(\text{SIN } \beta)]$ or $-.6481$

A LATHE EXAMPLE

A typical lathe programming situation is depicted in Figures 8–9 and 8–10. A 37 degree angle is to be turned, intersecting with a 1.500 diameter. At the intersection points of the angle and part face and the intersection of the angle and the 1.500 diameter, a .005-inch radius is to be generated. This small radius serves to deburr the part. A .015 radius turning tool will be used to turn the part. Most programmers will deburr a part in this manner even if not specifically called for on the blueprint. Most companies have some type of engineering standard that applies in situations such as this and will determine the maximum edge break that is allowed.



SOLUTIONS OF PRELIMINARY INFORMATION
A = .250 (TAN 37°) A = .18839
ΔK = .005 (TAN 18.5°) ΔK = .00167
ΔI = .005 (TAN 26.5°) ΔI = .00249

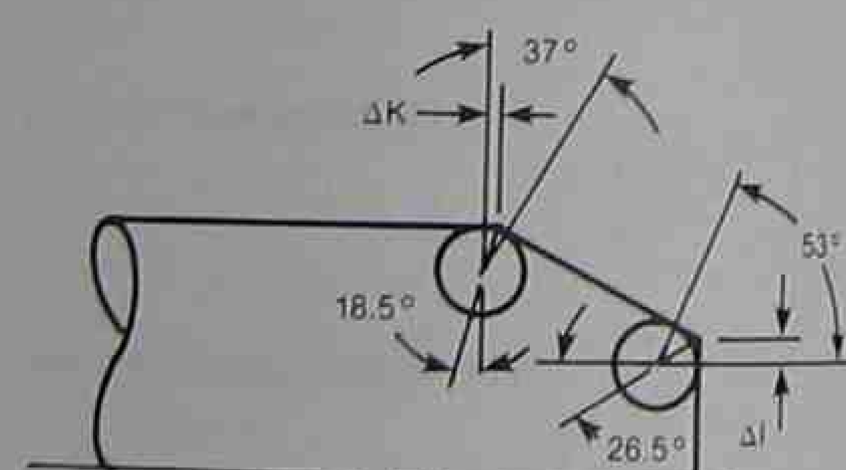


FIGURE 8–9

There are four cutter locations that must be calculated as shown in Figure 8–10: P1, P2, P3, and P4. To determine the cutter locations it is necessary to use a twofold approach. First, A, I, and K must be determined. Second, the coordinates of the four cutter locations are determined using A, △I, and △K.

To determine the preliminary information:

COORDINATE SOLUTIONS	
P1: X = .750 − A − ΔI X = .55912	Z = −.005 + .020 Z = .015
P2: X = .55912 + .020 (COS 37°) X = .55912 + .01597 X = .57509	Z = −.005 + .02 (SIN 37°) Z = −.005 + .01204 Z = .00704
P3: X = .745 + .020 X = .765	Z = −(.250 + ΔK) Z = −.25167
P4: X = .745 + .020 (COS 37°) X = .745 + .01597 X = .76097	Z = −(.25167 − .020 (SIN 37°)) Z = (.25167 − .01204) Z = −.23963

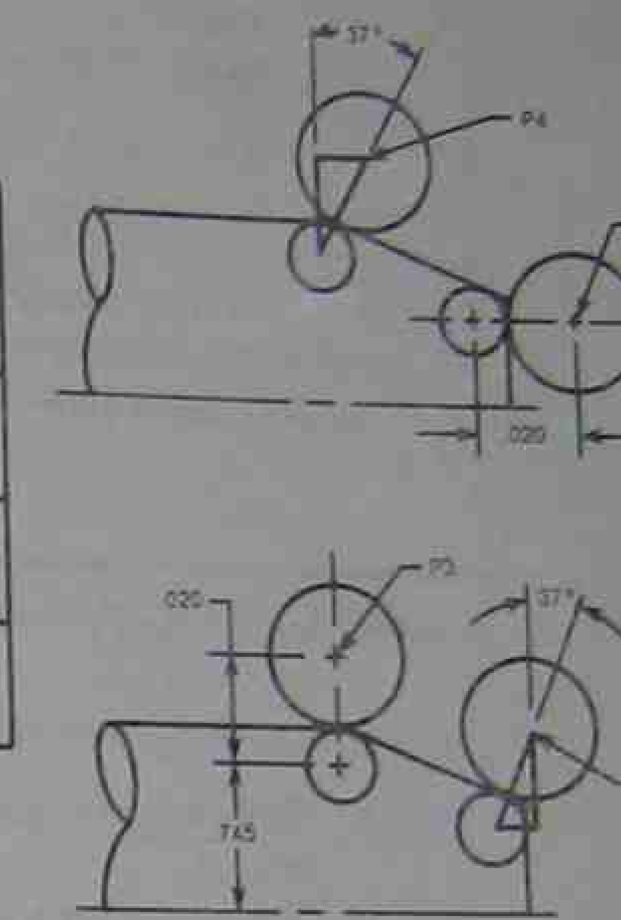


FIGURE 8–10

A = .250(TAN 37) or .18839

△K = .005(TAN 18.5) or .00167

△I = .005(TAN 26.5) or .00249

To determine the cutter locations:

P1:

X = .750 − A − △I   Z = −.005 + .020

X = .55912   Z = .015

P2:

Note: .55912 in X formula from P1 X calculation.

X = .55912 + .020(COS 37°)   Z = −.005 + .02(SIN 37°)

X = .57509   Z = .00704

P3:

X = .745 + .020   Z = −(.250 + △K)

X = .765   Z = −.25167

P4:
Note: .25167 in Z formula from P3 Z calculation.

X = .745 + .020(COS 37°)   Z = −[.25167 − .020(SIN 37°)]

X = .76097   Z = −.23963

It is often necessary when determining coordinates to systematically solve a series of triangles in order to derive the necessary values for the final cutter solution, as this case demonstrates.

SUMMARY

The important concepts presented in this chapter are:

- Right-angle trigonometry is the mathematical science of solving right triangles.
- The sine of an angle equals the side opposite the angle divided by the hypotenuse of the triangle.
- The cosine of an angle equals the side adjacent to the angle divided by the hypotenuse of the triangle.
- The tangent of an angle equals the side opposite the angle divided by the side adjacent to the angle.
- The use of trigonometry is necessary for determining cutter offsets for linear and circular interpolation and for determining other part information from a blueprint.

REVIEW QUESTIONS

1. What is the sine of an angle? The cosine? The tangent?
2. What are the sine, cosine, and tangent of the triangle in Figure 8–11? What is angle B?
3. What are the coordinates of the holes in the part in Figure 8–12, assuming that the center is X0/Y0?
4. What are the coordinates of the four cutter locations indicated in Figure 8–13?

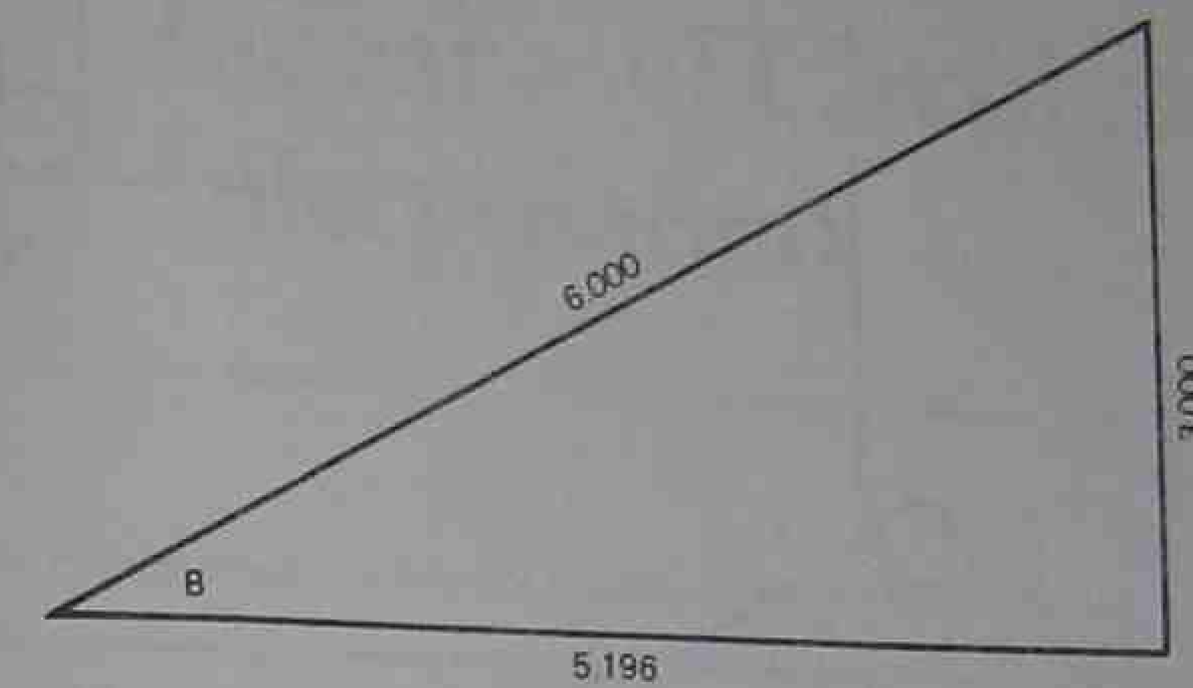


FIGURE 8–11
Triangle for review question #2

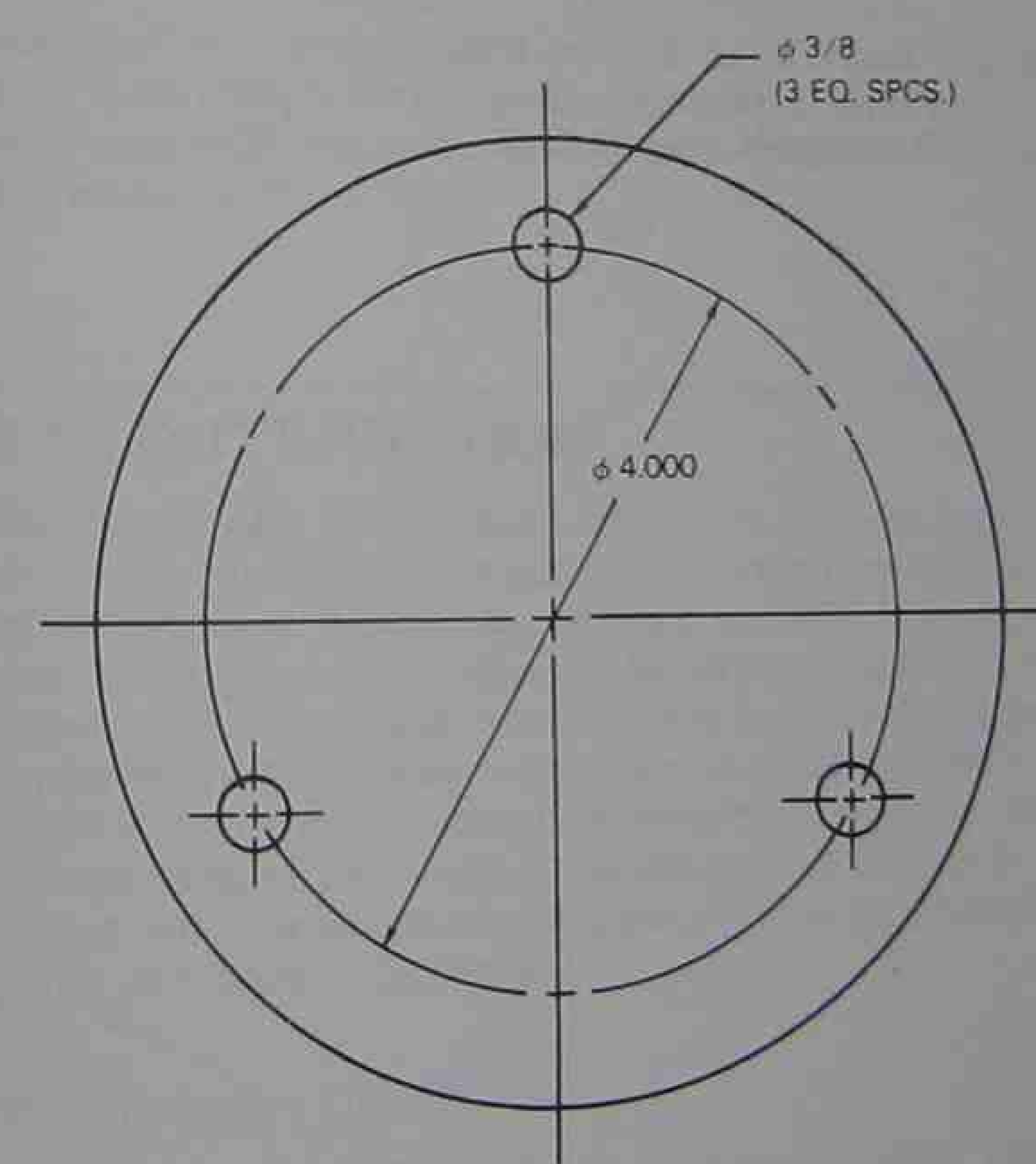


FIGURE 8–12
Part drawing for review question #3

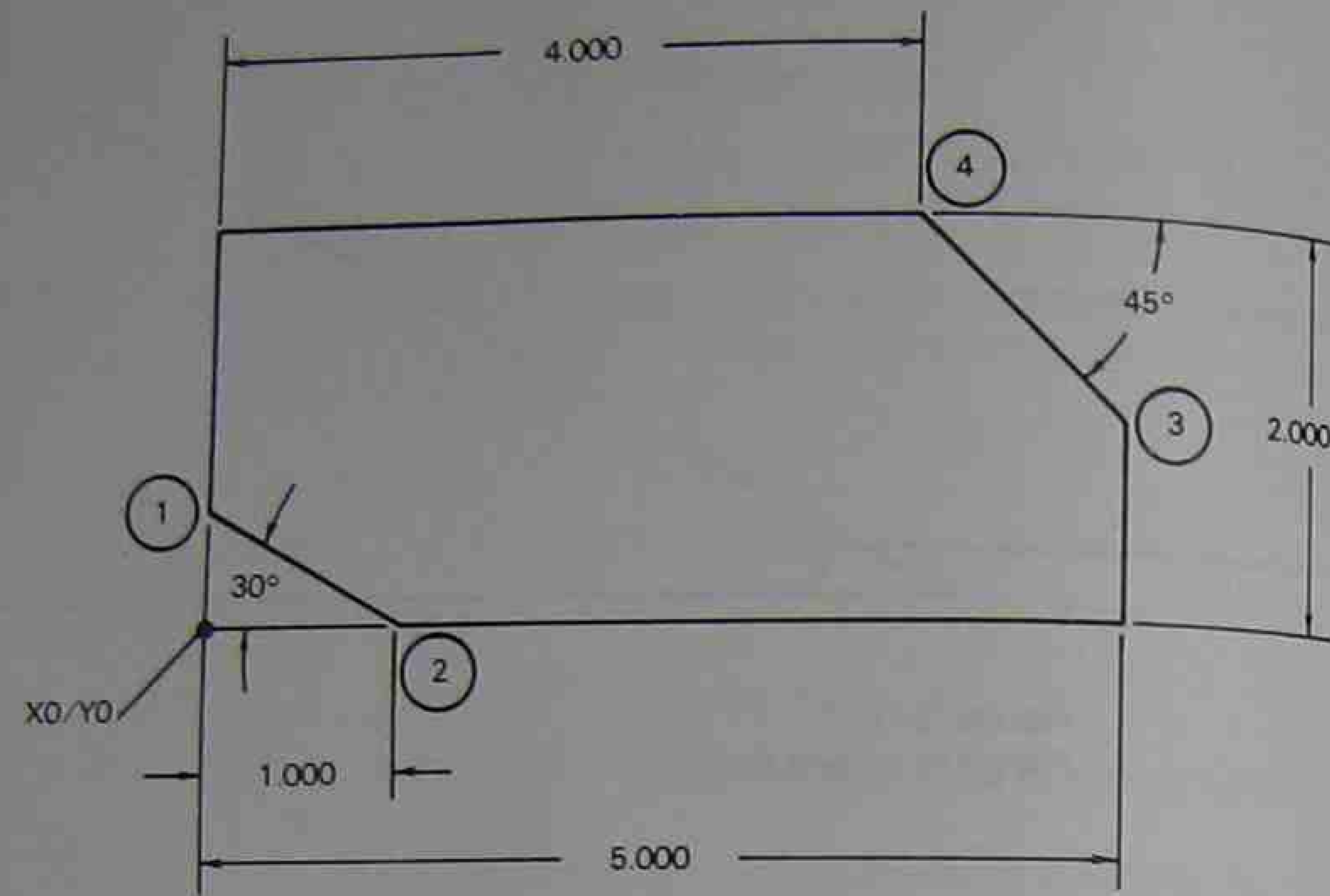


FIGURE 8–13
Part drawing for review question #4

CHAPTER 9

Linear and Circular Interpolation

OBJECTIVES Upon completion of this chapter, you will be able to:

- Write programs using linear interpolation to cut simple angles in both word address format and Machinist Shop Language.
- Write programs using circular interpolation to mill arcs in both word address format and Machinist Shop Language.

Simply put, *linear interpolation* is the ability to cut angles, and *circular interpolation* is the ability to cut arcs or arc segments. Without the ability to cut arcs and angles, a CNC machine is quite limited in its uses. In this chapter both concepts will be introduced.

LINEAR INTERPOLATION

Machines capable of linear interpolation have a continuous-path control system, meaning that the drive motors on the various axes can operate at varying rates of speed. When cutting an angle, the MCU calculates the angle based on the programmed coordinates. Since the MCU knows the current spindle location, it can calculate the difference in the X coordinate between the current position and the programmed location. The change in the Y coordinate divided by the change in the X coordinate yields the slope of the cutter centerline path. The computer then simply moves the drive motors in this ratio. Linear interpolation can be accomplished using the X/Y axes (X/Y plane), Z/X axes (Z/X plane), or Y/Z axes (Y/Z plane).

Calculating Cutter Offsets

Figure 9–1 shows a part on which an angle is to be milled. The cutter has already been positioned at location #1, Figure 9–2. A .500-inch-diameter end

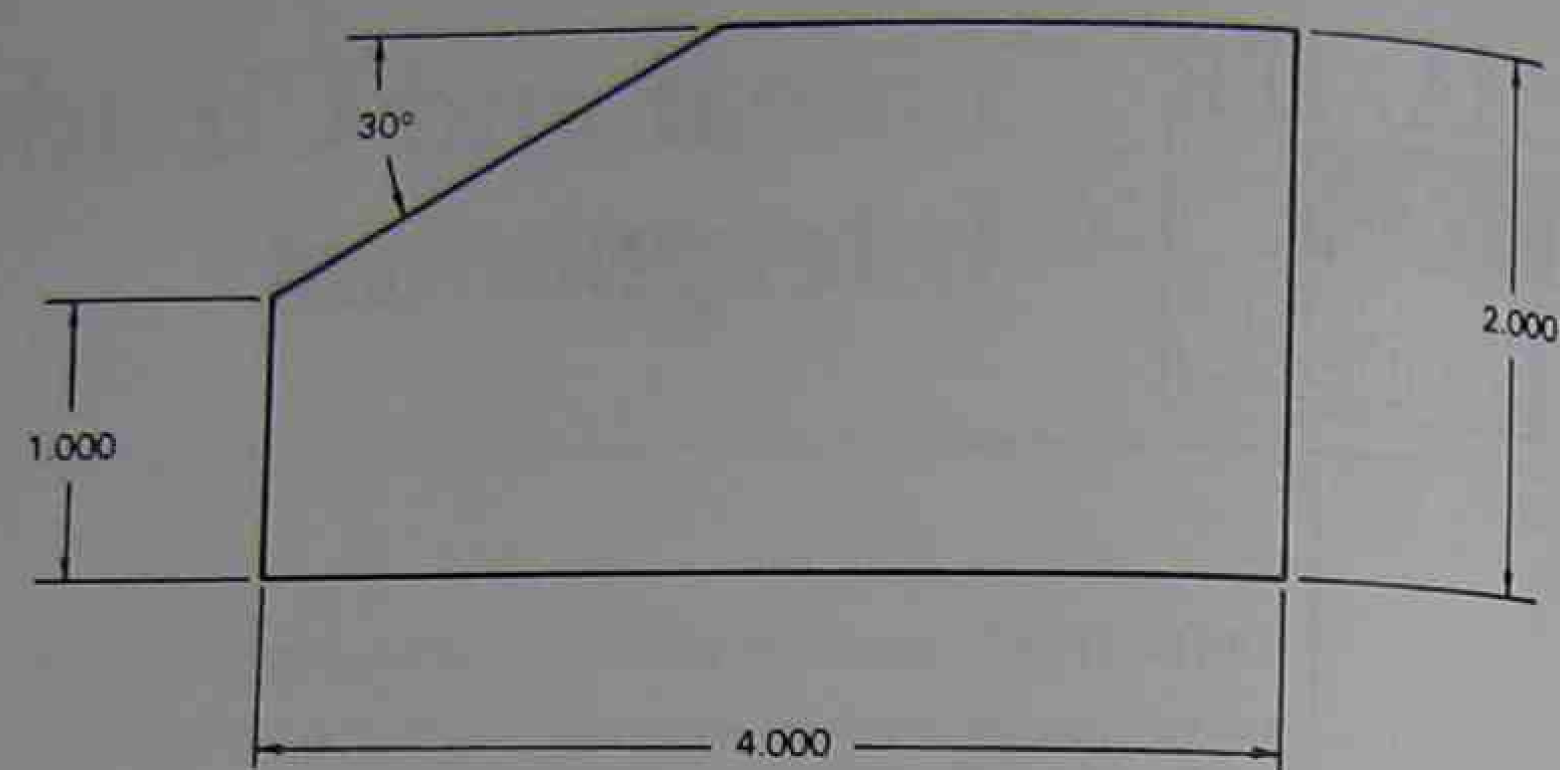


FIGURE 9–1
Part drawing

mill is being used. Before the angle can be cut, it is necessary first to position the spindle at location #2, Figure 9–2. Notice that the Y axis coordinate for location #2, as dimensioned on the part, is not the same point as the edge of the angle. In order to determine this Y axis cutter offset, it will be necessary to determine the amount that must be added to the dimension on the part print to

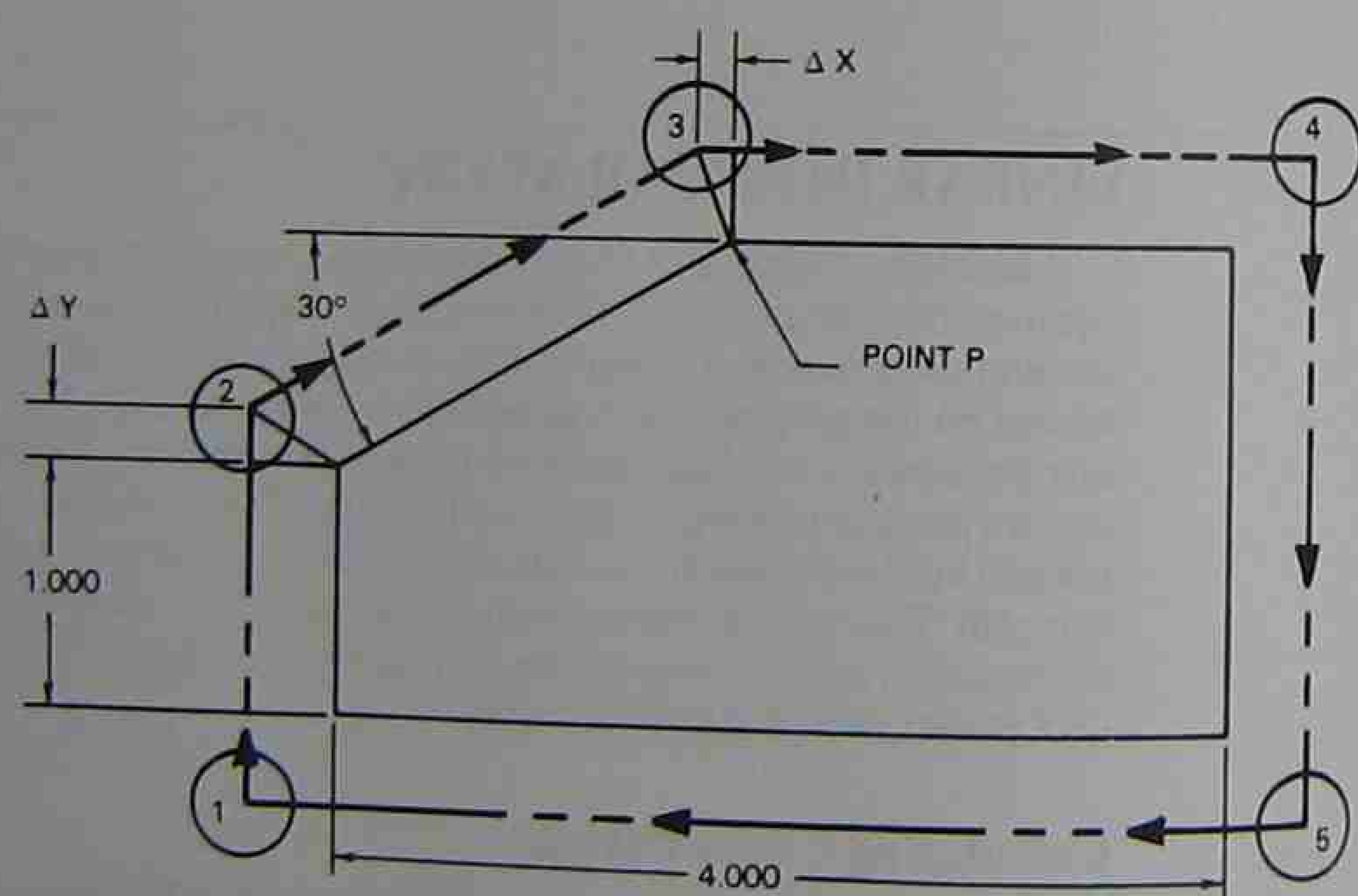


FIGURE 9–2
Cutter path for part in Figure 9–1

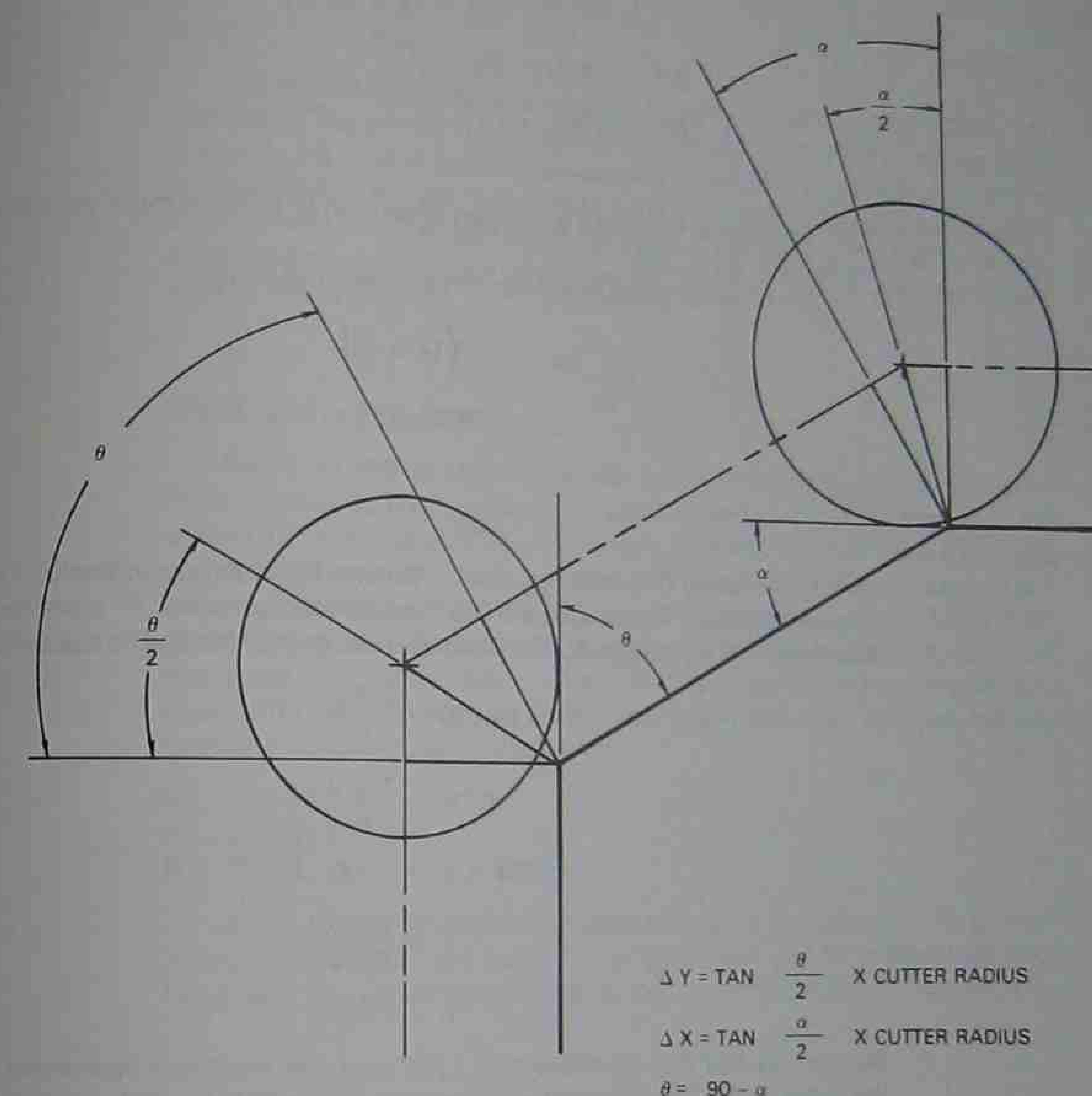


FIGURE 9–3
Determining cutter offset

place the spindle at location #2. Similarly, it will be necessary to calculate an amount to be subtracted from the point on the part designated as "P" to arrive at the X axis coordinate for location #3. This is the same cutter offset situation presented in the last chapter. Figure 9–3 represents an enlarged view of locations #2 and #3, illustrating the triangles involved in determining the offsets. The formulas from Appendix 6, Figure 1, can be used to determine the offsets as follows:

$$\Delta Y = CR\left[TAN\left(\frac{\theta}{2}\right)\right] \quad (CR = \text{cutter radius})$$

$$\Delta Y = .25(TAN\ 30)$$

$$\Delta Y = .25(.5774)$$

$$\Delta Y = .144$$

The ΔY offset to be added to the part dimension to arrive at the Y coordinate for location #2 is .144.

The offset for location #3 can be determined as follows:

$$\Delta X = CR\left[TAN\left(\frac{\alpha}{2}\right)\right]$$

$$\Delta X = .25(TAN\ 15)$$

$$\Delta X = .25(.26794)$$

$$\Delta X = .067$$

Before using this information to determine the X axis coordinate, it will also be necessary to calculate the coordinate location of point "P" along the X axis. Again, using the trigonometry formulas, the coordinate can be calculated:

$$TAN\ 30 = \frac{1.000}{b}$$

$$.5774 = \frac{1.000}{b}$$

$$.5774 \times b = 1.000$$

$$b = \frac{1.000}{.5774}$$

$$b = 1.732$$

Subtracting .067 (the ΔX offset) from 1.732 produces the X-axis coordinate for the cutter, 1.665. The ΔY offset, which was found earlier to be .144, can now be added to the 1.000 Y-axis dimension on the part to arrive at a Y-axis coordinate of 1.144. This information can now be used to write a program to mill the angle specified on the part drawing.

Machinist Shop Language

In writing the Machinist Shop Language routine, absolute positioning will be used. It will be assumed that the spindle has been positioned at location #1. The events necessary to program the movement from location #1 to location #4 are:


```
Y 1.144          F A
X 1.665  Y 2.25  F A
X 4.25           F A
```

The spindle is first sent at feedrate to location #2, using the coordinate just calculated. The spindle is then sent to location #3. Two coordinates are used to specify the desired location, because a change in both the X and Y axes is required. Notice that the X coordinate is the coordinate calculated earlier, while the Y coordinate is the one required to position the cutter at the top of the part. After completing the cut, the cutter is sent to location #4. This move is a normal straight line milling cut.

Word Address Format

Milling an angle with word address is almost the same as with Machinist Shop Language. The necessary coordinates are simply programmed along with a G01, which causes a feedrate move. In both Machinist Shop Language and word address, any line is considered to be an angle. A move along the X axis would cut an angle of 0 degrees. A move along the Y axis would cut an angle of 90 degrees. The code G01 is technically defined as linear interpolation, with linear interpolation defined simply as a feedrate move between two programmed points. To illustrate linear interpolation with word address, the program is again picked up at location #1.

```
N . . . G01 Y1.144
N . . . X1.665 Y2.25
N . . . X4.25
```

A G01 is given to institute a straight milling cut to location #2. Next the X/Y coordinates calculated earlier are programmed, just as in Machinist Shop Language. The coordinates are then given to send the spindle to location #4.

Additional Example

Linear interpolation is not difficult. Aside from calculating the cutter offsets necessary to position the spindle, it is the same as straight line milling. The only real difference is that an X and a Y coordinate are specified for the ending point of the angle since there is a change in position in both axes.

Figure 9–4 shows another cutter offset situation. This part has two angles which intersect each other. In this case, the calculation of the cutter offsets becomes somewhat more complicated. In order to program locations #1 and #3, the formula in Appendix 6, Figure 1, can be used. For location #2, the formula from Appendix 6, Figure 2, can be used. A .500-inch cutter will be assumed.

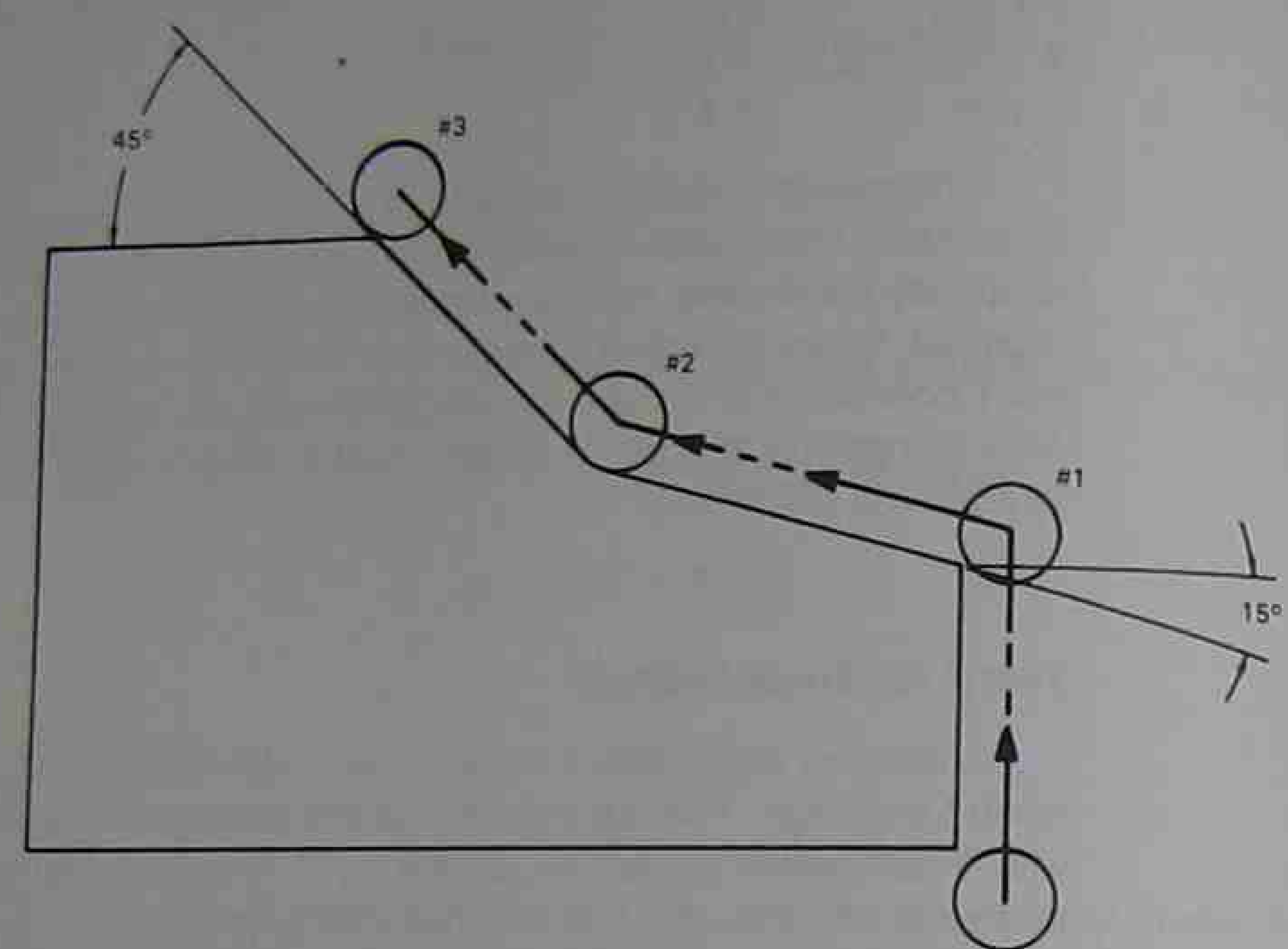


FIGURE 9–4
Part drawing with cutter path shown

Offset for location #1:

$$\Delta Y = CR\left[TAN\left(\frac{75}{2}\right)\right] \quad \Delta X = CR$$

$$\Delta Y = .25(TAN\ 37.5)$$

$$\Delta Y = .25(.7673)$$

$$\Delta Y = .1918$$

For location #2:

$$\Delta X = CR \times \frac{\left[SIN\left(\frac{(45+15)}{2}\right)\right]}{\left[COS\left(\frac{(45-15)}{2}\right)\right]}$$

$$\Delta X = .25 \times \left(\frac{SIN\ 30}{COS\ 15}\right)$$

$$\Delta X = .25 \times \left(\frac{.5}{.9659}\right)$$

$$\Delta X = .25 \times .5176$$

$$\Delta X = .1294$$

$$\Delta Y = CR \times \frac{\left[COS\left(\frac{(45+15)}{2}\right)\right]}{\left[COS\left(\frac{(45-15)}{2}\right)\right]}$$

$$\Delta Y = .25 \times \frac{(COS\ 30)}{(COS\ 15)}$$

$$\Delta Y = .25 \times \left(\frac{.866}{.9659}\right)$$

$$\Delta Y = .25 \times .8966$$

$$\Delta Y = .2241$$

For location #3:

$$\Delta X = CR\left[TAN\left(\frac{45}{2}\right)\right] \quad \Delta Y = CR$$

$$\Delta X = .25(TAN\ 22.5)$$

$$\Delta X = .25(.4142)$$

$$\Delta X = .1036$$

Other cutter situations will present themselves in NC part programming, such as arcs tangent to an angle, or arcs tangent to other arcs. A good working knowledge of the formulas listed in Figures 8–6 and 8–7 and the figures in Appendix 6 should be developed by the prospective NC part programmer.

CIRCULAR INTERPOLATION

In cutting arcs, the MCU uses its ability to generate angles to approximate an arc. Since the machine axes do not revolve around a centerpoint in a typical three-axis arrangement, the cutting of a true arc is not possible. The CNC machine calculates and then cuts a series of chord segments to generate an arc, as illustrated in Figure 9–5. These chord segments are very small and practically indistinguishable from a true arc.

Figure 9–6 shows a part with a radius to be machined. In order to generate the radius, circular interpolation will be used to send the cutter from location #3 to location #4, Figure 9–7. A .500-inch-diameter end mill will be used.

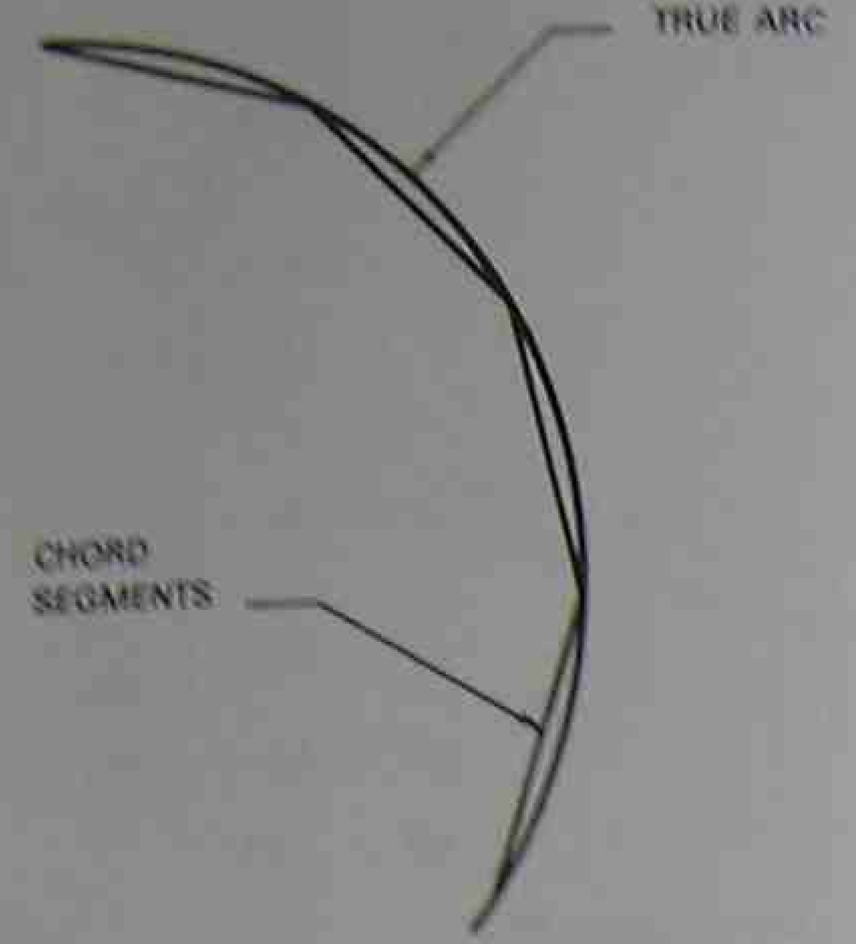


FIGURE 9–5
Circular interpolation

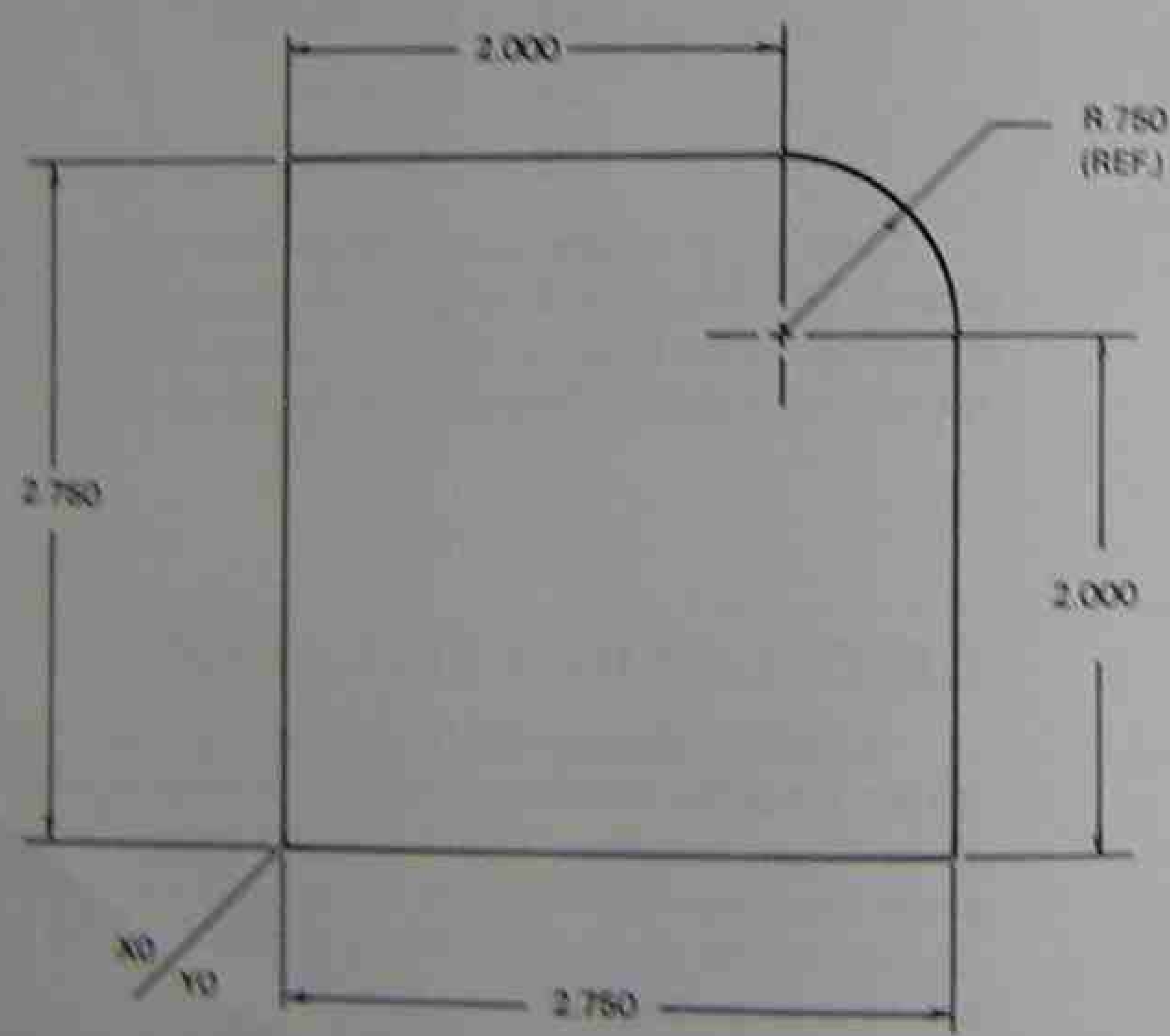


FIGURE 9–6
Part with radius to be machined

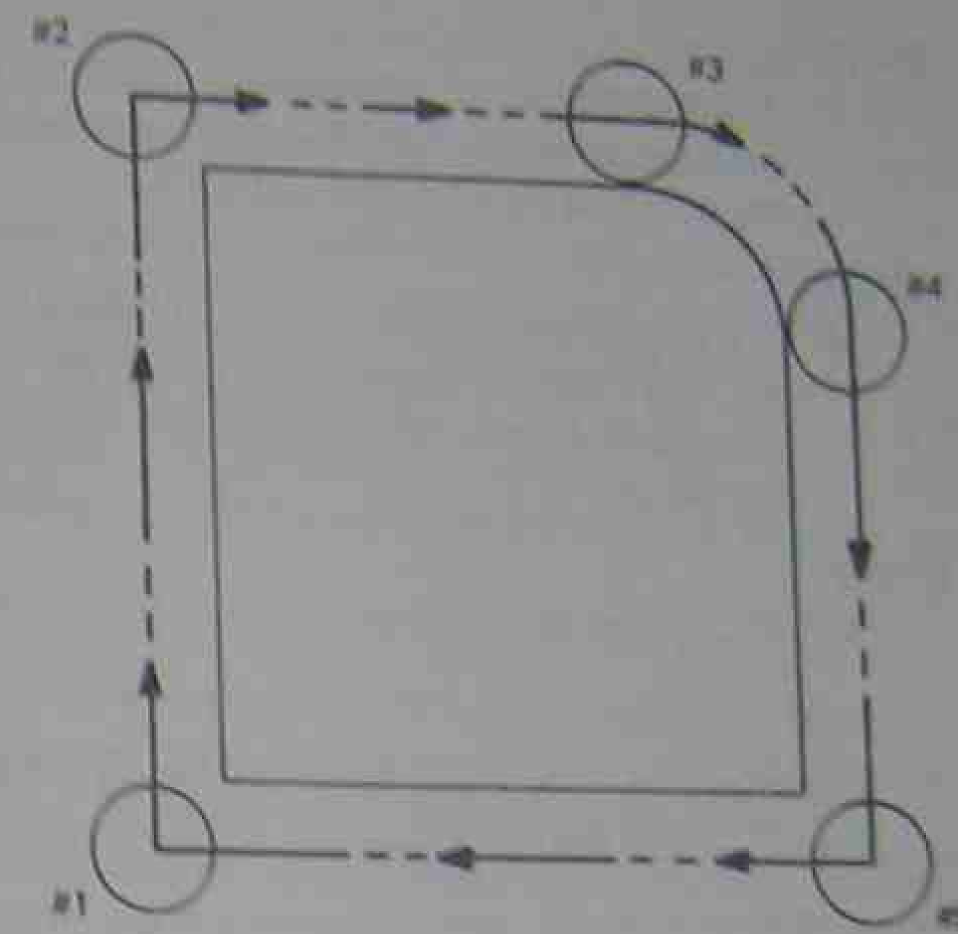


FIGURE 9–7
Cutter path for part shown in Figure 9–6

Machinist Shop Language

As with linear interpolation, it is necessary for the cutter to be positioned at the starting point of the cut (point of *arc tangency*) before the commands to generate the arc are given. To cut the arc, some new commands will be used:

ARC—This command tells the machine to cut an arc. It is also used with a direction command to define the arc direction to the machine's computer.
CW—This stands for clockwise direction. When used with ARC, it tells the machine's computer that a clockwise arc is to be cut.
CCW—This stands for counterclockwise. When used with ARC, it tells the machine's computer that a counterclockwise arc is to be cut.

A four-step procedure is used to cut an arc in Machinist Shop Language:

EVENT 1

- ARC CW/CCW—The ARC command combined with either CW or CCW tells the machine that an arc is to be cut and the direction in which it is to be cut.

EVENT 2

- X/Y coordinates of the center of the arc—Both an X and Y coordinate must be entered even though the cutter may be positioned at one of the coordinates when the cut starts.

EVENT 3

- X/Y coordinates of the endpoint of the arc cut—Both an X and Y coordinate must be entered. If no endpoint is entered, the computer will assume that the starting point is also the endpoint and generate a 360-degree arc.

EVENT 4

- ARC—The ARC command by itself constitutes the actual cutting of the arc.

Assuming the cutter has been positioned at location #1, Figure 9–7, the program routine to cut the arc will be as follows (absolute positioning is used):

```
Y3     FA
X2     FA
ARC/CW
X2   Y2 A
X3   Y2 A
ARC    FA
Y—.25 FA
```

First the cutter is sent from location #1 to #2 in a normal straight-line feedrate move. The cutter is then sent to the start of the arc. This is also a straight feedrate move. The arc is then defined as being clockwise in direction. The X/Y coordinates of the arc centerpoint (X2, Y2) are then given. The X/Y coordinates of the endpoint of the arc cut (X3, Y2) follow. The ARC command is given last in the sequence. This initiates the cutting of the arc based upon the information the computer received in the previous three events, moving the cutter from location #3 to location #4.

Word Address

Circular interpolation can be accomplished in two ways using word address format, depending on the controller. Most controllers accept information defining an arc by the arc centerpoint and endpoint of the cut. In addition, some controllers allow an arc to be defined by the radius and the endpoint of the cut. To use circular interpolation, some new codes will be needed.

G02—This code tells the MCU to cut an arc in a clockwise direction.
G03—This code tells the MCU to cut an arc in a counterclockwise direction.

These two G codes also institute the cutting of the arc.

I—This command defines the X-axis centerpoint of an arc. On some controllers I is the absolute X coordinate of the arc center. On others the I value is the incremental distance from the current cutter location to the center of the arc.
J—This command defines the Y-axis centerpoint of an arc. It can be used depending on the controller, as the absolute Y coordinate of an arc, or as the incremental distance from the current cutter location to the center of the arc.
K—This command, used like I and J, defines the Z-axis centerpoint of the arc if performing circular interpolation in either the X/Z or Y/Z planes.
R—This defines the arc radius when the radius is used instead of the centerpoint.

For consistency with Machinist Shop Language, this text will use I, J, and K as the absolute coordinates of an arc center. It should be noted that Fanuc and other similar controls use I, J, K as incremental distances.

As in Machinist Shop Language, the cutter must be positioned at the point of arc tangency before the commands are given to cut the arc. With some controllers, a 90-degree arc is the largest arc segment that can be cut. Cutting 360 degrees must be programmed as four arcs of 90 degrees each. Other controllers allow the cutting of a 360-degree arc in one block of information.

In word address format, a three-step process is followed to cut an arc. All three steps are usually contained in one program line.

For Centerpoint Programming

1. Give the G code for circular interpolation in the direction desired.
2. Give the X/Y coordinates of the endpoint of the arc, using X and Y to define the point.
3. Give the X/Y coordinates of the arc centerpoint, using I and J to define the point.

For Radius Programming

1. Give the G code for circular interpolation in the direction desired.
2. Give the X/Y coordinates of the arc endpoint, using X and Y to define the point.
3. Give the radius of the arc preceded by the R address.

The blocks to cut the arc moving from location #1 to #5 are as follows:

By the Centerpoint Method

N...G01 Y3
N...X2
N...G02 X3 Y2 I2 J2
N...G01 Y—.25

The first block is a straight milling cut to feed the cutter from location #1 to location #2. The second block is a straight milling cut to feed the cutter from location #2 to location #3 (the starting point of the arc cut). The third block initiates circular interpolation in a clockwise direction using G02. The X/Y coordinates of the arc endpoint and arc centerpoint are given, using I to define the X-axis centerpoint and J to define the Y-axis centerpoint. This block programs the entire arc, feeding the cutter from location #3 to location #4. The last block feeds the cutter from location #4 to location #5. Note that G01 was specified to put the machine back into the feedrate mode.

By the Radius Method

N...G01 Y3
N...X2
N...G02 X3 Y2 R.75
N...G01 Y—.25

The first block is a straight milling cut to feed the cutter from location #1 to location #2. The second block is a straight milling cut to feed the cutter from location #2 to location #3 (the starting point of the arc cut). The third block initiates circular interpolation in a clockwise direction using G02. X and Y define the endpoint of the arc cut. The arc's radius is given using the R address. As in the preceding example, this block programs the entire arc, feeding the cutter from location #3 to location #4. The last block feeds the cutter from location #4 to location #5.

Additional Example

The programs just given are for simple arcs which intersect a line parallel to a machine axis. In many cases, however, an arc will intersect an angle or another arc. Figures 9–8 and 9–9 are examples of such cases. The cutter offsets for these situations can be found by using the formulas from Appendix 6. The cutter radius (CR) in the following examples is .250 inch.

To calculate ΔX and ΔY in Figure 9–8, it is necessary to calculate Δi and Δj:

$$\Delta j = 1.25 - .75$$

$$\Delta j = .5$$

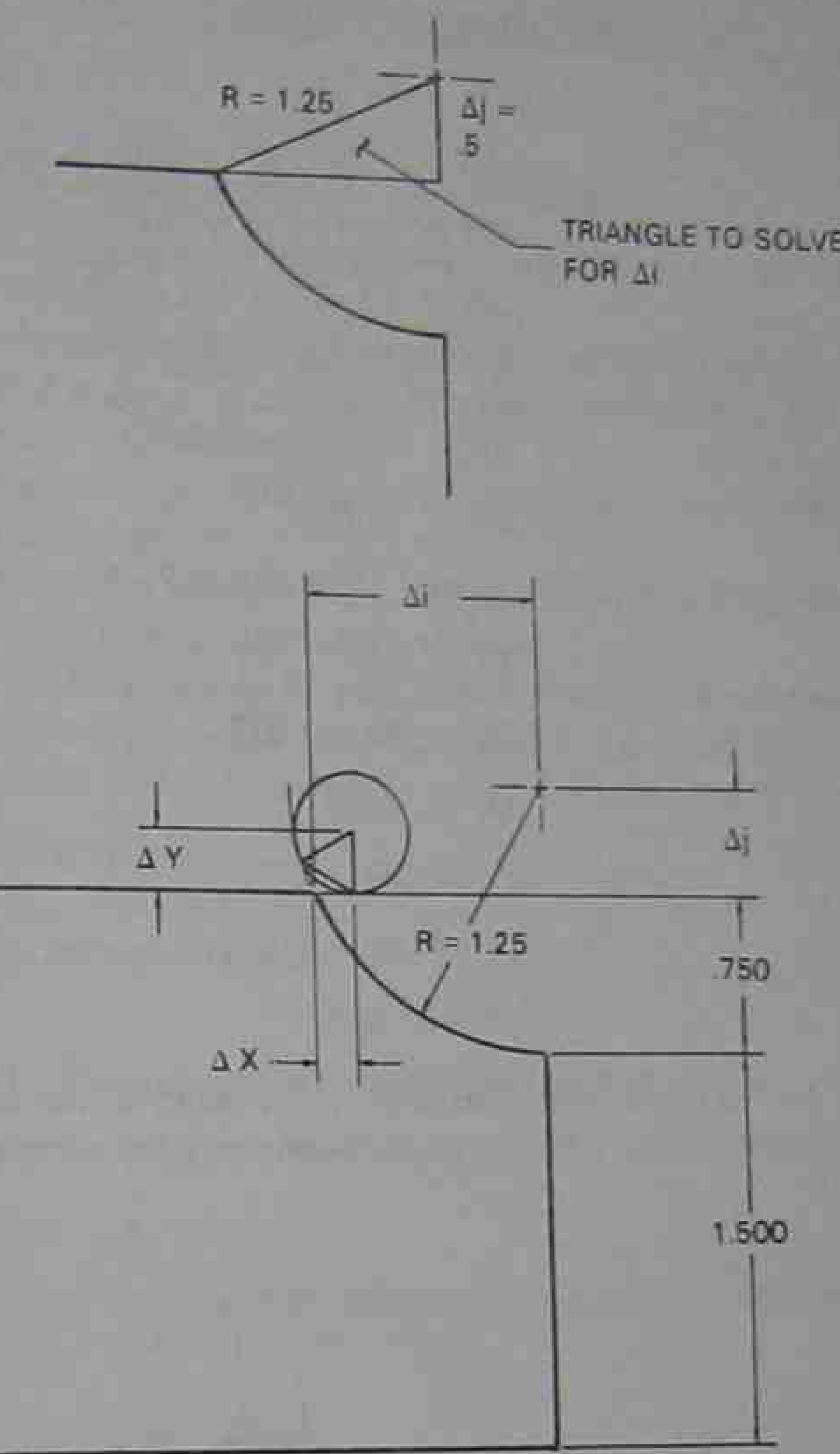


FIGURE 9–8

Using the Pythagorean theorem, i can be determined:

$$\Delta i = \sqrt{(1.25^2 - .5^2)}$$

$$\Delta i = \sqrt{(1.562 - .25)}$$

$$\Delta i = \sqrt{1.312}$$

$$\Delta i = 1.1454$$

This information can then be used to determine X and Y:

$$\Delta Y = CR$$

$$\Delta X = \sqrt{\Delta i - (R-CR)^2 - (\Delta j - CR)^2}$$

$$\Delta X = 1.1454 - \sqrt{(1.25-.25) - (.5-.25)}$$

$$\Delta X = 1.1454 - \sqrt{1-.0625}$$

$$\Delta X = 1.1454 - \sqrt{.9375}$$

$$\Delta X = 1.1454 - .96825$$

$$\Delta X = .17715$$

To calculate ΔX and ΔY in Figure 9–9:

$$\Delta X = CR \times \text{SIN } 45$$

$$\Delta X = .25 \times .7071$$

$$\Delta X = .1769$$

$$\Delta Y = CR \times \text{COS } 45$$

$$\Delta Y = .25 \times .7071$$

$$\Delta Y = .1769$$

In this case, since the angle is 45 degrees, the offsets for ΔX and ΔY are the same. Had a different angle been used, the offsets would be different.

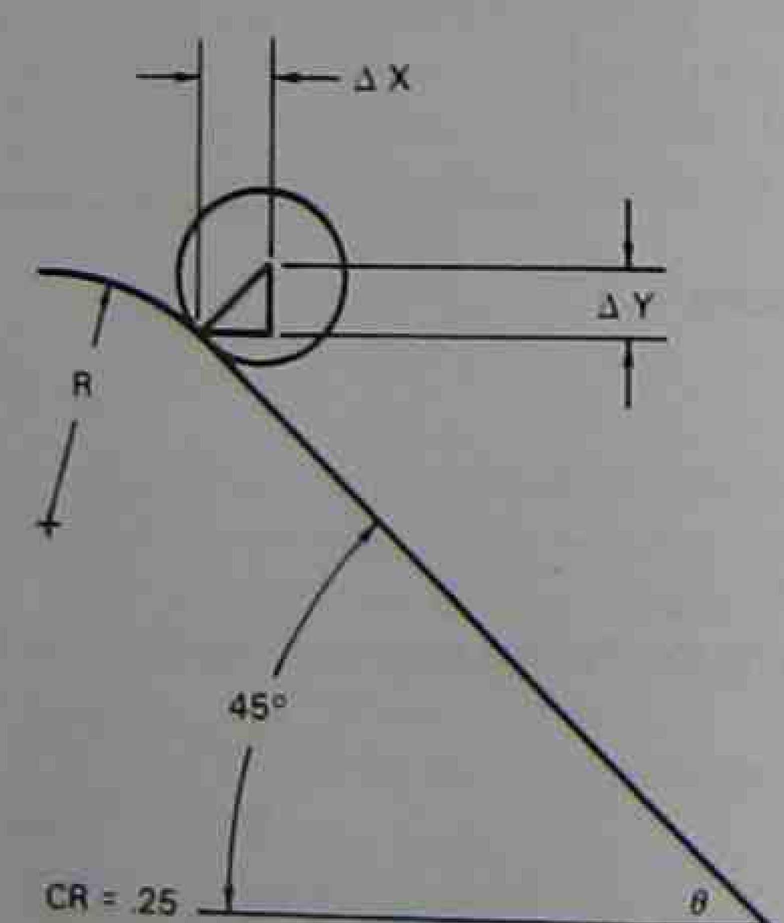


FIGURE 9–9

SUMMARY

The important concepts presented in this chapter are:

- Linear interpolation is the ability to cut angles. It is simply a feedrate move between two points.
- Circular interpolation is the ability to cut arcs or arc segments. Arcs are cut by means of a series of chordal segments generated by the MCU to approximate the arc curvature.
- It is necessary to calculate the cutter offset coordinates when using linear and circular interpolation.
- G01 is the code to institute linear interpolation in word address; in Machinist Shop Language it is treated as a feedrate move.
- G02 and G03 are used to institute circular interpolation in word address; in Machinist Shop Language the ARC command is used.
- The Machinist Shop Language format for circular interpolation is:

  ARC CW/CCW
  X/Y (centerpoint coordinates)
  X/Y (endpoint coordinates)
  ARC

- The word address format for circular interpolation is:

  For centerpoint programming:

  G02/G03 X . . . Y . . . I . . . J . . .

  Where X and Y are the endpoint coordinates and I and J are the centerpoint coordinates.

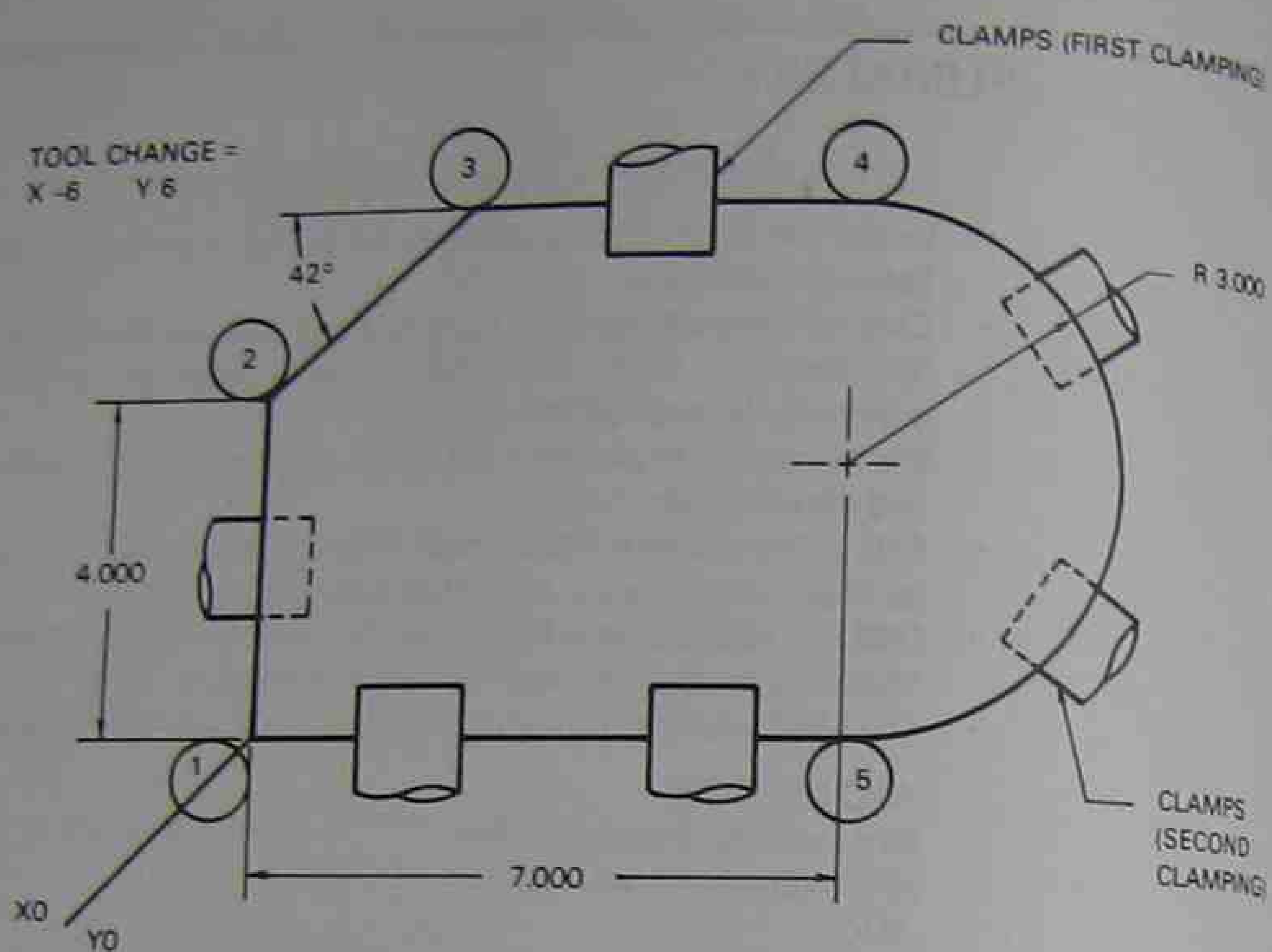
  For radius programming:

  G02/G03 X . . . Y . . . R . . .

  Where X and Y are the endpoint coordinates and R is the arc radius.

REVIEW QUESTIONS

1. What will be the result of cutting the angle in Figure 9–10 if the Y offset is not calculated, but the 4.000 dimension is used instead?
2. What two formulas can be used in calculating coordinates for simple angles where the angle intersects a line parallel to a machine axis?



MATERIAL: 1.0 THICK 7075 T-6 ALUMINUM

INSTRUCTIONS: MILL FROM 1 TO 2
MILL FROM 2 TO 3
JUMP CLAMPS
MILL FROM 4 TO 5
MOVE TO TOOL CHANGE
DWELL TO MOVE CLAMPS
MILL FROM 3 TO 4
MOVE TO 5
MILL FROM 5 TO 1
MOVE TO TOOL CHANGE

FIGURE 9–10
Part drawing for review questions #1 and #7

3. What code is used in word address format to initiate linear interpolation? In Machinist Shop Lanugage?
4. What is the format for circular interpolation in Machinist Shop Language? In word address?
5. What are I and J used for? What is R used for?
6. When is the ARC command used in Machinist Shop Language?
7. Write a program to mill the part in Figure 9–10:
   a. In Machinist Shop Language.
   b. In word address.

CHAPTER 10

Cutter Diameter Compensation

OBJECTIVES Upon completion of this chapter, you will be able to:

- Define cutter diameter compensation.
- Describe ramp on and ramp off moves and explain their importance.
- List the precautions necessary when using cutter diameter compensation.
- Write programs in word address and Machinist Shop Language that utilize cutter diameter compensation.

DEFINITIONS AND CODES

Programs presented in previous chapters required an allowance for the cutter radius in the programmed coordinates. Some types of CNC machinery have a built-in feature called *cutter diameter compensation (cutter comp)* that allows the part line to be programmed. (Confusion may be caused by use of the terms "offset" and "compensation." In this text, "compensation" refers to cutter diameter offset. The term "offset" refers to tool length offset and the change in axis coordinates when programming arcs and angles.) Cutter comp is also called cutter radius offset (CRO) by some controller manufacturers. In computer-aided programming languages (such as APT) it is also called cutcom. These terms all refer to the same thing: a built-in cycle in the MCU that, when activated, alters the tool path by an amount contained in the cutter comp register. The value in the register is entered in by the setup person when the job is being prepared.

In both word address and Machinist Shop Language formats, cutter comp is accomplished through the use of G codes: G40, G41, G42.

G40—Cutter diameter compensation cancel. Upon receiving a G40, cutter diameter compensation is turned off. The tool will change from a compensated position to an uncompensated position on the next X, Y, or Z axis move.

G41—Cutter diameter compensation left. Upon receiving a G41, the tool will compensate to the left of the programmed surface. The tool will move to a compensated position on the next X, Y, or Z axis move after the G41 is received.

G42—Cutter diameter compensation right. Compensates to the right of the programmed surface.

Machinist Shop Language allows for compensation on the X and Y axes. Word address format allows compensation on any axis and uses a G code to determine which axes combination is to be used. If the part is to be machined using the X and Y axes, compensation is desired in the X/Y plane. If using the X and Z axes, compensation in the X/Y plane is needed. If using the Y and Z axes, compensation is needed in the Y/Z plane. The X/Y plane is used most commonly, which may explain why Machinist Shop Language does not offer a choice of axes. The G codes used to select the desired work plane in word address are:

G17—X/Y plane.
G18—Z/X plane.
G19—Y/Z plane.

Two terms are important for understanding cutter diameter compensation: *ramp on* and *ramp off*. Figure 10–1 will help to illustrate their meaning. If a tool is moved from point #1 to point #2 following a G41 command, the cutter will compensate in a plane perpendicular to the part surface and the spindle will move to point #3 rather than point #2. This initial compensation move is called the *ramp on* move. The machine is in the process of adjusting its path for the entire move from point #1 to point #2. By the time it reaches point #2, it has fully compensated its path.

In Figure 10–2 another type of situation is demonstrated. The cutter started at point #1 in this illustration, presenting the possibility that the corner of the part might be cut off in the process of moving to point #2 if the spindle were moving downward or already positioned there. If point #1 was the desired tool change location for this part, the spindle would need to be fully retracted and lowered after reaching the programmed location. When clamps or fixturing devices do not interfere, it is not uncommon to rapid the Z axis to depth on the same move that positions X and Y. Some controllers do not allow cutter comp to be instituted in two axes simultaneously. In these cases it is necessary to program a location away from the part surface, and ramp on the compensation 90 degrees from the desired part surface, as in Figure 10–3. Controllers are often particular about the manner in which cutter comp is ramped on. It is advisable when programming several different controllers or controllers of different ages to use the method in Figure 10–3. This is the most successful method of ramping on cutter comp.

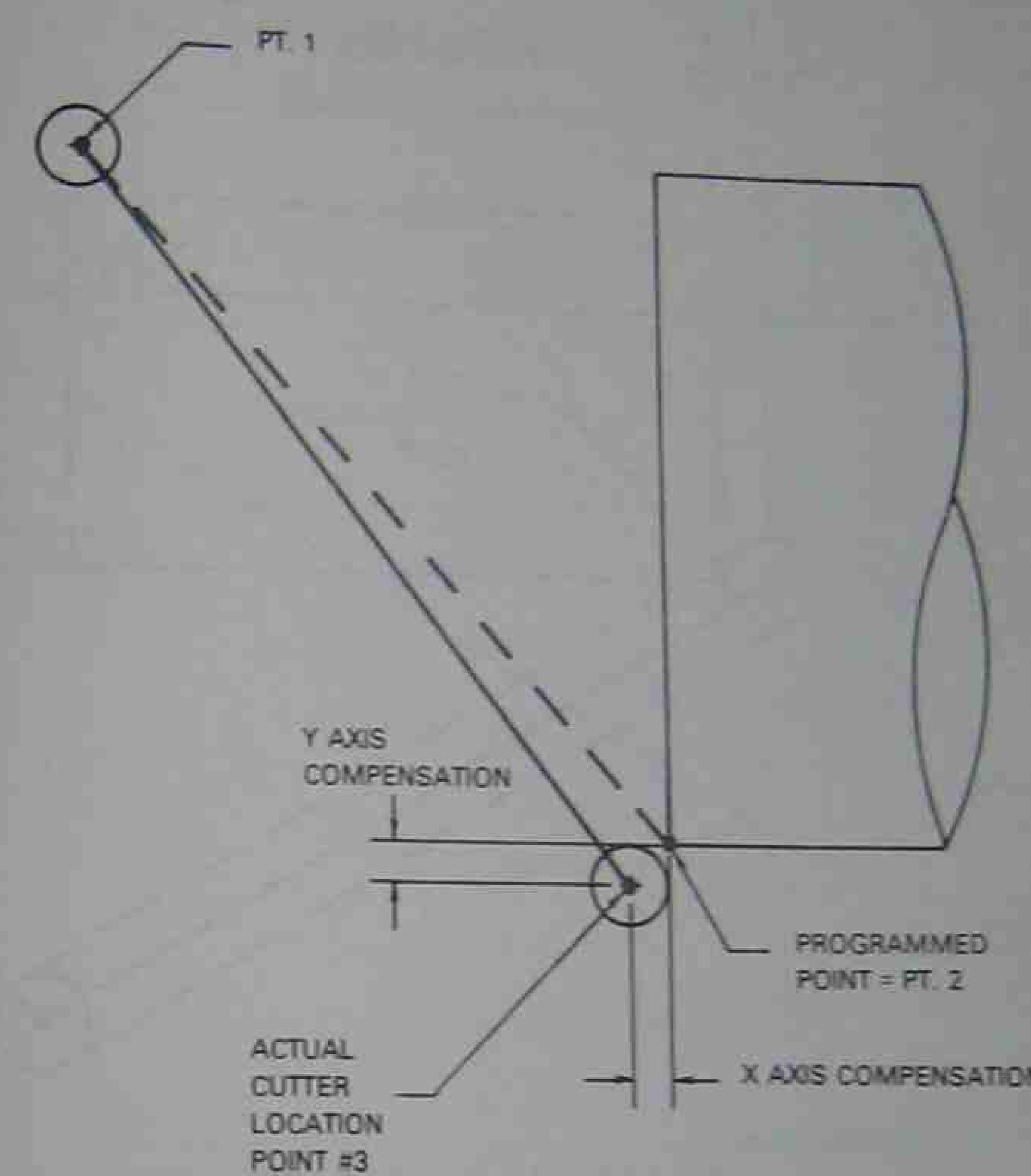


FIGURE 10–1
Ramp on move

The ramp off move is the opposite of the ramp on move and the same precautions are necessary. In Figure 10–2, assume that cutter comp is canceled and a move is made from point #2 to point #1. In this case, the corner of the part may also be cut off. Remember, *the compensation is not turned off completely until the ramp off move is completed.*

Two additional points should be noted. First, with many controllers, cutter comp must be turned on after the length offset is initiated. Similarly, cutter comp must be cancelled prior to cancelling the tool length offset. Failure to do this will result in the controller halting executing of the program, and an alarm signaling at the MCU console.

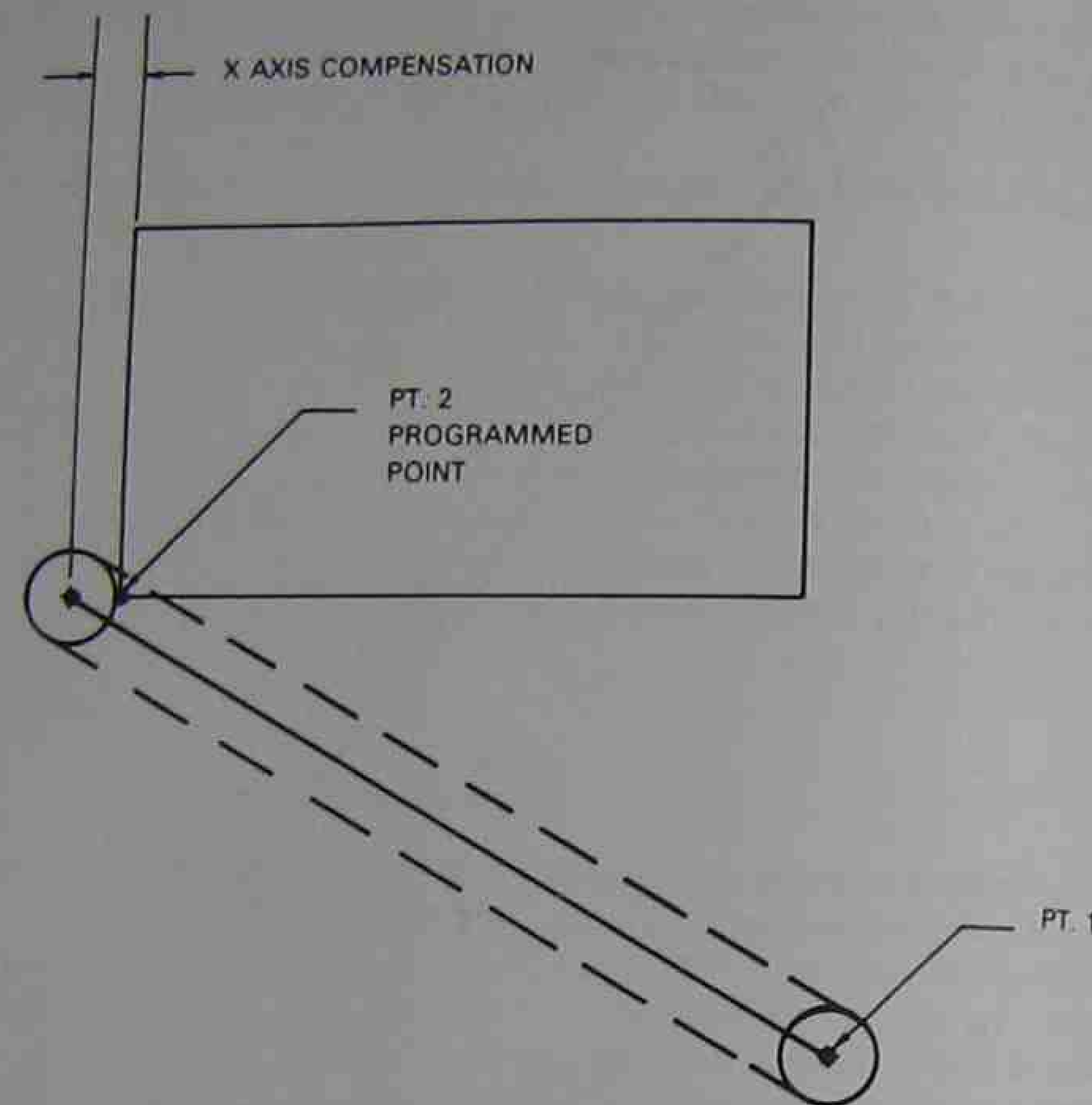


FIGURE 10–2

Second, in word address format controllers cutter comp must be commanded in rapid or feedrate modes, not in circular interpolation mode. If G40, G41, or G42 are commanded after G02 or G03, a machine controller alarm will result.

A short program to mill the part in Figure 10–4 is given in Figures 10–5 (Machine Shop Language) and 10–6 (word address format). The programs contain both a roughing and a finishing cut. One way to accomplish this without changing tools or programmed coordinates is to program the diameter of the cutter as two separate diameters. This program uses a .500-inch-diameter end mill. By defining it as both a .520-inch diameter and a .500-inch diameter and using the same coordinates for both passes, the result is that .010 inch of stock is left for the finish pass.

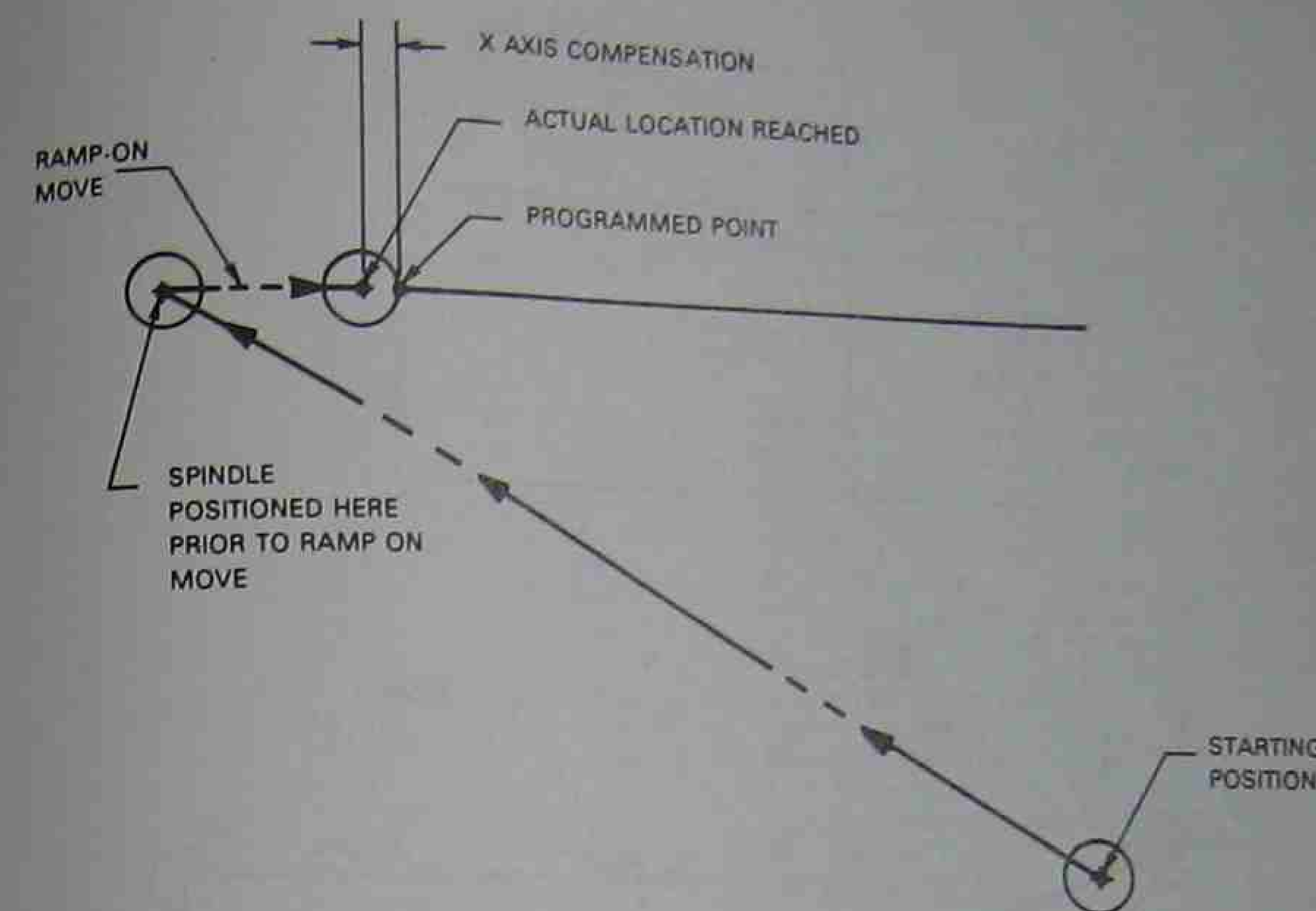


FIGURE 10–3

MACHINIST SHOP LANGUAGE

PROGRAM EXPLANATION

(Refer to Figure 10–5.)

EVENT 1

- TOOL 1—Tells the MCU that tool information is being assigned.

EVENT 2

- X.52—The diameter of tool #1. In reality a .500-inch-diameter end mill is being used. Defining the cutter as .520 inch will leave .010 of stock per side on the part.
- Z3—The tool offset value for tool length offset.
- A—Specifies absolute positioning.

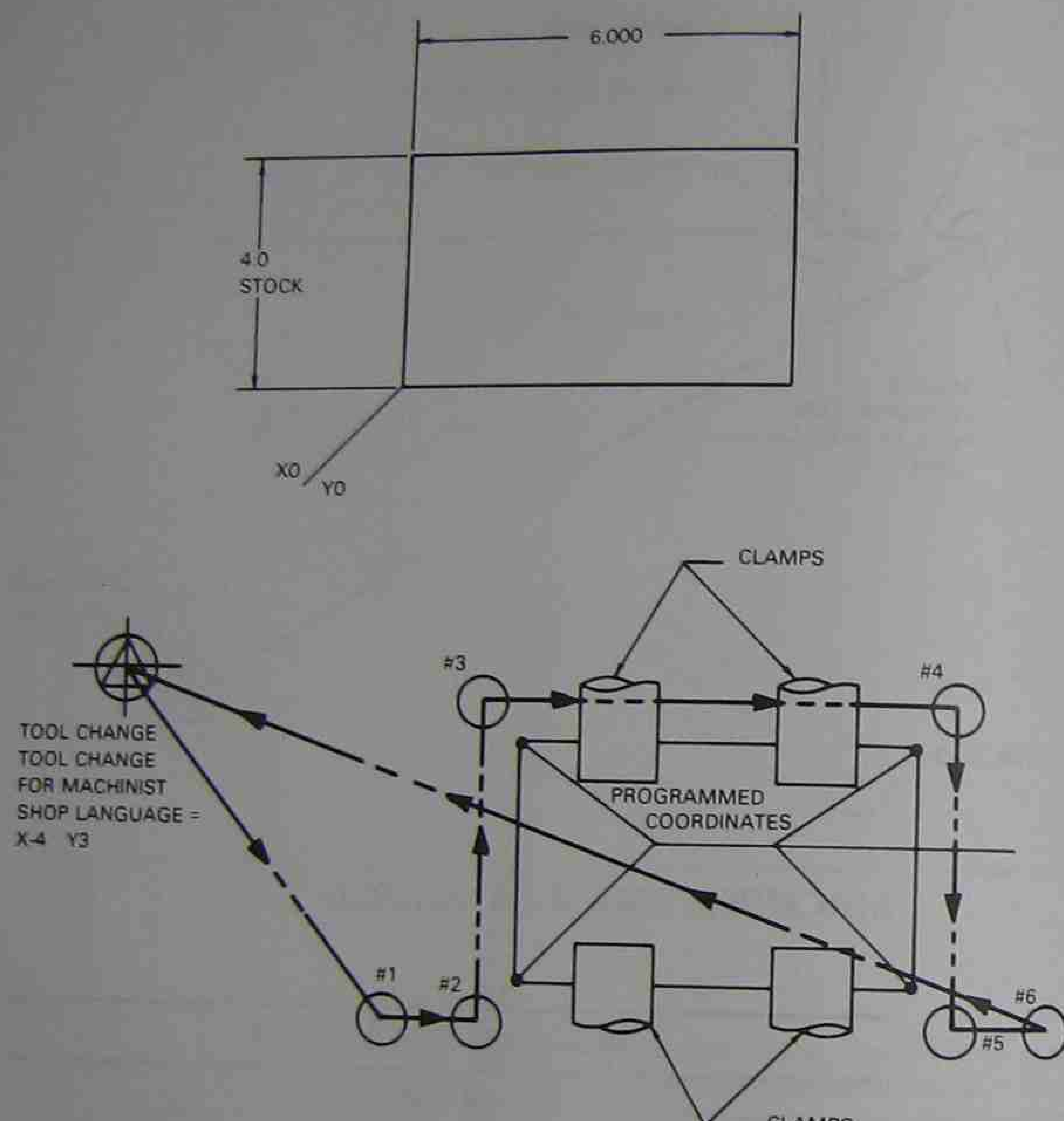


FIGURE 10–4
Part drawing and cutter path for cutter diameter compensation


```
XO/YO = LOWER LEFT CORNER
TOOL CHANGE = X-4 Y3
TOOLS:   .500 IN. END MILL
CLEARANCE OVER CLAMPS:  3.000 MIN.
BUFFER ZONE .100

 1 TOOL 1001
 2 X.52 Z3         A     REM:TOOL LENGTH/DIA FOR #1
 3 TOOL 1002
 4 X.5 Z3          A     REM:TOOL LENGTH/DIA FOR #2
 5 X-4 Y3          RA    REM:TCH
 6 TOOL 1
 7 FEED 12.8
 8 X-1 Y-.25       RA    REM:POSITION X/Y TO #1
 9 ZO              RA    REM:RAPID Z TO BUFFER START
10 Z-.62           FA    REM:FEED Z TO DEPTH
11 G41                   REM:CUTTER COMP LEFT
12 XO              FA    REM:RAMP ON MOVE TO #2
13 Y4              FA    REM:FEED FROM #2 TO #3
14 Z3              RA    REM:RAISE Z TO CLEAR CLAMP
15 X6              RA    REM:RAPID FROM #3 TO #4
16 ZO              RA    REM:RAPID Z TO BUFFER START
17 Z-.62           FA    REM:FEED Z TO DEPTH
18 YO              FA    REM:FEED FROM #4 TO #5
19 G40                   REM:CUTTER COMP CANCEL
20 X7                    REM:RAMP OFF MOVE FROM #5 TO #6
21 TOOL 0                REM:CANCEL TOOL OFFSET
22 ZO              RA    REM:RETRACT Z
23 X-1 Y-.25       RA    REM:MOVE TO #1
24 TOOL 2
25 ZO              RA    REM:RAPID Z TO BUFFER START
26 Z-.62           FA    REM:FEED Z TO DEPTH
27 G41                   REM:CUTTER COMP LEFT
28 XO              FA    REM:RAMP ON MOVE TO #2
29 Y4              FA    REM:FEED FROM #2 TO #3
30 Z3              RA    REM:RAISE Z TO CLEAR CLAMP
31 X6              RA    REM:RAPID FROM #3 TO #4
32 ZO              RA    REM:RAPID Z TO BUFFER START
33 Z-.62           FA    REM:FEED Z TO DEPTH
34 YO              FA    REM:FEED FROM #4 TO #5
35 G40                   REM:CUTTER COMP CANCEL
36 X7                    REM:RAMP OFF MOVE FROM #5 TO #6
37 TOOL 0                REM:CANCEL TOOL OFFSET
38 ZO              RA    REM:RETRACT Z
39 X-4 Y3          RA    REM:MOVE TO TCH
41 END
```

FIGURE 10–5
Cutter diameter compensation program to mill the part in Figure 10–4, Machinist Shop Language

EVENT 3
- TOOL 1002—Tells the MCU that tool #2 is being defined.

EVENT 4
- X.5—The diameter of tool #2. This is the actual diameter of tool #1.
- Z3—The tool length offset value for tool #3.
- A—Specifies absolute dimensioning.

EVENT 5
- X/Y coordinates—To rapid to the tool change location.
- R—Specifies rapid traverse mode.
- A—Specifies absolute positioning.

EVENT 6
- TOOL 1—Calls up tool #1's length and diameter. The length will be used for tool length offset; the diameter will be used for cutter diameter compensation.

EVENT 7
- FEED 12.8—Assigns a feedrate of 12.8 inches per minute to be used with all feedrate moves.

EVENT 8
- X−1 Y−.25—Positions the cutter at location #1, Figure 10−4. This location sets the cutter up for a ramp on move. Although the Anilam controller that uses Machinist Shop Language will permit compensation to occur in two axes simultaneously, it is not recommended by the manufacturer. Many controllers will not allow this; the type of ramp on move being used in this program is considered a safe move for all controllers.
- R—Specifies rapid traverse mode.
- A—Specifies absolute positioning.

EVENT 9
- Z0—Rapids the cutter to the height of the buffer zone. A .100-inch buffer is being used with this program.
- R—Specifies rapid traverse mode.
- A—Specifies absolute positioning.

EVENT 10
- Z−.62—Feeds the cutter to the proper depth, which is .020 inch below the part.
- F—Specifies feedrate mode.
- A—Specifies absolute positioning.

EVENT 11
- G41—Initiates cutter diameter compensation left. The machine will reach its compensated position at the end of the next programmed move.

EVENT 12
- X0—The ramp on move, positions the cutter at location #2, Figure 10−4.
- F—Specifies feedrate mode.
- A—Specifies absolute positioning.

EVENT 13
- Y4—The coordinate necessary to position the machine at location #3. Note that the part dimension has been programmed, not the cutter centerline.
- F—Specifies feedrate mode.
- A—Specifies absolute positioning.

EVENT 14
- Z3—Retracts the spindle to a height sufficient to clear the clamps. In this case, canceling the tool length offset to accomplish the Z-axis retraction is not possible. The same command that called up the tool length offset also defines the cutter diameter. The tool length offset should not be canceled until after a G40 cutter comp cancel has been completed. Not all controllers allow the Z axis to move once cutter comp has been initiated unless preceded by a particular G code. This will be demonstrated in the word address example.
- R—Specifies rapid traverse mode.
- A—Specifies absolute positioning.

EVENT 15
- X6—The coordinate necessary to position the machine at location #4, again a part coordinate.
- R—Specifies rapid traverse mode.
- A—Specifies absolute positioning.

EVENT 16
- Z0—Positions the cutter at the start of the buffer zone.
- R—Specifies rapid traverse mode.
- A—Specifies absolute positioning.

EVENT 17
- Z−.62—Feeds the cutter to milling depth.
- F—Specifies feedrate mode.
- A—Specifies absolute positioning.

EVENT 18

- Y0—Feeds the cutter from location #4 to location #5.
- F—Specifies feedrate mode.
- A—Specifies absolute positioning.

EVENT 19

- G40—Cancels cutter diameter compensation. The cutter will be uncompensated by the end of the next programmed move.

EVENT 20

- X7—The ramp off move, moving the cutter from location #5 to location #6. During the course of this move, the cutter will become uncompensated. It is important to ramp off to a point at least one cutter radius away from the part. This move can be made in either rapid or feedrate mode, as the cutter is clear of the part. A feedrate move was selected here, since the distance is so small.
- F—Specifies feedrate mode.
- A—Specifies absolute positioning.

EVENT 21

- TOOL 0—Cancels the tool length offset. Notice that cutter comp was canceled first.

EVENT 22

- Z0—Retracts the spindle.
- R—Specifies rapid traverse mode.
- A—Specifies absolute positioning.

EVENT 23

- X-1 Y-.25—The coordinates of location #1.
- R—Specifies rapid traverse mode.
- A—Specifies absolute positioning.

EVENT 24

- TOOL 2—Calls up tool #2's length and diameter. This time a .500-inch diameter will be used. The remaining .010 of stock will be removed from each side.

Events 25 to the end duplicate events 8 through 23. At the end of the program, the tool is sent back to the tool change location. The coordinates for the cutter locations could have been placed in a subroutine and so not repeated. Chapter 11 will deal with subroutines.

WORD ADDRESS FORMAT

PROGRAM EXPLANATION

(Refer to Figure 10–6.)

N010

- N010—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- G40—Cancels any active cutter comp. This code and the other codes in this line are used as a safety device. If any program was run on the machine prior to this one, there might be active codes detrimental to the execution of this program. This block eliminates any chance of accident.
- G49—Cancels any active tool offset.
- G70—Selects inch input.
- G80—Cancels any active canned cycle.
- G90—Selects absolute positioning.

N020

- N020—The sequence number.
- G10—Tells the MCU that tool information is being defined.
- H01—Selects tool register #1 to hold the tool definition information.
- X.52—Defines the tool diameter.
- Z3—Defines the tool length offset.

N030

- N030—The sequence number.
- G10—Tells the MCU that tool information is being defined.
- H02—Selects tool register #2.
- X.5—Defines the tool diameter.
- Z3—Defines the tool length offset.

N040

- N040—The sequence number.
- M06—Initiates an automatic tool change.
- T1—Selects tool #1 for use.

N050

- N050—The sequence number.
- G45 H01—Calls up the offset in register #1 for use with the tool.

```
X0/Y0 = LOWER LEFT CORNER
TOOLS:  .500 IN. END MILL
CLEARANCE ABOVE CLAMPS:  3.000 MIN.
BUFFER ZONE:  .100

N010 G00 G40 G49 G70 G80 G90  REM:SAFETY LINE
N020 G10 H01 X.52 Z3
N030 G10 H02 X.5 Z3
N040 M06 T1
N050 G45 H01                  REM:TOOL #1 OFFSET
N060 X-1 Y-.25 S2500 M03      REM:RAPID TO LOCATION #1
N070 Z0                       REM:RAPID TO BUFFER START
N080 G01 Z-.62 F12.8 M08      REM:FEED TO DEPTH
N090 G17 G41 X0               REM:RAMP ON MOVE TO #2
N100 Y4                       REM:FEED FROM #2 TO #3
N110 G00 G18 Z3               REM:RETRACT SPNDL
N120 X6                       REM:RAPID FROM #3 TO #4
N130 Z0                       REM:RAPID TO BUFFER START
N140 G01 Z-.62                REM:FEED TO DEPTH
N150 G17 Y0                   REM:FEED FROM #4 TO #5
N160 G40 X7                   REM:RAMP OFF TO #6
N170 G00 G49 Z0               REM:CANCEL OFFSET RETRACT SPNDL
N180 X-1 Y-.25                REM:RAPID TO #1
N190 G45 H02                  REM:TOOL #2 OFFSETS
N200 Z0                       REM:RAPID TO BUFFER START
N210 G01 Z-.62                REM:FEED TO DEPTH
N220 G17 G41 X0               REM:RAMP ON MOVE TO #2
N230 Y4                       REM:FEED FROM #2 TO #3
N240 G00 G18 Z3               REM:RETRACT SPNDL
N250 X6                       REM:RAPID FROM #3 TO #4
N260 Z0                       REM:RAPID TO BUFFER START
N270 G01 Z-.62                REM:FEED TO DEPTH
N280 G17 Y0                   REM:FEED FROM #4 TO #5
N290 G40 X7                   REM:RAMP OFF TO #6
N300 G00 G49 Z0 M09           REM:CANCEL OFFSET RETRACT SPNDL
N310 X-12 Y8 M05              REM:RAPID TO PARK POSITION
N320 M30
```

FIGURE 10–6
Cutter diameter compensation program to mill the part in Figure 10–4, word address format

N060

- N060—The sequence number.
- X−1 Y−.25—The coordinates of location #1.
- S2500—Sets the spindle speed to 2500 RPM.
- M03—Turns the spindle on clockwise.

N070

- N070—The sequence number.
- Z0—Rapids the spindle to the start of the buffer zone. A .100 buffer is being used with this program.

N080

- N080—The sequence number.
- G01—Puts the machine in feedrate mode.
- Z−.62—The Z-axis coordinate to feed the cutter to depth.
- F12.8—Assigns a feedrate to be used in feedrate moves.
- M08—Turns the coolant on.

N090

- N090—The sequence number.
- G17—Selects the X/Y plane.
- G41—Initiates cutter diameter compensation left.
- X0—The coordinate of the part surface. Since a G41 was issued, the cutter will be positioned the distance of the tool radius away from the cutter at location #2. This tool was defined as having a .520-inch diameter. It is in reality .500 inch. This will leave stock for finishing on the part.

N100

- N100—The sequence number.
- Y4—Feeds the cutter from location #2 to location #3.

N110

- N110—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- G18—Selects the X/Z plane, since movement is required in Z to retract the spindle to clear the clamps on the next move.
- Z3—Raises the spindle to clear the clamps. Since a 3-inch clearance was specified to clear the clamps in the setup sheet, this is a valid move. The disadvantage to this technique is that the setup man may not have established the correct clearance. An alternative move would be to cancel the cutter comp and tool offset, raise the spindle to Z0, and reinstitute the tool offset and cutter comp after repositioning the X and Y axes.

N120

- N120—The sequence number.
- X6—The coordinate to move from location #3 to location #4. The move is in rapid traverse mode.

N130

- N130—The sequence number.
- Z0—Rapids the cutter to the buffer height.

N140

- N140—The sequence number.
- G01—Puts the machine in feedrate mode.
- Z−.62—Feeds the cutter to proper milling depth.

N150

- N150—The sequence number.
- G17—Selects the X/Y plane. It is used to reestablish movement along the X/Y axes canceled when the G18 was issued in block N110.
- Y0—The coordinate for the feedrate move from location #4 to location #5.

N160

- N160—The sequence number.
- G40—Cancels the compensation.
- X7—Moves the cutter from location #5 to location #6. This is a ramp off move.

N170

- N170—The sequence number.
- G00—Puts the machine in rapid traverse mode.
- G49—Cancels the active tool offset. Note that the compensation was canceled first.
- Z0—Retracts the spindle.

N180

- N180—The sequence number.
- X−1 y−.25—The coordinates to rapid from location #6 to location #1.

N190

- N190—The sequence number.
- G45 H02—Calls up the tool information in register #2 for use with the tool.

Blocks N200 on repeat blocks N070 through N170. The coolant is turned off in block N300, and the spindle is turned off in block N310.

SPECIAL CONSIDERATIONS

Figure 10–7 illustrates the correct method for turning compensation on or off when machining an inside pocket. Point B must be a minimum of one cutter radius away from the corner of the pocket. If point C were programmed as the ramp on move, the cutter would cut into the corner as in Figure 10–8. The direction of the cut depends on whether the X or Y axis is programmed as the first move following the G41.

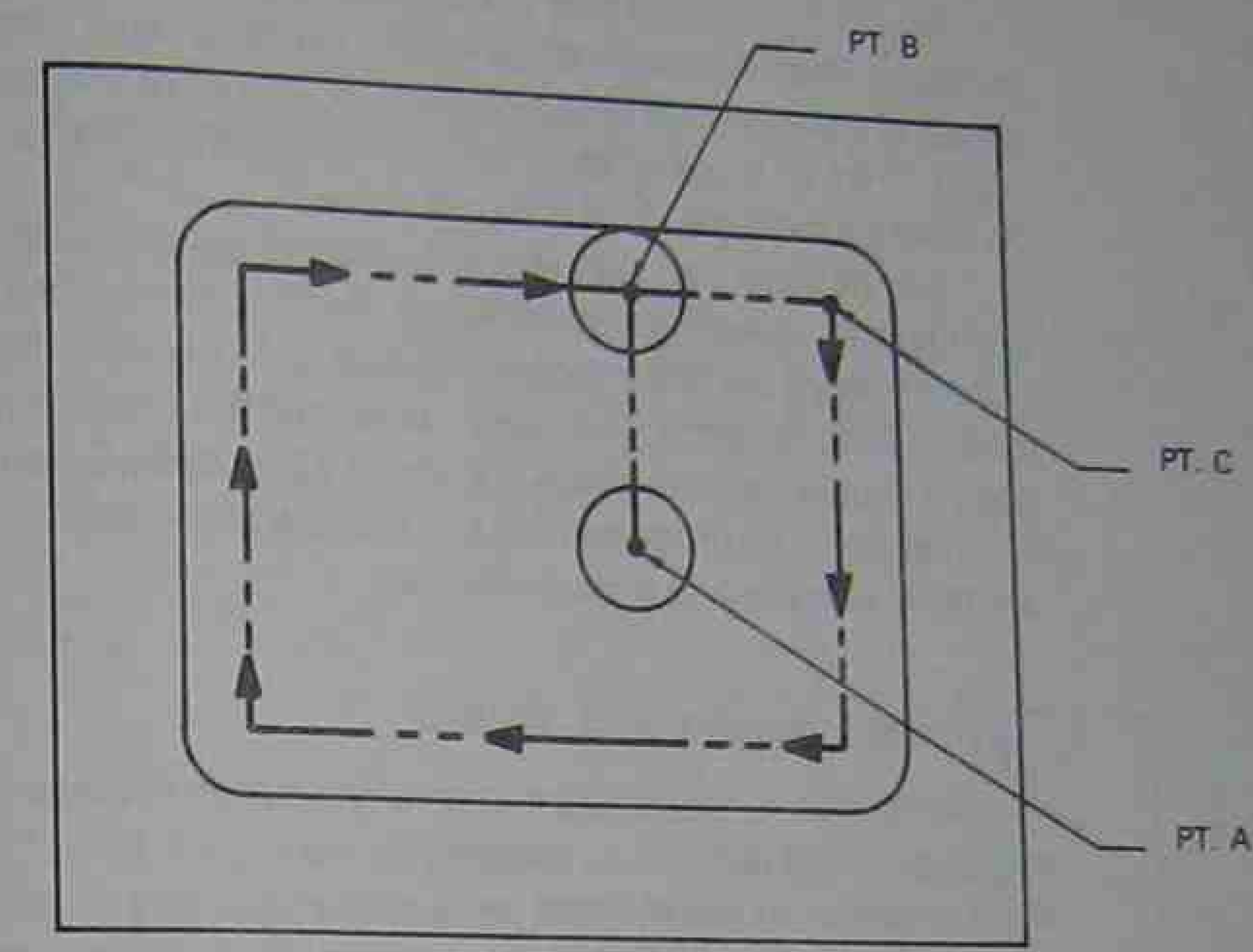


FIGURE 10–7
Turning compensation on or off when machining an inside pocket to prevent cutting into the corner

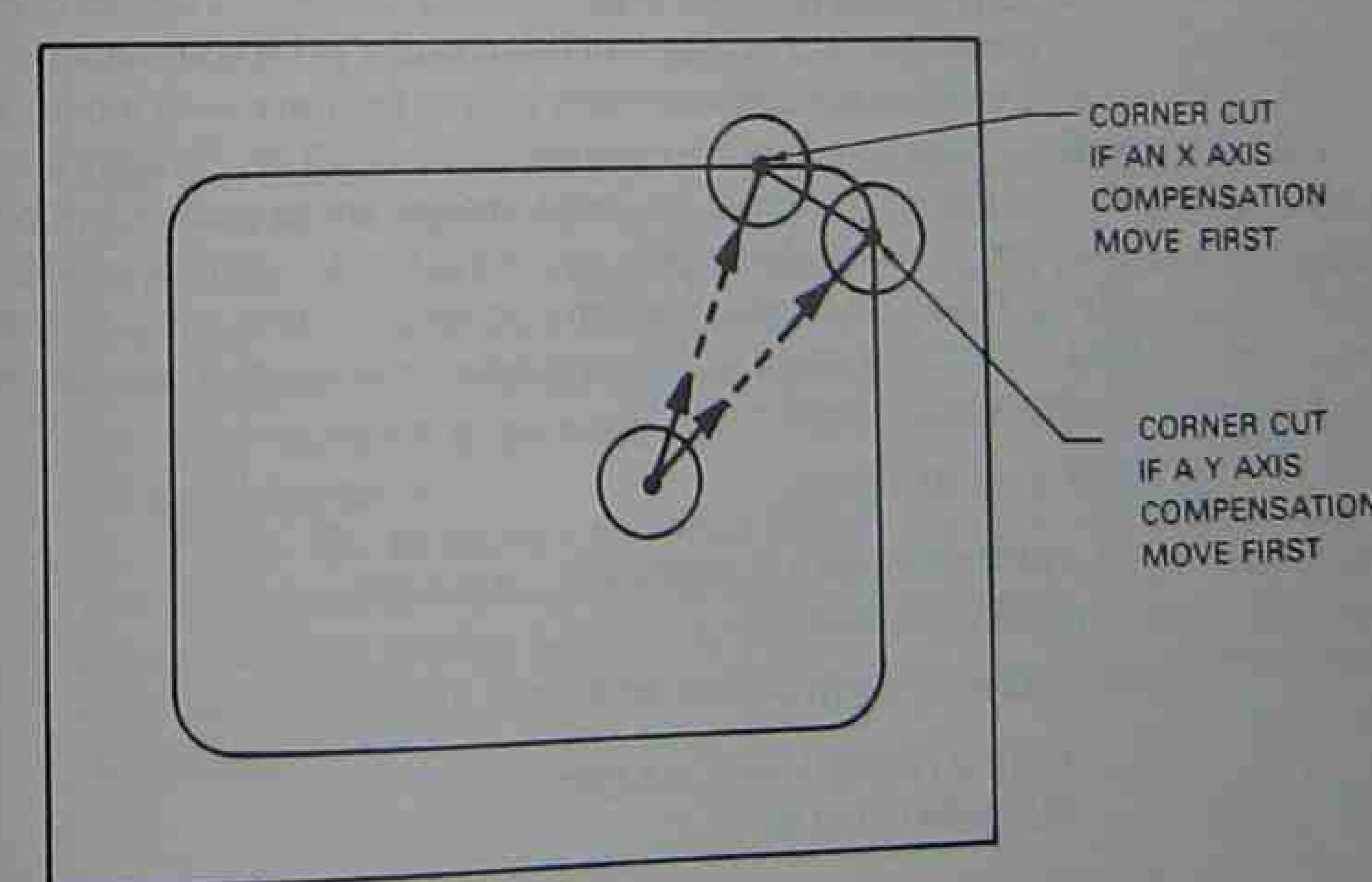


FIGURE 10–8

Figure 10–9 illustrates the precautions necessary when ramping on or off an angle. Point A should not be used for a ramp on or ramp off move since the corner of the angle will be cut off the part, and there may also be damage to the cutter. Point C, or some other point roughly perpendicular to the angle, should be used for the ramp on or ramp off move.

Two different methods of positioning are used for cutter comp with respect to angles, as demonstrated in Figure 10–10. On older CNC machinery, the machine positions the cutter tangent to point A. A G code is then used to initiate the rotation from Y1 to Y2. On newer machinery, the cutter is positioned directly to point P, tangent to both line A and line Y. No special G codes are necessary in this instance. The programming manual for a particular machine will tell the programmer whether a G code is required.

Approach Angles and Vectors

Another factor to consider when using cutter diameter compensation is the approach angle used when ramping on. As Figure 10–11 illustrates, there are three possible angles that can be used during a ramp–on move: 90 degrees to the next cut, less than 90 but greater than 45 degrees to the next cut and less than 45 degrees to the next cut. Some controllers will accept any of these approach angles, others will not. If an unacceptable approach angle is used, the cutter will move to the programmed coordinates, but the cutter compensation will not take place. When programming a number of controllers, or if the NC program will be run on more than one type of controller, it is best to use a 90 degree approach angle to eliminate problems when ramping on cutter comp.

Sometimes, a controller requires a vector to be commanded with the G41 or G42 to orient the cutter correctly prior to the ramp–on move. Technically, a vector is a geometric entity that has both magnitude (length) and direction. In NC programming, vectors are simply mathematical arrows that point the cutter in a given direction. To utilize a vector the I and J addresses are used. Figure 10–12 illustrates some cutter comp vectors. If cutter comp was to be initiated from point A (Figure 10–12) and ramped on to point B, the following program blocks would be used.

For Figure 10–12 (A):

```
N010 G17 G42 I-.5 J-.866 D21
N020 G00 X6.0 Y-.5
```

For Figure 10–12 (B):

```
N010 G17 G42 I-.866 J-.5 D21
N020 G00 X6.0 Y-.5
```

For Figure 10–12 (C):

```
N010 G17 G42 I-1.0 J0 D21
N020 G00 X6.0 Y-.5
```

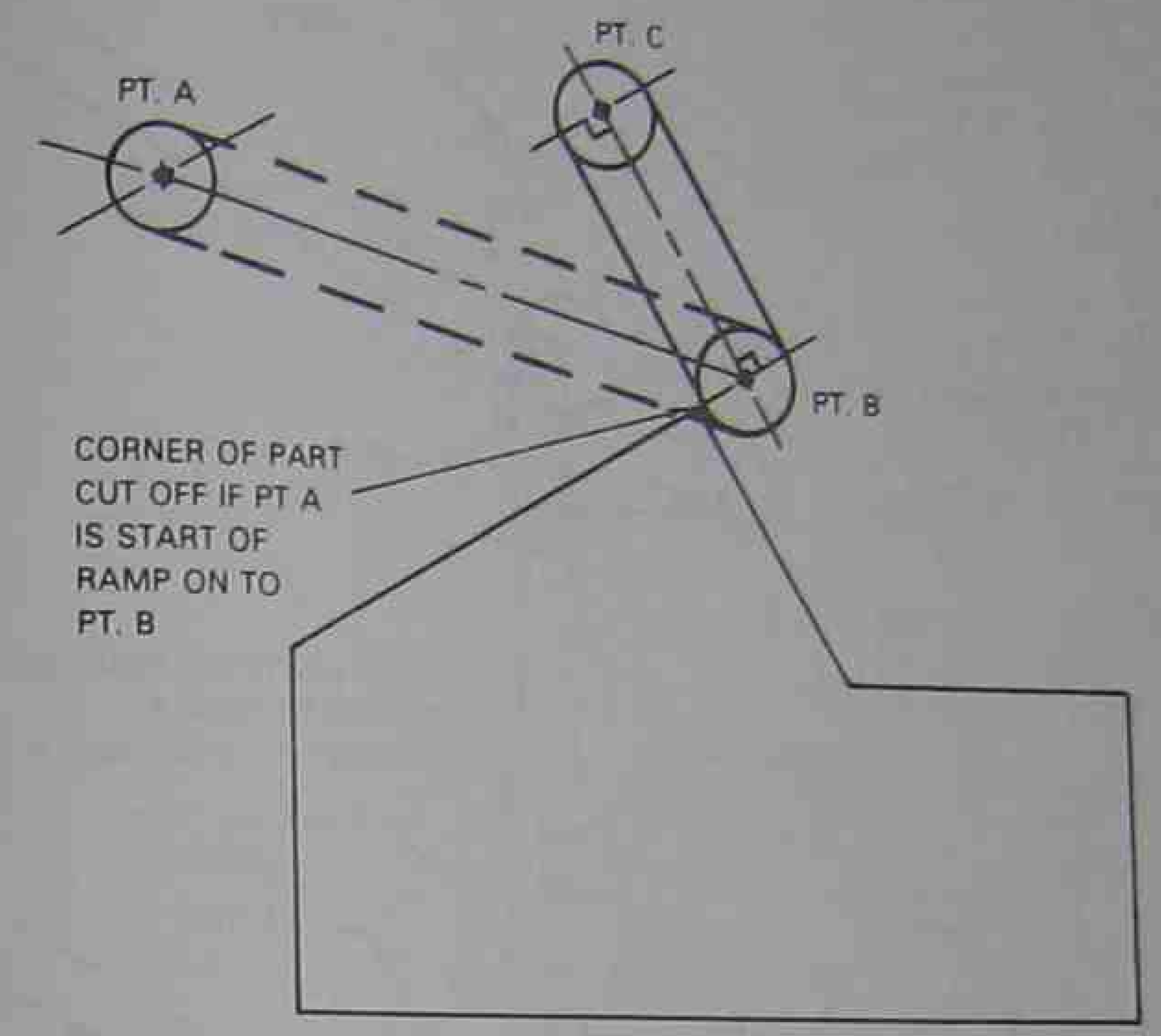



FIGURE 10–9
Ramping on or off an angle

Note that in each of these cases I is the X-axis component, and J is the Y-axis component of a vector 1.0 inch long. This is called a unit vector. In Figure 10–12 (A) the approach angle is 30 degrees, therefore, I equals the sine of 30 degrees, while J equals the cosine. In Figure 10–12 (B) the approach angle is 60 degrees. I equals the sine of 60 degrees, while J equals the cosine of 60 degrees. Since the approach angle in Figure 10–12 (C) is 90 degrees, I simply equals 1.0 and J equals zero.

Figure 10–13 shows a part to be milled using cutter diameter compensation. A program to mill the part is given in Figure 10–14. It is assumed that the part is clamped through two already existing holes. With the information given thus far, the student should be able to follow this program without further explanation.

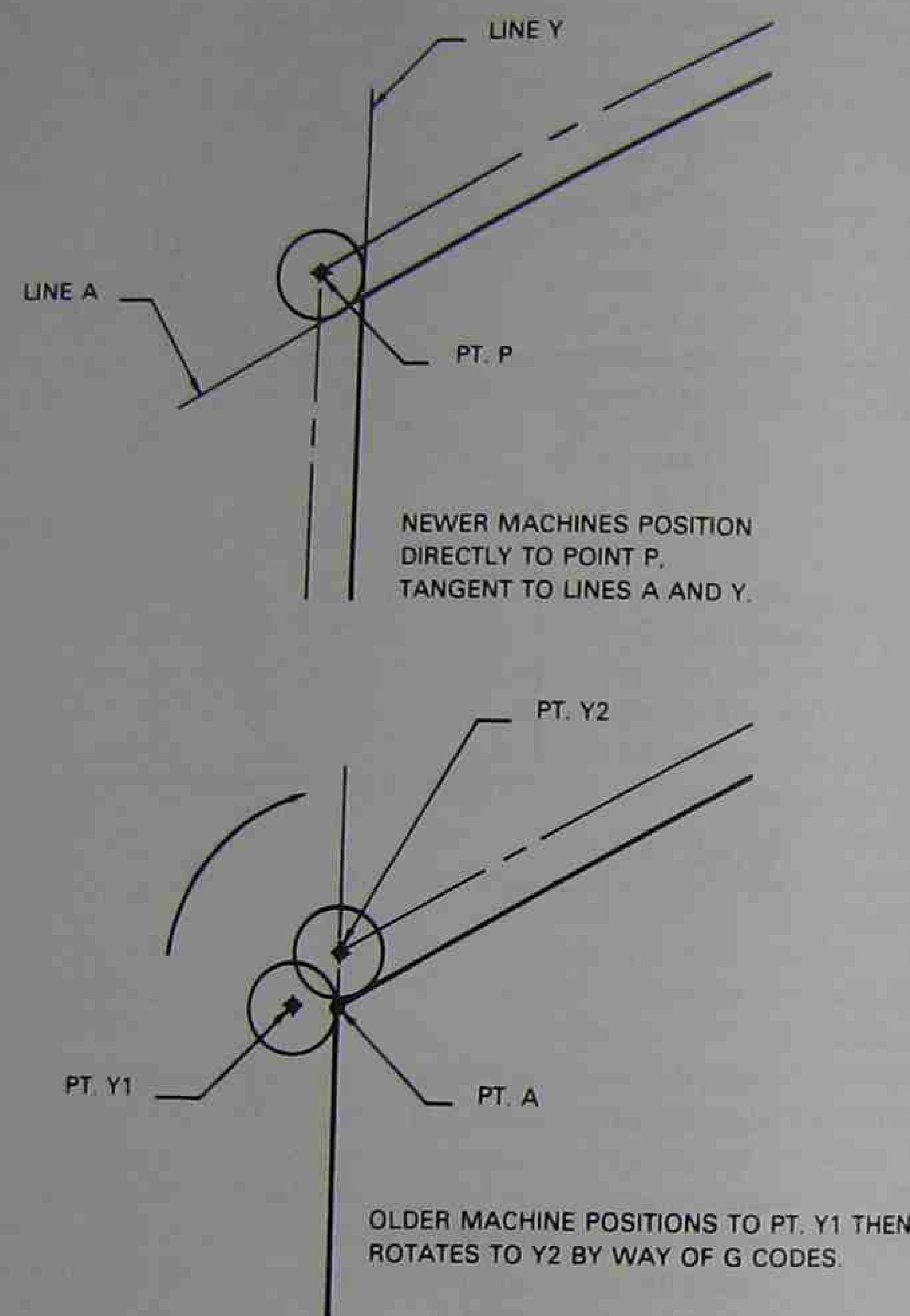


FIGURE 10–10
Cutter diameter compensation of angles

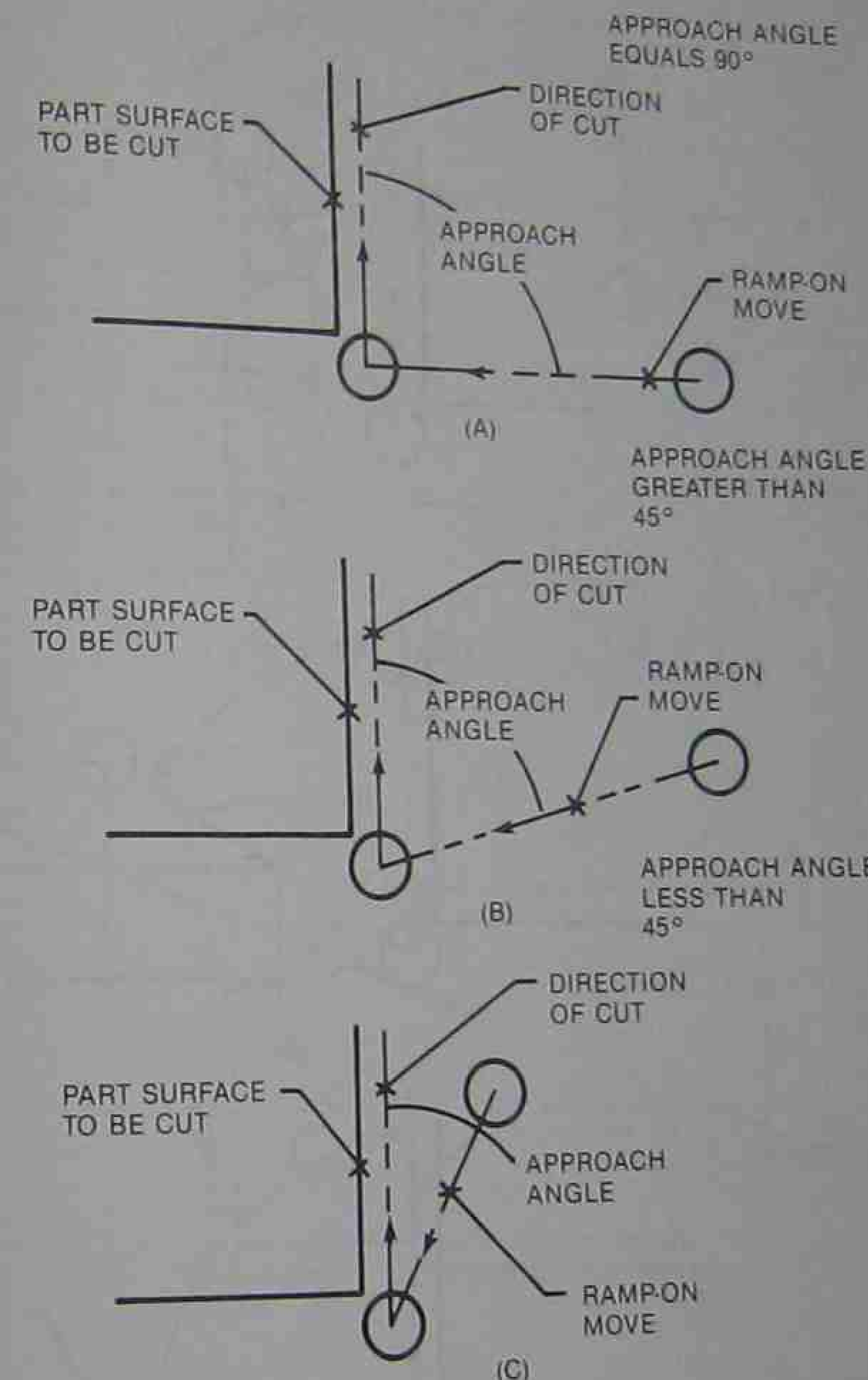


FIGURE 10–11
Cutter compensation approach angles

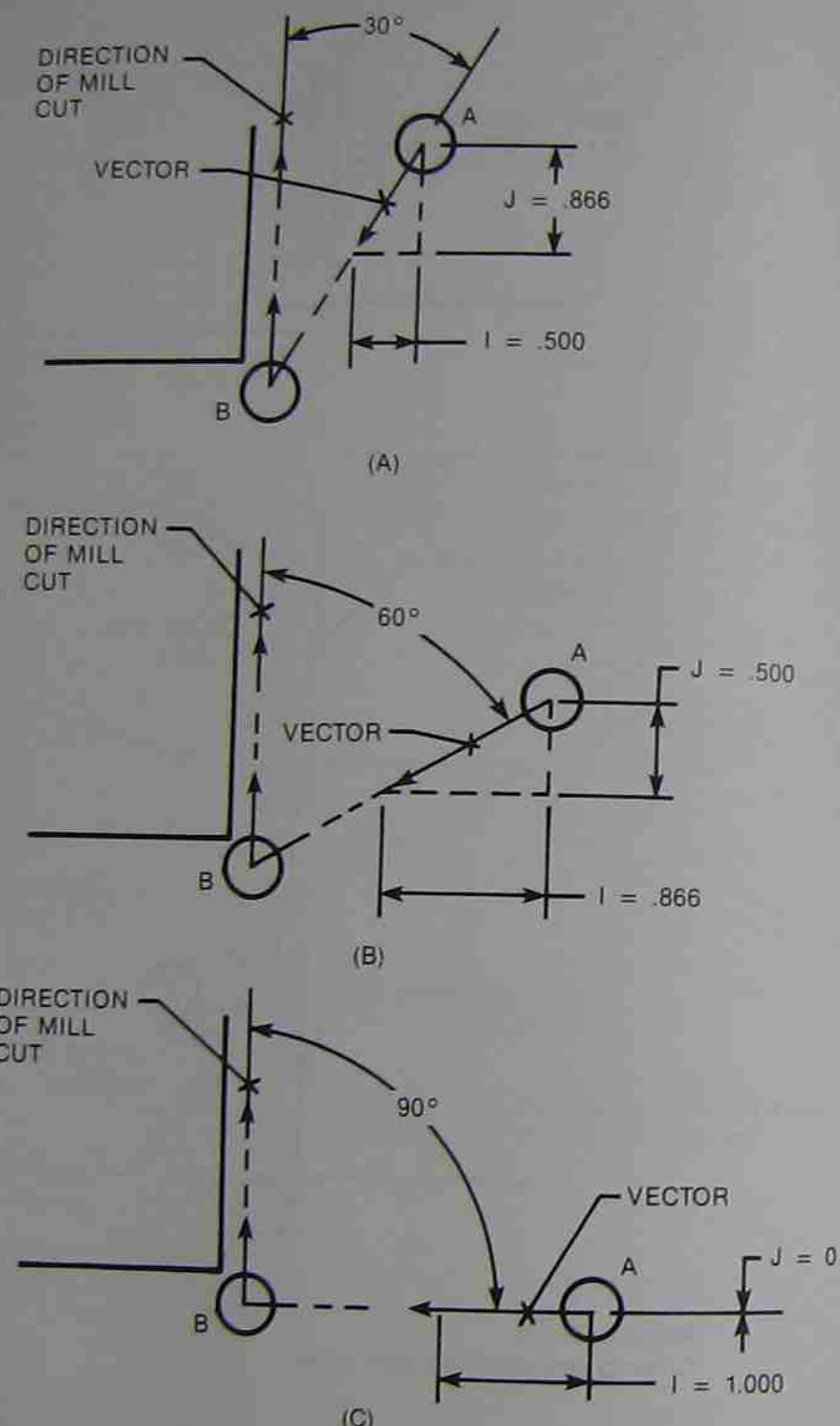


FIGURE 10–12
Cutter compensation vectors

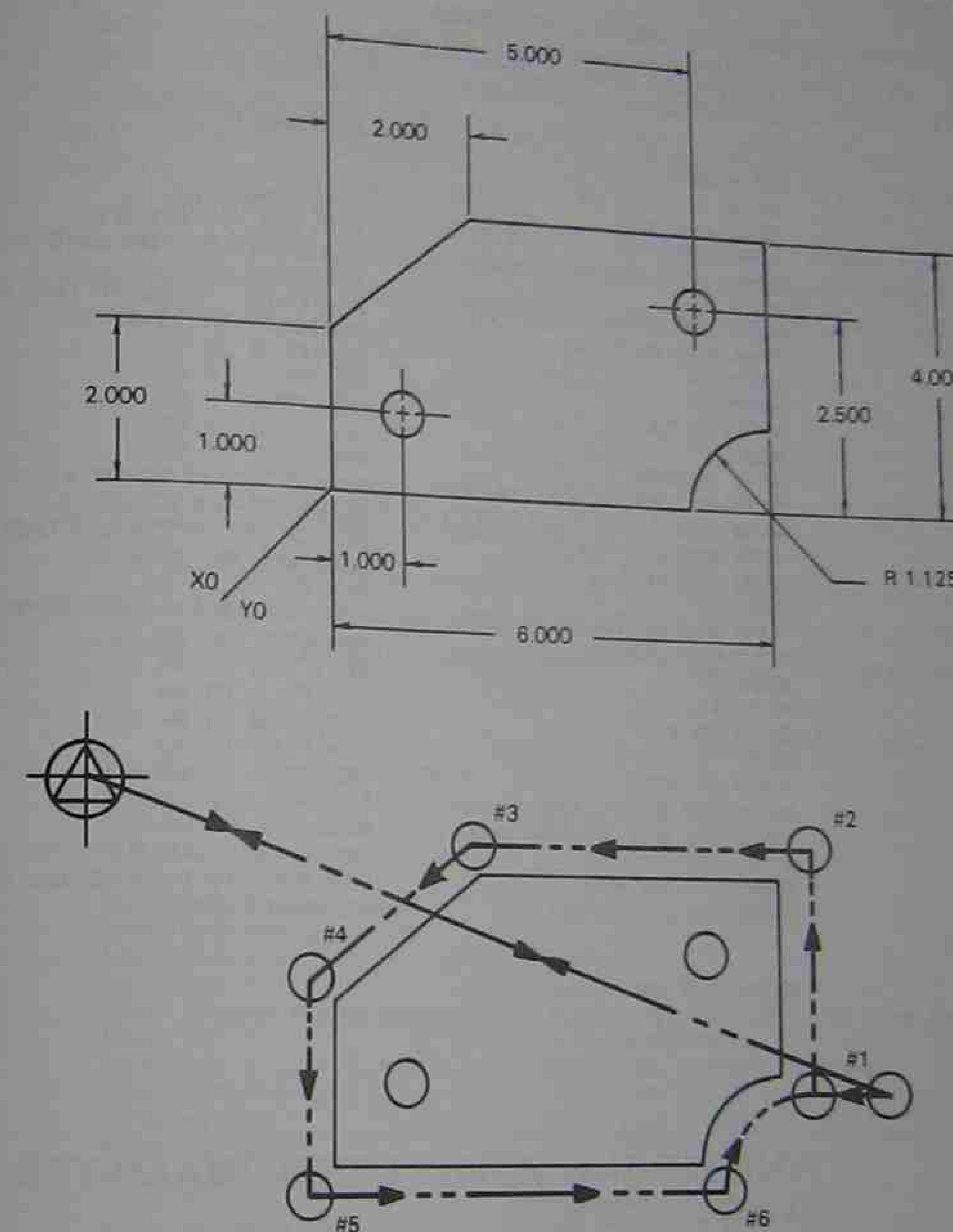


FIGURE 10–13
Part drawing and cutter path

```
XO/YO = LOWER LEFT CORNER
TOOLS:   1.000 IN. 4 FLUTE END MILL
CLEARANCE ABOVE CLAMPS:   3.000 MIN.
BUFFER ZONE:   .100 IN.

N010 G00 G40 G70 G90             REM:SAFETY LINE
N020 G10 H01 X1.02 Z3
N030 G10 H02 X1 Z3
N040 M06 T01                     REM:1 IN. E/M
N050 G45 H01                     REM:TOOL #1 OFFSET
N060 X7 Y.875 S400 M03           REM:POSITION FOR SAFE RAMP-ON
N070 Z0                          REM:RAPID TO Z0
N080 G01 Z-.89 F6.8 M08          REM:FEED Z AXIS TO DEPTH
N090 G17 G42 X6                  REM:RAMP ON TO #1
N100 Y4                          REM:FEED TO #2
N110 X2                          REM:FEED TO #3
N120 X0 Y2                       REM:FEED TO #4
N130 Y0                          REM:FEED TO #5
N140 X4.875                      REM:FEED TO #6
N150 G02 X6 Y1.125 I6 J0         REM:ARC FROM #6 TO #1
N160 G00 G40 X7                  REM:RAMP-OFF
N170 G00 G49 Z0                  REM:CANCEL OFFSET/RETRCT SPNDL
N180 G45 H02                     REM:TOOL OFFSETS #2
N190 Z0                          REM:RAPID TO Z0
N200 G01 Z-.89 F6.8              REM:FEED Z AXIS TO DEPTH
N210 G17 G42 X6                  REM:RAMP-ON TO #1
N220 Y4                          REM:FEED TO #2
N230 X2                          REM:FEED TO #3
N240 X0 Y2                       REM:FEED TO #4
N250 Y0                          REM:FEED TO #5
N260 X4.875                      REM:FEED TO #6
N270 G02 X6 Y1.125 I6 J0         REM:ARC FROM #6 TO #1
N280 G00 G40 X7.500 M09          REM:RAMP-OFF
N290 G00 G49 Z0                  REM:CANCEL OFFSET/RETRCT SPNDL
N300 X-8 Y6 M05                  REM:MOVE TO PARK/SPNDL OFF
N310 M30                         REM:END PROGRAM
```

FIGURE 10–14

FINE TUNING WITH CUTTER DIAMETER COMPENSATION

Up to this point, cutter diameter compensation has been used to program the part line; the program coordinates have matched the part dimensions. Another way cutter comp is employed is to fine tune the cutter path. In this type of programming, the part is programmed using the parallel path method used in Chapters 6, 7, and 9. Cutter comp is used to compensate for the difference between the programmed and actual cutter diameter. For example, if a program is written for a .500 diameter end mill, but a resharpened end mill measuring .490 diameter is used, the .020 diameter difference can be compensated for by using cutter comp.

In the fine tune method, cutter comp is usually used to compensate for a cutter which is smaller than the programmed diameter. When using the part line method exactly the opposite is the case. Cutter comp is used to compensate for a cutter which is larger than the zero diameter cutter programmed (the part line). For this reason it is necessary to use a minus (−) value in the cutter comp register when using the fine tune method.

Figure 10 – 15 is a word address program for the part in Figure 10 – 13, illustrating the fine tune method. Note that allowance is once again being made for the cutter radius. The cutter diameter compensation allows reground, undersize cutters to be used.

SPECIAL FANUC SECTION

Figure 10–16 contains a program to mill the part in Figure 10–8 using half the cutter diameter in the cutter comp register. Figure 10–17 contains a program to mill the part in Figure 10–13 using the fine tune method. The programs are

```
XO/YO = LOWER LEFT CORNER
TOOLS:   1.000 IN. 4 FLUTE END MILL
CLEARANCE ABOVE CLAMPS:   3.000 MIN.
BUFFER ZONE:   .100 IN.

N010 G00 G40 G70 G90             REM:SAFETY LINE
N020 G10 H01 X.990 Z3
N030 M06 T01                     REM:1 IN. E/M
N040 G45 H01                     REM:TOOL #1 OFFSET
N050 X7.500 Y.875 S400 M03       REM:POSITION FOR SAFE RAMP-ON
N060 Z0                          REM:RAPID TO Z0
N070 G01 Z-.89 F6.8 M08          REM:FEED Z AXIS TO DEPTH
N080 G17 G42 X6.510              REM:RAMP ON TO #1
N090 Y4.510                      REM:FEED TO #2
N100 X1.7829                     REM:FEED TO #3
N110 X-.510 Y2.2171              REM:FEED TO #4
N120 Y-.510                      REM:FEED TO #5
N130 X5.3850                     REM:FEED TO #6
N140 G02 X6.500 Y.615 I6 J0      REM:ARC FROM #6 TO #1
N150 G01 Y4.500                  REM:FEED TO #2
N160 X1.7929                     REM:FEED TO #3
N170 X-.500 Y2.2071              REM:FEED TO #4
N180 Y-.500                      REM:FEED TO #5
N190 X5.375                      REM:FEED TO #6
N200 G02 X6.500 Y.626 I6 J0      REM:ARC FROM #6 TO #1
N210 G00 G40 X7.500 M09          REM:RAMP-OFF
N220 G00 G49 Z0                  REM:CANCEL OFFSET/RETRCT SPNDL
N230 X-8 Y6 M05                  REM:MOVE TO PART/SPNDL OFF
N240 M30                         REM:END PROGRAM
```

FIGURE 10–15

```
%
01006
(--------------------------------------------------)
(X0/Y0 - LOWER LEFT CORNER OF PART)
(Z0 - .100 ABOVE TOP OF PART)
(MIN. CLEARANCE ABOVE CLAMPS - 3.0 INCHEs)
(--------------------------------------------------)
()
(MOVE TO TOOL CHANGE POSITION)
(ORIENT SPINDLE FOR TOOL CHANGE)
N001G90G17G40G80
N101G91G30X0Y0Z0M19
N102T01M06
()
(CALL UP WORK COORDINATE - PUT TOOL 2 IN STANDBY)
N103G90G00G54S2500T02B0
(MOVE TO START POSITION - PICK UP OFFSET ON Z MOVE)
N104X7.5Y.755M03
N105G45Z0H01M08
N106G01Z-.62F12.8
(INITIATE AND RAMP ON CRO - USES REGISTER D11)
N107G17G41X0.D11
N108Y4.
N109G00Z3.
N110X6.
N111Z0.
N112G01Z-.62
N113Y0.
(CANCEL AND RAMP OFF CRO)
N114G40X7.
N115G00Z3.
(MOVE TO START OF 2ND CUT )
N116X-1.Y-.25
N117Z0.
N118G01Z-.62F12.8
(INITIATE AND RAMP ON CRO - USES REGISTER D12)
N119G17G41X0D12
N120Y4.
N121G00Z3.
N122X6.
N123Z0.
N124G01Z-.62
N125Y0.
(CANCEL AND RAMP OFF CRO)
N126G40X7.
(CANCEL OFFSET AND RETRACT Z BY RETURNING TO TOOL CHANGE)
N127G91G30Z0M09
N128G30X0Y0Z0M19
N129M30
%
```

FIGURE 10-16


```
%
01013
(--------------------------------------------------)
(X0/Y0 - LOWER LEFT CORNER OF PART)
(Z0 - .100 ABOVE TOP OF PART)
(--------------------------------------------------)
()
N001G90G17G40G80
N101G91G30Z0Y0Z0M19
N102T01M06
N103G90G00G54S400T02B0
N104X7.5Y.875M03
N105G45Z0H01M08
N106G01Z-.89F6.8
(RAMP ON CRO - USES REGISTER D11)
N107G17G42X6.51D11
(BEGIN ROUGH PASS - LEAVE .01 STK. TO FINISH)
N108Y4.51
N109X1.7829
N110X-.51Y2.2171
N111Y-.51
N112X5.385
N113G02X6.5Y.615I.615J0
N114G01Y4.5
(BEGIN FINISH PASSS)
N115X1.7929
N116X-.5Y2.2071
N117Y-.5
N118X5.375
N119G02X6.5Y.625I.625J0
(CANCEL AND RAMP OFF CRO)
N120G00G40X7.5
(CANCEL OFFSETS BY RETURNING TO TOOL CHANGE)
N121G91G30Z0M09
N122G30X0Y0Z0M19
N123M30
%
```

FIGURE 10-17

written for a Hitachi Seiki HC-500 four-axis machining center with a Fanuc 11M control. It will be assumed that B0 positions the rotary table to the proper position to machine the part. The B axis (rotary motion around the Y axis) is commanded to B0 at the start of the program and left there throughout the program cycle. There are several important differences between this machine and the text examples that need to be understood.

Fanuc controls utilize a cutter comp called cutter radius offset (CRO). The radius value of the cutter is entered in a register rather than the cutter diameter. If using the fine tune method, it is the difference between the programmed cutter radius and the actual cutter radius that is entered in the register. A "D" address is used to specify the register used.

Like many horizontal machining centers, the HC-500 uses a preset tool change position. On this machine, the spindle must be commanded to the tool change position. The spindle must also be oriented to allow the tool changer to properly change the tools. The command used to accomplish this is G91G30Z0Y0Z0M19. The M19 orients the spindle. The length offset is canceled by returning the Z axis to the tool change position. Once this is done, X and Y are returned to tool change as follows:

```
G91G30Z0M09
G30X0Y0M19
```

After changing tools, a T02 command is included on the line following the T01M06 command to place the next tool in the standby position, ready to be changed at the next tool change command.

This machine also uses work coordinates. The G54 command follows the tool change command line to call up the work coordinate. The work coordinate values have been manually entered.

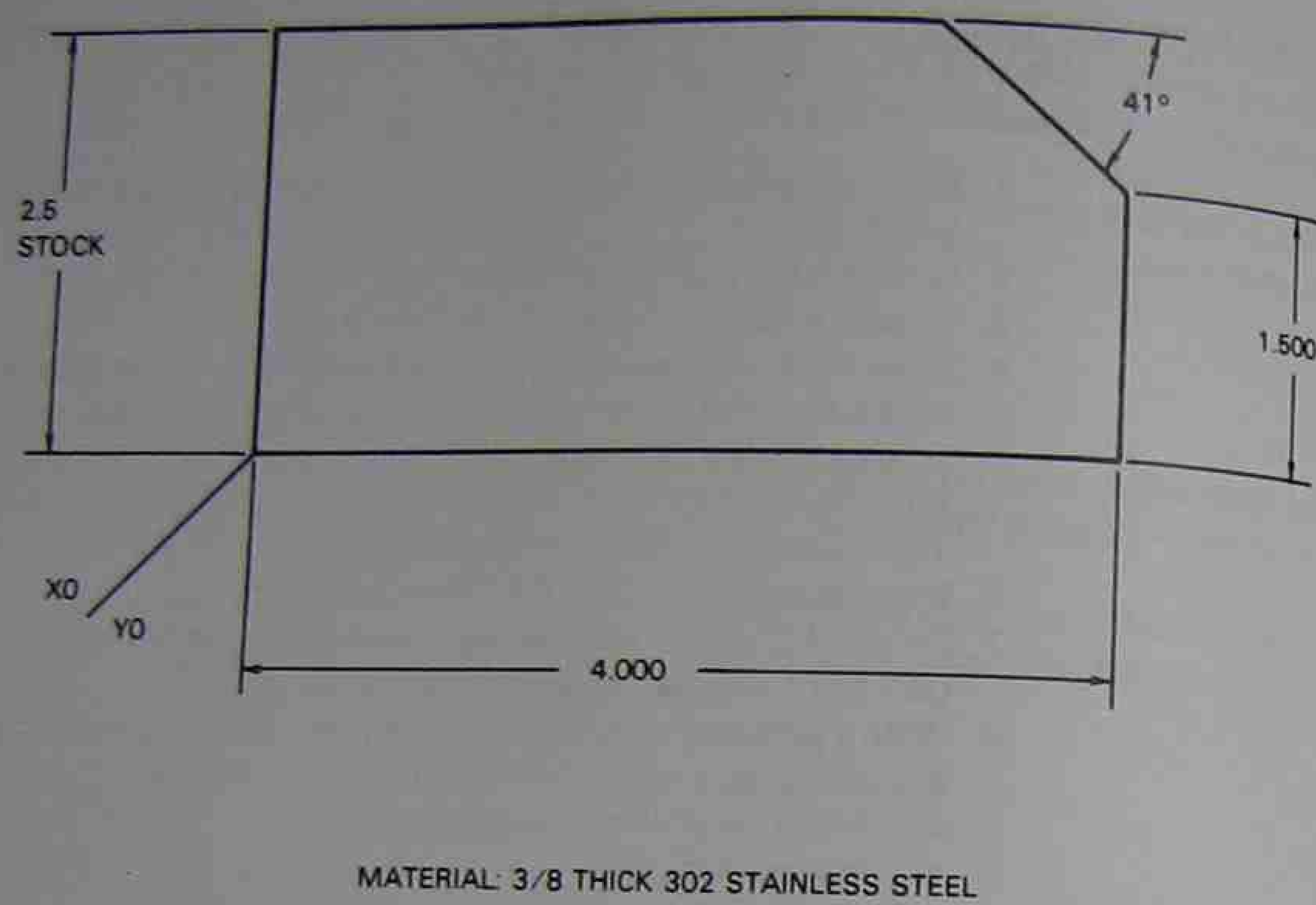
SUMMARY

The important concepts presented in this chapter are:

- Cutter diameter compensation is the automatic calculation of the cutter path by the machine control unit, based on the part line and cutter information contained in the program.
- Cutter diameter compensation is instituted and canceled through use of the codes G40, G41, and G42. G41 is cutter compensation left, G42 is cutter compensation right, and G40 is cutter compensation cancel.
- The "ramp on" move is the initial compensation of the cutter. The compensation occurs 90 degrees to the next axis movement following the G41 or G42. Care must be taken with the spindle position prior to the ramp on move to avoid cutting the part in the wrong area.
- The "ramp off" move is the opposite operation. Ramp off will occur 90 degrees to the next axis movement following a G40. The compensation will be completely eliminated by the end of this move.

REVIEW QUESTIONS

1. What is cutter diameter compensation? How does it differ from tool length offset?
2. What is a ramp on move? When does it occur?
3. What is a ramp off move? When does it occur?
4. Draw a sketch illustrating the proper technique for ramping on, assuming the machine does not have the capability to compensate in two axes simultaneously. Draw a sketch illustrating an improper ramp on.
5. What cautions must be observed when instituting cutter compensation inside a pocket? When milling angles?
6. What do the codes G40, G41, and G42 do?
7. Do all CNC machines directly position the cutter with respect to an angle? If not, how is the rotation accomplished?
8. Write a program to mill the part in Figure 10–18, using a roughing and a finishing pass with a 1.000-inch-diameter end mill:
   a. In Machinist Shop Language.
   b. In word address.



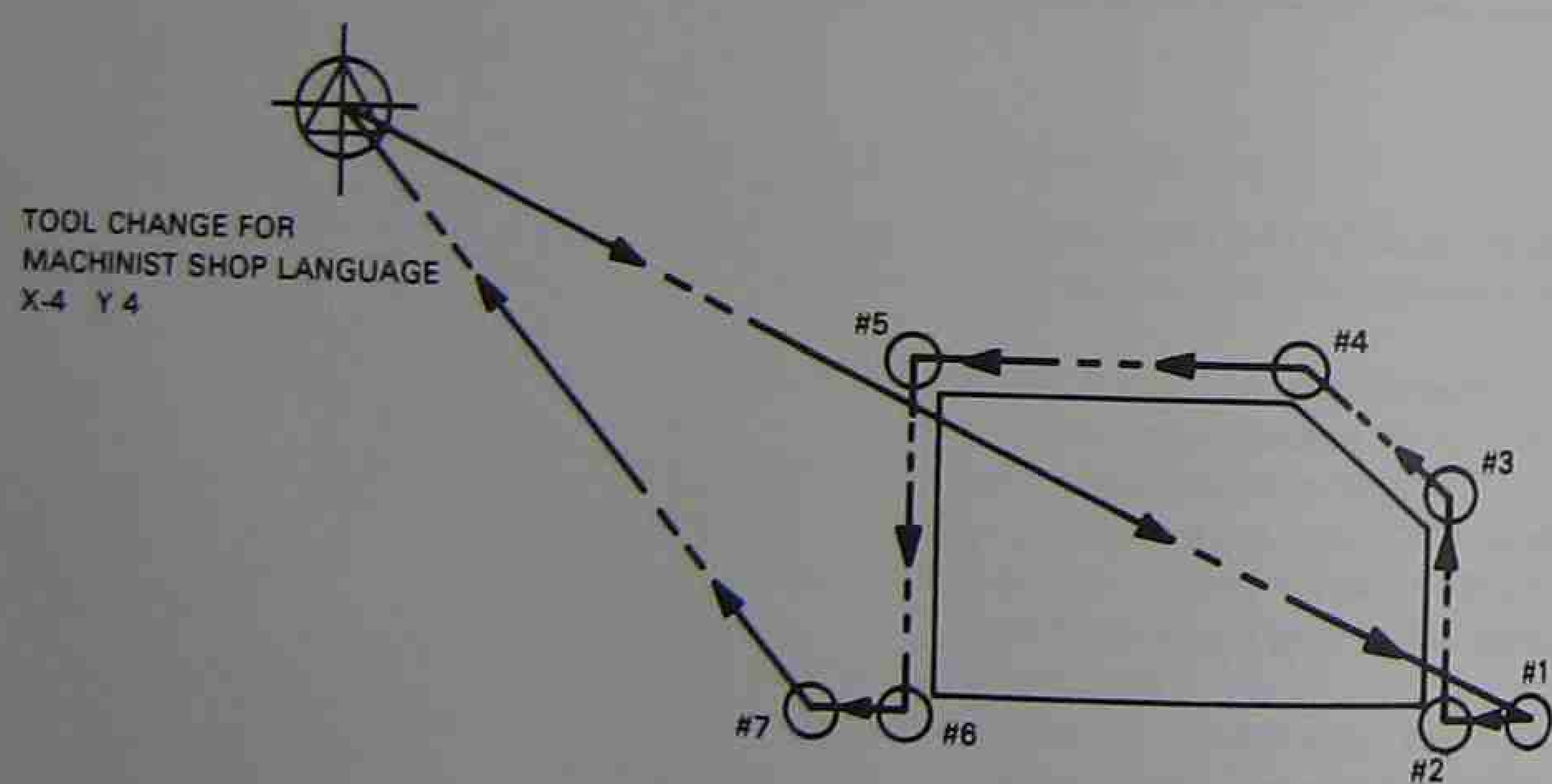


FIGURE 10–18
Part drawing for review question #8

CHAPTER 11 Do Loops and Subroutines

OBJECTIVES Upon completion of this chapter, you will be able to:

- Describe a do loop.
- Describe a subroutine.
- Describe nested loops.
- Write simple programs in word address and Machinist Shop Language using do loops, subroutines, and nested loops.

DO LOOPS

Figure 11–1 shows a part with a series of holes to be drilled, equally spaced. If an operation is to be repeated over a number of equal steps, it may be programmed in what is referred to as a do loop. In a *do loop*, the MCU is instructed to repeat an operation (in this case, drill a hole five times) rather than programmed for five separate hole locations.

Machinist Shop Language

The format for a do loop in Machinist Shop Language is:

1. DO n
2. X/Y/Z I
3. END

Where DO is the command to repeat the operation that follows, n is the number of times the operation is to be repeated, X/Y/Z is the coordinate information for the loop, I specifies incremental positioning, and END signals the end of the do loop.

Figure 11–2 is a program written in Machinist Shop Language to drill the five holes in Figure 11–1 using a do loop. The tool change location is at X−2.000, Y2.000.

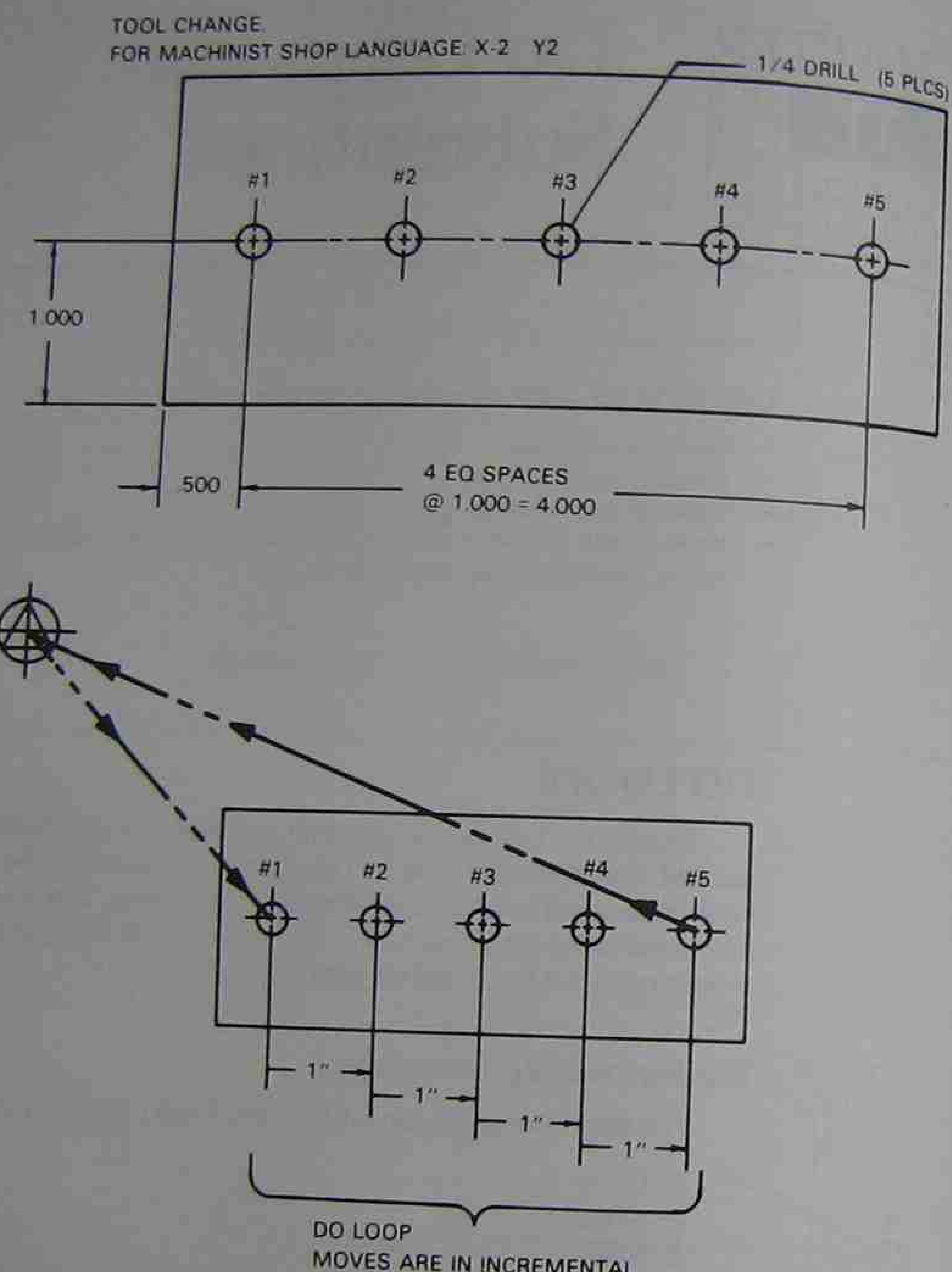


FIGURE 11–1


```
XO/YO = LOWER LEFT CORNER
TOOL CHANGE = X-2 Y2
TOOLS:  #3 C'DRILL, 1/4 DRILL
CLEARANCE OVER CLAMPS = 2.500 IN. MIN.

 1 TOOL 1001
 2 Z4
 3 TOOL 1001         A
 4 Z2.5
 5 X-2 Y2 Z0         A
 6 TOOL 1            RA
 7 V20 10.5               REM:C'DRILL, 3500 RPM
 8 V21 .1                 REM:FEEDRATE FOR DRILL CYCLE
 9 G81               A    REM:DEFINE BUFFER ZONE
10 X.5 Y1 Z-.1620    RA   REM:INITIATE DRILL CYCLE
11 DO 4                   REM:C'DRILL HOLE #1
12 X1                RI   REM:INITIATE LOOP
13 END                    REM:INCREMENTAL COOR. IN LOOP
14 G80                    REM:END LOOP
15 TOOL 0                 REM:CANCEL DRILL CYCLE
16 Z0                RA   REM:CANCEL TOOL OFFSET
17 X-2 Y2            RA   REM:RETRACT SPNDL
18 TOOL 2                 REM:RAPID TO TCH
19 V20 21                 REM:1/4 DRILL, 3500 RPM
20 V21 .1                 REM:DRILL CYCLE FEEDRATE
21 G81                    REM:DEFINE BUFFER
22 X.5 Y1 Z-.375     RA   REM:INITIATE DRILL CYCLE
23 DO 4                   REM:DRILL HOLE #1
24 X1                RI   REM:INITIATE LOOP
25 END                    REM:INCREMENTAL COOR. IN LOOP
26 G80                    REM:END LOOP
27 TOOL 0                 REM:CANCEL DRILL CYCLE
28 Z0                RA   REM:CANCEL TOOL OFFSET
29 X-2 Y2            RA   REM:RETRACT QUILL
30 END                    REM:RAPID TO TCH
                          REM:END OF PRGM
```

FIGURE 11–2
Do loop program for part in Figure 11–1, Machinist Shop Language

PROGRAM EXPLANATION

(Refer to Figure 11–2.)

EVENTS 1–4

- Assign tool information to the computer.

EVENTS 5–6

- Send the spindle to tool change and assign the tool length offset for tool #1 (as presented in Chapter 7).

EVENT 7
- V20 10.5—Assigns 10.5 as the feedrate for canned cycle operation to variable register 20. The MCU uses the value programmed in the V20 register for the Z-axis feedrate with a G81.

EVENT 8
- V21 .1—Assigns a buffer zone of .100 inch to be used for Z-axis moves with the G81 cycle.
- A—Specifies absolute positioning.

EVENT 9
- G81—Initiates the canned drilling cycle.

EVENT 10
- X.5 Y1—Coordinates of hole #1.
- Z–.1620—Z-axis depth for center drilling. The depth of the center drill into the part is .0620 inch. Since V21 defined a .100-inch rapid level, .100 inch must be added to .062 to arrive at the required Z axis coordinate.

EVENT 11
- DO 4—Tells the computer to repeat four times the operation defined in the events that follow.

EVENT 12
- X1—Incremental distance from one hole to the next on the X axis. The holes are in line, so no movement along the Y axis is necessary.
- R—Specifies that the moves from hole to hole be made at rapid traverse. Since this is a drilling operation, there is no need to move between holes at feedrate.
- I—Specifies incremental positioning. This is an example of the use of both absolute and incremental positioning in a program. The X1 must be an incremental dimension because each hole to be drilled is referenced to the previous one, not to the part X0/Y0 point.

EVENT 13
- END—Specifies the end of the do loop. END has three uses: end of program, end of do loop, and end of subroutine. The machine will act four times on the information contained between the DO statement and END. Since the G81 will be active from the first hole, it is not necessary to include it in the do loop.

EVENT 14
- G80—Cancels the active G81.

EVENT 15
- TOOL 0—Cancels the active tool offset.

EVENT 16
- Z0—Retracts the spindle.

EVENT 17
- X–2 Y2—Coordinates of tool change location.

EVENT 18
- TOOL 2—Calls up the offset for the 1/4-inch drill.

EVENTS 19–21
- Initiate the drilling cycle and assign the feedrate and buffer zone.

EVENT 22
- X/Y coordinates—Position the machine to hole #1.
- Z coordinate—For drilling, determined by adding .3 times the drill diameter to the part thickness of .250 and the .100-inch buffer zone. (Note: An additional .050 was added to this result to allow for tooling and part tolerance.)

Events 23 through 29 duplicate events 11 through 17. Event 30 signals the end of the program.

Word Address Format

The same principle is now demonstrated using word address. Naturally there is a G code to institute a do loop. The format for a do loop in word address is:

1. G51 Nn
2. X/Y/Z
3. G50

Where G51 signals the start of a do loop, N is the address and n is the number of times an operation is to be repeated, X/Y/Z is the program information contained in the loop, and G50 signals the end of the do loop.

It should be noted here that the codes for do loops vary from one controller to another. The programming manual will need to be consulted for the proper codes. The coding used here is a type of General Numerics format used on Fanuc controllers also.

PROGRAM EXPLANATION

(Refer to Figure 11–3.)

N010

- Safety line to cancel any active G codes and bring the spindle to home position.

N020–N030

- Assigns tool information to tool length registers.

N040

- M06 T1—Initiates an automatic tool change, selecting tool #1 from the storage magazine.

```
XO/YO = LOWER LEFT CORNER
TOOLS:   TOOL #1 = #3 C'DRILL.
         TOOL #2 = 1/4 DRILL.
CLEARANCE OVER CLAMPS = 2.500 IN. MIN.

N010 G00 G40 G49 G80 G70 G90 X0 Y0 Z0    REM:SAFTEY LINE
N020 G10 H01 Z4                          REM:TOOL 1 OFFSETS
N030 G10 H02 Z2.5                        REM:TOOL 2 OFFSETS
N040 M06 T1                              REM:ATC #3 C'DRILL
N050 G45 H01                             REM:OFFSET 1
N060 S3500 F10.5 M03                     REM:SPNDL ON
N070 G81 G99 X.5 Y1 Z-.162 R0 M08        REM:C'DRILL HOLE #1,
COOL.ON
N080 G51 N4                              REM:INITIATE LOOP
N090 G91 X1                              REM:INC. COOR. IN LOOP
N100 G50                                 REM:END OF LOOP
N110 G80 G90 G49 Z0 M09                  REM:RETRACT Z, COOL. OFF
N120 M06 T2                              REM:ATC 1/4 DRILL
N130 G45 H02                             REM:#2 OFFSETS
N140 S3500 F21 M03
N150 G81 G99 X.5 Y1 Z-.375 R0 M08        REM:DRILL HOLE #1
N160 G51 N4                              REM:INITIATE LOOP
N170 G91 X1                              REM:INC. COOR. IN LOOP
N180 G50                                 REM:END LOOP
N190 G80 G90 G49 Z0 M09                  REM:RETRACT Z, COOL. OFF
N200 X-12 Y8 M05                         REM:TO SPNDL PARK/SPNDL
OFF.
N210 M30                                 REM:END PRGM
```

FIGURE 11–3
Do loop program for part in Figure 11–1, word address format

N050

- G45 H01—Calls up the tool offsets in register #1. Note that the codes for tool offsets also differ from machine to machine. This is just one example of tool offset coding. The important point is that tool offsets must be coded.

N060

- S3500—Assigns the spindle speed.
- F10.5—Assigns the feedrate. In this example, the feedrate moves are Z-axis movement during the canned cycle operation.
- M03—Turns the spindle on clockwise.

N070

- G81—Initiates the canned drilling cycle.
- G99—Selects a Z-axis return to the reference (rapid) level, which is the start of the buffer zone.
- X.5 Y1—Coordinates of the first hole.
- Z–.162—Z-axis center-drilling depth.
- R0—Sets reference level for the drilling cycle.
- M08—Turns the coolant on.

N080

- G51—Initiates the do loop.
- N4—Instructs the MCU to repeat the operation contained in the loop four times.

N090

- G91—Selects incremental positioning.
- X1—Incremental distance between the holes to be drilled inside the loop.

N100

- G50—Signals the end of the loop information.

N110

- G80—Cancels the G81.
- G90—Selects absolute positioning.
- G49—Cancels the active tool offset.
- Z0—Retracts the spindle.
- M09—Turns the coolant off.

N120

- M06 T2—Initiates an automatic tool change, selecting the #2 position in the storage magazine from which to take the new tool.

N130

- G45 H02—Calls up the offsets contained in register #2.

N140

- S3500—Sets the spindle speed.
- F21—Sets the feedrate.
- M03—Turns the spindle on clockwise.

N150

- G81—Initiates the canned drilling cycle.
- G99—Selects a return to reference (rapid level)A.
- X.5 Y1—Coordinates of hole #1.
- Z−.375—Z-axis depth to drill through the part.
- R0—Sets the reference level for the drilling cycle.
- M08—Turns the coolant on.

N160

- G51—Initiates the do loop.
- N4—Instructs the MCU to perform the loop four times.

N170

- G91—Selects incremental positioning.
- X1—Incremental distance between the holes.

N180

- G50—Signals the end of the loop information.

N190

- G80—Cancels the canned drilling cycle.
- G90—Selects absolute positioning.
- G49—Cancels the tool offsets.
- Z0—Retracts the spindle.
- M09—Turns off the coolant.

N200

- X−12 Y8—Coordinates of the spindle park position. Any coordinates that safely position the cutter out of the way are adequate.
- M05—Turns off the spindle.

N210

- M30—Signals the end of the program, resetting the computer memory to the start of the sequence.

SUBROUTINES

A *subroutine* is a program within a program, placed at the end of the main program. For example, on the part in Figure 11–4, note that the holes occur in the same geometric and dimensional pattern in four different locations. A do loop could be programmed to drill the holes, but programming steps can be minimized by placing the pattern in a subroutine. The drill can be sent to hole #1 and the subroutine called to drill the four holes A, B, C, and D. Hole #2 can then be positioned and the subroutine called again, and so on.

One way to use a subroutine is to place one or more do loops in the subroutine. This is known as *nesting*. Subroutines may also be nested in other subroutines, or nested within do loops. This gives the programmer a great deal of flexibility and a powerful programming tool.

Machinist Shop Language

The format for a subroutine in Machinist Shop Language is:

1. SUBR n
2. Programming information
3. END

Where SUBR signals the start of the subroutine, n is the subroutine number, the programming information describes the operations required, and END signals the end of the subroutine. A program written in Machinist Shop Language for the part shown in Figure 11–4 is presented in Figure 11–5.

The subroutine at the end of the program is as follows:

SUBR 1
X−.5 Y.5 RI
Y−1 RI
X1 RI
Y1 RI
END

SUBR 1—Signals that the following is subroutine #1.
X−.5 Y.5 RI—Causes an incremental movement −.500 inch in X and .500 inch in Y at rapid traverse.
Y−1 RI—Causes an incremental rapid movement −1.000 inch in Y.
X1 RI—Causes an incremental rapid movement of 1.000 inch in X.
Y1 RI—Causes an incremental rapid movement of 1.000 inch in Y.
END signals the end of the subroutine.

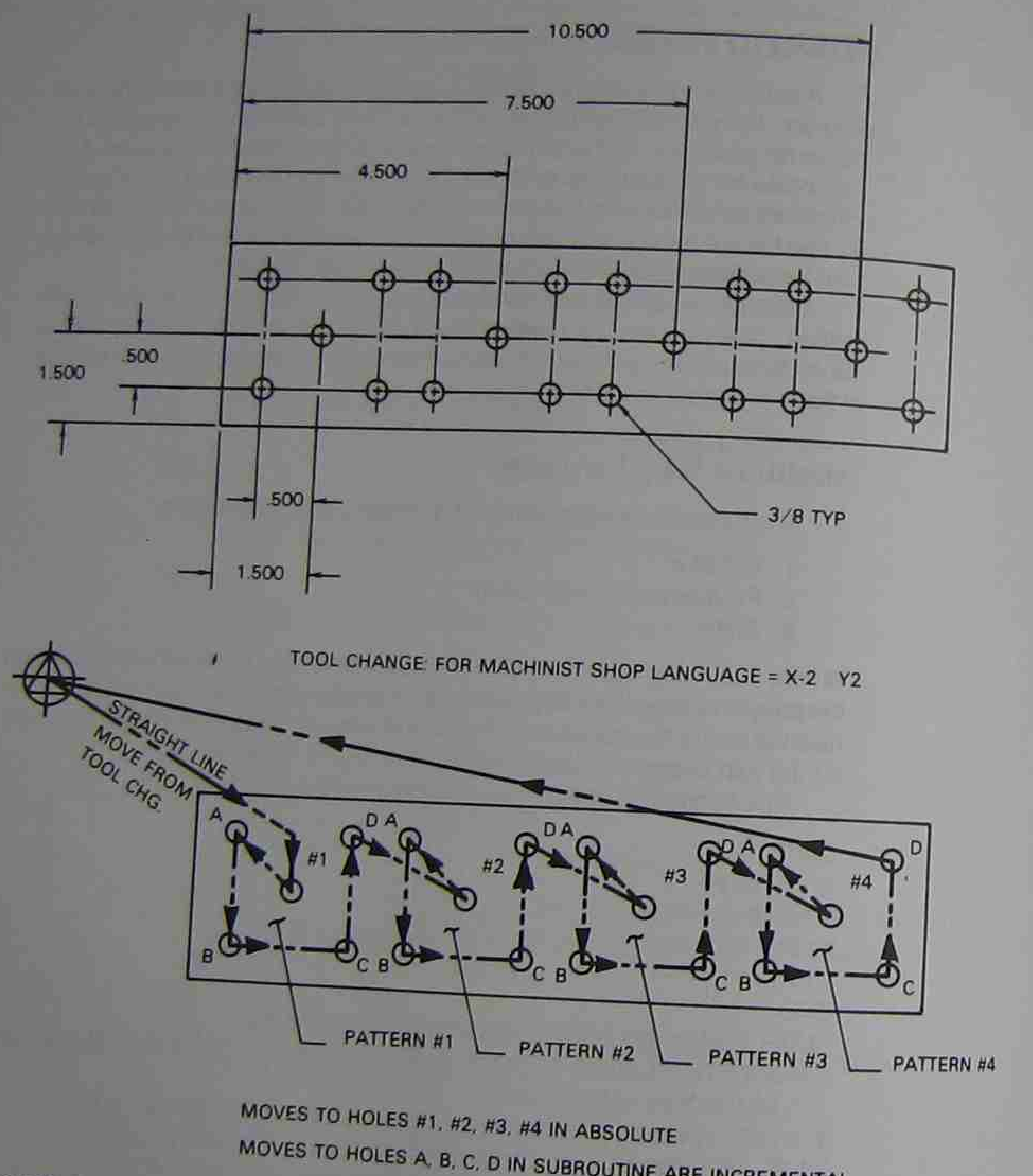


FIGURE 11–4
Part drawing and tool path


```
XO/YO = LOWER LEFT CORNER
TOOLS:   TOOL #1 = #3 C'DRILL.
         TOOL #2 = 3/8 DRILL.
CLEARANCE OVER CLAMPS = 3.000 IN. MIN.

 1 TOOL 1001
 2 Z4
 3 TOOL 1002             A
 4 Z3
 5 X-2 Y2 Z0
 6 TOOL 1                RA    REM:RAPID TO TCH
 7 V20 10.5                    REM:#3 C'DRILL
 8 V21 .1                      REM:SET DRILLING FEED
 9 G81                    A    REM:DEFINE BUFFER ZONE
10 X1.5 Y1.5 Z-.162 RA         REM:INITIATE DRILLING CYCLE
11 CALL 1                      REM:POSITION TO #1 AND C'DRILL
12 X4.5 Y1.5                   REM:CALL SUBR 1
13 CALL 1                RA    REM:POSITION TO AND C'DRILL#2
14 X7.5 Y1.5                   REM:CALL SUBR #1
15 CALL 1                RA    REM:POSITION TO AND C'DRILL #3
16 X10.5 Y1.5                  REM:CALL SUBR 1
17 CALL 1                RA    REM:POSITION TO AND C'DRILL #4
18 G80                         REM:CALL SUBR 1
19 TOOL 0                      REM:CANCEL DRILL CYCLE
20 Z0                          REM:CANCEL TOOL OFFSET
21 X-2 Y2                RA    REM:RETRACT SPNDL
22 TOOL 2                RA    REM:RAPID TO TCH
23 V20 21                      REM:3/8 DRILL
24 V21 .1                      REM:SET DRILLING FEED
25 G81                    A    REM:DEFINE BUFFER
26 X1.5 Y1.5 Z-.375 RA         REM:INITIATE DRILL CYCLE
27 CALL 1                      REM:POSITION TO AND DRILL #1
28 X4.5 Y1.5                   REM:CALL SUBR 1
29 CALL 1                RA    REM:POSITION AND DRILL #2
30 X7.5 Y1.5                   REM:CALL SUBR 1
31 CALL 1                RA    REM:POSITION AND DRILL #3
32 X10.5 Y1.5                  REM:CALL SUBR 1
33 CALL 1                RA    REM:POSITION AND DRILL #4
34 G80                         REM:CALL SUBR 1
35 TOOL 0                      REM:CANCEL DRILL CYCLE
36 Z0                          REM:CANCEL TOOL OFFSET
37 X-2 Y2                RA    REM:RETRACT SPNDL
38 END                   RA    REM:RAPID TO TCH
39 SUBR 1                      REM:END OF MAIN PRGM
40 X-.5 Y.5                    REM:DEFINES START OF SUBROUTINE 1
41 Y-1                   RI    REM:POSITIONS TO HOLE A
42 X1                    RI    REM:POSITIONS TO HOLE B
43 Y1                    RI    REM:POSITIONS TO HOLE C
44 END                   RI    REM:POSITIONS TO HOLE D
                               REM:END OF SUBROUTINE
```

FIGURE 11–5
Subroutine program for part in Figure 11–4, Machinist Shop Language

PROGRAM EXPLANATION

(Refer to Figure 11–5.)

EVENTS 1–5
- Assign the tool information and position the machine at tool change.

EVENT 6
- TOOL 1—Calls up tool #1's offsets. It also issues a dwell to allow the operator to put the center drill in the spindle. A combination drill can also be used to drill this part. By using two tools, the use of the subroutine can be better demonstrated.

EVENT 7
- V20 10.5—Assigns a feedrate of 10.5 inches per minute to be used with the G81 drill cycle.

EVENT 8
- V21 .1—Defines a .100-inch buffer zone between the top of the part and the end of the tool.
- A—Specifies absolute positioning.

EVENT 9
- G81—Initiates the drilling cycle.

EVENT 10
- X1.5 Y1.5—Coordinates for hole #1.
- Z–.162—Depth for drilling, with the buffer zone added.
- R—Specifies rapid traverse mode, which will be used throughout the drilling program.
- A—Specifies absolute positioning.

EVENT 11
- CALL 1—Instructs the MCU to go to subroutine #1 and carry out the instructions there. At the end of the subroutine, the MCU automatically resets to the line following the CALL statement.

EVENT 12
- X4.5 Y1.5—Coordinates for hole #2. Note that both an X and Y coordinate are necessary. At the end of the subroutine, the tool is positioned over hole D.
- A—Specifies absolute positioning, which must be reselected following the subroutine. The subroutine coordinates are incremental.

EVENT 13
- CALL 1—Instructs the MCU to carry out the instructions contained in subroutine #1.

EVENT 14
- X7.5 Y1.5—Positions the machine to hole #3. Since the G81 has not been canceled, a hole is drilled here.

EVENT 15
- CALL 1—Once again calls up the subroutine instructions, which drill holes A, B, C, and D.

EVENT 16
- X10.5 Y1.5—Positions the machine to hole #4.

EVENT 17
- CALL 1—Calls the subroutine to drill the other four holes in the pattern.

EVENT 18
- G80—Cancels the active drilling cycle.

EVENT 19
- TOOL 0—Cancels the tool offset.

EVENT 20
- Z0—Retracts the spindle.

EVENT 21
- X–2 Y2—Coordinates to send the spindle to tool change, where the drill will be installed in the spindle.

EVENT 22
- TOOL 2—Calls up tool #2's offsets.

EVENT 23
- V20 21—Assigns a feedrate of 21 inches per minute to be used for drilling.

EVENT 24
- V21 .1—Establishes a buffer zone of .100 inch between the tool and part surface.

EVENT 25
- G81—Initiates the canned drilling cycle.

EVENTS 26–37
- Duplicate events 10 through 21.

EVENT 38
- END—Signals the end of the main program.

EVENT 39

- SUBR 1—Defines the start of subroutine #1. This subroutine could have been given any number, but it makes sense to use 1 for the first subroutine, 2 for the second, and so on.

EVENT 40

- X–.5 Y.5—Incremental coordinates to send the spindle from the hole in the center of the pattern to hole A.
- R—Specifies rapid traverse mode, which has been used throughout.
- I—Specifies incremental positioning. The coordinates within the subroutine must be incremental, in order to take advantage of the relationships common to all four hole patterns. Since the centerpoint of each pattern is in a different location, the absolute coordinates of the holes in each pattern are unique.

EVENT 41

- Y–1—Incremental coordinate to move from hole A to hole B.

EVENT 42

- X1—Incremental coordinate to move from hole B to hole C.

EVENT 43

- Y1—Incremental coordinate to move from hole C to hole D.

EVENT 44

- END—Signals the MCU that this is the end of the subroutine.

This program could also have been written using the common incremental distance between hole pattern centers of 3.000 inches and a nested subroutine or a nested loop.

Word Address Format

In word address, the subroutine is not identified by a subroutine number as in Machinist Shop Language but is simply added with a sequence number following the main program. All blocks in a subroutine are then numbered from N010 consecutively as if the subroutine were an independent program. The codes associated with a word address subroutine are:

M98—Tells the machine to jump to a subroutine.
M99—Tells the machine to return to the main program.
P—The address P indicates a block sequence number when calling up or returning from a subroutine. P120 would mean N120, as will be demonstrated in a moment.
L—The address L tells the machine how many times to repeat the subroutine.

The format for a word address subroutine is:

1. SEQ # Pn1 Lnn M98
2. :PROG # N010
3. Programming information
4. SEQ # M99 Pn2

Where n1 is the block number that starts the subroutine, nn is the number of times the subroutine is to be repeated, :PROG # is the sequence number of the subroutine, N010 sets the sequence number of the subroutine equal to sequence number N010, and n2 is the sequence number of the main program to be returned to. Some controllers use an O (letter "O") address in place of the colon.

PROGRAM EXPLANATION

(Refer to Figure 11–6.)

N010–N030

- Assign the tool information to the tool registers.

N040

- M06 T1—Initiate an automatic tool change, with tool position #1 being used for the new tool.

N050

- S3500—Sets the spindle speed to 3500 RPM.
- F10.5—Sets the feedrate to 10.5 inches per minute.
- M03—Turns the spindle on clockwise.

N060

- G45 H01—Call up the offsets in register #1, which will be used for the center drill.

N070

- G81—Initiates the canned drilling cycle.
- G99—Selects a Z-axis return to the reference (rapid) level when the G81 is cycled.
- X1.5 Y1.5—Positions the center drill over hole #1.
- Z–.162—Z-axis depth for the center drilling.
- R0—Sets the reference level to Z0, which was set to be .100 off the top of the part at setup.
- M08—Turns on the coolant.

```
X0/Y0 = LOWER LEFT CORNER
TOOLS:  TOOL #1 = #3 C'DRILL.
        TOOL #2 = 3/8 DRILL.
CLEARANCE OVER CLAMPS = 3.000 IN. MIN.

N010 G00 G40 G49 G80 G70 G90 X0 Y0 Z0 REM:SAFTEY LINE
N020 G10 H01 Z4
N030 G10 H02 Z3
N040 M06 T1
N050 S3500 F10.5 M03                  REM:SET SPEED/FEED, SPNDL
ON
N060 G45 H01                          REM:CALL UP OFFSET #1
N070 G81 G99 X1.5 Y1.5 Z-.162 R0 M08  REM:POSITION AND C'DRILL #1
N080 P300 M98                         REM:JUMP TO SUBR 1
N090 G90 X4.5 Y1.5                    REM:POSITION AND C'DRILL #2
N100 P300 M98                         REM:JUMP TO  SUBR 1
N110 G90 X7.5 Y1.5                    REM:POSITION AND DRILL #3
N120 P300 M98                         REM:JUMP TO SUBR 1
N130 G90 X10.5 Y1.5                   REM:POSITION AND DRILL #4
N140 P300 M98                         REM:JUMP TO SUBR 1
N150 G49 G90 G80 Z0 M09               REM:RETRACT SPNDL
N160 M06 T2                           REM ATC, TOOL #2
N170 S3500 F21 M03                    REM:SET SPEED/FEED, SPNDL
ON
N180 G45 H02                          REM:OFFSET #2
N190 G81 G99 X1.5 Y1.5 Z-.375 R0 M08  REM:POSITION AND DRILL #1
N200 P300 M98                         REM:JUMP TO SUBR. 1
N210 G90 X4.5 Y1.5                    REM:POSITION AND DRILL #2
N220 P300 M98                         REM:JUMP TO SUBR 1
N230 G90 X7.5 Y1.5                    REM:POSITION AND DRILL #3
N240 P300 M98                         REM:JUMP TO SUBR 1
N250 G90 X10.5 Y1.5                   REM:POSITION AND DRILL #4
N260 P300 M98                         REM:JUMP TO SUBR 1
N270 G49 G80 G90 Z0 M09               REM:RETRACT SPNDL, COOL.
OFF
N280 X-12 Y8 M05                      REM:RAPID TO PARK POSTION
N290 M30                              REM:END OF PGRM
:300 N010 G91 X-.5 Y.5                REM:START OF SUBR, DRILL A
N020 Y-1                              REM:DRILL HOLE B
N030 X1                               REM:DRILL HOLE C
N040 Y1                               REM:DRILL HOLE D
N050 M99                              REM:JUMP TO MAIN PGRM
```

FIGURE 11-6
Subroutine program for part in Figure 11-4, word address format

N080

- P300—Sequence number of the subroutine starting location. P300 is used here instead of N300.
- M98—Tells the machine to jump to the subroutine. The MCU will go to the program block specified as P300, labeled :300, and execute the instructions listed there. The last command in the subroutine instructs the MCU to return to the main program.

N090

- G90—Selects absolute positioning. The subroutine coordinates are incremental; therefore absolute must be specified here.
- X4.5 Y1.5—Coordinates for hole #2. The G81 is still active so that a hole is drilled at every programmed location.

N100

- P300 M98—Again tell the MCU to carry out the instructions in the subroutine in block :300.

N110

- G90—Selects absolute positioning.
- X7.5 Y1.5—Coordinates for hole #3.

N120

- P300 M98—Initiate a jump to the subroutine.

N130

- G90—Selects absolute positioning.
- X10.5 Y1.5—Coordinates for hole #4.

N140

- P300 M98—Again causes a jump to the subroutine.

N150

- G49—Cancels the tool offset.
- G80—Cancels the drilling cycle.
- G90—Selects absolute positioning.
- Z0—Retracts the spindle.
- M09—Turns off the coolant.

N160

- M06 T2—Initiate an automatic tool change, taking tool #2 from the magazine and placing it in the spindle.

N170–N270

- Duplicate blocks N050–N150.

N290

- M30—Signals the end of the program, resetting the computer memory to the start.

:300

- :300—Identifies this block as N300, the beginning of a subroutine.
- N010—Identifies this block as block N010 of the subroutine.
- G91—Selects incremental positioning. Incremental coordinates are used throughout this subroutine.

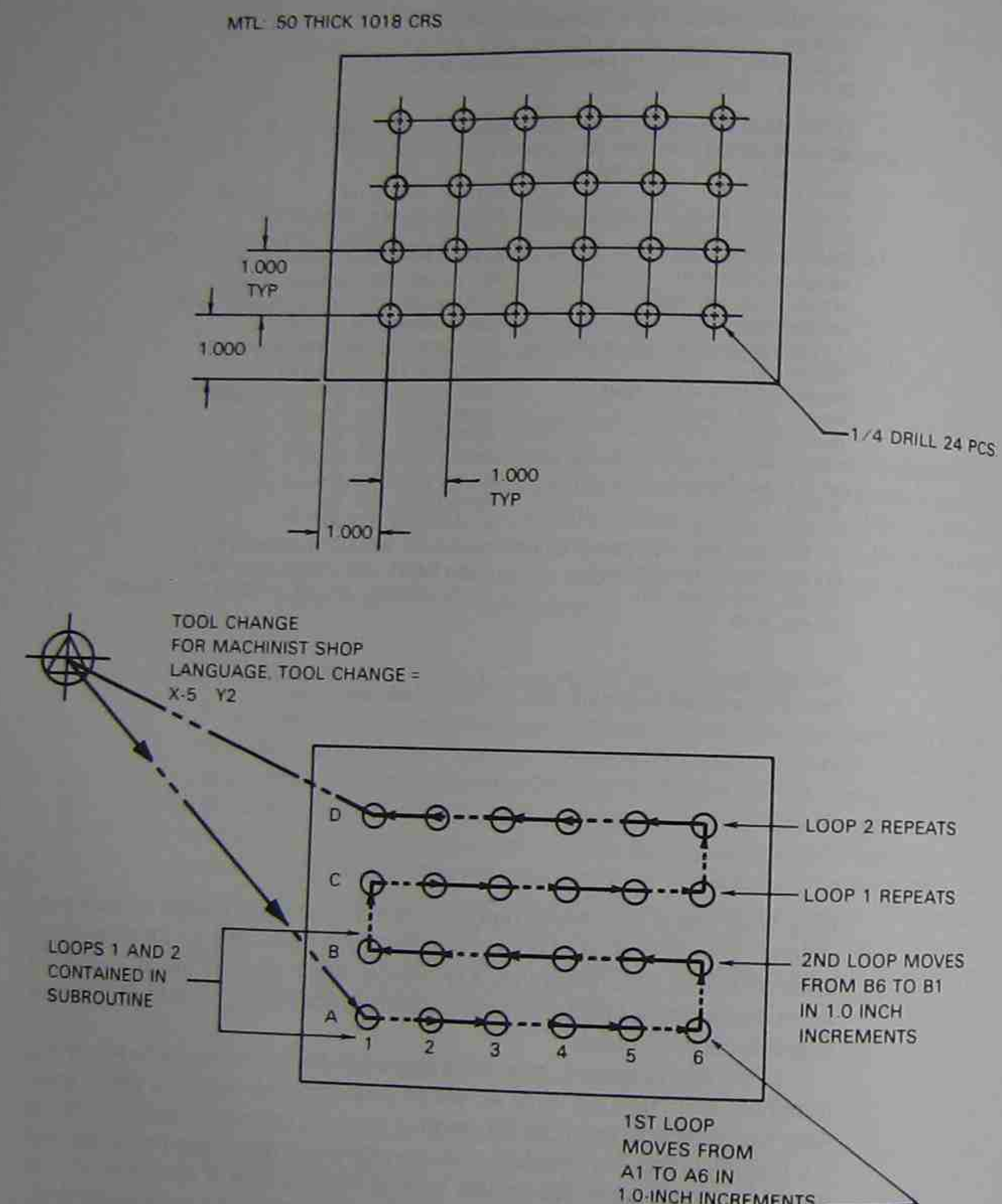


FIGURE 11–8
Part drawing and tool path


```
XO/YO = LOWER LEFT CORNER OF PART
TOOL CHANGE = X-5 Y2
TOOLS:  #3 C'DRILL, 1/4 DRILL
CLEARANCE OVER CLAMPS:  2.5 IN. MIN.

 1 TOOL 1001
 2 Z4
 3 TOOL 1002           A    REM:C'DRILL
 4 Z2.5
 5 X-2 Y5              A    REM:1/4 DRILL
 6 TOOL 1             RA    REM:TCH - C'DRILL
 7 V20 5.1                  REM:1700 RPM
 8 V21 .1                   REM:C'DRILL FEEDRATE
 9 G81                 A    REM:C'DRILL BUFFER
10 X1 Y1 Z-.162       RA    REM:INITIATE DRILL CYCLE
11 CALL 1                   REM:C'DRILL HOLE 1A
12 Y1                       REM:CALL SUBR 1
13 CALL 1             RI    REM:C'DRILL HOLE C1
14 G80                      REM:CALL SUBR 1
15 TOOL 0                   REM:CANCEL DRILLING CYCLE
16 Z0                       REM:CANCEL TOOL OFFSET
17 X-5 Y2             RA    REM:RETRACT Z
18 TOOL 2             RA    REM:TCH - 1/4 DRILL
19 V20 4.8                  REM:1200 RPM
20 V21 .1                   REM:DRILL FEEDRATE
21 G81                 A    REM:DRILL BUFFER
22 X1 Y1 Z-.7         RA    REM:DRILL HOLE A1
23 CALL 1
24 Y1                 RI    REM:DRILL HOLE C1
25 CALL 1                   REM:CALL SUBR 1
26 G80                      REM:CANCEL DRILL CYCLE
27 TOOL 0                   REM:CANCEL TOOL OFFSET
28 Z0                 RA    REM:RETRACT Z
29 X-2 Y5             RA    REM:TOOL CHANGE
30 END                      REM:END PROGRAM
31 SUBR 1
32 DO 5
33 X1                 RI
34 END                      REM:END DO LOOP
35 Y1                 RI    REM:START NEXT ROW
36 DO 5
37 X-1                RI
38 END                      REM:END DO LOOP
39 END                      REM:END OF SUBROUTINE
```

FIGURE 11–9
Nested loop program for part in Figure 11–8, Machinist Shop Language

Machinist Shop Language

In Machinist Shop Language, the do loop to drill the holes for row A (assuming the spindle is first positioned over hole A1) is:

DO 5
X1 R I (the incremental distance between holes)
END

```
XO/YO = LOWER LEFT CORNER OF PART
TOOLS:  #3 C'DRILL, 1/4 DRILL
CLEARANCE OVER CLAMPS:  2.5 IN.

%                                         REM:SAFETY LINE
N010 G00 G90 G80 G40 Z0                   REM:C'DRILL
N020 G10 H01 Z4.0                         REM:1/4 DRILL
N030 G10 H02 Z2.5                         REM:TOOL CHANGE
N040 T01 M06
N050 S1700 M03                            REM:CALL OFFSET 01
N060 G45 H01 M08                          REM:C'DRILL HOLE AT A1
N070 G81 G99 X1.0 Y1.0 Z-.162 R0 F5.1     REM:JUMP TO SUB
N080 P240 M98                             REM:C'DRILL HOLE C1
N090 Y1.0                                 REM:JUMP TO SUB
N100 P240 M98                             REM:CANCEL G81
N110 G80 G90 M09                          REM:CANCEL OFFSET
N120 G49
                                          REM:TOOL CHANGE
N130 T02 M06
N140 S1200 M03                            REM:CALL OFFSET 02
N150 G45 H02 M08                          REM:DRILL HOLE AT A1
N160 G81 G99 X1.0 Y1.0 Z-.7 R0 F4.8       REM:JUMP TO SUB
N170 P240 M98                             REM:DRILL HOLE C1
N180 Y1.0                                 REM:JUMP TO SUB
N190 P240 M98                             REM:CANCEL G81
N200 G80 G90 M09                          REM:CANCEL OFFSET
N210 G49                                  REM:PARK SPINDLE
N220 X-12.0 Y8.0 M05                      REM:END PROGRAM
N230 M30

:240 N010 G51 N5                          REM:START SUB & LOOP
N020 G91 X1.0                             REM:CONTENTS OF LOOP
N030 G50                                  REM:END LOOP
N040 Y1.0                                 REM:START NEXT ROW
N050 G51 N5                               REM:START LOOP
N060 X-1.0                                REM:CONTENTS OF LOOP
N070 G50                                  REM:END LOOP
N080 M99                                  REM:PROGRAM RETURN
%
```

FIGURE 11–10

For row B (assuming the spindle is first positioned over hole B6), the do loop is:

DO 5
X – 1 R I (the incremental distance between holes)
END

These two loops are placed inside a subroutine along with some other programming information to drill the rows of holes. The subroutine is written as follows:

```
SUBR 1
DO 5      REM:DO LOOP FOR ROW A
X1 R I
END       REM:END OF LOOP
```



```
Y1 R I    REM:POSITION FOR START OF NEXT LOOP
DO 5      REM:DO LOOP FOR ROW B
X – 1 R I
END       REM:END OF LOOP
END       REM:END OF SUBROUTINE
```

By calling up this subroutine, rows A and B will be drilled. After positioning the machine to hole C1, calling up the subroutine a second time will drill rows C and D.

It cannot be overstressed that the examples presented here are only that. Codes vary from controller to controller, even in different controllers from the same manufacturer. The EIA standards are the guidelines manufacturers follow; however, many controllers break with standard coding on occasion for one reason or another. **When in doubt, check the programming manual.** Don't take chances with the machine's and, more importantly, the operator's safety.

SPECIAL FANUC SECTION

Figure 11 – 11 contains the program to drill the part in Figure 11 – 1 using a do loop. Figure 11 – 12 is the program to drill the part in Figure 11 – 4 using a subprogram. Fanuc controllers do not use subroutines as such. Each subroutine is actually a mini program, stored in the controller under its own "O" number. The subroutines are called subprograms for this reason. A colon (:) can be substituted for the letter "O" if desired, as was done in the text examples.

The programs in Figures 11 – 11 and 11 – 12 were written for a Tsugami "Lightning" horizontal machining center. A work coordinate is used to origin the part coordinate system using G54.

The length offsets are picked up using a G43 command. They are cancelled by returning the spindle to the home zero position, which is also the tool change position, using a G28 command.

This machine is a multipallet machine; up to 30 pallets can be used. The M91 command issued as part of the program startup statements is a pallet recognition command. This tells the controller the machine has been equipped with the multipallet option. In the examples given in this section, only one pallet is used.

```
%
O1103
(--------------------------------)
(XO/YO - LOWER LEFT CORNER)
(ZO - .100 ABOVE TOP OF PART)
(--------------------------------)
(TOOL NO. 1 - NO. 3 C-DRILL)
N001G17G90M91
N101T01M06
N102G54S3500M03T02
N103G00X.5Y1.M08
N104G43G90Z2.5H01
N105G81G99X.5Y1.Z-.162R0.F10.5
()
(BEGIN DO LOOP)
N106G51N4
N107G91X1.
N108G50
(END OF LOOP)
()
N109M09
N110G00G91G28Z0M05
N111G28X0Y0M01
()
(--------------------------------)
(TOOL NO. 2 - 1/4 DRILL)
N002G90
N201T02M06
N202G54S3500M03T01
N203G00X.5Y1.M08
N204G43G90Z2.5H02
N205G81G99X.5Y1.Z-.375R0.F21.
()
(BEGIN DO LOOP)
N206G51N4
N207G91X1.
N208G50
(END OF LOOP)
()
N209M09
N210G00G91G28Z0M05
N211G28X0Y0
N212M30
%
```

FIGURE 11-11


```
%
O1104
(--------------------------------)
(XO/YO - LOWER LEFT CORNER)
(ZO - .100 ABOVE TOP OF PART)
(--------------------------------)
(TOOL NO. 1 - NO. 3 C-DRILL)
N001G17G90M91
N101T01M06
N102G54S3500M03T02
N103G00X1.5Y1.5M08
N104G43G90Z3.H01
N105G81G99X1.5Y1.5Z-.162R0M08F10.5
N106P1000M98
N107G90X4.5Y1.5
N108P1000M98
N109G90X7.5Y1.5
N110P1000M98
N111G90X10.5Y1.5
N112P1000M98
N113G80M09
N114G91G28Z0M05
N115G28X0Y0Z0M01
()
(--------------------------------)
(TOOL NO. 2 - 3/8 DRILL)
N002G90
N201T02M06
N202G54S3500M03T01
N203G00X1.5Y1.5M08
N204G43G90Z3.H02
N205G81G99X1.5Y1.5Z-.375R0.F12.
N206P1000M98
N207G90X4.5Y1.5
N208P1000M98
N209G90X7.5Y1.5
N210P1000M98
N211G90X105.Y1.5
N212P1000M98
N213G80M09
N214G00G91G28Z0M05
N215G28X0Y0
N216M30
()
()
O1000
(START OF SUB PROGRAM 1000)
N001G91X-.5Y.5
N002Y-1.
N003X1.
N004Y1.
N005M99
%
```

FIGURE 11-12

SUMMARY

The important concepts presented in this chapter are:

- A do loop instructs the MCU to repeat a series of instructions a specified number of times.
- The format for a do loop in Machinist Shop Language is:

  DO #
  Programming information
  END

  Where # is the number of times the loop is to be repeated.
- The format for a do loop in word address is:

  G51 N#
  Programming information
  G50

  Where # is the number of times the loop is to be repeated.
- A subroutine is a program within a program, placed at the end of the main program.
- The format for a Machinist Shop Language subroutine is:

  SUBR #
  Programming information
  END

  Where # is the number of the subroutine.
- The format for a word address subroutine as used in the text examples is:

  SEQ # Pn1 Lnn M98
  :PROG # N010
  Programming information
  SEQ # M99 Pn2

  Where Pn1 = Sequence number at which subroutine starts, and Pn2 = the block number of the main program to be returned to. If no block number is specified, the MCU returns to the block following the M98 command last issued.
- Nested loops are loops placed inside other loops or inside subroutines.
- The codes for subroutines and do loops vary from controller to controller. To program a particular machine, it will be necessary to consult the programming manual for the machine in question.

REVIEW QUESTIONS

1. What is a do loop? A subroutine? A nested loop?
2. What is the format for a do loop in Machinist Shop Language? In word address?
3. What is the format for a subroutine in Machinist Shop Language? In word address?
4. Write a do loop to drill the hole patterns in Figure 11–13:
   a. In Machinist Shop Language.
   b. In word address.
5. Write a program using a subroutine to mill the slots in Figure 11–14:
   a. In Machinist Shop Language.
   b. In word address.

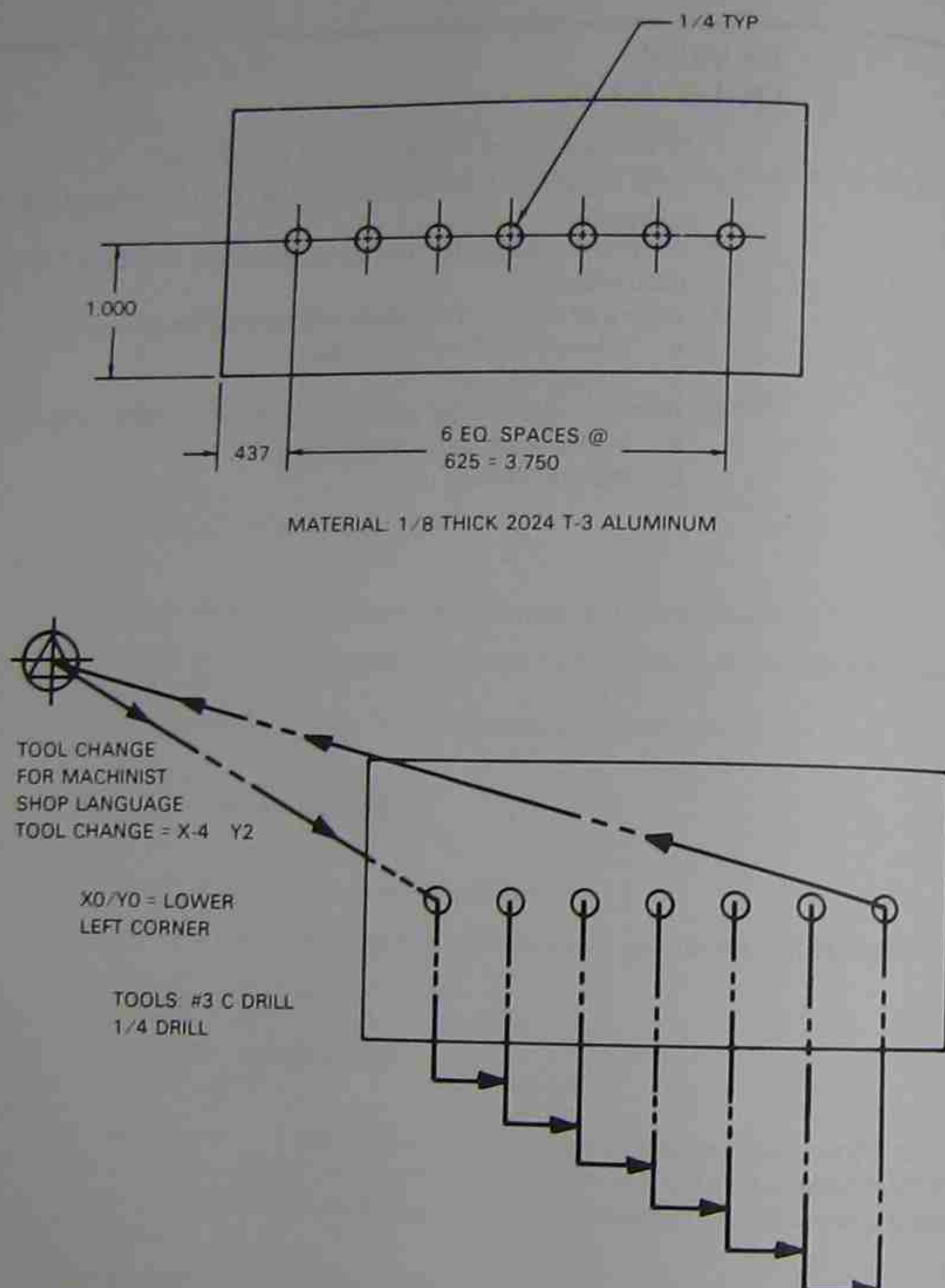


FIGURE 11–13
Part drawing for review question #4

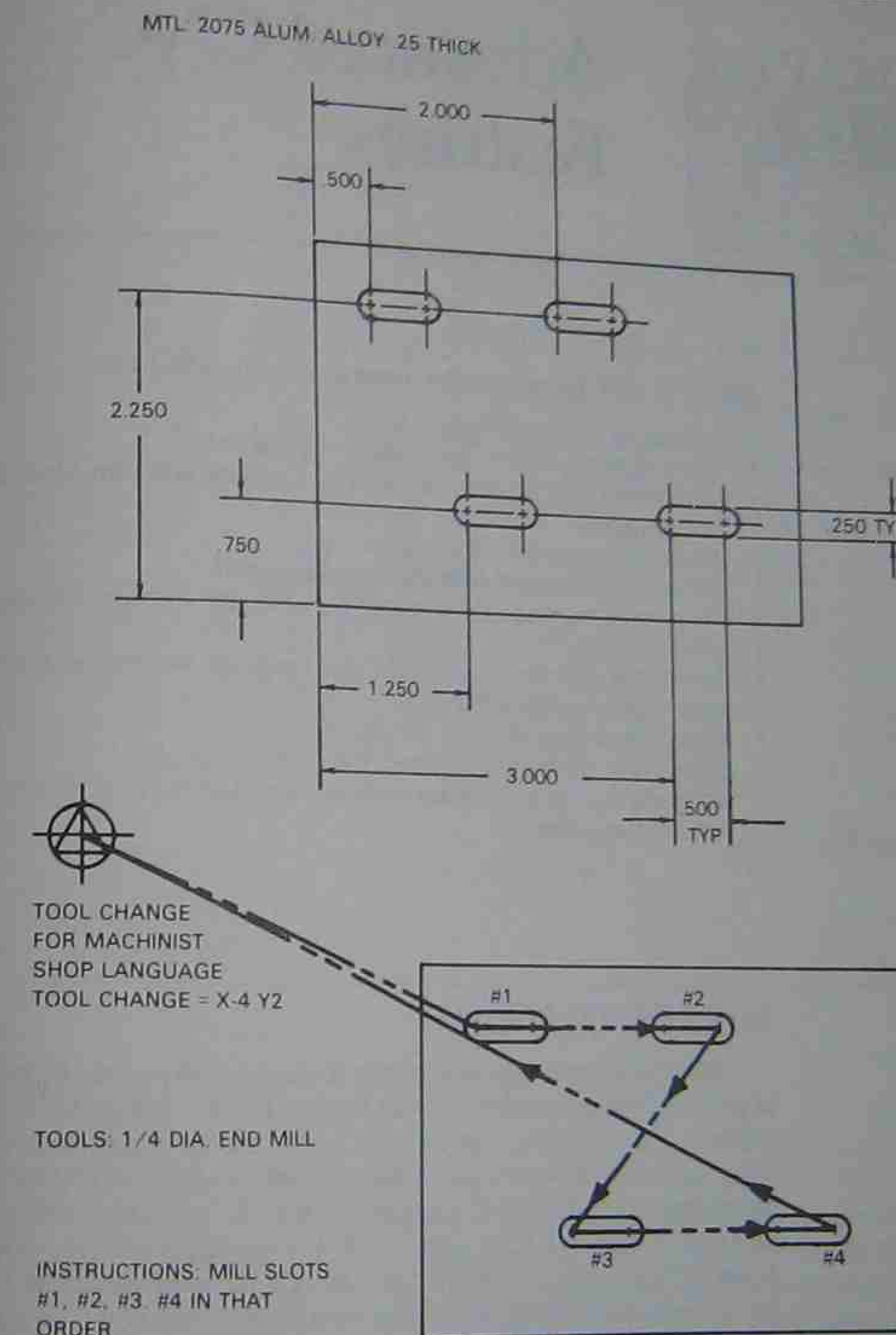


FIGURE 11–14
Part drawing for review question #5

CHAPTER 12

Advanced CNC Features

OBJECTIVES Upon completion of this chapter, you will be able to:

- Explain the concept of mirror imaging.
- Decide when the use of mirror imaging is appropriate.
- Write simple programs in Machinist Shop Language and word address that employ mirror imaging.
- Explain the concept of polar rotation.
- Decide when the use of polar rotation is appropriate.
- Write simple programs in Machinist Shop Language and word address that employ polar rotation.
- Write simple programs in Machinist Shop Language and word address that employ polar rotation used in a do loop.
- Explain the concept of helical interpolation.
- Decide when the use of helical interpolation is appropriate.
- Write simple programs in Machinist Shop Language and word address that employ helical interpolation.

MIRROR IMAGING

Mirror imaging is a simple concept that can be very useful in programming. In essence, *mirror imaging* reverses the sign (+ or −) of an axis direction. For example, mirror imaging can be employed to shorten the amount of programming required to make the part shown in Figures 12−1 and 12−2. Calling the centerline of this part X0/Y0, the pattern of holes to the right of the centerline can be programmed in a subroutine. After this pattern is drilled, mirror imaging along the X axis can be instituted and the subroutine called again. This will drill the same pattern of holes in the second quadrant, with no additional programming save the mirror imaging command. The process can be repeated, mirror imaging the Y axis to drill the pattern in the third quadrant. Canceling the mirror image on the X axis and leaving it active on Y will drill the pattern in the fourth quadrant.

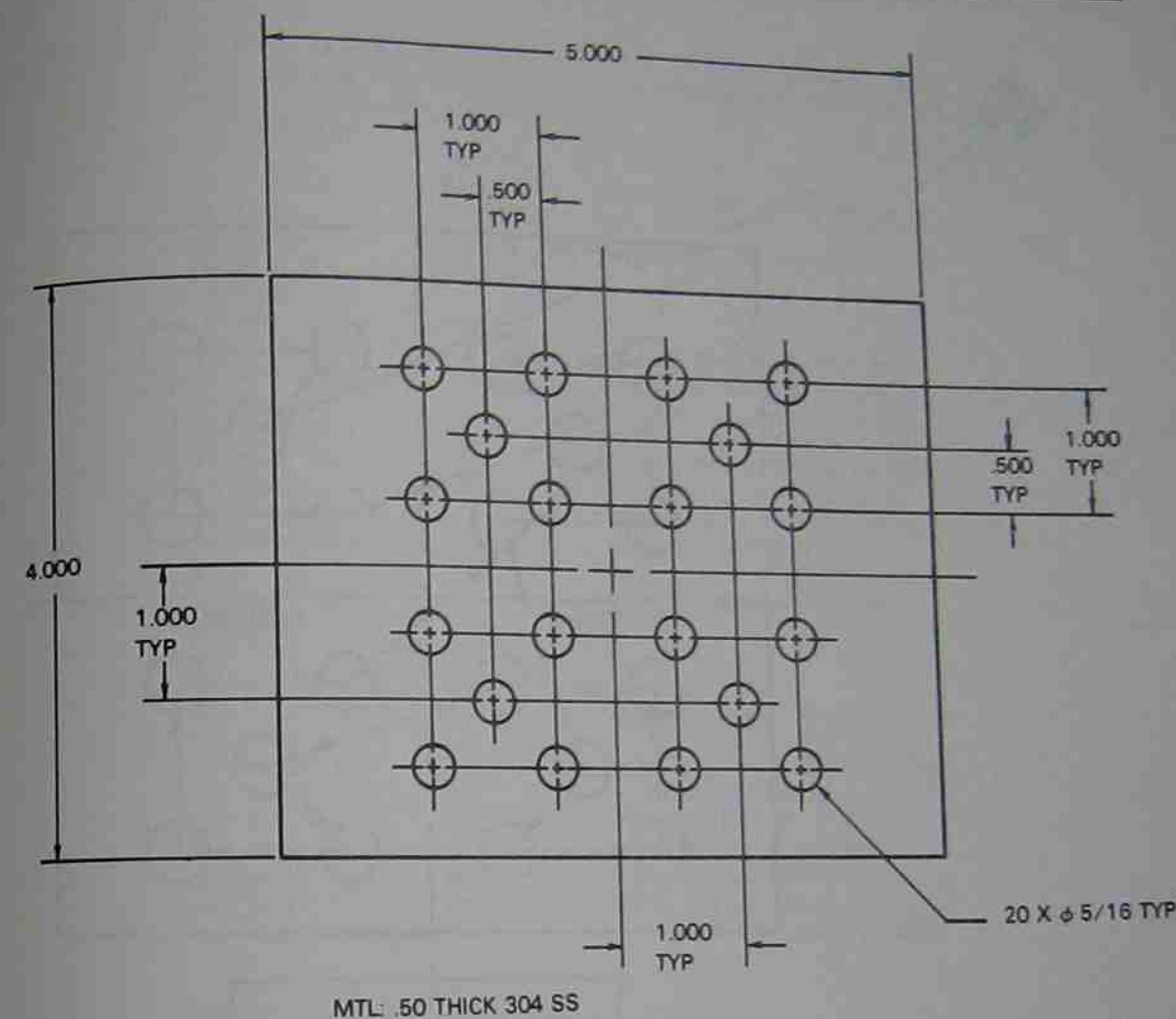


FIGURE 12−1
Part drawing

Machinist Shop Language

In Machinist Shop Language, mirror imaging is instituted by use of an auxiliary (AUX) code. Appendix 2 lists various auxiliary codes used with Machinist Shop Language. These codes are generally used for the same purposes for which miscellaneous functions are used in word address format. The Machinist Shop Language program used to drill this part (see Figures 12−1 and 12−2) is Figure 12−3. The program uses a combination drill/center drill. Notice how short the program is, considering the number of holes, when the right combination of machine features and tooling is used.

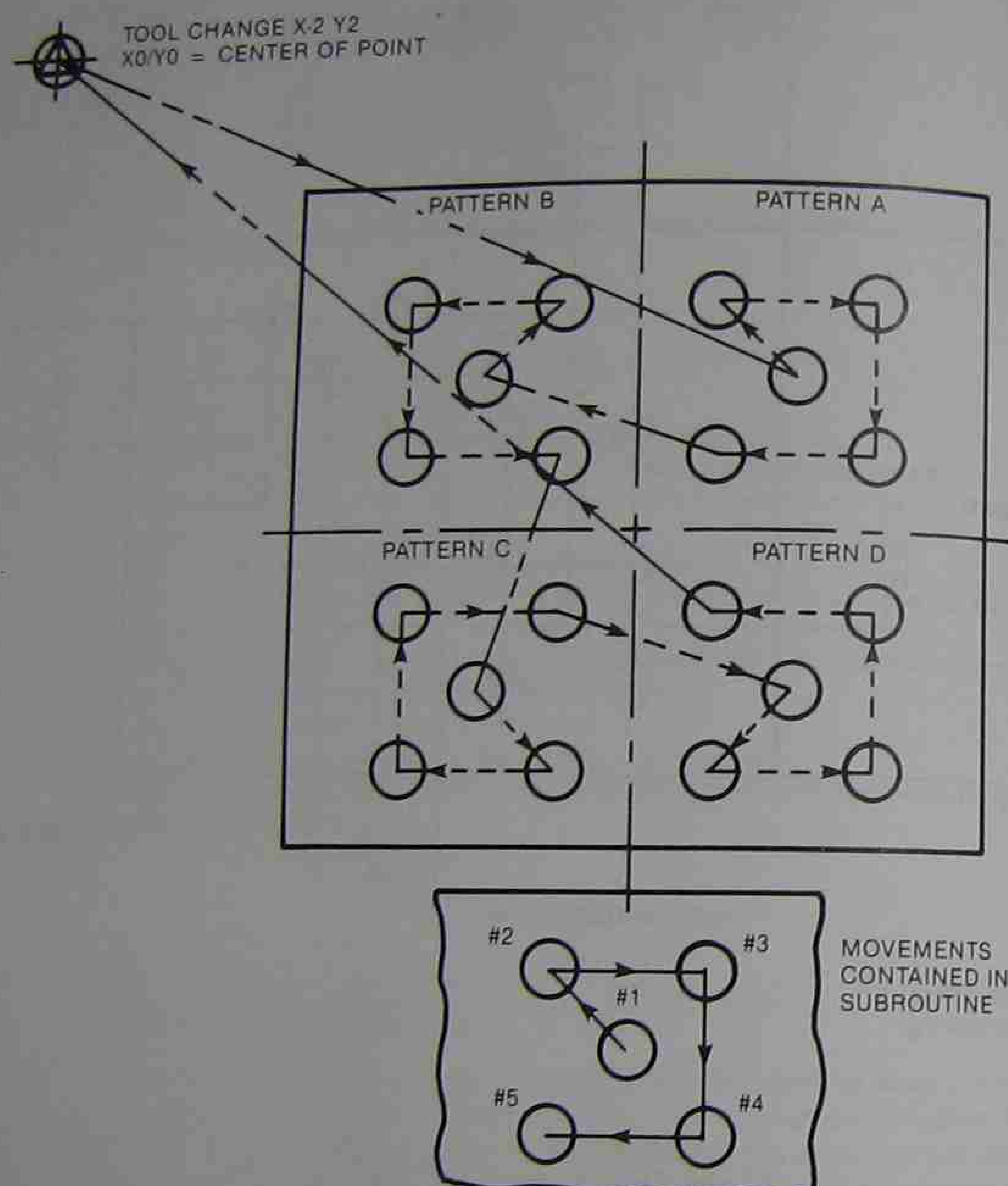


FIGURE 12–2
Tool path for the part shown in Figure 12–1


```
X0/Y0 = CENTER OF PART
TOOL CHANGE = X-2 Y2
TOOLS:  5/16 DIA. COMB. DRILL
 1 TOOL 1001
 2 Z2                A
 3 X-2 Y2 Z0        RA        REM:TO TOOL CHANGE
 4 TOOL 1
 5 X0 Y0            RA        REM:MOVE TO X0/Y0
 6 V20 5
 7 V21 .1
 8 G81 Z-.7         RA
 9 CALL 1                     REM:DRILL PATTERN #A
10 AUX 100                    REM:MIRROR IMAGE IN X
11 CALL 1                     REM:DRILL PATTERN #B
12 AUX 800                    REM:MIRROR IMAGE OFF
13 AUX 300                    REM:MIRROR IMAGE X AND Y
14 CALL 1                     REM:DRILL PATTERN #C
15 AUX 800                    REM:MIRROR IMAGE OFF
16 AUX 200                    REM:MIRROR IMAGE Y
17 CALL 1                     REM:DRILL PATTERN D
18 AUX 800                    REM:MIRROR IMAGE OFF
19 G80
20 TOOL 0
21 Z0
22 X-2 Y2           RA        REM:TO TOOL CHANGE
23 END
24 SUBR 1                     REM:START SUBR 1
25 X1 Y1            RA        REM:DRILL HOLE #1
26 X-.5 Y.5         RI        REM:DRILL HOLE #2
27 X1               RI        REM:DRILL HOLE #3
28 Y-1              RI        REM:DRILL HOLE #4
29 X-1              RI        REM:DRILL HOLE #5
30 END                        REM:END OF SUBROUTINE
```

FIGURE 12–3
Mirror imaging program for the part in Figure 12–1, Machinist Shop Language

PROGRAM EXPLANATION

EVENTS 1–4

- These events assign and call up the tool information.

EVENT 5

- X0 Y0—Position the part to the X0/Y0 point, which is the center of the part. The reason for this move is to allow the subroutine to control movement to the center hole in each of the hole patterns. The machine stops only momentarily at this location, with no spindle movement taking place.

- R—Specifies a rapid move. Rapid traverse will be used throughout since this is a drilling program.
- A—Specifies absolute positioning. This program will demonstrate the value of manipulating absolute and incremental positioning.

EVENT 6
- V20—Sets the drilling feedrate to 5 inches per minute.

EVENT 7
- V21 .1—Defines the buffer as .100 inches.

EVENT 8
- G81—Initiates the drilling cycle.
- Z−.7—Is the Z axis depth for drilling.

EVENT 9
- CALL 1—Calls the subroutine; pattern A is drilled.

EVENT 10
- AUX 100—Mirror images the X axis. When the subroutine is called again, the identical sequence of events will take place with the exception of the X axis moves, which will now be reversed.

EVENT 11
- CALL 1–Calls the subroutine again. This time the AUX 100 is active, so that pattern B is drilled.

EVENT 12
- AUX 800—Turns off the mirror image. Before issuing a mirror image command a second time, it is necessary to cancel the active command. Although some controllers may allow the commands to be issued one after another without a cancel command in between, it is always wise to play it safe.

EVENT 13
- AUX 300—Mirror images the X and Y axes. Since the event preceding this one canceled any active mirror imaging, the movement with AUX 300 active will be reversed on both the X and Y axes.

EVENT 14
- CALL 1—Calls the subroutine again. This time, with both X and Y reversed from pattern A, pattern C will be drilled.

EVENT 15
- AUX 800—Cancels the mirror image.

EVENT 16
- AUX 200—Mirror images the Y axis.

EVENT 17
- CALL 1—Calls the subroutine again. This time through the Y axis movements are reversed from pattern A, and pattern D is drilled.

EVENT 18
- AUX 800—Cancels the mirror imaging.

EVENT 19
- G80—Cancels the active drilling cycle.

EVENT 20
- TOOL 0—Cancels the active tool offset.

EVENT 21
- Z0—Retracts the spindle.

EVENT 22
- X−2 Y2—Are the tool change coordinates.

EVENT 23
- END—Signals the end of the program.

EVENT 24
- SUBR 1—Defines the start of subroutine #1.

EVENT 25
- X1 Y1—Are the absolute coordinates from the X0/Y0 point to hole #1 in pattern A. When mirror imaging commands are active, the positioning will take place to the center holes of other patterns, depending on the active mirror imaging code.
- R—Specifies rapid traverse mode.
- A—Specifies that this is an absolute coordinate.

EVENT 26
- X−.5 Y.5—Incremental coordinates to move from hole #1 to hole #2.
- R—Specifies rapid traverse mode.
- I—Specifies incremental positioning.

EVENT 27
- X1—Incremental coordinate to move from hole #2 to hole #3.

EVENT 28
- Y−1—Incremental coordinate to move from hole #3 to hole #4.

EVENT 29
- X−1—Incremental coordinate to move from hole #4 to hole #5.

EVENT 30
- END—Signals the end of the subroutine.

Word Address Format

In word address, the same procedure is accomplished through either G codes or M functions, depending on the controller. In the following example, M functions are used as follows:

M21—Mirror image X axis.
M22—Mirror image Y axis.
M23—Mirror image off.

On some CNC machines, mirror imaging is selected at the MDI console by means of a switch. When programming such a machine, a dwell must be programmed at the place where mirror imaging is to be instituted, and instructions given for the operator to set the switches prior to restarting the program. The program to drill the part is shown in Figure 12–4.

```
X0/Y0 = CENTER OF PART
TOOL CHANGE = X-2 Y2
TOOLS:  5/16 DIA. COMB. DRILL
BUFFER:  2.5 IN. MIN.
N010 G00 G40 G49 G70 G90 Z0    REM:SAFTEY LINE
N020 G10 H01 Z2.5
N030 M06 T1
N040 S641 F5 M03               REM:SET SPEED/FEED
N050 G45 H01
N060 X0 Y0                     REM:POSITION TO X0/Y0
N070 G81 G99 Z-.7 R0 M08       REM:INITIATE DRILL CYCLE
N080 P190 M98                  REM:JUMP TO SUBROUTINE
N090 M21                       REM:MIRROR IMAGE X
N100 P190 M98                  REM:JUMP TO SUBROUTINE
N110 M22                       REM:MIRROR IMAGE Y
N120 P190 M98                  REM:JUMP TO SUBROUTINE
N130 M23                       REM:CANCEL MIRROR IMAGE
N140 M22                       REM:MIRROR IMAGE Y
N150 P190 M98                  REM:JUMP TO SUBROUTINE
N160 G80 G49 Z0 M09            REM:CANCEL DRILLING, OFFSETS
N170 X-12 Y8 M05               REM:MOVE TO PARK, SPNDL OFF
N180 M30                       REM:END OF MAIN PGRM
:190 N010 X1 Y1                REM:START SUBROUTINE, DRILL #1
N020 G91 X-.5 Y.5              REM:DRILL #2
N030 X1                        REM:DRILL #3
N040 Y-1                       REM:DRILL #4
N050 X-1                       REM:DRILL #5
N060 M99
```

FIGURE 12–4
Mirror imaging program for the part in Figure 12–1, word address format

PROGRAM EXPLANATION

(Refer to Figure 12–4.)

N010

- This is the safety block, canceling any codes that may have been left active following a previous program.

N020–N030

- These blocks assign the tool information and select the tool.

N040

- S641—Sets the spindle speed to 641 RPM.
- F5—Sets the feedrate to 5 inches per minute.
- M03—Turns the spindle on clockwise.

N050

- G45 H01—Call up the tool offsets in register #1.

N060

- X0 Y0—Position the machine to the center of the part, where the subroutine starts.

N070

- G81—Initiates the drilling cycle.
- G99—Selects a return to rapid level.
- Z–.7—Z-axis depth for drilling. Since a G81 code will not move the Z axis until after an X, Y, or X/Y move, no movement takes place along the Z axis yet.
- R0—Sets the start of the buffer (Z0 with a tool offset active) at the rapid level.
- M08—Turns the coolant on.

N080

- P190 M98—Instruct the MCU to jump to the subroutine that starts in block 190.

N090

- M21—Mirror images the X axis.

N100

- P190 M98—Causes a jump to the subroutine.

N110

- M22—Mirror images the Y axis.

N120

- P190 M98—Causes a jump to the subroutine.

N130
- M23—Cancels the active mirror image commands.

N140
- M22—Mirror images the Y axis. It was necessary to cancel the mirror image in block N130 because the X axis was mirror imaged along with the Y. Once canceled, an M22 is used to reestablish the mirror image on the Y axis.

N150
- P190 M98—Causes a jump to the subroutine.

N160
- G80—Cancels the drill cycle.
- G49—Cancels the tool offset.
- Z0—Retracts the spindle.
- M09—Turns off the coolant.

N170
- X – 12 Y8—Coordinates of the park position. As in other word address examples, any place that safely positions the tool out of the way can be used. It is assumed in these examples that the tool change location is at approximately X – 12, Y8 from the part X0/Y0.
- M05—Turns the spindle off.

N180
- M30—Signals the end of the main program and resets the computer memory.

:190
- :190—Identifies this as block 190 of the main program.
- N010—Further identifies this as block N010 of the subroutine.
- X1 Y1—Absolute coordinates to move from the center of the part to hole #1.

N020
- G91—Selects incremental positioning.
- X – .5 Y.5—Incremental coordinates to move from hole #1 to hole #2.

N030
- X1—Incremental coordinate to move from hole #2 to hole #3.

N040
- Y – 1—Incremental coordinate to move from hole #3 to hole #4.

N050
- X – 1—Incremental coordinate to move from hole #4 to hole #5.

N060
- G90—Selects absolute positioning.
- M99—Instructs the MCU to return to the main program.

POLAR ROTATION

Consider the part shown in Figure 12–5, in which four slots are to be milled. A machinist making this part on a conventional vertical milling machine would probably set up the workpiece on a rotary table, rotate 45 degrees from the nominal 0-degree location, and mill the first slot. The other three slots could then be milled, moving the various axes, or the machinist could simply index the part 90 degrees from the first slot to mill the second without excess movement along the X and Y axes. The same type of machining may be accomplished on a CNC machining center or CNC mill equipped with polar rotation.

A polar axis coordinate system is formed by constructing a line whose slope is not the same as either the X or Y axis. For example, in Figure 12–6, a line has been constructed between the origin (point #1) and point #2 on the graph. That line is a polar axis. Notice that point #2 is located 1.0 inch from the origin as measured along the polar axis. If point #2 is specified as (1,0) measured along the polar axis, then point #2 is called a polar coordinate. In mathematics more scientific definitions exist for a polar axis, but for the purposes of CNC programming, *polar rotation* can be thought of as rotating the Cartesian coordinate system.

When polar rotation is instituted in a CNC program, the MCU will triangulate the points necessary to position the tool to the desired coordinates from the program information that it is given. Polar rotation is supplied on most controllers as an optional feature. As with most options, the coding for polar rotation varies greatly from machine to machine. The examples given here can serve only to demonstrate the concept. The NC part programmer will have to consult the programming manual to program polar rotation successfully on a given machine.

Despite the differences in controllers, there is certain information that every MCU needs in order to carry out a polar rotation:

- The X axis coordinate of the center of rotation.
- The Y axis coordinate of the center of rotation.
- The *index angle*, or the angle as measured counterclockwise from the +X axis to the start of the rotation. In the case shown in Figure 12–5, the index angle is 45 degrees. This value is the angular rotation from the X axis to slot #1.

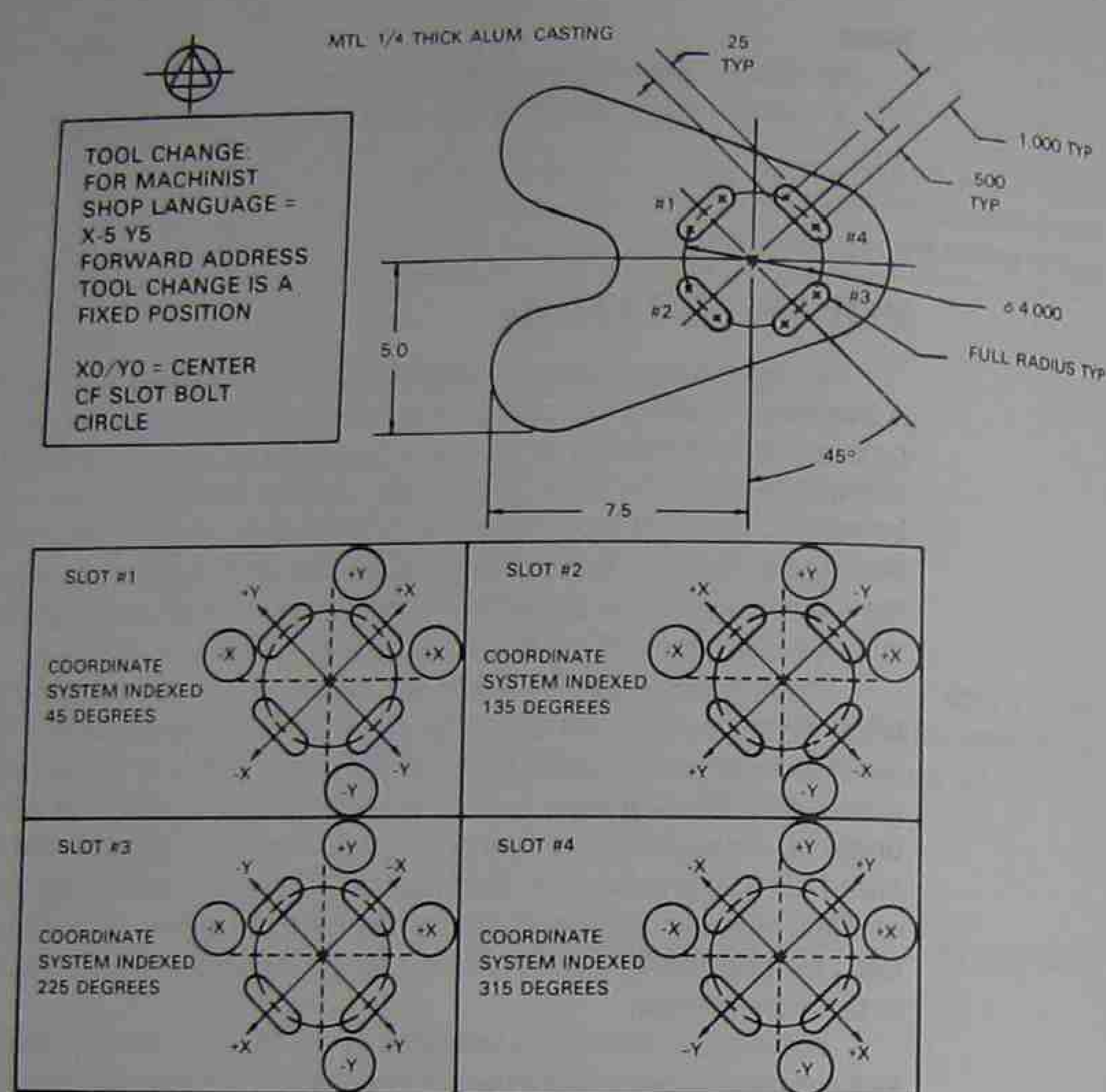


FIGURE 12–5
Part drawing

- The amount of the rotation. Following the initial rotation to the index angle, subsequent rotations may be specified as some angular value other than the index angle. The rotations will occur in a counterclockwise direction. In the case shown in Figure 12–5, this amount is 90 degrees. In other words, following the initial index of the coordinate system 45 degrees, subsequent rotations will be 90 degrees until the cancel command is given.
- A code to initiate polar rotation.
- A code to cancel polar rotation.

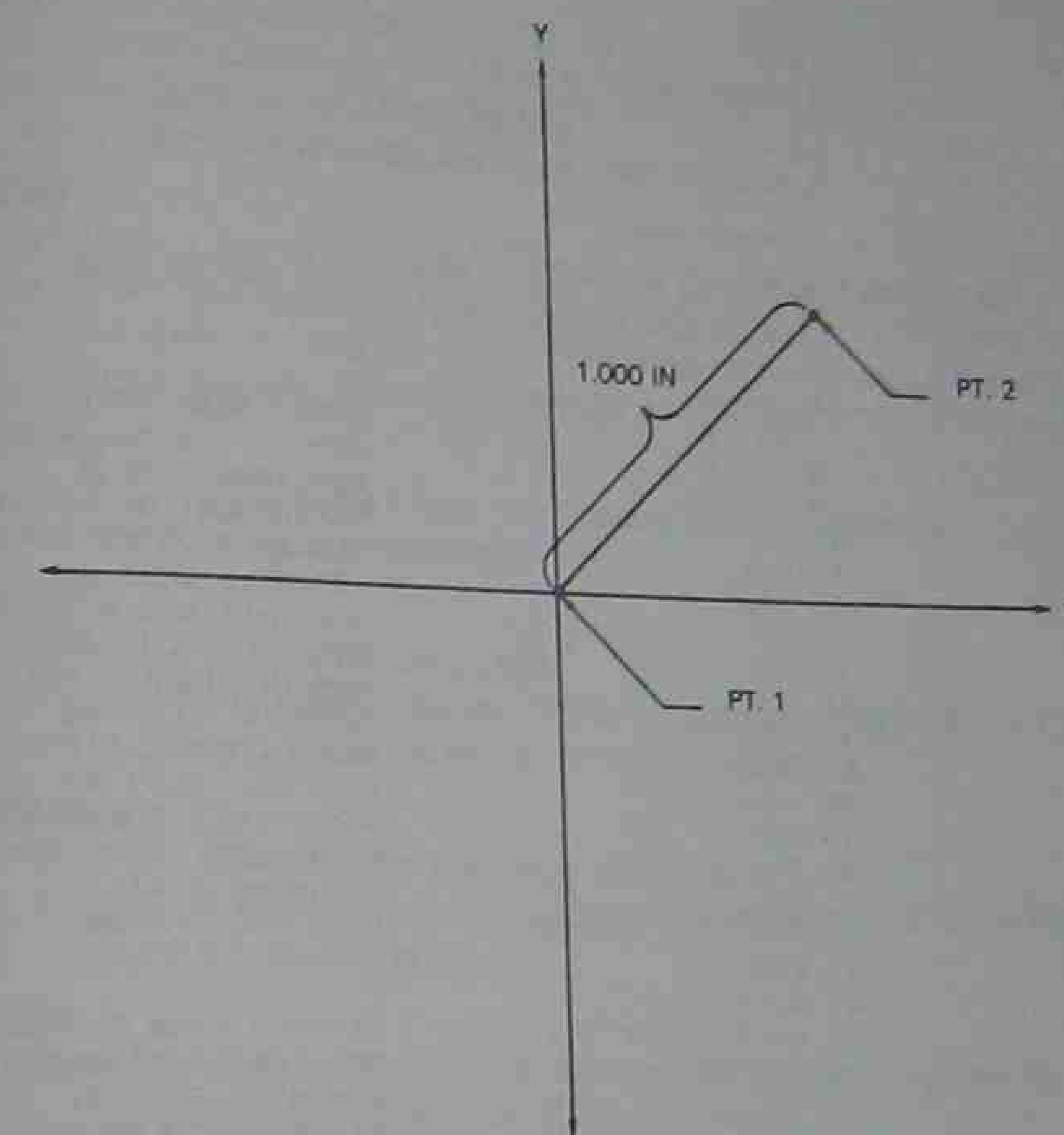


FIGURE 12–6

Machinist Shop Language

The format for instituting polar rotation in Machinist Shop Language is:

V11 — X-axis coordinate of the polar rotation.
V12 — Y-axis coordinate of the polar rotation.
V13 — Index angle.
V15 — Amount of rotation.
G51 — Code for instituting polar rotation. (Contains specific programming information)
G52 — Code for canceling the polar rotation.

The program to mill the aluminum casting shown in Figure 12–5 is presented in Figure 12–7. Only the slots need be milled.

```
XO/YO = CENTER OF SLOT ROTATION
TOOL CHANGE = X-5 Y5
TOOLS:  .250 END MILL
BUFFER:  .100 INCHES
CLEARANCE OVER CLAMPS:  3.000 IN. MIN.

 1 TOOL 1001
 2 Z3              A
 3 X-5 Y5 Z0      RA        REM:TOOL CHANGE
 4 TOOL 1
 5 FEED 28
 6 X0 Y0          RA
 7 V11 0                    REM:X AXIS POLAR CENTER
 8 V12 0                    REM:Y AXIS POLAR CENTER
 9 V13 45                   REM:INDEX ANGLE
10 V15 90                   REM:AMOUNT OF ROTATION
11 G51                      REM:INSTITUTES ROTATION
12 CALL 1                   REM:MILL SLOT #1
13 G51                      REM:INITIATE A POLAR ROTATION
14 CALL 1                   REM:MILL SLOT #2
15 G51                      REM:INITIATE A POLAR ROTATION
16 CALL 1                   REM:MILL SLOT #3
17 G51                      REM:INITIATE A POLAR ROTATION
18 CALL 1                   REM:MILL SLOT #4
19 G52                      REM:CANCEL POLAR ROTATION
20 TOOL 0                   REM:CANCEL TOOL OFFSET
21 Z0             RA        REM:HOME Z AXIS
22 X-5 Y5         RA        REM:TOOL CHANGE
23 END
24 SUBR 1
25 X-.5 Y2 Z0     RA        REM:POSITION SLOT 1
26 Z-.360         FA        REM:FEED Z TO DEPTH
27 X.5            FA        REM:MILL SLOT #1
28 Z0             RA        REM:RAISE SPINDL
29 END
```

FIGURE 12-7
Polar rotation program for part in Figure 12-5, Machinist Shop Language

PROGRAM EXPLANATION

(Refer to Figure 12-7.)

A program for this part could be written in several different ways. The example that follows is one fairly simple way to demonstrate not only the polar rotations involved but also the value of subroutine programming.

EVENTS 1-5

- These events assign the tool information and feed rate.

EVENT 6

- X0/Y0—Coordinates of the center of the slot bolt circle diameter.
- R—Specifies rapid movement.
- A—Specifies absolute positioning.

EVENT 7

- V11—Code for the X-axis coordinate of the center of the rotation.
- 0—X-axis coordinate for the rotation center. Had the X0/Y0 point been the lower left corner, the value 7.5 would have been entered with the V11. In this example, the center of rotation is conveniently the X0/Y0 point.

EVENT 8

- V12—Code for the Y-axis coordinate of the center of rotation.
- 0—Y-axis center coordinate.

EVENT 9

- V13—Code for the index angle.
- 45—Index angle in degrees. The angle is measured from the +X axis (3 o'clock position), counterclockwise to the first slot.

EVENT 10

- V15—Code for defining subsequent rotations.
- 90—All G51 commands issued after the initial one will rotate the coordinate system 90 degrees.

EVENT 11

- G51—Code to initiate a polar rotation. The coordinate system has now in effect rotated 45 degrees counterclockwise.

EVENT 12

- CALL 1—Calls up subroutine #1. The coordinates for milling the slot are contained in the subroutine. Slot #1 is milled in this event.

EVENT 13

- G51—Institutes the next polar rotation. The rotation will be 90 degrees, as specified in the V15 register, from the current coordinate system location.

EVENT 14

- CALL 1—Calls the subroutine; slot #2 is milled.

EVENT 15

- G51—Initiates the third polar rotation.

EVENT 16

- CALL 1—Calls the subroutine; slot #3 is milled.

EVENT 17

- G51—Initiates the fourth polar rotation.

EVENT 18
- CALL 1—Calls the subroutine; slot #4 is milled.

EVENT 19
- G52—Cancels polar rotation, returning the machine to its normal positioning.

EVENT 20
- TOOL 0—Cancels the active tool offset.

EVENT 21
- Z0—Retracts the spindle.

EVENT 22
- X-5 Y5—Tool change location coordinates.

EVENT 23
- END—Signals the end of the main program, resetting the computer memory.

EVENT 24
- SUBR 1—Defines the beginning of subroutine #1.

EVENT 25
- X-.5—X coordinate to position the tool at one end of the slot.
- Y 2—Y coordinate to position the tool on the slot centerline.
- Z0—Rapids the tool to the buffer zone.

EVENT 26
- Z-.36—Z-axis coordinate to feed the tool to milling depth.
- F—Specifies a feedrate move.
- A—Specifies absolute positioning.

EVENT 27
- X.5—Coordinate to mill from one end of the slot to the other.

EVENT 28
- Z0—Z coordinate to retract the spindle to the start of the buffer zone (the tool offset is active).

EVENT 29
- END—Signals the end of the subroutine.

Word Address Format

To demonstrate polar rotation in word address, the same machining strategy just demonstrated in Machinist Shop Language will be used. The coding format here is designed to be generic for the purposes of instruction. Every controller uses a different coding method for polar rotations, and many controllers

do not offer the capability. Polar rotation is used generally on three-axis machinery to compensate for the lack of a fourth rotary axis. The format for word address polar rotations used in this book is:

G61 X.... Y.... A... D... L..
Programming information
G60

Where G61 is the code to institute polar rotation, X.... is the X axis center of rotation, Y.... is the Y axis center of rotation, A.... is the index angle measured in degrees from the X axis, D.... is the subsequent amount of rotation measured in degrees, L.... is the number of rotations to be performed, and G60 is the code to cancel the rotation.

PROGRAM EXPLANATION

(Refer to Figure 12-8.)

N010-N040
- These blocks assign the tool information, speed, and feedrate and turn the spindle on clockwise.

N050
- X0 Y0—Coordinates of the bolt circle diameter of the slots. The subroutine is designed to start from this location.

N060
- Z0—Rapids the spindle to the rapid level.
- M08—Turns on the coolant.

N070
- G61—Initiates the first polar rotation. The first rotation will be to the index angle.
- X0—Defines the X0 position as the X-axis center of the polar rotation.
- Y0—Defines the Y0 position as the Y-axis center of the polar rotation.
- A45—Defines the index angle as 45 degrees.
- D90—Defines the rotations to occur after the initial rotation to the index angle as 90 degrees.
- L4—Tells the MCU that four polar rotations will be performed.

N080
- P180 M98—Instructs the MCU to jump to subroutine. Slot #1 is milled.

N090
- G61—Initiates the second polar rotation.

```
X0/Y0 = CENTER OF SLOT BOLT CIRCLE
TOOLS:  .250 DIA. END MILL
BUFFER:  .100 IN.
CLEARANCE:  3.000 IN. MIN.

N010 G00 G40 G49 G90 G80        REM:SAFETY LINE
N020 G10 H01 Z3.0000
N030 G45 H01
N040 S3500 F28 M03
N050 X0 Y0
N060 Z0 M08
N070 G61 X0 Y0 A45 D90 L4       REM:INSTITUTE 1ST ROTATION
N080 P180 M98                   REM:JUMP TO SUBR MILL #1
N090 G61                        REM:INITIATE 2ND ROTATION
N100 P180 M98                   REM:JUMP TO SUBR MILL #2
N110 G61                        REM:INITIATE 3RD ROTATION
N120 P180 M98                   REM:JUMP TO SUBR :MILL #3
N130 G61                        REM:INITIATE 4TH ROTATION
N140 P180 M98                   REM:JUMP TO SUBR MILL #4
N150 G00 G60 G49 Z0 M09         REM:RETRACT Z, CANCEL ROTATION
N160 X-12 Y8 M05
N170 M30
:180 N010 G00 X-.5 Y2 Z0        REM POSITION TO SLOT
N020 G01 Z-.36                  REM:FEED Z TO DEPTH
N030 X.5                        REM:MILL SLOT
N040 G00 Z0                     REM:RAISE SPINDLE
N050 M99                        REM:RETURN TO MAIN PRGM
```

FIGURE 12–8
Polar rotation program for part in Figure 12–5, word address format

N100
- P180 M98—Second jump to subroutine. Slot #2 is milled.

N110
- G61—Initiates the third polar rotation.

N120
- P180 M98—Third jump to subroutine. Slot #3 is milled.

N130
- G61—Initiates the fourth rotation.

N140
- P180 M98—Jumps to subroutine to mill slot #4.

N150
- G00—Selects rapid traverse mode.
- G60—Cancels the polar rotation.
- G49—Cancels the tool offset.
- Z0—Retracts the spindle.
- M09—Turns off the coolant.

N160
- X−12 Y8—Coordinates of park position.
- M05—Turns off the spindle.

N170
- M30—Signals the end of the program.

:180
- :180—Identifies this as main program block 180.
- N010—Further identifies this as subroutine block 010.
- X−.5 Y2—Polar coordinates of a slot, positioning the tool at one end.
- Z0—Rapids the spindle to the rapid level.

N020
- G01—Selects feedrate mode.
- Z−.36—Z-axis milling depth.

N030
- X.5—Polar coordinate to feed the tool from one end of the slot to the other.

N040
- G00—Selects rapid traverse mode.
- Z0—Retracts spindle to rapid level (tool offset is active).

N050
- M99—Return to main program command.

HELICAL INTERPOLATION

Helical interpolation is another useful feature of CNC machinery. *Helical interpolation* allows circular interpolation to take place in two axes (usually X and Y), while subsequently feeding linearly with the third (usually Z). This makes possible the milling of helical pockets and threads.

Figure 12–9 shows a part on which a 1.000-20 thread is to be machined. An oddly shaped part like this can be cut on a CNC machine as easily as setting it up on a face plate on a lathe or a four-jaw chuck. In the case of a production run, machining this part on the mill eliminates the need for extra fixturing. It can be threaded in the same setup used to mill it to shape. The programs presented here will assume that the part has been cast separately, however, leaving only the thread to be milled. The thread will be cut by circular interpolation with the X and Y axes, while feeding with the Z axis.

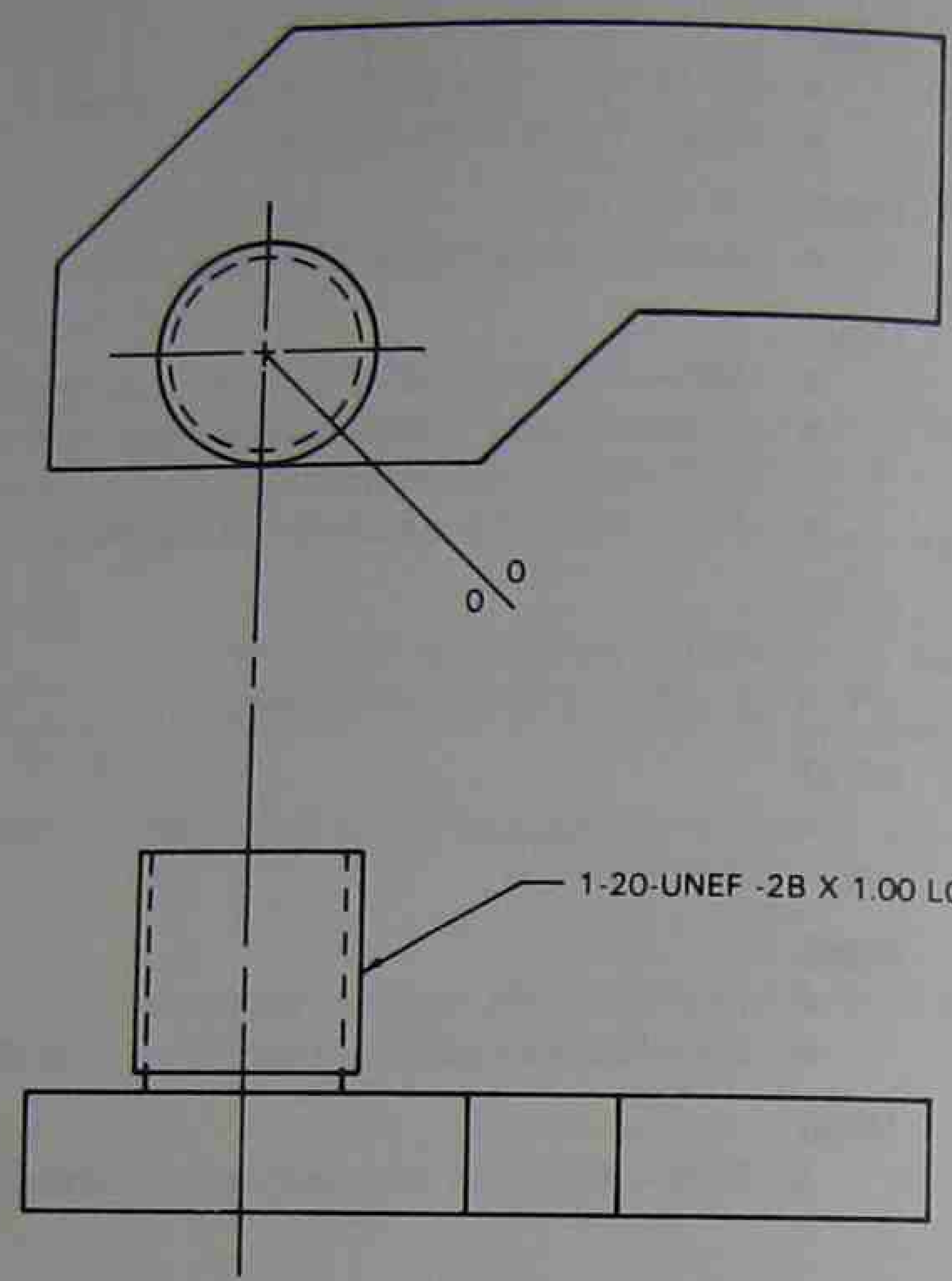


FIGURE 12–9
Part drawing

A special type of milling cutter, called a thread hob, will be used to mill the thread. The hob will be sent to a start position, fed into the workpiece, then helically interpolated for three turns. The cutter will then be withdrawn from the part. With each turn the hob makes around the part, the Z axis will advance downward an amount equal to the lead of the thread.

The lead of the thread can be determined by the formula:

$$L = P \times I$$

Where L is the lead of the thread, P is the pitch of the thread, and I is the number of leads on the thread. The pitch of a thread is 1 divided by N, where N is the number of threads per inch. For a 20 thread, the pitch is 1 divided by 20 or

.050 inch. The lead for a single lead 20 thread is .050 times 1, or .050. This means that the thread will advance .050 inch in one revolution. Note that the value of the lead and the pitch on a single lead thread are identical; however, the lead and pitch are not the same thing.

A 60-degree thread milling cutter is used to mill the thread on the part in Figure 12–9. Set up as shown in Figure 12–10.

Machinist Shop Language

The format for helical interpolation in Machinist Shop Language is:

ARC/DIRECTION—Clockwise or counterclockwise.
X.... Y....—Centerpoint of the arc.
V42—Number of 360-degree arcs to be cut (if less than 1, 0 is entered).
X.... Y.... Z....—Endpoint of the helix given in all three axes.
ARC—Code to initiate the arcs.

For the thread to be milled into this part, the coding for the helical interpolation is:

```
ARC/CW
X0 Y0
V42 2
X.9694
ARC
```

The program to mill the thread is given in Figure 12–11.

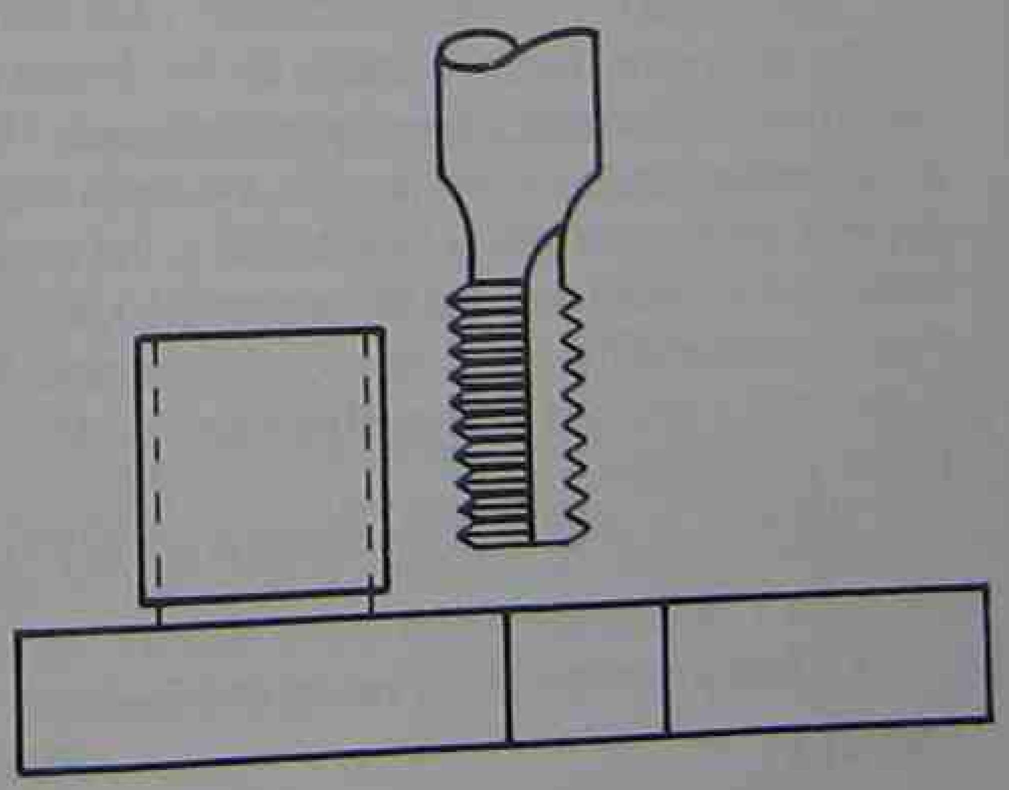


FIGURE 12–10

```
XO/YO = CENTER OF THREAD DIAMETER
TOOL CHANGE = X-2 Y2
TOOLS:   1.000 DIA. THREAD HOB
CLEARANCE:   3.000 MIN.
BUFFER:   ZERO BUFFER

 1 TOOL 1001                     REM:TOOL CHANGE
 2 X2 Z3             A
 3 X-2 Y2           RA
 4 TOOL 1
 5 FEED 7                        REM:POSITION X/Y TO START
 6 X1.6 YO ZO       RA           REM:POSTION Z TO START
 7 Z-.818           RA           REM:FEED INTO PART
 8 X.9694           FA
 9 ARC/CW                        REM:SET NUMBER OF ARCS TO 3
10 V42 3                         REM:ARC CENTER
11 XO YO                         REM:ARC ENDPOINTS
12 X.9694 YO Z.968 FA            REM:INITIATE ARC CUT
13 ARC                           REM:WITHDRAW FROM PART
14 X1.6             FA
15 TOOL 0                        REM:RETRACT Z
16 ZO               RA           REM:TOOL CHANGE
17 X-2 Y2           RA
18 END
```

FIGURE 12–11
Helical interpolation program for the thread in Figure 12–9, Machinist Shop Language

PROGRAM EXPLANATION

(Refer to Figure 12–11.)

This program mills the thread in one pass using two turns of the cutter.

To determine the coordinates for the thread depths, subtract the thread depth from half the major diameter of the thread. Thread depths can be found in a machinists' handbook. In this case, the depth of the thread is .03066. The final depth of .9694 is arrived at by subtracting the thread depth of .03066 from the radius of the thread (half the major diameter), which is .500. This leaves .4694 from the center of the arc to the root of the thread. Since the cutter is 1.000 inch in diameter, a radius of .500 must be added to .4694 to arrive at the proper cutter coordinate of .9694.

The number of turns of the thread is set to 20 using the V42 code.

EVENTS 1–5

- Assign the tool offset values and feedrate.

EVENT 6

- X1.6 Y0 Z0 RA—Position the cutter near the part.

EVENT 7

- Z –.818 RA—Positions the Z axis to the proper starting position.

EVENT 8

- X.9694 FA—Feeds the cutter into the workpiece to the thread minor diameter.

EVENT 9

- ARC/CW—Tells the MCU that a clockwise arc is to be cut.

EVENT 10

- V42—V code signaling the number of arcs to be cut.
- 3—The number arcs.

EVENT 11

- X0 Y0—Arc centerpoint coordinates.

EVENT 12

- X.9694 Y0 Z.968—Endpoint coordinates of the arc.

EVENT 13

- ARC—Initiates helical interpolation.

EVENT 14

- X1.6 FA—Withdraws the cutter from the part at feedrate.

EVENT 15

- TOOL 0—Cancels the tool offset.

EVENT 16

- Z0—Retracts the Z axis.

EVENT 17

- X –2 Y2 RA—Rapids to the tool change position.

EVENT 18

- END—Signals end of program.

Word Address Format

Not every CNC machine has helical interpolation. It is usually an optional feature, purchased at additional cost. Helical interpolation in word address can be accomplished in any one of three plane combinations (X/Y, Z/X, or Y/Z). To select the planes, the following G codes are used:

G17—X/Y plane.
G18—X/Z plane.
G19—Y/Z plane.

The format for helical interpolation in word address is as follows:

- *For the X/Y plane*—G17 G02/G03 X.... Y.... I.... J.... Z.... F...
- *For the X/Z plane*—G18 G02/G03 X.... Y.... I.... K.... Z.... F...
- *For the Y/Z plane*—G19 G02/G03 X.... Y.... J.... K.... Z.... F...

Where: G17, G18, and G19 select the plane, G02 and G03 select the direction of helical interpolation (G02 clockwise, G03 counterclockwise), X, Y, and Z are the arc endpoint coordinates, I, J, and K are the arc centerpoint coordinates, F sets the Z-axis feedrate.

To mill the part in Figure 12–9, the word address program in Figure 12–12 can be used. This program is identical in operation to the Machinist Shop Language example.

PROGRAM EXPLANATION

(Refer to Figure 12–12.)

N010–N040

- These blocks assign the tool offset and set spindle speed.

N050

- G00 X1 .6 Y0—Position the cutter to the start position.

N060

- Z–.818—Position Z to the starting depth.

```
XO/YO = CENTER OF THREAD DIA.
TOOLS:  1.000 IN. DIA. THREAD HOB
CLEARANCE:  3.000 MIN.
BUFFER:  ZERO BUFFER

N010 G80 G40 G49 G90 G98                   REM:SAFETY LINE
N020 G10 H01 Z2.0000
N030 G45 H01
N040 S800 M03
N050 G00 X1.6 YO ZO                        REM:POSITION TO START
N060 Z-.818                                REM:Z TO START POINT
N070 G01 X.9694 F7.00                      REM:FEED INTO PART
N080 G17 G02 X.9694 YO Z-.8680 IO JO       REM:1ST TURN
N090 G02 X.9694 YO Z-.9180 IO JO           REM:2ND TURN
N100 G02 X.9694 YO Z-.9680 IO JO           REM:3RD TURN
N110 G01 X1.6                              REM:FEED OUT OF PART
N120 G00 G49 ZO M09                        REM:RETRACT Z
N130 X-12 Y8 M05                           REM:TOOL CHANGE
N140 M30
```

FIGURE 12–12
Helical interpolation program for the thread in Figure 12–9, word address format

N070

- G01 X.9694 F7.00—Feeds the cutter into the workpiece to the minor diameter of the thread at a feedrate of 7 inches per minute.

N080

- G17—Selects the XY plane for interpolation.
- G02—Selects clockwise interpolation.
- X.9694 Y0 Z–.8680—The endpoint coordinates of the first arc.
- I0 J0—The centerpoint coordinates of the arc.

N090

- G02—Selects clockwise interpolation.
- X.9694 Y0 Z–.9180—The endpoint coordinates of the second arc.
- I0 J0—The centerpoint coordinates of the arc.

N100

- G02—Selects clockwise interpolation.
- X.9694 Y0 Z–.9680—The endpoint coordinates of the third arc.
- I0 J0—The centerpoint coordinates of the arc.

N110

- G01 X1 .6—Withdraws the cutter from the part at feedrate.

N120

- G00 G49 Z0—Cancels the tool offset, retracting the Z axis at rapid.
- M09—Turns off the coolant.

N130

- X–12 Y8—Rapids to tool change position.
- M05—Turns off the spindle.

N140

- M30—Signals end of program.

SUMMARY

The important concepts presented in this chapter are:

- Mirror imaging means changing the sign (+ or –) of an axis movement.
- Mirror imaging is used in a program to save repetitive programming when the direction of movement is the only difference between part features.
- Mirror imaging is normally used in conjunction with subroutines or do loops.
- Polar rotation is an indexing of the NC machine's Cartesian coordinate system to some angle other than its normal state.

- Polar rotation may be used to perform operations that otherwise would require the use of a rotary axis or lengthy coordinate calculations.
- Polar rotations may be used in conjunction with do loops or subroutines.
- Helical interpolation is circular interpolation with two axes while simultaneously feeding at a linear rate with the third. The result of this type of operation is a helix.
- Care must be taken in calculating the number of turns and the lead of a helix, be it a thread or other type of part.
- Helical interpolation may be used inside of or in conjunction with do loops and subroutines.

REVIEW QUESTIONS

1. What is mirror imaging? Why is it used?
2. When would mirror imaging be used in a program?

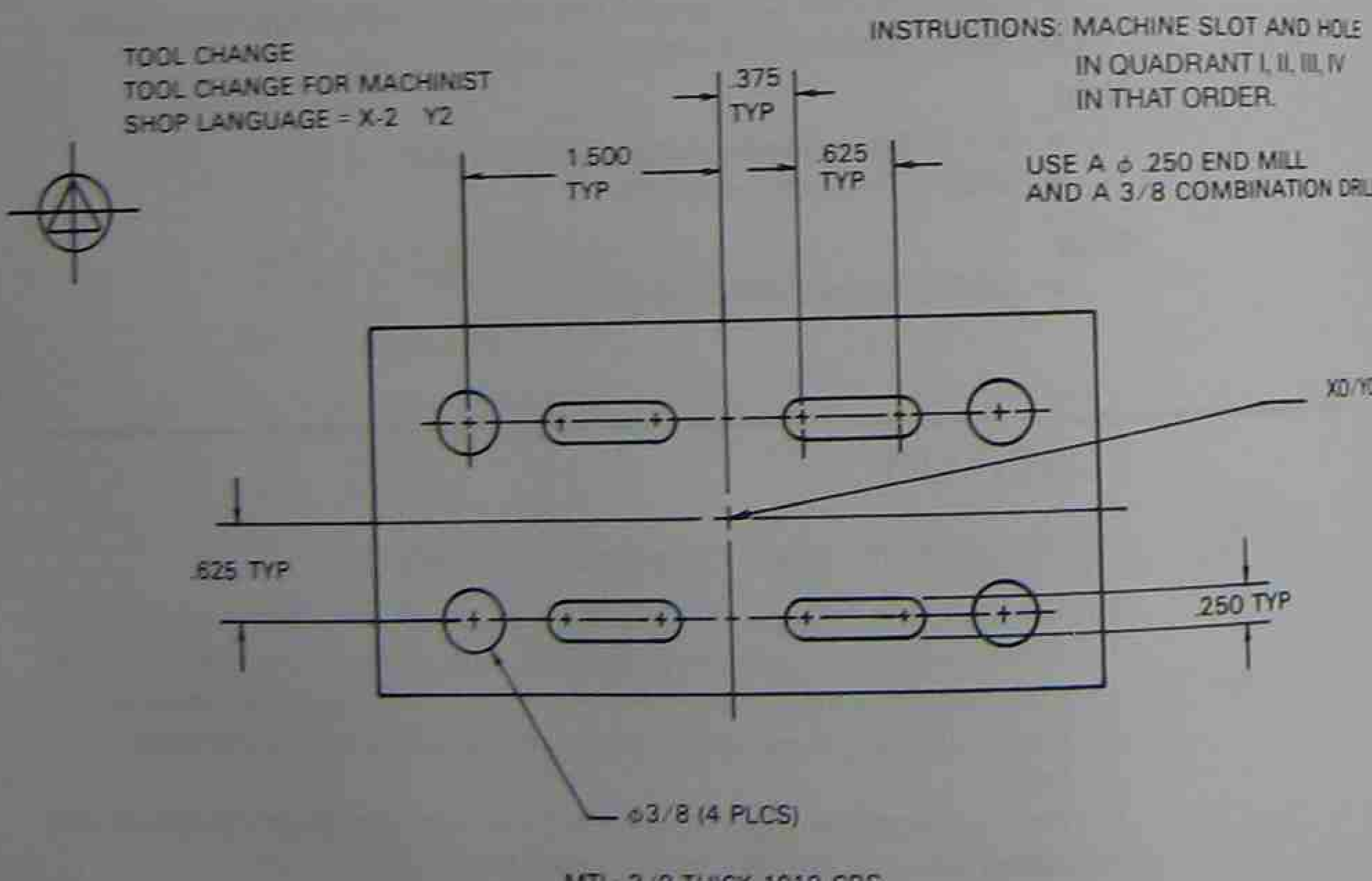


FIGURE 12–13
Part drawing for review question #3

3. Write a program to mill the slots and drill the holes in the part shown in Figure 12–13:
   a. In Machinist Shop Language.
   b. In word address.
4. What does polar rotation do?
5. What types of equipment can polar rotations substitute for?
6. Can polar rotations be used with subroutines and do loops?
7. What type of information must be given in the program for the MCU to perform polar rotation?
8. Write a program to mill the slots in the part shown in Figure 12–14:
   a. In Machinist Shop Language.
   b. In word address.
9. What is helical interpolation?
10. When would helical interpolation be useful in a program?

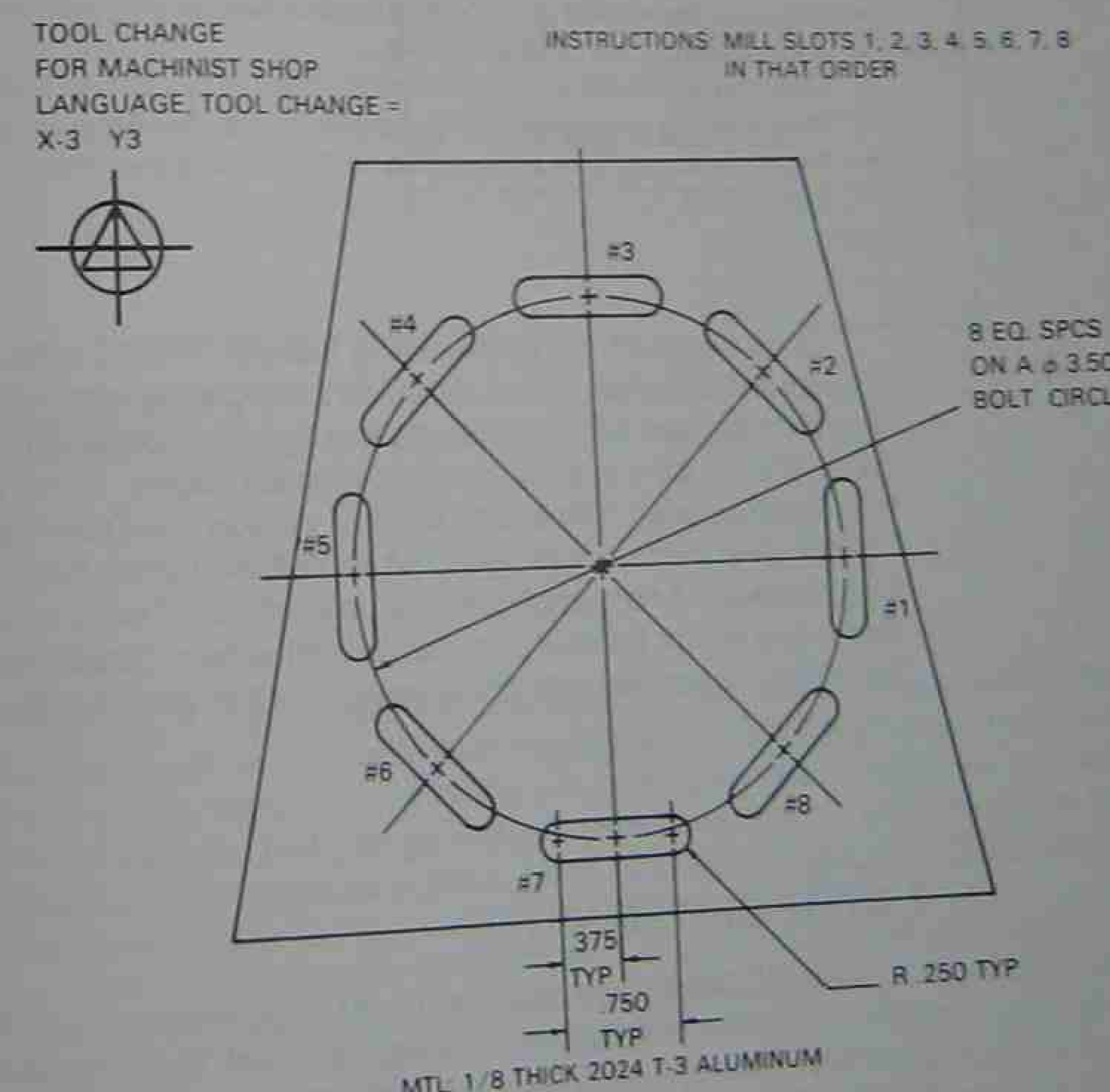


FIGURE 12–14
Part drawing for review question #8

CHAPTER 13

The Numerical Control Lathe

OBJECTIVES Upon completion of this chapter, you will be able to:

- Describe the difference between a conventional lathe bed arrangement and a slant bed arrangement, listing the advantages of the slant bed for NC.
- Explain axis movement on a CNC lathe.
- Describe the method of toolholding used on CNC turning machines.
- Explain what a tool offset number is.
- Describe two methods of tool selection used on CNC turning machines.
- Describe how spindle speed is designated on gear head and variable speed lathes.
- Explain how feedrates are specified on CNC turning equipment.
- Define TNR.

Up to this point, the programming features of CNC mills have been discussed, but numerical control is used for turning equipment as well. In the milling examples, both Machinist Shop Language and word address formats were given. For the turning programs discussed in Chapter 14, only word address format will be used. The coding will be a version used with Fanuc lathe controllers, designed to be generic and so to illustrate the basic programming steps involved. A numerical control lab in a school will have equipment that differs in one way or another from that presented here. Students are advised to familiarize themselves with the codes used for the machines they will be using.

LATHE BED DESIGN

Older NC lathes, and those that have been converted to numerical control with retrofit units, look like traditional engine lathes. The lathe carriage rests on the ways. The ways are in the same plane and are parallel to the floor, as illustrated in Figure 13–1. This arrangement allows the machinist to reach all the controls readily. Since the CNC lathe performs its operations automatically,

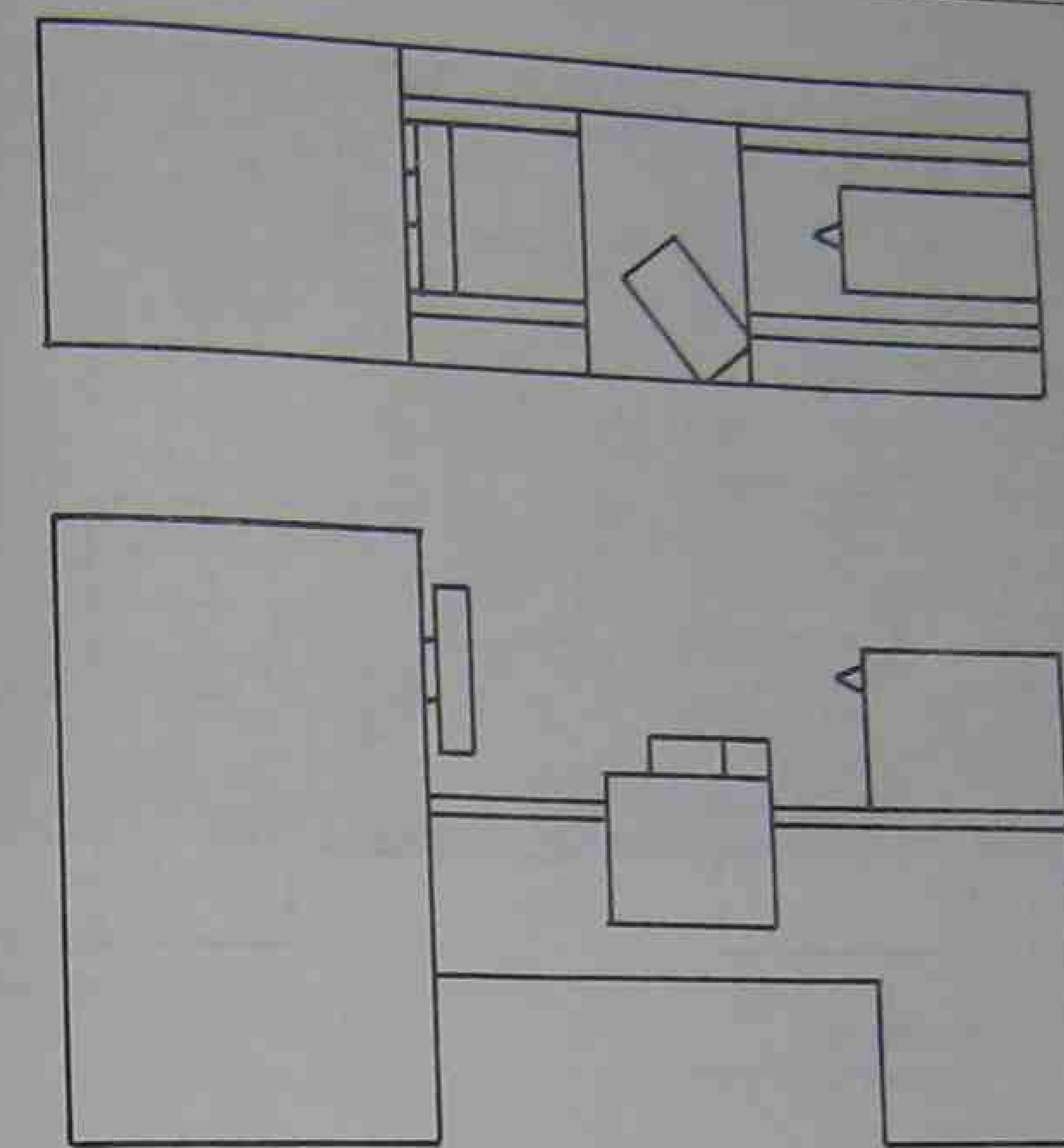


FIGURE 13–1
Bed arrangement on a conventional lathe

this type of arrangement is not necessary. In fact, it is quite awkward, as the operator will be busy with other responsibilities while the program is running and so will not necessarily be there to brush the chips off the ways. In a conventional lathe bed arrangement, the chips have nowhere to fall except on the ways. To overcome this problem, many CNC lathes make use of the slant bed design illustrated in Figure 13–2.

On many NC lathes, the turret tool post is mounted on the opposite side of the saddle, compared to a conventional lathe, to take advantage of the slant bed design. The slant bed allows the chips to fall into the chip pan, rather than on tools or bedways. Despite its odd appearance, the slant bed NC lathe functions just like a conventional lathe. Figures 13–3 and 13–4 show modern CNC turning machines. Notice the slant bed arrangement.

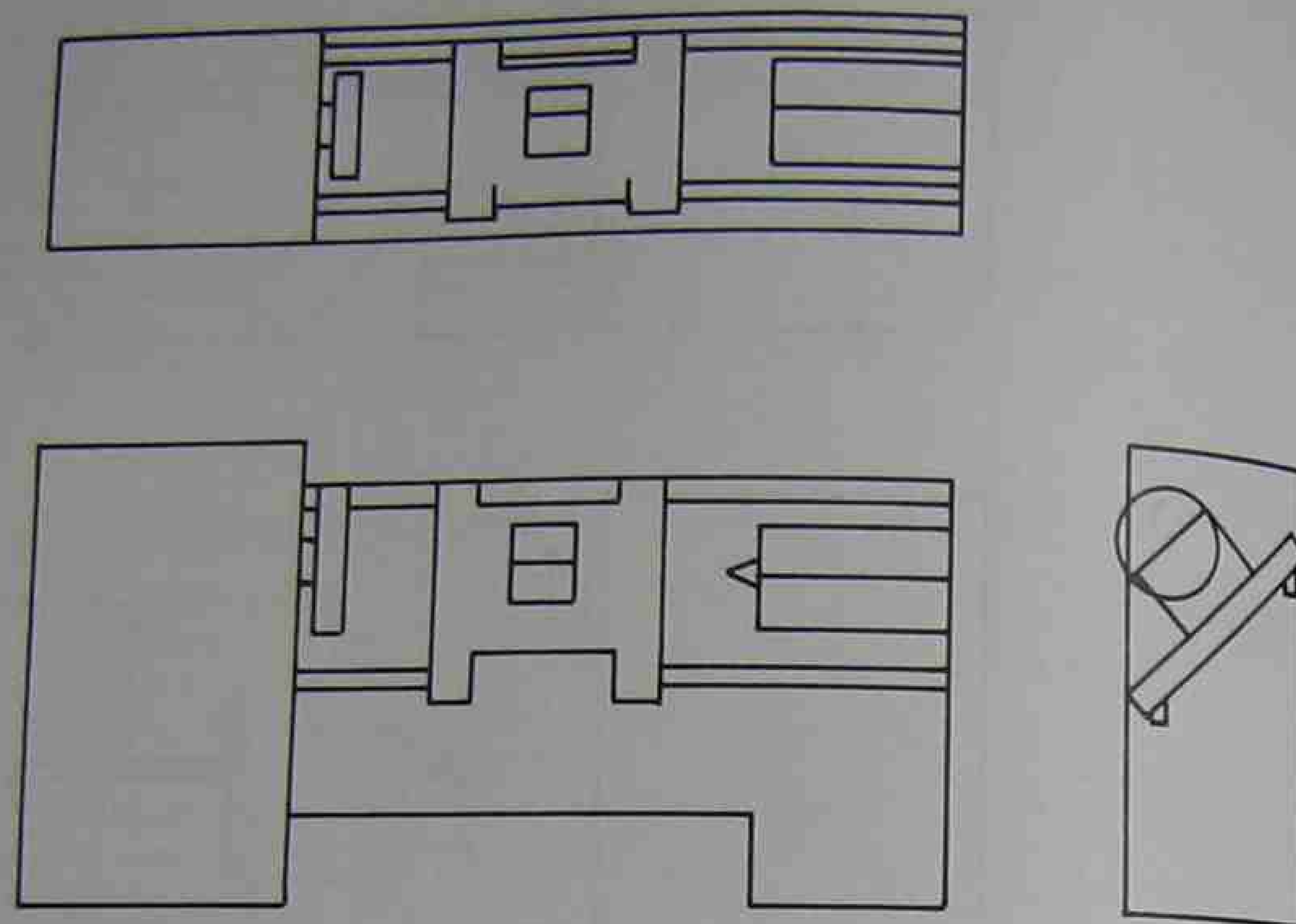


FIGURE 13-2
Slant bed for NC or CNC lathe

FIGURE 13-3
A modern CNC turning center employing automatic tool change *(Photo courtesy of Lodge and Shipley Co.)*

FIGURE 13-4
A four-axis CNC turning center *(Photo courtesy of Cincinnati Milacron)*

AXIS MOVEMENT

The axis movement of a basic CNC lathe is diagrammed in Figure 13-5. Some turning machines, such as that shown in Figure 13-4, are four-axis machines. In this book, only the basic two-axis machine is programmed. The programming concepts learned on a two-axis machine are the foundation necessary to program more complex machinery.

The basic lathe has only two axes, X and Z. Since the Z axis is always parallel to the spindle, longitudinal (carriage) travel is designated Z. The cross slide movement is designated X, since it is the primary axis perpendicular to Z. If it were possible to move the carriage up and down, that axis would be Y. There is, however, a potential problem with this arrangement. There appear to be two Z axes: the carriage movement and the tailstock movement. To eliminate this problem the tailstock is usually called the W axis on lathes with programmable tailstocks. Programmable tailstocks, which are rear turret assemblies on CNC equipment, are the third and sometimes fourth axis on more complex equipment. The turning center in Figure 13-4 has two programmable saddles. In such cases the axes of the second saddle are usually designated W and U, with

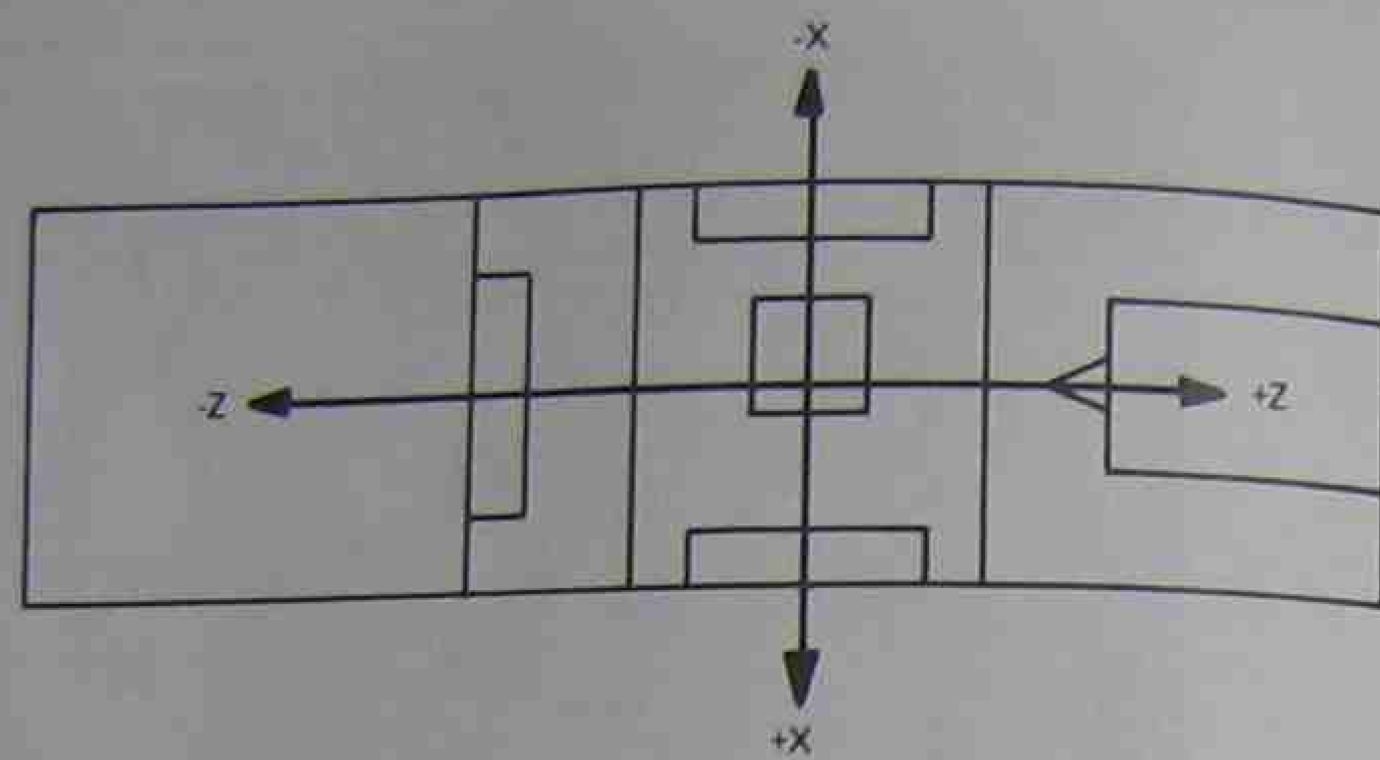


FIGURE 13–5
Lathe axis movement

W being saddle travel and U being cross slide travel. There are some imported lathes on which the X-axis direction is reversed. The programmer must determine if such a situation exists before writing the lathe program.

TOOLHOLDERS AND TOOL CHANGING

Either a rigid toolholder or a tool turret is used to hold the tools on an NC lathe. Figure 13–3 shows a CNC chucker employing a rigid toolholder. The turning center in Figure 13–4 employs a tool turret, in which the various tools needed for lathe operations are placed in toolholders. When a tool change is necessary, the appropriate turret is indexed to the next tool needed. Simple lathes use six-sided turrets; larger turning machines use eight-, ten-, and twelve-sided turrets.

With the development of robotics, new tool changing and work handling schemes are appearing. Figure 13–6 shows a robot arm used for handling workpieces, and Figure 13–7 illustrates the robot in operation. To teach the basics of CNC programming, this text will focus on nonrobotic tool change.

The toolholders used on NC turning machines are of very rigid design. The tools used for turning are of the carbide insert type, made to much more exacting tolerances than conventional lathe insert tooling.

A tool change command in a turning program either changes the turret position or causes an automatic tool change, depending on the type of machine used.

FIGURE 13–6
A robot arm used for part load and unload *(Photo courtesy of Cincinnati Milacron)*

FIGURE 13–7
A robot arm in action *(Photo courtesy of Cincinnati Milacron)*

Automatic Tool Change

In a CNC turning program for a machine with a rigid toolholder, M06 is used to initiate an automatic tool change. The T address is used (as it is in milling programs) to specify the desired tool. The T address also calls up the tool offsets. The format for automatic tool change is:

M06 T n1 n2

Where M06 initiates the tool change, T is the tool address, n1 is the tool number, and n2 is the tool offset number.

Turret Position

T is used in a similar manner with turret tool selection. The format is:

T n1 n2

Where the first number is the turret position and the second is the tool offset number.

Since one tool may be used in several positions, a turret position is used rather than a tool number. The turret position corresponds to the turret station number. T01 will index the tool in station one into position. Some NC lathes can utilize more than one tool on a single station. It is possible, therefore, for T0101 to refer to one tool and T0111 to refer to another. This is referred to as piggy-backing a tool station.

One other point should be kept in mind when changing tools: the carriage (or tailstock) does not necessarily move to a tool change location. It is often necessary, therefore, first to move the carriage or tailstock turret out of the way before making a tool change. It may also be necessary to program a dwell (G04) to halt the program, giving the tool time to index to position safely.

Tool Offset Numbers. Each turning tool used on a lathe has a radius. When programming the coordinates for a location, the centerline of the tool radius is programmed. Tool offsets allow the center of the tool nose to be programmed and thus compensate for minor differences in length that exist between tools, and the effects of tool wear.

When the tools are set up, the operator enters the offsets into the tool registers. When the offsets are active, the MCU compensates for them, eliminating the need for premeasured tooling. In this manner, the programmer can treat all tools as being the same length, just as in milling.

The *offset number* is the number of the register in which a particular tool's offset is stored. Generally the register number will match the tool number.

The tool information entered manually prior to the start of the program is entered in this form:

X Offset
Z Offset
Tool Nose Radius
Standard Tool Nose Vector Number

Tool Nose Radius and Standard Tool Nose Vector Numbers

The tool nose radius and tool nose vector numbers are optional. They are entered if using cutter comp. Cutter diameter compensation is called *tool nose radius compensation* (TNR comp) on turning machines. The tool radius tells the MCU the amount of compensation that is to be used. With NC machining centers this value was entered in a comp register.

TNR comp is utilized just as cutter comp was in Chapter 10. It can be used to program the part line, or fine tune the tool path to compensate for tool wear. The major difference is lathe tools are not completely circular as is a milling cutter. To aid in proper compensation of the tool path and correctly identify alarm conditions, a tool nose vector number is entered in the register. Tool nose vector numbers tell the MCU the orientation of the tool nose. Figure 13–8 shows the various directions in which a tool may be oriented. These directions are referred to as vectors. Each vector has a number associated with it that is used to describe the tool orientation to the MCU.

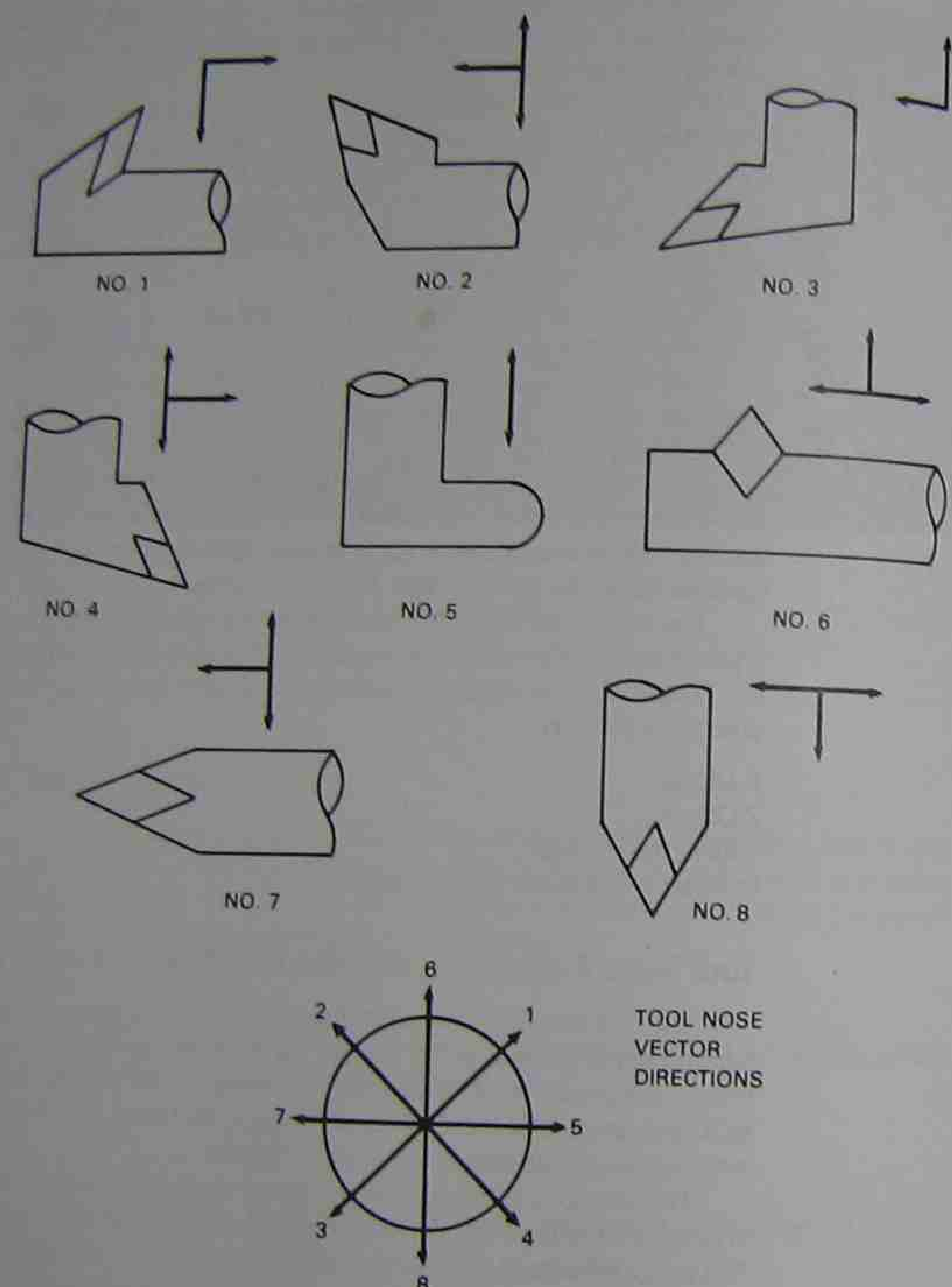


FIGURE 13–8
Tool nose vectors

Tool Edge Vs Centerline Programming

The tool nose may be programmed in one of two ways when TNR comp is not active: by the tool edge or by the tool nose radius centerline.

Tool edge programming is adequate for simple straight line cuts where the part surfaces intersect each other at right angles. Problems are encountered, however, when angles and especially arcs are programmed this way. Figure 13–9A illustrates this point. If the tool edge is programmed, the I and K centerpoints of the illustrated arc must be shifted. This results in a tool path that does not follow the desired arc exactly. The amount of error that is induced depends upon the size of the cutter and the radius of the arc. In any case, tool edge programming should not be used when encountering arcs and angles.

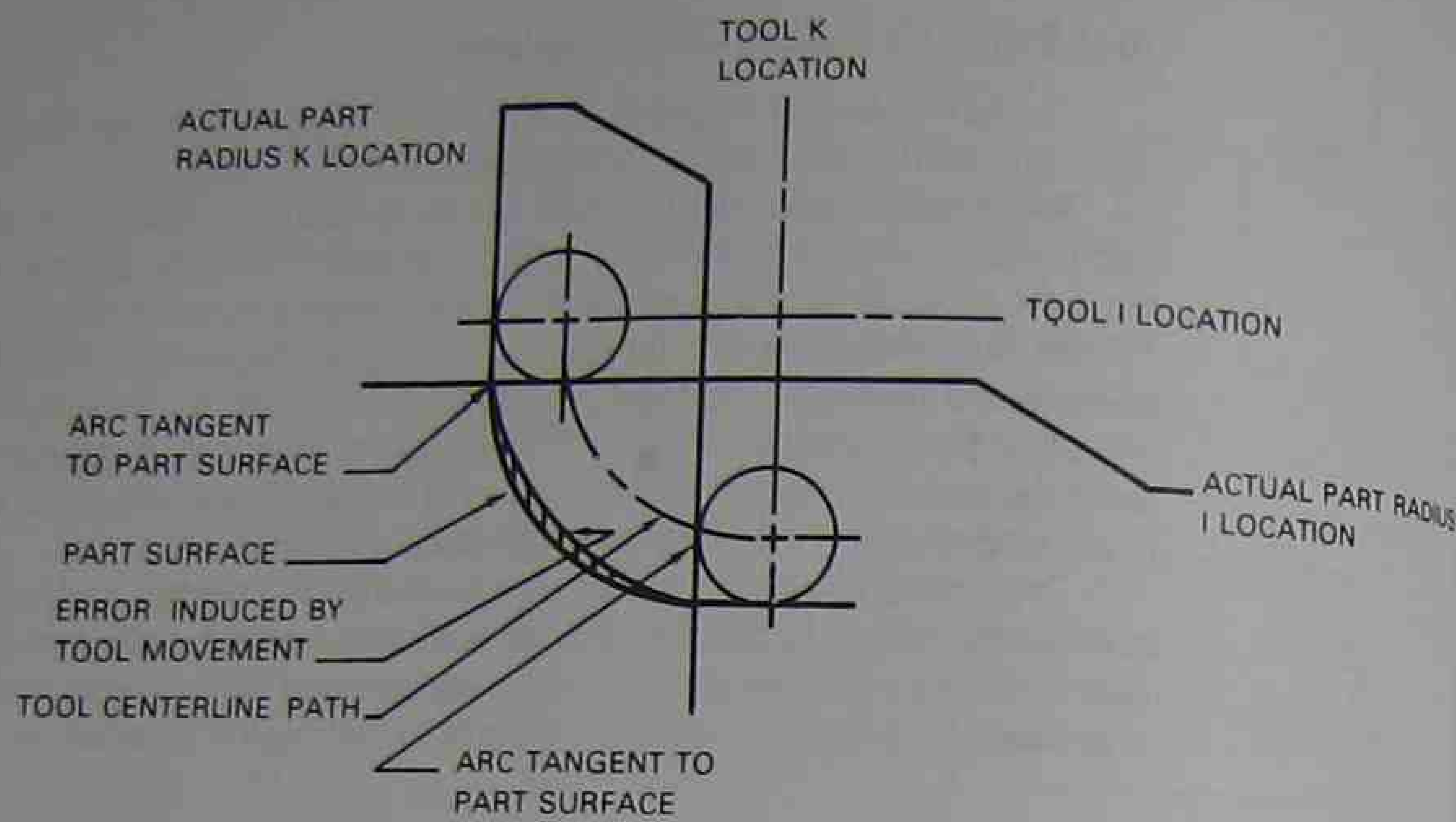
Tool centerline programming is identical to the centerline programming done when milling. Figure 13–9B demonstrates how the cutter centerlines and part surface centerlines coincide when the center of the tool nose radius is programmed. This type of programming is demonstrated in Chapter 14.

SPINDLE SPEEDS

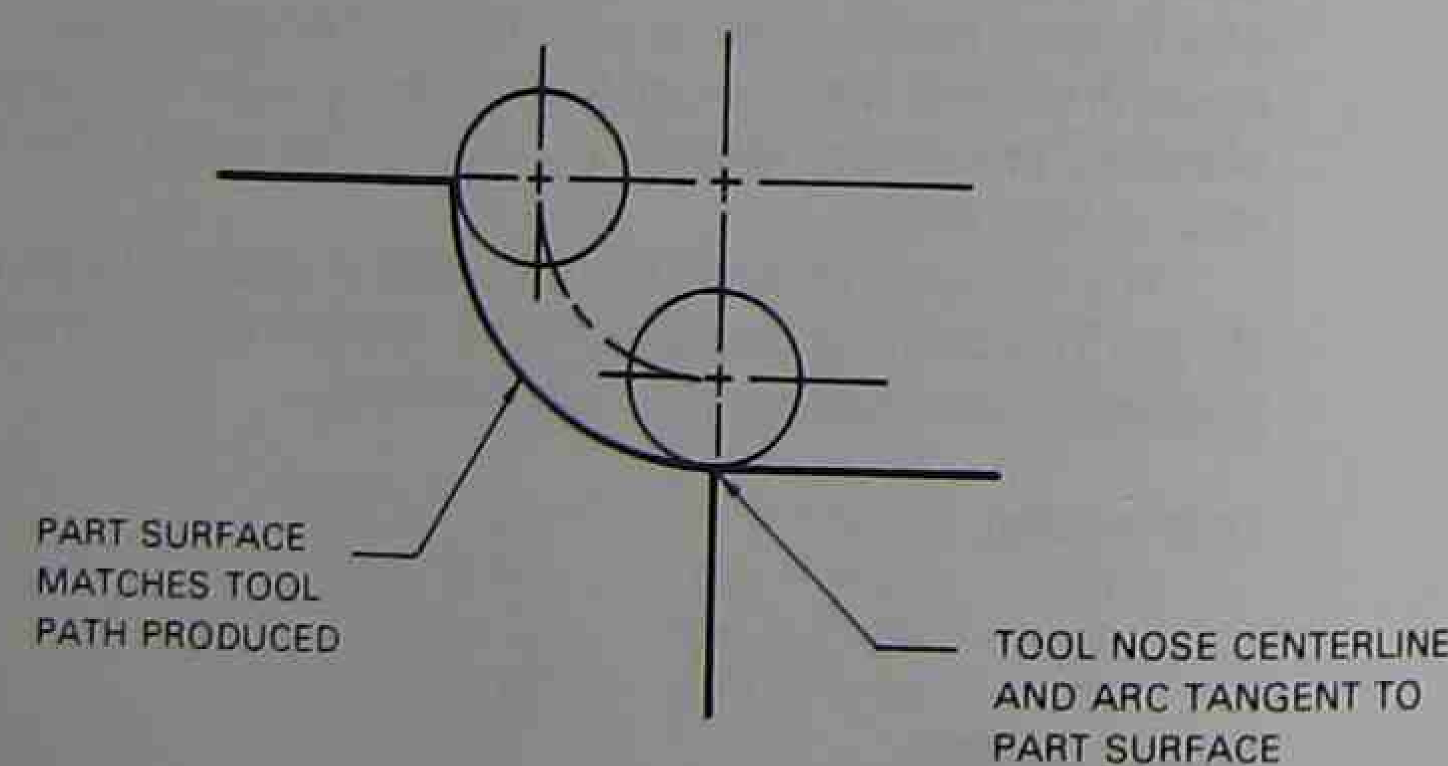
Spindle speed is specified using an S address, just as in milling. On turning machines with a gear head design, the spindle speed is changed by shifting gears in the headstock. On gear head machinery, there are usually two or more gear ranges. An M function is used to select the gear range in which the desired speed is located. M40 through M46 generally serve this purpose. For gear head examples in this text, M40 will be used for low range, M41 for mid range, and M42 for high range.

The following chart shows a sample of speed ranges for gear head machines. This chart is not for a particular machine but is representative of the type of spindle speed spread found on a machine.

LOW RANGE			
10	15	20	25
30	40	50	65
75	90	110	125
MEDIUM RANGE			
55	70	95	120
140	155	175	200
235	260	290	300
HIGH RANGE			
285	335	380	450
530	660	900	1200
1800	2100	2500	3000



A. Error induced by programming tool edge



B. Tool nose centerline programming

FIGURE 13–9

Some CNC turning machines use a variable speed drive with which an infinite number of speeds are available between the highest and lowest speeds. In these cases the speed is selected using the S address as it is in milling.

FEEDRATES

With a CNC lathe, assigning feedrates is quite simple. A G98 or G94 code (depending on the controller) tells the MCU that the following feedrate is in inches per minute. For example, G98 F7 specifies a feedrate of 7 inches per minute. A G99 or G95 in a turning program specifies a feedrate in inches per revolution. For example, G99 F.015 specifies a feedrate of .015 inch per revolution. Appendix 3 contains a list of common G codes used in lathe programming.

MACHINE ORIGIN AND WORK COORDINATE SYSTEMS

An NC lathe generally has a fixed-zero position assumed by the executive program upon power-up. This position is known as the home zero or machine origin. The physical location of this position varies from controller to controller and machine model to machine model. It is usually one of two locations: X0 = centerline of the spindle, Z0 = the chuck mounting surface of the spindle, or X0 = extreme X+ location, Z0 = extreme Z+ location. It is usually necessary to establish a zero point on the part different from the machine origin location. This position is called the *work coordinate system* or *part zero*. There are two methods used to accomplish this.

The first method involves the use of an axis preset command—G50. The G50 transfers the zero point from the home zero to the coordinates specified with the command. The format for a G50 command is:

G50 Xxx.xxxx Zzz.zzzz

Where: G50 = the axis preset command
xx.xxxx = the X-axis distance to the part zero
zz.zzzz = the Z-axis distance to the part zero

A G50 command is issued at the start of each tool. Since the programmer will not know the axis preset distances in advance, zeros should be used or some other prearranged value in the G50 line. The actual values will be determined by the setup person and edited in the control when the job is set up.

The second method uses registers called *work coordinates*. These are registers in the MCU that tell the MCU the distance from home zero to the part zero. If a machine has more than one available work coordinate, multiple zero points may be used for complex programming. Another advantage to multiple work coordinates is the ability to have more than one program loaded in the MCU, each with its own work coordinate. This is a decided advantage when running several repeating jobs through a turning center. Each work coordinate is called by a G code. If a program were to use four work coordinates, they would be selected by the codes G54, G55, G56, and G57. The first work coordinate (G54 in this case) is the default work coordinate. This work coordinate is automatically activated upon power-up. If using only the default work coordinate, the G code may be omitted. The work coordinate values are entered by the setup person when the job is prepared. The programmer must instruct the setup personnel the position on the part of the part zero location.

QUICKSETTERS

A fairly recent development has been the use of quicksetters—arms with tool sensors on them. During job setup the arm is lowered into position, the operator jogs a tool to the presetting position and touches it off on the sensor. The quicksetter automatically sets the values of the work coordinate and the tool offset registers.

SUMMARY

The important concepts presented in this chapter are:

- CNC turning machines often use a slant bed arrangement to protect the machine ways from chips. Although different in appearance, the functioning of a slant bed and conventional bed machine is identical.
- There are two basic axes, X and Z, on a CNC lathe. If the lathe has additional axes, they are generally designated U and W.
- TNR stands for tool nose radius compensation. TNR is the equivalent in CNC turning to cutter diameter compensation in milling.
- A tool turret or a rigid toolholder is used to hold the tools on an NC lathe.
- Tool offsets are entered into the MCU prior to running the program to compensate for minor setup adjustments.

- A standard tool nose vector number is used to identify the orientation of a particular tool when using TNR.
- A tool change command in turning programs will either change the turret position or cause an automatic tool change, depending on the type of machine used.
- The tool change format for turret changing is: T n1 n2
  Where T is the tool change command, n1 is the turret position and n2 is the tool offset number.
- The format for automatic tool change is: M06 T n1 n2
  Where M06 initiates the tool change, T is the tool address, n1 is the tool number, and n2 is the tool offset number.
- Spindle speeds are specified directly using the S address. On gear head machines, it is necessary to specify the gear range when selecting a range outside the active one.
- Feedrates on CNC lathes can be specified either in inches per minute (using G94 or G98), or in inches per revolution (using G95 or G99).
- To set a part at X0/Z0 point, it is necessary to transfer the machine origin to the workpiece using a G code.

REVIEW QUESTIONS

1. What is the difference between a slant and conventional lathe bed arrangement? What is the advantage of a CNC slant bed lathe?
2. Draw a sketch illustrating the axis movement on a lathe.
3. What type of toolholding is used on CNC turning machines?
4. What is the purpose of a tool offset register?
5. What is a standard tool nose vector number?
6. What types of turrets in addition to the four-sided turret are used on CNC lathes?
7. How are spindle speeds designated on CNC turning machines? What additional coding is required on gear head machines?
8. How are feedrates specified on CNC lathes? What codes are used?
9. What is the format for a turret tool change command? For an automatic tool change?
10. What does TNR stand for?
11. What is the machine origin?
12. How is an X0/Y0 point established on a workpiece?

CHAPTER 14

Programming CNC Turning Machines

OBJECTIVES Upon completion of this chapter, you will be able to:

- Write simple turning and facing routines.
- Write simple taper turning routines.
- Write simple routines to perform circular interpolation, using programmed arc centers and programmed radius value methods.
- Write simple thread-turning routines using single pass and multipass threading.

CNC lathe controllers vary in their coding to an even greater extent than mill controllers. It is, therefore, difficult to discuss programming practices. EIA standards specify axis movement, for example, but some lathes use a left-hand coordinate system, with the X and Z axes reversed from the standard configuration. Other lathes reverse the X axis direction and not the Z. On lathes using twin turrets, the X axis is often reversed. The uses of coding and the cycles available also differ to a large extent. The EIA codes pertaining to lathes are generally used, but many other codes may be added.

This chapter will discuss basic lathe programming routines for turning, facing, taper turning, circular interpolation, and thread cutting. Each routine is placed in a miniprogram. Each program can be thought of as a building block; to machine a complete part, these building blocks can be linked together in one program as will be demonstrated.

MACHINE REFERENCE POINT

A machine *reference point* is a fixed position on the machine. Upon receiving the proper G code, the machine automatically returns to the reference point location. This point is often the home zero location, used for tool changing and as a park position at the end of the program. Often it is necessary to send the tool back to the reference point by way of another point, called an *intermediate*

point. The code used in this chapter to return the tool to reference is G28. An X and Z coordinate for the intermediate point are specified along with the G28. Upon receiving the G28, the tool moves to the intermediate point and then proceeds to the reference point.

DIAMETER VS RADIUS PROGRAMMING

The difference between radius programming and diameter programming is an important one. *Diameter programming* references the X-axis coordinate to the diameter of the workpiece. This means that every .001 inch programmed moves the tool .0005 inch as measured radially. If the X axis advances .500 inch into the part, .500 inch is removed from the diameter. To accomplish this, the X axis moves only .250 inch, or half the programmed amount.

In *radius programming*, the X axis moves the programmed amount. If .500 inch of movement along the X axis is programmed, the tool advances .500 inch. When the Z-axis move is made, 1.000 inch of material is removed from the part.

Canned cycles on a machine call for the information to be entered in either diameter or radius coordinates, depending on the cycle's function. The machine manual must be consulted to determine the type of coordinate expected. The coordinates may be either incremental or absolute, depending on whether G90 or G91 is active. As in milling, G90 selects absolute positioning and G91 selects incremental. Other controllers use a "W" address for incremental X and a "U" address for incremental Z.

TURNING AND FACING

Figure 14–1 shows a part to be turned and faced in a lathe. Note that the position of the tool turret relative to the X0/Y0 location and the machine origin is given. The machine coordinate system may be transferred to the workpiece either within the program by use of G codes or by the operator during machine setup. It is usually more efficient to define the work coordinate system during setup. For routines in this chapter, this will be assumed. Figure 14–2 shows a part similar to the one in Figure 14–1 but with metric dimensions. Figure 14–3 presents a short program to turn and face the part drawn in Figure 14–1. Figure 14–4 presents a metric version. To program this part, the following codes will be used:

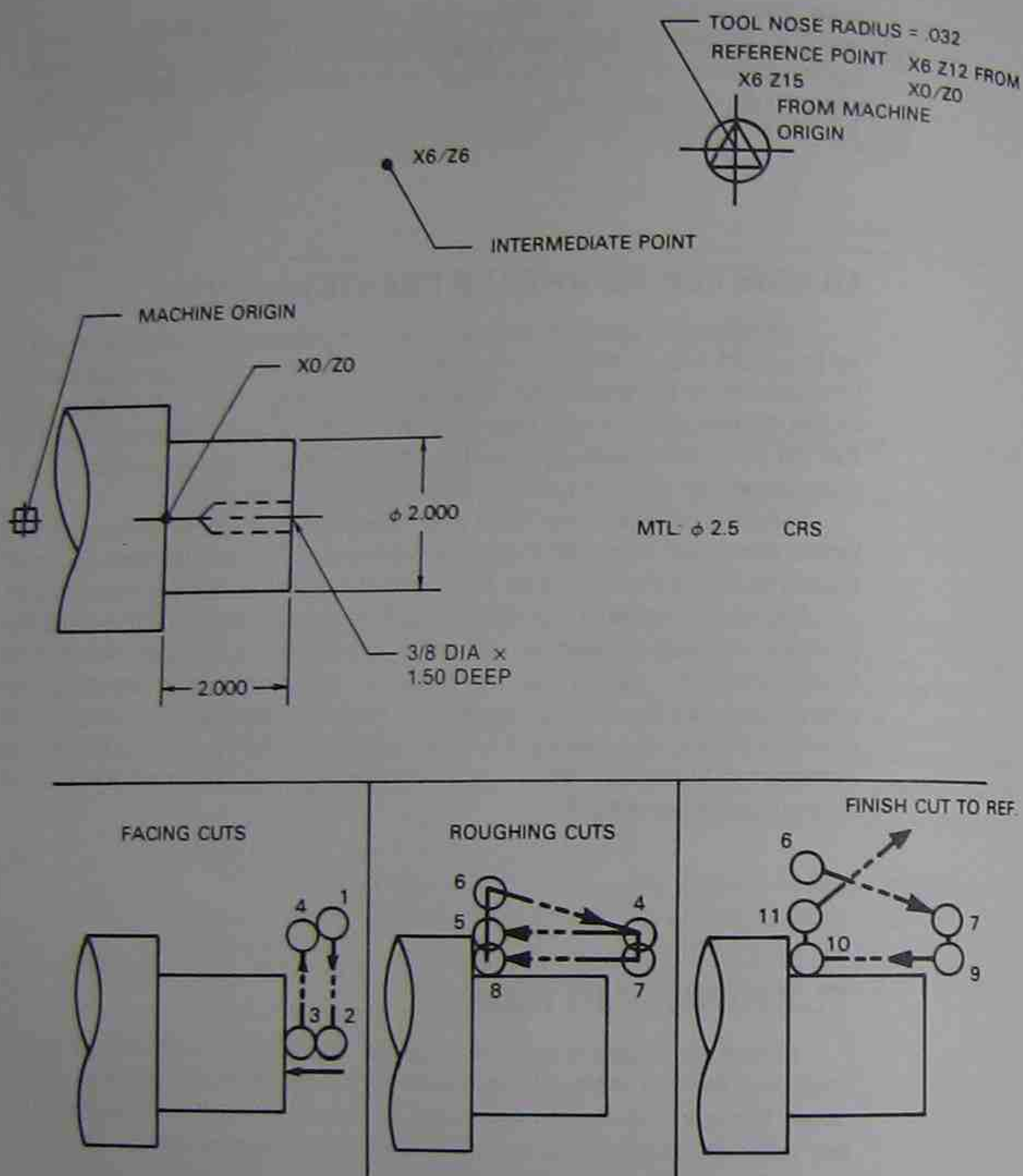


FIGURE 14–1
Part to be turned and faced in a lathe

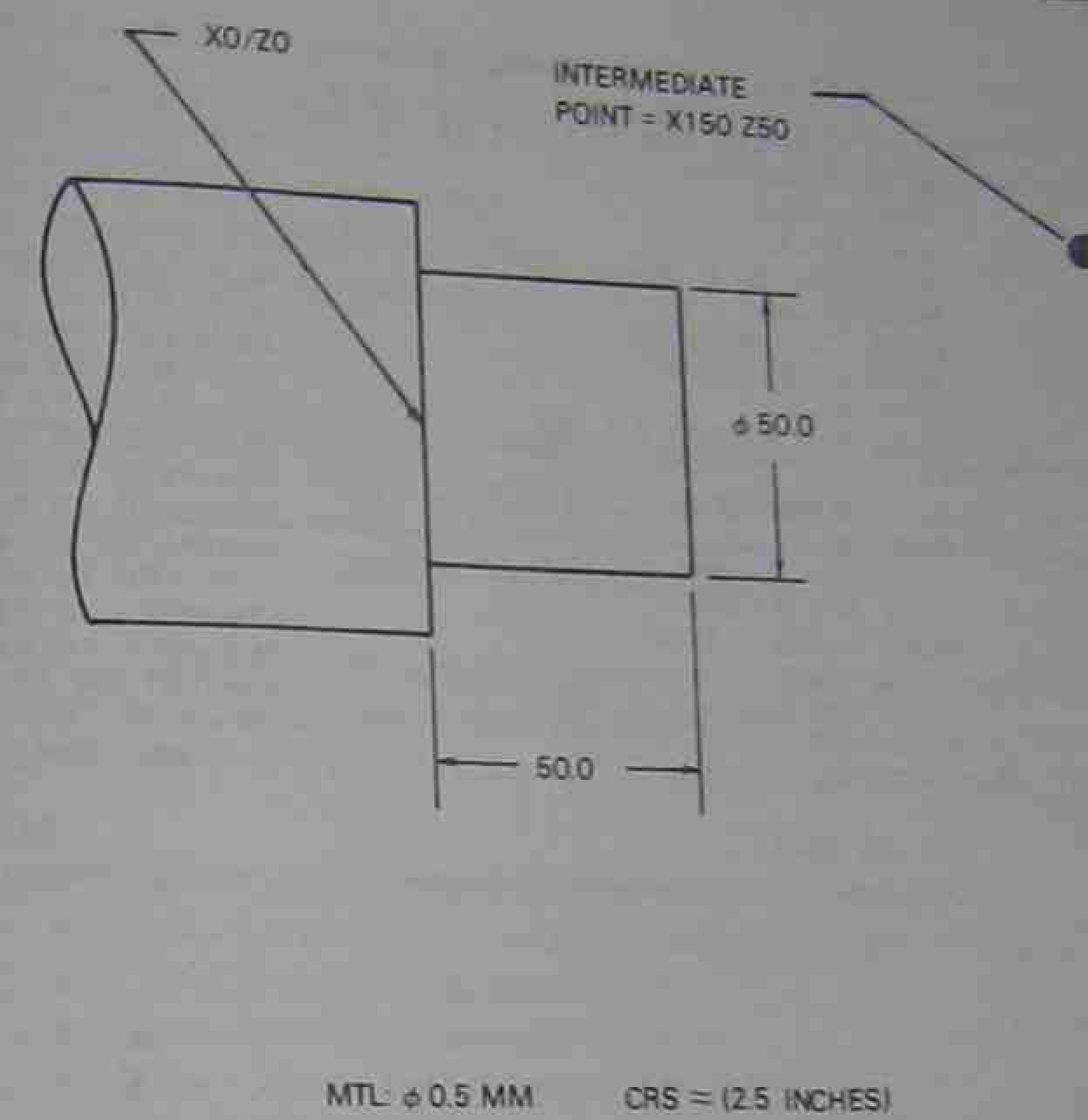


FIGURE 14–2
Part from Figure 14–1 with metric dimensions

G00—As in milling programs, G00 puts the machine in rapid traverse mode.

G01—Linear interpolation. As with milling, the machine will position the tool to the programmed coordinates at feedrate, in a straight line.

G28—Return to reference point. A G28 is programmed with an X and Z coordinate. Upon receiving the G28, the machine positions the tool at the fixed machine reference point, passing through the programmed X/Z location, called the intermediate point.

G70—Selects inch input.

G71—Selects metric input.

G90—Selects absolute positioning.

G91—Selects incremental positioning.

G94—Selects inches per minute or millimeters per minute feedrate. Feedrates are treated just as milling feedrates are.

```
XO = CENTERLINE OF SPNDL  ZO = PART SHOULDER

N010 G00 G40 G90 G95 G28 X6 Z6 M08   REM:SAFETY LINE, COOLNT ON
N020 T0101 M42                       REM:TURRET POS, HIGH RANGE
N030 S1200 M03                       REM:SPNDL ON
N040 X2.6 Z2.042                     REM:POSITION TO #1
N050 G01 X0 F.007                    REM:FEED TO #2
N060 Z2.032                          REM:FEED TO #3
N070 X2.314 F.003                    REM:FEED TO #4
N080 Z.042 F.007                     REM:FEED TO #5
N090 X2.6                            REM:FEED TO #6
N100 G00 X2.320 Z2.132               REM:RAPID TO #4
N110 G01 X2.0840                     REM:FEED TO #7
N120 Z.042 F.003                     REM:FEED TO #8
N130 X2.6                            REM:FEED TO #6
N140 G00 X2.084 Z2.132               REM:RAPID TO #7
N150 G01 X2.062                      REM:FEED TO #9
N160 Z.032 F.003                     REM:FEED TO #10
N170 X2.55                           REM:FEED TO #11
N180 G00 G28 X6 Z6 M09               REM:RETURN TO REF & COOLNT OFF
N190 M05                             REM:SPNDL OFF
N200 M30                             REM:END PRGM
```

FIGURE 14-3
Lathe facing and turning program for part in Figure 14–1, word address format, nonmetric

```
XO = CENTERLINE OF SPNDL  ZO = PART SHOULDER

N010 G00 G40 G90 G95 G28 X150 Z150 M08   REM:SAFETY LINE
N020 T0101 M42                           REM:TURRET POS, HIGH RANGE
N030 S1200 M03                           REM:SET SPEED
N040 X67 Z52                             REM:POSITION TO #1
N050 G01 X0 F.5                          REM:FEED TO #2
N060 Z51                                 REM:FEED TO #3
N070 X60 F.13                            REM:FEED TO #4
N080 Z2 F.5                              REM:FEED TO #5
N090 X67                                 REM:FEED TO # 6
N100 G00 X60 Z101                        REM:RAPID TO #4
N110 G01 X53                             REM FEED TO #7
N120 Z2 F.13                             REM:FEED TO #8
N130 X67                                 REM:FEED TO #6
N140 G00 X53 Z101                        REM:RAPID TO #7
N150 G01 X51                             REM:FEED TO #9
N160 Z1                                  REM:FEED TO #10
N170 X66                                 REM:FEED TO #11
N180 G00 G28 X150 Z150 M09               REM:RETURN TO REF & COOLNT OFF
N190 M05                                 REM:SPNDL OFF
N200 M30                                 REM:END PGRM
```

FIGURE 14-4
Lathe facing and turning program for part in Figure 14–2, word address format, metric

G95—Selects inches per revolution or millimeters per revolution feedrates. The feedrates are the programmed value per revolution of the spindle. A G95 F.01 advances the tool .010 inch for every revolution of the spindle.
M40—Selects the low gear range.
M41—Selects the middle gear range.
M42—Selects the high gear range.

PROGRAM EXPLANATION

(Refer to Figures 14–3 and 14–4.)

A .032-inch tool nose radius is used on the tool in the nonmetric program. A 1-mm tool nose radius is used in the metric program. One roughing and one finish facing cut are used; two roughing and one finish turning cuts are used.

N010

- G00—Selects the rapid traverse mode.
- G40—Cancels any active tool nose radius compensation.
- G90—Selects absolute positioning.
- G95—Selects per revolution feedrate.
- G28—Causes a return to reference point.
- X/Z coordinates—Intermediate point location. The intermediate point should be chosen so that tool movement will be free of the lathe chuck and part.
- M08—Turns on the coolant.

N020

- T0101—Selects a tool number and calls the tool offset in register #1.
- M42—Selects high gear range.

N030

- S1200—Sets the spindle speed to 1200 RPM.
- M03—Turns on the spindle.

N040

- X/Z coordinates—Rapid the tool to location #1, Figure 14–1. The X axis coordinate is diameter programmed, as are all the X coordinates in this program.

N050

- G01—Selects feedrate movement.
- X0—Feeds the tool to location #2. This is the rough facing cut.
- F.007—Sets the feedrate to .007 inch per spindle revolution (.5 mm metric.)

N060

- Z coordinate—Feeds the tool from location #2 to location #3. This sets the Z axis depth for the finish facing cut.

N070

- X coordinate—Feeds the tool from location #3 to location #4. The coordinate is diameter programmed.
- F.003 (F.13 metric)—Sets finish feedrate.

N080

- Z coordinate—Feeds the tool from location #4 to location #5. This is the first roughing pass.
- F.007 (F0.5 metric)—Sets the roughing pass feedrate.

N090

- X coordinate—To feed from location #5 to location #6. This cut rough faces the shoulder of the part and retracts the tool for the return move.

N100

- G00—Selects rapid traverse. This is a return to start of cut move. No feedrate is necessary.
- X/Z coordinates—Move the tool at rapid from location #6 to location #4.

N110

- G01—Selects linear interpolation (feedrate mode).
- X coordinate—Feeds the tool from location #4 to location #7. This move could also have been made in rapid traverse. Using a feedrate here eliminated the possibility of chipping the tool cutting edge on the corner of the stock.

N120

- Z coordinate—Feeds the tool from location #7 to location #8. This is the second rough turning pass.
- F.003 (F.13 metric)—Sets finish feedrate.

N130

- X coordinate—Rough faces the shoulder, retracting the tool.

N140

- G00—Selects rapid traverse.
- X/Z coordinate—Positions the tool to location #7.

N150

- G01—Selects feedrate movement.
- X coordinate—Feeds the tool from location #7 to location #9. This positions the X axis depth for the finish pass.

N160

- Z coordinate—Feeds the tool from location #9 to location #10. This completes the turning.

N170

- X coordinate—Feeds the tool from location #10 to location #11. This move finish faces the part shoulder.

N180

- G00—Selects rapid traverse.
- G28—Initiates a return to reference.
- X/Z coordinates—Intermediate point.
- M09—Turns the coolant off.

N190

- M05—Turns the spindle off.

N200

- M30—Signals the end of program.

TAPER TURNING

Linear interpolation on a lathe is used to turn tapers. It is similar in use to linear interpolation to cut angles when milling. On the part pictured in Figure 14–5 is a taper to be bored. The part is a steel casting, requiring that the taper be rough and then finish machined. (The short program to perform these operations is shown in Figure 14–7.)

Cutter offset calculations necessary with taper turning are similar to those used when calculating angle cuts for milling. Figure 14–6 depicts the relationship of the lathe tool nose to the tapered part surfaces. Two coordinate locations require cutter offsets. Both locations present the identical situation, so that calculating one offset will automatically yield the other. This is the same simple cutter-to-angle relationship first discussed in Chapters 8 and 9, and the formula given in Appendix 6, Figure 1, can be used. In this case, the Y axis in the formula is the X axis on the lathe, and the X axis in the formula is the Z axis on the lathe. The offset is calculated as follows, where CR is the tool nose radius:

$$X = \text{TAN}\left(\frac{\theta}{2}\right) \times \text{CR}$$

$$X = \text{TAN } 40 \times .032$$

$$X = .8391 \times .032$$

$$X = .02685 \text{ or } .027$$

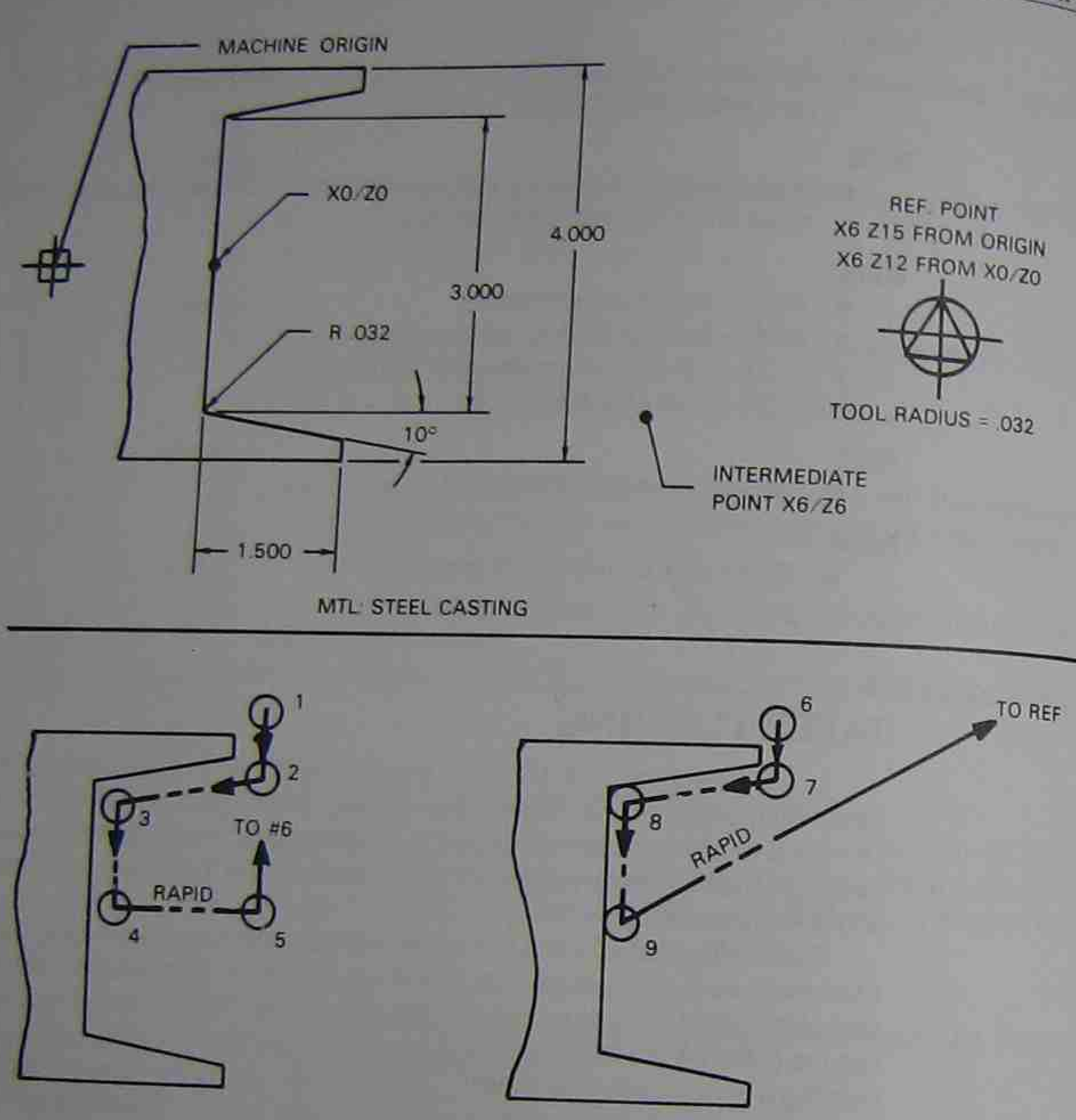


FIGURE 14–5
Taper turning

Before the cutter offset can be used, however, it is necessary to calculate the location of point B, Figure 14–6. By solving the indicated triangle for side b and adding that length to the known radius of the taper (1.5 inches), the radius dimension from the part center line to point B can be determined.

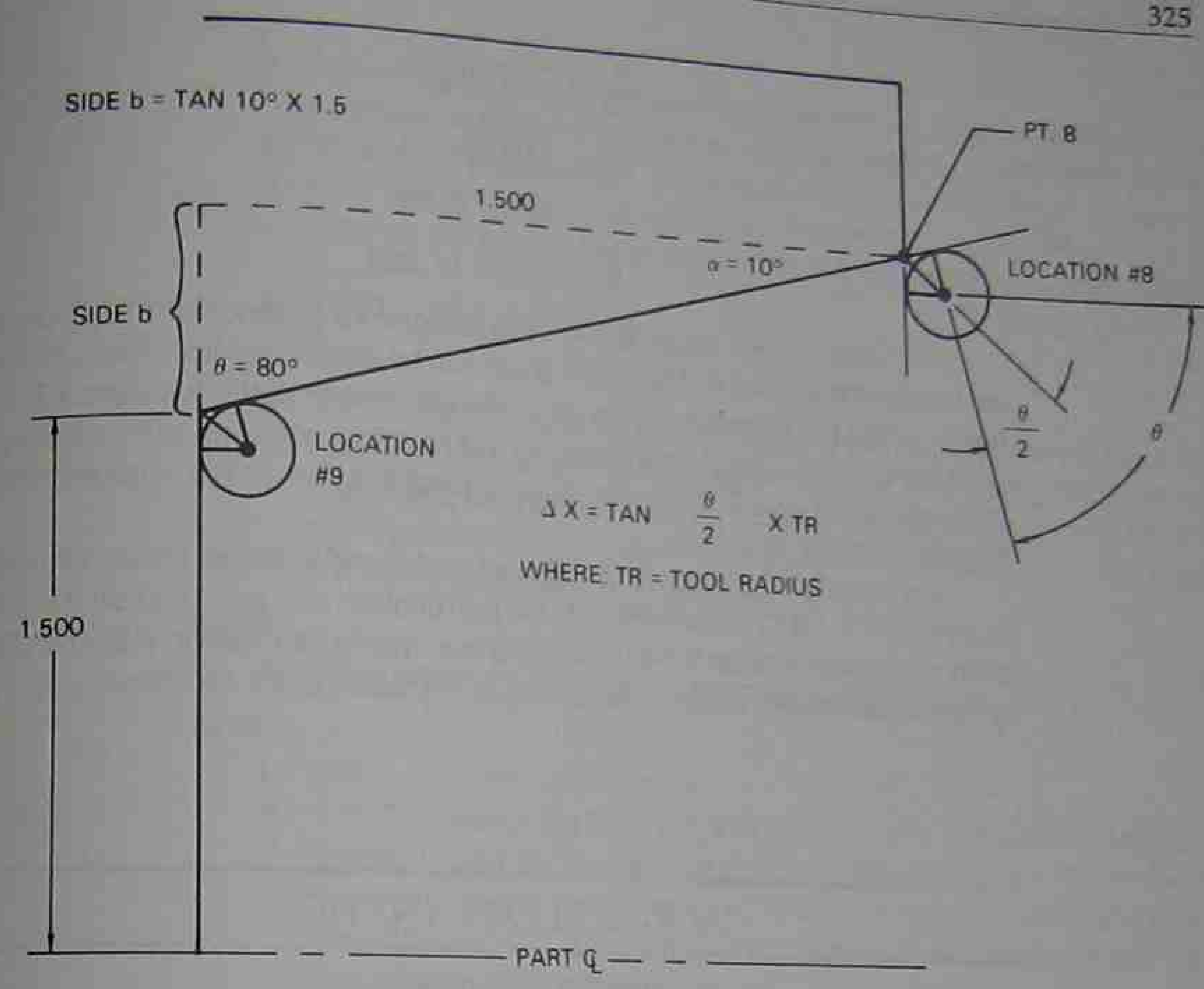


FIGURE 14–6
Determining cutter offsets

```
XO = CENTERLINE OF SPINDLE  ZO = PART SHOULDER

N010 G00 G40 G90 G95 G28 X6 Z6 M08  REM:SAFETY LINE, COOLNT ON
N020 T0101 M42                      REM:TURRET POS, HIGH RANGE
N030 S800 M03                       REM:SPNDL ON
N040 X4.1 Z1.51                     REM:POSITION TO #1
N050 G01 X3.454 F.007               REM:FEED TO #2
N060 X2.974 Z.042                   REM:FEED TO #3
N070 XO                             REM:FEED TO #4
N080 G00 Z1.542                     REM:RAPID TO #5
N090 X4.1 Z1.532                    REM:RAPID TO #6
N100 G01 X3.474 F.003               REM:FEED TO #7
N110 X2.946 Z.032                   REM:FEED TO #8
N120 XO                             REM:FEED TO #9
N130 G00 X6 Z6 M09                  REM:RETURN TO REF & COOLNT OFF
N140 M05                            REM:SPNDL OFF
N150 M30                            REM:END PRGM
```

FIGURE 14–7
Lathe taper turning program for part in Figure 14–5, word address format

$$\frac{b}{1.5} = \text{TAN } 10$$

$$b = \text{TAN } 10 \times 1.5$$

$$b = .1763 \times 1.5$$

$$b = .26445 \text{ or } .264$$

The value of .264 added to the 1.5 radius gives a distance of 1.764 from the part centerline to point B. The cutter offset can be subtracted from the 1.764 distance to find the dimension from the part centerline to cutter location #7. This distance is 1.737. The X coordinate for this location, however, will be diameter programmed. The 1.737 must now be doubled to arrive at the X coordinate to be programmed, or 3.474.

The calculated tool offset can also be subtracted from the 1.5 known radius to arrive at the 1.473 dimension from the part centerline to tool location #8. Doubling this distance gives 2.946, the X axis coordinate for location #8. The offset for the Z axis in both these cases is simply the radius of the tool nose.

PROGRAM EXPLANATION

(Refer to Figure 14–7.)

N010

- G00—Selects rapid traverse.
- G40—Cancels any active TNR comp.
- G70—Specifies inch input.
- G95—Specifies inches per revolution feedrate.
- G90—Selects absolute positioning.
- G28—Initiates a return to reference point.
- X6 Z6—Intermediate point coordinates for the reference point return.
- M08—Turns the coolant on.

N020

- T0101—Select the tool and the offset.
- M42—Selects high gear range.

N030

- S800—Sets the spindle speed to 800 RPM.
- M03—Turns the spindle on.

N040

- X4.1 Z1.51—Position the tool to location #1, Figure 14–5.

N050

- G01—Selects linear interpolation. The tool will feed in a straight line between the next coordinate programmed and the current tool location.
- X3.454—Feeds the tool from location #1 to location #2. This coordinate was determined by adding approximately the desired amount of finished stock to the cutter coordinate of location #8, calculated previously.
- F.007—Sets the feedrate.

N060

- X2.974 Z.042—Coordinates to feed the tool from location #2 to location #3. The X coordinate was determined by subtracting .020 from the calculated finished location coordinate. Although this coordinate will not leave exactly .010 inch of stock per side to be removed during finishing, the amount left will be close to that.

N070

- X0—Feeds the tool from location #3 to location #4.

N080

- G00—Selects rapid traverse.
- Z1.542—Sends the tool at rapid to location #5. This is an intermediate location used before sending the tool to location #6. If the tool were moved from location #4 to location #6 directly, the corner of the part would be cut off. Laying a straightedge between location #4 and location #6 will demonstrate the point.

N090

- X4.1 Z1.532—Feeds the tool from location #5 to location #6 at rapid (G00 is active).

N100

- G01—Selects linear interpolation.
- X3.474—Feeds the tool from location #6 to location #7. This is the coordinate location calculated earlier.
- F.003—Sets the finish pass feedrate to .003 inch per revolution.

N110

- X2.946 Z.032—Coordinates of location #8.

N120

- X0—Feeds the tool from location #8 to location #9.

N130

- G00—Specifies rapid traverse.
- G28—Initiates a return to reference.
- X6 Z6—Intermediate point coordinates for the reference return.
- M09—Turns the coolant off.

N140

- M05—Turns the spindle off.

N150

- M30—Ends the program.

CIRCULAR INTERPOLATION

Circular interpolation on a lathe does not differ significantly from circular interpolation when milling. There are two ways that an arc center can be programmed using CNC turning machines. The centerpoint can be programmed using I and K, or the center may be specified on some machinery as a radius value. Some machining centers may have an arc centerpoint specified by the radius method also.

When I and K are used, I is programmed as the X-axis coordinate of the arc centerpoint, and K is programmed as the Z-axis coordinate. The format is:

N... G02/G03 X.... Z.... I.... K....

Where G02 is clockwise circular interpolation, and G03 is counterclockwise circular interpolation; X is the X axis endpoint of the arc; Z is the Z axis endpoint of the arc; I is the X axis coordinate of the arc centerpoint; and K is the Z axis coordinate of the arc centerpoint.

When the center is specified using a radius, the R address is used. R is programmed as an incremental value from the current tool position. The format is:

N... G02/G03 X.... Z.... R....

Two programs are presented here for turning a spherical end on a 2.000-inch-diameter piece of 304 stainless steel (see Figure 14–8). Figure 14–9 is a program to turn the end using I and K; Figure 14–11 is identical except that R is used instead.

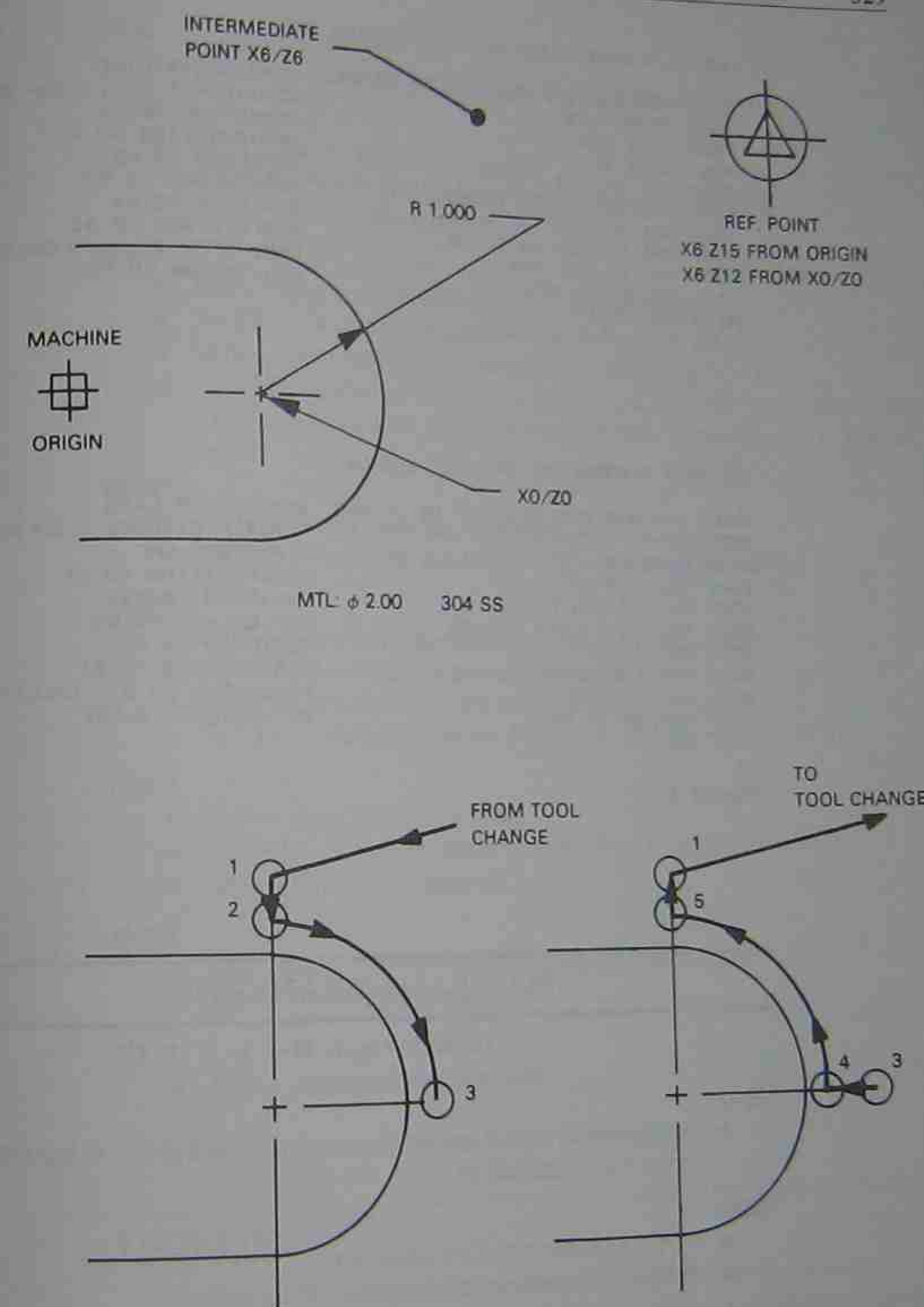


FIGURE 14–8
Turning a spherical end

```
X0/Z0 = CENTERLINE OF PART RADIUS

N010 G00 G40 G70 G90 G28 X6 Z6 M08   REM:SAFETY LINE
N020 T0101 M42                       REM:TURRET POS, HIGH RANGE
N030 S150 M03                        REM:SPNDL ON
N040 X2.1 Z0                         REM:POSITION TO #1
N050 G01 X2.084 F.003                REM:FEED TO #2
N060 G02 X0 Z1.042 I0 K0             REM:CW ARC TO #3
N070 G01 Z1.032                      REM:FEED TO #4
N080 G03 X2.062 Z0 I0 K0             REM:CCW ARC TO #1
N090 G00 X2.084 M09                  REM:RAPID TO #1, COOLNT OFF
N100 G28 X6 Z6 M05                   REM:RETURN TO REF
N110 M30
```

```
X0/Z0 = CENTERLINE OF PART RADIUS

N010 G00 G40 G70 G90 G28 X6 Z6 M08   REM:SAFETY LINE
N020 T0101 M42                       REM:TURRET POS, HIGH RANGE
N030 S150 M03                        REM:SPNDL ON
N040 X2.1 Z0 M03                     REM:POSITION TO #1
N050 G01 X2.084 F.003                REM:FEED TO #2
N060 G02 X0 Z1.042 R1.042            REM:CW ARC TO #3
N070 G01 Z1.032                      REM:FEED TO #4
N080 G03 X2.062 Z0 R1.032            REM:CCW ARC TO #1
N090 G00 X2.084 M09                  REM:RAPID TO #1, COOLNT OFF
N100 G28 X6 Z6 M05                   REM:RETURN TO REF
N110 M30
```

FIGURE 14–9

PROGRAM EXPLANATION

(Refer to Figure 14–9.)

N010

- Safety line to cancel any active codes, returns the tool to the reference point. Turns coolant on.

N020

- T0101—Selects tool #1, offset #1.
- M42—Selects high gear range.

N030

- S150—Sets the spindle speed to 150 RPM.
- M03—Turns the spindle on.

N040

- X2.1 Z0—Positions the tool to location #1, Figure 14–8.

N050

- G01—Selects feedrate movement.
- X 2.084—Feeds the tool from location #1 to location #2.
- F.003—Assigns the feedrate.

N060

- G02—Selects clockwise circular interpolation.
- X0 Z1.042—Arc endpoint coordinates, location #3.
- I0 K0—Centerpoints of the arc, Figure 14–9, top.
- R1.042—Radius value, Figure 14–9, bottom. The 1.042 value is the incremental distance from the arc start point (location #2) to the arc center.

N070

- G01—Selects feedrate movement.
- Z1.032—Feeds the tool from location #3 to location #4.

N080

- G03—Selects counterclockwise circular interpolation.
- X2.062 Z0—Endpoint coordinates of the arc.
- I0 K0—Centerpoints of the arc, Figure 14–9, top.
- R 1.032—Radius of the arc, Figure 14–9, bottom.

N090

- G00—Selects rapid traverse.
- X2.084—Rapids the cutter from location #5 to location #1.
- M09—Turns the coolant off.

N100

- G28 X6 Z6—Returns the tool to the reference point.
- M05—Turns the spindle off.

N110

- M30—Signals end of program.

DRILLING

Drilling on NC lathes is accomplished in similar manner to turning and boring. The tool is sent to a desired start position, and the coordinates are given to move along the proper path. When drilling, the tool point is programmed since there is no tool radius involved. Canned cycles like those used for drilling on NC mills will be discussed in a later section.

To drill a ⅜ diameter hole 1.500 inches deep in part Figure 14–1, a center-drill and a ⅜ drill can be added to the program in Figure 14–3. This has been done in Figure 14–10. The program explanation follows.

```
XO = CENTERLINE OF SPINDLE  ZO = PART SHOULDER

N010 G00 G40 G90 G95 G28 X6 Z6 M08
N020 T0101 M42
N030 S1200 M03
N040 X2.6 Z2.042
N050 G01 X0 F.007
N060 Z2.032
N070 X2.314 F.003
N080 Z.042 F.007
N090 X2.6
N100 G00 X2.320 Z2.132
N110 G01 X2.0840
N120 Z.042 F.003
N130 X2.6
N140 G00 X2.084 Z2.132
N150 G01 X2.062
N160 Z.032 F.003
N170 X2.55
N180 G00 G28 X6 Z6 M09
N190 M01                         REM:OPSTOP

N200 M08                         REM:COOLNT ON
N210 T0202 M42                   REM:TURRET POS & HIGH RANGE
N220 S1800 M03                   REM:SPNDL ON, 1800 RPM
N230 G00 X0 Z2.100               REM:POSITION TO START
N240 G01 Z-1.850 F.003           REM:FEED TO DEPTH
N250 G00 Z2.100                  REM:RAPID TO START POS.
N260 G28 X6 Z6 M09               REM:RETURN TO REF, COOLNT OFF
N270 M01                         REM:OPSTOP

N280 M08                         REM:COOLNT ON
N290 T0303 M42                   REM:TURRET POS & HIGH RANGE
N300 S1600 M03                   REM:SPNDL ON, 1600 RPM
N310 G00 X0 Z2.100               REM:RAPID TO START POS.
N320 G01 Z1.625 F.003            REM:FEED TO 1ST PECKING DEPTH
N330 G00 Z2.500                  REM:RAPID OUT OF PART
N340 Z1.630                      REM:RAPID TO START OF PECK
N350 G01 Z1.375                  REM:FEED TO 2ND PECKING DEPTH
N360 G00 Z2.500                  REM:RAPID OUT OF PART
N370 Z1.380                      REM:RAPID TO START OF PECK
N380 G01 Z1.000                  REM:FEED TO 3RD PECKING DEPTH
N390 G00 Z2.500                  REM:RAPID OUT OF PART
N400 Z1.005                      REM:RAPID TO START OF PECK
N410 G01 Z.625                   REM:FEED TO 4TH PECKING DEPTH
N420 G00 Z2.500                  REM:RAPID OUT OF PART
N430 Z.630                       REM:RAPID TO START OF PECK
N440 G01 Z.387                   REM:FEED TO FINISH DEPTH
N450 G00 Z.100                   REM:RAPID TO START POSITION
N460 G28 Z6 Z6 M09               REM:RETURN TO REF, COOLNT OFF
N470 M05                         REM:SPNDL OFF
N480 M30                         REM:END PRGM
```

FIGURE 14–10

PROGRAM EXPLANATION

N010–N180 are identical to Figure 14–3.

N190

- Optional stop code. This code aids the operator during setup. If the optional stop switch is turned on at the console, the program will stop at this line. The operator can then inspect the workpiece during setup. It is common practice to include an M01 at the end of each tool.

N200–N220

- Selects the tool, offset, gear range. Turns the spindle and coolant on.

N230

- G00—Rapid traverse mode.
- X0 Z2.100—Rapids the centerdrill to the start position, .100 away from the workpiece face.

N240

- G01—Feedrate mode.
- Z–1.850—Depth of centerdrilling (.150 deep).
- F.003—Sets feedrate at .003 IPR.

N250

- G00—Rapid traverse mode.
- Z2.100—Returns tool to the start position.

N260

- Returns tool to the reference point and cancels the tool offset.

N270

- M01—Optional stop code.

N280–N300

- Selects tool, offset, gear range. Turns on spindle and coolant.

N310

- Rapids tool tip to the start point.

N320

- G01—Feedrate mode.
- Z1.625—Depth of first drill peck.
- F.003—Sets the feedrate to .003 IPR.

N330

- G00—Rapid traverse mode.
- Z2.500—Sends the tool tip .500 away from the part face. The .500 distance gives the coolant sufficient area to enter the section of hole just drilled to lubricate the drill point on the next drill peck.

N340

- Z1.650—Sends the tool tip to the start of the next peck, .005 from the end point of the previous drill peck.

N350

- G01—Feedrate mode.
- Z1.375—End point to the second drill peck.

N360

- Rapids tool .500 out of part.

N370

- Rapids tool tip to start of third peck.

N380–N440

- The pecking cycle is repeated until final hole depth is achieved.

N450

- Tool rapids out of part to original start position.

N460

- Returns to reference line.

N470

- Spindle off.

N480

- END of program.

THREADING

When threading on CNC lathes, one of three threading cycles is used: single pass threading (G33), multiple pass threading (G92), or multiple pass threading (G76). When a G33 is issued, the tool travels the length of the thread and stops. The tool then has to be retracted from the thread, returned to the starting point, and the whole procedure repeated. When a G92 command is issued, the tool moves to a programmed X coordinate, feeds across the length of the thread to the programmed Z coordinate, and returns to the start point. This process is automatically repeated with the X-axis moving to a new programmed

X coordinate until the final X coordinate has been executed. When a G76 is issued, the machine makes a threading pass, then automatically retracts the tool to the X axis reference position and returns it to the Z axis start position. It then automatically repeats the procedure until the final depth of the thread is achieved.

Three types of threads can be cut using a CNC lathe: constant lead, increasing lead, and decreasing lead. The lead of a thread is the distance that the thread advances in one revolution. Some CNC lathes are capable of cutting only constant lead threads, depending on the thread-cutting options selected when the machine is purchased. Threads of increasing and decreasing lead are specialized applications and will not be dealt with in this text.

When cutting threads, the relationship between spindle speed and tool feedrate is very important. When a G code is used for thread cutting, the feedrate override controls on the MCU console, which allow the operator to adjust the feedrate during machining, will not function. When beginning a threading pass, a certain distance (A in Figure 14–11) must be allowed ahead of the part face, to give the lathe carriage time to accelerate to the proper feedrate. Failure to allow this distance will result in improper leads on the first several threads.

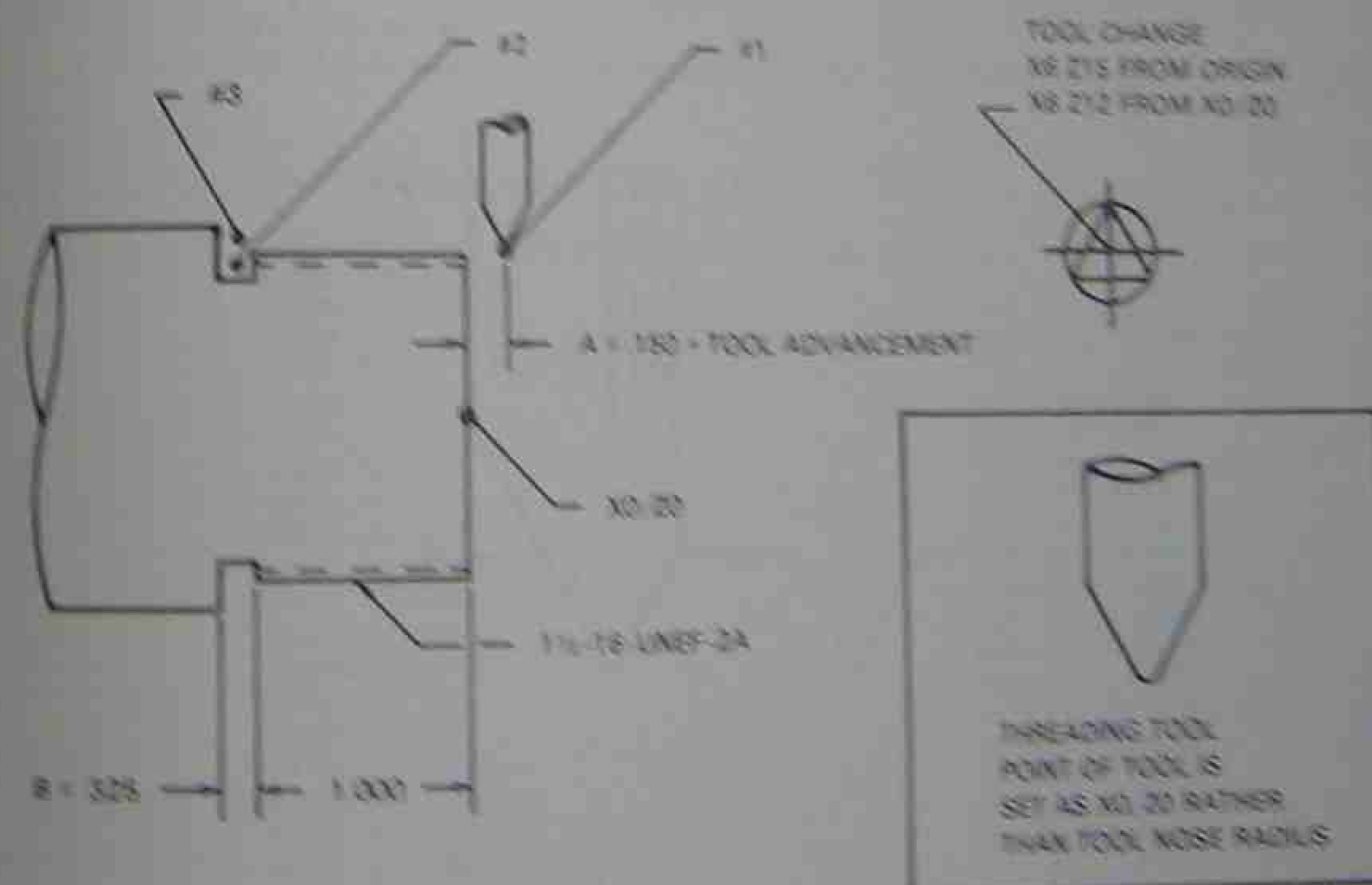


FIGURE 14–11
Part to be threaded

Starting distance A varies from machine to machine. Charts giving the distance for a particular thread on a particular machine will be found in the programming manual. If a chart is not available, the following formula can be used:

$$A = (RPM \times LEAD \times .006) + Z$$

Where Z is the amount of tool advancement in the Z axis. *Tool advancement* occurs, prior to the start of a threading cut, along two axes, as illustrated in Figure 14–12. Advancement along the Z axis is calculated by the formula:

$$Z = X\,(TAN\ 30)$$

Some programmers prefer to feed the tool in at a 29-degree angle instead of 30. In this case the formula would be:

$$Z = X\,(TAN\ 29)$$

The stopping distance is similar to the starting distance. This distance is shown in Figure 14–11 as dimension B. The minimum stopping distance can be calculated by the following formula if a chart is not available:

$$B = RPM \times LEAD \times .013$$

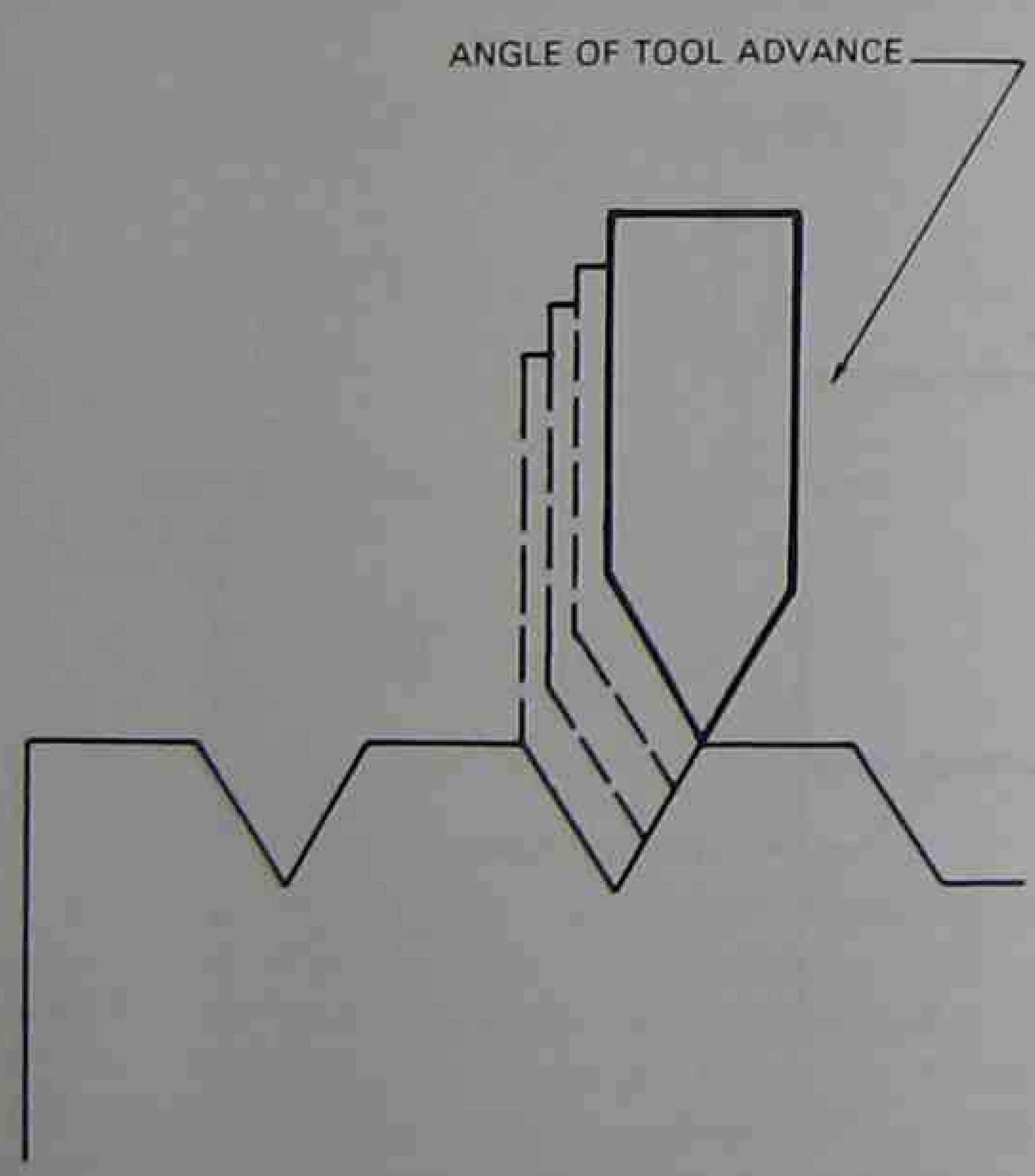


FIGURE 14–12
Tool advancement

Three threading programs have been written for the part shown in Figure 14–10. The program in Figure 14–13 cuts the thread using single pass threading. The program in Figures 14–14 and 14–15 cut the thread using multiple pass threading. The format for single pass threading is:

n... G33... Z.... F....

```
XO = SPINDLE CENTERLINE ZO = PART FACE

N010 G00 G40 G70 G90 G95 G28 X6 Z6  REM:SAFTEY LINE, REF. RETURN
N020 M06 T0101
N030 S400 M03
N040 X1.47 Z.015                    REM:POSITION TO #1
N050 G91 G33 Z-1.15 F.0625          REM:1ST THREAD PASS
N060 G00 X.015                      REM:RETRACT X
N070 Z1.168                         REM:RETURN TO START
N080 X-.032 Z-.018                  REM:ADVANCE TOOL
N090 G33 Z-1.168 F.0625             REM:2ND THREAD PASS
N100 G00 X.032                      REM:RETRACT X
N110 Z1.186                         REM:RETURN TO START
N120 X.032 Z-.018                   REM:ADVANCE TOOL
N130 G33 Z-1.186 F.0625             REM:FINISH PASS
N140 G00 X.032 M09                  REM:RETRACT X
N150 G90 G28 X6 Z6 M05              REM:RETURN TO REF.
N160 M30
```

FIGURE 14–13
Single pass threading program for part in Figure 14–11, word address format

```
XO = SPINDLE CENTERLINE  ZO = PART FACE

N010 G00 G40 G70 G90 G95 G28 X6 Z6 M08
N020 T0101 M42
N030 S700 M03
N040 G00 X1.600 Z.150               REM:THD. START POINT
N050 G92 X1.580 Z-1.150             REM:1ST PASS
N060 X1.570                         REM:2ND PASS
N070 X1.550                         REM:3RD PASS
N080 X1.530                         REM:4TH PASS
N090 X1.510                         REM:5TH PASS
N100 X1.490                         REM:6TH PASS
N110 X1.470                         REM:7TH PASS
N120 X1.460                         REM:8TH PASS
N130 X1.455                         REM:9TH PASS
N140 X1.450                         REM:10TH PASS
N150 X1.445                         REM:11TH PASS
N160 X1.443                         REM:12TH PASS
N170 X1.440                         REM:13TH PASS
N180 X1.438                         REM:14TH PASS
N190 X1.437                         REM:15TH PASS
N200 X1.436                         REM:16TH PASS
N210 G28 X6 Z6 M09
N220 M05
N230 M30
```

FIGURE 14–14

```
XO = SPINDLE CENTERLINE ZO = PART FACE

N010 G00 G40 G70 G90 G95 G28 X6 Z6          REM:SAFTEY LINE,
REF. RETURN
N020 M06 T0101
N030 S400 M03
N040 Z.15
N050 G76 X1.436 Z1 I0 K.032 F.0625 D.015 A60 REM:THREADING PASS
N060 G00 G28 X6 Z6 M09                       REM:RETURN TO REF.
N070 M05
N080 M30
```

FIGURE 14-15
Multipass threading program for part in Figure 14-11, word address format

Where G33 is the thread-cutting G code, Z is the length of the threading cut, and F is the lead of the thread. (Some lathe controllers use K to specify the lead of the thread.)

The format for G92 multipass threading is:

N... G92 X.... Z.... F....
N... X...
N... X...
.
.
.
N... X...

Where: G92 = multipass threading code
X = X coordinate of the first threading pass
Z = Z coordinate of the threading end point
F = the feedrate (lead) of the thread

X = depth of second pass
X = depth of third pass

etc. until

X = depth of final pass

Usually the lead can be given to only four decimal places, so that some round-off error will occur. This is usually so slight that it will affect only threads several feet long. Some machines have the capacity to accept thread leads to five or six decimal places.

The format for G76 multiple pass threading is:

N... G76 X.... Z.... I.... K.... D.... F.... A..

Where: G76 = multipass threading G code
X = minor diameter of the thread
Z = length of thread
I = difference in thread radius from one end of the thread to the other. This value is used for cutting tapered threads. For straight threads, a value of zero is entered.
K = height of the thread (a radius value, given from the crest of the thread to the root)
D = depth of cut for the first pass
F = lead of the thread
A = angle of the tool tip. (For Unified, American National, and IFI metric threads, the angle is 60 degrees.)

The main differences between G92 and G76 are first, G76 requires only one line of program code, and second, G92 plunges the tool straight into the workpiece rather than feeding in at an angle. The infeed direction with G76 is 30 degrees. The infeed direction with G33 is controlled by the programmer.

PROGRAM EXPLANATION

(Refer to Figure 14-13.)

N010
- Safety line, returns tool to reference.

N020
- M06 T0101—Selects tool #1, offset #1.

N030
- S400—Sets the spindle speed to 400 RPM.
- M03—Turns on the spindle.

N040
- X1.47 Z.15—Coordinates of location #1, Figure 14-11. The X coordinate is diameter programmed and positions the tool to the depth of the first pass. The Z coordinate is the starting distance. Subsequent passes will add to the starting distance the amount of Z-axis tool advancement.

N050
- G91—Selects incremental positioning.
- G33—Initiates single pass threading.
- Z1.15—Feeds the tool from location #1 to location #2, Figure 14-11.
- F.0625—Lead of the thread.

N060
- G00—Selects rapid traverse.
- X.015—Incremental coordinate to rapid the tool from location #2 to location #3.

N070
- Z−1.168—Incremental distance to rapid the tool back to the starting point. This coordinate also compensates for the additional starting distance required by the tool advancement for the next pass.

N080
- X−.032—Incremental coordinate to advance the tool for the next cut. Two .015-inch roughing cuts are being made. This coordinate advances the X axis the .015 inch the tool was retracted at the end of the first pass, plus the .015 inch desired for the second.
- Z−.018—Calculated Z-axis tool advancement to cause the tool to advance on a 30-degree angle.

N090
- G33—Initiates the threading cycle.
- Z−1.168—Feeds the tool from the start point (location #1) to the end of the thread point (location #2).
- F.0625—Lead of the thread.

N100
- G00—Selects rapid traverse.
- X.032—Retracts the X axis from the thread.

N110
- Z1.168—Returns the tool to the starting point of the thread.

N120
- X−.32 Z−.018—Advances the tool to final thread depth.

N130
- G33—Initiates thread cutting.
- Z−1.168—Feeds the tool from #1 to #2.
- F.0625—Lead of the thread.

N140
- G00—Selects rapid traverse.
- X.032—Retracts the tool from the thread.
- M09—Turns off the coolant.

N150
- G90—Selects absolute positioning.
- G28—Returns the tool to the reference point.
- X6 Z6—Intermediate point coordinates.
- M05—Turns off the spindle.

N160
- M30—Signals end of program.

PROGRAM EXPLANATION

(Refer to Figure 14–14.)

N010
- Safety line, returns to reference.

N020
- T0101—Selects tool and offset.
- M42—Selects high gear range.

N030
- S700 M03—Turns the spindle on at 700 RPM.

N040
- X1.600 Z.150—Start position of the thread.

N050
- G92—Initiates threading cycle.
- X1.580—X coordinate of first threading pass.
- Z−1.150—Z coordinate of the ending point.
- F.0625—The thread lead.

N060–N200
- X coordinates of the succeeding thread passes. N200 is the last pass. Note the passes gradually remove less and less stock per pass to eliminate tearing of the thread.

N210–N220
- Returns the tool to reference. Turns off coolant and spindle.

N230
- END of program.

PROGRAM EXPLANATION

(Refer to Figure 14–15.)

N010
- Safety line.

N020
- M06 T0101—Selects tool #1, offset #1.

N030
- S400—Sets the spindle speed.
- M03—Turns the spindle on.

N040
- Z1.5—Positions the Z axis at the start of the thread.

N050
- G76—Initiates multipass threading.
- X1.436—Minor diameter of the thread.
- Z1—Length of the thread.
- I0—Difference in radius of the thread from the starting point to the finish point.
- K.032—Height of the thread measured from the crest to the root.
- D.015—Specifies a .015-inch first pass.
- F.0625—Lead of the thread.
- A60—Specifies a 60-degree thread.

N060
- G00—Selects rapid traverse.
- G28—Initiates a return to reference.
- X6 Z6—Intermediate point coordinates.
- M09—Turns off the coolant.

N070
- M05—Turns off the spindle.

N080
- M30—Signals the end of program.

Note how the amount of programming is reduced when using the multipass cycle.

Do loops and subroutines may also be used in lathe programming; they are programmed in just as when milling. Tool nose radius compensation may also be used. TNR comp has not been discussed here in order to concentrate on the basics of tool nose centerline programming. It is used in similar fashion to cutter

diameter compensation in CNC milling programs. Once tool nose centerline programming is understood, there should be no problem in using TNR comp. The same ramp on/ramp off precautions apply in turning as in milling.

A COMPLETE LATHE EXAMPLE

Up to this point, small lathe programming routines have been presented. These routines illustrate various lathe operations which usually are parts of a single lathe program. Figure 14–16 is a part for which a program has been written. The program is contained in Figure 14–17. A brief program explanation follows. There are several codes used in this program that should be noted.

G98—used to select inch per revolution feedrates.
G97—used to select direct RPM programming.
M24—used when threading to cause the tool to pull straight out of the part. The default condition for a thread cycle is for the tool to pull out at a 60 to 45 degree angle.

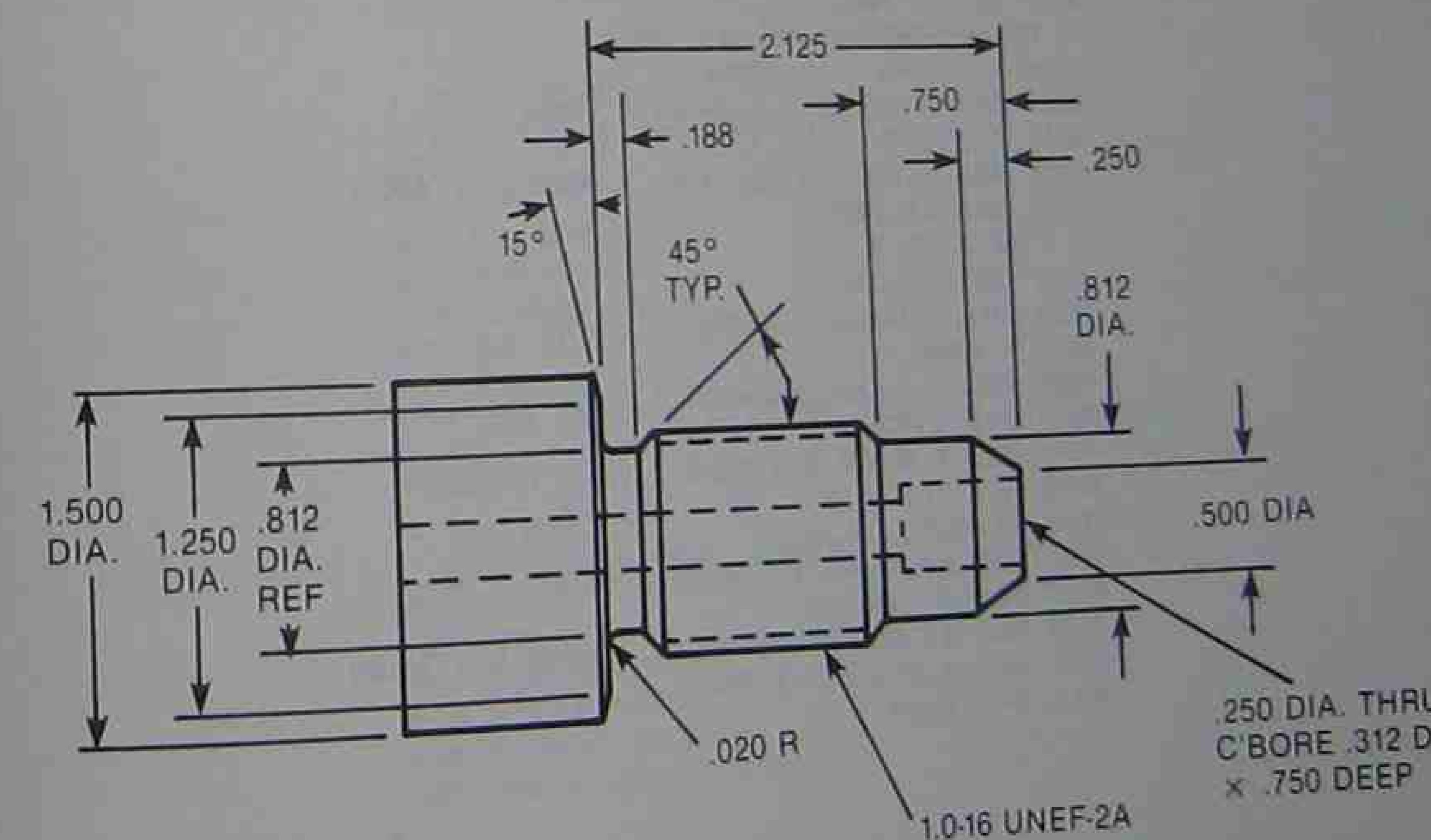


FIGURE 14–16
Part drawing

```
LATHE PROGRAMMING EXAMPLE
XO = CENTERLINE OF PART
ZO = FACE OF PART

( .031R X 80 DEG. TURNING TOOL)
N1 G97
N2 G99
N3 M08
N4 G00 T0101
N5 S2133 M03
( ROUGH FACE PART - LEAVE .005 STK.)
N6 X1.0000 Z.0310
N7 G01 X.0000 F.0070
N8 G00 Z.1000
( ROUGH TURN 1.0 DIA. IN 2 PASSES - LEAVE .005 STK./SIDE)
N9 X1.1720
N10 G01 Z-2.0890 F.0070
N11 X1.6720
N12 G00 Z.1000
N13 X1.0720
N14 G01 Z-2.0890 F.0070
N15 X1.2594
N16 X1.6720 Z-2.1443
N17 G00 Z.1360
N18 G28 X6.0000 Z6.0000
N19 M01

( .007R X 35 DEG. TURNING TOOL)
N20 G99
N21 M08
N22 G00 T0202
N23 S2133 M03
( ROUGH THREAD RELIEF AREA)
N24 X1.0740 Z-1.8230
N25 G01 X.8360 Z-1.9420 F.0030
N26 Z-2.1050
N27 G02 X.8520 Z-2.1130 I.8520 K-2.1050
N28 G01 X1.0840
( FINISH O.D.)
N29 G00 Z.0070
N30 G01 X.0000 F.0030
N31 X.4890
N32 G03 X.5178 Z-.0010 I.4889 K-.0100
N33 G01 X.8208 Z-.2439
N34 G03 X.8260 Z-.2529 I.7920 K-.2529
N35 G01 Z-.7471
N36 X1.0040 Z-.8361
N37 G03 X1.0140 Z-.8481 I.9800 K-.8481
N38 G01 Z-1.8389
N39 G03 X1.0040 Z-1.8509 I.9800 K-1.8389
N40 G01 X.8260 Z-1.9399
N41 Z-2.1050
```



```
N42 G02 X.8520 Z-2.1180 I.8520 K-2.1050
N43 G01 X1.2518
N44 X1.6140 Z-2.1665
N45 G00 Z.1070
N46 G28 X6.0000 Z6.0000
N47 M01

( THREADING TOOL)
( THREAD O.D. 1-16-2A)
N48 G99
N49 M08
N50 G00 T0303
N51 S900 M03
N52 X-.5000 Z.6000 M74
N53 G92 X.9900 Z-2.1000 F.0625
N54 X.9800
N55 X.9718
N56 X.9654
N57 X.9600
N58 X.9552
N59 X.9510
N60 X.9470
N61 X.9434
N62 X.9400
N63 X.9368
N64 X.9336
N65 X.9308
N66 X.9278
N67 X.9252
N68 X.9234
N69 X.9234 Z-2.1000
N70 G28 X6.0000 Z6.0000
N71 M01

( NO. 4 C'DRILL)
( C'DRILL TO .260 DIA.)
N72 G99
N73 M08
N74 G00 T0404
N75 S3000 M03
N76 X.0000 Z.1000
N77 G01 Z-.2780 F.0030
N78 G00 Z.1000
N79 G28 X6.0000 Z6.0000
N80 M01

( 1/4 DRILL)
( DRILL .250 DIA. THRU)
N81 G99
N82 M08
N83 G00 T0505
N84 S2000 M03
N85 X.0000 Z.1000
N86 G01 Z-.3000 F.0030
```

```
N87 G00 Z.5000
N88 Z-.2950
N89 G01 Z-.6000 F.0030
N90 G00 Z.5000
N91-.5950 $
N92 G01 Z-.9000 F.0030
N93 G00 Z.5000
N94 Z-.8950
N95 G01 Z-1.2000 F.0030
N96 G00 Z.5000
N97 Z-1.1950
N98 G01 Z-1.5000 F.0030
N99 G00 Z.5000
N100 Z-1.4950
N101 G01 Z-1.8000 F.0030
N102 G00 Z.5000
N103 Z-1.7950
N104 G01 Z-2.1000 F.0030
N105 G00 Z.5000
N106 Z-2.0950
N107 G01 Z-2.4000 F.0030
N108 G00 Z.5000
N109 Z-2.3950
N110 G01 Z-2.7000 F.0030
N111 G00 Z.5000
N112 Z-2.6950
N113 G01 Z-3.0000 F.0030
N114 G00 Z.5000
N115 Z-2.9950
N116 G01 Z-3.2500 F.0030
N117 G00 Z.1000
N118 G28 X6.0000 Z6.0000
N119 M01

( .005R BORING BAR)
N120 G99
N121 M08
N122 G00 T0606
N123 S3500 M03
( ROUGH C'BORE - LEAVE .005 STK/SIDE)
N124 X.2920 Z.0350
N125 G01 Z-.7400 F.0020
N126 X.1520
N127 G00 Z.0400
( FINISH C'BORE - DEBURR EDGE WITH .01R)
N128 X.3320
N129 G01 Z.0105 F.0020
N130 G02 X.2920 Z-.0095 I.3320 K-.0095
N131 G01 Z-.7400
N132 X.1320
N133 G00 Z.1100
N134 G28 X6.0000 Z6.0000 M09
N135 M05
N136 M30
```

FIGURE 14-17
Program for the part in Figure 14-16

PROGRAM EXPLANATION

First Tool:

N1–N5
- Selects first tool. Turns on spindle and coolant.

N6–N8
- Part is rough faced with .005 stock left for finishing.

N9–N11
- First roughing pass on o.d.

N12–N17
- Second roughing pass on o.d. The 15 degree angle is also rough turned at this time.

N18
- Tool is returned to reference point. Tool offset cancelled.

Second Tool:

N20–N23
- Selects second tool. Turns on spindle and coolant.

N24–N28
- Thread relief area is rough turned. .005 stock is left for finishing.

N29–N31
- Face of part is finished.

N32
- Deburring radius is turned at the intersection of the first angle and the face of the part.

N33
- First angle is finish turned.

N34
- Deburring radius is turned at the intersection of the first angle and the .812 diameter.

N35
- The .812 diameter is finish turned.

N36
- The front thread chamfer is finish turned.

N37
- A radius is turned at the intersection of the thread chamfer and major diameter.

N38
- The major diameter of the thread is turned.

N39
- A radius is turned at the intersection of the back thread chamfer and major diameter.

N40
- The back thread chamfer is turned.

N41
- The .812 diameter thread relief is turned.

N42
- The .020 radius is turned.

N43
- The 2.125 dimension is faced.

N44
- The 15 degree angle is finish turned.

N45–N47
- The tool is returned to reference. The offset is cancelled.

Third Tool:

N48–N51
- Tool and offset selected, spindle and coolant turned on.

N52
- The tool is sent to the start position for threading.
- The M74 turns off the thread chamfering at the end of a thread pass.

N53
- G92 multi-pass thread cycle initiated.

N54–N68
- Succeeding X values for the G92 cycle. Each X value is used on a separate thread pass.

N69
- Last threading pass which is a repeat pass. The Z coordinate is optional.

N70–N71
- Returns the tool to reference. Offset is cancelled.

Fourth Tool:

N72–N75
- Tool, offset, spindle speed selected.

N76–N78
- Drill sent to start point, fed to depth, and rapids back to start position.

N79–N80
- Returns to reference.

Fifth Tool:

N81–N84
- Tool, offset, spindle speed selected.

N85–N116
- Peck drilling of ¼ inch through hole. Each peck is .300 deep. At end of peck the tool is sent at rapid z.500 to clear out chips and allow coolant into the hole. The tool sequence repeats until final depth is achieved in N116.

N117
- Tool is returned to the starting position.

N118–N119
- Returns to reference.

Sixth Tool:

N120–N123
- Tool, offset, spindle speed selected.

N124–N127
- The c'bore is rough bored. .005 stock is left for finishing.

N128–N130
- A deburring radius is turned at the intersection of the c'bore and the part face.

N131–N133
- The c'bore is finish bored and tool retracted from part.

N134
- Return to reference line, coolant off.

N135
- Spindle off.

N136
- END of program.

CANNED CYCLES

Most modern CNC lathe controllers contain a number of built-in canned cycles. The threading cycles G33, G92, and G76 are standard from controller to controller. Other canned cycles are options offered by the controller manufacturer. These cycles are often unique to a given controller manufacturer (sometimes unique to a given model of controller) and therefore not transportable from controller to controller. With the current CNC lathe investment strategies by small and midsized companies, canned cycles will become as standardized as mill cycles at some future point. It is not possible to cover the number of cycle variations in a text of this size. The student should be aware, however, that these cycles exist. Documentation on the use of these cycles will be contained in the programming and operational manuals for a given machine.

How much a company relies on canned cycles for lathe programming is dependent on their use or non-use of computer-aided programming. Where computer-aided or graphics programming is utilized, there is little need for canned cycles aside from the standard lathe threading cycles. Where MDI programming is used, canned cycles can save many hours of programming time. The cycles used in these situations usually include: rough turning and boring cycle, rough facing cycle, finish turning and boring cycle, finish facing cycle, peck drilling cycle, step drilling cycle, chamfering cycle, and grooving cycle.

One caution should be noted by the programmer: Canned cycles valid for one controller can cause a crash situation if run on an incompatible controller if the controller does not stop and put out an alarm message when the canned cycle is encountered.

Appendix 7 contains a sample program utilizing canned cycles, along with an explanation of how the cycles are used.

SUMMARY

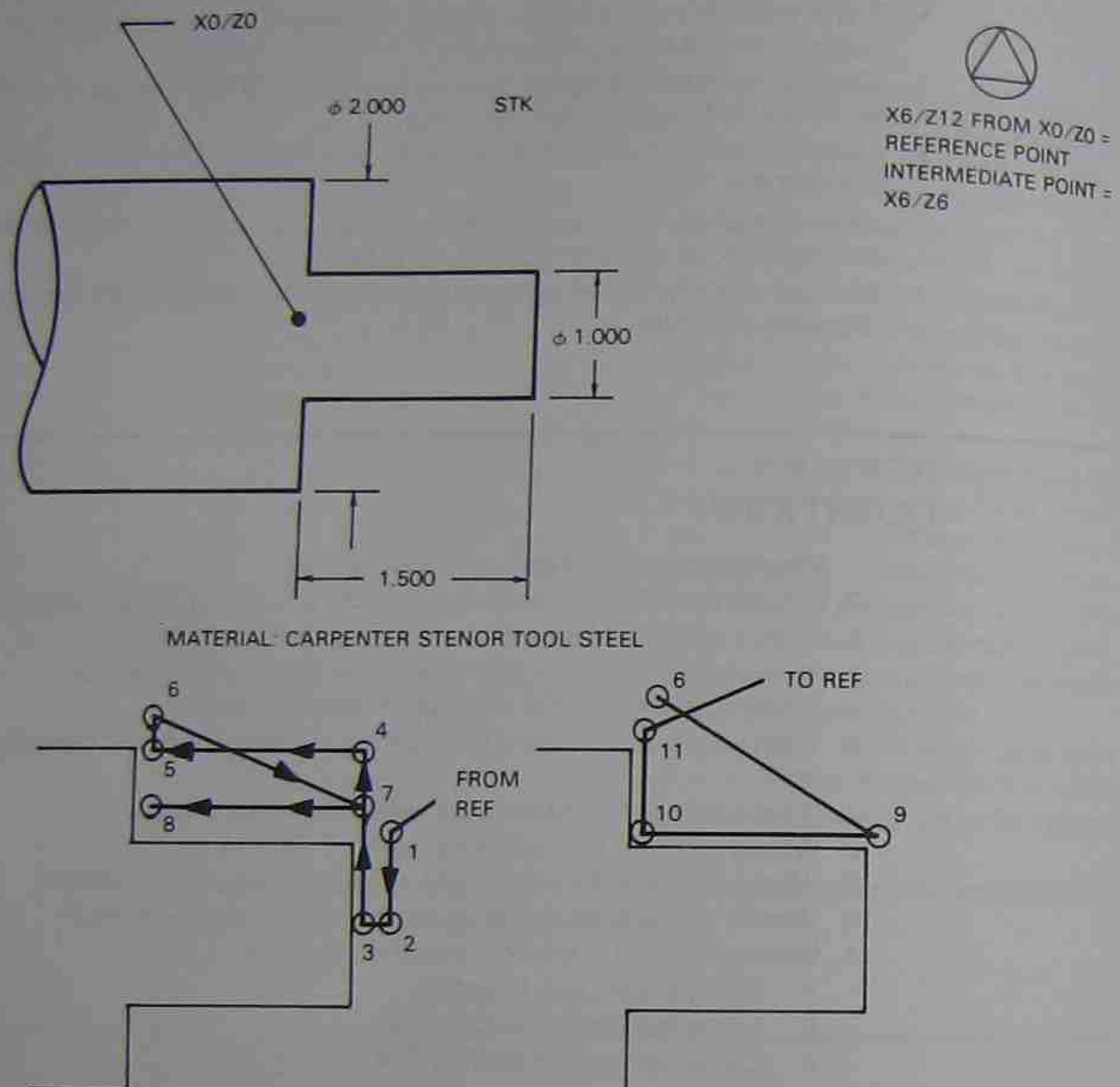
The important concepts presented in this chapter are:

- In diameter programming, the X-axis coordinates are one-half the actual tool movement.
- In radius programming, the X-axis coordinates and the tool movement are the same.
- G01, linear interpolation, is used for feedrate moves.
- Coordinates for taper turning must be calculated using trigonometry (or other math methods), just as when milling angles.
- G02 and G03 are used for circular interpolation.
- I and K are the addresses used to program the center points of an arc.

- The R address is used in place of I and K to program an arc using the arc radius instead of the arc centerpoints.
- Single pass threading cycles produce one threading cut. The cycle must be reinitiated for each threading pass.
- Multipass threading can produce an entire finished thread without additional programming.
- When threading, the Z axis tool advance must be calculated from the X-axis depth of cut by the formula Z = X TAN(30).
- Minimum starting and stopping distances must be calculated for use in a threading program.

REVIEW QUESTIONS

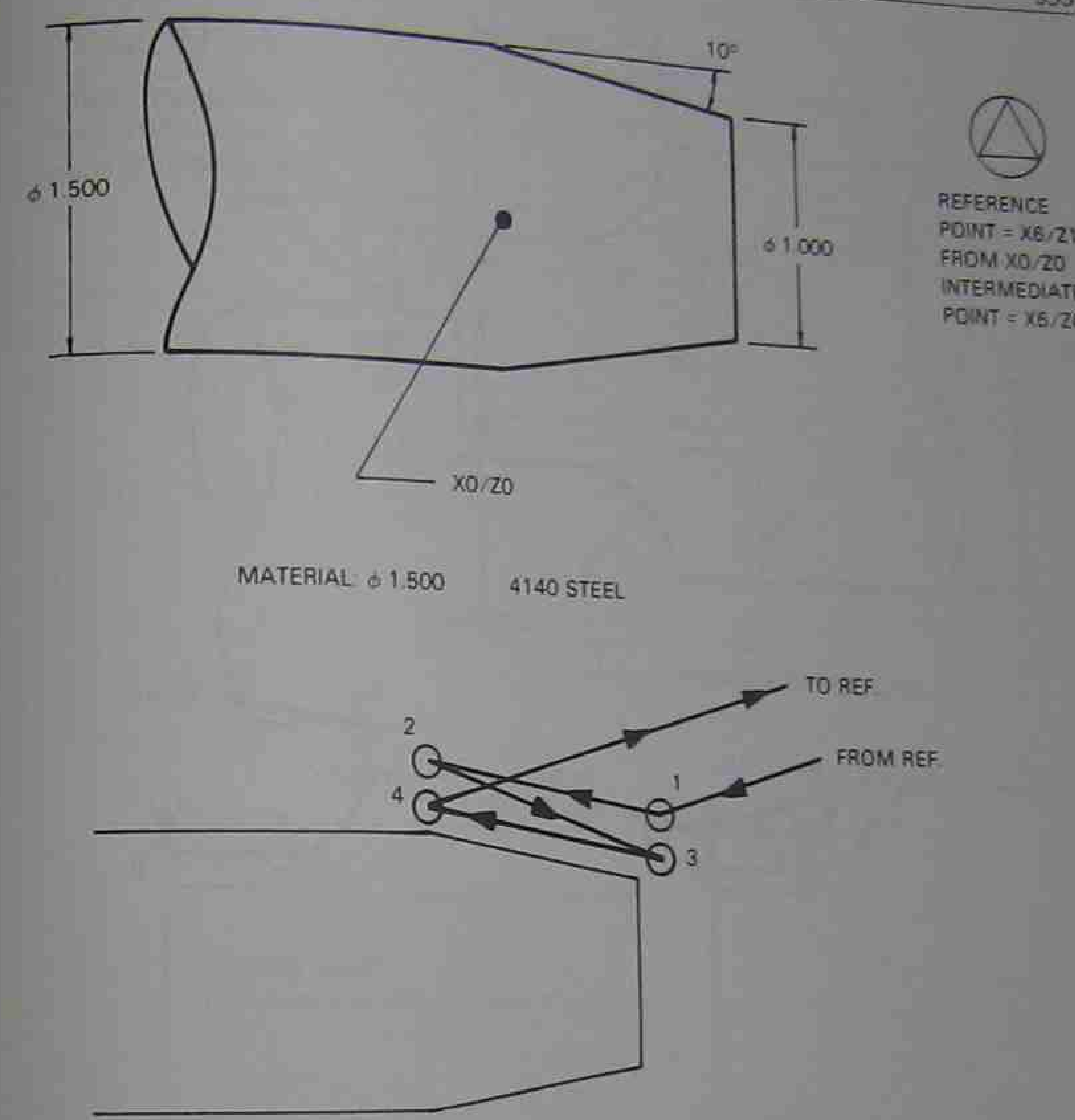
1. What G codes are used for feedrate moves?
2. What is the difference between diameter and radius programming?
3. Write a program to turn and face the part in Figure 14–18.
4. Write a program to turn the taper on the part in Figure 14–19.
5. What codes are used to institute circular interpolation?
6. What addresses are used to define the X and Z axis center point of an arc?
7. What address is used to define the arc using the arc radius?
8. Write a program to machine the part in Figure 14–20.
9. What is the code for single pass threading? What is the format?
10. What is the code for multipass threading? What is the format?
11. Write a program to thread the part in Figure 14–21:
    a. Using single pass threading.
    b. Using multipass threading G92.
    c. Using multipass threading G76.



MOVE TO	1	
ROUGH FACE	1 TO 2	LEAVE .010 STK
MOVE	2 TO 3	FOR FINISH.
FINISH FACE	3 TO 4	
ROUGH TURN	4 TO 5	ROUGH TURN
ROUGH FACE	5 TO 6	ϕ .495 STK
MOVE	6 TO 7	PER CUT
ROUGH TURN	7 TO 8	LEAVE .010 STK
ROUGH FACE	8 TO 6	FOR FINISH.

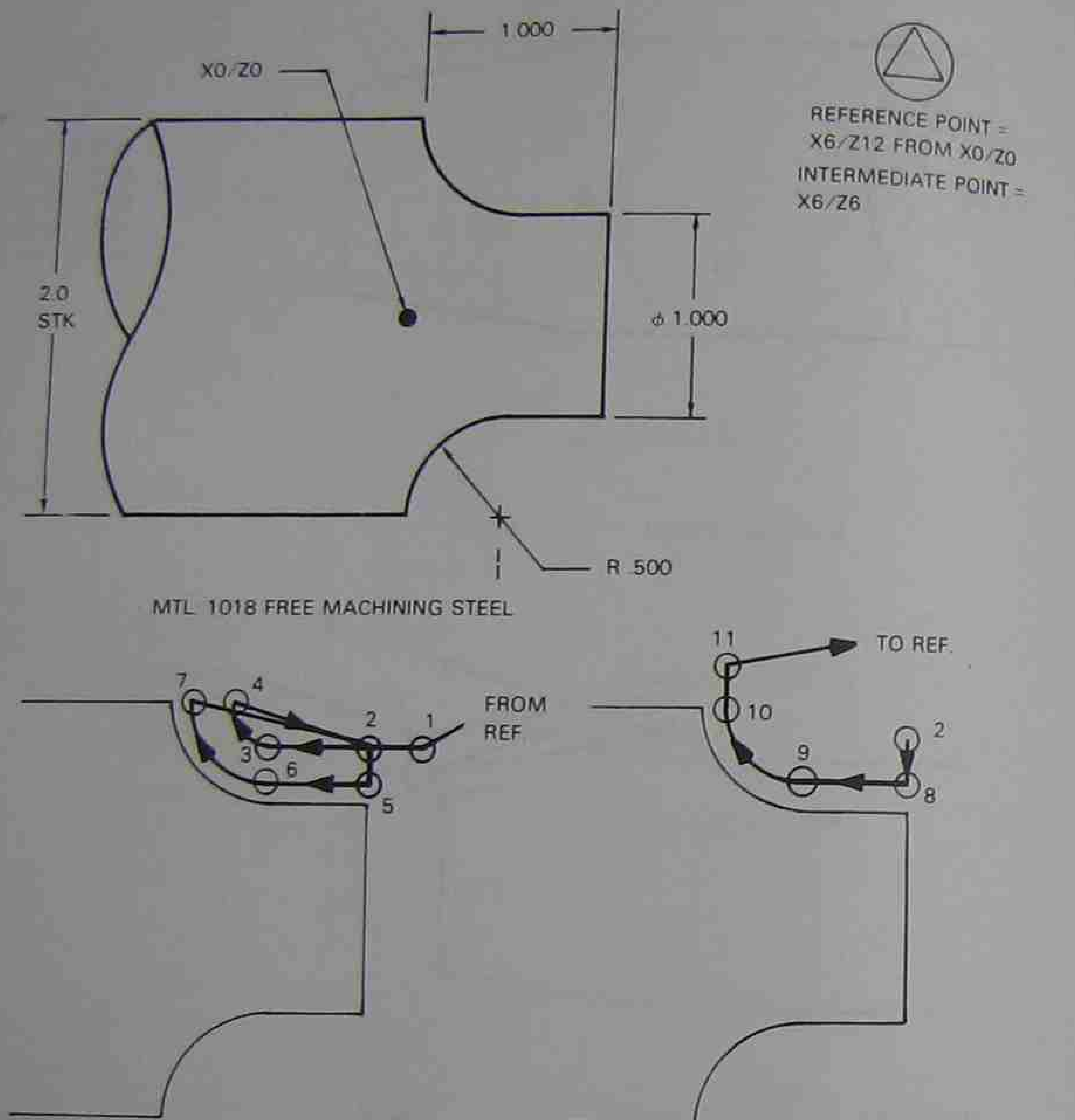
MOVE	6 TO 9
FINISH TURN	9 TO 10
FINISH FACE	10 TO 11
MOVE TO REF	

FIGURE 14–18
Part drawing for review question #3



MOVE TO 1	
ROUGH TURN	1 TO 2
MOVE	2 TO 3
FINISH TURN	3 TO 4
MOVE	TO REF

FIGURE 14–19
Part drawing for review question #4



MOVE TO	1		
FEED	1 TO 2		
ROUGH TURN	2 TO 3	REMOVE ϕ .495	STK
ROUGH ARC	3 TO 4		
MOVE	4 TO 2		
FEED	2 TO 5		
ROUGH TURN	5 TO 6	REMOVE ϕ .495	STK
ROUGH ARC	6 TO 7		
MOVE	7 TO 2		

FEED	2 TO 8		
FINISH TURN	8 TO 9	REMOVE ϕ.495	STK
FINISH ARC	9 TO 10		
MOVE	10 TO 11		
MOVE	11 TO REF.		

FIGURE 14–20
Part drawing for review question #8

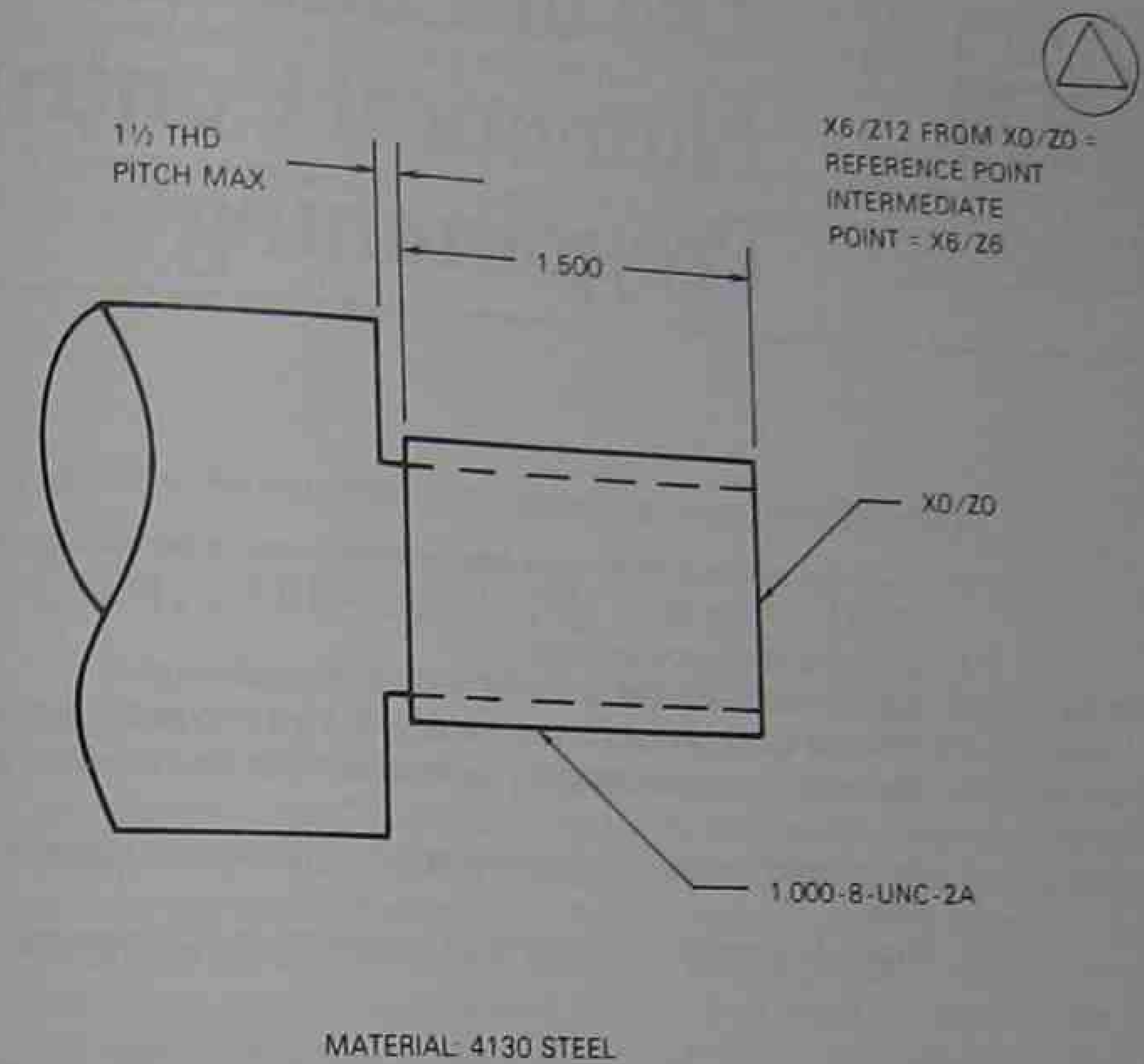


FIGURE 14–21
Part drawing for review question #11

CHAPTER 15

Use of Computers in Numerical Control Programming

OBJECTIVES Upon completion of this chapter, you will be able to:

- Describe the three basic ways computers are used in numerical control programming.
- Describe offline programming.
- Explain the advantages of computer-aided programming.
- Describe the three types of statements used in a computer-aided program.
- Understand basic part geometry and tool motion statements in APT and COMPACT II.
- Describe two types of computer graphics programming systems and how they differ.
- Explain how graphics programming simplifies writing NC programs.

Computers are becoming commonplace on shop floors. This is particularly true in numerical control programming. Only a few years ago, computers for NC programming were limited to large companies, but with the advent of inexpensive computer memory, even the smallest mold shops can now afford them. Computers can be used to help write NC programs in three basic ways: in offline programming, computer-aided programming, and computer graphics programming. This chapter will introduce all three of these uses. It is not the purpose of this text to teach computer-aided or computer graphics programming but, rather, to make the student aware of the far-reaching effects that computers are having on manufacturing.

OFFLINE PROGRAMMING TERMINALS

Technically speaking, all uses of computers to assist programming are offline programming methods. *Offline programming* is programming that is performed away from the machine, not at the CNC computer keyboard. An offline

programming terminal usually refers to a computer that is used as a text editor for writing programs. This type of programming station does not "aid" the programmer except by allowing the program to be entered into the computer exactly as if it were being entered via the MDI console. The advantage is that one program may be written while another program is being run on the machine. The program being written is simply saved on a computer disk, magnetic tape, punched tape, or a combination of these three.

COMPUTER-AIDED PROGRAMMING

Computer-aided programming had its beginnings in the 1950s when point-to-point tape machinery made manual programming an enormously laborious task. Computer-aided programming languages act as translators. The programmer "talks" to the computer via the keyboard in a computer language designed specifically for numerical control programming. The main advantage of a computer-aided language is that a number of different numerical control machines can be programmed using the same language. The computer is told the machine and tools to be used, the part to be made, and the path the cutter will take. The computer takes all this information, and through another program, called a postprocessor, writes an NC program (usually in word address) to machine the part on the NC machine specified. This type of programming requires a large amount of computer memory; the computer commands vary, depending on the programming language used.

The language a particular company uses depends on the parts it produces, the machines it uses, the cost of the programming system, and the computers that are available. Following are some of the more common programming languages.

APT (Automatic Programmed Tools). APT is the oldest and the largest of the computer-aided programming languages. It can be used on only large computers and will perform the mathematical calculations required for complex curved surfaces using four and five axes.

AD-APT (Adaption of APT). This is a version of APT that can run on smaller computers. It uses about half the commands of APT and can be used for two-axis contouring with a third axis of linear motion.

AUTOMAP (Automatic Machining Program). AUTOMAP is another adaptation of APT. It will run on medium size computers, has a limited number of commands, and is used for programming straight lines and circles.

COMPACT II. Compact II can accomplish the same tasks as APT but is limited to three-axis work. The main difference between them is that Compact II uses a somewhat more conversational set of commands, without some of the strict rules of syntax that must be observed in APT. The computer will also aid in debugging the finished program by interactive conversation with the programmer.

UNIAPT. UNIAPT is very similar to APT but is designed to run on dedicated minicomputers—that is, small computers that are used for only one task. UNIAPT will handle four- and five-axis programming.

NUFORM. NUFORM differs from most other computer-aided languages in that the programmer inserts codes or dimensions in their appropriate location within the NUFORM format. NUFORM uses numeric rather than alphabetic codes.

In this chapter, APT and COMPACT II will be used as examples. An APT or COMPACT II program has three parts: part geometry definitions, auxiliary function statements (tool changes, speeds, feedrates, etc.), and tool motion statements.

Part Geometry Definitions in APT

Parts are defined to the computer by describing features such as points, lines, planes, circles, and their relationships to each other. Following are some of the simpler APT geometry commands.

To Define a Point. Points are the basis for defining other geometric features. Points may be defined by Cartesian coordinates or by relationship to other geometric features such as lines or circles. The following examples are all point definitions.

Point defined by Cartesian coordinates, Figure 15–1(a):
SYMBOL FOR POINT = POINT/X,Y,Z
Example: P1 = POINT/6,6,6
Meaning: P1 = a point X6.0000, Y6.0000, Z6.0000 from 0/0.

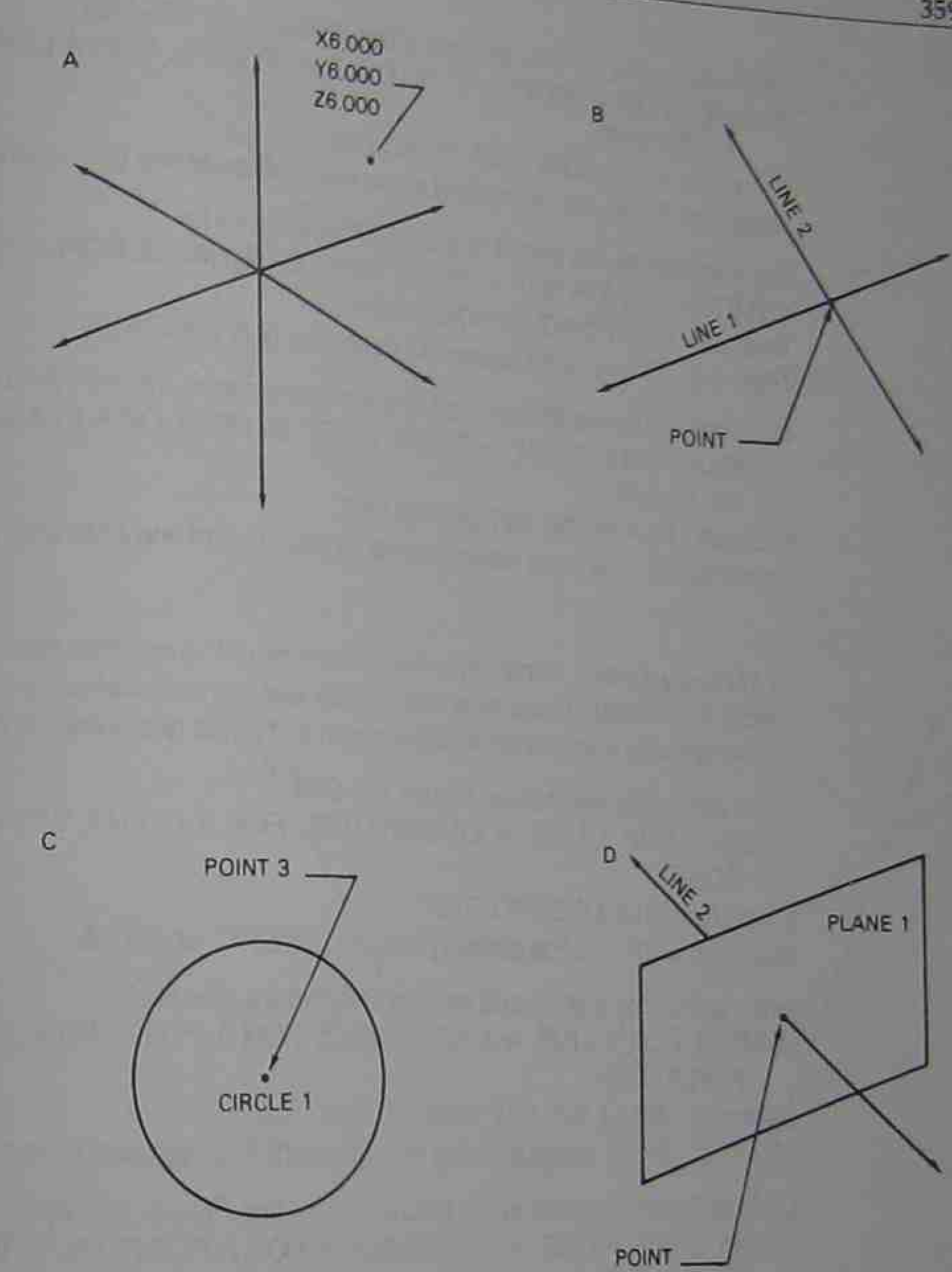


FIGURE 15–1
Defining a point in APT

Point defined by the intersection of two lines, Figure 15–1(b):
SYMBOL FOR A POINT = POINT/INTOF, SYMBOL FOR A LINE, SYMBOL FOR A LINE
Example: P2 = POINT/INTOF,LN1,LN2
Meaning: P2 = a point located at the intersection of line LN1 and line LN2.

Point defined by the center of a circle, Figure 15–1(c):
SYMBOL FOR A POINT = POINT/CENTER, SYMBOL FOR A CIRCLE
Example: P3 = POINT/CENTER,CIR1
Meaning: P3 = a point located at the center of circle CIR1.

Point defined by the intersection of a line and a plane, Figure 15–1(d):
SYMBOL FOR A POINT = POINT/INTOF, SYMBOL OF A PLANE, SYMBOL OF A LINE
Example: P4 = POINT/INTOF,PN2,LN3
Meaning: P4 = a point located at the intersection of line LN3 with plane PN2.

To Define a Line. Lines may be defined by using coordinates, other lines, points, and circles. Lines extend to infinity and are calculated by the computer mathematically from the information given it. It takes two points to define a line.

Lines defined by two points, Figure 15–2(a):
SYMBOL FOR A LINE = LINE/SYMBOL FOR A POINT, SYMBOL FOR A POINT
Example: LN1 = LINE/PT1,PT2
Meaning: LN1 = a line passing through point PT1 and PT2.

Lines defined by a point and parallel line, Figure 15–2(b):
SYMBOL FOR A LINE = LINE/SYMBOL FOR A POINT, PARLEL, SYMBOL FOR A LINE
Example: LN2 = LINE/PT2,PARLEL,LN1
Meaning: LN2 = a line passing through point PT2, parallel to line LN1.

Lines defined by a point and a perpendicular line Figure 15–2(c):
SYMBOL FOR A LINE = LINE/SYMBOL FOR A POINT, PERPTO, SYMBOL FOR A LINE
Example: LN3 = LINE/PT3,PERPTO,LN2
Meaning: LN3 = a line passing through point PT3, perpendicular to line LN2.

Lines defined by a point and tangency to a circle, Figure 15–2(d):
SYMBOL FOR A LINE = LINE/SYMBOL FOR A POINT, LEFT/RIGHT, TANTO, SYMBOL FOR A CIRCLE
Example: LN4 = LINE/PT2,LEFT,TANTO,CIR1
Meaning: LN4 is the line passing through point 2 and tangent to a circle with radius 1 on the left side.

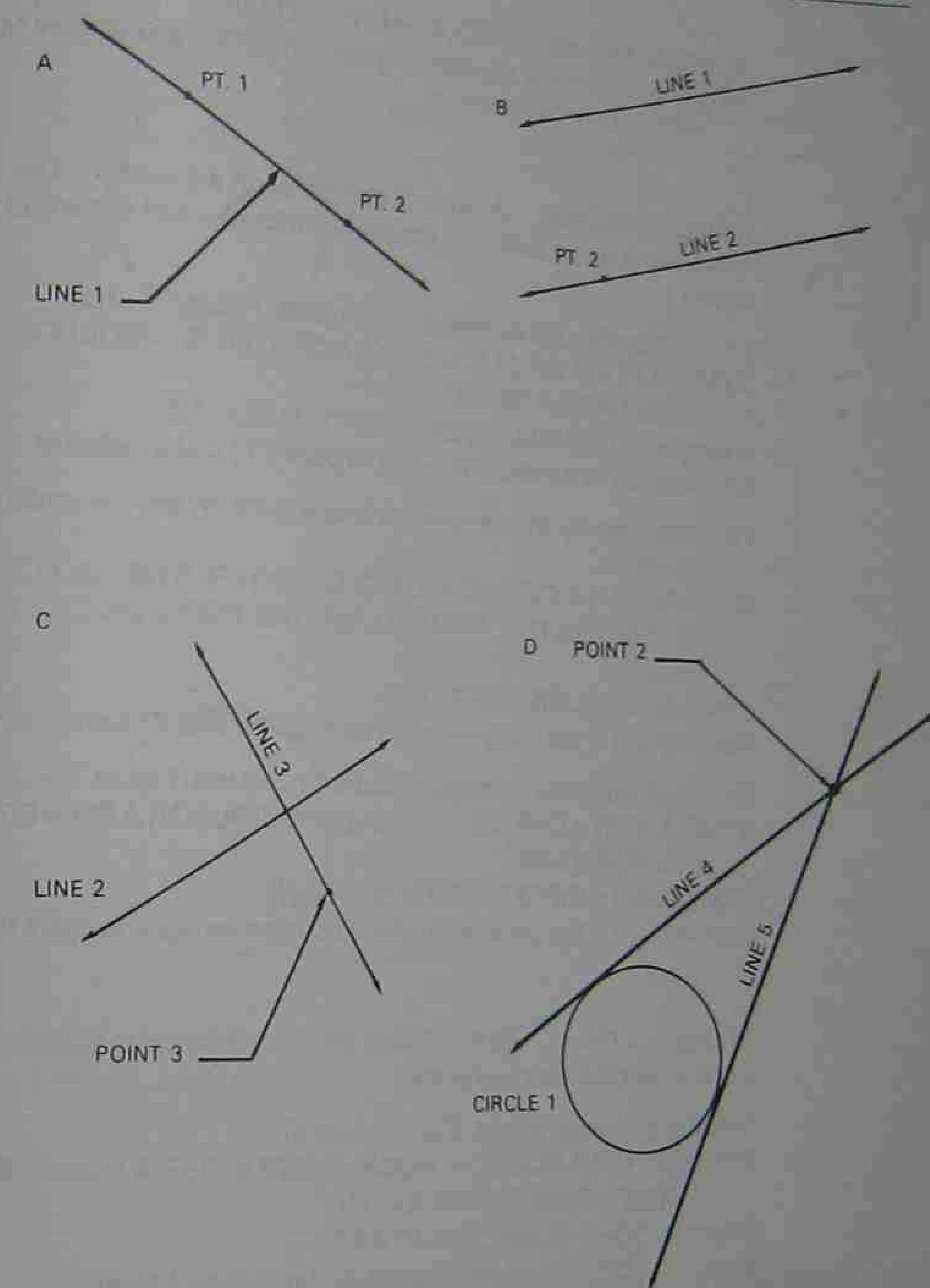


FIGURE 15–2
Defining a line in APT

Example: LN5 = LINE/PT3,RIGHT,TANTO,CIR1
Meaning: LN5 is the line passing through point 3 and tangent to a circle with radius 1 on the right side.

To Define a Circle. A circle is defined by an arc section. The section of the circle that is a part feature is programmed later as part of the tool motion statements.

Circle defined by the centerpoint and radius, Figure 15–3(a):
SYMBOL FOR A CIRCLE = CIRCLE/CENTER, SYMBOL FOR A POINT, RADIUS, RADIUS VALUE
Example: C1 = CIRCLE/CENTER,PT1,RADIUS,1.0
Meaning: C1 is a circle with center at point PT1 and a radius of 1.0 inch.

Circle defined by the centerpoint and a point on the circumference, Figure 15–2(b):
SYMBOL FOR A CIRCLE = CIRCLE/CENTER, SYMBOL FOR THE POINT AT THE CENTER, SYMBOL FOR THE POINT ON THE CIRCUMFERENCE
Example: C2 = CIRCLE/PT2,PT4
Meaning: C2 is the circle with PT2 as its center and PT4 on its circumference.

Circle defined by the centerpoint and a tangent line, Figure 15–3(c):
SYMBOL FOR A CIRCLE = CIRCLE/SYMBOL FOR A POINT, TANTO, SYMBOL FOR A LINE
Example: C3 = CIRCLE/CENTER,PT5,LN1
Meaning: C3 is the circle with point 5 as its centerpoint, tangent to line 1.

To Define a Plane. Part surfaces are often defined by planes. Following are some simple definitions of planes.

Defining a plane by points, Figure 15–4(a):
SYMBOL FOR A PLANE = PLANE/SYMBOL FOR A POINT, SYMBOL FOR A POINT, SYMBOL FOR A POINT
Example: TOP = PLANE/PT1,PT2,PT3
Meaning: TOP is the plane passing through points 1, 2, and 3.

Defining a plane by a point and a parallel plane, Figure 15–4(b):
SYMBOL FOR A PLANE = PLANE/SYMBOL FOR A POINT, PARLEL, SYMBOL FOR A PLANE
Example: BOTTOM = PLANE/PT4,PARLEL,TOP
Meaning: BOTTOM is the plane passing through point 4 and parallel with plane TOP.

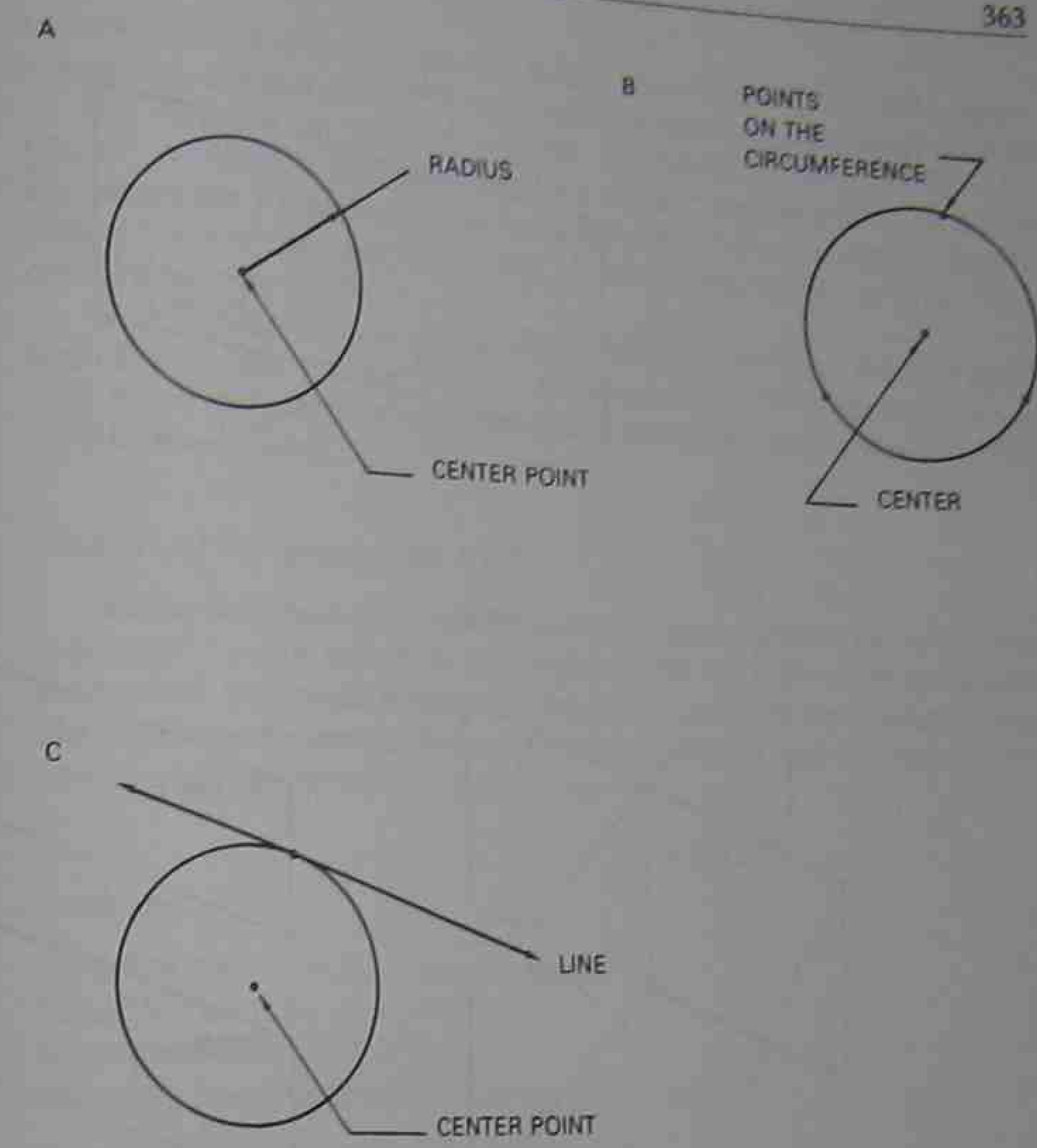


FIGURE 15–3
Defining a circle in APT

Defining a plane through two points, perpendicular to another plane, Figure 15–4(c):
SYMBOL FOR A PLANE = PLANE/PERPTO, SYMBOL FOR A PLANE, SYMBOL FOR A POINT, SYMBOL FOR A POINT
Example: PLN3 = PLANE/PERPTO,PLN2,PT3,PT5
Meaning: PLN3 is the plane perpendicular to plane PLN2, and passing through points PT3 and PT5.

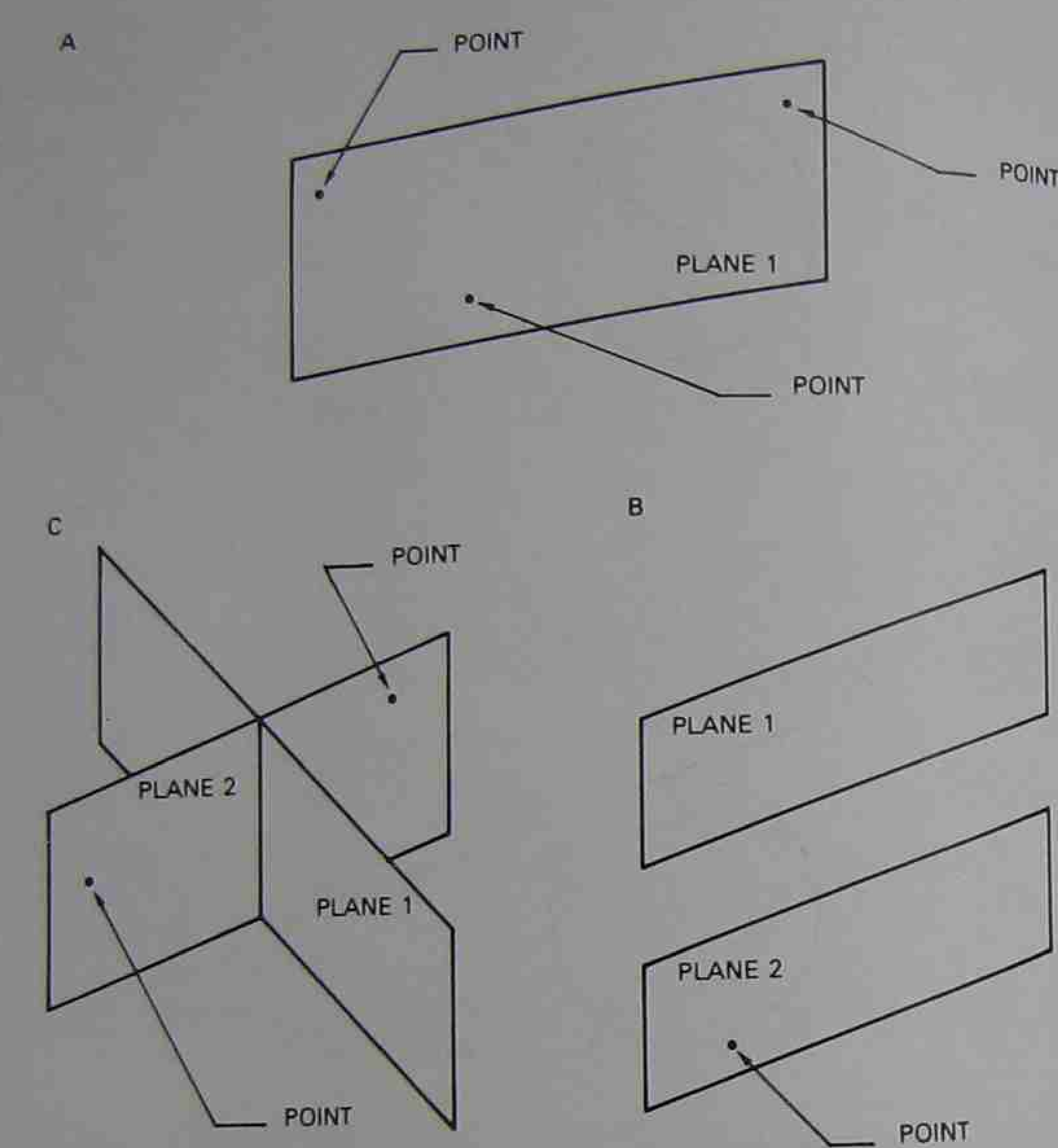


FIGURE 15–4
Defining a plane in APT

Geometric definitions can also contain modifiers, which help in clarifying a definition. These modifiers are:

XLARGE	XSMALL
YLARGE	YSMALL
ZLARGE	ZSMALL

LARGE means that the side with the greater value is specified. SMALL specifies the side of lesser value. The value is determined by the coordinate position of the feature in question. The statement LN2 = LINE/PARLEL,LN1,YLARGE,1.000 means that LN2 is line 2, parallel to line LN1 on the side where the Y coordinates are the largest and 1.000 inch away from LN1. YSMALL would mean that the line was 1.000 inch away on the side where the Y coordinates were smaller.

Other types of features can be used to define the geometric shape of a part, such as cylinders, vectors, surfaces, and angles. An APT dictionary, or a textbook on computer-aided programming, will contain a comprehensive list of APT geometric definitions.

Auxiliary Statements in APT

Before defining a cutter path using tool motion statements, the various tools to be used must be defined to the computer. The tool statements are auxiliary statements. The spindle speeds and feedrates used with the various tools are also auxiliary statements. Following are examples of tool, machine, spindle speed, and feedrate auxiliary statements.

Feedrates.

To define a feedrate in inches per minute:
FEDRAT/[feedrate value],IPM
Example: FEDRAT/20,IPM
Meaning: Feedrate is set to 20 inches per minute

To define a feedrate in inches per revolution:
FEDRAT/[feedrate value],IPR
Example: FEDRAT/.007,IPR
Meaning: Feedrate is set to .007 inch per revolution.

Tool Changes.

To change a tool when premeasured tools are not used:

LOADTL/[tool no.],ADJUST,[tool length offset register number]
Example: LOADTL/1,AJ,1
Meaning: Results in tape code to load tool number 1 into the spindle and call up tool length offset register number 1.

To change a tool when premeasured tools are used:
LOADTL/[tool no.],LENGTH,[tool length],ADJUST,[tool register]
Example: LOADTL/2,LENGTH,9,ADJUST,2

Meaning: Results in tape code to load tool number 2 into the spindle and call up tool length offset register number 2. All Z-axis coordinates will be modified by the 9-inch length of the tool.

Spindle Speeds.

To define a clockwise spindle speed:
SPINDL/[RPM],CLW
Example: SPINDL/2000,CLW
Meaning: Spindle speed is 2000 RPM in a clockwise direction.

To define a counterclockwise spindle speed:
SPINDL/[RPM],CCLW
Example: SPINDL/2300,CCLW
Meaning: Spindle speed is 2300 RPM in a counterclockwise direction.

Machine Statements. Machine statements tell the computer where to find the instructions to write a program for a particular machine. The syntax is as follows:

MACHIN/[machine identifier]
Example: MACHIN/UNIV 1
Meaning: The machine to program is Universal machine #1.

Tool Statements. Tool statements define to the computer the various tools that will be used during the program. There are two types of cutter definitions: simple and complex. Simple cutters have only a diameter and radius. Complex cutters have multiple features. Figure 15–5 shows both simple and complex cutters. The tool definition is done with a CUTTER statement.

Simple cutter:
CUTTER/d,r
Example: CUTTER/.75,.0625
Meaning: The cutter has a diameter of .750 inch and a corner radius of .0625 inch. The height of the cutter (h) is assumed to be 5.0 inches.

Complex cutter:
CUTTER/d,r,e,f,a,b,h
Example: CUTTER/.75,.0625,.3125,.0837,15,5,6
Meaning: The cutter has a diameter of .750 inch, a corner radius of .0625 inch, E distance is .3125 inch, F distance is .0837 inch, angle A is 15 degrees, angle B is 5 degrees, and the height of the cutter is 6.0 inches.

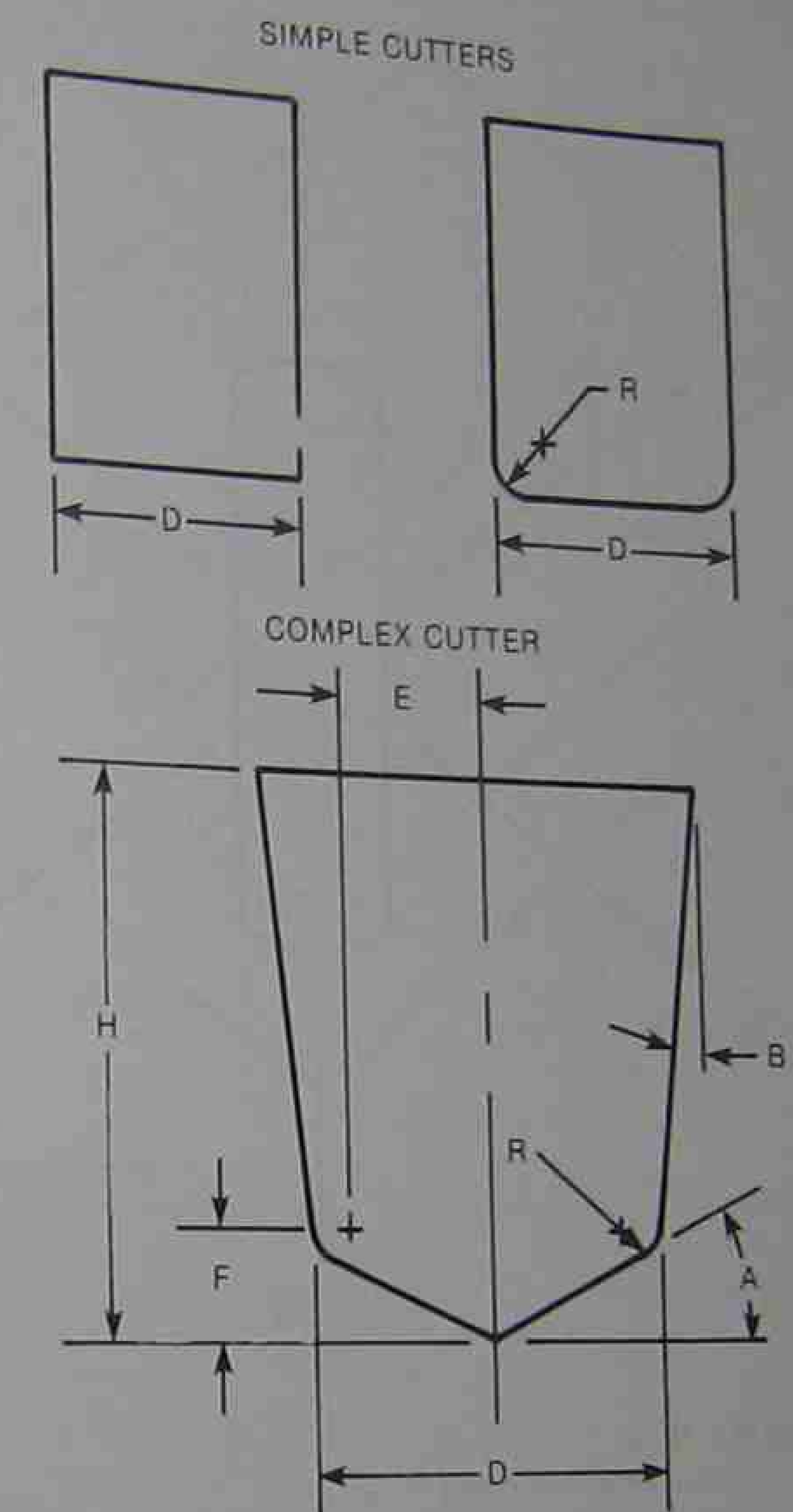


FIGURE 15–5
Cutter dimensions

Tool Motion Statements in APT

Once the part geometry has been defined, the cutter path can be described to the computer using what are known as tool motion statements. Tool motion is controlled using the following commands: GOUP, GODOWN, GOFWD, GOBACK, GORGT (right), and GOLFT (left). The relationship of these commands can be seen in Figure 15–6.

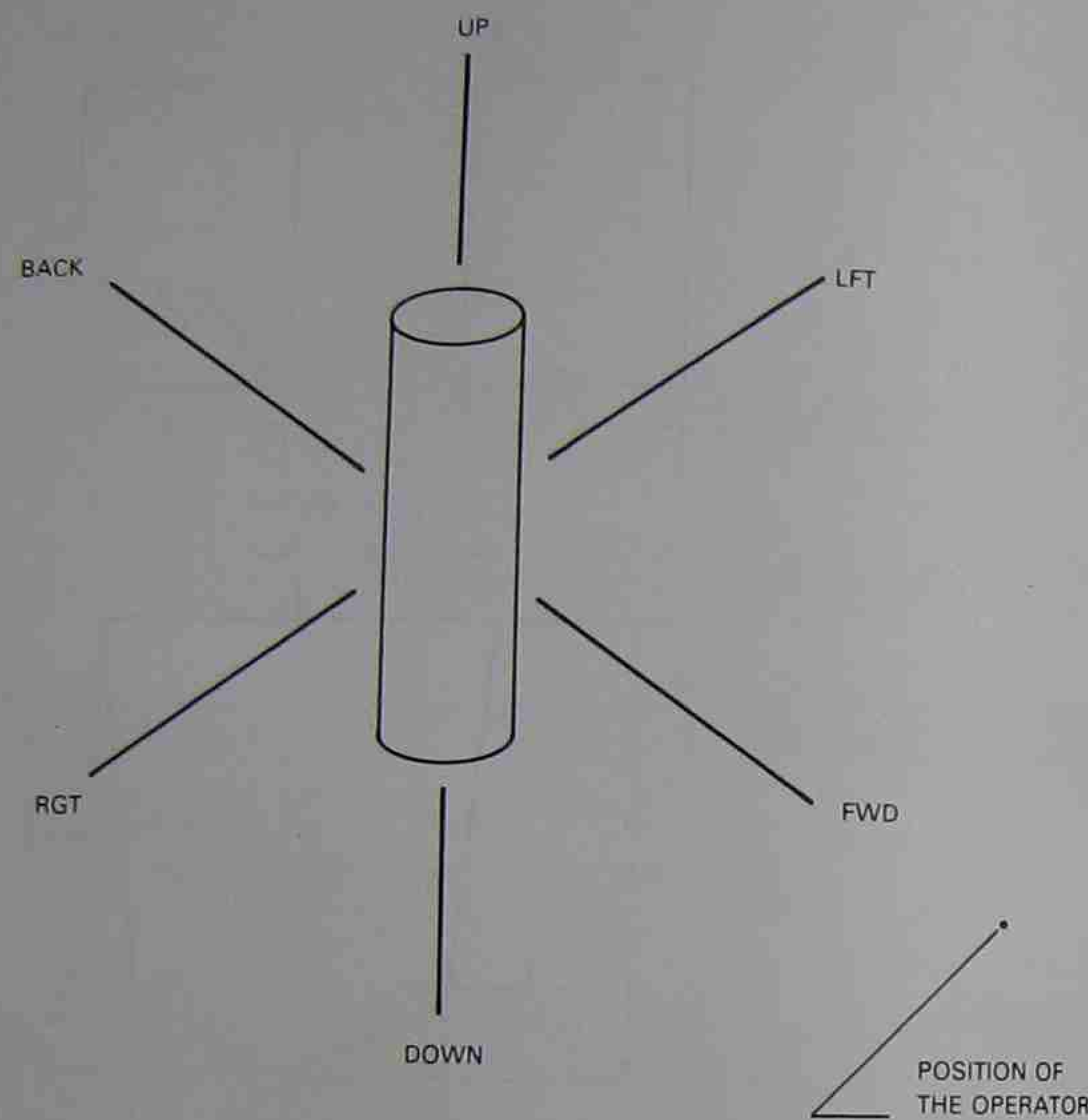


FIGURE 15–6
Tool motion commands in APT

Tool movement is also controlled by three surfaces known as the part surface, the drive surface, and the check surface (see Figure 15–7). The tool bottom is guided by the part surface, while the side of the tool is guided by the drive surface. Once initiated, tool motion continues until stopped by a check surface, which signals the end of the cut.

The six tool commands are used with one of four modifiers to define the check surface. These modifiers—TO, ON, PAST, and TANTO—are illustrated in Figure 15–8.

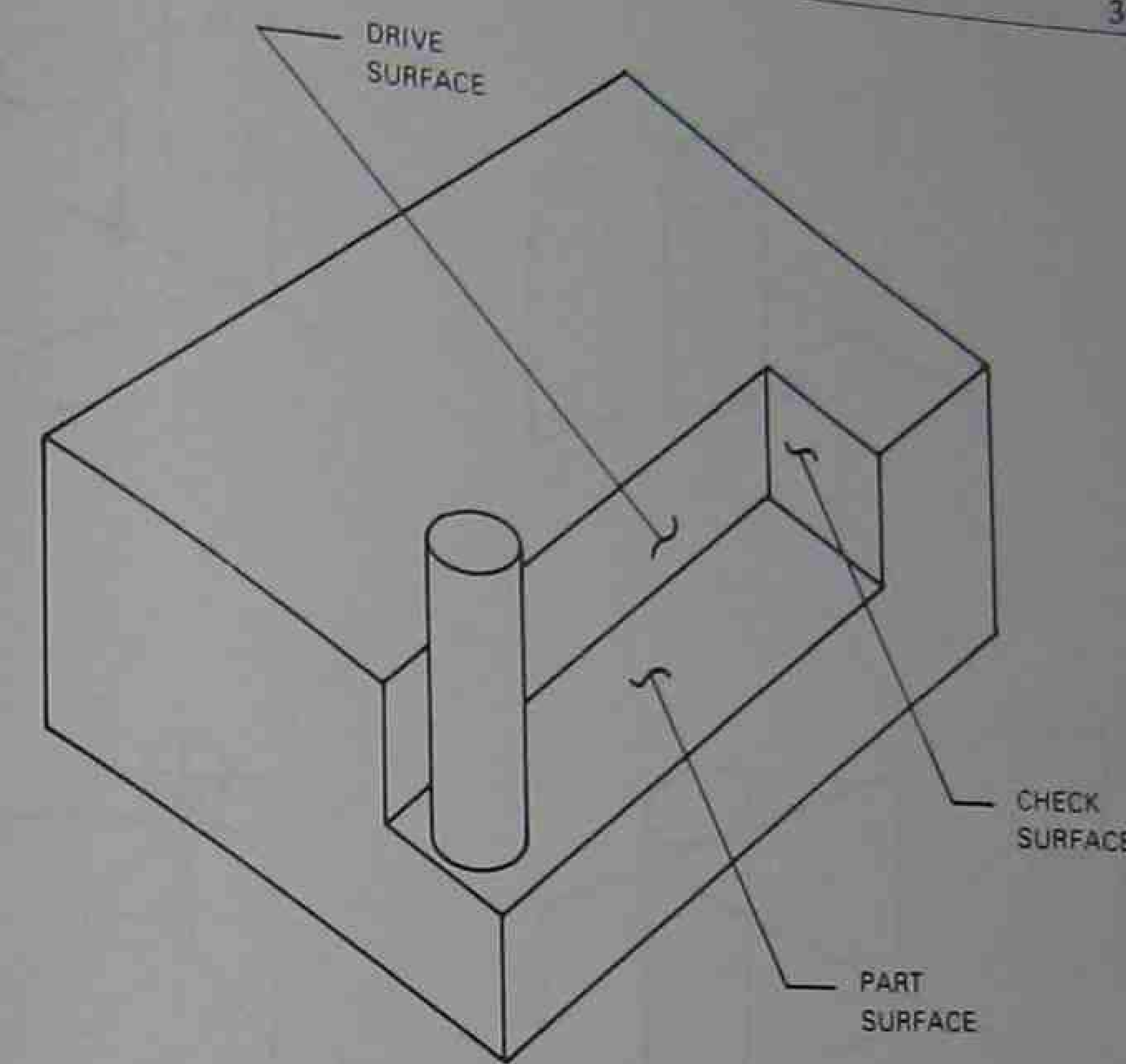


FIGURE 15–7
Controlling surfaces in APT

Figure 15–9 shows a rectangular piece defined by the APT language. The four sides have been labeled S1 through S4. Tool motion statements to move the tool from position A to B, and then mill the periphery of the part would be as follows:

```
GOTO/S1,PASTS2
GOFWD/S1,PASTS4
GORGT/S4,PASTS3
GORGT/S3,PASTS2
GORGT/S2,PASTS1
```

Figure 15–10 shows a part to be APT programmed. Figure 15–11 identifies the geometric figures used to define the part. Figure 15–12 is a simplified APT program written to mill the part periphery.

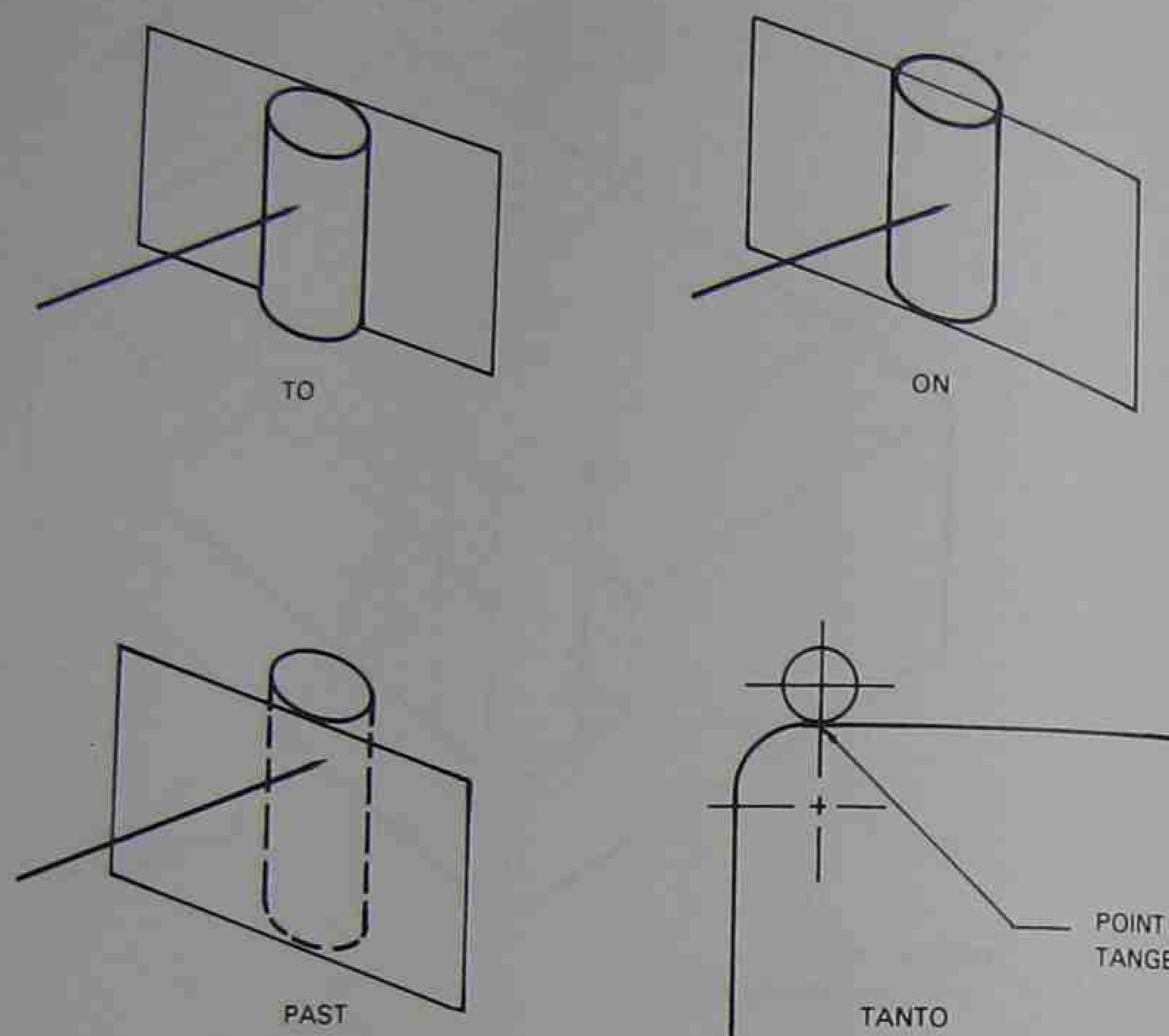


FIGURE 15–8
Modifiers for tool commands in APT

Part Geometry Definitions in COMPACT II

To Define a Point. Points may be defined by Cartesian coordinates or by relationship to other geometric features such as lines or circles.

Point defined by Cartesian coordinates, Figure 15–1(a):
DPTn,nXB,nYB,nZB
Example: DPT1,6XB,6YB,6ZB
Meaning: Point 1 is the point X6.0000, Y6.0000, Z6.0000 from 0/0. The B specifies that a work coordinate system, called a BASE, is being referenced.

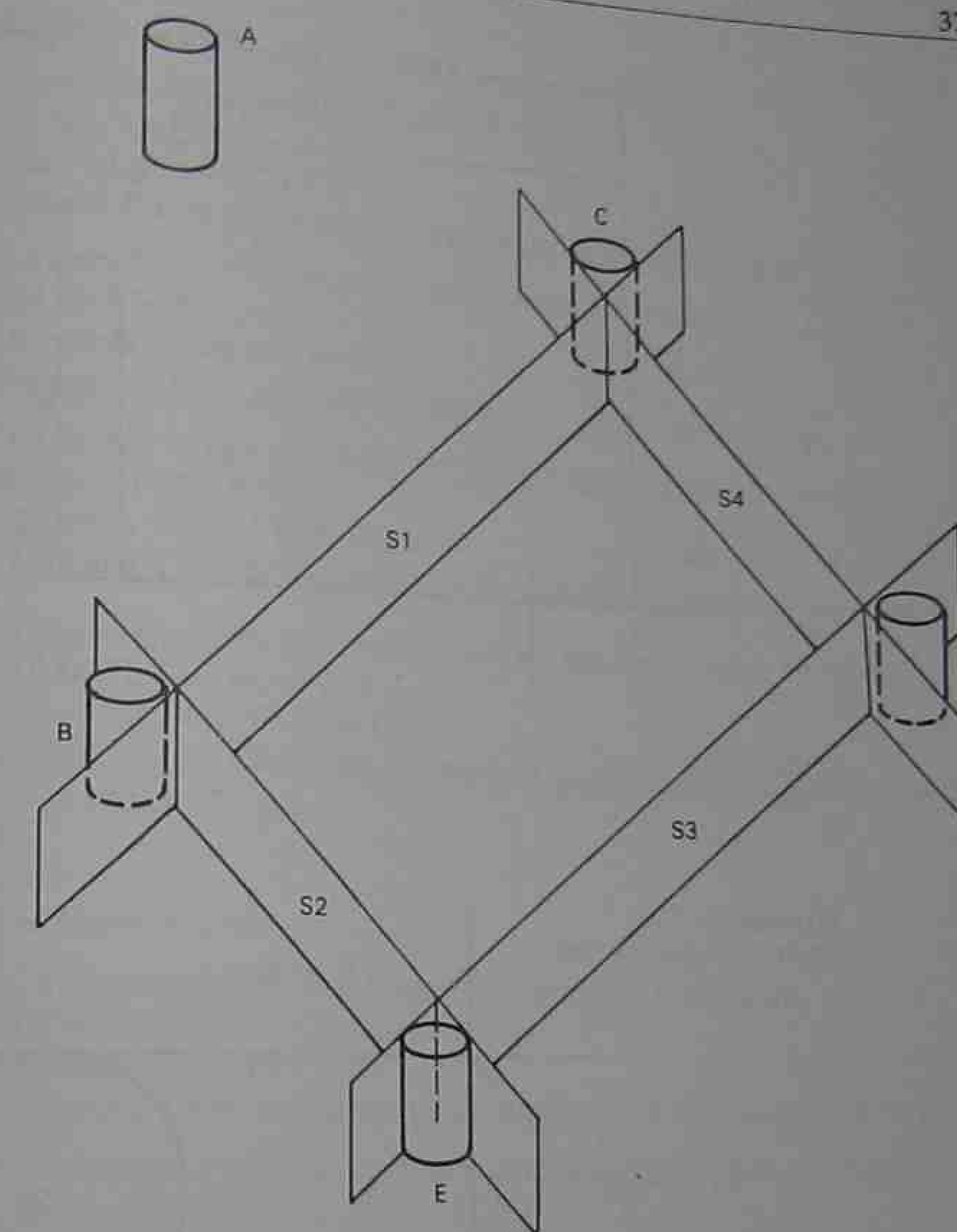


FIGURE 15–9

Point defined by the intersection of two lines, Figure 15–1(b):
DPTn,LNn,LNn
Example: DPT2,LN1,LN2
Meaning: Point 2 is the point formed by the intersection of line 1 and line 2.

Point defined by the center of a circle, Figure 15–1(c):
DPTn,CIRn,CNTR
Example: DPT3,CIR1,CNTR
Meaning: Point 3 is the point at the center of circle 1.

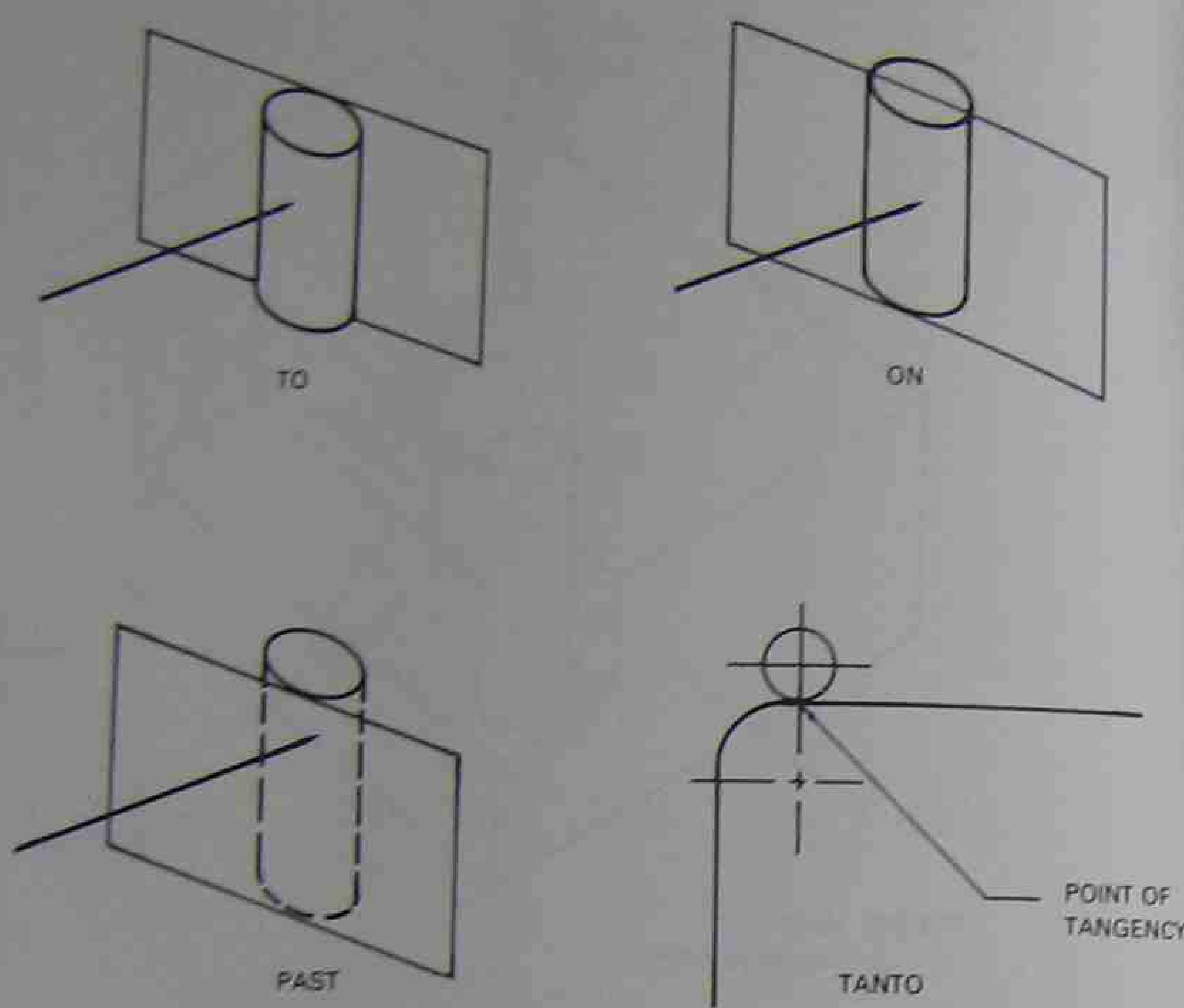


FIGURE 15–8
Modifiers for tool commands in APT

Part Geometry Definitions in COMPACT II

To Define a Point. Points may be defined by Cartesian coordinates or by relationship to other geometric features such as lines or circles.

Point defined by Cartesian coordinates, Figure 15–1(a):
DPTn,nXB,nYB,nZB
Example: DPT1,6XB,6YB,6ZB
Meaning: Point 1 is the point X6.0000, Y6.0000, Z6.0000 from 0/0. The B specifies that a work coordinate system, called a BASE, is being referenced.

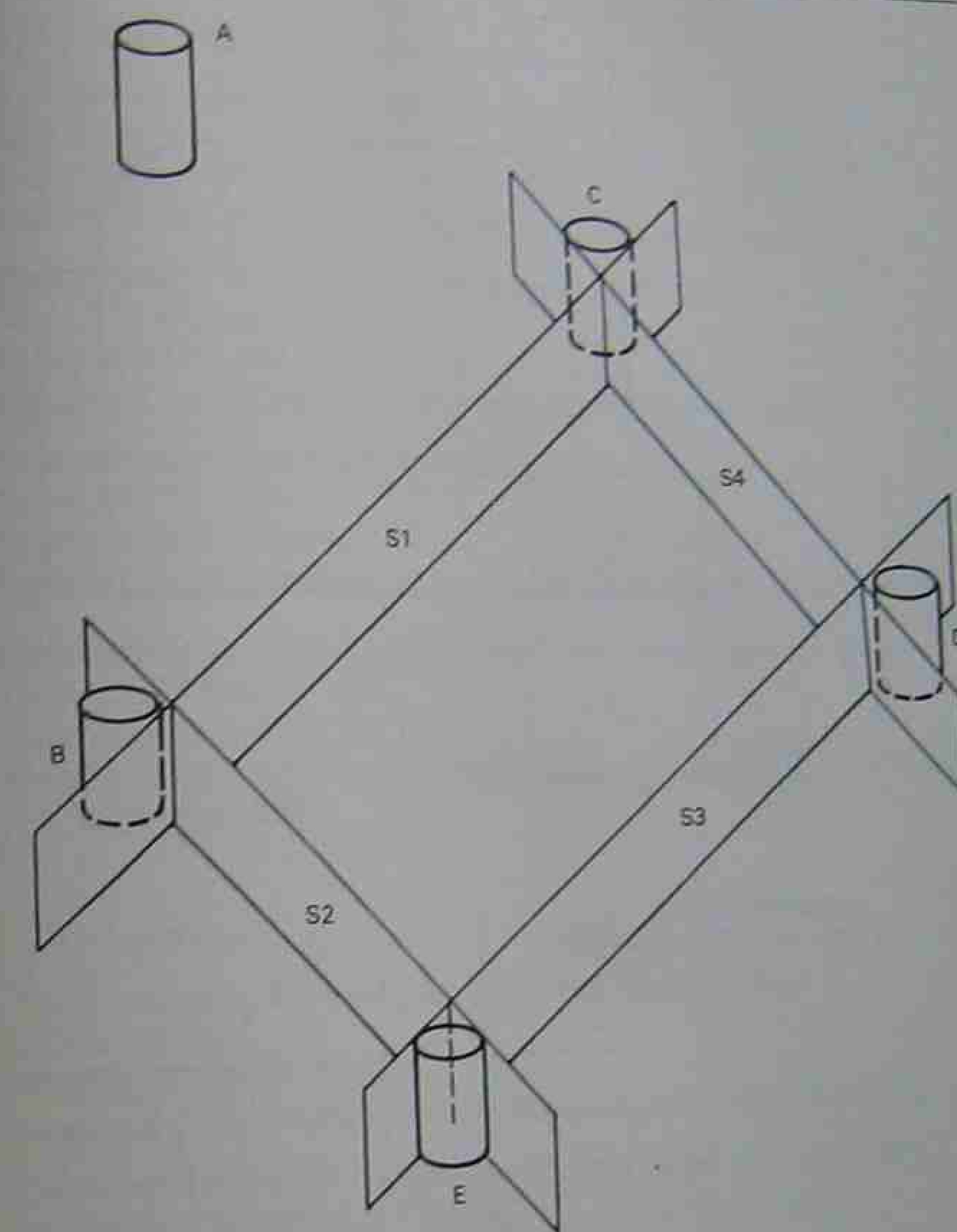


FIGURE 15–9

Point defined by the intersection of two lines, Figure 15–1(b):
DPTn,LNn,LNn
Example: DPT2,LN1,LN2
Meaning: Point 2 is the point formed by the intersection of line 1 and line 2.

Point defined by the center of a circle, Figure 15–1(c):
DPTn,CIRn,CNTR
Example: DPT3,CIR1,CNTR
Meaning: Point 3 is the point at the center of circle 1.

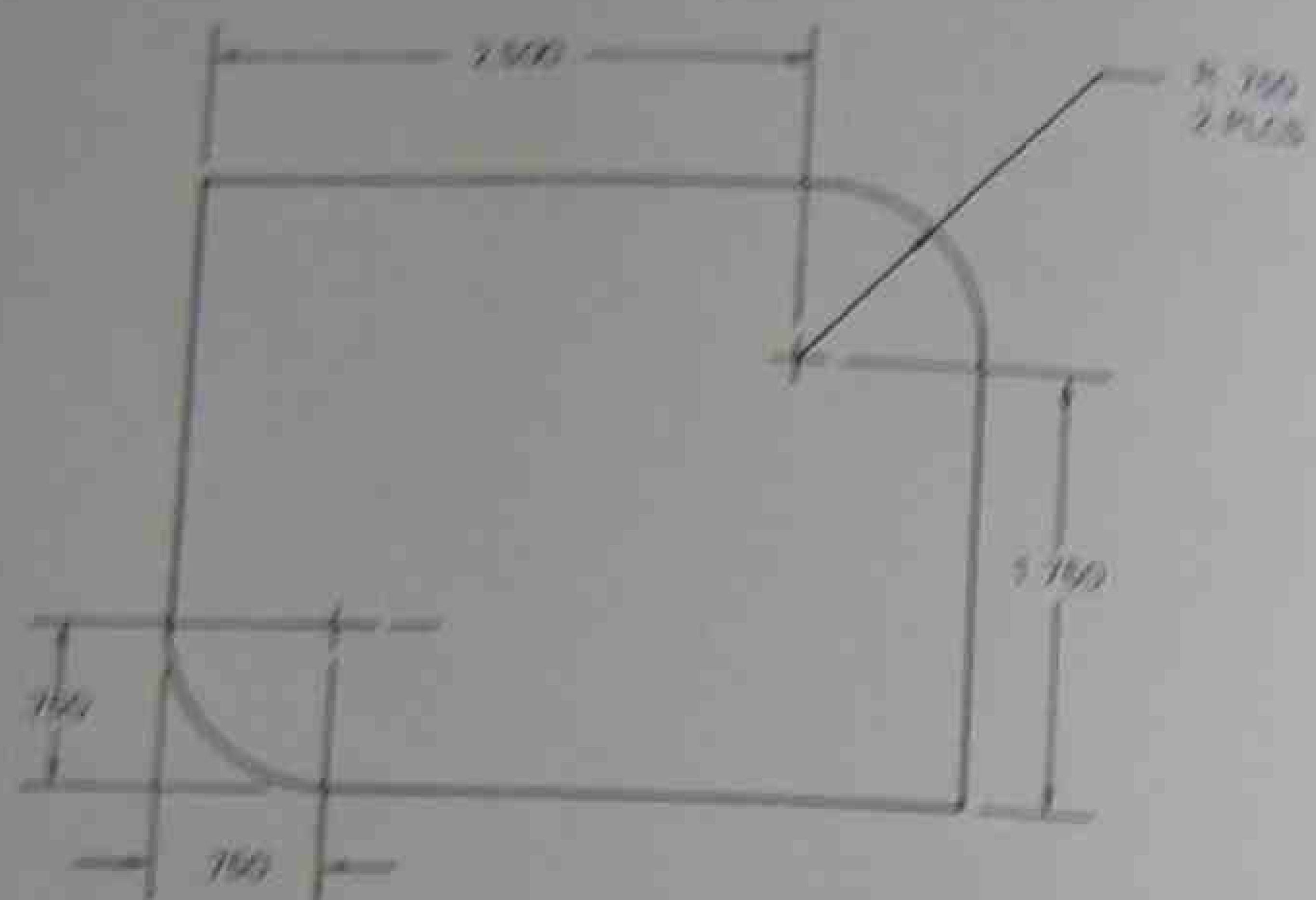


FIGURE 15–10
Part to be APT programmed

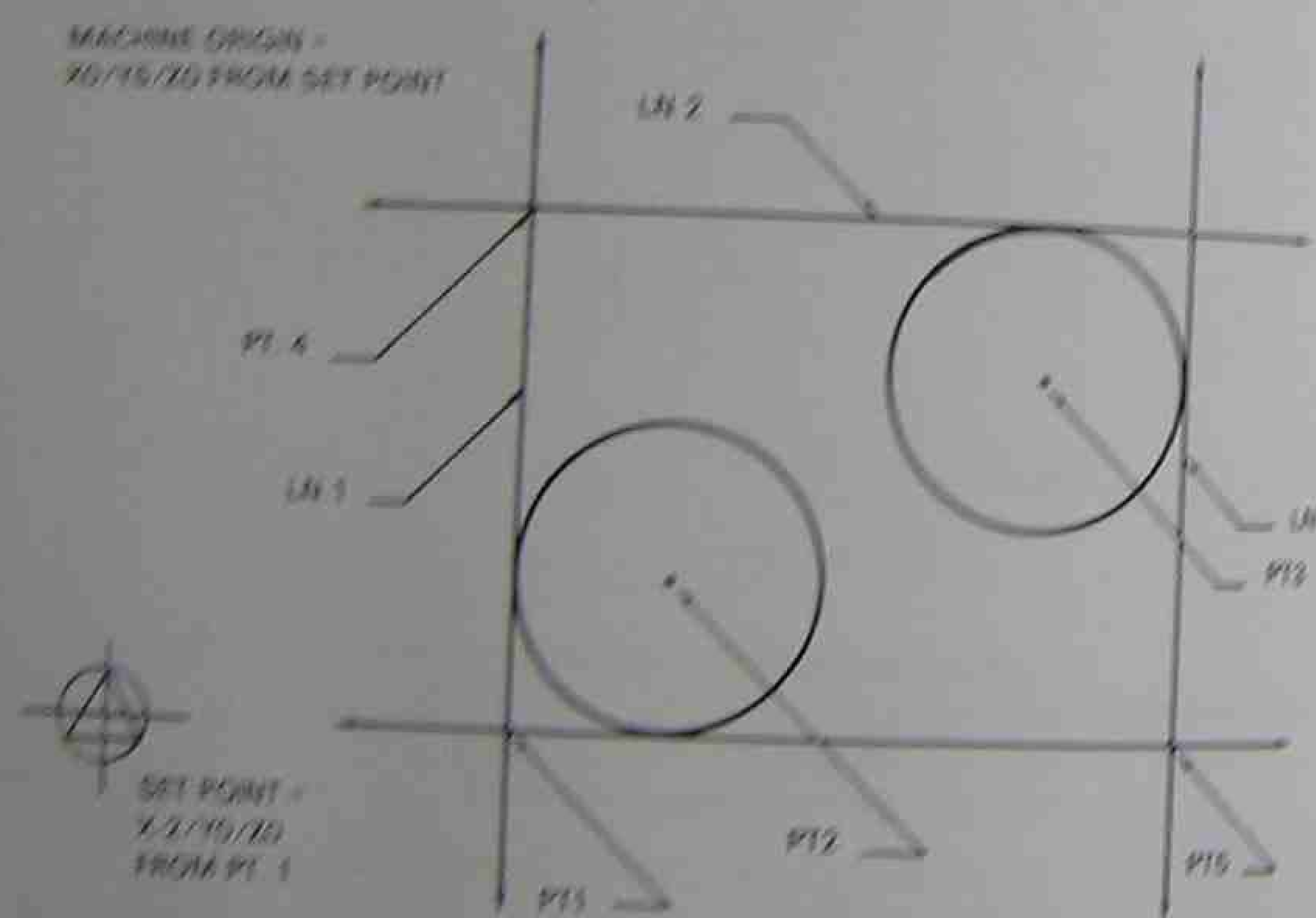


FIGURE 15–11
Geometry of part in Figure 15–10

To Define a Line.

Lines defined by two points, Figure 15–2(a):
DLNn,PTn,PTn
Example: DLN1,PT1,PT2
Meaning: Line 1 is the line passing through point 1 and point 2.

Lines defined by a point and parallel line, Figure 15–2(b):
DLNn,PTn,PARLNn
Example: DLN2,PT2,PARLN1
Meaning: Line 2 is the line passing through point 2 parallel to line 1.

Lines defined by a point and a perpendicular line, Figure 15–2(c):
DLNn,PTn,PERLNn
Example: DLN3,PT3,PERLN2
Meaning: Line 3 is the line passing through point 3 perpendicular to line 2.

Lines defined through a point and tangent to a circle, Figure 15–2(d):
DLNi,PTi,CIRi,MODIFIER
Example: DLN4,PT2,CIR1,YL
Meaning: Line 4 is the line passing through point 2, tangent to circle 1 on the Y large (right) side.

To Define a Circle.

Circle defined by the centerpoint and radius, Figure 15–3(a):
DCIRn,PTn,R
Example: DCIR1,PT1,1.0R
Meaning: Circle 1 is the circle with center at point 1 and a radius of 1.0 inches.

Circle defined by three points on the circumference, Figure 15–3(d):
DCIRn,PTn,PTn,PTn
Example: DCIR2,PT2,PT3,PT4
Meaning: Circle 2 is the circle with points 2, 3, and 4 on its circumference.

Auxiliary Statements in COMPACT II

Five auxiliary statements are used to initiate and set up the COMPACT II program.

The machine statement MACHIN is used first. MACHIN,UNIVER1 would signal the computer that a machine called UNIVER (Universal) 1 was being used. This tells the computer which postprocessor to use.

The next statement is called the identification statement. This statement identifies the program so that it can be cataloged and retrieved at a later date.

```
MACHIN/UNIVERSAL-1
$$
$$GEOMETRY
$$
PL1     =PLANE/XYPLAN,-1

SETPT   =POINT/-2,0,0
PT1     =POINT/0,0,-1
PT2     =POINT/.75,.75,-1
PT3     =POINT/2.75,1.75,-1
PT4     =POINT/0,2.5,-1
PT5     =POINT/3.25,0,-1

CIR1    =CIRCLE/CENTER,PT2,RADIUS,.75
CIR2    =CIRCLE/CENTER,PT3,RADIUS,.75

LN1     =LINE/PT1,PT4
LN2     =LINE/PT2,LEFT,TANTO,CIR2
LN3     =LINE/PT5,PARLEL,LN1
LN4     =LINE/PT1,PT5
$$
$$MOTION
$$
FROM/SETPT
LOADTL/1,ADJUST,1
CUTTER/.75
SPINDL/2000,CLW
FEDRAT/12,IPM
RAPID,GO/LN1,PL1,PAST,LN4
TLLFT,GOLFT/LN1,PAST,LN2
GORGT/LN2,TANTO,CIR2
GOFWD/CIR2,TANTO,LN3
GOFWD/LN3,PAST,LN4
GORGT/LN4,PAST,LN1
RAPID,GOTO/SP
FINI
```

FIGURE 15–12
APT program to mill the part in Figure 15–10

The word IDENT begins the statement. IDENT,EXAMPLE,PARTNO.1 would identify the program as example 1, part number 1.

The next statement is called the initialization statement (INIT). This statement tells the computer what modes will be used for input and output. INIT,INCH/IN,INCH/OUT tells the computer that all input and output dimensions are in inches.

The fourth statement in the program is the setup statement (SETUP). This statement identifies the machine origin, tool change location, tool travel limits, positioning system to be used, and any other special requirements that may be necessary for the particular machine.

The fifth statement is the base statement. The base statement establishes a work coordinate system. BASE 6XA,6YA,6ZA establishes the work coordi-

nate system 6.0 inches in X, 6.0 inches in Y, and 6.0 inches in Z from the machine origin. The A following the axes indicates machine absolute system (the machine origin). In geometry definition statements, the suffix B is used to indicate base coordinate system (the work coordinate system).

Other statements used in COMPACT II are: ATCHG to initiate an automatic tool change; CON to turn the coolant on; STK to identify the amount of stock to be left for finishing; FRM to assign a feedrate in inches per minute; IPR to assign a feedrate in inches per revolution.

Tool Motion Statements in COMPACT II

The tool motion statements used in COMPACT II are similar to those used in APT. The modifiers ON, TO, and PAST are used in identical fashion. Other statements are:

MOVE—To designate a move in rapid traverse.
CUT—To designate a move at feedrate.
ICON—To identify an inside contour.
OCON—To identify an outside contour.
S—To identify the starting location of a cut.
F—To identify the finish location of a cut.

In Figure 15–13 the geometry of a part is illustrated. The COMPACT II statements to move the tool from home to #1 and then around the part periphery are:

MOVE,TOLN1,PASTLN4
CUT,PARLN1,PASTLN2
CUT,PARLN2,TAN,CIR1
OCON,CIR1,CW,S(90),F(0)

S(90) means the starting point is at 90 degrees; F(0) means the finish point is at 0 degrees.

The Postprocessor

The postprocessor is a separate program that translates the part program into the codes necessary for the particular machine defined to the computer through the machine statement. *Postprocessor* is really a short term for post-processing program. Most machinery programmed using computer-aided programming languages uses RS-274 (word address) format. The postprocessor converts the APT or COMPACT II program (part geometry, auxiliary, and tool motion statements) to a word address format numerical control program. This program may then be punched into a tape using either RS-244 or RS-358 cod-

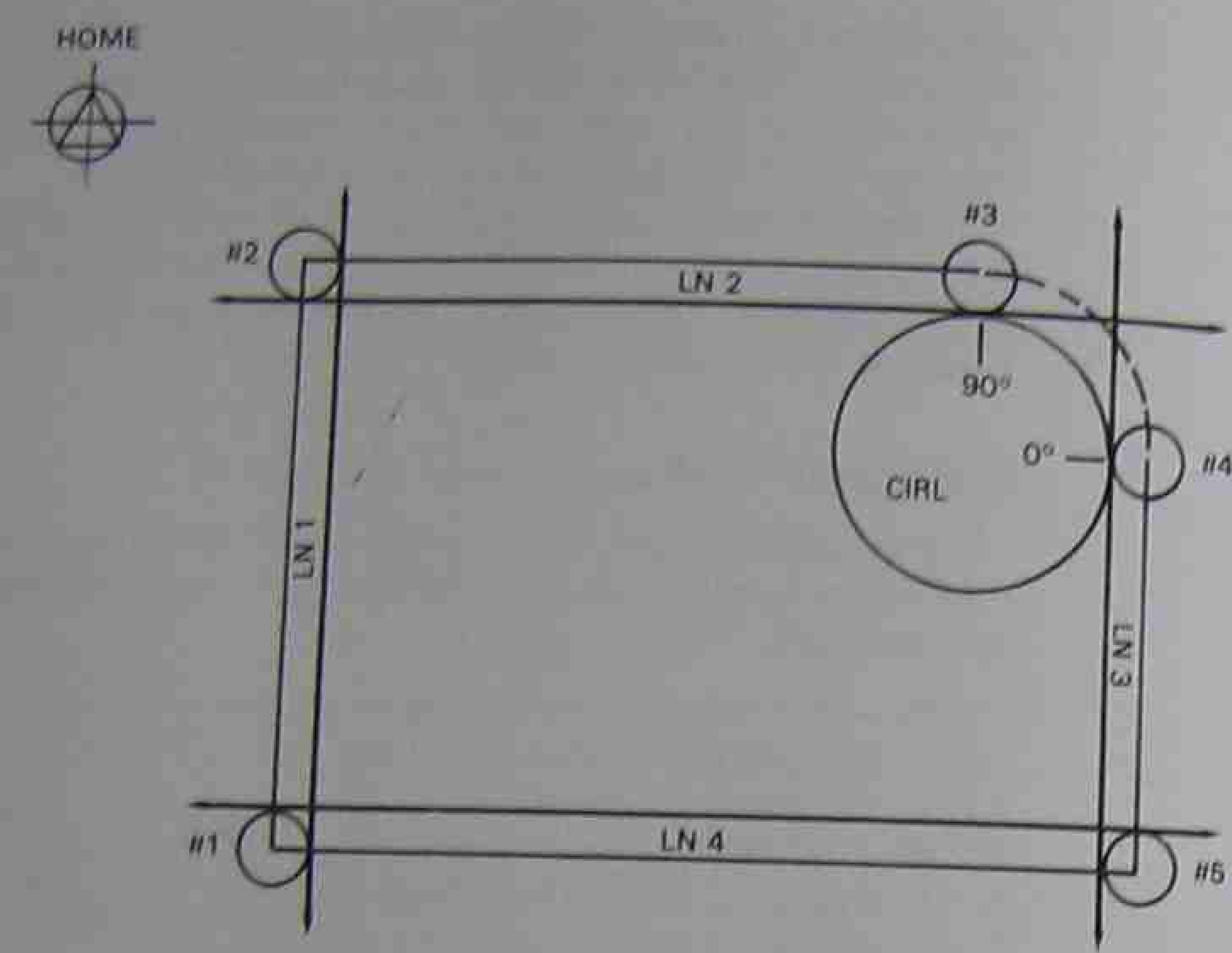


FIGURE 15–13

ing; or it may be transferred directly from the main computer to the machine's computer. The advantage of computer-aided programming languages over straight word address programming is the computer's ability to perform the calculations for very complex surfaces. Manual programming of parts requiring four and five axes is far more complicated than two- and three-axis programming. Figure 15–14 shows a part for which an APT program is written and postprocessed. Figure 15–15 shows the part geometry, and Figure 15–16 shows the APT program and the postprocessor output.

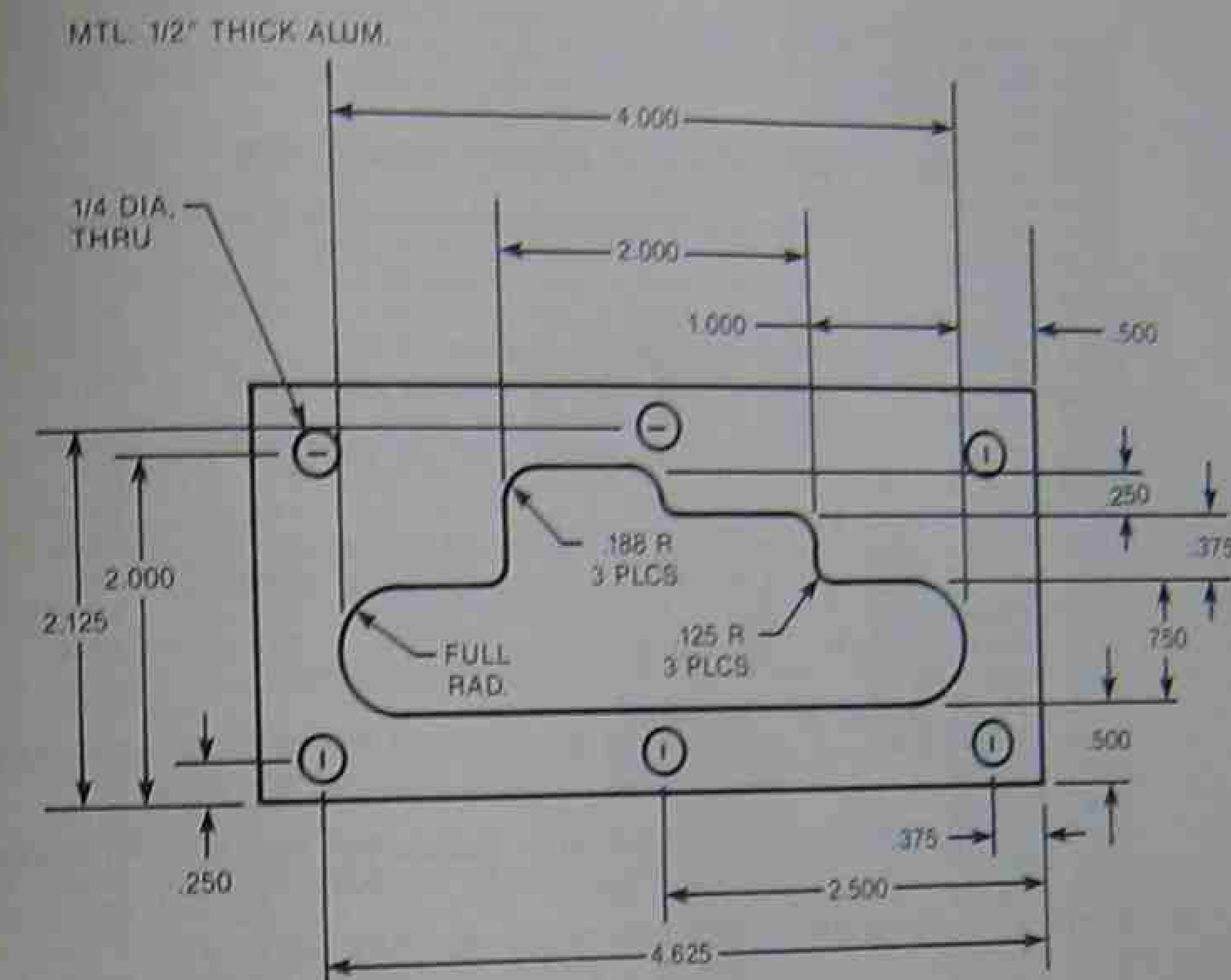


FIGURE 15–14
Part drawing

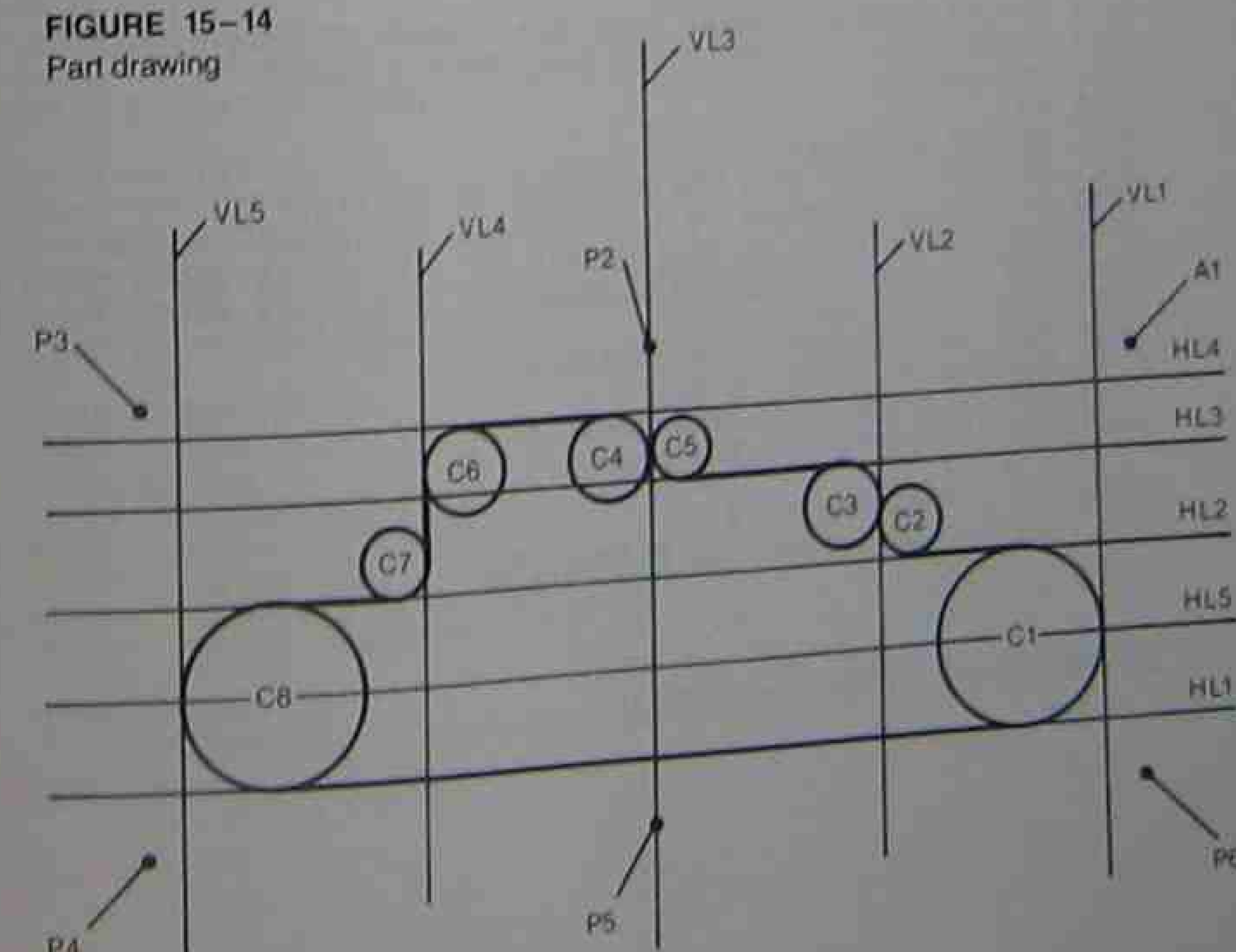


FIGURE 15–15
Part geometry

```
PARTNO APT MILL EXAMPLE
MACHIN/UNCMC1,01
MACHIN/PCPLOT,1
$$
$$MACRO TO PRINT A ROW OF ASTERIX ACROSS PAGE
ASTRX  =MACRO/
PPRINT ************************************************************
TERMAC
$$
CALL/ASTRX
PPRINT COORDINATE SYSTEM ORIGIN:
PPRINT X0 = LOWER RIGHT CORNER OF PART
PPRINT Y0 = LOWER RIGHT CORNER OF PART
PPRINT Z0 = 2.000 INCHES ABOVE TOP OF PART
CALL/ASTRX
PPRINT TOOL LIST
PPRINT TOOL 1:  3/8 DIA. 2 FLUTE CARBIDE END MILL
PPRINT TOOL 2:  1/4 DIA. 2 FLUTE CARBIDE END MILL
PPRINT TOOL 3:  NO. 4 X 90 DEG. C'DRILL
PPRINT TOOL 4:  1/4 DRILL
CALL/ASTRX
PPRINT MACHINING SEQUENCE
PPRINT TOOL 1:      SPEED 3800 RPM, FEEDRATE .003 IPR, CRO REGISTER D11
PPRINT              ROUGH MILL INSIDE CONTOUR OF PART.
PPRINT
PPRINT TOOL 2:      SPEED 4000 RPM, FEEDRATE .002 IPR, CRO REGISTER D21
PPRINT              FINISH MILL INSIDE CONTOUR OF PART.
PPRINT
PPRINT TOOL 3:      SPEED 3800 RPM, FEEDRATE .004 IPR
PPRINT              C'DRILL & C'SINK 1/4 HOLES TO .280 DIA. 6 PLCS.
PPRINT
PPRINT TOOL 4:      SPEED 3000 RPM, FEEDRATE .004 IPR
PPRINT              DRILL 1/4 DIA. THRU 6 PLCS.
CALL/ASTRX
$$
$$PART GEOMETRY
$$
PL1  =PLANE/XYPLAN,-2                   $$TOP OF PART
PL2  =PLANE/PARLEL,PL1,ZSMALL,.51       $$.01 BELOW BOTTOM OF PART
CLR  =PLANE/PARLEL,PL1,ZLARGE,.05       $$.05 ABOVE TOP OF PART
VL1  =LINE/YAXIS,-.5
VL2  =LINE/PARLEL,VL1,XSMALL,1
VL3  =LINE/YAXIS,-2.5
VL4  =LINE/PARLEL,VL2,XSMALL,2
VL5  =LINE/PARLEL,VL1,XSMALL,4

HL1  =LINE/XAXIS,.5
HL2  =LINE/PARLEL,HL1,YLARGE,.75
```

FIGURE 15-18 (Continues on page 380)


```
HL3  =LINE/PARLEL,HL2,YLARGE,.375
HL4  =LINE/PARLEL,HL3,YLARGE,.25
HL5  =LINE/PARLEL,HL1,YLARGE,.375

C1   =CIRCLE/XSMALL,VL1,YLARGE,HL1,YSMALL,HL2
C2   =CIRCLE/XLARGE,VL2,YLARGE,HL2,RADIUS,.125
C3   =CIRCLE/XSMALL,VL2,YSMALL,HL3,RADIUS,.188
C4   =CIRCLE/XSMALL,VL3,YSMALL,HL4,RADIUS,.188
C5   =CIRCLE/YLARGE,HL3,XLARGE,OUT,C4,RADIUS,.125
C6   =CIRCLE/XLARGE,VL4,YSMALL,HL4,RADIUS,.188
C7   =CIRCLE/XSMALL,VL4,YLARGE,HL2,RADIUS,.125
C8   =CIRCLE/YLARGE,HL1,XLARGE,VL5,YSMALL,HL2

ZSURF/PL1
        P1   =POINT/-.375,2
        P2   =POINT/-2.5,2.125
        P3   =POINT/-4.625,2
        P4   =POINT/-4.625,.25
        P5   =POINT/-2.5,.25
        P6   =POINT/-.375,.25
        PAT1 =PATERN/P1,P2,P3,P4,P5,P6
$$
$$MOTION STATEMENTS
$$
PPLOT/1
FROM/0,4,0
CALL/ASTRX
PPRINT 3/8 END MILL - USES CRO REGISTER D11
PPRINT ROUGH MILL INSIDE CONTOUR - LEAVE .01 STK. TO FINISH.
LOADTL/1,ADJUST,1
CUTTER/.375
SPINDL/3800,CLW
COOLNT/ON
FEDRAT/.003,IPR

PPRINT
PPRINT POSITION TO START XY AND FEED TO DEPTH
RAPID,GO/ON,VL2,CLR,ON,HL5
THICK/0,.01,.01
GO/ON,VL2,PL2
CUTCOM/LEFT,11

PPRINT
PPRINT MILL .750 WIDE SLOT
GO/HL2
    GOLFT/HL2,TANTO,C8
    GOFWD/C8,TANTO,HL1
    GOFWD/HL1,TANTO,C1
    GOFWD/C1,TANTO,HL2
    GOFWD/HL2,PAST,VL2

PPRINT
PPRINT MILL BALANCE OF SLOT
GO/HL3
```

```
0097      GOLFT/HL3,PAST,VL3
0098      GORGT/VL3,HL4
0099      GOLFT/HL4,VL4
0100      GOLFT/VL4,ON,HL5

0101 PPRINT
0102 PPRINT RETRACT SPINDLE, CANCEL CUTTER COMP, CANCEL TOOL OFFSET
0103 RETRCT
0104 CUTCOM/OFF
0105 COOLNT/OFF
0106 RESET
0107 $$
0108 PPLOT/2
0109 CALL/ASTRX
0110 PPRINT 1/4 END MILL - USES CRO REGISTER D21
0111 PPRINT FINISH MILL SLOT
0112 LOADTL/2,ADJUST,2
0113 CUTTER/.250
0114 SPINDL/4000,CLW
0115 COOLNT/ON
0116 FEDRAT/.002,IPR

0117 PPRINT
0118 PPRINT POSTITON TO CENTER OF C1 AND FEED TO Z DEPTH
0119 THICK/0
0120 RAPID,GO/ON,(LINE/PARLEL,VL1,XSMALL,.375),CLR,ON,HL5
0121 NOPS,GO/PL2
0122 AUTOPS

0123 PPRINT
0124 PPRINT FINISH MILL CONTOUR OF I.D.
0125 CUTCOM/LEFT,21
0126 GO/HL2
0127      GOLFT/HL2,TANTO,C2
0128      GOFWD/C2,TANTO,VL2
0129      GOFWD/VL2,TANTO,C3
0130      GOFWD/C3,TANTO,HL3
0131      GOFWD/HL3,TANTO,C5
0132      GOFWD/C5,TANTO,C4
0133      GOFWD/C4,TANTO,HL4
0134      GOFWD/HL4,TANTO,C6
0135      GOFWD/C6,TANTO,VL4
0136      GOFWD/VL4,TANTO,C7
0137      GOFWD/C7,TANTO,HL2
0138      GOFWD/HL2,TANTO,C8
0139      GOFWD/C8,TANTO,HL1
0140      GOFWD/HL1,TANTO,C1
0141      GOFWD/C1,TANTO,HL2
0142      GOFWD/HL2,PAST,VL2

0143 PPRINT
0144 PPRINT RETRACT SPINDLE, CANCEL CUTTER COMP, CANCEL TOOL OFFSET
0145 RETRCT
0146 CUTCOM/OFF
```



```
0147 COOLNT/OFF
0148 RESET
0149 $$
0150 PPLOT/3
0151 CALL/ASTRX
0152 PPRINT NO. 4 C'DRILL - C'DRILL 6 PLCS. TO .280 DIA.
0153 LOADTL/3,ADJUST,3
0154 SPINDL/3500,CLW
0155 COOLNT/ON
0156 PREFUN/99,NEXT
0157 CYCLE/DRILL,.183,.004,IPR,.1
0158          GOTO/PAT1
0159 CYCLE/OFF
0160 COOLNT/OFF
0161 RESET
0162 $$
0163 DRAFT/OFF
0164 CALL/ASTRX
0165 PPRINT 1/4 DRILL - PECK DRILL .250 THRU 6 PLCS.
0166 LOADTL/4,ADJUST,4
0167 SPINDL/3000,CLW
0168 COOLNT/ON
0169 PREFUN/99,NEXT
0170 CYCLE/DEEP,.6,.003,IPR,.1,INCR,.25
0171          GOTO/PAT1
0172 CYCLE/OFF
0173 COOLNT/OFF
0174 RESET
0175 FINI


  NO DIAGNOSTICS ELICITED DURING TRANSLATION PHASE
 185 N/C SOURCE RECORDS (SYSIN)


       SECTION 1 ELAPSED CPU TIME IN MIN/SEC IS 0000/00.4433
       SECTION 2 ELAPSED CPU TIME IN MIN/SEC IS 0000/00.1866
```

```
UNCM01 03.000.I370 MACHIN/UNCM01, 1   DATE 88.319      PAGE   1
APT MILL EXAMPLE                                                        (INCH)
INPUT  CLREC N3G2X34Y34R34Z34Q34I34J34K34A43P4F32S4T2D2H2M2
    7      9  APT MILL EXAMPLE
    7      9 $
    7      9 LEADER/   20.0
    7      9 $
    7      9 N001 G90 G00 G70 G80 G40$
    7      9  **************************************************
   11     11  COORDINATE SYSTEM ORIGIN:
   12     13  X0 = LOWER RIGHT CORNER OF PART
   13     15  Y0 = LOWER RIGHT CORNER OF PART
   14     17  Z0 = 2.000 INCHES ABOVE TOP OF PART
    7     20  **************************************************
   16     22  TOOL LIST
   17     24  TOOL 1:  3/8 DIA. 2 FLUTE CARBIDE END MILL
   18     26  TOOL 2:  1/4 DIA. 2 FLUTE CARBIDE END MILL
   19     28  TOOL 3:  NO. 4 X 90 DEG. C'DRILL
   20     30  TOOL 4:  1/4 DRILL
    7     33  **************************************************
   22     35  MACHINING SEQUENCE
   23     37  TOOL 1:  SPEED 3800 RPM, FEEDRATE .003 IPR, CRO REGISTER D11
   24     39           ROUGH MILL INDSIDE CONTOUR OF PART.
   25     41
   26     43  TOOL 2:  SPEED 4000 RPM, FEEDRATE .002 IPR, CRO REGISTER D21
   27     45           FINISH MILL INSIDE CONTOUR OF PART.
   28     47
   29     49  TOOL 3:  SPEED 3500 RPM, FEEDRATE .004 IPR
   30     51           C'DRILL & C'SINL 1/4 HOLES TO .260 DIA. 6 PLCS.
   31     53
   32     55  TOOL 4:  SPEED 3000 RPM, FEEDRATE .004 IPR
   33     57           DRILL 1/4 DIA. THRU 6 PLCS.
    7     60  **************************************************
   71     64 FROM/XYZ =      0.0000     4.0000     0.0000
    7     67  **************************************************
   73     69  3/8 END MILL - USES CRO REGISTER D11
   74     71  ROUGH MILL INSIDE CONTOUR - LEAVE .01 STK. TO FINISH.
   75     73 TOOL TIME =     0.04
   75     73 $
   75     73 N002 G90 G00 X.0000 Y4.0000 T01$
   75     73 N003 S0500 M03$
   75     73 N004 G44 Z.0000 H01$
   77     77 N005 S3800$
   78     79 N006 M08$
   80     83
   81     85  POSITION TO START XY AND FEED TO DEPTH
   82     88 N007 G00 X-1.5000 Y.8750$
   82     88 N008 Z-1.9500$
                  TAPE      TIME     WARNING
          PAGE    6.38      0.15        0
          TOTAL   6.38      0.15        0
```



```
UNCM01 03.000.I370 MACHIN/UNCM01, 1   DATE 88.319      PAGE   2
APT MILL EXAMPLE
INPUT  CLREC N3G2X34Y34R34Z34Q34I34J34K34A43P4F32S4T2D2H2M2          (INCH)
   84     92 N009 G01 Z-2.5100 F11.40$
   85     94 N010 G17$
   86     96
   87     98  MILL .750 WIDE SLOT
   88    100 N011 G41 X-1.5000 Y.8760 J1.0000 D11$
   88    100 N012 G01 Y1.0525$
   89    102 N013 X-4.1250$
   91    107 N014 G03 X-4.1250 Y.6975 I-4.1250 J.8750$
   91    107 N015 G01 X-.8750$
   93    112 N016 G03 X-.8750 Y1.0525 I-.8750 J.8750$
   93    112 N017 G01 X-1.6975$
   94    114
   95    116  MILL BALANCE OF SLOT
   96    118 N018 Y1.4275$
   97    120 N019 X-2.6975$
   98    122 N020 Y1.6775$
   99    124 N021 X-3.3025$
  100    126 N022 Y.8750$
  101    128
  102    130  RETRACT SPINDLE, CANCEL CUTTER COMP, CANCEL TOOL OFFSET
  103    132 N023 G00 Z.0000$
  104    134 N024 G40$
  105    136 N025 M09$
  106    138 N026 G00 Z.0000$
  106    138 N027 G49$
  106    138 N028 M01$
    7    143  **************************************************
  110    145  1/4 END MILL - USES CRO REGISTER D21
  111    147  FINISH MILL SLOT
  112    149 TOOL TIME =    1.09
  112    149 $
  112    149 N029 G90 G00 X-3.3025 Y.8750 T02$
  112    149 N030 S0500 M03$
  112    149 N031 G44 Z.0000 H02$
  114    153 N032 S4000$
  115    155 N033 M08$
  117    159
  118    161  POSTITON TO CENTER OF C1 AND FEED TO Z DRPTH
  120    166 N034 G00 X-.8750$
  120    166 N035 Z-1.9500$
  121    168 N036 G01 Z-2.5100 F8.00$
  123    171
  124    173  FINISH MILL CONTOUR OF I.D.
  126    177 N037 G41 X-.8750 Y.8760 J1.0000 D21$
  126    177 N038 G01 Y1.1250$
                  TAPE      TIME     WARNING
          PAGE    3.70      1.21        0
          TOTAL  10.08      1.36        0
```

```
UNCM01 03.000.I370 MACHIN/UNCM01, 1    DATE 88.319      PAGE   3
APT MILL EXAMPLE                                                 (INCH)
INPUT  CLREC N3G2X34Y34R34Z34Q34I34J34K34A43P4F32S4T2D2H2M2
  127    179 N039 X-1.3750$
  129    184 N040 G02 X-1.6250 Y1.3750 I-1.3750 J1.3750$
  129    184 N041 G01 Y1.4370$
  131    189 N042 G03 X-1.6880 Y1.5000 I-1.6880 J1.4370$
  131    189 N043 G01 X-2.3814$
  133    194 N044 G02 X-2.6263 Y1.6997 I-2.3814 J1.7500$
  134    197 N045 G03 X-2.6880 Y1.7500 I-2.6880 J1.6870$
  134    197 N046 G01 X-3.3120$
  136    202 N047 G03 X-3.3750 Y1.6870 I-3.3120 J1.6870$
  136    202 N048 G01 Y1.3750$
  136    207 N049 G02 X-3.6250 Y1.1250 I-3.6250 J1.3750$
  138    207 N050 G01 X-4.1250$
  140    212 N051 G03 X-4.1250 Y.6250 I-4.1250 J.8750$
  140    212 N052 G01 X-.8750$
  142    217 N053 G03 X-.8750 Y1.1250 I-.8750 J.8750$
  142    217 N054 G01 X-1.6250$
  143    219
  144    221  RETRACT SPINDLE, CANCEL CUTTER COMP, CANCEL TOOL OFFSET
  145    223 N055 G00 Z.0000$
  146    225 N056 G40$
  147    227 N057 M09$
  148    229 N058 G00 Z.0000$
  148    229 N059 G49$
  148    229 N060 M01$
    7    234  ****************************************************
  152    236  NO. 4 C'DRILL - C'DRILL 6 PLCS. TO .260 DIA.
  153    238 TOOL TIME =     1.38
  153    238 $
  153    238 N061 G90 G00 X-1.6250 Y1.1250 T03$
  153    238 N062 S0500 M03$
  153    238 N063 G44 Z.0000 H03$
  154    240 N064 S3500$
  155    242 N065 M08$
  158    248 N066 G81 G99 X-.3750 Y2.0000 R-1.9000 Z-2.1830 F14.00$
  158    249 N067 X-2.5000 Y2.1250$
  158    250 N068 X-4.6250 Y2.0000$
  158    251 N069 Y.2500$
  158    252 N070 X-2.5000$
  158    253 N071 X-.3750$
  159    255 N072 G80$
  160    257 N073 M09$
  161    259 N074 G00 Z.0000$
  161    259 N075 G49$
  161    259 N076 M01$
    7    264  ****************************************************
                   TAPE      TIME      WARNING
         PAGE      5.63      1.37        0
         TOTAL    15.72      2.73        0
```



```
UNCM01 03.000.I370 MACHIN/UNCM01, 1    DATE 88.319     PAGE   4
APT MILL EXAMPLE                                                 (INCH)
INPUT  CLREC N3G2X34Y34R34Z34Q34I34J34K34A43P4F32S4T2D2H2M2
  165    266   1/4 DRILL - PECK DRILL .250 THRU 6 PLCS.
  166    268 TOOL TIME =     0.21
  166    268 $
  166    268 N077 G90 G00 X-.3750 Y.2500 T04$
  166    268 N078 S0500 M03$
  166    268 N079 G44 Z.0000 H04$
  167    270 N080 S3000$
  168    272 N081 M08$
  171    278 N082 G83 G99 X-.3750 Y2.0000 R-1.9000 Z-2.6000 Q.2500 F9.00$
  171    279 N083 X-2.5000 Y2.1250$
  171    280 N084 X-4.6250 Y2.0000$
  171    281 N085 Y.2500$
  171    282 N086 X-2.5000$
  171    283 N087 X-.3750$
  172    285 N088 G80$
  173    287 N089 M09$
  174    289 N090 G00 Z.0000$
  174    289 N091 G49$
  174    289 N092 M01$
  175    291 TOOL TIME =     0.56
  175    291 N093 G00 Y4.0000 T15$
  175    291 N094 M30$
  175    291 =$
                   TAPE      TIME      WARNING
         PAGE      2.13      0.56        0
         TOTAL    17.85      3.29        0
```

```
UNCM01 03.000.I370 MACHIN/UNCM01, 1    DATE 88.319     PAGE   5
APT MILL EXAMPLE                                                 (INCH)
INPUT  CLREC N3G2X34Y34R34Z34Q34I34J34K34A43P4F32S4T2D2H2M2
            TOOL SEQUENCE
INPUT  CLREC        TOOL   OFFSET   LENGTH Z
  75     73           1      1       0.0000
 112    149           2      2       0.0000
 153    238           3      3       0.0000
 166    268           4      4       0.0000
       TOTAL PART PROGRAM CPU TIME IN MIN/SEC IS 0000/02.3966
              **** END OF APT PROCESSING ****
```

FIGURE 15–16
APT program and postprocessor output for part shown in Figure 15–14

COMPUTER GRAPHICS PROGRAMMING

The newest form of NC programming is called CAM. CAM stands for computer aided manufacturing. When using a CAM system, the programmer either calls up an existing part drawing, or defines the part geometry to the computer. Next, the cutter path is drawn around the part and the necessary information on cut direction, tool, speeds, and feeds are input. This information is then converted into either an APT file, or a cutter centerline data file by a postprocessor. This data is then fed through a secondary postprocessor, which produces the necessary NC code for a given machine.

There are a number of different CAM systems on the market. The following is a brief explanation of several different types.

Digitizing Systems

Digitizing systems use an existing part drawing to obtain the geometry information. The drawing used must be drawn to a true scale of the finished part. The scale drawing is fed into a digitizer which is connected to the CAM system computer. A *digitizer* is a device consisting of a table with a probe or other sensor attached. The sensing device is passed over the drawing, converting drawing lines in the necessary mathematical information into electronic form which the computer needs to recreate the drawing. The cutter path is then defined by the programmer and the cutter, speed, and feed information input. The result is then postprocessed into the necessary tape code for the CNC machine.

Digitizing is one of the simplest CAM methods, but it is also the least accurate. The accuracy of the part geometry is dependent upon the accuracy of the scaled drawing which was digitized. Fortunately this shortcoming is minimized by the fact that drawings 30 times size or larger can be used.

Scanning Systems

Scanning systems use the part itself rather than a part drawing to obtain the geometric database to machine the part. Scanning is used when complex curves are to be machined which are difficult to draw and which do not fit a true mathematical model. Automobile bodies are an example of such curves. A *scanner* is a probing device connected to the CAM system computer which is passed over a model of the part. The probe feeds information concerning the part geometry into the computer which then calculates the points necessary to define the part shape. The cutter path, tool data, speeds, and feeds are then input. The information is then fed to the postprocessor which will convert the information into the necessary tape coding.

CAM Systems

Although the two methods mentioned above are referred to as CAM systems, CAM is also the term used to describe computer graphics programming. CAM systems can be run on a mainframe computer, or on a PC. Mainframe systems are found in larger plants primarily involved with four- or five-axis programming. Microcomputer-based CAM systems function well for three-axis. New systems can handle four-axis programming as well. Microcomputer systems are an economical choice for many small to midsized shops. Figure 15–17 illustrates a PC-based CAM system running on an APPLE Macintosh. Figure 15–18 depicts another microbased system.

In CAM (graphics) programming, the programmer defines the part geometry to the computer using one or more of several input devices. These devices may be the keyboard, a mouse, a digitizer, or a light pen. After the part geometry is defined, the cutter path is drawn around the part. Information on the cut direction, tool data, speeds, and feeds are then input. This information is then translated by a series of postprocessors into the necessary NC code. Figure 15–19 shows a cutter and tool path generated on a mainframe-based CAM system.

FIGURE 15–17
Mac EZ-CAM II *(Photo courtesy of Bridgeport Machine Inc.)*

FIGURE 15–18
EZ-CAM II *(Photo courtesy of Bridgeport Machine Inc.)*

Once programmed and postprocessed, the cutter path can be plotted out on paper as a preliminary prove-out of the program. Figure 15–20 shows one type of plotter. The program plot can help the programmer spot errors that have been overlooked when the cutter path was on the computer monitor screen.

There are many types of CAM systems on the market. One need only glance at the pages of an industry trade magazine to appreciate the number that exist.

CAD/CAM Programming

CAD/CAM programming is one of the more sophisticated CAM programming systems. In a CAD/CAM system, the part geometry is obtained from the CAD (computer aided design) drawing itself. The cutter path, tool, speeds, and feeds are then defined and the program postprocessed into the NC tape code. There are two basic types of CAD/CAM systems: stand alone and modular.

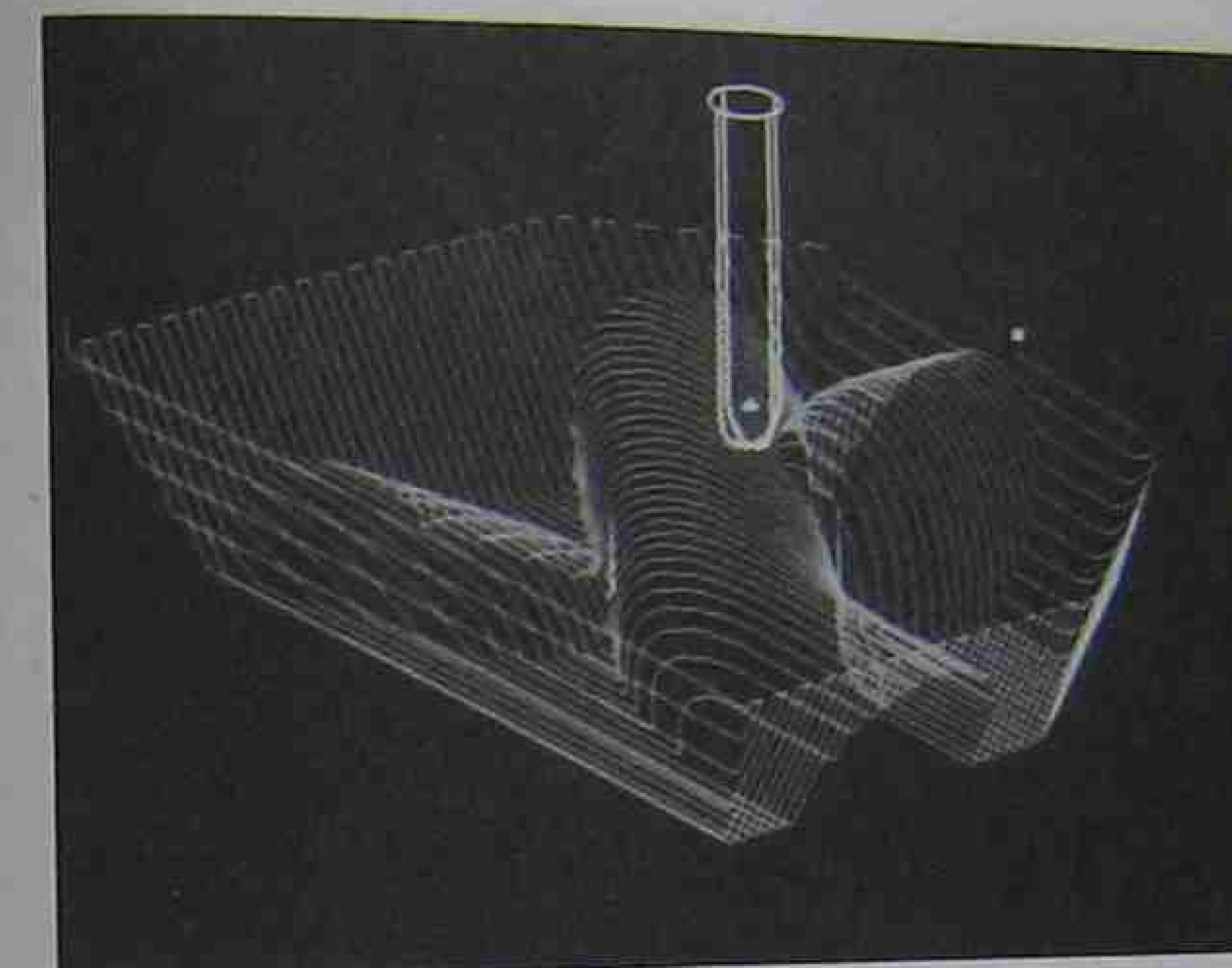


FIGURE 15–19
Cutter and tool path generated on a mainframe-based CAM system
(Photo courtesy of Battelle Inc.)

Stand alone systems contain both the CAD and the CAM modules in one system, furnished by one supplier. Generally, these systems are intended to function only with the supplied modules and will not link with other vendors products.

Modular systems consist of a CAD package made by one manufacturer, and a CAM package by another. The CAM packages are designed to link to the most popular CAD packages such as AutoCad, Micro CADAM, and GEOSPOT.

CIM

CAD/CAM programming coupled with DNC systems for program delivery to the individual machines form the fundamental building block for CIM (computer integrated manufacturing). CIM is not a true reality at the time of this publication, but is the wave of the future. In a CIM system, the entire manufacturing process is done with the aid of computers. Product design, manufacturing engineering, production control, quality assurance, procurement, accounting, and management functions would all be linked together in a shared database. The goal of all this is to eliminate paperwork, eliminate duplication of effort, and reduce overall costs while improving part quality and delivery schedules. Much is needed in the way of standardization to achieve a CIM environment, but much has already been accomplished.

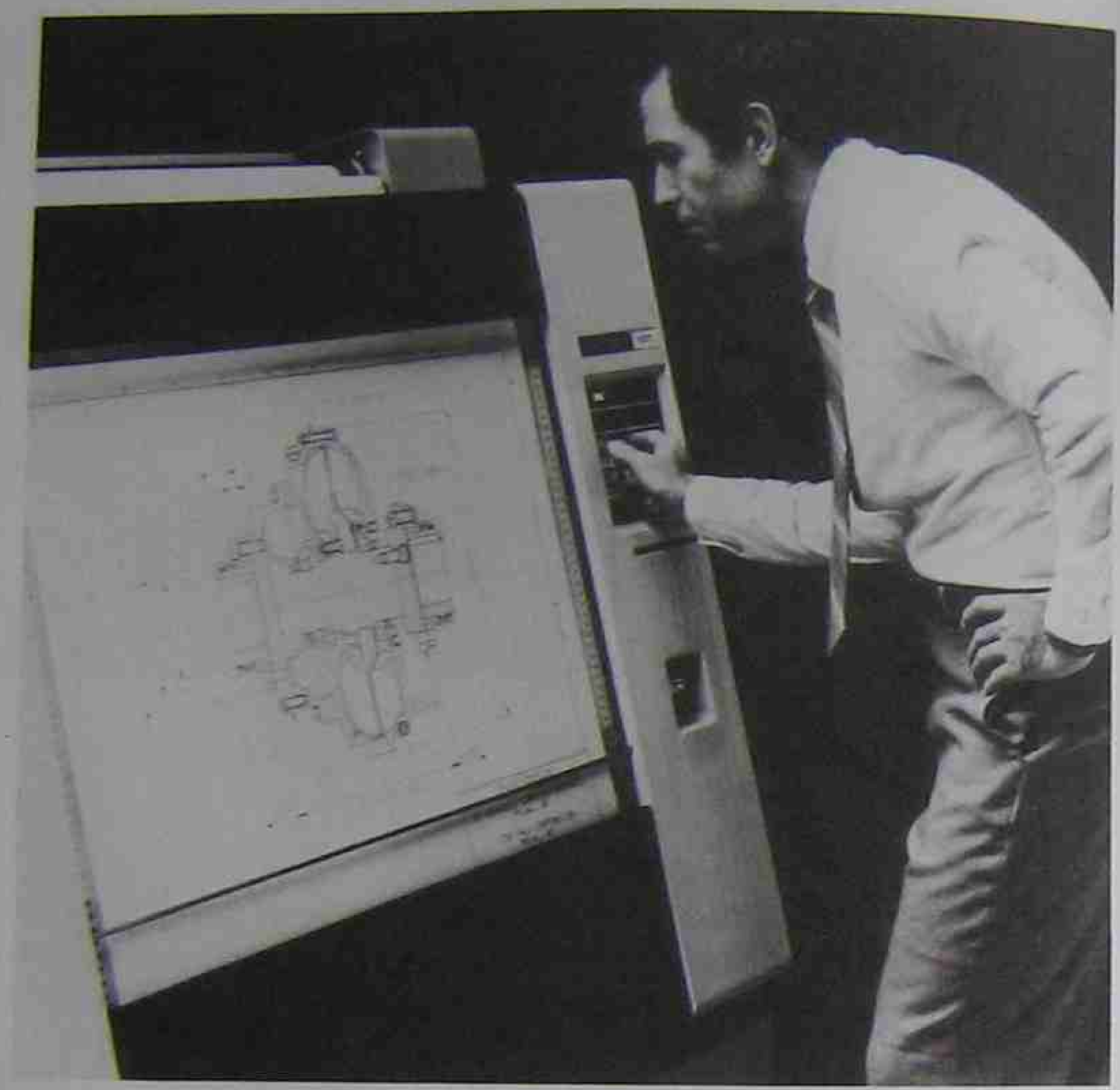


FIGURE 15–20
Plotter *(Photo courtesy of Calcomp Inc.)*

For a company to be competitive in the coming years, a move toward computer integration will become a necessity. It is important therefore, for students desiring to make a career in the manufacturing fields to learn and improve upon their computer skills.

SUMMARY

The important concepts presented in this chapter are:

- Computers are used three basic ways in numerical control programming: in offline programming, computer-aided programming, and computer graphics programming.
- Offline programming terminals allow the NC program to be written away from the machine. They do not assist the programmer in any way other than eliminating the need to enter the program at the machine's computer console.
- Computer-aided programming permits the programming of parts for many machines using one programming language. The computer handles the necessary mathematical calculations for the cutter path. A postprocessor then translates this information into codes for a particular machine.
- Three types of statements are used in a computer-aided program: part geometry statements, auxiliary statements, and tool motion statements.
- There are a number of computer-aided programming languages. The one a company uses depends upon the parts it produces and the computers available.
- Graphics programming called CAM programming is being used to simplify the part programming process.
- There are four basic types of CAM systems: digitizing, scanning, CAM, and CAD/CAM.
- There are two basic types of CAD/CAM systems: stand alone and modular.

REVIEW QUESTIONS

1. What three ways are computers used in numerical control programming?
2. What is offline programming?
3. What does computer-aided programming allow the programmer to do?
4. What three types of statements are used in computer-aided programs?
5. What is CAM programming?
6. What are four types of CAM systems?
7. What is the difference between a scanning and a digitizing system?
8. What is the difference between a CAM and a CAD/CAM system?

CHAPTER 16

The Future of Numerical Control

OBJECTIVES Upon completion of this chapter, you will be able to:

- Explain why the use of CNC will increase in prototype and small lot job shops.
- Describe a flexible machining system.
- Describe a machining cell.
- Describe the responsibilities of the NC electronics technician, machine operator/set-up operator, and part programmer.

Numerical control will play an increasingly important role in manufacturing in the coming years. CNC is already being applied to machine tools, punch presses, sheet metal brakes, electrical discharge machines, welding machinery, and inspection equipment. Smaller sized production shops, prototype operations, and large manufacturing concerns will all benefit from recent and continuing developments in numerical control, robotics, and computer technology.

NC IN PROTOTYPE AND JOB SHOPS

The ongoing development of less expensive numerical control systems will offer increasing options to companies that today cannot justify a numerical control system. The lower cost of acquiring machining and turning centers, coupled with the ease of programming and other features of the newest generation of CNC controllers, will result in the adoption of CNC machinery by more and more small job shops. Competition from foreign sources is forcing all companies to look for ways to improve quality while making the changes in design that market conditions so often require. CNC machinery can fulfill both requirements: (1) The repeatability of CNC can improve the overall quality of parts produced; and (2) since CNC uses software programs to produce part shapes, what would have been major retooling becomes the editing and revising of the part program.

Recently, a new type of DNC retrofit system was introduced specifically geared to prototype shop requirements. In this system a microcomputer such as the popular Apple II is used as the controller. The executive program for the system is software, not firmware, requiring little or no modification of the computer. This system is a surprisingly low cost way for a shop to acquire a DNC system. It is not designed for the demands of a manufacturing operation but will handle the one-of-a-kind parts made in prototype, die, or moldmaking shops. Designed for used on vertical mills, it is a three-axis contouring system capable of circular, helical, and linear interpolation.

A problem common to all companies is the shortage of skilled machinists. In smaller companies, the shortage of general machinists, tool and die makers, and mold makers is most acute. In coming years the shortage of skilled prototype machinists and instrument makers is likely to be felt by scientific and research organizations that have their own prototype shops. In addition, increasingly complex part geometries are being required for new technology applications. CNC offers solutions to all these problems.

CNC IN MANUFACTURING

The most exciting developments in NC applications are taking place in large scale manufacturing. In many industries, computer integration of the entire manufacturing process is believed to be possible in the coming decades. The computer capability for *computer integrated manufacturing (CIM)* already exists, but software bases and computer standards to allow networking of design, manufacturing, purchasing, inventory, and marketing functions must be developed and refined. In a CIM system, these various functions are interconnected, using the instant access to information that the computer allows, to eliminate duplication of effort, reduce inventories, reduce part handling, and provide a higher percentage of chip-making time. Although not yet a reality, one of the major building blocks in a CIM system is currently being produced and used by some industries—the flexible machining system (FMS).

Flexible Machining System

A *flexible machining system (FMS)* is a system of CNC machines, robots, and part transfer vehicles that can take a part from raw stock or casting and perform all necessary machining, part handling, and inspection operations to make a finished part or assembly. It is an entire unmanned, software-based, manufacturing/assembly line. An FMS consists of four major components: the CNC machines, coordinate measuring machines, part handling and assembly robots, and part/tool transfer vehicles. Figure 16–1 illustrates a small flexible

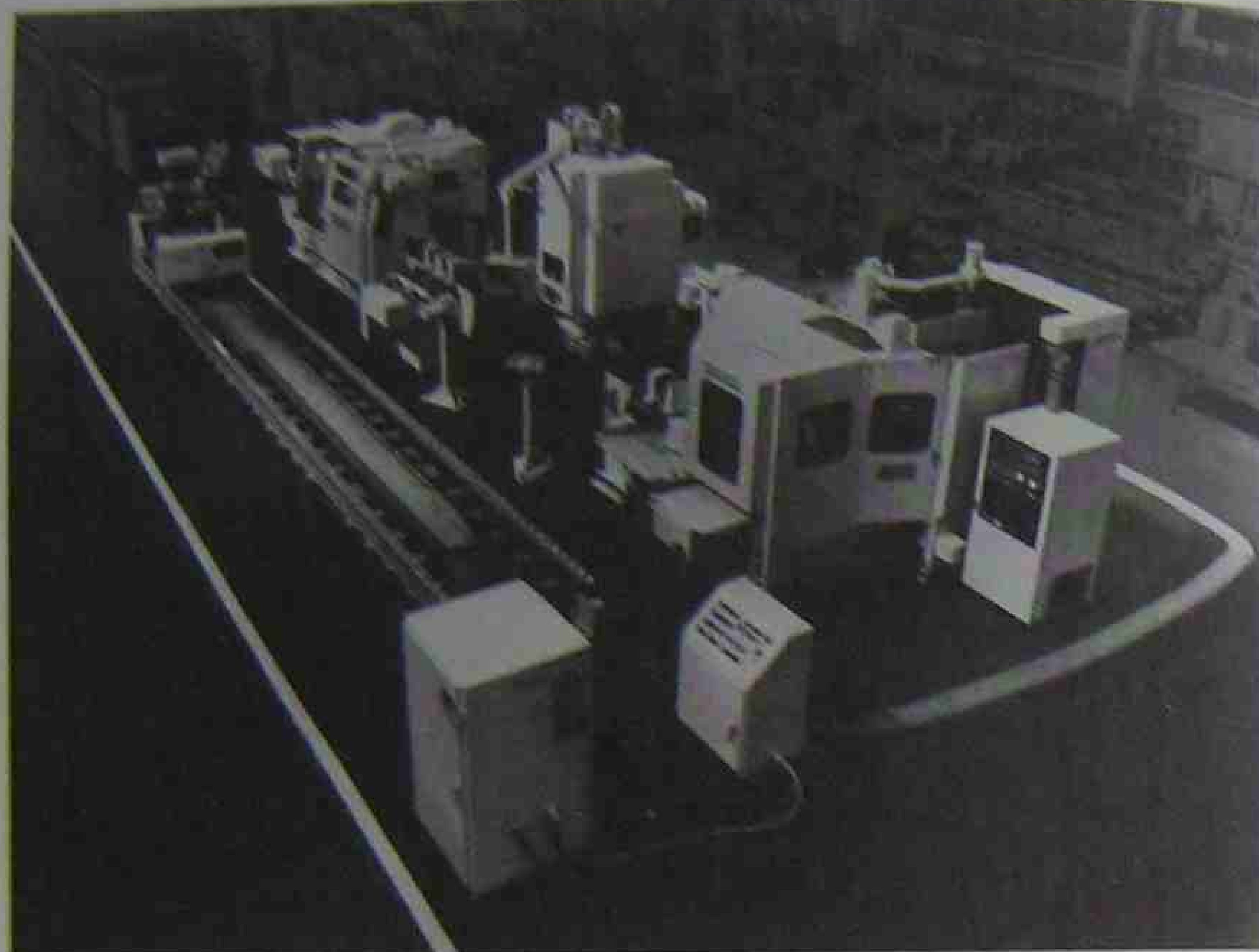


FIGURE 16–1 *(Photo courtesy of Cincinnati Milacron)*

machining system. This system employs a turning center, a horizontal machining center, and a vertical machining center. A single track-guided robot is used as both a load/unload robot and a transfer vehicle.

The main element in an FMS is the CNC machining or turning center. The automatic tool changing capability of these machines allows them to run untended, given the proper support system. Tool monitoring systems built into the CNC machine are used to detect and replace worn tools. The major obstacles in an FMS are not the machining centers but the support systems for the machines, such as part load/unload and part transfer.

Inspection in an FMS is accomplished through the use of coordinate measuring machines. These operate much like CNC machinery in that they are programmed to move to different positions on a workpiece. Instead of using a rotating spindle and a cutting tool, a coordinate measuring machine is equipped with electronic gaging probes which measure features on a workpiece. The results of the gaging are compared to acceptable limits programmed into the machine.

Robots are frequently used in an FMS to load and unload parts from the machines. Since robots are programmed pieces of equipment that lack the ability to make judgments, special workholding fixtures are employed on the transfer vehicles to orient the workpiece so that the robot can handle it correctly. Specially designed machine fixtures and clamping mechanisms are employed to ensure correct placement and clamping of the part on the machine. All part handling must be accomplished in a specific orderly fashion, with coordination of the part transfer vehicle, the robot, and the CNC machines. Future robots will probably employ some type of artificial intelligence which will enable them to make limited judgments as to workpiece orientation and take the necessary corrective actions.

The third critical component of an FMS is the tool and workpiece transfer vehicles. These vehicles shuttle workpieces from machine to machine. They also shuttle tool magazines to and from the machinery to maintain an adequate supply of sharp cutting tools at each CNC machine. Transfer vehicles employed in current flexible manufacturing systems are of four major types: automatic guided vehicles (AGV), wire guided vehicles, air cushion vehicles, and hardware guided vehicles.

Automatic guided vehicles rely on onboard sensors and/or a program to determine the path they take. There is no hardware connecting them to the system. An advantage of AGVs is that they can be reprogrammed to take different routes, eliminating the need to run tracks or wires for each route change. The corresponding disadvantage of AGVs is that they are the most difficult of the part delivery vehicles to make function, because of the lack of hardware connection.

A *wire guided vehicle* uses a wire buried in the floor to define its path. A sensor on the vehicle detects the location of the wire. A major advantage of wire guided vehicles is the ability to use the wire as opposed to an AGV without the need to have a hardware system such as an overhead wire or track on the floor. The disadvantage of wire guided vehicles is the necessity of installing new wire in the floor if a route change is required.

An *air cushion vehicle* is guided by some external hardware device, such as an overhead wire, but glides on a cushion of air rather than a track system. When using air cushioned vehicles, particular attention to chip removal and control must be built into the FMS. Chips in the path of an air cushion vehicle will stop its progress. These vehicles are generally used for straight paths.

Hardware guided vehicles are the most reliable but least flexible of the transfer vehicles. A track on the floor or an overhead guide rail controls the vehicle path. The advantages of these vehicles are their reliability and the ease of coordinating them with the rest of the system. The major disadvantage is, of course, the need to run new rail or track whenever a vehicle route change or new route is deemed necessary. A large FMS may employ several different types of vehicles, depending on the requirements of different parts of the manufacturing line.

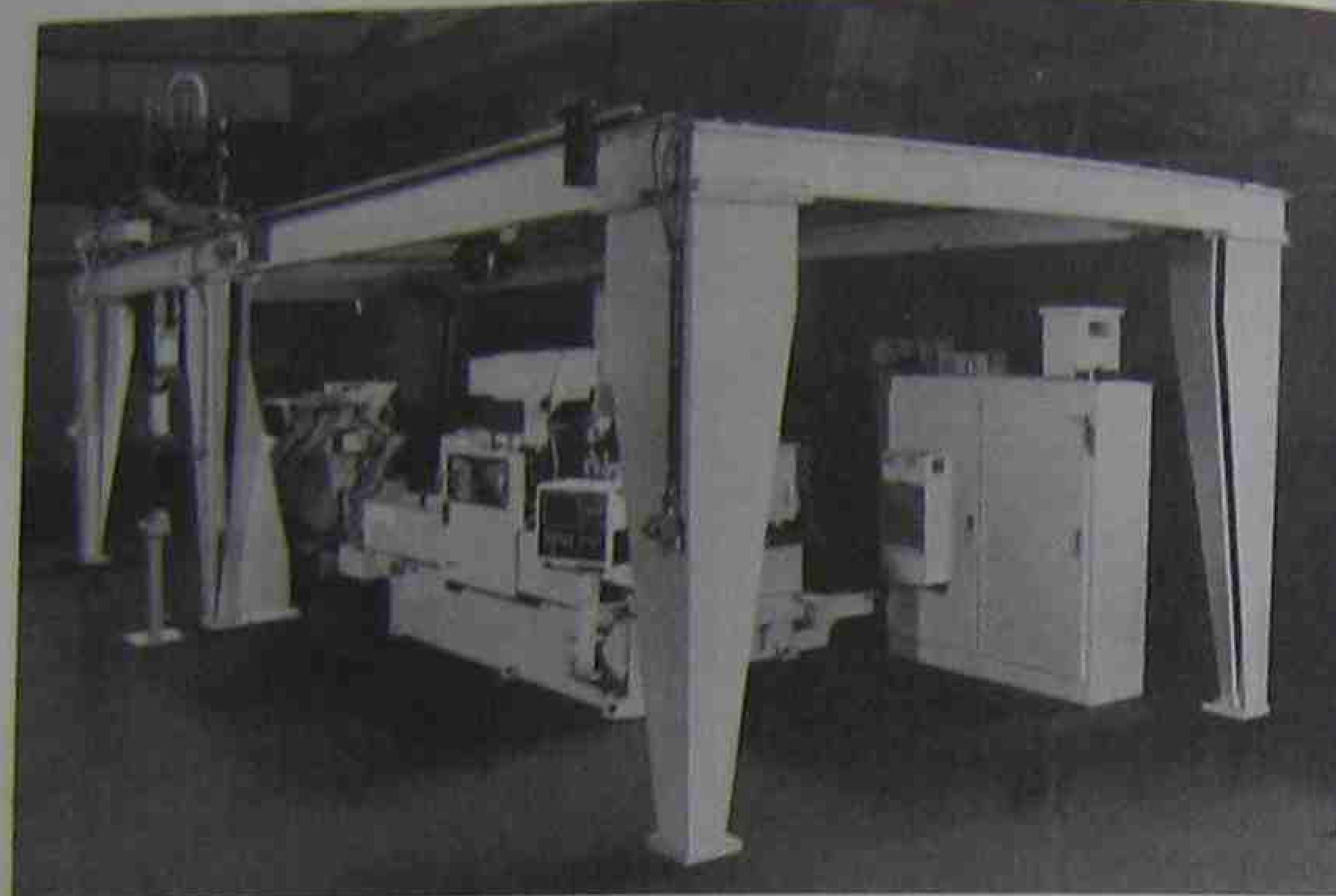


FIGURE 16–2 *(Photo courtesy of Cincinnati Milacron)*

Machining Cells

Large flexible machining systems are often a collection of smaller coordinated units called machining cells. A *machining cell* is a system consisting of one or more CNC machines and a parts handling device, such as a robot. The cell performs a machining operation or a specific sequence of operations. Flexible machining systems are not in widespread use at present although their numbers are increasing. Stand-alone machining cells, however, are widely employed by manufacturers, frequently with a view to incorporating them into an FMS at a later date. Figure 16–2 shows a machining cell consisting of a turning center and a grinder. A robot on an overhead gantry services the two machines.

EMPLOYMENT OPPORTUNITIES IN NC

A number of skilled positions have been created by numerical control. The most common jobs are NC electronics technician, machine operator/setup operator, and part programmer.

Electronics Technician

Numerical control and computer numerical control equipment are electrical systems interfaced to a machine tool. The electronics necessary for a CNC machine to function are complex. The NC electronics technician is a skilled technician who specializes in the maintenance of numerical control equipment. The NC technician must be well trained in digital electronics and possess a knowledge of the cycles and functions of NC machinery. The technician must be able to troubleshoot and correct problems that occur in the electronic circuitry of various NC machines.

NC technicians generally acquire their skills through a two-year junior college program in digital electronics. Additional education in numerical control is often provided by the employer in the form of NC manufacturers' technical school classes and seminars.

Machine Operator/Setup Operator

The machine operator/setup operator is responsible for preparing an NC machine to run a program and for setting up the fixtures, tools, and workpieces. The operator must possess a knowledge of general machine shop practices and techniques, as well as the cycles and functions of an NC machine. The operator is responsible for overriding programmed speeds and feeds if required during machining. The operator also assigns the tool length offsets to the appropriate tool registers and may be called upon to single-step a program through its first cycle. The operator must also be trained in the use of precision measuring instruments as he or she is often responsible for measuring the parts as they are finished.

Machine operators/setup operators acquire their training either by years of running other types of manufacturing equipment and then transferring to an NC operator's position, or through a two-year junior college program. Factory seminars and other coursework may be provided by the employer as required.

Part Programmer

The part programmer is a highly skilled individual responsible for writing the programs that run on numerically controlled equipment. He or she must be trained in general machine shop practice, mathematics, and the use of computers. Based on the part drawing, the programmer selects a machine to machine the part and devises a machining strategy, listing the tools to be used and the coordinates necessary to accomplish the operations. This information is then assembled into a part program written for the particular machine selected.

An NC programmer may acquire training through a two-year junior college, a four-year engineering technology degree program, or by transferring from positions as journeyman machinists or tool and die makers. NC programmers take additional coursework and factory seminars as required by the employer. The educational requirements for a programmer vary with the employer.

SUMMARY

The important concepts presented in this chapter are:

- The use of CNC will increase in prototype and small job shops due to the arrival of lower cost controllers containing many advanced programming features.
- A flexible machining system is an unmanned manufacturing/assembly line that can take a part from raw stock and perform all the necessary operations to produce a finished part or assembly.
- A machining cell is a system of one or more CNC machines and part handling robots that performs a specific sequence of operations.
- An NC electronics technician is responsible for maintaining the electronics of an NC or CNC system.
- An NC operator/setup operator is responsible for preparing a machine prior to running a program and monitoring the machine during the program execution.
- An NC part programmer is responsible for creating the part program.

REVIEW QUESTIONS

1. For what reason will the use of CNC increase in one-of-a-kind and prototype shops?
2. What is a flexible machining system?
3. What are the four major components of an FMS?
4. What are the four types of part/tool transfer vehicles?
5. What is a machining cell?
6. What are the responsibilities of the CNC electronics technician?
7. What are the responsibilities of the CNC machine operator/setup operator?
8. What are the responsibilities of the CNC part programmer?

APPENDIX 1

EIA Codes

PREPARATORY FUNCTIONS

G00—Denotes rapid traverse for point-to-point positioning.
G01—Linear interpolation.
G02—Circular interpolation clockwise.
G03—Circular interpolation counterclockwise.
G04—Dwell.
G05–07—Unassigned.
G08—Acceleration at a smooth rate.
G09—Deceleration at a smooth rate.
G10–16—Unassigned.
G13–16—Axis selection codes.
G17—XY plane selection.
G18—ZX plane selection.
G19—YZ plane selection.
G20–32—Unassigned.
G33—Thread cutting, constant lead.
G34—Thread cutting, increasing lead.
G35—Thread cutting, decreasing lead.
G36–39—Unassigned.
G40—Cutter diameter compensation cancel.
G41—Cutter diameter compensation left.
G42—Cutter diameter compensation right.
G43—Cutter compensation inside corner (used to adjust for differences in programmed and actual cutter size).
G44—Cutter compensation outside corner (used to adjust for differences in programmed and actual cutter size).
G45–49—Unassigned.
G50–59—Used with adaptive controls.
G60–69—Unassigned.
G70—Inch programming.
G71—Metric programming.
G72—Three-dimensional circular interpolation clockwise.
G73—Three-dimensional circular interpolation counterclockwise.
G74—Multiquadrant circular interpolation cancel.

G75—Multiquadrant circular interpolation.
G76–79—Unassigned.
G80—Cycle cancel.
G81—Drill cycle.
G82—Drill cycle with dwell.
G83—Intermittent or deep hole drilling cycle.
G84—Tapping cycle.
G85–89—Boring cycles.
G90—Absolute positioning.
G91—Incremental positioning.
G92—Register preload code.
G93—Inverse time feedrate.
G94—Inches (millimeters) per minute feedrate.
G95—Inches (millimeters) per revolution feedrate.
G96—Unassigned.
G97—Revolutions per minute spindle speed.
G98–99—Unassigned.

MISCELLANEOUS FUNCTIONS

M00—Program stop.
M01—Optional (planned) stop.
M02—End of program.
M03—Spindle on clockwise.
M04—Spindle on counterclockwise.
M05—Spindle off.
M06—Tool change.
M07—Coolant on (flood).
M08—Coolant on (mist).
M09—Coolant off.
M10—Automatic clamp.
M11—Automatic unclamp.
M12—Synchronize multiple axes.
M13—Spindle clockwise and coolant on.
M14—Spindle counterclockwise and coolant on.
M15—Rapid motion positive direction.
M16—Rapid motion negative direction.
M17–18—Unassigned.
M19—Spindle orient and stop.
M20–29—Unassigned.
M30—End of tape, will rewind tape automatically.

M31—Interlock bypass.
M32–39—Unassigned.
M40–46—Gear changes if used, otherwise unassigned.
M47—Continues program execution from the start of program.
M48—Cancel M47.
M49—Deactivate manual speed or feed override.
M50–57—Unassigned.
M58—Cancel M59.
M59—RPM hold.
M60–99—Unassigned.

OTHER ADDRESSES

A—Rotary motion about the X axis.
B—Rotary motion about the Y axis.
C—Rotary motion about the Z axis.
D—Angular dimension around a special axis. Also used for a third feed function.
E—Angular dimension around a special axis, or special feed function.
H—Unassigned.
I—X axis arc centerpoint.
J—Y axis arc centerpoint.
K—Z axis arc centerpoint.
L—Unassigned.
O—Used on some controllers in place of N address for sequence numbers.
P—Special rapid traverse code, or a third axis parallel to the X axis.
Q—Special rapid traverse code, or a third axis parallel to the Y axis.
R—Special rapid traverse code, or a third axis parallel to the Z axis. Also used for radius designation.
U—Secondary axis parallel to X.
V—Secondary axis parallel to Y.
W—Secondary axis parallel to Z.

APPENDIX 2

Machinist Shop Language Commands

COMMANDS

This is a list of Machinist Shop Language commands used with the CNC machine in this text.

A (absolute)—Specifies absolute positioning.

ARC—If used by itself, institutes the cutting of an arc. If used with CW or CCW, tells the computer that an arc is to be cut in a clockwise or counterclockwise direction. Following an ARC/direction command the computer will look for information describing the arc in the following two events.

AUX (auxiliary)—Allows changes to be made in normal control functions. The direction of the X, Y, and Z axes may be changed and mirror imaging may be instituted, for example. AUX codes act like the miscellaneous functions in word address format.

CALL—Executes a subroutine. CALL 1 for example, tells the machine to carry out the instructions in subroutine 1.

CCW—Specifies a counterclockwise arc rotation.

CW—Specifies a clockwise arc rotation.

DO—Do loop. Anything that is repeated over equal intervals of space (a row of holes, for example) may be placed in a do loop. Do 5 tells the machine to perform the operation that follows 5 times.

DWELL—Halts execution of the program until the start button is manually depressed. When start is pressed, the program continues, starting at the next event.

END—There are three uses for the END command. (1) In a do loop, END signals the end of the loop. (2) In a subroutine, END signals the end of the subroutine. (3) In a program, END signals the end of the program.

F (feed)—Tells the machine to make tool movements at the programmed feedrate.

FEED—Assigns a feedrate.

G (G code)—A preparatory function, G code calls up certain "canned" or standard cycles contained within the computer for such operations as drilling, boring, and reaming.

I (incremental)—Specifies incremental positioning.

R (rapid)—Tells the machine to make tool movements at rapid traverse.

SUBR (subroutine)—Like a miniprogram within a program, sections of a program that are to be repeated are often placed in a subroutine to eliminate having to program the same information twice. The subroutine is instituted by using the CALL command.

TOOL—Like a dwell, halts the program so that a tool can be inserted in the spindle. If the machine is equipped with three axes, TOOL also acts to assign certain tool length and/or cutter diameter values.

V (variable)—Assigns values to certain program variables such as canned cycle feedrates and feed engagement points.

PREPARATORY FUNCTIONS (G CODES)

Following is a list of preparatory functions used in conjunction with Machinist Shop Language.

G40—Cutter diameter compensation cancel.
G41—Cutter diameter compensation left.
G42—Cutter diameter compensation right.
G51—Institute polar rotation.
G52—Polar rotation cancel.
G53—Institute scaling.
G54—Scaling cancel.
G76—Hole milling.
G77—Circular pocket milling.
G78—Rectangular pocket milling.
G79—Bolt circle pattern.
G80—Canned cycle cancel.
G81—Basic drilling cycle.
G82—Counter-boring/spot-facing cycle (feed in, timed dwell, rapid out).
G83—Peck drilling cycle (feed in, rapid out, feed in, etc.).
G85—Boring cycle (feed in, feed out).
G86—Boring in one direction cycle (feed in, rapid out).
G87—Chip breaking cycle (feed in, retract .050, feed in, etc.).
G89—Flat bottom boring cycle (feed in, timed dwell, feed out).

VARIABLE (V) CODES

Following is a list of variable codes commonly available in Machinist Shop Language.

V11—X axis polar center (must be absolute dimension).
V12—Y axis polar center (must be absolute dimension).
V13—Polar rotation index angle (must be incremental). A negative number indicates clockwise rotation, a positive number counterclockwise rotation.
V14—Radius for polar moves (value must be positive).
V15—Angle for polar moves or angle of first hole in a bolt circle pattern.
V16—Angle of last hole in a bolt circle pattern or X axis scaling value.
V17—Number of holes to be machined in a bolt circle or Y axis scaling value.
V18—Diameter of bolt circle or Z axis scaling value.
V20—Feedrate for G80 series canned cycle.
V21—Buffer zone for G80 series canned cycles. Must be .100 for G83 or G87.
V22—Dwell time when using G82 or G89.
V23—Maximum peck when using G83 or G87.
V40—Z axis start height for pecked milling.
V41—Length of pocket on X axis (must be incremental).
V42—Width of pocket on Y axis (must be incremental) or number of rotations for helical interpolation.
V43—Depth of pocket on Z axis.
V44—Pocket corner radius or diameter of circle if circular pocket milling.
V45—Stepover value for pocket milling.
V46—Maximum depth of cut.
V47—Stock left for finish pass.
V48—Finish pass feedrate.
V49—Tool diameter for pocket milling (cutter comp cannot be active for pocket milling).

AUXILIARY (AUX) CODES

Following is a complete list of auxiliary codes commonly used in Machinist Shop Language.

AUX 100—Reverses sign of X axis.
AUX 200—Reverses sign of Y axis.
AUX 300—Reverses sign of X and Y axes.
AUX 400—Reverses sign of Z axis.
AUX 500—Reverses sign of X and Z axes.

AUX 600—Reverses sign of Y and Z axes.
AUX 700—Reverses sign of X, Y, and Z axes.
AUX 800—Turns off mirror image.
AUX 1000—Causes machine to continue to the next move before reaching its target (used only with contouring operations).
AUX 1101—Absolute zero shift.
AUX 1110—Turns off software limits.
AUX 1111—Turns on software limits.
AUX 1400—Feed percentage override for feedrate moves.
AUX 1401—Feed percentage override for feed and rapid moves.
AUX 1900—Single-step event mode.
AUX 1901—Single-step axis movement mode.
AUX 2000—Cancels AUX 1000.
AUX 2500—Sets control to use Z axis.
AUX 2600—Sets control to allow manual use of Z axis.

APPENDIX 3

Word Address Codes Used in Text Examples

PREPARATORY FUNCTIONS (G CODES) USED IN MILLING

Following is a list of preparatory functions used in CNC milling examples in this text. Other codes commonly used on General Numeric controllers are also listed.

G00—Rapid traverse positioning.
G01—Linear interpolation (feedrate movement).
G02—Circular interpolation clockwise.
G03—Circular interpolation counterclockwise.
G04—Dwell.
G10—Tool length offset value.
G17—Specifies X/Y plane.
G18—Specifies X/Z plane.
G19—Specifies Y/Z plane.
G20—Inch data input (on some systems).
G21—Metric data input (on some systems).
G22—Safety zone programming.
G23—Cross through safety zone.
G27—Reference point return check.
G28—Return to reference point.
G29—Return from reference point.
G30—Return to second reference point.
G40—Cutter diameter compensation cancel.
G41—Cutter diameter compensation left.
G42—Cutter diameter compensation right.
G43—Tool length compensation positive direction.
G44—Tool length compensation negative direction.
G45—Tool offset increase.
G46—Tool offset decrease.
G47—Tool offset double increase.
G48—Tool offset double decrease.

G49—Tool length compensation cancel.
G50—Scaling off.
G51—Scaling on.
G73—Peck drilling cycle.
G74—Counter tapping cycle.
G76—Fine boring cycle.
G80—Canned cycle cancel.
G81—Drilling cycle.
G82—Counter boring cycle.
G83—Peck drilling cycle.
G84—Tapping cycle.
G85—Boring cycle (feed return to reference level).
G86—Boring cycle (rapid return to reference level).
G87—Back boring cycle.
G88—Boring cycle (manual return).
G89—Boring cycle (dwell before feed return).
G90—Specifies absolute positioning.
G91—Specifies incremental positioning.
G92—Program absolute zero point.
G98—Return to initial level.
G99—Return to reference (R) level.

MISCELLANEOUS (M) FUNCTIONS USED IN MILLING AND TURNING

Following is a list of miscellaneous functions used in the milling and turning examples in this text. Other M functions common to General Numeric and Fanuc controllers are also listed.

M00—Program stop.
M01—Optional stop.
M02—End of program (rewind tape).
M03—Spindle start clockwise.
M04—Spindle start counterclockwise.
M05—Spindle stop.
M06—Tool change.
M08—Coolant on.
M09—Coolant off.
M13—Spindle on clockwise, coolant on (on some systems).
M14—Spindle on counterclockwise, coolant on.
M17—Spindle and coolant off (on some systems).

M19—Spindle orient and stop.
M21—Mirror image X axis.
M22—Mirror image Y axis.
M23—Mirror image off.
M30—End of program, memory reset.
M41—Low range.
M42—High range.
M48—Override cancel off.
M49—Override cancel on.
M98—Jump to subroutine.
M99—Return from subroutine.

PREPARATORY FUNCTIONS (G CODES) USED IN TURNING

Following is a list of preparatory functions used in CNC milling examples in this text. Other codes commonly used on Fanuc controllers are also listed.

G00—Rapid traverse positioning.
G01—Linear interpolation (feedrate movement).
G02—Circular interpolation clockwise.
G03—Circular interpolation counterclockwise.
G04—Dwell.
G10—Tool length offset value setting.
G17—Specifies X/Y plane.
G18—Specifies X/Z plane.
G19—Specifies Y/Z plane.
G20—Inch data input (on some systems).
G21—Metric data input (on some systems).
G22—Stored stroke limit on.
G23—Stored stroke limit off.
G27—Reference point return check.
G28—Return to reference point.
G29—Return from reference point.
G30—Return to second reference point.
G40—Tool nose radius compensation cancel.
G41—Tool nose radius compensation left.
G42—Tool nose radius compensation right.
G50—Programming of work coordinate system.
G68—Mirror image for double turrets on.
G69—Mirror image for double turrets off.

G70—Inch programming (some systems) or finish cycle.
G71—Metric programming (some systems) or stock removal in turning code.
G72—Stock removal in facing code.
G73—Pattern repeat.
G74—Z axis peck drilling.
G75—Groove cutting cycle, X axis.
G76—Multipass thread cutting.
G90—Absolute positioning.
G91—Incremental positioning.
G94—Per minute feed (some systems).
G95—Per revolution feed (some systems).
G98—Per minute feed (some systems).
G99—Per revolution feed (some systems).

APPENDIX 4

Codes in Common Use with Tape Machinery

PREPARATORY FUNCTIONS (g CODES)

(Note: On tape machinery, lowercase letters are generally used.)

g01—Linear interpolation.
g02—Circular interpolation clockwise.
g03—Circular interpolation counterclockwise.
g78—Mill cycle stop. A milling code used to position a spindle before lowering it. Upon receiving a g78 the spindle moves at rapid traverse to the programmed x/y coordinates, then rapids down to the feed engagement point, then feeds down to final depth at feedrate. The spindle is then clamped (either manually or automatically).
g79—Mill cycle. Usually (though not always) used following a g78. Upon receiving a g79 the spindle moves to the programmed x/y coordinates at feed rate, then rapids and subsequently feeds down to depth. If used following a g78, the spindle moves to the programmed coordinates at feedrate, since the spindle is already down.
g80—Cancel cycle.
g81—Drill cycle.
g84—Tapping cycle.
g85—Boring cycle.

MISCELLANEOUS (m) FUNCTIONS

m00—Program stop.
m02—End of program.
m03—Spindle on clockwise.
m04—Spindle on counterclockwise.
m05—Spindle off.
m06—Tool change.

m07—Flood coolant on.
m08—Mist coolant on.
m09—Coolant off.
m10—Clamp spindle.
m11—Spindle unclamp.
m13—Spindle on clockwise, flood coolant on.
m14—Spindle on counterclockwise, flood coolant on.
m17—Spindle on clockwise, mist coolant on.
m18—Spindle on counterclockwise, mist coolant on.
m26—Pseudo tool change. Used primarily for clamp changes. On some machines the spindle positions at rapid traverse to the tool change location but no tool change takes place. On other machines the spindle retracts to its "home" position.
m30—End of tape. Rewinds the tape to the start on machines where m02 does not do so.
m50–59—Z axis cam selection. Selects which of nine cams is to control z axis motion. m50 specifies no cam while m51–59 specify cams 1–9 respectively. Some machines use a 'w' function instead of an 'm' function to select cams.

APPENDIX 5

Safety Rules for Numerical Control

SAFETY RULES FOR OPERATING MACHINES

1. Use common sense in all situations.
2. Wear safety glasses at all times on the shop floor.
3. Wear safety shoes.
4. Keep long hair covered when operating or standing near a machine.
5. Do not wear jewelry (including rings), neckties, long sleeves, or loose clothing while operating machines.
6. Keep the floor free of obstructions.
7. Clean oil and grease spills immediately.
8. Do not play with compressed air or engage in general horseplay around machinery.
9. Do not use compressed air to clean machine slides. Chips blown under the machine ways will cause premature wear.
10. Do not perform grinding operations near NC machinery. Grinding grit will cause premature machine slide wear.
11. Platforms around machinery should be kept clean and have antislip surfaces.
12. Use caution when lifting heavy parts, tooling, or fixturing. Lift with the legs, not the back.
13. Keep tools and other parts off the machine.
14. Keep hands away from the spindle while it is revolving, and away from other moving parts of the machine.
15. Use a cloth or gloves when handling tools by their cutting edges.
16. Use caution when changing tools.
17. Use caution to avoid inadvertently bumping any NC controls.
18. Do not operate controls unless you have been instructed in their use.
19. Keep electrical panels in place. Electrical work should be performed only by qualified service personnel.
20. Make sure safety guards and devices are in place and working before operating the machine.

21. Do not remove chips from the machine or workpiece with hands or fingers. Use a brush. Do not remove chips with the spindle running.
22. Respect the programmer's knowledge of the machine.

SAFETY RULES FOR PROGRAMMERS

1. Never assume! When in doubt check the manual.
2. Do not attempt to program a machine without access to the programming manual for the machine tool and controller.
3. Cancel all modal commands in the first line of the program to ensure commands are not active when the program is cycled the first time.
4. Be sure all modal commands have been canceled at the end of the program so that no codes are active at the start of the next program.
5. Use a buffer zone between the part and the feed engagement point for all tool moves into the workpiece.
6. Respect the machine operator's knowledge of the machine.
7. When on the shop floor:
   a. Wear safety glasses at all times.
   b. Wear safety shoes.
   c. Remove neckties or tuck them inside your shirt.
   d. Keep hands away from moving machine parts.
   e. Keep long hair safely secured or covered.

APPENDIX 6

Useful Machining Formulas and Data

MACHINING FORMULAS

To determine spindle RPM:

$$RPM = \frac{(CS \times 4)}{D}$$

Where: CS is the material cutting speed in surface feet per minute, and D is the diameter of the part or cutter revolving in the spindle.

To determine feedrates:

1. Milling feedrates:

$$FEED = RPM \times T \times N$$

Where: T is the chip load per tooth and N is the number of teeth on the cutter.

2. Lathe feedrates (commonly .002 – .025 inch per revolution):

$$FEED\ (in./rev) = I/RPM$$

Where: I is the feedrate in inches per minute.

$$FEED\ (in./min) = RPM \times r$$

Where: r is the feedrate in inches per revolution.

To determine lead of a thread:

$$LEAD = P \times I$$

Where: P is the pitch of the thread and I is the number of leads on the thread.

To determine pitch of a thread:

$$PITCH = 1/N$$

Where: N is the number of threads per inch.

To determine tap drill diameter of a thread (Unified threads):

$$MD - \left[\frac{1.08254 \times \%}{N}\right]$$

Where: MD is the major diameter of the thread, % is the percentage of thread engagement desired, and N is the number of threads per inch.

To determine length of a drill point:

$$DRILL\ POINT = .3 \times DRILL\ DIAMETER$$

To determine depth of countersink to achieve a given diameter:

$$DEPTH = A - (B \times C)$$

Where: A is the diameter of countersink desired, B is the diameter of the hole, and C is a constant as follows:

.35 for a 110-degree countersink
.50 for a 90-degree countersink
.57 for a 82-degree countersink
.35 for a 60-degree countersink

To determine circumference of a circle:

$$CIRCUMFERENCE = DIAMETER \times PI$$

To determine diameter of a circle:

$$DIAMETER = CIRCUMFERENCE \times .31831$$

To determine area of a circle:

$$AREA = PI \times RADIUS^2$$

$$AREA = \frac{1}{2}C \times \frac{1}{2}D$$

Where: C is the circumference and D is the diameter.

To determine surface area of a sphere:

$$SURFACE = DIAMETER^2 \times PI$$

To determine volume of a sphere:

$$VOLUME = DIAMETER^3 \times .5236$$

CUTTING SPEED DATA

The following rates are averages for high-speed steel cutters. For carbide cutters, double the cutting speed value.

Cutting speeds for lathes:

MATERIAL	CUTTING SPEED
Tool steel	50
Cast iron	60
Mild steel	100
Brass, soft bronze	200
Aluminum, magnesium	300

Cutting speeds for drills:

MATERIAL	CUTTING SPEED
Tool steel	50
Cast iron	60
Mild steel	100
Brass, soft bronze	200
Aluminum, magnesium	300

Cutting speeds for milling:

MATERIAL	CUTTING SPEED
Tool steel	40
Cast iron	50
Mild steel	80
Brass, soft bronze	160
Aluminum, magnesium	200

FEEDRATE DATA

Feeds for drilling:

DRILL SIZE	FEEDRATE
$< 1/8$	.001–.002
$1/8$–$1/4$	.002–.004
$1/4$–$1/2$	.004–.007
$1/2$–1.000	.007–.015
> 1.000	.025

Feeds per tooth for milling:

MATERIAL	FACE MILLS	SIDE MILLS	END MILLS
Low carbon steel	.010	.005	.005
Medium carbon steel	.009	.005	.004
High carbon steel	.006	.003	.002
Stainless steel	.006	.004	.002
Cast iron	.012	.006	.006
Brass and bronze	.013	.008	.006
Aluminum	.020	.012	.010

CUTTER CENTERLINE FORMULAS

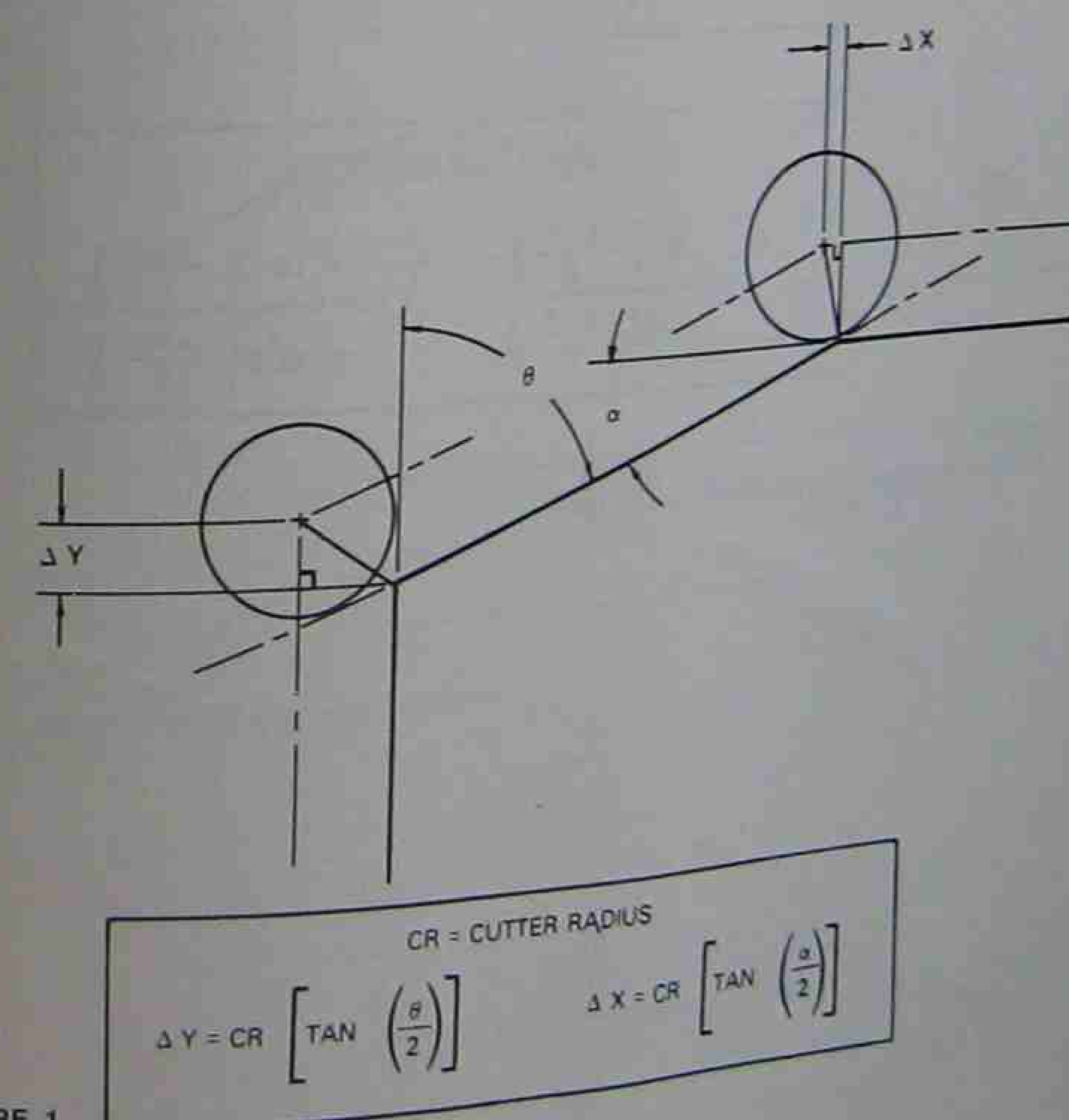
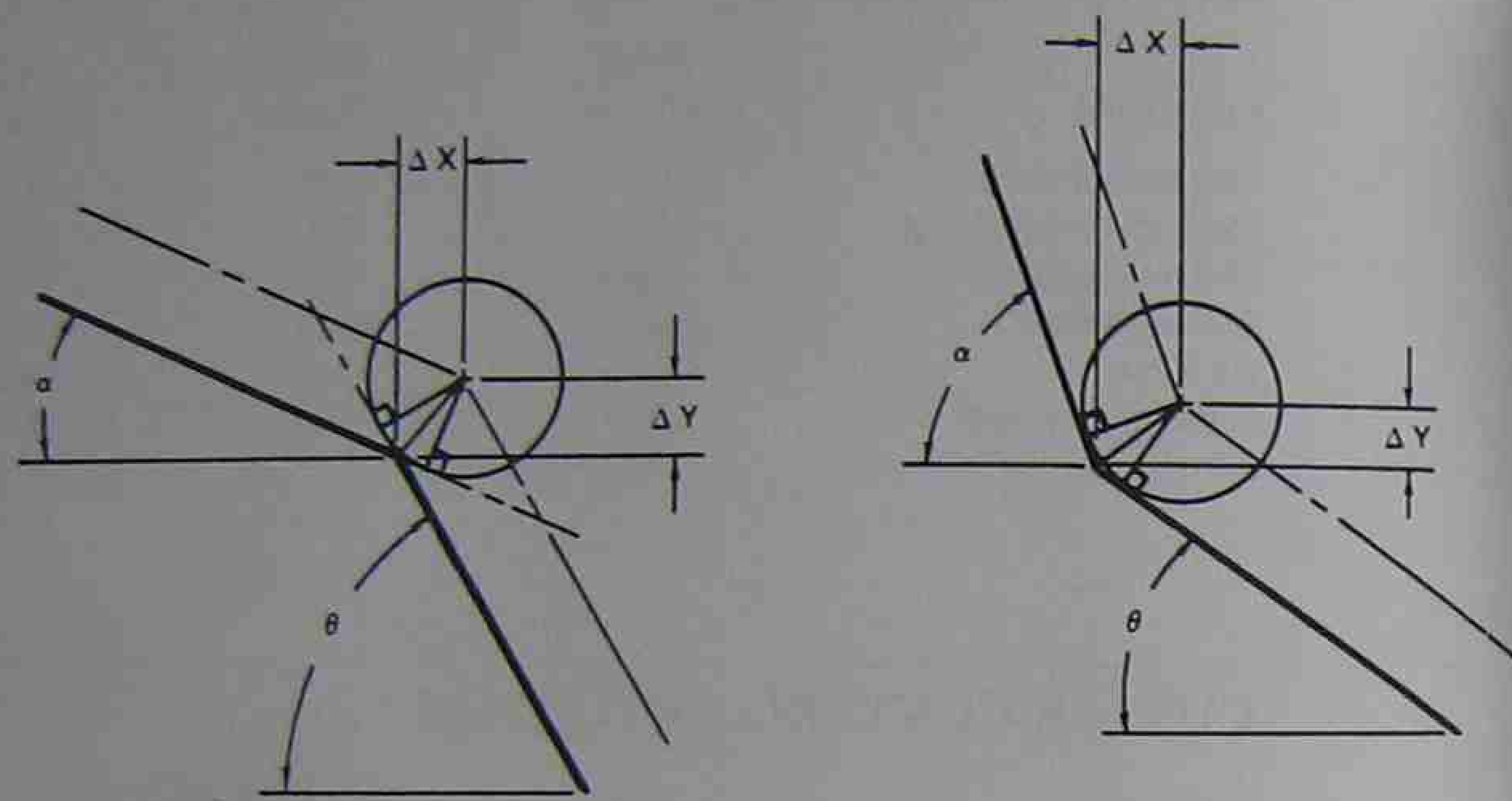


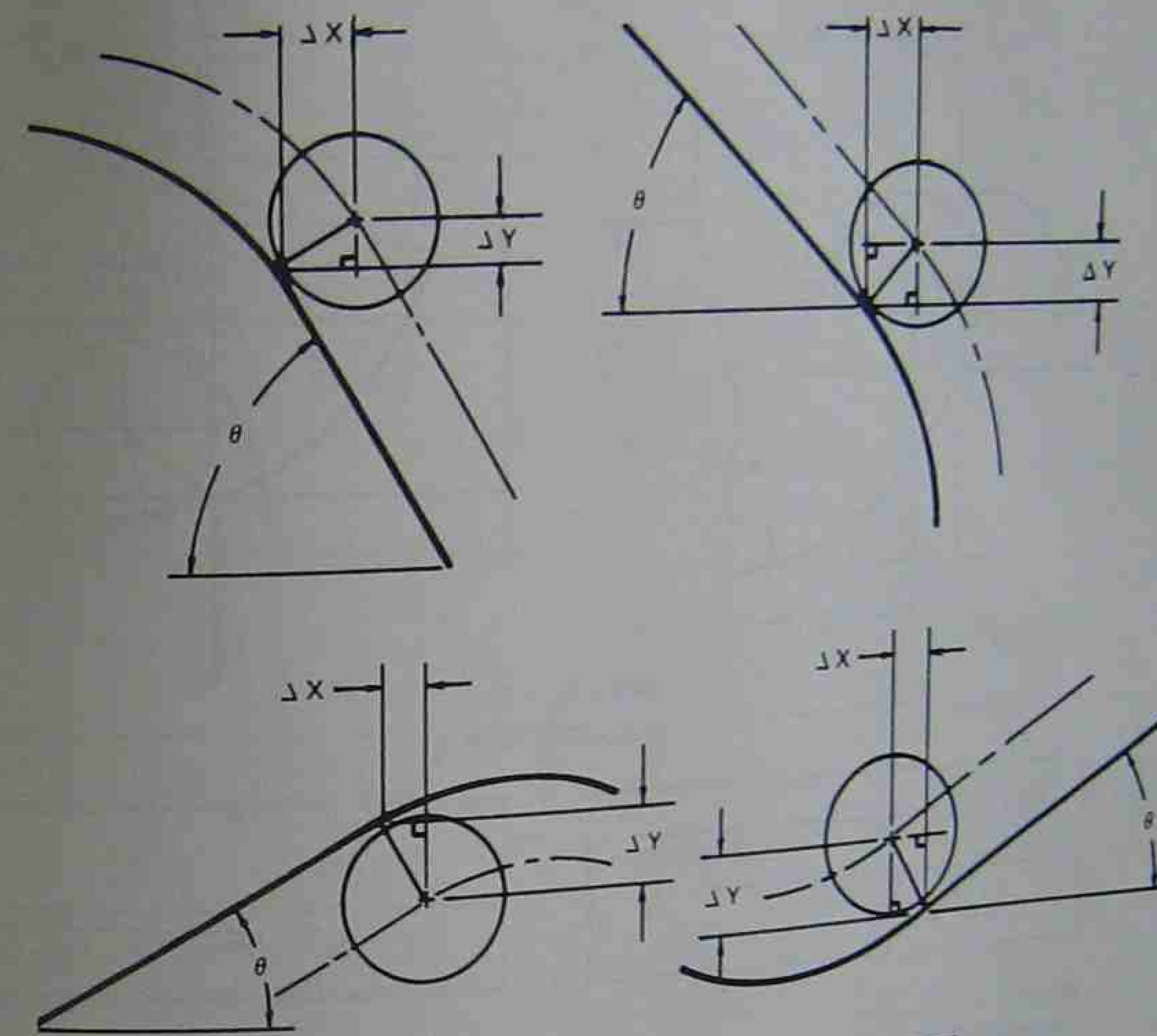
FIGURE 1



CR = CUTTER RADIUS

$$\Delta X = CR \times \frac{\sin\left(\frac{\alpha + \theta}{2}\right)}{\cos\left(\frac{\alpha - \theta}{2}\right)} \qquad \Delta Y = CR \times \frac{\cos\left(\frac{\alpha + \theta}{2}\right)}{\cos\left(\frac{\alpha - \theta}{2}\right)}$$

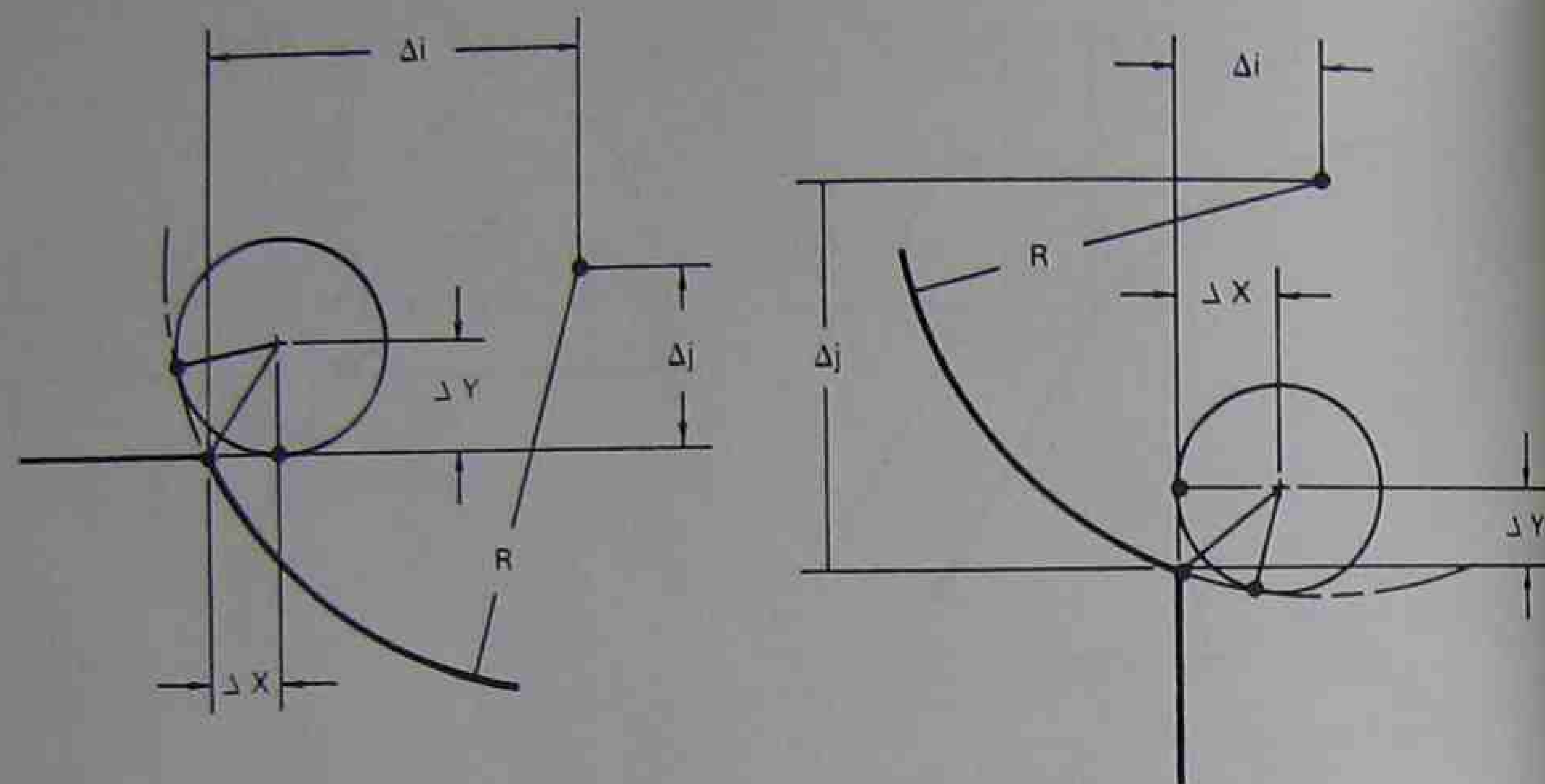
FIGURE 2
Two lines intersecting, not parallel to a machine axis



CR = CUTTER RADIUS

$$\Delta X = CR \times \sin\theta \qquad \Delta Y = CR \times \cos\theta$$

FIGURE 3
Line not parallel to a machine axis tangent to a circle



CR = CUTTER RADIUS

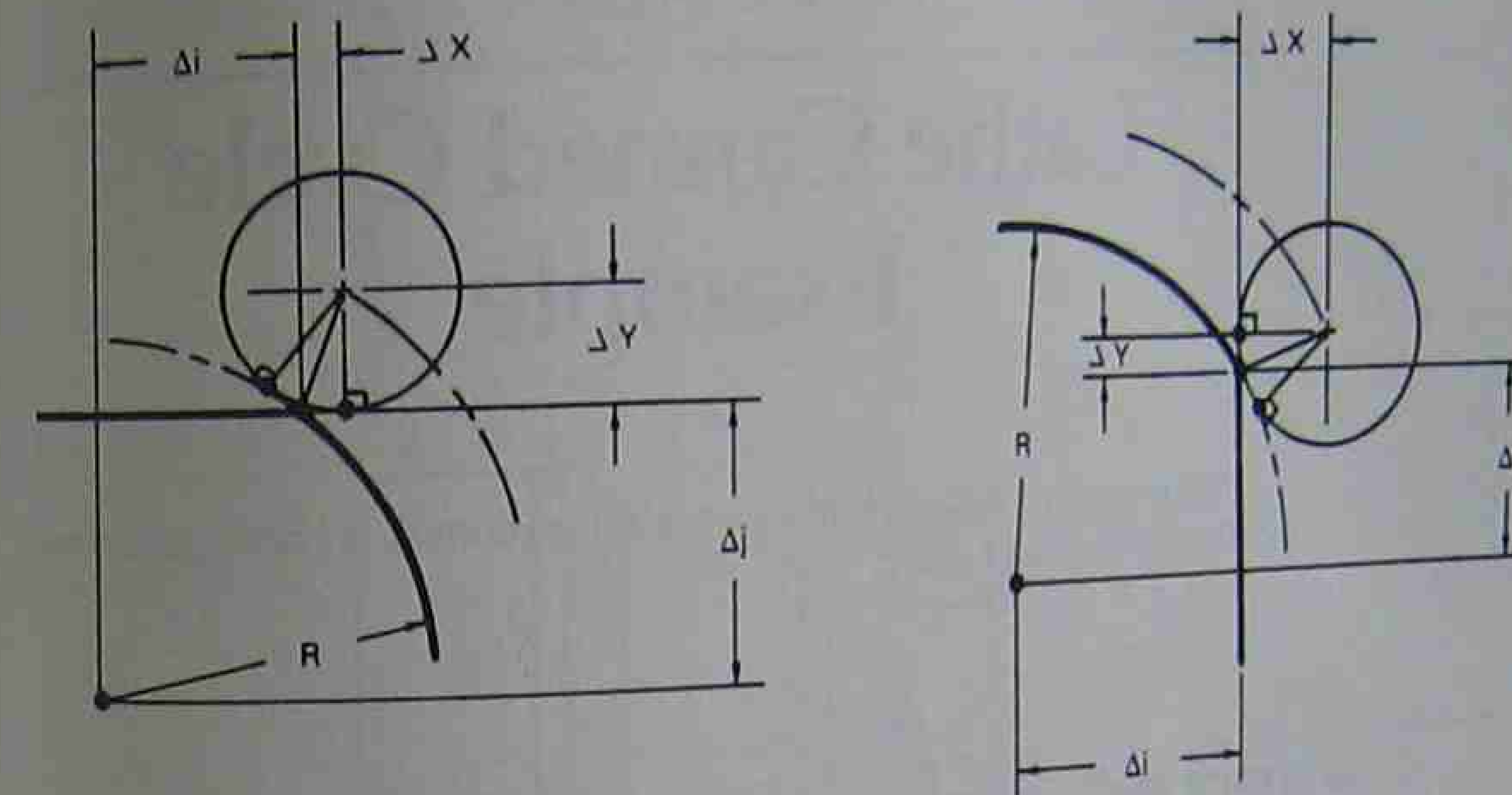
$$\Delta X = \Delta i - \sqrt{(R - CR)^2 - (\Delta j - CR)^2}$$

$$\Delta Y = CR$$

$$\Delta X = CR$$

$$\Delta Y = \Delta j - \sqrt{(R - CR)^2 - (\Delta i - CR)^2}$$

FIGURE 4
Intersection of a circle and a line parallel to a machine axis



CR = CUTTER RADIUS

$$\Delta X = \sqrt{(R - CR)^2 - (\Delta j - CR)} - \Delta i$$

$$\Delta Y = CR$$

$$\Delta X = CR$$

$$\Delta Y = \sqrt{(R - CR)^2 - (\Delta i + CR)^2} - \Delta i$$

FIGURE 5
Intersection of a circle and a line parallel to a machine axis

APPENDIX 7

Lathe Canned Cycle Example

The program contained in this section is the courtesy of Hardinge Brothers Inc. Refer to Figures 1 and 2.

TOOLING USED

TURRET STATION	TOOL OFFSET	TOOLING USED
1	1	Number 4 centerdrill
3	3	55 deg. × .030r O.D. turning tool
4	4	Number 7 drill
5	5	¼—20 tap
6	6	35 degrees × .015r O.D. turning tool
8	8	O.D. threading tool

SEQUENCE OF OPERATIONS

1. Centerdrill—1500 RPM, feedrate .009 IPR, .250 deep.
2. Rough face and turn—55 degree tool, G71 cycle, 400 constant surface speed with a maximum RPM of 4200, depth of cut .125, stock for finish pass .006, feedrate .008 IPR.
3. Tap drill—1900 RPM, feedrate .007 IPR, .875 deep.
4. Tap—250 RPM, G32 cycle, feedrate .049 IPR.
5. Finish face and turn—35 degree tool, G70 cycle, 450 constant surface speed with a maximum RPM of 5000, feedrate .004 IPM.
6. O.D. thread—G76 cycle, 1000 RPM, 8 passes used, 59 degree infeed angle.

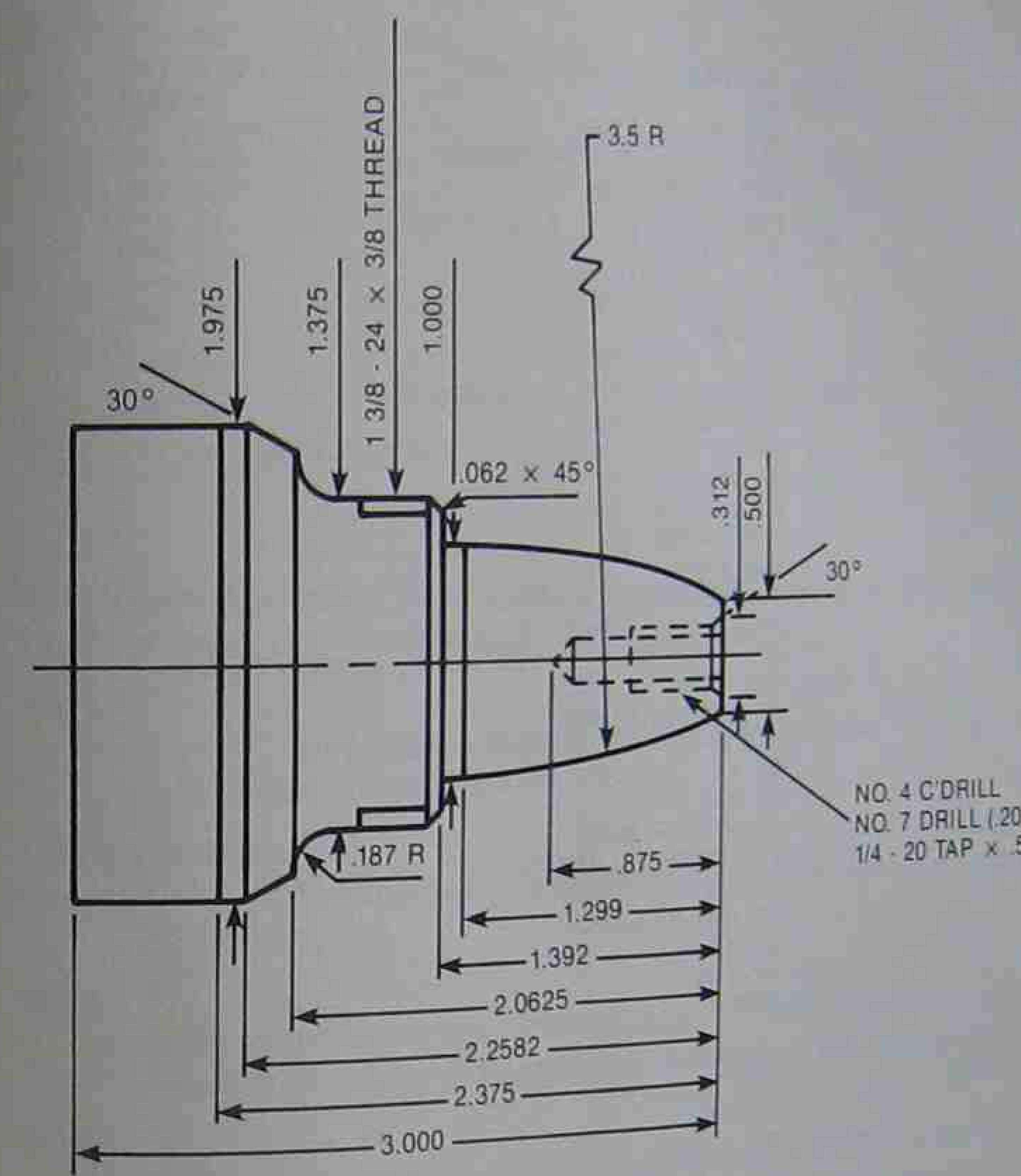


FIGURE 1
(Courtesy of Hardinge Brothers Inc.)

PROGRAM NOTES

The programmer of this part chose not to use sequence numbers on all lines. The uses of sequence numbers is up to the programmer, they are not required by the controller except where necessary for canned cycle or subroutine use.

```
%
O401 (SAMPLE PART PROGRAM)
     G20
     G65 P9150 D1.5
N1   (T0101 #4 CENTER DRILL)
     G97 S1000 M13
     M98P1
     G4 T0101
     X0 Z.100 S1500
     G99 Z-.25 F.009
     M98P2

N3   (T0303 ROUGH O.D. R.030 T3)
     G97  S1000 M13
     M98P1
     G4 T0303
     X2.16 Z.2
     G50 S4200
     G96 S400
     G42 X2.155 Z.1 F50.
     G99
     G71 P100 Q300 U.012 W.006 D12500 F.008
N100 G00 X.25
     G1 G99 Z0 F.004
     X.5
     G3 X1. Z-1.299 R3.5
     G1 Z-1.302
     X1.375 K-.062
     Z-2.0625 R.1875
     X1.75
     X1.975 Z-2.2582
     Z-2.375
N300 X2.155
     M98P2

N4   (T0404 #7 DRILL)
     G97 S1000 M13
     M98P1
     G4 T0404
     X0 Z.1 S1900
     G1 G99 Z-.875 F.007
     M98P2

N5   (T0505 1/4-20 TAP)
     G97 S250 M13
     M98P1
     G4 T0505
     X0 Z.25
     G32 Z-.406 F.049   (TAPPED .094 SHORT FOR DRIVER PULLOUT)
     G4 U.5
     G32 Z.25 F.05 M14
     M98P2
```



```
N6   (T0808 1 3/8-24 O.D. THREAD)
     G97 S1000 M13
     M98P1
     G4 T0808
     X1.4261 Z.1 S2880
     Z-1.4261 F50.
     G76 X1.3238 Z-1.767 K.0256 E.04166 D00904 A59
     M98P2
     T0100
     M30
     %
```

FIGURE 2
(Courtesy of Hardinge Brothers Inc.)

The uses of leading zeros is also optional at the programmer's discretion. In this example, G01 is written as G1, G02 as G2, etc.

0401

- The "O" number. Each program resident in the controller must have its own unique "O" number.

G65 P9150 D1.5

- Unique to Hardinge using Fanuc controls. This code sets the dwell time parameter.

M98P1

- Jump to subroutine call. This subprogram contains the startup code common to all tools. The subprogram is not part of the program printout.

M98P2

- Jump to subroutine call. This subprogram contains the tool cancel code common to all tools. The subprogram is not part of the program printout.

CANNED CYCLE NOTES

G71 P100 Q300 U.012 W.006 D12500 F.008

- G71 is the rough face and rough turn cycle.
- P100 is the sequence number (N100) that denotes the beginning of the finish pass lines.
- Q300 is the sequence number (N300) that denotes the end of the finish pass lines.

- U.012 tells the MCU to leave .012 stock on the X-axis for finishing. The lathe is diameter programmed, therefore, this will leave .006 stock per side to be finished.
- W.006 tells the MCU to leave .006 stock on the Z-axis for finishing.
- D12500 tells the MCU that the depth of each roughing pass is the .125 diameter stock removal.
- F.008 is the feedrate.

The G71 cycle will calculate backwards from the finished pass lines to generate the appropriate number of passes. The tool path shape is identical to the finish pass.

G32 Z – .406 F.049

- G32—This type of cycle was covered in Chapter 14. It is the single pass threading cycle. When used for tapping, no X infeed is necessary. It is the same as using a G01 except with G32 the speed and feed operator overrides will not function.
- F.049 is the feedrate. The tap is fed in at .001 IPR slower than the actual lead. This allows the tapping holder to pull out slightly. When the spindle is reversed at the bottom of the tapped hole, there is some travel in the holder to prevent tap breakage.

G70 P100 Q300

- G70 is the finish face and turn cycle.
- P100 is the sequence number (N100) of the finish pass lines.
- Q300 is the sequence number (N300) of the finish pass lines.

G70 will simply jump backwards in the program to the finish pass lines used with the G71 cycle and execute them. Note the G42 prior to the G70 line. Tool nose radius comp (TNR) is being used to allow for tool wear.

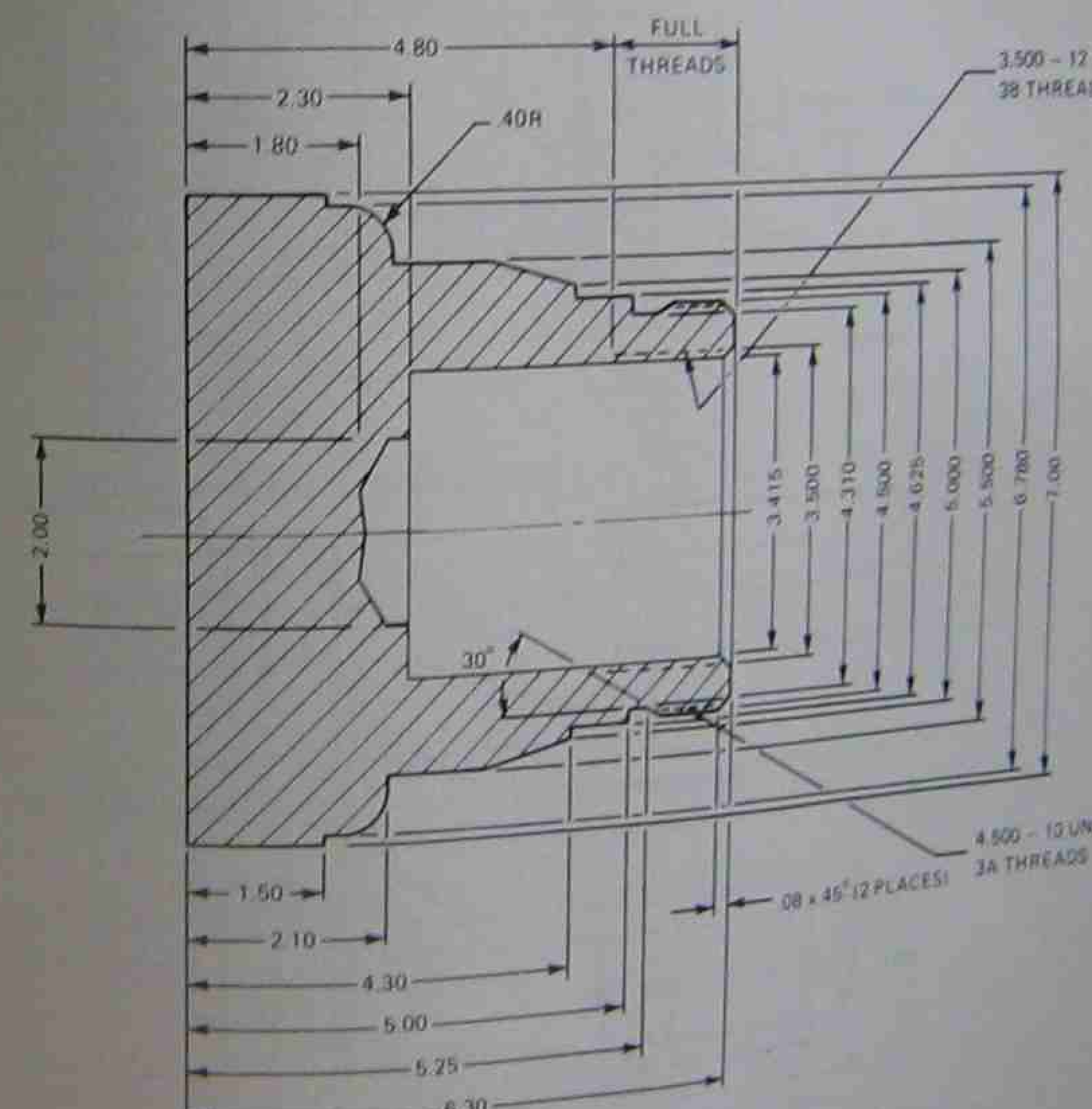
G76 X1.3238 Z – 1.767 K.0256 E.04166 D00904 A59

- This cycle was covered in Chapter 14. The A59 sets the infeed angle to 59 degrees. Other programmers prefer to use A60.

APPENDIX 8

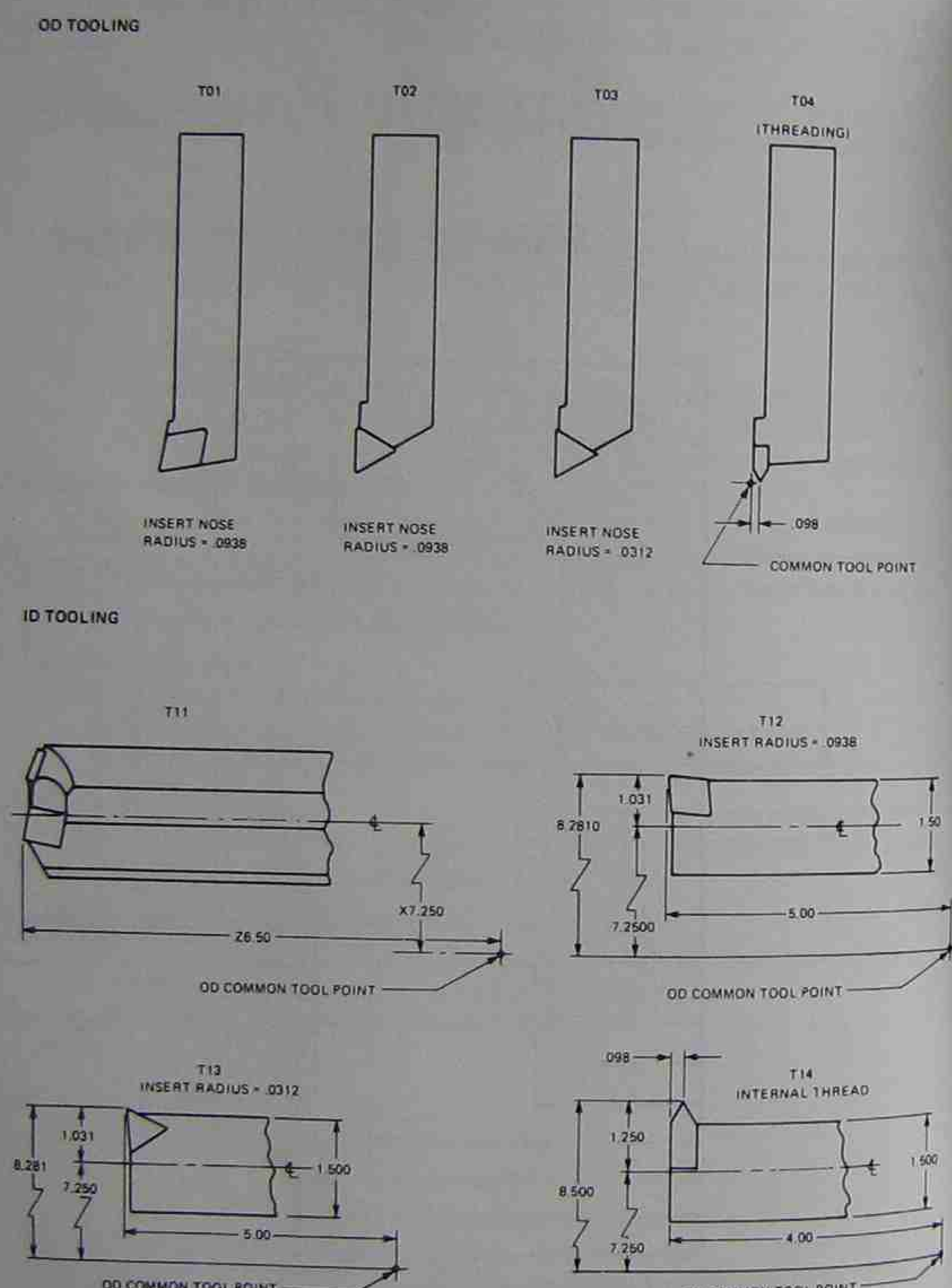
Sample Programs

SAMPLE PART PROGRAM

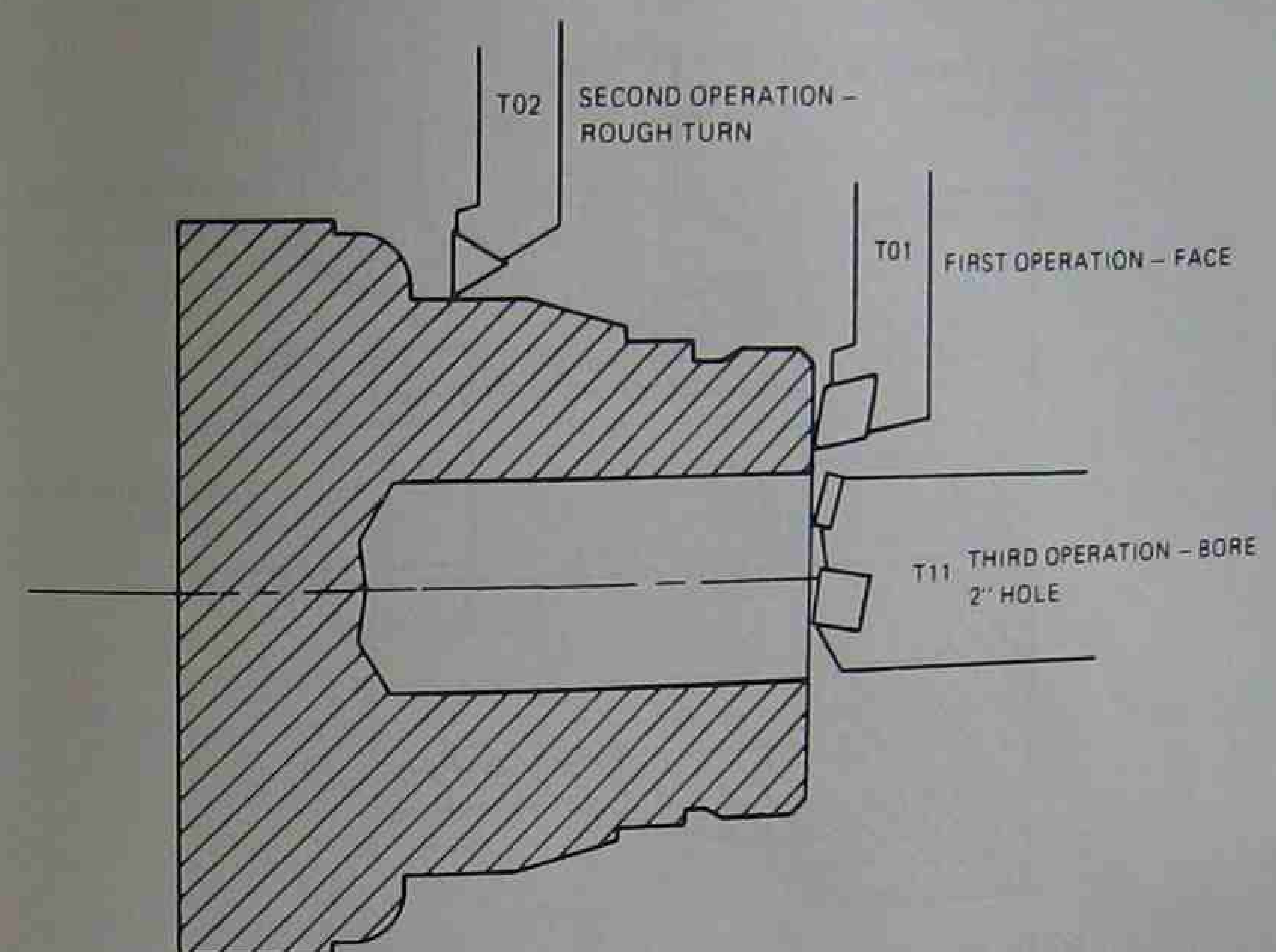


SAMPLE PART ENGINEERING DRAWING

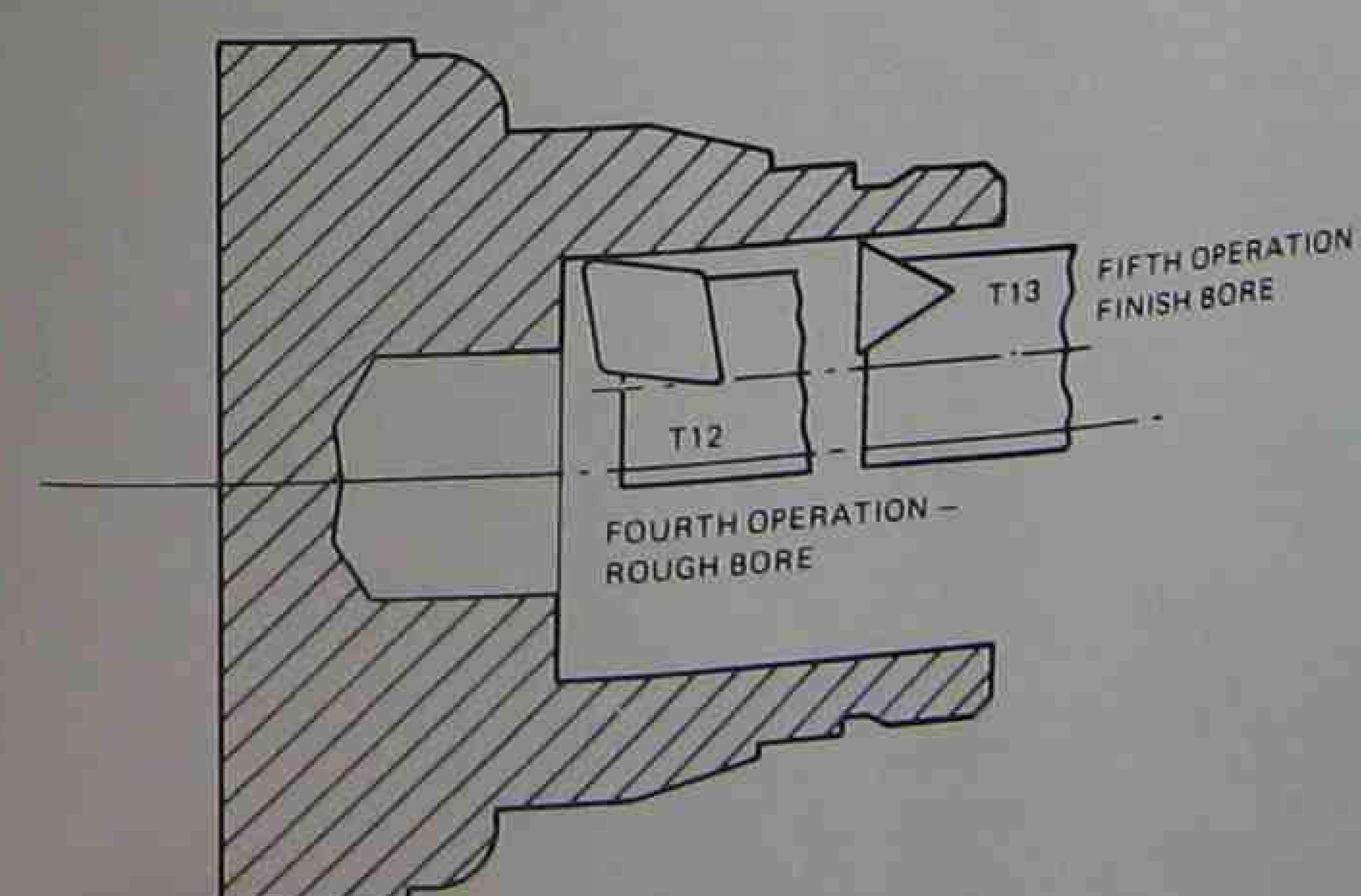
Courtesy of Cincinnati Milacron



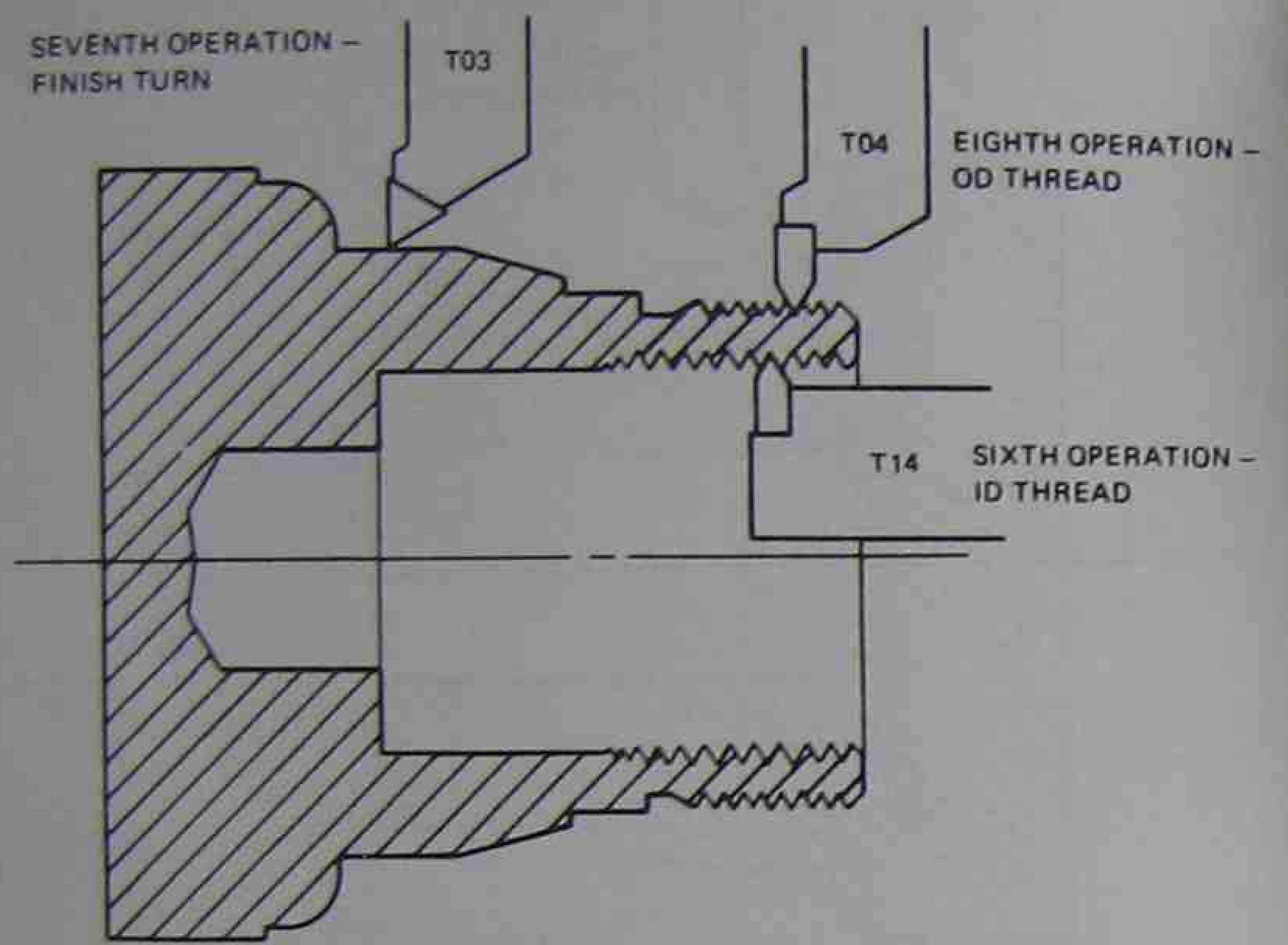
TOOLING LAYOUT



FIRST, SECOND AND THIRD OPERATIONS



FOURTH AND FIFTH OPERATIONS



SIXTH, SEVENTH AND EIGHTH OPERATIONS

SAMPLE PART PROGRAM NO. 1

```
First Operation    O10   G90
                   N20   G97 S100 M42
                   N30   G70 M03
                   N40   G00 X50000 Z85000 T0100 M06
                   N50   G95
                   N60   G92 S2500
                   N70   G96 R50000 S600
                   N80   G00 X37000 Z63500 M08
                   N90   G01 X-940 F150
                   N100  Z65500 F600
                   N110  X37000
                   N120  Z63000
                   N130  X-940 F150
                   N140  G00 Z65000
                   N150  X45000
Second Operation   O160  G90
                   N170  G97 S351 M41
                   N180  G70 M13
                   N190  G00 X45000 Z65000 T0200 M06
                   N200  G95
                   N210  G92 S2500
                   N220  G96 R45000 S600
                   N230  G00 X32600
                   N240  G01 Z19690 F150
                   N250  G03 X34100 Z16062 I28962 K16062
                   N260  G01 X37000
                   N270  G00 Z65000
                   N280  X30100
                   N290  G01 Z21200
                   N300  X32100
                   N310  G00 Z65000
                   N320  X27600
                   N330  G01 Z21100
                   N340  X28962
                   N350  G03 X34000 Z16062 I28962 K16062
                   N360  G01 Z15100
                   N370  X36000
                   N380  G00 X29600 Z65000
                   N390  X25100
                   N400  G01 Z42199
                   N410  X27600 Z32986
                   N420  G00 Z65000
                   N430  X22600
                   N440  G01 Z55989
                   N450  X21650 Z51659
                   N460  Z50100
                   N470  X23225
                   N480  Z43100
                   N490  X26000
                   N500  G00 Z130000
Third Operation    O510  G90
                   N520  G97 S600 M41
                   N530  G70 M14
                   N540  G00 X26000 Z130000 T1100 M06
```

Fourth Operation

[illegible]

Fifth Operation

[illegible]

Sixth Operation

[illegible]

[illegible]

Seventh Operation

[illegible]

Eighth Operation

[illegible]

```
                  N550  G95
                  N560  G00 X-72500
                  N570  G01 Z83000 F150
                  N580  G00 Z148000
Fourth Operation  O590  G90 M05
                  N600  G97 S1000 M42
                  N610  G70 M13
                  N620  G00 X-72500 Z148000 T1200 M06
                  N630  G95
                  N640  G92 S2500
                  N650  G96 R10310 S600
                  N660  G00 X-71560 Z115000
                  N670  G01 Z73100 F150
                  N680  X-73750
                  N690  G00 Z115000
                  N700  X-70310
                  N710  G01 Z73100
                  N720  X-72500
                  N730  G00 Z115000
                  N740  X-69060
                  N750  G01 Z73100
                  N760  X-71250
                  N770  G00 Z115000
                  N780  X-67810
                  N790  G01 Z73100
                  N800  X-70000
                  N810  G00 Z115000
                  N820  X-66560
                  N830  G01 Z73100
                  N840  X-68750
                  N850  G00 Z115000
                  N860  X-65835
                  N870  G01 Z73100
                  N880  X-68023
                  N890  G00 Z148000
Fifth Operation   O900  G90
                  N910  G97 S846 M42
                  N920  G70 M13
                  N930  G00 X-68023 Z148000 T1300 M06
                  N940  G95
                  N950  G92 S2500
                  N960  G96 R14787 S800
                  N970  G00 X-64752 Z115000
                  N980  G01 X-65735 Z112017 F100
                  N990  Z73000
                  N1000 X-73210
                  N1010 G00 Z148000
Sixth Operation   O1020 G90 M05
                  N1030 G97 S400 M41
                  N1040 G70 M14
                  N1050 G00 X-73210 Z148000 T1400 M06
                  N1060 X-67805 Z103000
                  N1070 G91
                  N1080 G33 X-1500 Z-18668 K8333
                  N1090 G00 Z18613
                  N1100 X1600
```

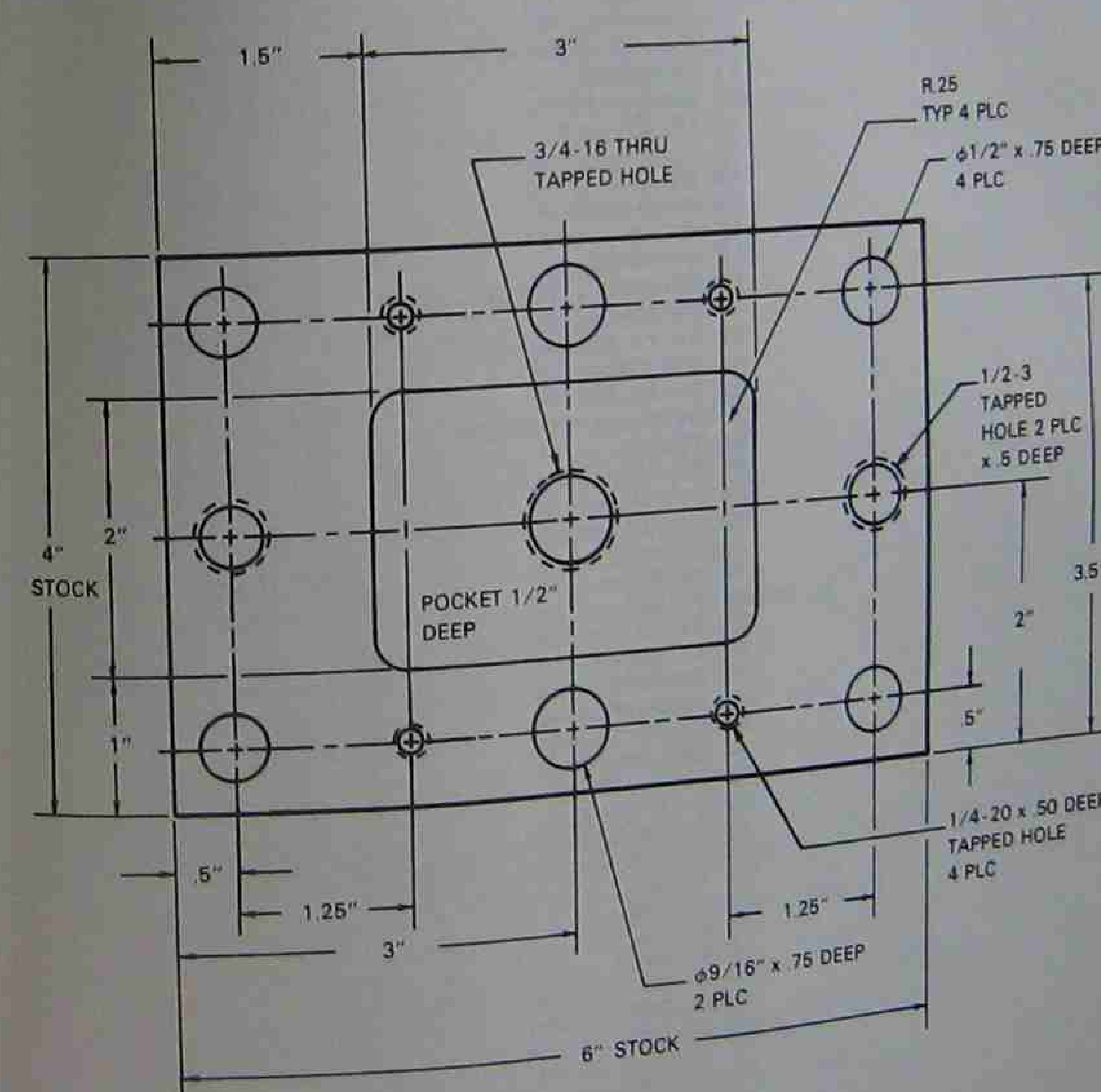


```
                  N1110 G33 X-1500 Z-18668 K8333
                  N1120 G00 Z18629
                  N1130 X1570
                  N1140 G33 X-1500 Z-18668 K8333
                  N1150 G00 Z18640
                  N1160 X1550
                  N1170 G33 X-1500 Z-18668 K8333
                  N1180 G00 Z18646
                  N1190 X1550
                  N1200 G33 X-1500 Z-18668 K8333
                  N1210 G00 Z18654
                  N1220 X1525
                  N1230 G33 X-1500 Z-18668 K8333
                  N1240 G00 Z18670
                  N1250 X1520
                  N1260 G33 X-1500 Z-18668 K8333
                  N1270 G00
                  N1280 G90
                  N1290 Z105000
                  N1300 X45000
Seventh Operation O1310 G90
                  N1320 G97 S780 M41
                  N1330 G70 M13
                  N1340 G00 X45000 Z105000 T0300 M06
                  N1350 G95
                  N1360 G92 S2500
                  N1370 G96 R45000 S800
                  N1380 G00 X19517 Z65000
                  N1390 G01 X22500 Z62017 F100
                  N1400 Z54373
                  N1410 X21550 Z52728
                  N1420 Z50000
                  N1430 X23125
                  N1440 Z43000
                  N1450 X24927
                  N1460 X27500 Z33399
                  N1470 Z21000
                  N1480 X29588
                  N1490 G03 X33900 Z16688 I29588 K16688
                  N1500 G01 Z15000
                  N1510 X37000
                  N1520 G00 X45000 Z65000
Eighth Operation  O1530 G90
                  N1540 G97 S300 M41
                  N1550 G70 M14
                  N1560 G00 X45000 Z65000 T0400 M06
                  N1570 X22300 Z66029
                  N1580 G33 Z50300 K10000
                  N1590 G00 X24500
                  N1600 Z65946
                  N1610 X22150
                  N1620 G33 Z50300 K10000
                  N1630 G00 X24500
                  N1640 Z65879
                  N1650 X22030
                  N1660 G33 Z50300 K10000
```

```
N1670 G00 X24500
N1680 Z65835
N1690 X21950
N1700 G33 Z50300 K10000
N1710 G00 X24500
N1720 Z65807
N1730 X21900
N1740 G33 Z50300 K10000
N1750 G00 X24500
N1760 Z65791
N1770 X21870
N1780 G33 Z50300 K10000
N1790 G00 X24500
N1800 Z65780
N1810 X21850
N1820 G33 Z50300 K10000
N1830 G00 X24500
N1840 X50000 Z85000
N1850 M30
```


SAMPLE MILLING PROGRAM

MAT: 103/1020 STEEL
6"x4" x 1" OR 1/2" THICK



Courtesy of Bayer Industries

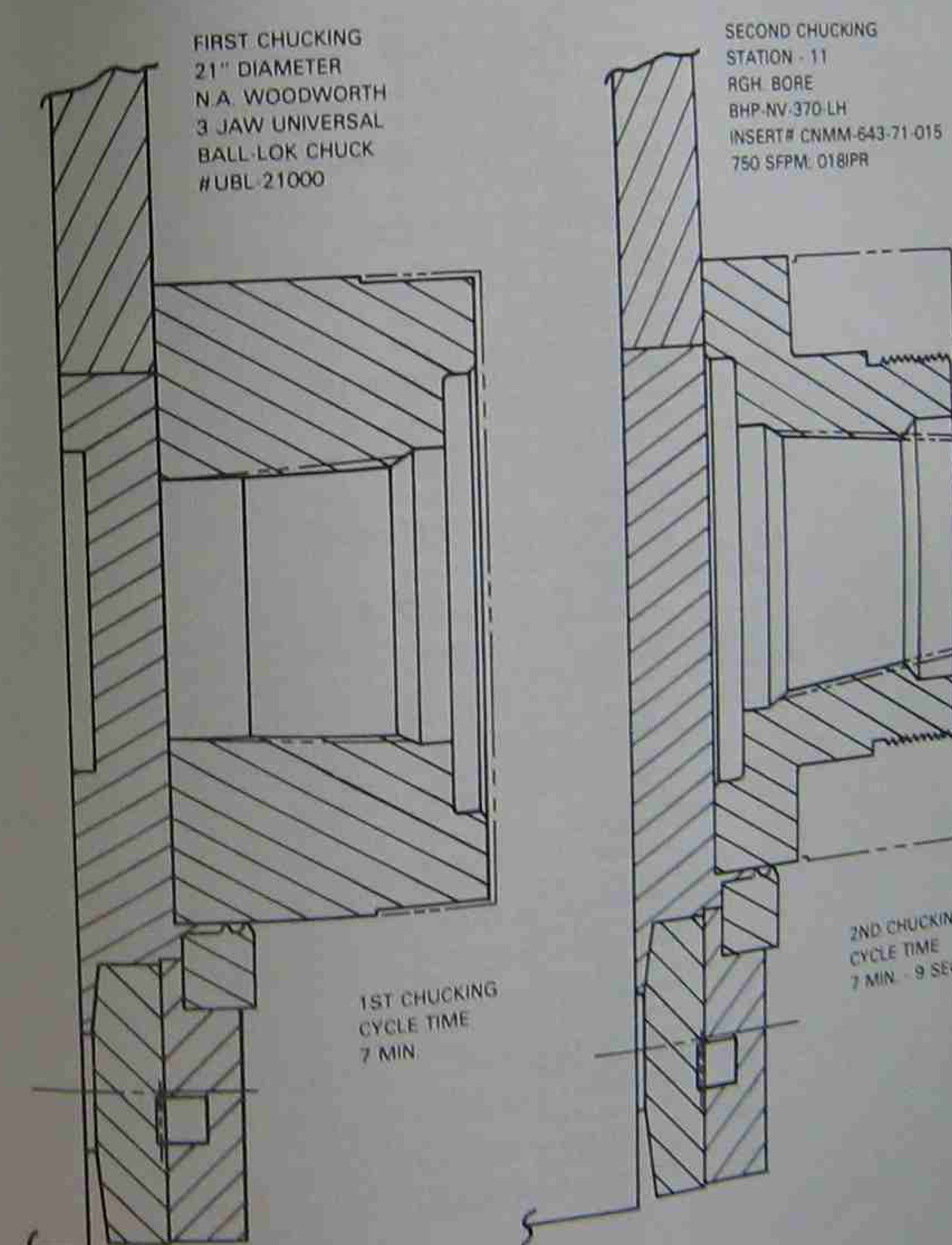
```
O0001
G20G40G49G80G90M03T01
G00X-1.65Y2.75S0637
G43Z0H01M08
G01Z-.15F50.
X6.25F22.93
Y1.25
X-1.65
G00X0H00T02
X-1.65Y2.75S1490
G43Z0H02
G01Z-.16F50
X6.25F14.9
Y1.25
X-1.65
G00Z0H00T03
X3.Y2.S0407
G43Z0H03
G81G99Z-.725R0F6.11
G80Z0H00T04
X3.Y2.S0637
G43Z0H04
G01Z-.59F10.
G41X4.115F6.37D24
Y2.65
X2.385
Y1.35
X4.49
Y2.99
X1.51
Y1.01
X4.49
Z-.58F10.
G00G40X3.Y2.
Z0H00T05
X3.Y2.S1192
G43Z05H05
G01Z-.58F15.
Z-.6F11.92
G41X3.375D25
Y2.325
X2.625
Y1.675
X3.75
Y2.625
X2.25
Y1.375
X4.125
Y2.85
X1.875
Y1.15
X4.5
Y3
X1.5
```



```
Y1.
X4.5
Z-.59F15.
G00G40X3.Y2.
Z0H00T06
X3.Y2.S0444
G43Z0H06
G81G98Z-1.25R-.5F6.66
G80Z0H00T07
X3.Y2.S0162
G43Z0H07
G84G99Z-1.35R-.4F9.62
G80Z0H00T08
X.5Y.5S1222
G43Z0H08
G81G99Z-.25R0F6.11
X1.75
X3.
X4.25
X5.5
Y2.
Y3.5
X4.25
X3.
X1.75
X.5
Y2.
G80Z0H00T09
X.5Y.5S0611
G43Z0H009
G81G99Z-.75R0F7.33
Y3.5
X5.5
Y.5
G80Z0H00T10
X3.Y.5S0543
G43Z0H10
G81G99Z-.77R0F6.52
Y3.5
G80Z0H00T11
X1.75Y3.5S1520
G43Z0H11
G83G99Z-.66R0Q.25F9.12
X4.25
Y.5
X1.75
G80Z0H00T12
X1.75Y.5S0326
G43Z.2H12
G84G99Z-.4R.2F15.49
X4.25
Y3.5
X1.75
G80X0H00T13
```

```
109 X.5Y2.S0724
110 G43Z0H13
111 G83G99Z-.73R0Q.375F8.69
112 X5.5
113 G80Z0H00T14
114 X5.5Y2.S0255
115 G43Z.2H14
116 G84G99Z-.45R.2F18.63
117 X.5
118 G80Z0H00T15
119 X.5Y2.S0795
120 G43Z0H15
121 G81G99Z-.375R0F12.5
122 Y3.5
123 X1.75Z-.23
124 X3.Z-.4
125 X4.25Z-.23
126 X5.5Z-.375
127 Y2.
128 Y.5
129 X4.25Z-.23
130 X3.Z-.4
131 X1.75Z-.23
132 X.5Z-.375
133  G80Z0H00M09
134 X-4.Y6.M05
135 M30
```


SAMPLE LATHE PROGRAM



Courtesy of Lodge & Shipley Co.

```
N0010G70M12
N002G9T0202
N003G92X15.Z22.782
N004G97S065M03
N0050G95
N0060G21X0.Z5.F.8
N007G01Z-00.1F.007M08
N008G21Z07.782F.8M09
N009X11.Z07.782
N01G9T0101M12
N011G92X06.7969Z20.2039S0712
N012G97S021M04
N0130G95
N014G21X05.2Z05.182F.8
N0150G96R5.1S06
N0160G01X1.7F.014M08
N0170Z5.212
N018G21X04.95F.8
N0190G01Z5.182F.022
N02G03X05.057Z05.075K00.107
N0210G01Z3.307
N022X05.15
N023G21X06.7969Z20.2039F.8M09
N024G9T0303
N0250M13
N026G92X12.4531Z12.5789S15
N027G97S071M04
N0280G95
N0290G21X1.9485Z5.295F.8
N0300G96R2.S075
N0310G01Z-.1F.018M08
N03540X1.8485
N0350G21Z5.272F.8
N0360X2.2509
N0370G01Z4.047F.018
N0380X2.1058Z3.7366
N0382X2.0293Z1.2875
N0384Z-.1
N0386X1.9293
N0390G21Z5.272F.8
N0400X2.4485
N0410G01Z4.557F.018
N0420X2.1985
N0430G21Z5.272F.8
N0440X2.6985
N0450G01Z4.557F.018
N0460X2.4485
N0470G21Z5.272F.8
N0480X2.9485
N0490G01Z4.557F.018
N0500X2.4485
N0510G21Z5.272F.8
N0520X3.1985
N0530G01Z4.55F.018
```



```
N0540X2.9485
N0550G21Z5.272F.8
N0560X3.433
N0570G01Z4.635F.018
N575G03X03.355Z04.557I00.78
N0580G01X3.1985
N059G21Z07.272F.8
N0592T0606
N0593G04X02.
N0595G92X03.1985Z07.272
N06G21X03.5406Z05.272
N0610G01Z5.172F.010
N0620X3.453Z5.0844
N0630Z4.625
N064G03X03.375Z04.547I00.78
N0642G01X02.3209
N065X02.2609Z04.487
N0651Z04.047
N0652X02.0958Z03.7266
N0653X02.0458
N0660G97S0350M03
N069G21X01.4531Z07.272F.8M09
N0692G9T101
N0693G92X.967Z10.1941
N0694G21X2.0764Z5.5F.8
N0695G21X2.237M08
N0696G33X2.1209Z1.75I0039K.125
N0697G33X1.9597Z1.5888I.125K.125
N0698G21Z5.5F.8
N0699X2.2476
N0700G33X2.1309Z1.75I.0039K.125
N0701G33X1.9597Z1.5888I.125K.125
N0702G21Z8.F.8
N0703X11.967Z15.501
N0705G97S054M04
N071G9T0909M82
N0720
N073G92X06.781Z20.188S15
N074G97S054M04
N0750G95
N0760G21X3.48Z5.256F.8
N0770G96R3.5S1
N0780G01Z5.156F.008M08
N079X04.93
N08G03X05.021Z05.065K00.091
N0810G01Z3.281
N0820X5.2
N0825G97S038
N083G21X10.781Z35.188F.8M09
N0840T0000
N0845
N0850M00
N0855M12
N086G9T1101
```

```
N087G92X10.7969Z35.2039S0712
N088G97S021M004
N09G21X05.2Z05.057F.8
G0910G96R5.1S06
N092G01X1.8F.014M08
N0930Z5.147
N0940G21X4.567F.8
N0945M11
N0960G01Z1.807F.022
N097X04.95
N098G03X05.057Z01.7K00.107
N099G01X5.147F.8
N1G21Z05.147F.8
N1005M12
N1010X4.067
N1020G01Z1.807F.022
N1030X4.6
N1040G21Z5.147F.8
N0150X3.567
N1060G01Z1.807F.022
N1070X4.1
N1080G21Z5.147F.8
N0190X3.1644
N1100G01Z5.057F.022
N1110X3.317Z4.9094
N1120Z3.307
N1130X3.442
N1140Z1.885
N115G02X03.52Z01.1807I00.078
N1160G01X5.
N1170G97S023
N118G21X06.7969Z20.2039F.8M09
N119G9T2111
N1205M13
N121G92X12.4531Z16.0789S15
N122G97S07M04
N1230G95
N1240G21X2.3241Z5.147F.8
N125G96R2.3S075
N126G01Z04.422F.018M08
N1270X1.9465Z4.0444F.01
N1280G21Z5.147F.8
N1330X2.433
N1340G01Z4.432F.018
N1350X2.3
N136G21Z07.147F.8
N1362T0808
N1363G04X02.
N1365G92X02.3Z07.077
N137G21X02.56Z05.147
N1380G01Z5.04F.01
N139G02X02.453Z04.94K00.107
N1400G01Z4.422
N141X02.2741
```

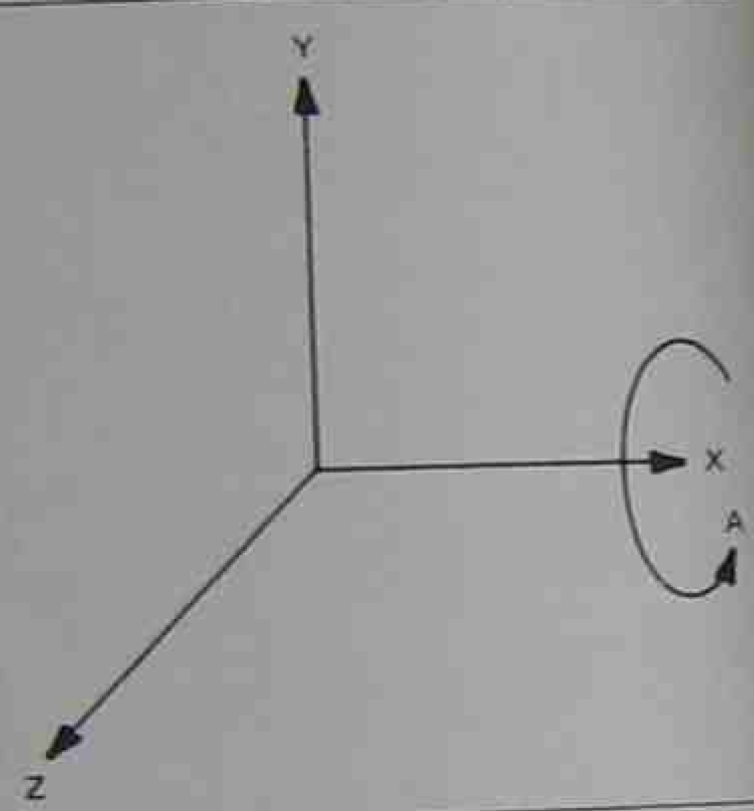


```
N1420G21Z5.24F.8
N1425G97S076
N143X12.4531Z16.0089M09
N144G9T1909
N145G92Z06.781Z20.188S15
N146G97S076M04
N1470G95
N1480G21X2.48Z5.131F.8
N1490G96R2.5S1
N1500G01Z5.031F.01M08
N1510X3.2029
N1520X3.281Z4.9529
N1530Z3.281
N1540X3.345
N155G03X03.406Z03.22K00.061
N1560G01Z1.875
N157G02X03.5Z01.781I00.94
N158G01X04.93
N159G03X05.021Z01.69K00.91
N1600G01Z1.59
N1610X5.031
N1615G97S029
N162G21X06.781Z20.188F.8M09
N1621G9T0707
N1622G92X06.75Z20.728
N1623G97S035M04
N1624G95
N1625G21X03.4Z03.25F.8
N1626G96R03.4S06M04
N1627G01X03.125F.006M08
N1628G21X03.35F.8M8
N1629Z03.43
N163G01X03.17Z03.25F.006
N1631G21X03.4F.8
N1632G21X06.75Z10.728M09
N1634G9T0404
N164G92X06.75Z20.157
N1645M12
N1650G97S0290M03
N1660G95
N1670G21X3.35Z5.5F.8
N1600M08
N1690G83X-.122Z-.007H01
N1700G33Z3.375K.0625
N1710G84X-.007Z-.004H01
N1720G84X-.0055Z-.0032H01
N1730G84X-.0046Z-.0027H01
N1740G84X-.0041Z-.0024H01
N1750G84X-.0038Z-.0022H01
N1760G84X-.0034Z-.002H01
N1770G84X-.0Z-.0H01
N178G21X10.75Z35.157F.8M09
N1790T0000
N1795
N1800M30
```

GLOSSARY

TERM AND DEFINITION | EXAMPLE

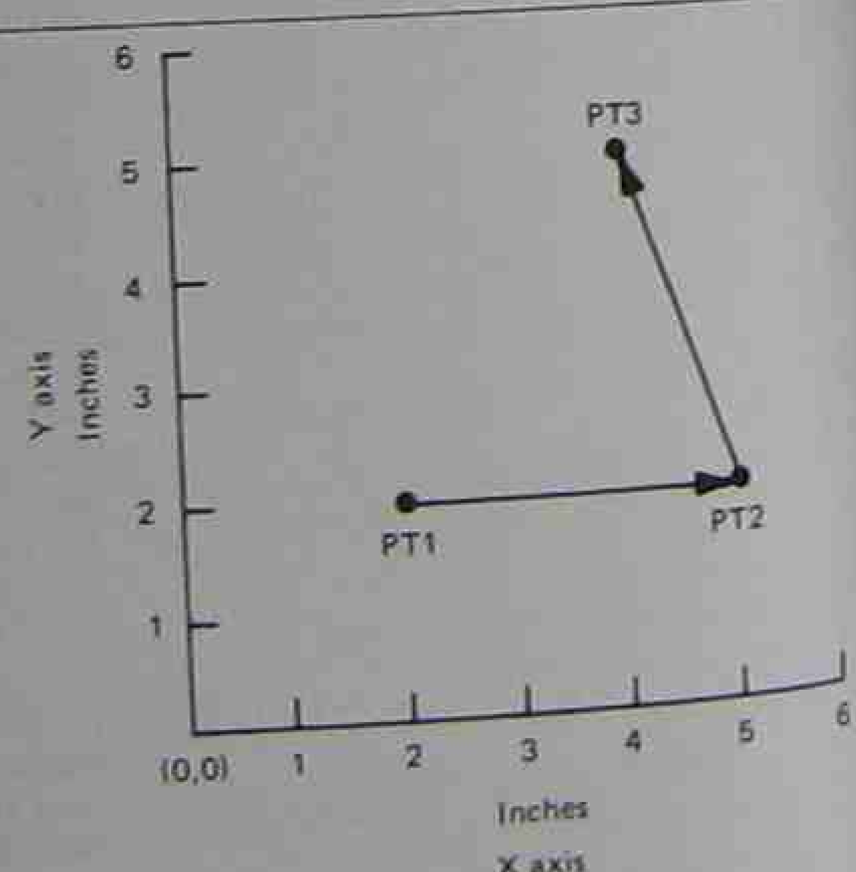
A AXIS The axis of circular motion of a machine tool member or slide about the X axis. (Usually called alpha.)



ABSOLUTE ACCURACY Accuracy as measured from a reference which must be specified.

ABSOLUTE READOUT A display of the true slide position as derived from the position commands within the control system.

ABSOLUTE SYSTEM A numerical control system in which all positional dimensions, both input and feedback, are given with respect to a common datum point. The alternative is the incremental system.

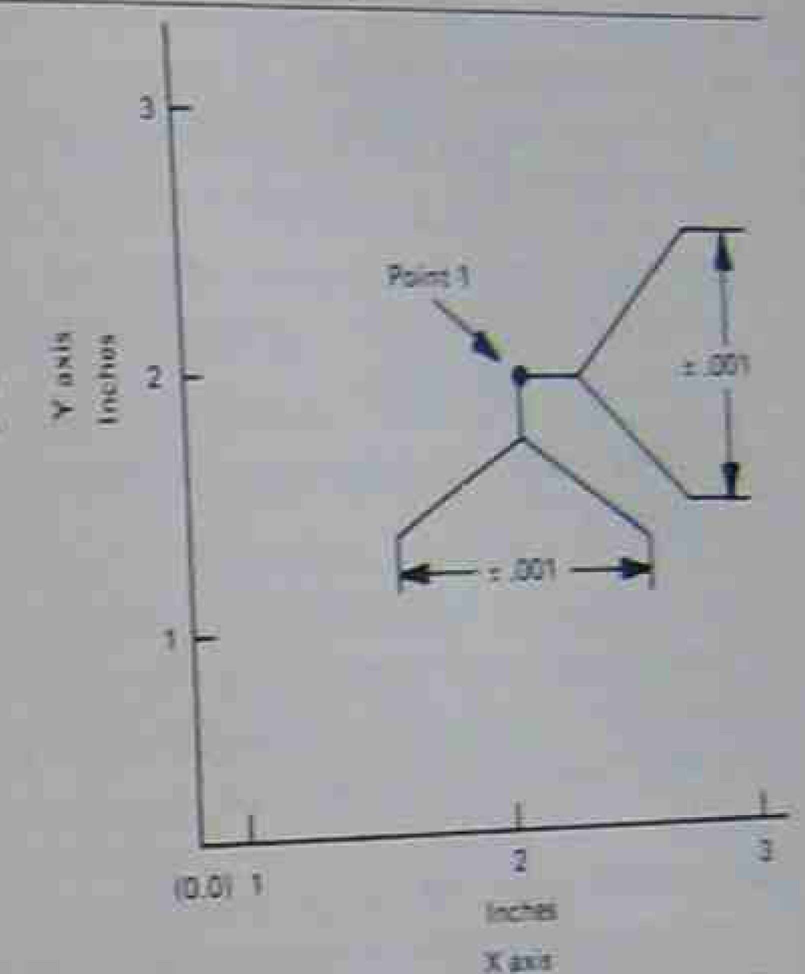


Coordinate Positions

Point	X value	Y value
PT1	2	2
PT2	5	2
PT3	4	5

In an absolute system, all points are relative to (0,0), and the absolute coordinates for each of the required points are programmed with respect to (0,0).

ACCURACY 1. Measured by the difference between the actual position of the machine slide and the position demanded. 2. Conformity of an indicated value to a true value, i.e., an actual or an accepted standard value. The accuracy of a control system is expressed as the deviation (the difference between the ultimately controlled variable and its ideal value), usually in the steady state or at sampled instants.



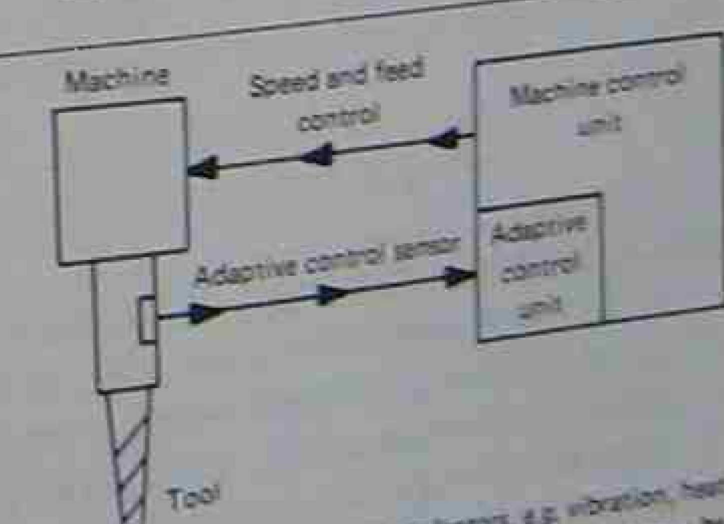
The position of point 1 in this example is X = 2 and Y = 2. If the machine accuracy is specified as ± .001, the X axis movement could be between X = 1.999 and X = 2.001. The Y axis movement could be between Y = 1.999 and Y = 2.001.

AD-APT An Air Force adaptation of APT program language with limited vocabulary. It can be used on some small to medium sizes of U.S. computers for NC programming.

C1 = circle/center, PT1, radius, 2.5

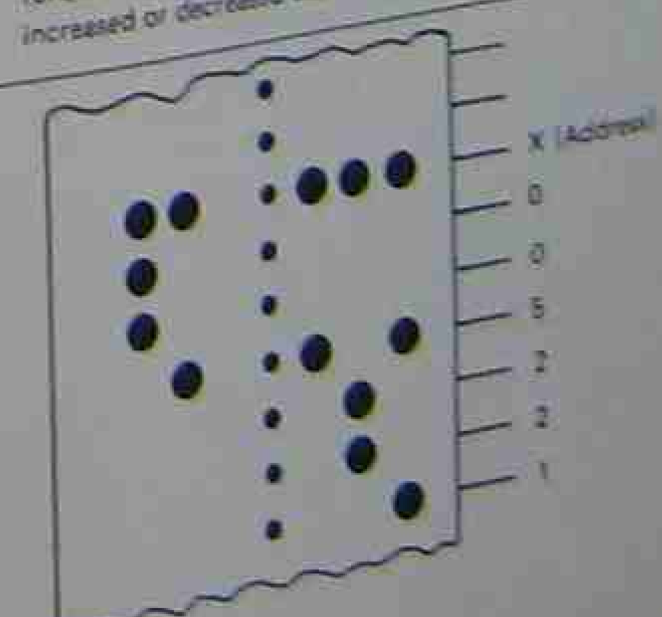
Similar to the APT language except it does not possess the advanced contouring capabilities of APT.

ADAPTIVE CONTROL A technique which automatically adjusts feeds and/or speeds to an optimum by sensing cutting conditions and acting upon them.



Sensors may measure variable factors, e.g. vibration, heat, torque, and deflection. Cutting speeds and feeds may be increased or decreased depending on conditions sensed.

ADDRESS 1. A symbol indicating the significance of the information immediately following. 2. A means of identifying information or a location in a control system. 3. A number which identifies one location in memory.



ALPHANUMERIC CODING A system in which the characters are letters A through Z and numerals 0 through 9.

APT and AD-APT statements use alphanumeric coding, e.g. GOFWD, CT12/PAST, 2, INTOF, L13

ANALOG 1. Applies to a system which uses electrical voltage magnitudes or ratios to represent physical axis positions. 2. Pertains to information which can have continuously variable values.

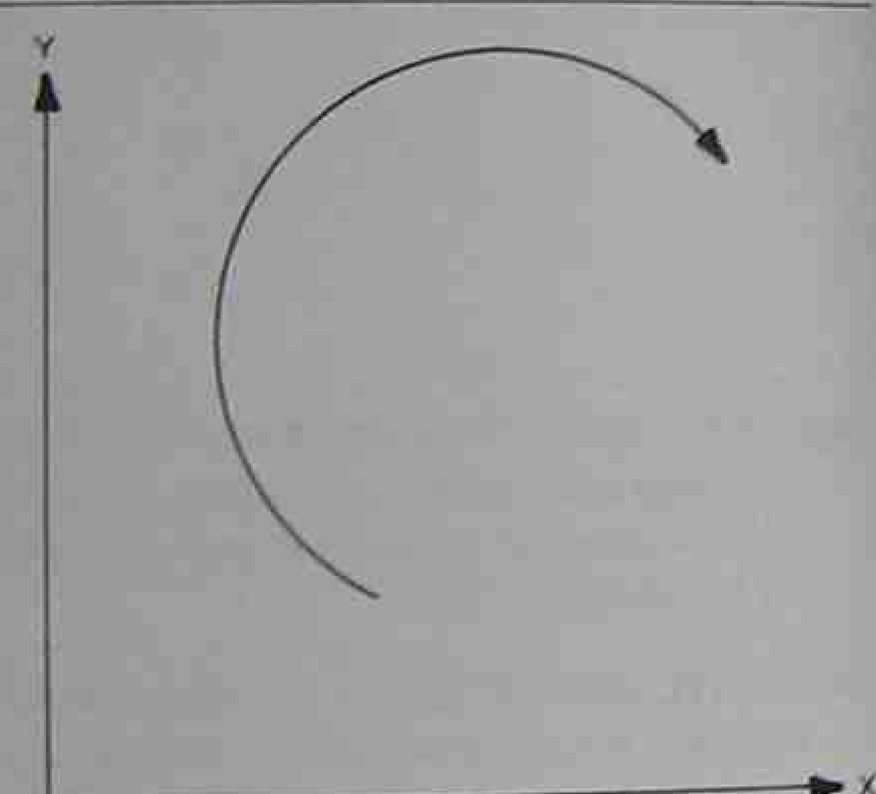
ANALYST A person skilled in the definition and development of techniques to solve problems.

APT (Automatic Programmed Tool) A universal computer-assisted program system for multiaxis contouring programming. APT III provides for five axes of machine tool motion.

Typical APT geometry definition statement:
C1 = CIRCLE/XLARGE, L12, XLARGE, L13, RADIUS, 3.5

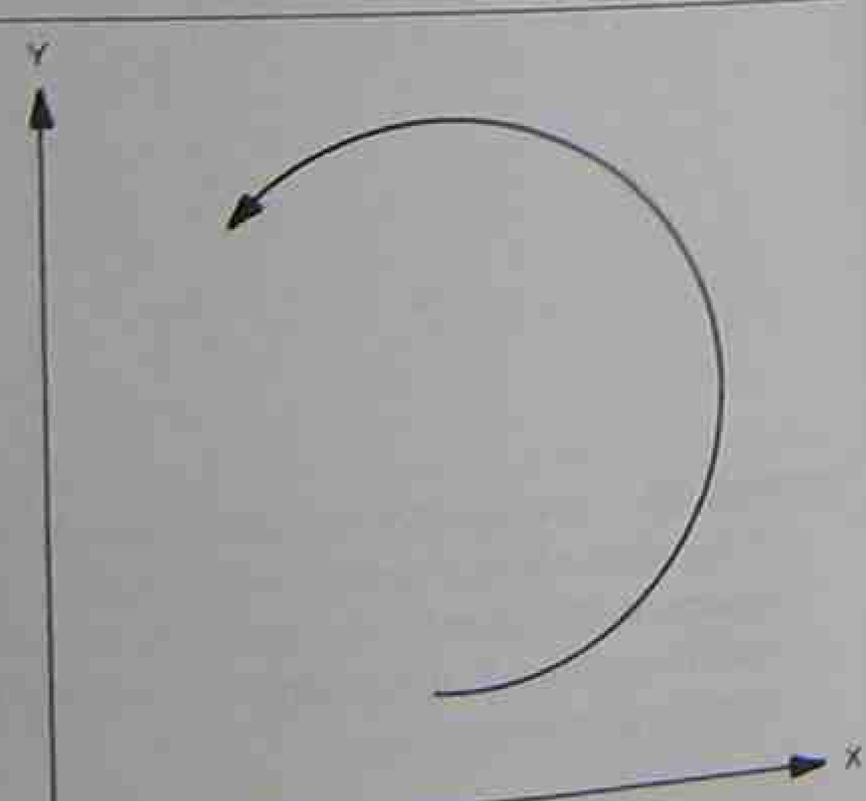
Typical APT tool motion statement:
TLRGT, GORGT/AL3, PAST, AL12

ARC CLOCKWISE An arc generated by the coordinated motion of two axes, in which curvature of the tool path with respect to the workpiece is clockwise, when viewing the plane of motion from the positive direction of the perpendicular axis.



Activated by G02 preparatory function

ARC COUNTERCLOCKWISE An arc generated by the coordinated motion of two axes, in which curvature of the tool path with respect to the workpiece is counterclockwise, when viewing the plane of motion from the positive direction of the perpendicular axis.



Activated by G03 preparatory function

ASCII (American Standard Code for Information Interchange) A data transmission code which has been established as an American standard by the American Standards Association. It is a code in which seven bits are used to represent each character. Formerly USASCII.

AUTO-MAP An abbreviation for AUTOmatic MAchining Programming. A computer-aided programming language which is a subset of APT. It is used for simple contouring and straight line programming.

AUTOMATION 1. The implementation of processes by automatic means. 2. The investigation, design, development, and application of methods to render processes automatic, self-moving, or self-controlling.

AUTOSPOT (Automatic System for Positioning of Tools) A computer-assigned program for NC positioning and straight-cut systems, developed in the U.S. by the IBM Space Guidance Center. It is maintained and taught by IBM.

AUX CODE Auxiliary function command in Machinist Shop Language. Used to control specific functions within a CNC program.

AUXILIARY FUNCTION A programmable function of a machine other than the control of the coordinate movements or cutter.

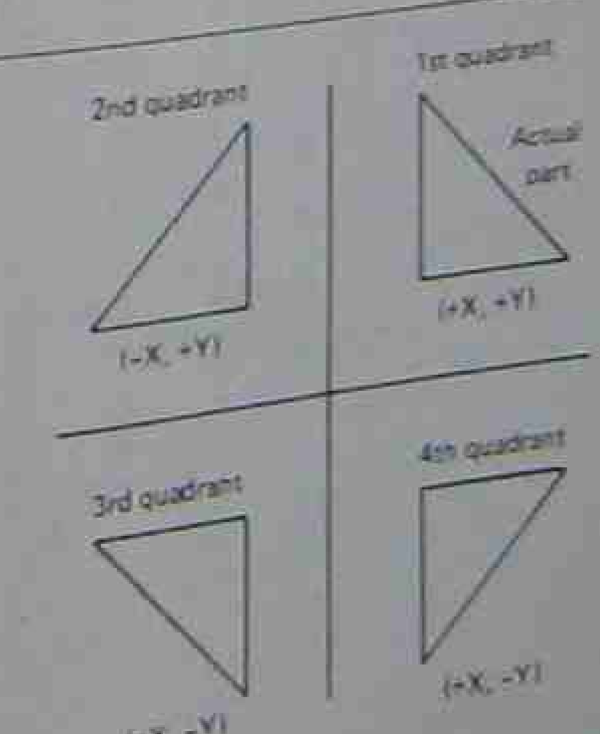
- Transferring a tool to the select tool position.
- Turning coolant ON or OFF.
- Starting or stopping the spindle.
- Initiating pallet shuttle or movement.

AXIS A principal direction along which the relative movements of the tool or workpiece occur. There are usually three linear axes, mutually at right angles, designated as X, Y, and Z.

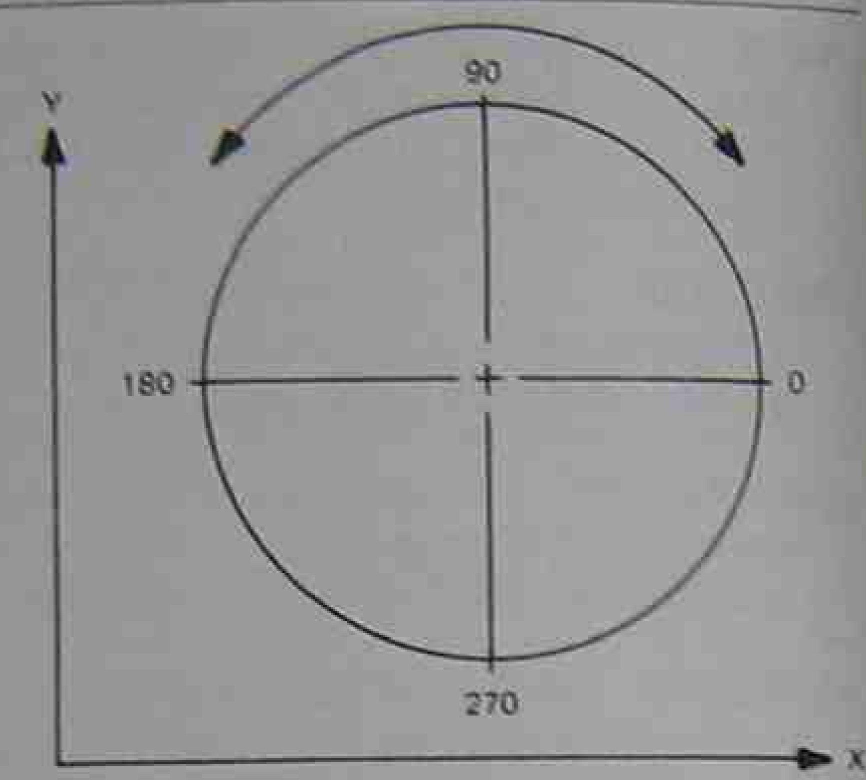
AXIS INHIBIT A feature of an NC unit which enables the operator to withhold command information from a machine tool slide.

AXIS INTERCHANGE The capability of inputting the information concerning one axis into the storage of another axis.

AXIS INVERSION The reversal of plus and minus values along an axis. This allows the machining of a left-handed part from right-handed programming or vice versa.

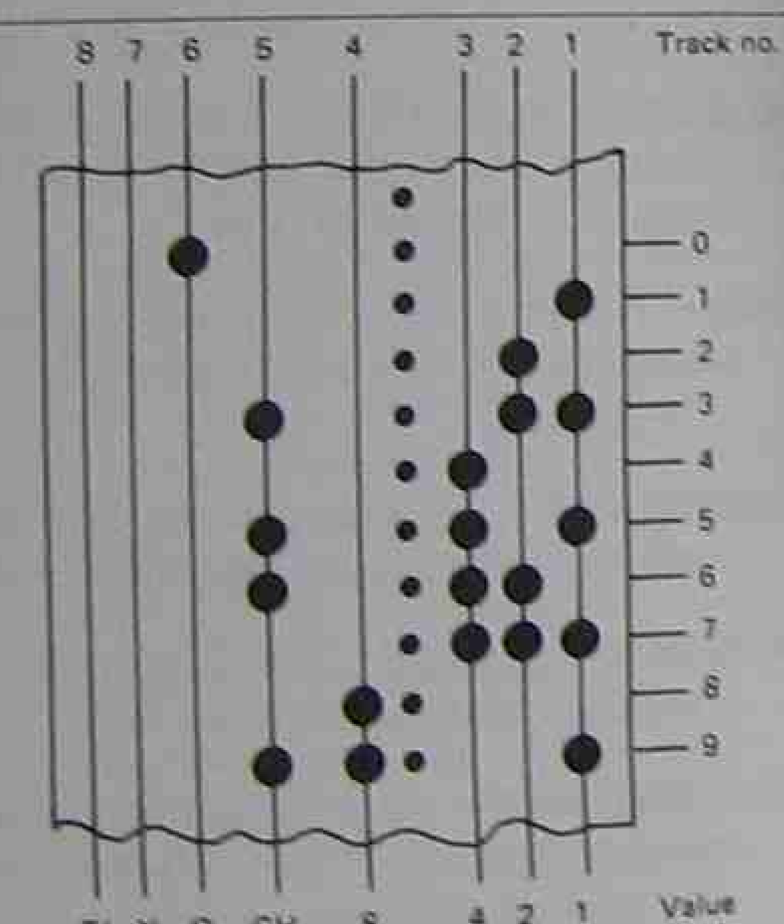


B (BETA) AXIS The axis of circular motion of a machine tool member or slide about the Y axis.



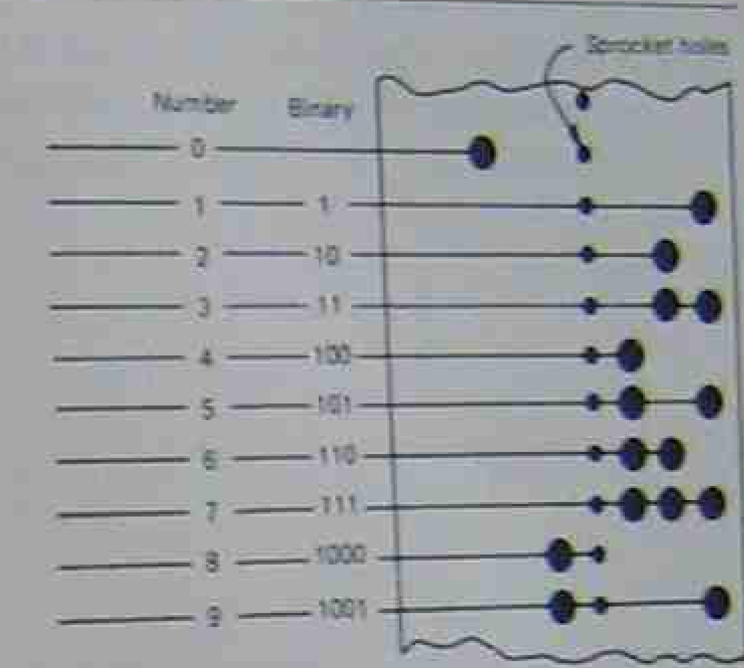
BACKLASH A relative movement between interacting mechanical parts as a result of looseness.

BCD (Binary-coded decimal) A system of number representation in which each decimal digit is represented by a group of binary digits forming a character.



Numbers and letters are expressed by punched holes across the tape for the code or value desired.

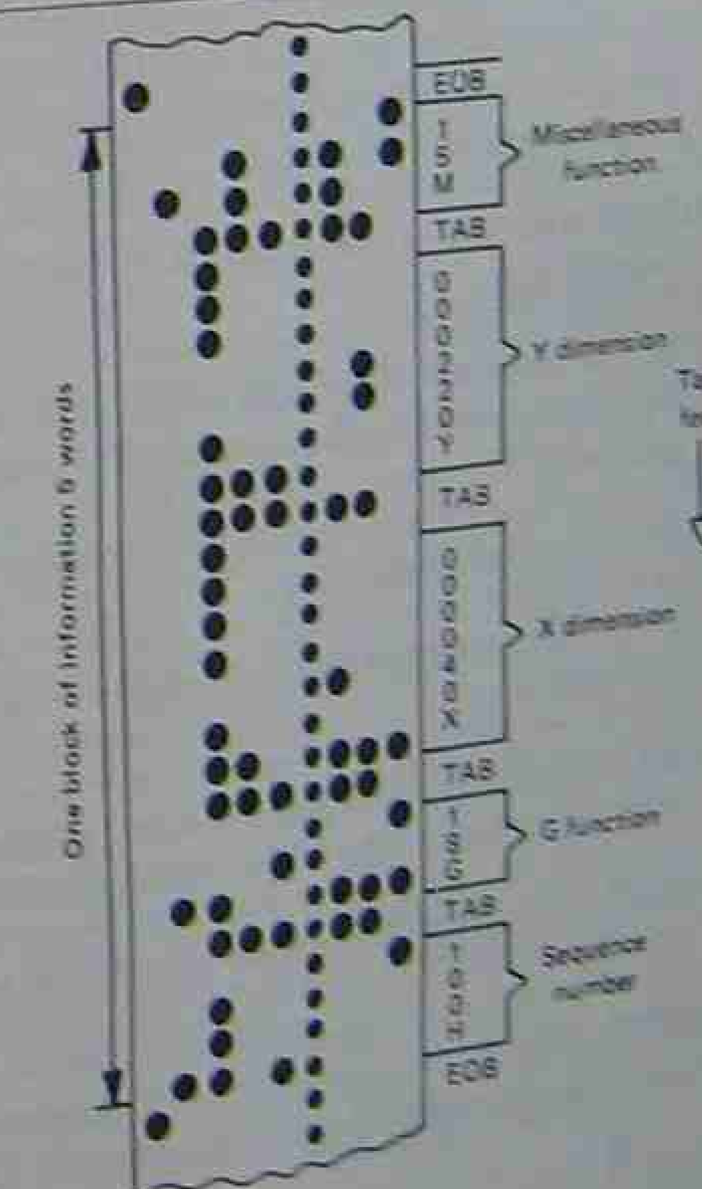
BINARY CODE Based on binary numbers, which are expressed as either 1 or 0, true or false, on or off.



Most computers operate on some form of binary system where a number or letter can be expressed as ON (a hole) or OFF (no hole).

BIT (Binary digit) 1. Binary digit having only two possible states. 2. A single character of a language using exactly two distinct kinds of characters. 3. A magnetized spot on any storage device.

BLOCK A word, or group of words, considered as a unit. A block is separated from other units by an end of block character. On punched tape, a block of data provides sufficient information for an operation.



BLOCK DELETE Permits selected blocks of tape to be ignored by the control system, at the operator's discretion with permission of the programmer.

This feature allows certain blocks of information to be skipped by programming a slash (/) code in front of the block to be skipped. One lot of parts with holes 1, 2, and 3 are required. On another lot, only holes 1 and 3 are required. The same tape could be used for both lots by activating the block delete switch on the second lot and eliminating hole 2. The (/) code would be in front of the block of information for hole 2.

BUFFER STORAGE A place for storing information in a control system or computer for planned use. Information from the buffer storage section of a control system can be transferred almost instantly to active storage (that portion of the control system commanding the operation at the particular time). Buffer storage allows a control system to act immediately on stored information rather than wait for the information to be read into the machine from the tape reader.

BUG 1. A mistake or malfunction. 2. An integrated circuit (slang).

BYTE A sequence of adjacent binary digits usually operated on as a unit and shorter than a computer word.

Eight bits equal one byte. A computer word usually consists of either sixteen or thirty-two bits (two or four bytes).

CAD Computer-aided design

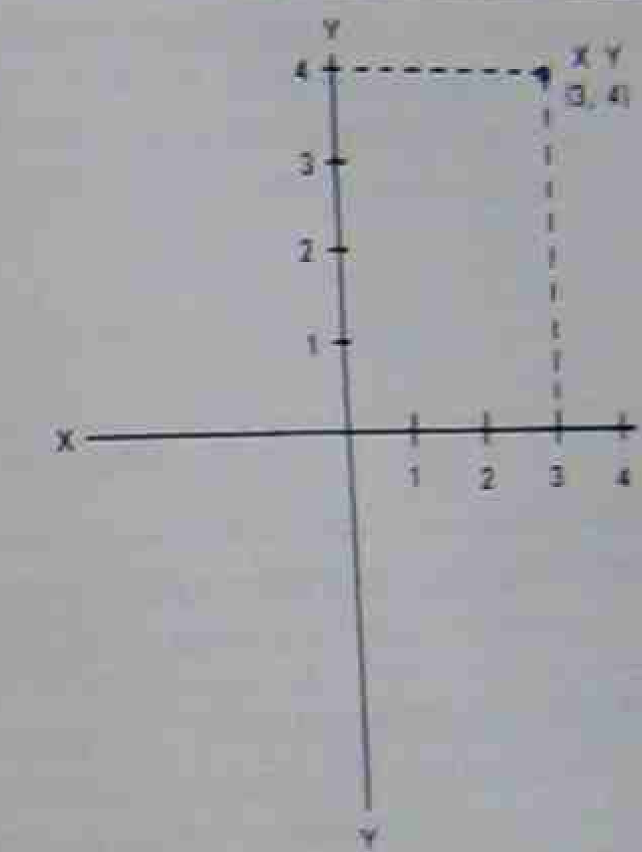
CAM (Computer-aided manufacturing) The use of computers to assist in phases of manufacturing.

CAM-I (Computer Aided Manufacturing International) The outgrowth and replacement organization of the APT Long Range Program.

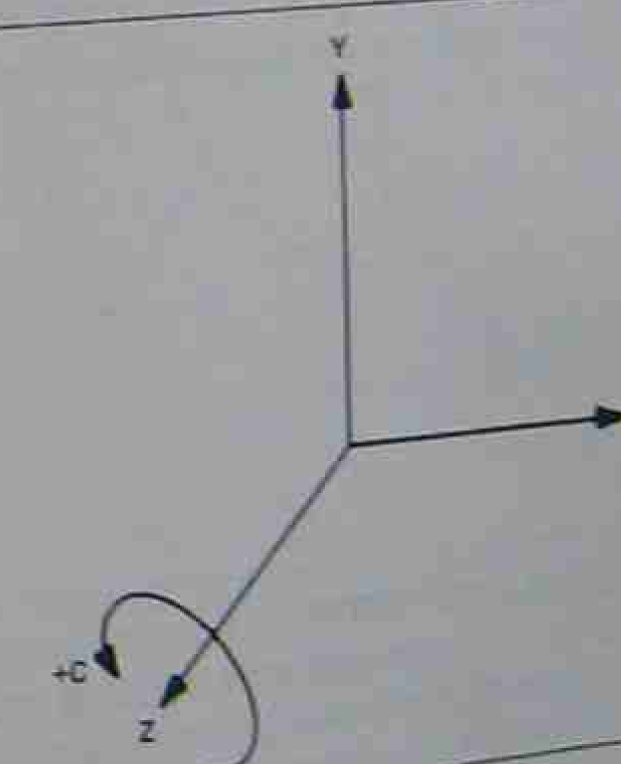
CANCEL A command which will discontinue any canned cycles or sequence commands.

CANNED CYCLE A preset sequence of events initiated by a single command. For example, code G84 will perform tap cycle by NC.

CARTESIAN COORDINATES A means whereby the position of a point can be defined with reference to a set of axes at right angles to each other.

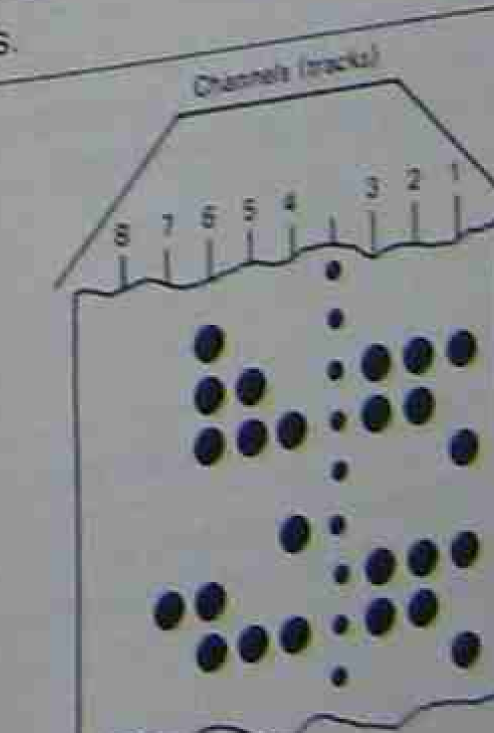


C AXIS Normally the axis of circular motion of a machine tool member or slide about the Z axis.

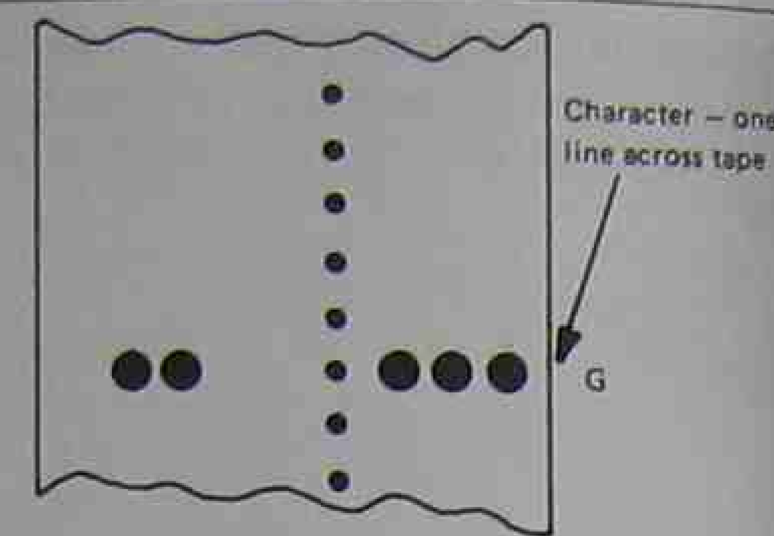


CHAD Pieces of material removed in card or tape operations.

CHANNELS Paths parallel to the edge of the tape along which information may be stored by the presence or absence of holes or magnetized areas. This term is also known as level or track. The EIA standard one-inch-wide tape has eight channels.

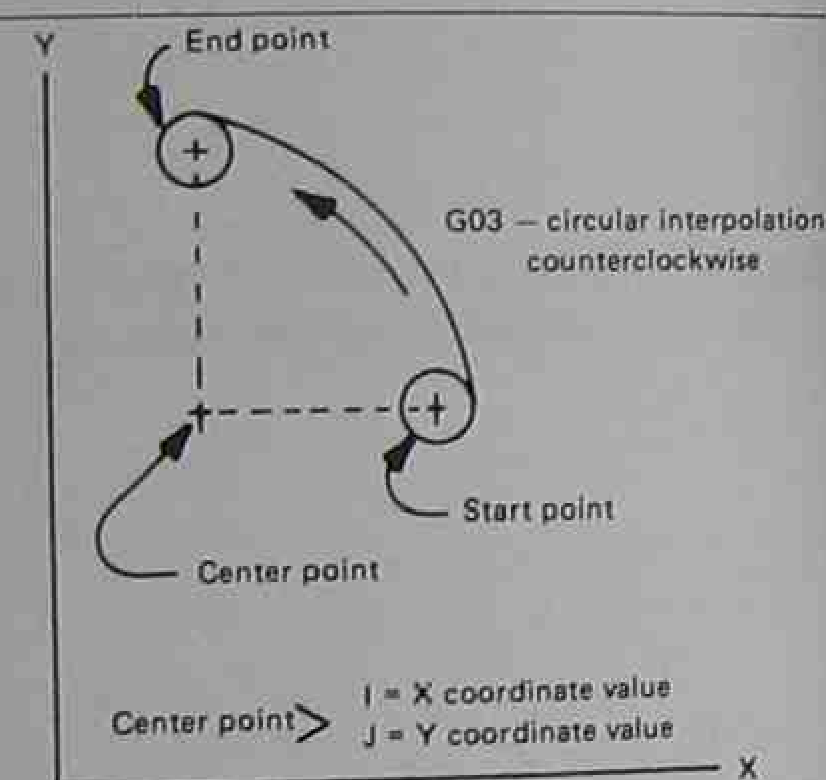


CHARACTERS A general term for all symbols, such as alphabetic letters, numerals, and punctuation marks. It is also the coded representation of such symbols.

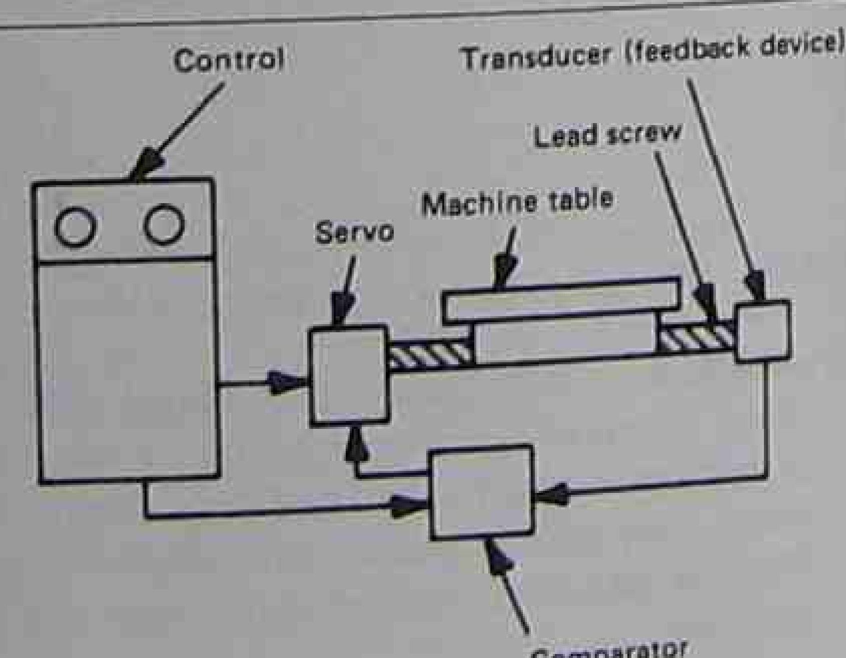


CHIP A single piece of silicon cut from a slice by scribing and breaking. It can contain one or more circuits but is packaged as a unit.

CIRCULAR INTERPOLATION 1. Capability of generating up to 360 degrees of arc using only one block of information as defined by EIA. 2. A mode of contouring control which uses the information contained in a single block to produce an arc of a circle.



CLOSED-LOOP SYSTEM A system in which the output, or some result of the output, is measured and fed back for comparison with the input. In an NC system, the output is the position of the table or head; the input is the tape information which ordinarily differs from the output. This difference is measured and results in a machine movement to reduce and eliminate the variance.



CNC Computer numerical control

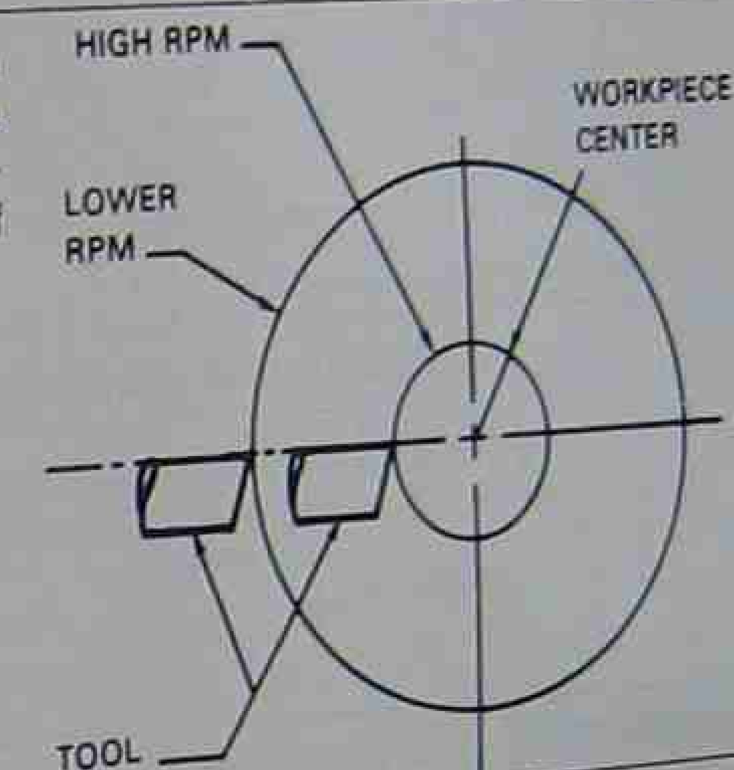
CODE A system describing the formation of characters on a tape for representing information, in a language that can be understood and handled by the control system.

COMMAND A signal, or series of signals, initiating one step in the execution of a program.

COMMAND READOUT A display of the slide position as commanded from the control system.

COMPUTER NUMERICAL CONTROL A numerical control system utilizing an on-board computer as an MCU.

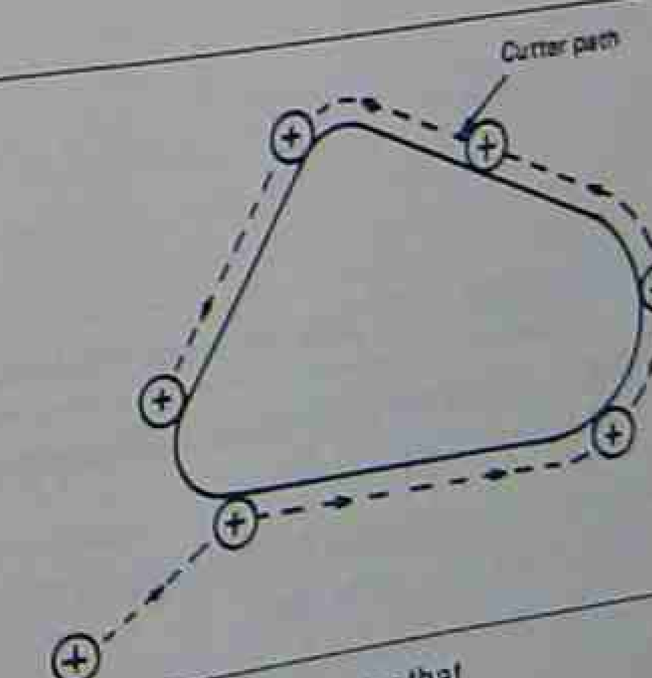
CONSTANT CUTTING SPEED The condition achieved by varying the speed of rotation of the workpiece relative to the tool, inversely proportional to the distance of the tool from the center of rotation.



See *contouring control system.*

CONTINUOUS-PATH OPERATION An operation in which rate and direction of relative movement of machine members is under continuous numerical control. There is no pause for data reading.

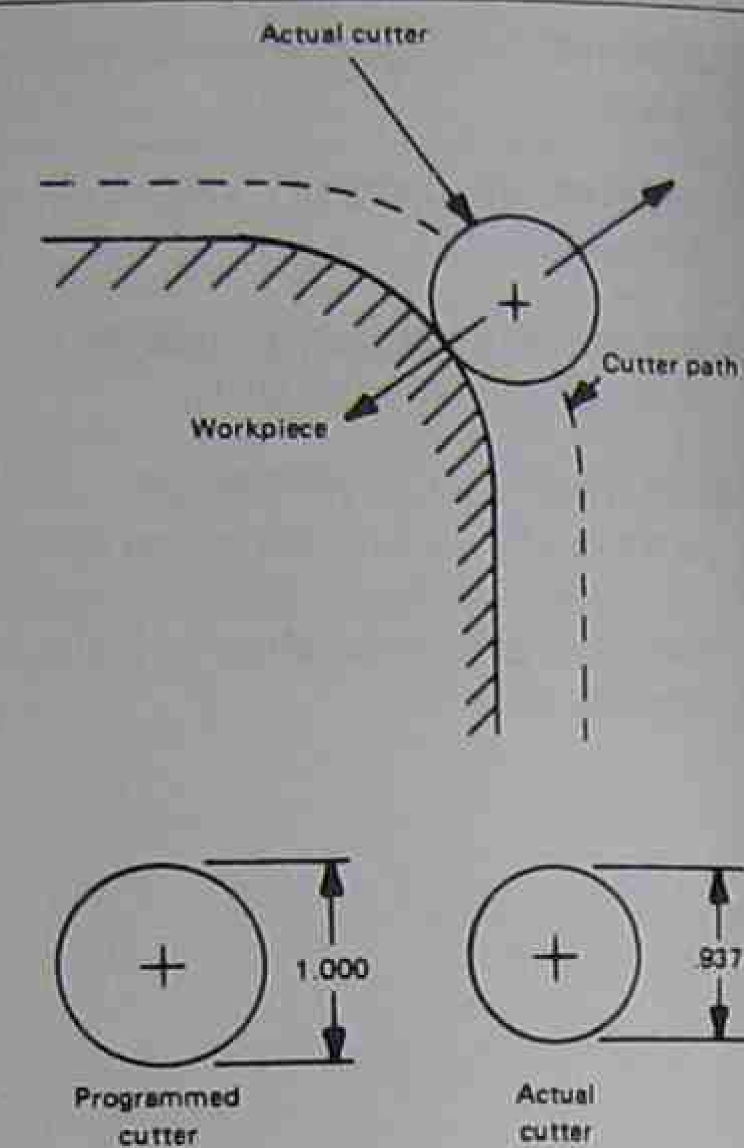
CONTOURING CONTROL SYSTEM An NC system for controlling a machine (e.g., milling, drafting) in a path resulting from the coordinated, simultaneous motion of two or more axes.



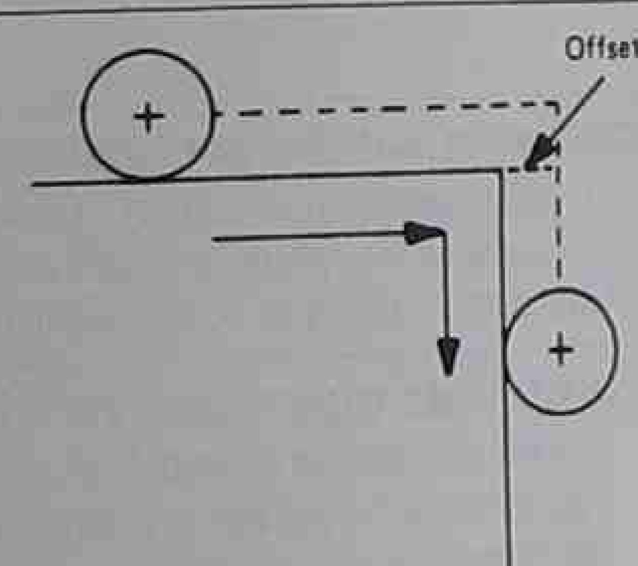
CPU Central processing unit of a computer. The memory or logic of a computer that includes overall circuits, processing, and execution of instructions.

CRT (Cathode Ray Tube) A device that represents data (alphanumeric or graphic) form by means of a controlled electron beam directed against a fluorescent coating in the tube.

CUTTER DIAMETER COMPENSATION A system in which the programmed path may be altered to allow for the difference between actual and programmed cutter diameters.



CUTTER OFFSET The distance from the part surface to the axial center of a cutter.

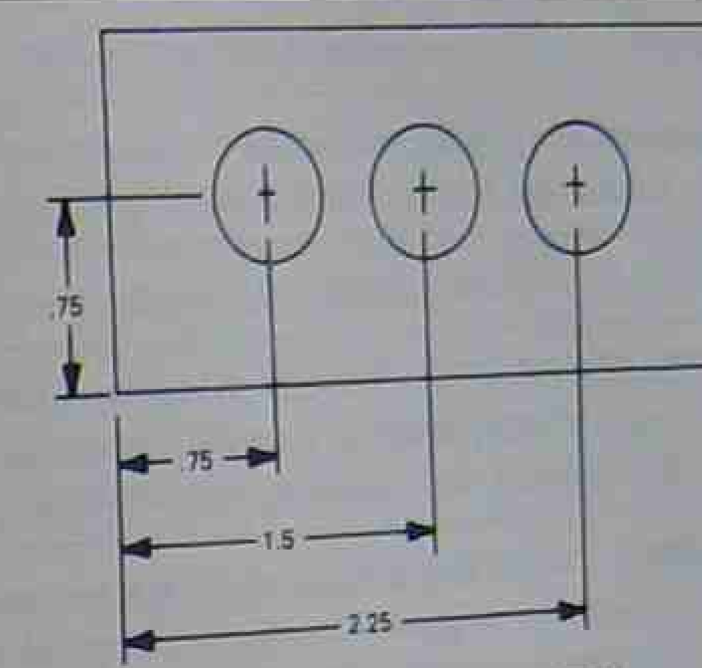


CUTTER PATH The path defined by the center of a cutter.

CYCLE 1. A sequence of operations that is repeated regularly. 2. The time it takes for one such sequence to occur.

DATA A representation of information in the form of words, symbols, numbers, letters, characters, digits, etc.

DATUM DIMENSIONING A system of dimensioning based on a common starting point.

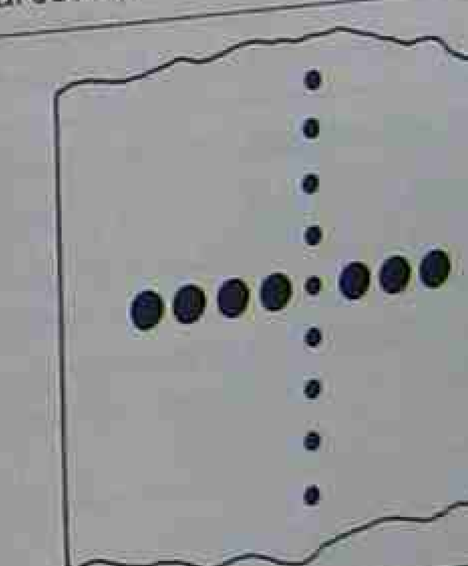


(Also known as absolute dimensioning)

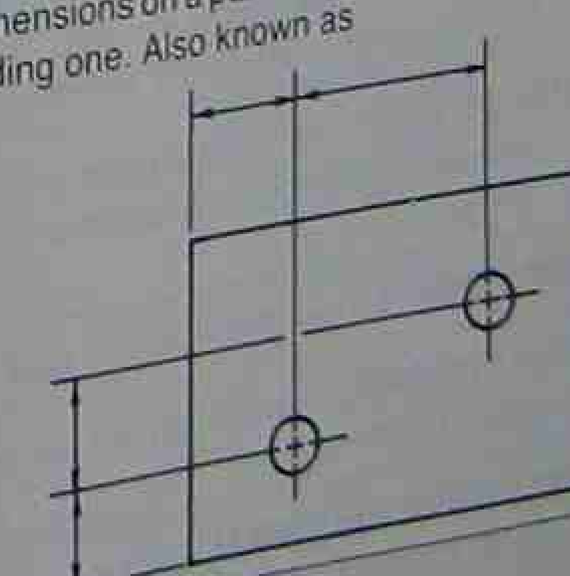
DEBUG 1. To detect, locate, and remove mistakes from a program. 2. Troubleshoot.

DECIMAL CODE A code in which each allowable position has one of ten possible states. (The conventional decimal number system is a decimal code.)

DELETE CHARACTER A character used primarily to obliterate any erroneous or unwanted characters on punched tape. The delete character consists of perforations in all punching positions.



DELTA DIMENSIONING A system used for defining part dimensions on a part drawing in which each dimension is referenced from the preceding one. Also known as incremental dimensioning in some shops.



DIAGNOSTIC TEST The running of a machine program or routine to discover a failure or potential failure of a machine element and to determine its location.

DIGIT A character in any numbering system.

DIGITAL 1. Refers to discrete states of a signal (on or off). A combination of these makes up a specific value. 2. Relating to data in the form of digits.

DISPLAY A visual representation of data.

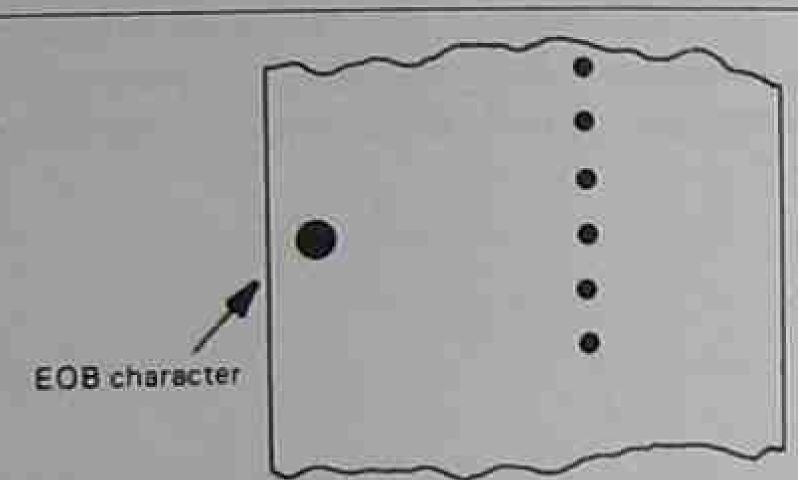
DOCUMENTATION Manuals and other printed materials (tables, magnetic tape, listing, diagrams) which provide information for use and maintenance of a manufactured product, both hardware and software.

DWELL A timed or untimed delay in a program's execution. A timed dwell will resume the program after the programmed duration. An untimed dwell requires operator intervention to continue the program.

EDIT To modify the form of data.

EIA STANDARD CODE A standard code for positioning, straight-cut, and contouring control systems proposed by the U.S. EIA in their Standard RS-244. Eight-track paper (one-inch wide) has been accepted by the American Standards Association as an American standard for numerical control.

END OF BLOCK CHARACTER 1. A character indicating the end of a block of tape information. Used to stop the tape reader after a block has been read. 2. The typewriter function of the carriage return when preparing machine control tapes.



END OF PROGRAM A miscellaneous function (M02) indicating the completion of a workpiece. Stops spindle, coolant, and feed after completion of all commands in the block. Used to reset control and/or machine.

END OF TAPE A miscellaneous function (M30) which stops spindle, coolant, and feed after completion of all commands in the block. Used to reset control and/or machine.

END POINT The extremities of a span.

ERROR SIGNAL Indication of a difference between the output and input signals in a servo system.

EXECUTIVE PROGRAM A series of programming instructions enabling a dedicated minicomputer to produce a specific output control. For example, it is the executive program in a CNC unit that enables the control to think like a lathe or machining center.

FEED The programmed or manually established rate of movement of the cutting tool into the workpiece for the required machining operation.

FEEDBACK The transmission of a signal from a late to an earlier stage in a system. In a closed-loop NC system, a signal of the machine slide position is fed back and compared with the input signal, which specifies the demanded position. These two signals are compared and generate an error signal if a difference exists.

FEED FUNCTION The relative motion between the tool or instrument and the work due to motion of the programmed axis.

FEEDRATE (CODE WORD) A multiple-character code containing the letter F followed by digits. It determines the machine slide rate of feed.

FEEDRATE DIVIDER A feature of some machine control units that gives the capability of dividing the programmed feedrate by a selected amount as provided for in the machine control unit.

FEEDRATE MULTIPLIER A feature of some machine control units that gives the capability of multiplying the programmed feedrate by a selected amount as provided for in the machine control unit.

FEEDRATE OVERRIDE A variable manual control function directing the control system to reduce the programmed feedrate.

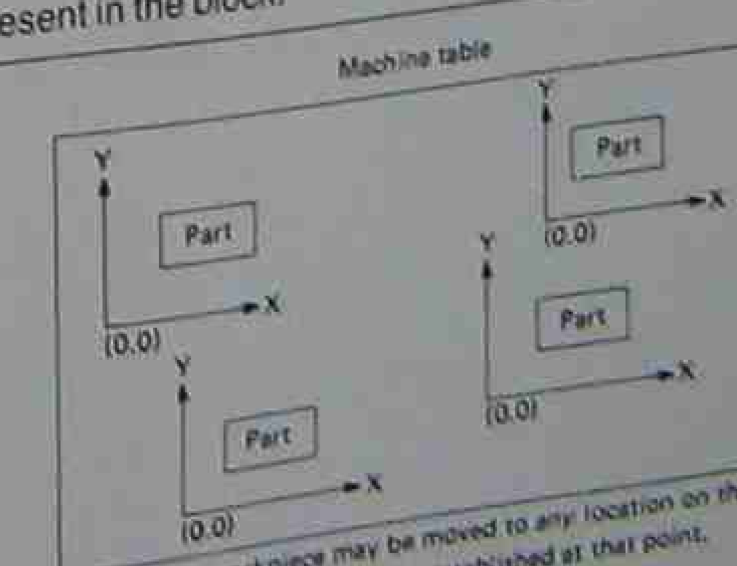
Feedrate override is a percentage function to reduce the programmed feed rate. If the programmed feed rate was 30 inches per minute and the operator wanted 15 inches per minute, the feedrate override dial would be set at 50 percent.

FIXED BLOCK FORMAT A format in which the number and sequence of words and characters appearing in successive blocks is constant.

FIXED CYCLE See canned cycle.

FIXED SEQUENTIAL FORMAT A means of identifying a word by its location in a block of information. Words must be presented in a specific order, and all possible words preceding the last desired word must be present in the block.

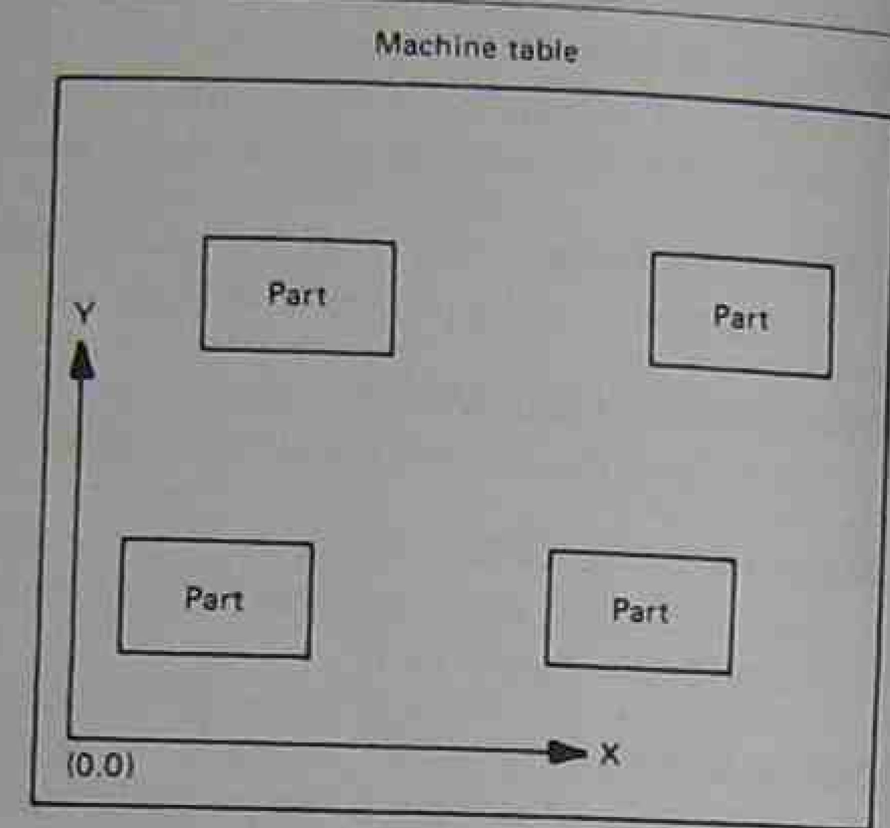
FLOATING ZERO A characteristic of a machine control unit permitting the zero reference point on an axis to be established readily at any point in the travel.



The part or workpiece may be moved to any location on the machine table and zero may be established at that point.

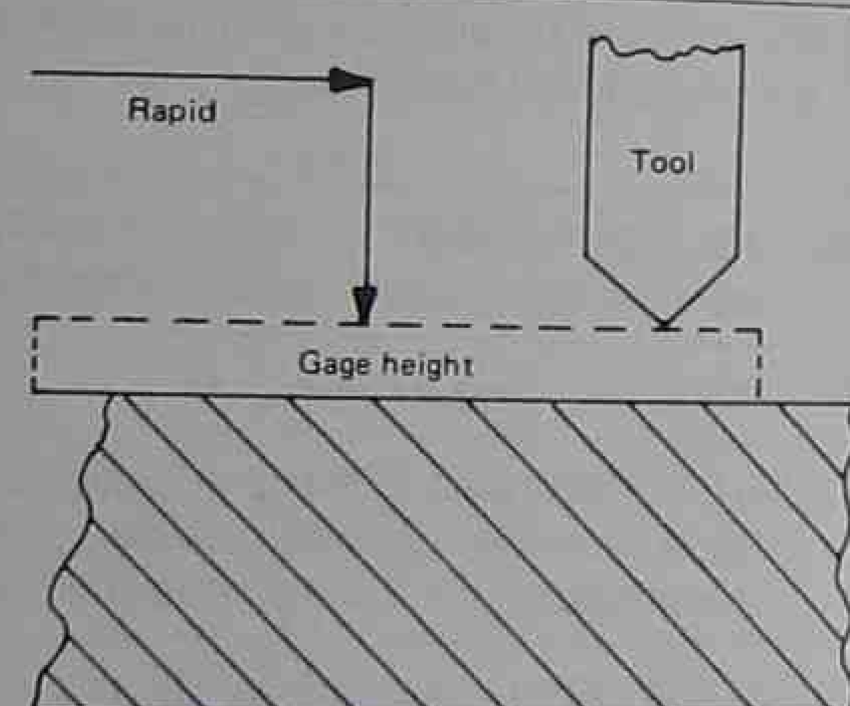
FORMAT (TAPE) The general order in which information appears on the input media, such as the location of holes on a punched tape or the magnetized areas on a magnetic tape.

FULL RANGE FLOATING ZERO A characteristic of a numerical machine tool control permitting the zero point on an axis to be shifted readily over a specified range. The control retains information on the location of permanent zero.



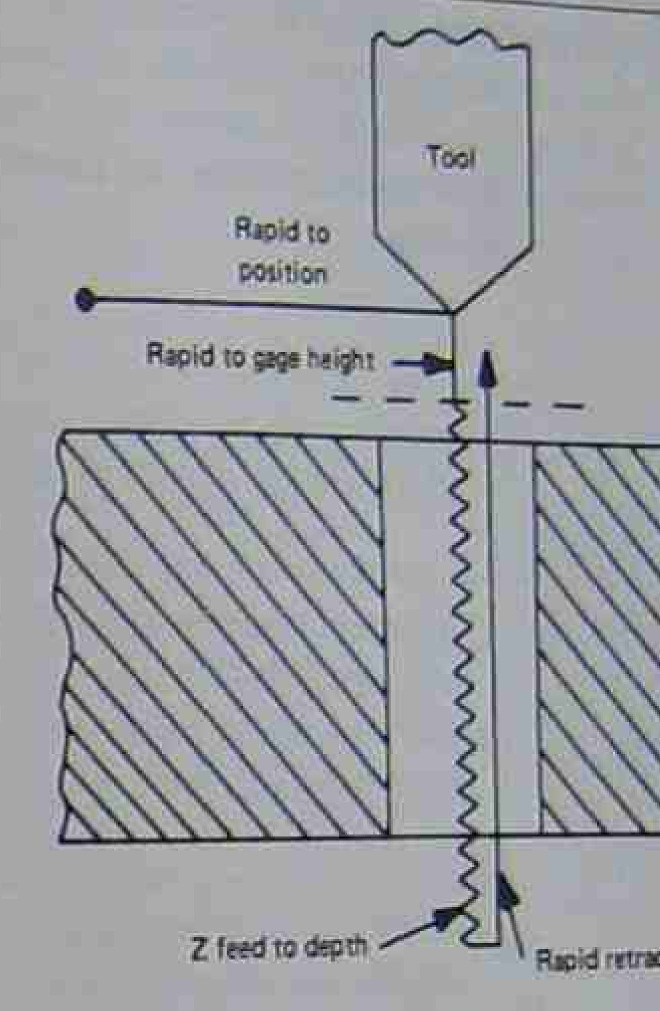
The part or workpiece may be shifted to any position on the machine table, but the actual position of permanent zero remains constant.

GAGE HEIGHT A predetermined partial retraction point along the Z axis to which the cutter retreats from time to time to allow safe XY table travel. Also called the reference or rapid level.



Gage height, usually .100 to .125, is a set distance established in the control or set by the operator. Gage height allows the tool, while advancing in rapid traverse, to stop at the established distance (gage height) and begin feed motion. Without gage height, the tool would rapid into the part causing tool damage or breakage and potential operator injury.

G CODE A word addressed by the letter G and followed by a numerical code defining preparatory functions or cycle types in a numerical control system.



G81 – Drill Cycle

GENERAL PROCESSOR 1. A computer program for converting geometric input data into cutter path data required by an NC machine. 2. A fixed software program designed for a specific logical manipulation of data.

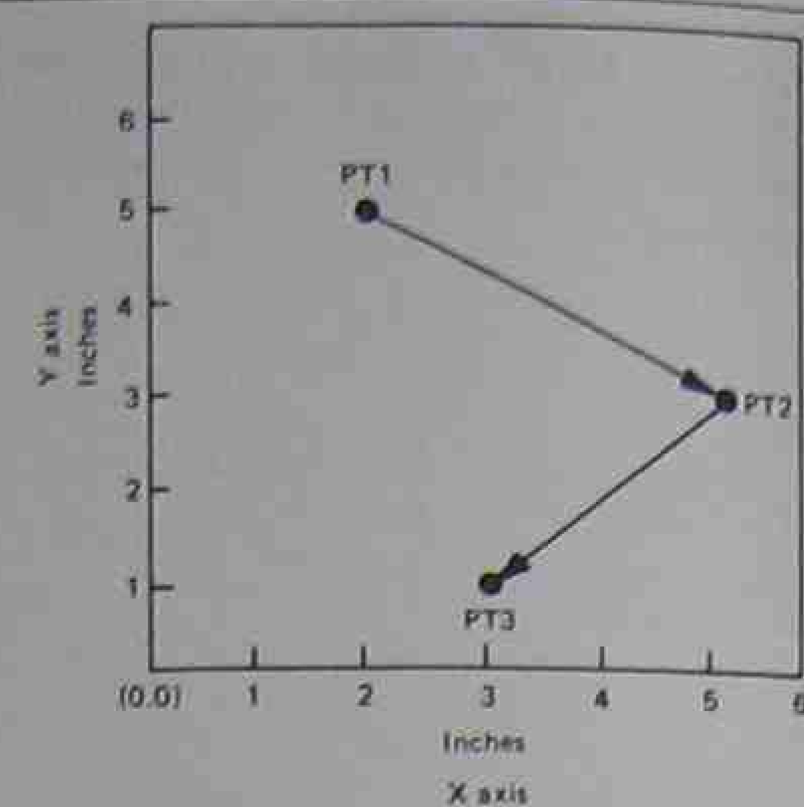
HARD COPY A readable form of data output on paper.

HARDWARE The component parts used to build a computer or control system, e.g. integrated circuits, diodes, transistors.

HARD-WIRED Having logic circuits interconnected on a backplane to give a fixed pattern of events.

HIGH-SPEED READER A reading device which can be connected to a computer or control so as to operate on line without seriously holding up the computer or control.

INCREMENTAL SYSTEM A control system in which each coordinate or positional dimension, both input and feedback, is taken from the past position rather than from a common datum point, as in the absolute system.



Coordinate positions

Point	X value	Y value
PT1	2	5
PT2	3	-2
PT3	-2	-2

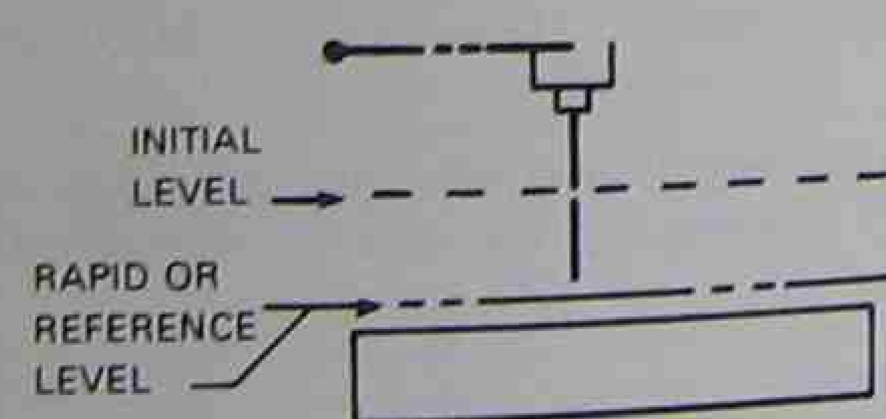
In an incremental system, all points are expressed relative to the preceding point.

INDEX TABLE A multiple-character code containing the letter B followed by digits. This code determines the position of the rotary index table in degrees.

See B (Beta) axis.

INHIBIT To prevent an action or acceptance of data by applying an appropriate signal to the appropriate input.

INITIAL LEVEL The position of the spindle at the beginning of a canned cycle operation.



INPUT Transfer of external information into the control system.

INPUT MEDIA 1. The form of input such as punched cards and tape or magnetic tape. 2. The device used to input information.

INTERCHANGEABLE VARIABLE BLOCK FORMAT A programming arrangement consisting of a combination of the word address and tab sequential formats to provide greater compatibility in programming. Words are interchangeable within the block. Length of block varies since words may be omitted.

This is one of the most sophisticated tape formats in use today.

See *block*.

INTERCHANGE STATION The position where a tool of an automatic tool changing machine awaits automatic transfer to either the spindle or the appropriate coded drum station.

INTERMEDIATE TRANSFER ARM The mechanical device in automatic tool changing that grips and removes a programmed tool from the coded drum station and places it into the interchange station, where it awaits transfer to the machine spindle. This device then automatically grips and removes the used tool from the interchange station and returns it to the appropriate coded drum station.

INTERPOLATION 1. The insertion of intermediate information based on an assumed order or computation. 2. A function of a control whereby data points are generated between given coordinate positions.

INTERPOLATOR A device which is part of a numerical control system and performs interpolation.

ISO International Organization for Standardization.

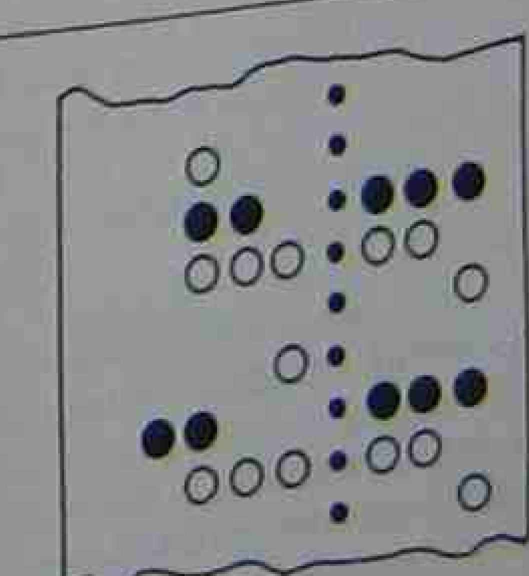
JOG A control function which momentarily operates a drive to the machine.

LEADING ZEROES Redundant zeroes to the left of a number.

Leading zeroes
X + 0062500

LEADING ZERO SUPPRESSION See zero suppression.

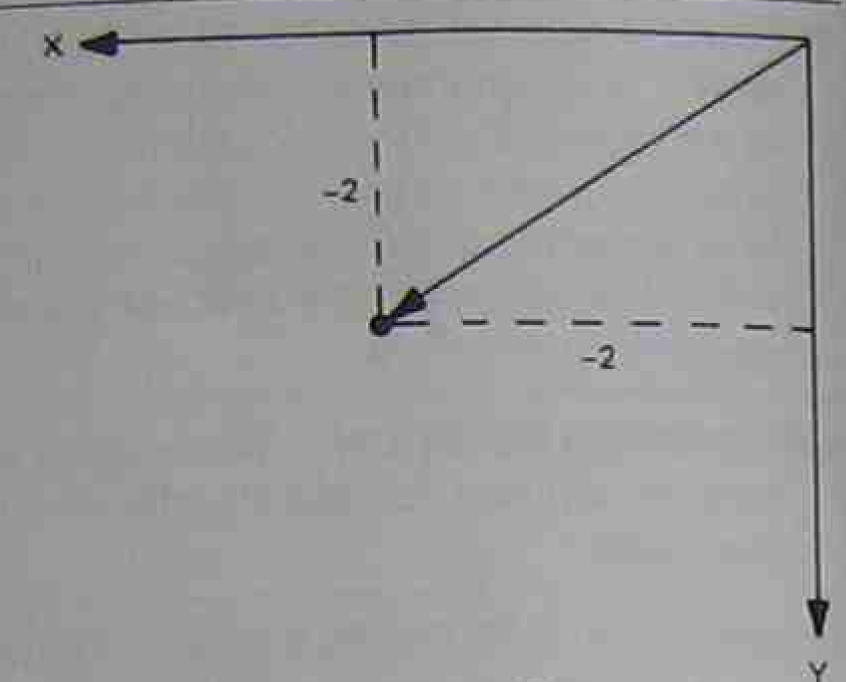
LETTER ADDRESS The method by which information is directed to different parts of the system. All information must be preceded by its proper letter address, e.g., X, Y, Z, M.



X and G address

An identifying letter inserted in front of each word.

LINEAR INTERPOLATION A function of a control whereby data points are generated between given coordinate positions to allow simultaneous movement of two or more axes of motion in a linear (straight) path.



The control system moves X and Y axes proportionately to arrive at the destination point.

LOOP TAPE A short piece of tape, with joined ends, which contains a complete program or operation.

MACHINING CENTER Machine tools, usually numerically controlled, capable of automatically drilling, reaming, tapping, milling, and boring multiple faces of a part. Equipped with a system for automatically changing cutting tools.

MACRO A group of instructions which can be stored and recalled as a group to solve a recurring problem.

An APT macro could be as follows:

```
DRILL1 = MACRO/X, Y, Z, Z1, FR, RR
    GOTO/POINT, X, Y, Z, RR
    GODLTA/-Z1, FR
    GODLTA/+Z1, RR
TERMAC
```

X, Y, Z, Z1, FR, and RR would be variables which would have values assigned when the macro is called into action. The variables would be as follows:

X = X position
Y = Y position
Z = Z position (above work surface)
Z1 = Z feed distance
FR = feed rate
RR = rapid rate

The call statement could be:

```
CALL/DRILL1, X = 2, Y = 4, Z = .100, Z1 = 1.25,
FR = 2, RR = 200
```

MAGIC-THREE CODING A feedrate code that uses three digits of data in the F word. The first digit defines the power of ten multiplier. It determines the positioning of the floating decimal point. The last two digits are the most significant digits of the desired feedrate.

To program a feed rate of 12 inches per minute in magic-three coding:

1) count the number of decimal places to the left of the decimal. 12 = 2
2) Add magic "3" to the number of counted decimal places. (3 + 2 = 5)
3) write the F word address, the added digit, and the first two digits of the actual feed rate to be programmed. (F512)
4) F512 would be the magic "3" coded feed rate.

This method of feed rate coding is now almost obsolete.

MAGNETIC TAPE A tape made of plastic and coated with magnetic material. It stores information by selective polarization of portions of the surface.

MANUAL DATA INPUT A mode or control that enables an operator to insert data into the control system. The data are identical to information that could be inserted by tape.

MANUAL PART PROGRAMMING The preparation of a manuscript in machine control language and format to define a sequence of commands for use on an NC machine.

Manual, or hand, programming is programming the actual codes, X and Y positions, functions, etc. as they are punched in the N/C tape.

H001 G81 X+37500 Y+52500 W01

MANUSCRIPT A written or printed copy, in symbolic form, containing the same data as that punched on cards or tape or retained in a memory unit.

MEMORY An organized collection of storage elements, e.g., disc, drum, ferrite cores, into which a unit of information consisting of a binary digit can be stored and from which it can later be retrieved.

A computer with a 64,000-word capacity is said to have a memory of 64 K.

MIRROR IMAGE See axis inversion.

MODAL Information that is retained by the system until new information is obtained and replaces it.

MODULE An interchangeable plug-in item containing components.

NC (Numerical control) The technique of controlling a machine or process by using command instructions in coded numerical form.

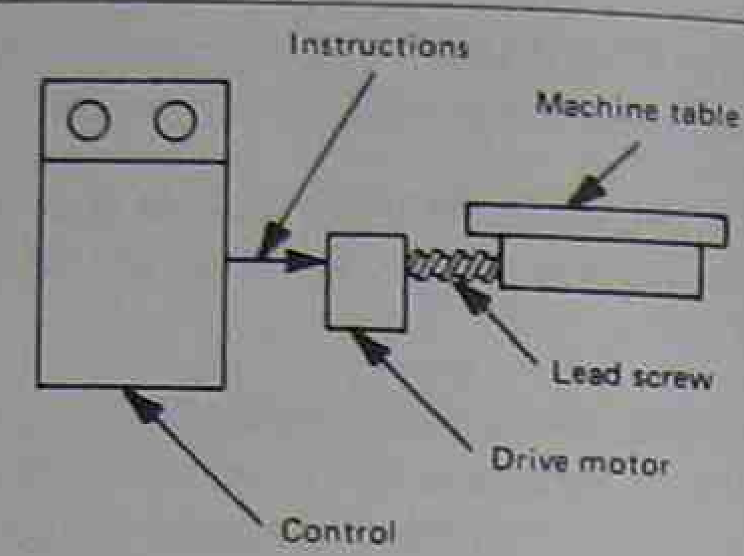
NULL 1. Pertaining to no deflection from a center or end position. 2. Pertaining to a balanced or zero output from a device.

NUMERICAL CONTROL SYSTEM A system in which programmed numerical values are directly inserted, stored on some form of input medium, and automatically read and decoded to cause a corresponding movement in a machine or process.

OFFLINE PROGRAMMING The development of an NC part program away from the machine console to be transferred to the MCU at a later time.

OFFSET A displacement in the axial direction of the tool which is the difference between the actual tool length and the programmed tool length.

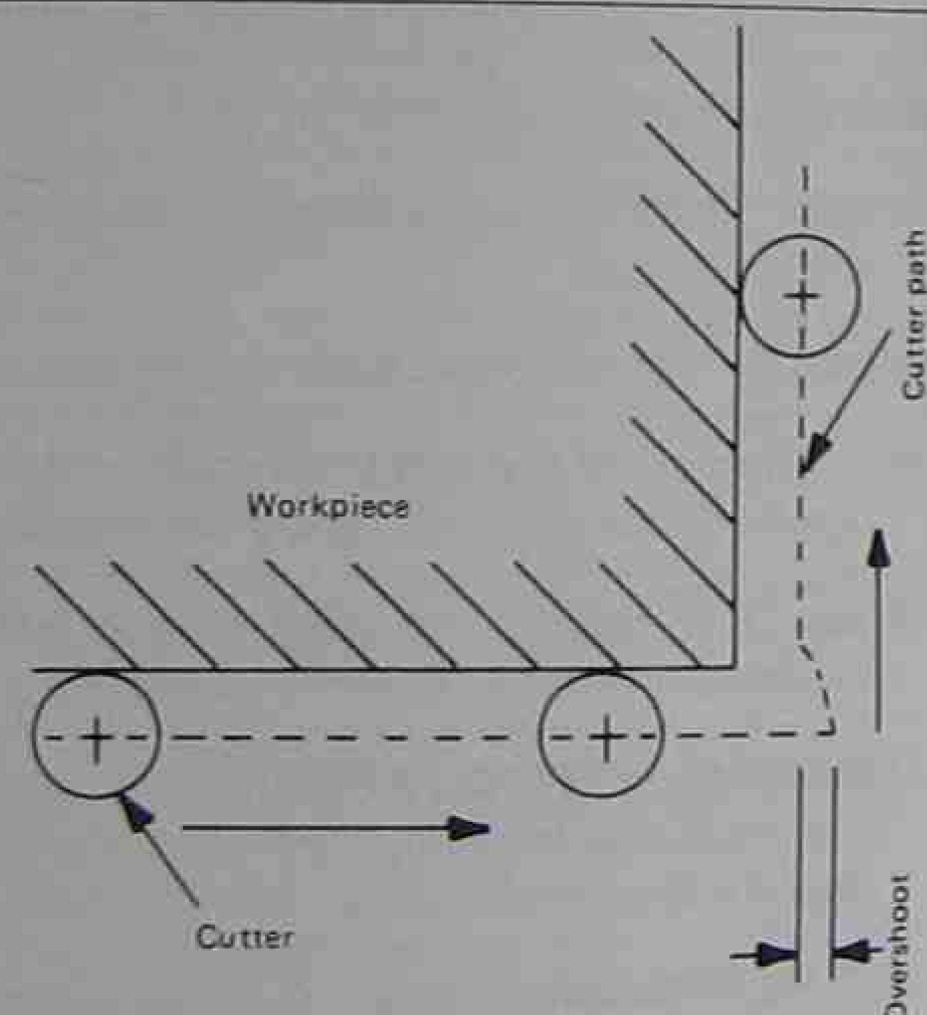
OPEN-LOOP SYSTEM A control system that has no means of comparing the output with the input for control purposes. No feedback.



OPTIMIZE To rearrange the instructions or data in storage so that a minimum number of transfers are required in the running of a program. To obtain maximum accuracy and minimum part production time by manipulation of the program.

OPTIONAL STOP A miscellaneous function (M01) command similar to Program Stop except the control ignores the command unless the operator has previously pushed a button to validate the command.

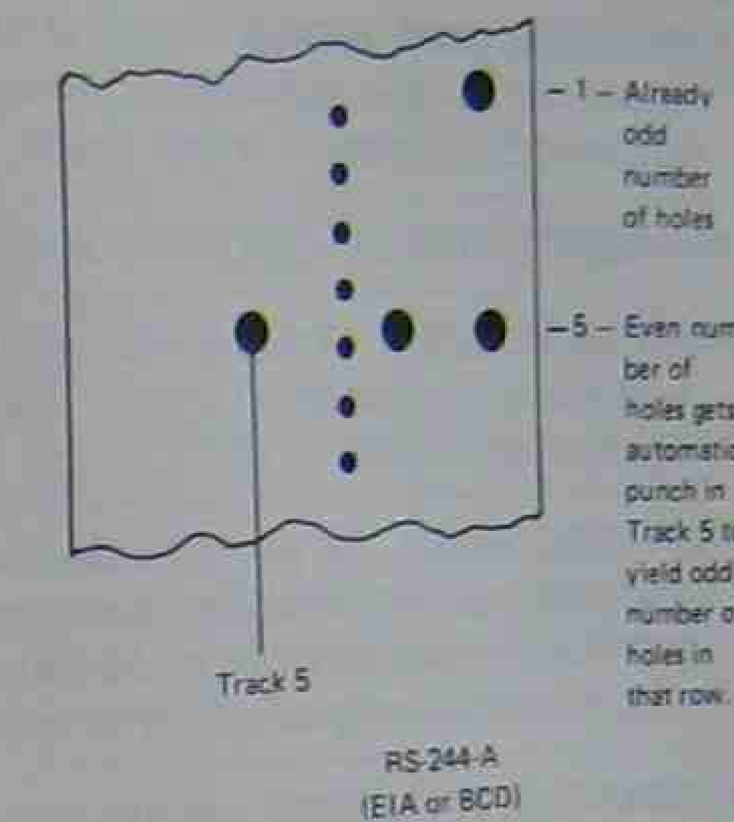
OVERSHOOT A term applied when the motion exceeds the target value. The amount of overshoot depends on the feedrate, the acceleration of the slide unit, or the angular change in direction.



PARABOLA A plane curve generated by a point moving so that its distance from a fixed second point is equal to its distance from a fixed line.

PARABOLIC INTERPOLATION Control of cutter path by interpolation between three fixed points by assuming the intermediate points are on a parabola.

PARITY CHECK 1. A hole punched in one of the tape channels whenever the total number of holes is even, to obtain an odd number, or vice versa depending on whether the check is even or odd. 2. A check that tests whether the number of ones (or zeroes) in any array of binary digits is odd or even.



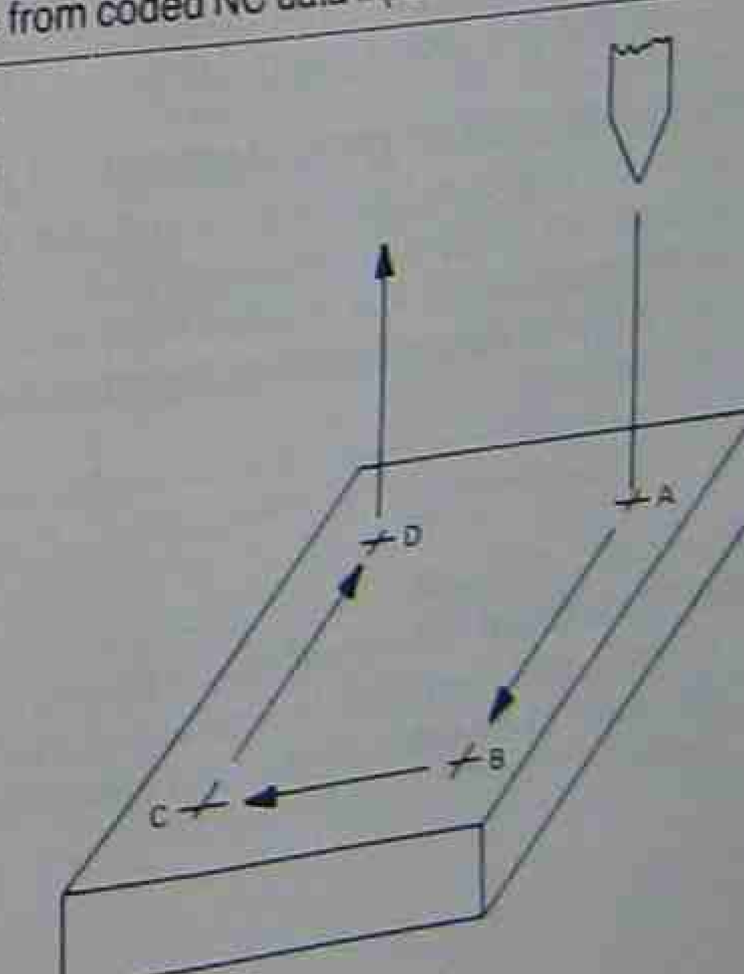
PART PROGRAM A specific and complete set of data and instructions written in source languages for computer processing or in machine language for manual programming to manufacture a part on an NC machine.

PART PROGRAMMER A person who prepares the planned sequence of events for the operation of a numerically controlled machine tool.

PERFORATED TAPE A tape on which a pattern of holes or cuts is used to represent data.

PLOTTER A device which will draw a plot or trace from coded NC data input.

POINT-TO-POINT CONTROL SYSTEM A numerical control system in which controlled motion is required only to reach a given end point, with no path control during the transition from one end point to the next.



POSITIONING/CONTOURING A type of numerical control system that has the capability of contouring, without buffer storage, in two axes and positioning in a third axis for such operations as drilling, tapping, and boring.

POSITIONING SYSTEM See point-to-point control system.

POSITION READOUT A display of absolute slide position as derived from a position feedback device (transducer) normally attached to the lead screw of the machine. See command readout.

POSTPROCESSOR The part of the software which converts the cutter path coordinate data into a form which the machine control can interpret correctly. The cutter path coordinate data are obtained from the general processor and all other programming instructions and specifications for the particular machine and control.

PREPARATORY FUNCTION An NC command on the input tape changing the mode of operation of the control. (Generally noted at the beginning of a block by the letter G plus two digits.)

Some preparatory functions are:

G84 – tap cycle
G01 – linear interpolation
G82 – dwell cycle
G02 – circular interpolation – clockwise
G03 – circular interpolation – counter clockwise

See G code.

PROGRAM A sequence of steps to be executed by a control or a computer to perform a given function.

PROGRAMMED DWELL The capability of commanding delays in program execution for a programmable length of time.

PROGRAMMER (PART PROGRAMMER) A person who prepares the planned sequence of events for the operation of a numerically controlled machine tool. The programmer's principal tool is the manuscript on which the instructions are recorded.

Manual part programming instructions:

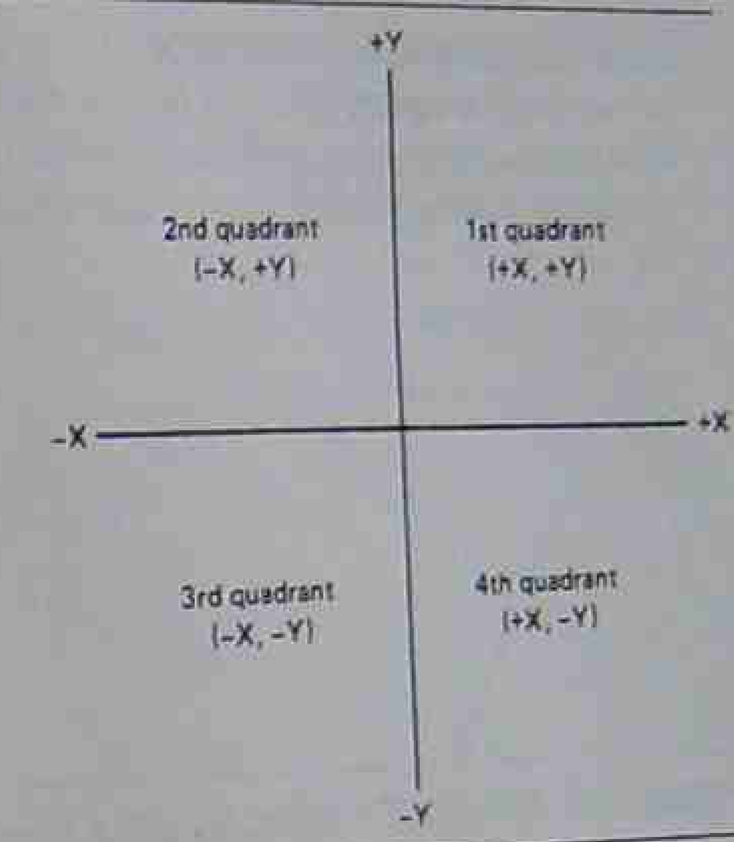
H001	G81	X+123750	Y+62500	W01	
N002		X+105000			
N003			Y+51250		M06

Computer part programming instructions:

```
TLRGT, GORGT/HL3, TANTO, C1
       GOFWD/C1, TANTO, HL2
       GOFWD/HL2, PAST, VL2
```

PROGRAM STOP A miscellaneous function (M00) command to stop the spindle, coolant, and feed after completion of the dimensional move commanded in the block. To continue with the remainder of the program, the operator must push a button.

QUADRANT Any of the four parts into which a plane is divided by rectangular coordinate axes in that plane.



RANDOM Not necessarily in a logical order of arrangement according to usage, but having the ability to select from any location and in any order from the storage system.

RAPID Positioning the cutter and workpiece into close proximity with one another at a high rate of travel speed, usually 150 to 400 inches per minute (IPM) before the cut is started.

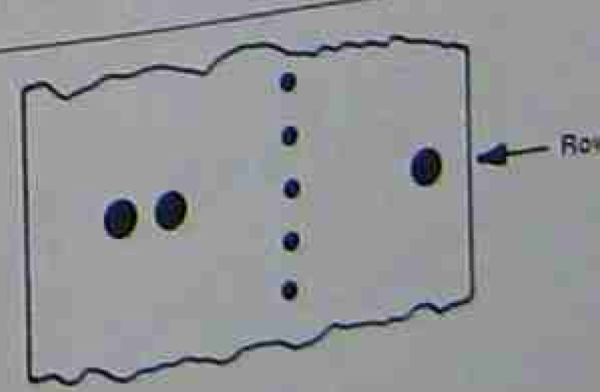
READER A pneumatic, photoelectric, or mechanical device used to sense bits of information on punched cards, punched tape, or magnetic tape.

REGISTER An internal array of hardware binary circuits for temporary storage of information.

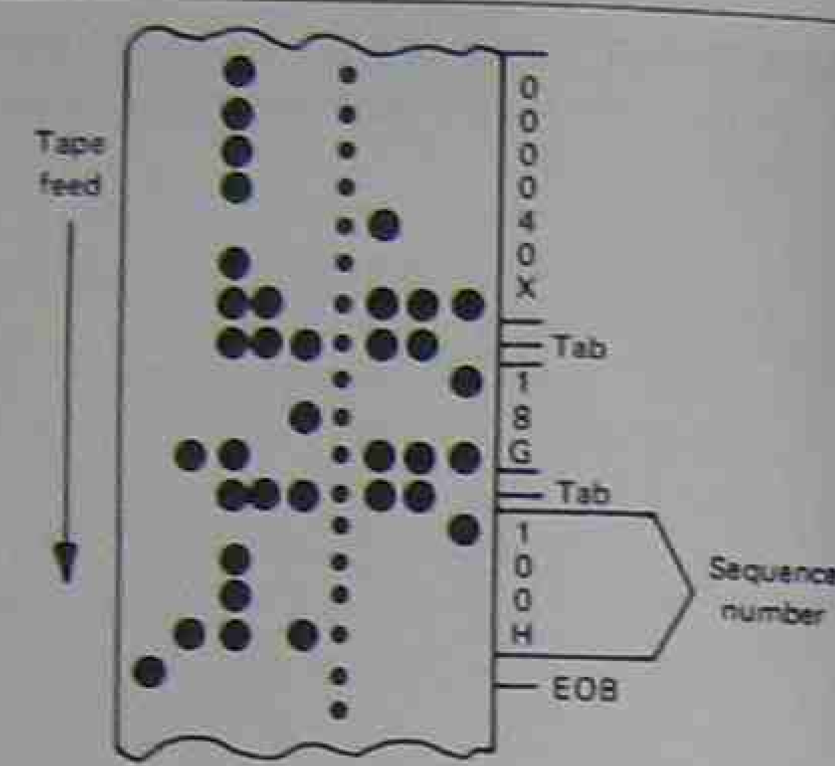
REPEATABILITY Closeness of, or agreement in, repeated measurements of the same characteristics by the same method, using the same conditions.

RESET To return a register or storage location to zero or to a specified initial condition.

ROW (TAPE) A path perpendicular to the edge of the tape along which information may be stored by the presence or absence of holes or magnetized areas. A character would be represented by a combination of holes.



SEQUENCE NUMBER (CODE WORD) A series of numerals programmed on a tape or card and sometimes displayed as a readout; normally used as a data location reference or for card sequencing.



SEQUENCE READOUT A display of the number of the block of tape being read by the tape reader.

SEQUENTIAL Arranged in some predetermined logical order.

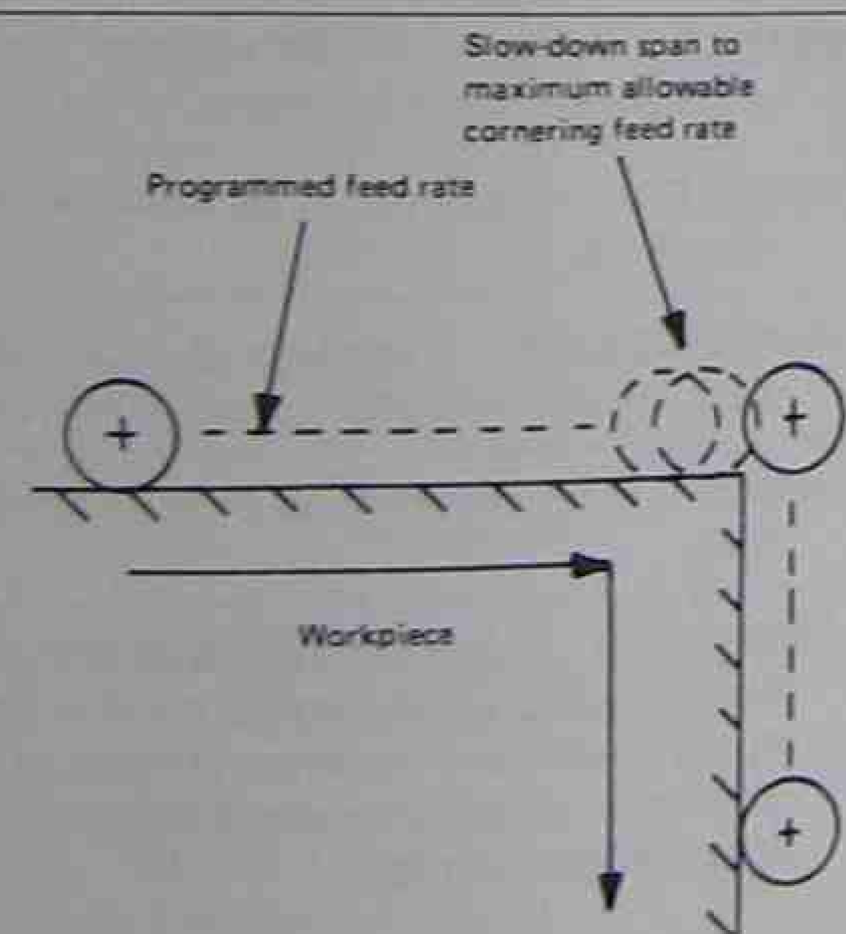
SIGNIFICANT DIGIT A digit that must be kept to preserve a specific accuracy or precision.

Significant digits

X + 0052500

Insignificant digits

SLOW-DOWN SPAN A span of information having the necessary length to allow the machine to decelerate from the initial feedrate to the maximum allowable cornering feedrate that maintains the specified tolerance.



Postprocessor output

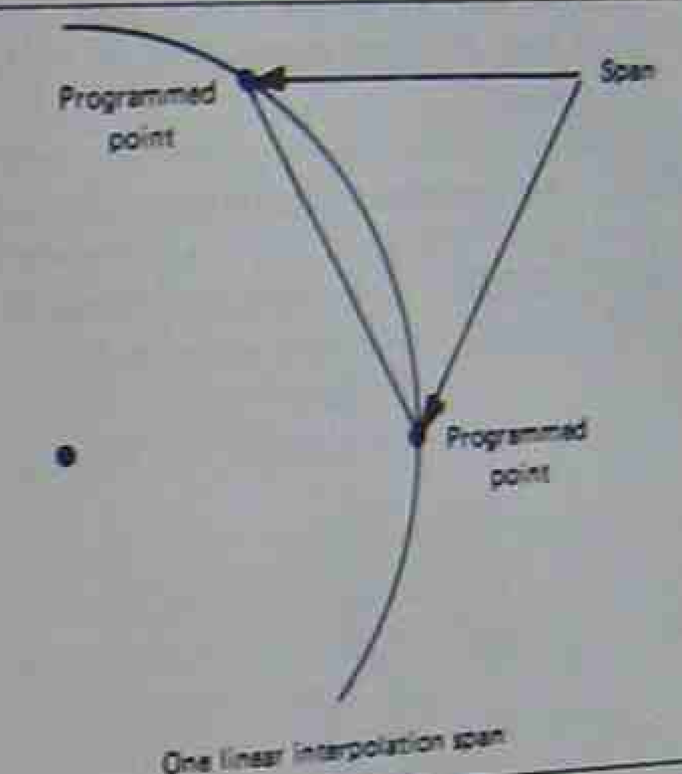
G01	X+42500	Y+100000	F200
	X+19750		F175
	X+17215		F140
	X+13750	Y+68750	F200

SOFTWARE Instructional literature and computer programs used to aid in part programming, operating, and maintaining the machining center.

Examples of software programs are:

APT
FORTRAN
COBOL
RPG

SPAN A certain distance or section of a program designated by two end points for linear interpolation; a beginning point, a center point, and an ending point for circular interpolation; and two end points and a diameter point for parabolic interpolation.

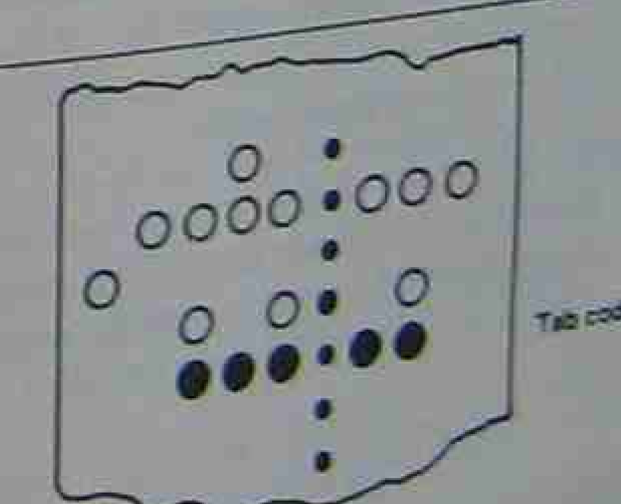


SPINDLE SPEED (CODE WORD) A multiple-character code containing the letter S followed by digits. This code determines the RPM of the cutting spindle of the machine.

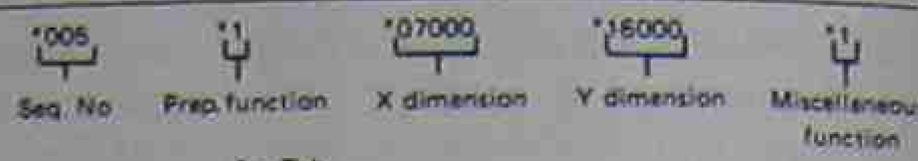
STORAGE A device into which information can be introduced, held, and then extracted at a later time.

STORAGE MEDIA A device onto which information can be transferred and retained for later use. Storage media may also be used as input media, thereby serving a dual purpose.

TAB A nonprinting spacing action on tape preparation equipment. A tab code is used to separate words or groups of characters in the tab sequential format. The spacing action sets typewritten information on a manuscript into tabular form.



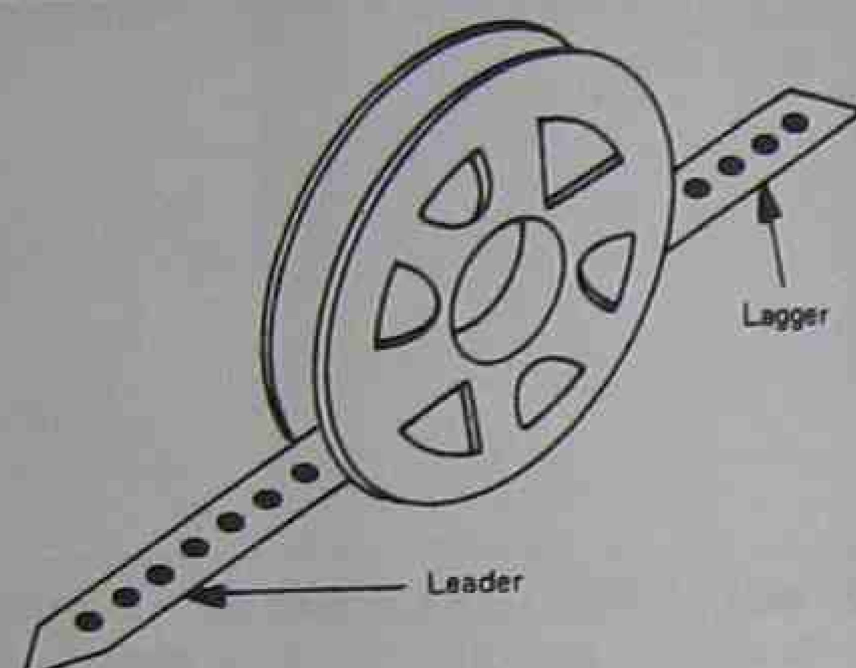
TAB SEQUENTIAL FORMAT Means of identifying a word by the number of tab characters preceding the word in a block. The first character of each word is a tab character. Words must be presented in a specific order, but all characters in a word, except the tab character, may be omitted when the command represented by that word is not desired.



The tab sequential format is, for the most part, obsolete.

TAPE A magnetic or perforated paper medium for storing information.

TAPE LAGGER The trailing end portion of a tape.
TAPE LEADER The front or lead portion of a tape.



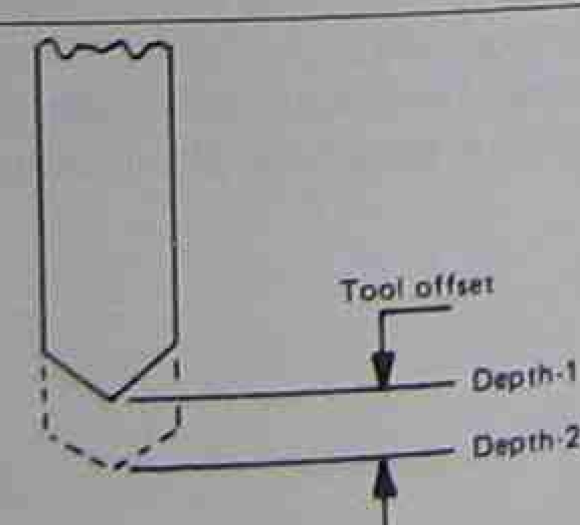
Reel tapes should have a leader and lagger of approximately three feet with just sprocket holes for tape loading and threading purposes.

TOOL FUNCTION A tape command identifying a tool and calling for its selection. The address is normally a T word.

T06 would be a tape command calling for the tool assigned to spindle or pocket 6 to be put in the spindle.

TOOL LENGTH COMPENSATION A manual input, by means of selector switches, to eliminate the need for preset tooling; allows the programmer to program all tools as if they are of equal length.

TOOL OFFSET 1. A correction for tool position parallel to a controlled axis. 2. The ability to reset tool position manually to compensate for tool wear, finish cuts, and tool exchange.



Tool offsets are used as final adjustments to increase or decrease depths due to cutting forces and tool deflection. In this case, a tool offset could be used to increase the drill depth from depth-1 to depth-2.

TRAILING ZERO SUPPRESSION See zero suppression.

TURNKEY SYSTEM A term applied to an agreement whereby a supplier will install an NC or computer system so that he has total responsibility for building, installing, and testing the system.

USASCII United States of America Standard Code for Information Interchange. See ASCII.

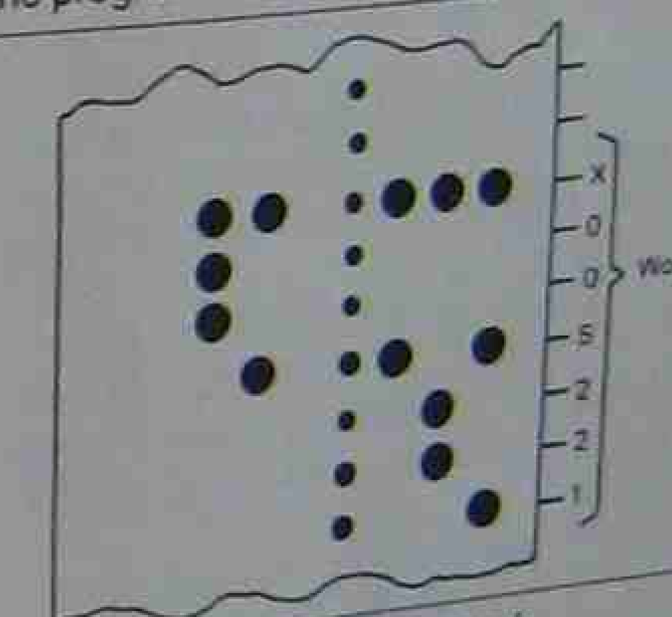
VARIABLE BLOCK FORMAT (TAPE) A format which allows the quantity of words in successive blocks to vary.

Same as word address. Variable block means the length of the blocks can vary depending on what information needs to be conveyed in a given block. See *block*.

VECTOR A quantity that has magnitude, direction, and sense; is represented by a directed line segment whose length represents the magnitude and whose orientation in space represents the direction.

VECTOR FEEDRATE The feedrate at which a cutter or tool moves with respect to the work surface. The individual slides may move slower or faster than the programmed rate, but the resultant movement is equal to the programmed rate.

WORD An ordered set of characters which is the normal unit in which information may be stored, transmitted, or operated upon.



See address and block.

WORD ADDRESS FORMAT The specific arrangement of addressing each word in a block of information by one or more alphabetical characters which identify the meaning of the word.

See word.

WORD LENGTH The number of bits or characters in a word.

X AXIS Axis of motion that is always horizontal and parallel to the workholding surface.

Y AXIS Axis of motion that is perpendicular to both the X and Z axes.

Z AXIS Axis of motion that is always parallel to the principal spindle of the machine.

ZERO OFFSET A characteristic of a numerical machine tool control permitting the zero point on an axis to be shifted readily over a specified range. The control retains information on the location of the permanent zero.

See *full range floating zero* and *floating zero*.

ZERO SHIFT A characteristic of a numerical machine tool control permitting the zero point on an axis to be shifted readily over a specified range. (The control does *not* retain information on the location of the permanent zero.)

See *floating zero*. Consult chapter 4 for additional details.

ZERO SUPPRESSION Leading zero suppression: the elimination of insignificant leading zeroes to the left of significant digits usually before printing. Trailing zero suppression: the elimination of insignificant trailing zeroes to the right of significant digits usually before printing.

Leading zero suppression

X + 0043500

Insignificant digits

Could be written as:

X + 43500

Trailing zero suppression

X + 0043500

Insignificant digits

Could be written as:

X + 00435

RELATED SME TITLES

The Society of Manufacturing Engineers offers several related titles in CNC and Machine Trades.

Machinery's Handbook, E. Oberg, F. Jones, H. Horton
2500 pp, hardcover, 1988
Order Code: 1379–0802
$55.00 (SME Members: $49.00)

Speed and Feed Chart
Order Code: 825–0802
$15.00 (SME Members: $10.00)

Machine Controls (From TMEH Machining, Volume 1)
60 pp, softcover
Order Code: 1355–0802
$24.50 (SME Members: $21.50)

Fundamentals of Numerical Control, W. Luggen
287 pp, hardcover, 1988
Order Code: 1387–0802
$28.95 (SME Members: $26.95)

Troubleshooting Manufacturing Processes
545 pp, hardcover, 1988
Order Code: 1345–0802
$69.00 (SME Members: $64.00)

Cutting Fluids and Lubricants
350 pp, softcover, 1985
Order Code: 953–0802
$38.00 (SME Members: $33.00)

Nontraditional Machining Processes, 2nd Edition
255 pp, softcover, 1983
Order Code: 788–0802
$26.00 (SME Members: $23.00)

Machining Hard Materials
250 pp, hardcover, 1982
Order Code: 664–0802
$38.00 (SME Members: $32.00)

Machinists' Ready Reference Manual
466 pp, hardcover, 1988
Order Code: 1368-0802
$32.00 (SME Members: $27.00)

Modern Trends in Cutting Tools
300 pp, hardcover, 1982
Order Code: 686–0802
$38.00 (SME Members: $32.00)

Improving Production with Coolants and Lubricants
240 pp, hardcover, 1982
Order Code: 655–0802
$38.00 (SME Members: $32.00)

Machining Fundamentals
300 pp, softcover, 1980
Order Code: 514–0802
$29.50 (SME Members: $24.00)

Milling: Methods and Machines
250 pp, hardcover, 1980
Order Code: 669–0802
$38.00 (SME Members: $32.00)

For other titles in manufacturing, materials, design, quality control, automated systems, robots, and management, request a copy of the latest SME catalog from:

Society of Manufacturing Engineers
One SME Drive
P.O. Box 930
Dearborn, MI 48121

Call: 313/271–1500

INDEX

A axis, 444
Absolute accuracy, 444
Absolute positioning, 29
  programming coordinates for, 105
Absolute readout, 444
Absolute system, 444
Absolute zero shift, 32-34
Accuracy, 445
AC servo, 22
AD-APT (Adaption of APT), 357, 445
Adaptive control, 445
Address, 445
Air cushion vehicle, 395
Alphanumeric coding, 446
Analog, 446
Analyst, 446
APT (Automatic Programmed Tools), 357, 446
  auxiliary statements in, 365-366
  part geometry definitions in, 358-365
  tool motion statements in, 367-369
Arc clockwise, 446
Arc counterclockwise, 446
ASCII (American Standard Code for Information Interchange), 10, 447
AUTO-MAP (Automatic Machining Program), 358, 447
Automatic guided vehicle, 395
Automation, 447
AUTOSPOT, 447
AUX code, 277, 447
Auxiliary function, 447
Axis, 447
Axis inhibit, 447
Axis interchange, 447
Axis inversion, 447

Backlash, 448
Ball end mills, 50
B axis, 448
BCD (Binary-coded decimal), 448
Binary code, 449
Binary numbers, 7-8
Bit, 449
Block, 8, 449
Block delete, 450
Boring, 47
Buffer storage, 450
Bug, 450
Byte, 450

CAD/CAM programming, 388-389
CAD (Computer-aided design) , 450
CAM (Computer-aided manufacturing), 387-388, 450
CAM-I (Computer Aided Manufacturing International), 450
Cancel command, 450
Canned cycles, 350, 422-426, 450
Carbide inserts, 52-53
Carousel magazine tool storage, 92-93
Cartesian coordinate system, 25, 451
C axis, 451
Center drill, 46
Centerline programming, 311

Ceramic, 45
Chad, 451
Channels, 451
Characters, 452
Chip, 452
CIM (computer integrated manufacturing), 389-390, 393
Circular interpolation, 209-211, 214-216, 452
  definition of, 203
  and lathes , 328-331
  and Machinist Shop Language , 211-212
  and word address format, 212-214
Closed-loop system, 22-23, 452
CMOS memory, 5
CNC machines, 5
  absolute positioning, 29
  absolute zero shift, 32-34
  Cartesian coordinate system, 25
  closed loop systems, 22-23
  continuous-path systems, 20-21
  controller, 18
  datum dimensioning, 37
  delta dimensioning, 37
  drive motors, 22
  incremental positioning, 30
  machine tool, 18
  manual setting, 32
  open loop systems, 22
  point-to-point systems, 18-20
  spindle movement, 25
  work coordinates, 34-37
Codes, 120-122, 166-167, 177-181, 277, 399-401, 406-409, 409-411, 452
Command readout, 453
Commands, 111-113, 154-157, 453
COMPACT II, 358
  auxiliary statements in, 373-375
  part geometry definitions in, 370-373
  tool motion statements in, 375
Computer-aided programming, 1
  AD-APT, 357
  APT, 357-369
  AUTOMAP, 358
  COMPACT II, 358, 370-375
  NUFORM, 358
  postprocessor, 375-385
  UNIAPT, 358
Computer-aided programming languages, 2
Computer graphics programming:
  CAD/CAM, 388-389
  CAM systems, 387-388
  CIM, 389-390
  digitizing systems, 386
  scanning systems, 386
Computer numerical control (CNC), 1-2, 453
Computers, NC programming, 356-391
Constant cutting speed, 453
Continuous-path systems, 20-21, 453
Contouring control system, 453
Controller, 18
Coordinates, programming:
  absolute and incremental positioning, mixing, 105
  hole operations, 99-101
  jig borer, 188-190
  metric, 105-107
  milling operations, 101-103
  polar axis, 285
CPU (central processing unit), 453
CRT (Cathode Ray Tube), 453
Cutter centerline formulas, 417-421
Cutter diameter compensation, 454
  codes for, 219-222
  definition of, 219
  and Fanuc controllers, 241-244
  fine tuning with, 240-241
  and Machinist Shop Language, 223-228
  subroutines for, 232-240, 264
  and word address format, 229-232
Cutter offsets, 454
  calculating, 191-197
  calculating linear interpolation, 203-206
Cutter path, 454
Cutting speed, 59-62, 415-416
Cutting tools:
  ceramic, 45
  cutting speed, 59-62, 415-416
  drilling feedrates, 65
  feedrates, 62
  high speed steel, 42
  milling feedrates, 69-70, 72
  speed and feed example, 72-73
  tungsten carbide, 42, 45
  turning feedrates, 62, 64
Cycle, 454
Cyclo mill, 51

Data, 454
Datum dimensioning, 37, 455
DC servo, 22
Debug, 455
Decimal code, 455
Delete character, 455
Delta dimensioning, 37, 455
Diagnostic test, 455
Diameter programming, 317
Digit, 456
Digital, 456
Digitizing systems, 386
Dimensioning:
  datum, 37, 455
  delta, 37, 455
Direct numerical control (DNC), 2
Display, 456
Distributive numerical control (DNC), 2
Documentation, 456
Do loops, 272
  and Machinist Shop Language, 247-251
  and word address format, 251-254
  See also Nested loops
Down milling, 126-127
Drilling, two-axis:
  absolute positioning, Machinist Shop Language, 116-118
  absolute positioning, word address format, 123-124
  incremental positioning, Machinist Shop Language, 118-120
  incremental positioning, word address format, 124-126
  Machinist Shop Language, 113-116
  word address format, 121-122
Drilling feedrate, 65
Drills and drilling, 46
  on NC lathes, 331-334
Drive motors:
  AC servos, 22
  DC servos, 22
  hydraulic servos, 22
  stepper motors, 22
Dwell, 456

Edit, 456
EIA Standard Code, 456
Electronics Industries Association (EIA), 7, 8
Electronics technician, 397
Employment, NC, 396-398
End mills, 50-51
End of block character, 456
End of program, 456
End of tape, 456
End point, 456
Error signal, 456
Executive program, 5, 456

Face mills, 51-52
Facing, 317-323
Fanuc controller:
  absolute positioning program, 142-144
  cutter diameter compensation program for, 241-244
  incremental positioning program, 142-144
  three-axis programming for, 181-184
  using do loops, 269-271
Feed, 457
  CNC lathe, 313
  drilling feedrate, 65, 416
  example of, 72-73
  feedrate, 62
  milling feedrate, 69-70, 72, 417
  turning feedrate, 62, 64
Feedback, 457
Feed function, 457
Feedrate (code word), 457
Feedrate divider, 457
Feedrate multiplier, 457
Feedrate override, 457
Firmware, 5
Fixed block format, 457
Fixed cycle. *See* Canned cycles
Fixed sequential format, 457
Fixturing, 41
Flexible machining system (FMS), 393-396
Floating zero, 457
Format (tape), 458
Formulas, cutter centerline, 417-421
Formulas, machining, 414-415
Full range floating zero, 458

Gage height, 458
G code, 459
General processor, 459
Glossary, 444-472

Hard copy, 459
Hardware, 459
Hardware guided vehicle, 395
Hard-wired, 459
Helical interpolation, 293-295, 299-300
  and Machinist Shop Language, 295-297
  and word address format, 297-299
High-speed reader, 459
High speed steel (HSS), 42
Hole operations, 45
  boring, 47
  drilling, 46
  programming coordinates for, 99-101
  reaming, 47
  tapping, 47
Home zero, 31
Hydraulic servo, 22

Incremental positioning, 30, 460
  programming coordinates for, 105
Index table, 460
Inhibit, 460
Initial level, 460
Input, 460
Input media, 5, 460
  magnetic tape, 7
  punched tape, 6
Inserted cutters:
  carbide, 52-53
  lathe tooling, 53-54
Interchangeable variable block format, 461
Interchange station, 461
Intermediate transfer arm, 461
Interpolation, 217, 461
  circular, 209-216, 452
  helical, 293-299
  linear, 203-209, 462
  parabolic, 464
Interpolator, 461
ISO (International Organization for Standardization), 461

Job shops, 392
Jog, 461

Lathes, NC, 314-315
  axis movement, 305-306
  bed design, 302-303
  feedrates, 313
  machine origin, 313
  quicksetters, 314
  sample program, 439-443
  spindle speeds, 311-313
  toolholders and tool changing, 306-311
  work coordinates, 313-314
  See also Turning machines
Lathe tooling, 53-54
  math for programming, 198-200
Leading zeros, 113, 461
Leading zero suppression. *See* Zero suppression
Letter address, 461
Linear interpolation, 462
  calculating cutter offsets, 203-209
  definition of, 203
  and Machinist Shop Language, 206-207
  taper turning, 323-328
  and word address format, 207
Loop systems:
  closed, 22-23, 452
  open, 22, 464
Loop tape, 462

Machine control unit (MCU), 18
Machine operator, 397
Machine origin, 31
  absolute zero shift, 32-34
  manual setting, 32
  work coordinates, 34-37, 313-314
Machine reference point, 316-317
Machine selection, 40-41
Machine tool, 18
Machining cell, 396
Machining center, 462
Machining formulas, 414-415
Machining strategy, 41
Machinist Shop Language:
  and circular interpolation, 211-212
  commands for, 111-113, 154-157, 402-405
  cutter diameter compensation using, 223-228
  do loops format, 247-251
  and helical interpolation, 295-297
  and linear interpolation, 206-207
  and mirror imaging, 277-281
  nested loops format, 267-269
  and polar rotation, 287-290
  subroutine format, 255-260

three-axis milling in, 149-154
two-axis drilling in, 113-120
two-axis milling in, 126-136
See also Word address format
Macro, 462
Magic-three coding, 462
Magnetic tape, 7, 463
Manual data input (MDI), 1, 463
Manual part programming, 1, 463
Manual zero setting, 32
Manufacturing, NC applications, 393-396
Manuscript, 463
Math, NC programming:
basic trigonometry, 187-191
cutter offsets, calculation of, 191-197
lathe example, 198-200
Matrix magazine tool storage, 94
Memory, 463
Methodizing, 40
Metric coordinates, 105-107
Milling, two-axis:
absolute positioning, Machinist Shop Language, 128-133
absolute positioning, word address format, 138-140
incremental positioning, Machinist Shop Language, 133-136
incremental positioning, word address format, 140-142
Machinist Shop Language, 126-127
sample program, 435-438
word address format, 136-138
Milling cutters, 49
end mills, 50-51
face mills, 51-52
Milling feedrate, 69-70, 72
Milling operations, 49
down milling, 126-127
programming coordinates for, 101-103
up milling, 126-127
Mirror imaging, 276, 299-300
and Machinist Shop Language, 277-281
and word address format, 282-285
Modal commands, 165, 463
Module, 463
Multi-axis tool changing, 88-89

NC setup sheet, 42
Nested loops, 272
and Machinist Shop Language, 267-269
and word address format, 264-267
See also Do loops
Nonmodal commands, 165
NUFORM, 358
Null, 463
Numerical control machinery, 1, 17
applications in industry, 14
binary numbers, 7-8
CNC machines, 5
history of , 2
input media, 5-7
objectives of NC, 11-13
tape formats, 8-10
Numerical control systems, 463
Cartesian coordinate system, 25
components, 18
computer use in, 356-391
dimensioning, 37
future of, 392-396
loop systems, 22-23
positioning systems, 29-30
positive and negative movement, 25
safety rules for, 412-413
servomechanisms, 22
setting the machine origin, 31-37
types of control systems, 18-21

Offline programming, 356-357, 463
Offset, 463
Offset numbers, 309
One hundred eighty-degree rotation tool changer, 83-85
Open-loop system, 22, 464
Optimize, 464
Optional stop, 464
Overshoot, 464

Parabola, 464
Parabolic interpolation, 464
Parity check, 465
Parsons, John, 2
Part program, 465
sample, 427-434
Part programmer, 397-398, 465
Perforated tape, 465
Pivot insertion tool changer, 85-87

Plotter, 465
Plunge and profile cutter, 52
Point-to-point systems, 18-20, 465
Polar rotation, 285-286, 299-300
and Machinist Shop Language, 287-290
and word address format, 290-293
Positioning/contouring, 466
Positioning systems:
absolute, 29
incremental, 30
Position readout, 466
Postprocessor, 375-385, 466
Preparatory function, 466
Process planning, 40
example of, 54-58
fixturing, 41
machine selection, 40-41
machining strategy, 41
NC setup sheet, 42
tool selection, 41-42
Program, 466
Programmed dwell, 466
Programmer, 466
Programming:
computer-aided, 357-375
computer graphics, 386-390
diameter vs. radius, 317
math for, 187-200
offline, 356-357, 463
postprocessor, 375-385
radius, 317
sample programs, 427-443
three-axis, 149-184
tool edge vs. centerline, 311
two-axis, 111-146
Programming coordinates, 99-107
Program stop, 466
Prototype shop, 393
Punched tape, 6

Quadrant, 467
Quicksetters, 314

Radius programming, 317
Ramp on/ramp off, 220
RAM (random access memory), 5
Random, 467
Rapid, 467
Reader, 467
Reaming, 47
Register, 467
REM statements, 157
Repeatability, 77, 467
Reset, 467
Robotics, 306
ROM (read only memory), 5
Row (tape), 467
RS-244 binary coded decimal, 9
RS-358, 10

Safety rules, NC, 412-413
Scanning systems, 386
Sequence number (code word), 468
Sequence readout, 468
Sequential, 468
Setup operator, 397
Significant digit, 468
Slow-down span, 468
Software, 469
Soft-wired controller, 5
Span, 469
Speeds:
cutting speed, 59-62
example of, 72-73
spindle, 311-313, 469
Spindle direct tool change, 89-92
Spindle movement, 25
Spindle speed (code word), 469
Spindle speeds, 311-313
Split bushing, 80
Stepper motor, 22
Storage, 469
Storage media, 469
Subroutines, 272
for cutter diameter compensation, 264
and Machinist Shop Language, 255-260
and word address format, 260-264

Tab, 469
Tab sequential format, 470
Tape, 470
Tape formats, 8
RS-244 binary coded decimal, 9
RS-358, 10
Tape lagger, 470
Tape leader, 470

Tape machinery codes, 410-411
Tape puncher, 6
Taper turning, 323-328
Tapping, 47
Thread hob, 49
Threading, 334-343
Three-axis programming:
additional G codes, 177-181
Fanuc controller, 181-184
and Machinist Shop Language, 154-165
programming task, 149-154
and word address format, 166-177
Tool changers, automatic, 81
multi-axis, 88-89
one hundred eighty-degree rotation, 83-85
pivot insertion, 85-87
spindle direct, 89-92
turret head, 82
Tool changes, 75-76
automatic, 79-81
manual, 77-78
numerical control lathe, 306-311
Tool edge programming, 311
Tool function, 470
Tool holders, 306
Tooling:
boring, 47
carbide inserts, 52-53
ceramic tools, 45
drilling, 46
end mills, 50-51
face mills, 51-52
high speed steel tools, 42
lathe tooling, 53-54
reaming, 47
tapping, 47
tungsten carbide tools, 42, 45
Tool length compensation, 470
Tool nose radius, 309
Tool nose vector numbers, 309
Tool offset, 95-96, 156-157, 470
Tool register, programmable, 96
Tool retention knob, 80
Tool selection, 41-42
Tool storage:
carousel magazine, 92-93
matrix magazine, 94
Trailing zeros, 113
Trailing zero suppression. *See* Zero suppression
Trigonometry, 187-191
for calculating cutter offsets, 191-197
for lathe programming, 198-200
Tungsten carbide, 42, 45
Turning, 317-323
Turning feedrate, 62, 64
Turning machines, 350-351
canned cycles for, 350, 422-426
circular interpolation on, 328-331
complete lathe program, 343-349
diameter vs. radius programming, 317
drilling on, 331-334
machine reference point, 316-317
taper turning, 323-328
threading on, 334-343
turning and facing, 317-323
See also Lathes, NC
Turnkey system, 471
Turret head, 82
Twist drill, 46
Two-axis programming, 145-146
drilling, Machinist Shop Language, 113-120
drilling, word address format, 121-126
Fanuc controller applications, 142-144
and Machinist Shop Language, 111-113
milling, Machinist Shop Language, 126-136
milling, word address format, 136-142
and word address format, 120-121

UNIAPT, 358
Up milling, 126-127
USASCII. *See* ASCII

Variable block format (tape), 471
Vector, 471
Vector feedrate, 471

Wire guided vehicle, 395
Word, 471
Word address format, 8, 471
and circular interpolation, 212-214
codes for, 120-122, 166-167, 406-409
cutter diameter compensation using, 229-232
do loops format, 251-254
and helical interpolation, 297-299
and linear interpolation, 207
and mirror imaging, 282-285

nested loops format, 264-267
and polar rotation, 290-293
subroutine format, 260-264
three-axis drilling in, 167-177
two-axis drilling in, 121-126
two-axis milling in, 136-142
See also Machinist Shop Language
Word length, 471
Work coordinates, 34-37, 313-314

X axis, 471

Y axis, 471

Z axis, 471
Zero offset, 472
Zeros, leading and trailing, 113
Zero shift, 472
Zero suppression, 472

COMPUTER NUMERICAL CONTROL: CONCEPTS AND PROGRAMMING

2ND EDITION

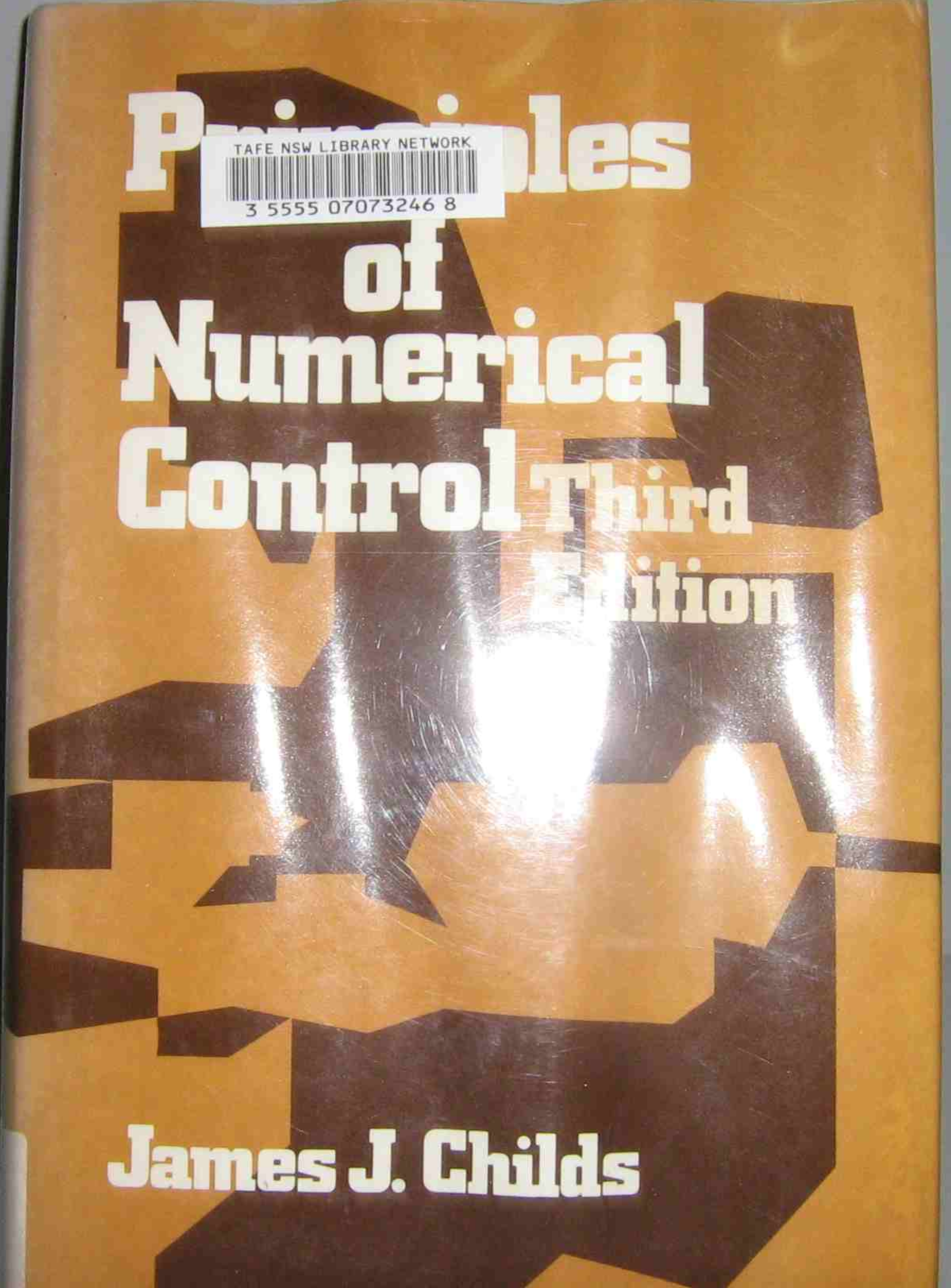
WARREN S. SEAMES

This revised edition provides students with a wealth of CNC knowledge from basic computer numerical control concepts and principles to everyday applications in machine tool technology.

SPECIAL FEATURES OF THE NEW EDITION

- New emphasis on the word address format in programming examples and problems.
- All new chapter on tooling shows the steps and documents utilized in planning the machining of a part prior to the actual programming of the machining process.
- "Special Fanuc Sections" have been added featuring part drawings and instructions to program cutting using a specific Fanuc controller.

ISBN 0-8273-3782-5
90000
9 780827 337824


TAFE NSW LIBRARY NETWORK
3 5555 07073246 8
of
Numerical
Control
Third
James J. Childs

PRINCIPLES OF NUMERICAL CONTROL
Third Edition
James J. Childs

The most comprehensive, clearly written introduction to the operation of numerical control systems available for the non-expert. This widely used, up-to-date reference provides all the information necessary for anyone—managers, students, supervisors, engineers—to gain a working understanding of its functions.

The emphasis is on CNC Systems and their programming capabilities, advantages, operation, and maintenance. Yet, a good portion of the hardwire material from past editions has been retained to satisfy those systems still in operation. A wide range of topics covers everything from specific information on plant requirements, to step-by-step economic justification procedures of systems (with tables listing comparative savings versus conventional hardwire equipment), to detailed part programming and basic computer operations. The latest in EIA Standards and Tables describe updated and standardized tape codes for easy reference. Extensively updated discussions of such topics as robotics, graphics, group technology, and DNC reflect advances and future trends in numerical control. Numerous examples of both APT and COMPACT II part programming provide a firm basis for more advanced programming courses for students.

An expanded glossary makes identification of terms an easy matter, and the exceptional clarity of the illustrations, photographs, and engineering sketches help reinforce concepts. At special request from our academic users of past editions, the end-of-chapter questions have been expanded to provide a more comprehensive study aid for students.

Third Edition

Principles of Numerical Control

James J. Childs

President, James J. Childs Associates, Inc.
Alexandria, Virginia

INDUSTRIAL PRESS INC.
200 Madison Avenue
New York, NY 10157

Library of Congress Cataloging in Publication Data

Childs, James J.
Principles of numerical control.

Includes index.
1. Machine-tools—Numerical control. I. Title.
TJ1189.C47 1982 621.9'023 81-20296
ISBN 0-8311-1135-6 AACR2

Sixth Printing

PRINCIPLES OF NUMERICAL CONTROL

Contents

Preface to Third Edition vii
Preface to Second Edition ix
Preface to First Edition xi
1. Introduction 1
2. How the Numerical Control Installation Operates 28
3. Tape—The Controlling Medium 53
4. Manual Part Programming 79
5. Computer-Aided Part Programming 110
6. Advantages of Numerical Control 145
7. Justifying Numerical Control Equipment 170
8. Organizing for Numerical Control 202
9. Future Trends 232
10. Numerical Control Terms, Definitions, and Standards 251
Index 297

CHAPTER ONE

Introduction

What is Numerical Control? Probably the most universally accepted definition of numerical control is that offered by the Electronics Industries Association which defines it as: "A system in which actions are controlled by the direct insertion of *numerical* data at some point. The system must automatically interpret at least some portion of this data." In a somewhat broader sense, numerical control may be considered as an extremely versatile means of automatically operating machines through the use of discrete numerical values introduced to the machine by some form of stored input medium such as a punched tape or directly from a computer. However, the real significance of numerical control lies beyond these definitions in the effects it has produced and is producing in almost every area of manufacturing, both direct and indirect. In fact, its influence has even begun to pervade the activities of the engineering product designer.

Granted that numerical control is a relatively new form of automation, the inevitable question arises as to how it differs from previous types of automatic control devices. With regard to its distinction when compared to former relatively versatile machines, the most notable point about numerical control is probably the inherent accuracy attainable. Thus numerical control electronic systems are generally more accurate than the machines they control, while the

Fig. 1–1. Numerical control lathes being utilized on a high production machining line. The key advantage in this case lies in their inherent accuracy.

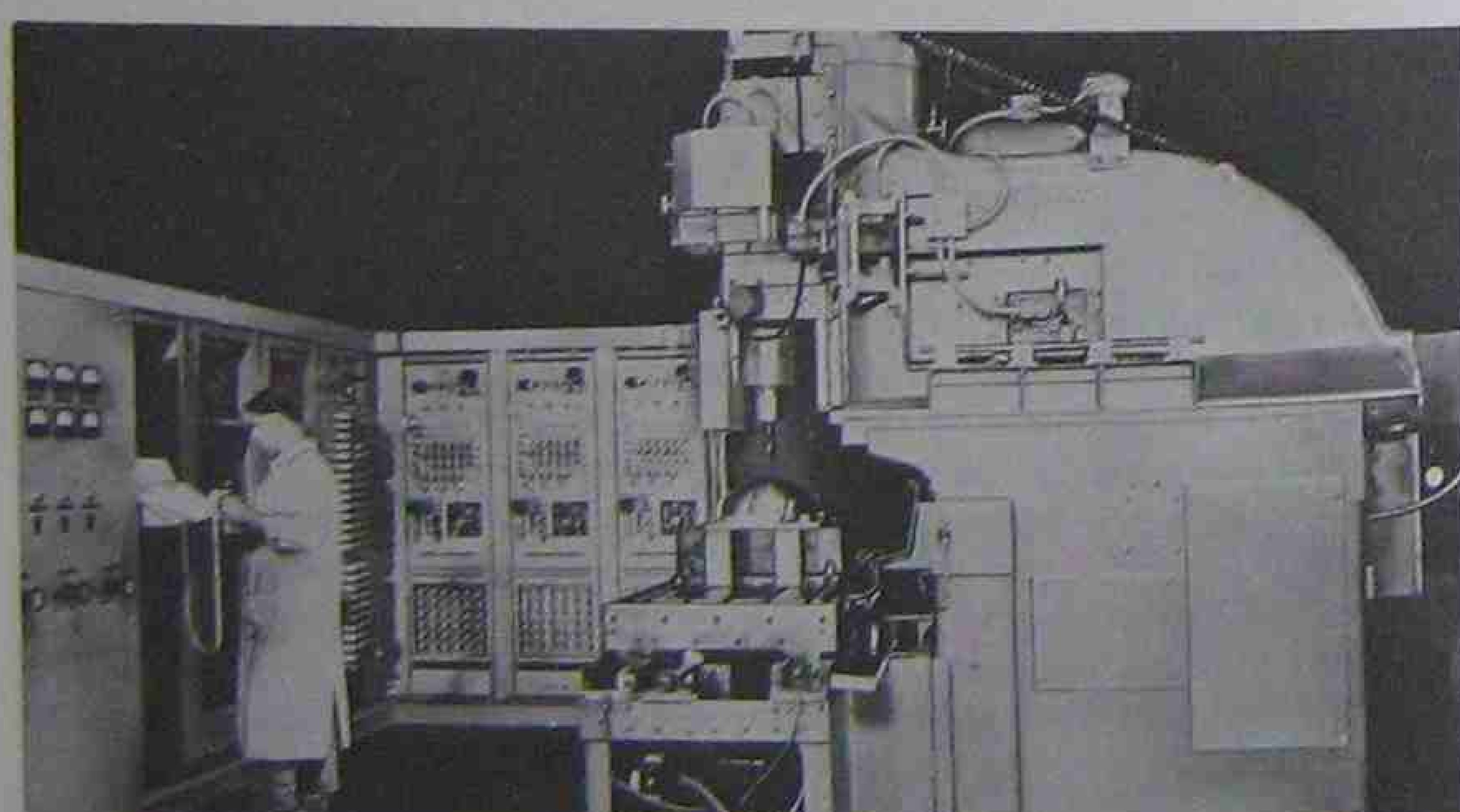
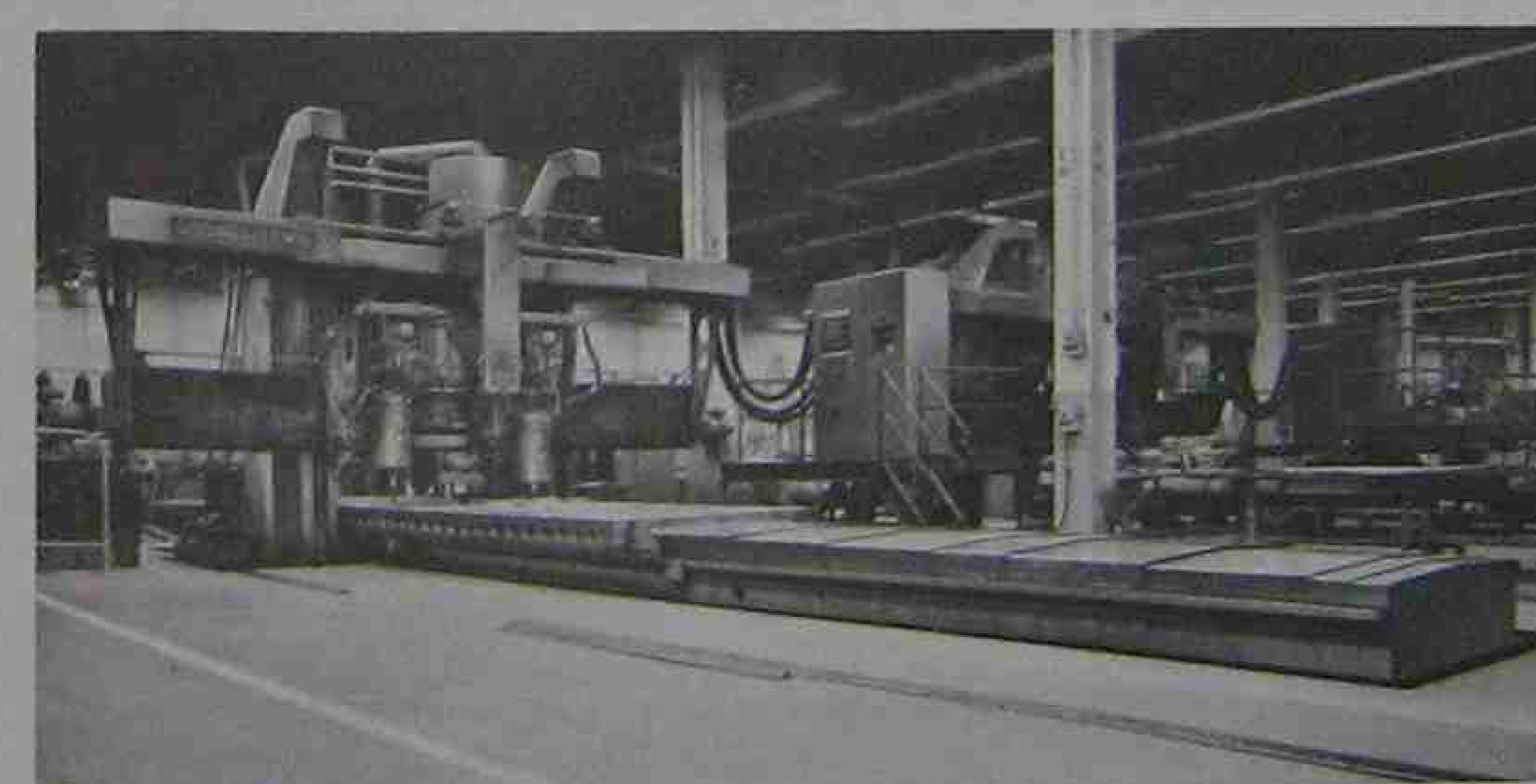


Fig. 1–2. Believed to be the first successful numerical control application, this milling machine was demonstrated in 1952 at the Servo Mechanisms Laboratory of the Massachusetts Institute of Technology.

accuracy of "conventional" automatic equipment is normally limited to the profile accuracy of a cam or to the mechanical setting of stops.

The popularity of numerical control, however, lies not in the fact that it is essentially a *different* and *successful* engineering concept but almost solely in its ability to make possible the manufacture of superior products *far more economically* than with manual control or previous types of automatic controls. Numerical control machines are *faster*, *consistently* more accurate, far more *versatile* and equally as efficient at ten minutes to five in the afternoon as at ten after nine in the morning. Details describing the economic advantages of numerical control will be explained more fully in a later chapter.

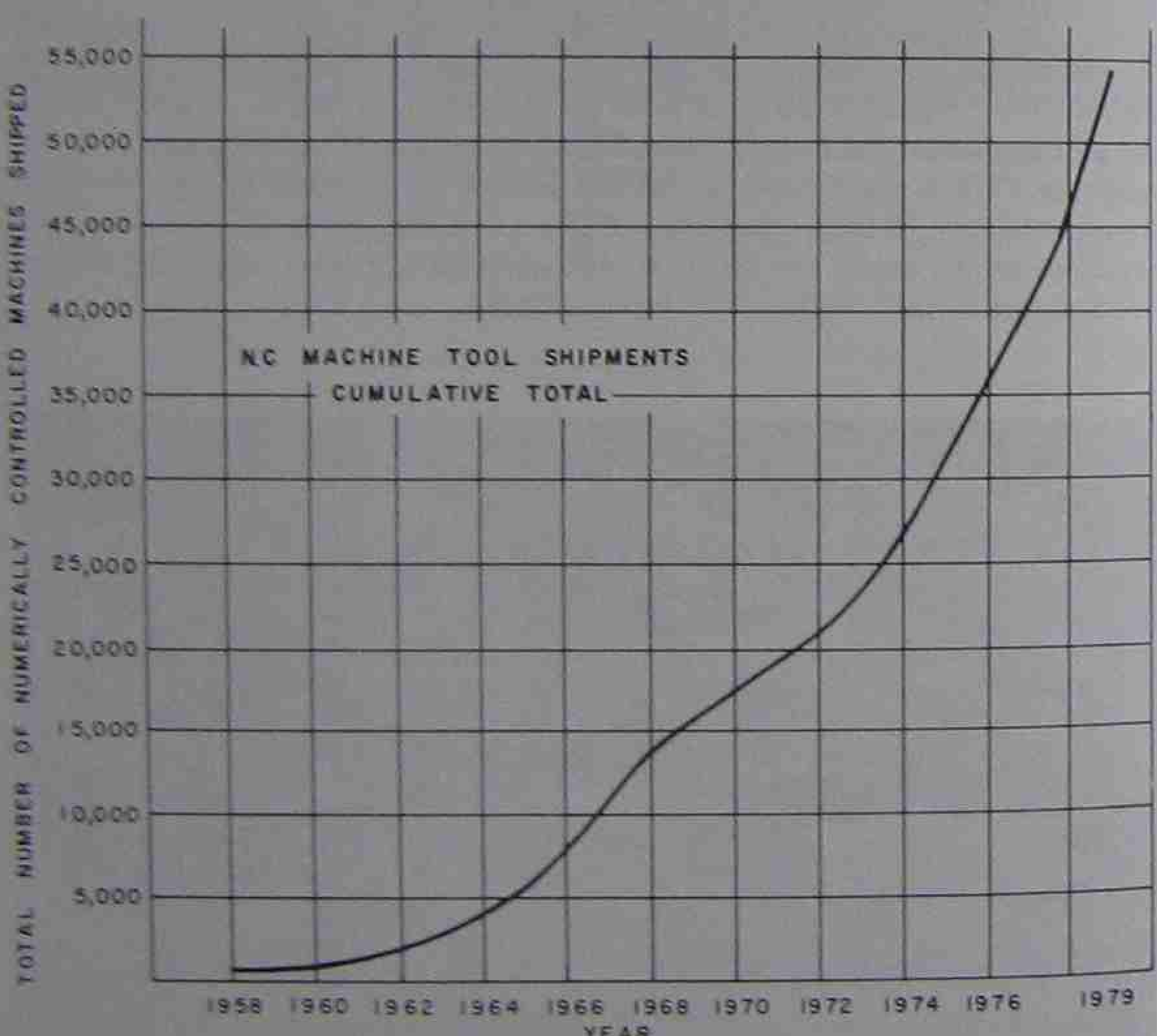
It may be helpful at this point to clarify a possible misconception that numerical control is merely another form of automation, which is restricted to applications involving large quantities of production, particularly large quantities of industrial parts. Although suitable for large quantities under special circumstances (see Fig. 1–1), numerical control is best suited to small lot production and has gained its reputation by substantially reducing set-up time which is a very significant portion of unproductive man-hours when manufacturing a considerable number of different parts in small lots. The so-called "economic lot size" may now be much smaller than ever before due to the reduced set-up time. It has even been found more economical to manufacture one part by numerical control means than by conventional methods in many instances. On the other hand, numerical control is *not normally prescribed* for the manufacture of large quantities of like parts that are to be run on a long-term continuous cycle. For this, a special-purpose machine, or special tooling arrangement, would probably be more suitable.



Courtesy of Republic Aviation Corp.

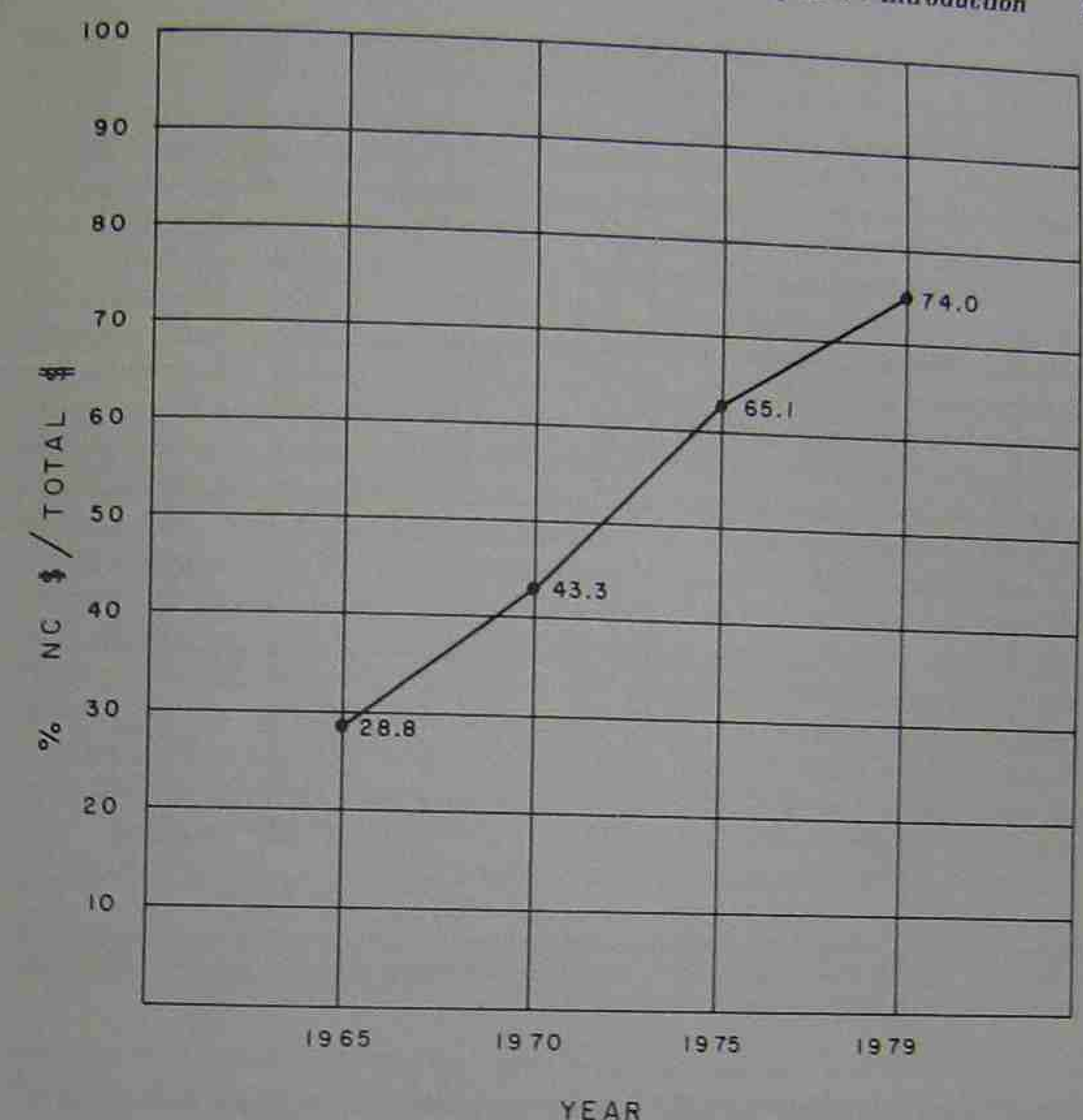
Fig. 1–3. The large numerically controlled skin milling machine shown here is capable of machining two parts simultaneously. This machine was part of the original 35-million dollar numerical control machine tool procurement program initiated in 1955 by the U.S. Air Force.

What is the History of Numerical Control? As with many inventions, it came into being because of a need for manufacturing a product by a far simpler method than then existed. The U.S. Air Force found itself in this position shortly after World War II when it was faced with the complex machining of aircraft components and inspection fixtures to close accuracies on a repeatable basis. During this period Mr. John T. Parsons had been working on a project for developing equipment that would machine flat templates for inspecting the contour of helicopter blades. A proposal covering a machine to prepare these templates was presented to the Air Force by the Parsons Corporation of Traverse City, Michigan, and resulted in a development contract in 1948. In 1949 Parsons was joined by MIT as a major subcontractor on the project. Development work continued, and in 1951 MIT was awarded a prime contract which resulted in the successful demonstration, in 1952, of a three-motion milling machine as shown in Fig. 1–2. The following three years were largely devoted to hardware refinements and mathematical techniques for tape preparation. After



Source: *U.S. Department of Commerce*

Fig. 1–4. The rate of growth of numerical control in the United States has been rapidly increasing since 1960.

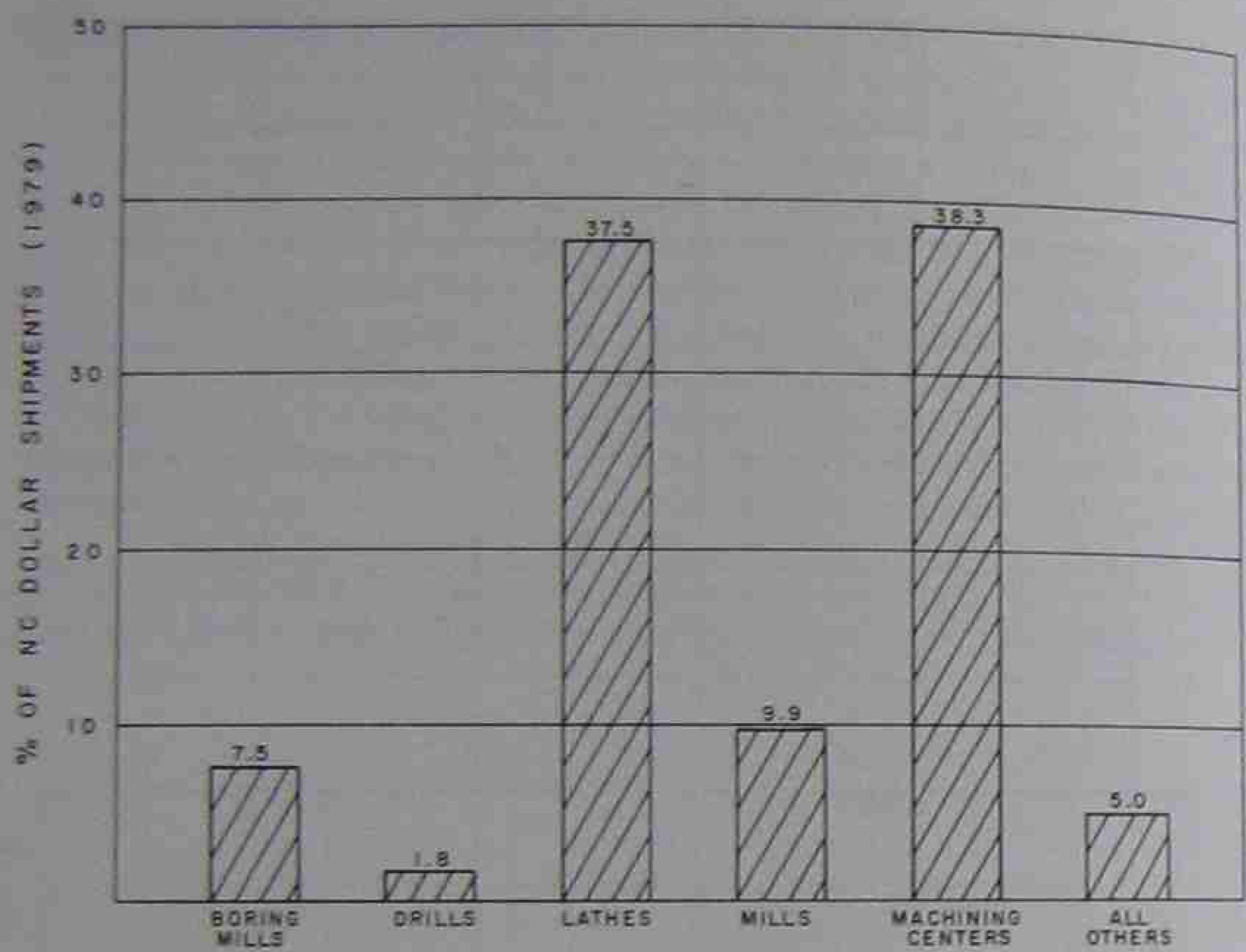


Source: *U.S. Department of Commerce*

Fig. 1–5. The graph describes the percentage of dollars of numerical control shipments as compared to the total dollar shipments of all metal-cutting machine tools of a comparable type, that is, considering mills, drills, lathes, boring mills, and machining centers. Excluded would be machines such as gear cutters, honing and lapping machines, and grinders. Even if all metal-cutting machine tools are considered, the percentage has risen from 15.2% in 1965 to 30.7% in 1979.

studying the applicability of numerical control machining for high-speed aircraft structures, the Aerospace Industries Association,[1] an organization comprised of aerospace manufacturers, recommended to the Air Force that forthcoming machines be equipped with numerical controls and in 1955 the Air Force began awarding some 35 million dollars for the manufacture of approximately 100 numerically controlled milling machines. A significant number of these were large contour skin-milling machines measuring up to 45 feet in length, such as that shown in Fig. 1–3, and costing almost a half million dollars.

[1] Headquarters at 1725 DeSales St., Washington, D.C. 20006.



Source: *U.S. Department of Commerce*

Fig. 1–6. The percentage breakdown of numerical control machine tools shipped in 1979 is shown. Lathes and machining centers predominate on an almost equal basis, although lathes had a relatively slow start in the NC field.

Many of these machines are still actively producing aircraft and missile components. Credit is undoubtedly due the Air Force planners for their prophetic decision.

It was not until approximately 1960 that numerical control began to appear on a reasonably wide commercial scale. At that time the trend of equipment procurement was heavily weighted toward the less expensive point-to-point variety such as is utilized with drill presses rather than the more complicated form of continuous-path control used with milling operations. (Differences between these two types of numerical control will be explained later in this chapter.) The growth in the number of numerically controlled machines, while not phenomenal, has been accelerating rapidly. Approximately 50% of these are of the point-to-point variety, although the percentage of point-to-point machines being shipped each year is diminishing and now stands at less than 10% of the total being shipped; and the bulk of these are for punch presses. Clearly the tide has turned in favor of contouring systems, which are also capable of performing point-to-point operations. The chief reason is that the difference in the cost between a contouring and point-to-point system has narrowed substantially due to the development of lower cost electronic components such as microprocessors. It might also be added that the cost of nu-

merical control systems has decreased to almost one-third of what they were a decade ago while the reliability and capability of the systems have increased.

The growth of shipments of numerically controlled (NC) machines in the United States, according to the U.S. Department of Commerce, is shown in Fig. 1–4. The number of NC machines operating in the United States is probably somewhat higher since these figures, issued by the Commerce Department, do not include imports (principally from Japan) nor retrofits nor a quantity of NC machines sold by control system builders. These additions together with the attrition of older machines result in an estimated figure of 60,000 machines operating in the United States as of October, 1980. While this number may appear small if one considers that there are approximately 2,200,000 machine tools in the field, there are some extenuating considerations. First, a large share of the 2,200,000 machine tools are idle or in storage; also, many are of a low value nature or obsolete. What is significant is that the trend in the shipments of NC machine tools has been increasing at a steady and high rate. Figure 1–5 describes the dollar shipments of NC metal-cutting tools as a percentage of the total when considering comparable type tools (i.e., mills, drills, lathes, boring mills, machining centers); excluded would be machine tools that are not readily suitable for NC, such as gear cutters, honing and lapping machines, and grinders. It will be noted that, in 1979, almost three-quarters of the dollar shipments were for NC machines. Even if *all* metal-cutting tools were considered the percentage has risen from 15.2% in 1965 to 30.7% in 1979. Figure 1–6 describes the breakdown of types of NC machine tools shipped in 1979. It is interesting to note that, while lathes vie with machining centers on an almost equal basis, at 37.5%, they had a very slow start and accounted for only 7.4% of the total dollar NC shipments in 1961.

How Does it Work? Although a more detailed explanation will be offered in the next chapter, it is appropriate at this point to describe the fundamental components that comprise a numerical control installation, together with their

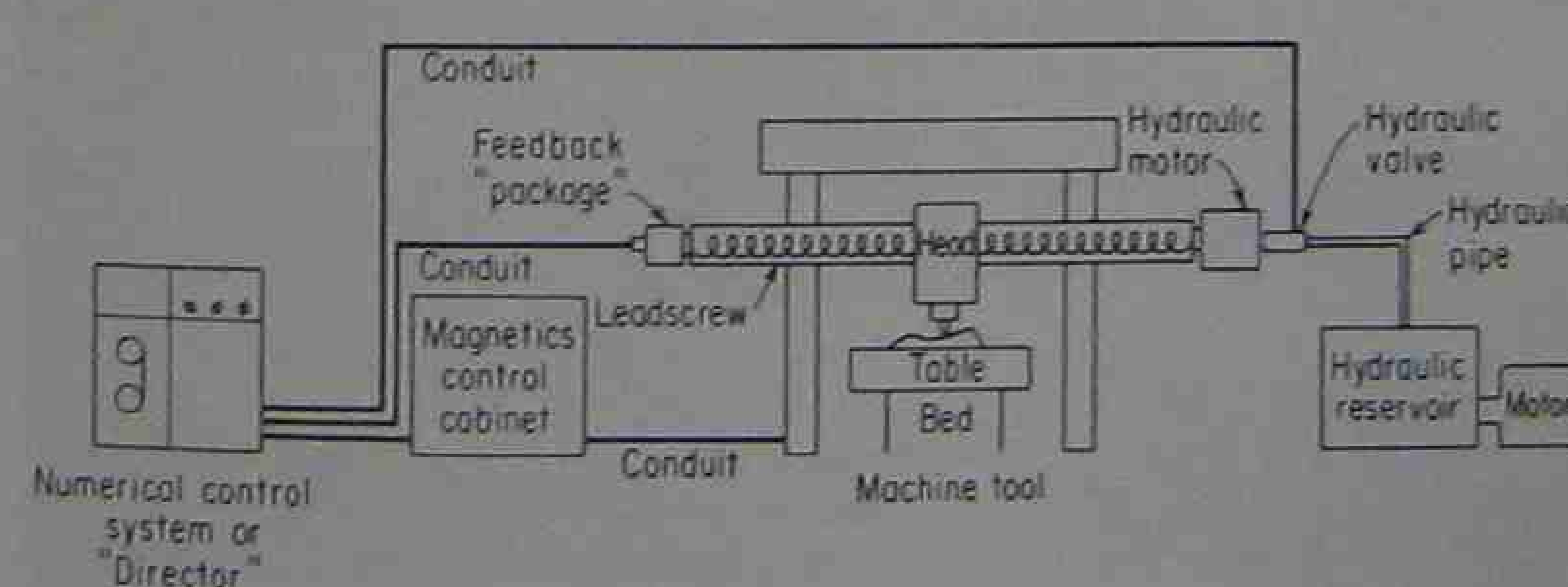


Fig. 1–7. Major components comprising a numerical control machine tool installation. Although hydraulic units are shown in this example, electrical power drives are also utilized. Assuming this to be a 3-motion (3-axes) machine, the hydraulic motor, valve, and feedback "package" would be duplicated for the vertical motion of the head and table motions.

Courtesy of The Bunker Ramo Corp.

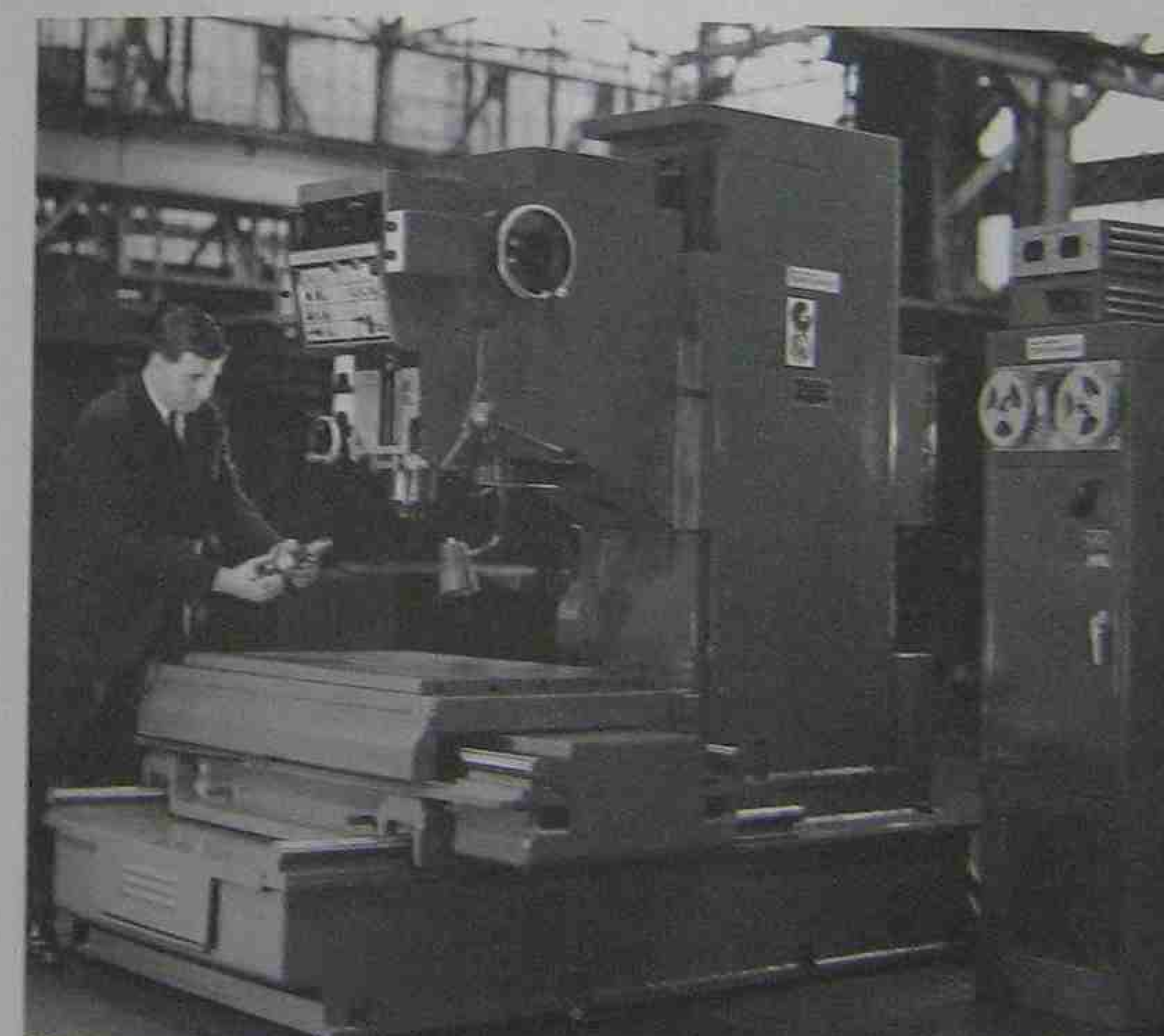
Fig. 1–8. The size of the machine does not determine the size of the control system. Thus the director shown in the left background is also capable (with slight modification) of controlling the large skin mill shown in Fig. 1–3. The operator's console is shown in the right foreground.

basic functions. The major components of a numerical control installation are shown in Fig. 1–7.

The numerical control system or *director* consists of a mechanism for automatically reading the tape plus the electronic hardware and software[2] for converting the coded tape information into machine tool instructions. The

[2] Software is another term for the computer programs that operate with computer numerical control (CNC) systems. CNC systems are explained in Chapter 2.

director is the heart of the numerical control operation and constitutes a significant portion of the overall equipment cost of a numerical control installation. The size of the machine has little effect on the size or complexity of the director for essentially the same director is capable of controlling a machine of the size shown in Fig. 1–3 or of the size shown in Fig. 1–8. The higher the cost of the machine tool, the lower the percentage cost of the control system. Actually the cost relationship between the control system and the machine tool, or the machine tool and any of the numerical control components, is of little significance. Nor is a comparison between the cost of a conventional piece of equipment and a similar numerical control piece of equipment of any real significance. What is important is the comparative costs of the product that would be turned out by both types of equipment. If, as is often the case, a numerical control machine can produce four times the amount of work in a given time period as a conventional machine and the cost of a numerical



Courtesy of Pratt & Whitney Co., Inc.

Fig. 1–9. Controls for manual operation of relatively simple machines such as this drill press may be incorporated in a control box attached to the base of the machine or as an integral part of a major component of the machine such as the head, as shown in this illustration.

control machine is twice that of the conventional machine, the numerical control machine still offers twice the economy of the conventional machine. This is to say nothing of the better quality, lower tooling costs, and higher accuracy attainable with the numerical control machine.

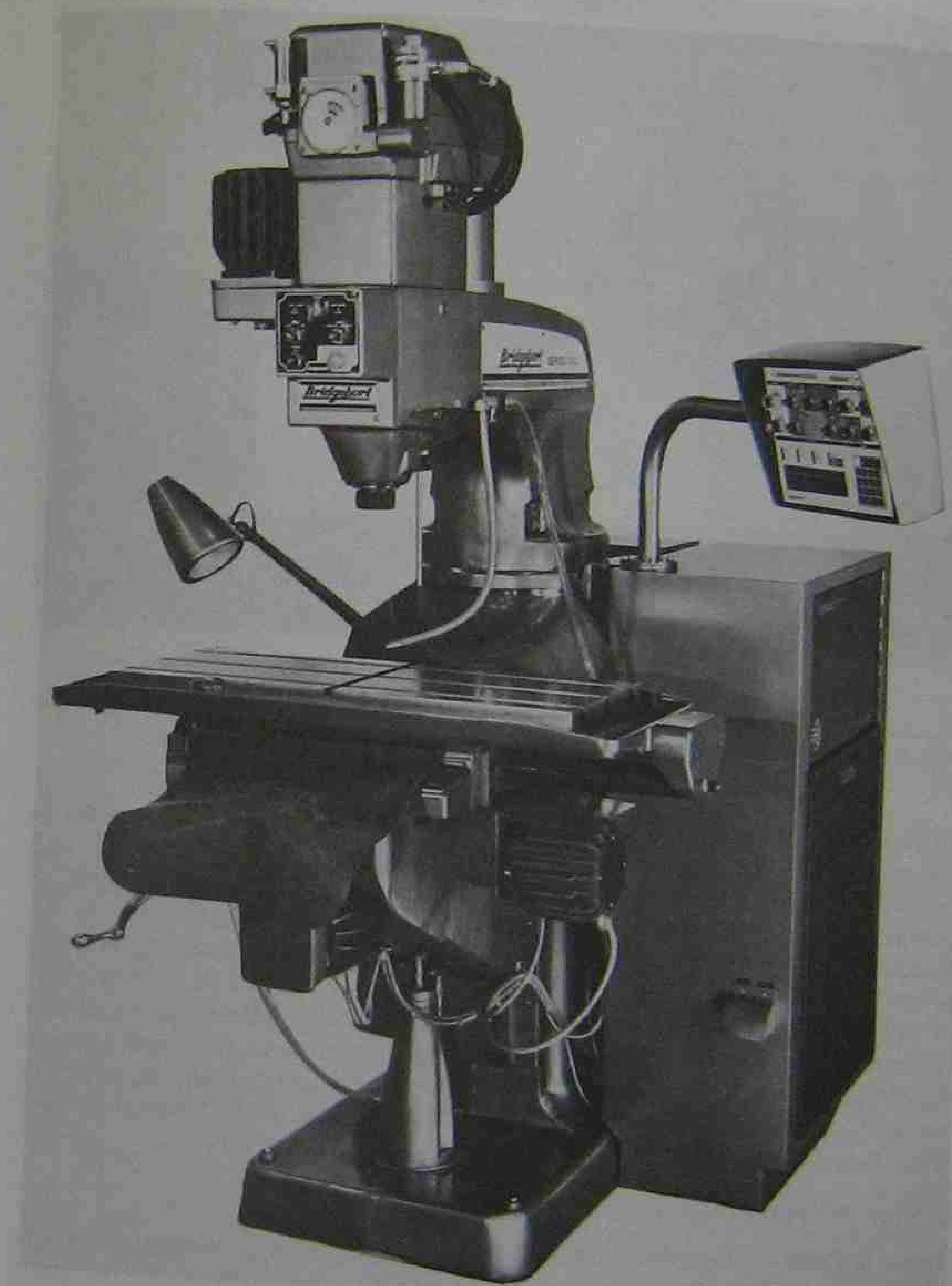
Between the director and the machine are a number of nonelectronic items which are not normally found with conventional machine tools. The *operator's console* which is located as close to the machine as possible contains the necessary controls for operating the machine manually, thus offering the facility for "setting-up" workpieces. The console may be on a *pedestal* or *rolling cabinet* or in an overhanging *pendant*. The main distinction between this console and the "operator panels" on conventional machine tools is in the increased number of command switches and buttons, some of which may be duplicates of those on the panel of the director. If convenient, the director and operator's console may be placed adjacent to each other or contained in the same cabinet. Normally, however, the director requires far less operator attention than the console and, with some types of machine tools, it may be advantageous not to locate the director in the proximity of the machine tool drive motors because of possible electrical "noise" pick-up and magnetic interference. Thus, the relative positions of director and operator's console may be as shown in Fig. 1–8. The console is shown in the right foreground of the photo while the director is shown to the left and rear of the photo. Smaller machine tools such as the numerically controlled drill press shown in Fig. 1–9 incorporate the operator's console in the head section of the machine tool. Newer CNC machines, utilizing microprocessors, can house both the director and the operator's console in a single unit. The small mill in Fig. 1–10 has the director and operator's console stored in a pendant, while the lathe shown in Fig. 1–11 houses both the director and the operator's console as an integral part of the machine.

Another major item of the system is the *magnetics cabinet*. This unit contains the magnetic relays and starting switches for controlling the flow of electrical power to the hydraulic and coolant pumps, spindle motor, and other electrical devices. These devices within the magnetics "box" are controlled by signals from the director.

The hydraulic pump and reservoirs supply the fluid power for operating the hydraulic motors which are attached to the lead-screws which in turn move the head and table. The hydraulic fluid is regulated by a valve which is controlled by the director. The amount of fluid and its rate of flow determine the distance moved and rate of travel of the moving elements.

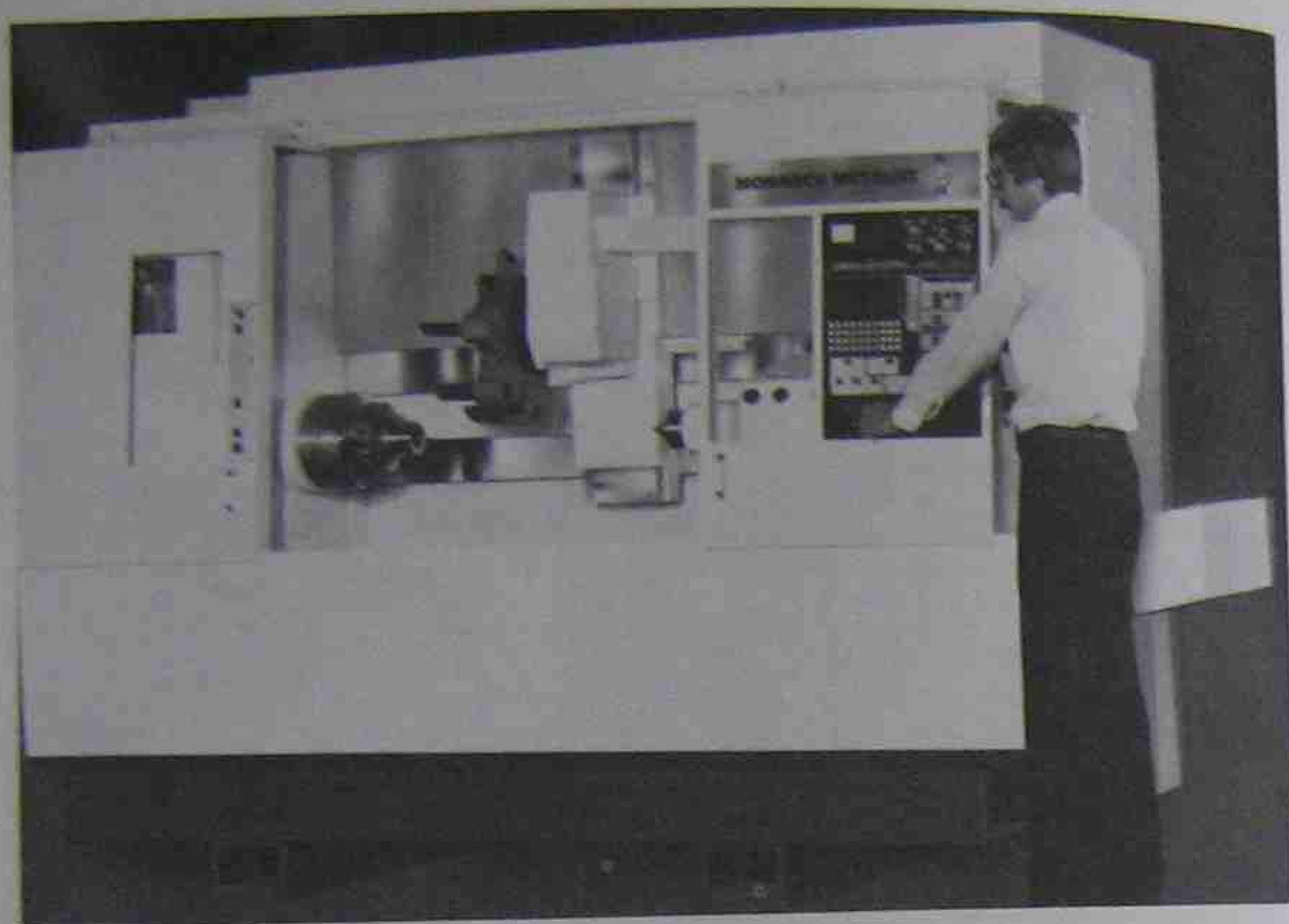
As the lead-screw rotates, the *feedback device*, which may be attached to either end of the lead-screw or be entirely independent of the lead-screw, records the movement and sends signals back to the director. The director then compares these signals with the input instructions as described by the tape, and continues to send activating signals to the valve until a balanced condition exists between the *command* signals as read from the tape and the *feedback* signals as generated by movement of the machine components. (This will be covered more fully in the next chapter.)

The primary difference when employing electrical instead of hydraulic



Courtesy of Bridgeport Machines Division of Textron

Fig. 1–10. The CNC machine shown has both the director (control system) and the operator's console incorporated in the relatively small pendant shown at the right side of the machine. The magnetics cabinet is located below the pendant.



Courtesy of Monarch Machine Tool Company

Fig. 1–11. This numerical control lathe combines the director (control system) and the console into a single integral unit.

motors is in the drive mechanisms which are attached to the lead-screws. In the example shown, electric motors would replace the hydraulic motors and valves. The hydraulic pump and reservoir would be replaced by an electrical power conversion unit for supplying electricity, in the proper form, to the electrical drive motors; although hydraulic drives can satisfy certain requirements best, the electric drive has become more popular.

Also, the lead-screw may be replaced by a hydraulic cylinder or a rack and pinion, although these drive mechanisms are uncommon. Feedback devices may be of a linear type which are affixed to the table or other sliding machine element, or of the small rotary type which are attached to the lead-screw. The latter type is the more popular and generally less expensive.

POINT-TO-POINT VS. CONTOURING CONTROL

What is Point-to-Point Numerical Control? This form of numerical control is usually associated with equipment that performs operations at specific positions and does not affect the work-piece when moving from one point to the next. A common example of a point-to-point numerically controlled machine tool is a simple drill press. The cutting tool or part is directed to specific points and, after arriving at the proper position, the machine tool automatically performs its drilling operation. After the hole has been drilled, the spindle then automatically moves to the next position. The exact path that the spindle takes

in moving from point to point is immaterial, providing, of course, the time required is reasonable and the spindle does not collide with either the part or holding fixture. Figure 1–12 illustrates the motions that would occur with a simple drill press operation. In accordance with the coded tape instructions, the drill would be moved to a position directly over the center of the hole *A* to be drilled. Once in position, the drill is automatically lowered, at a predetermined speed and feed which may be mechanically set at the start of the machining cycle or regulated from coded instructions on the tape. After drilling hole *A*, the head moves out of the hole at a *rapid retract* rate. The drill then moves to point *B*, and the cycle is repeated. The same sequence is followed for holes *C*, *D*, and *E*. After the completion of the last hole the drilling head may be instructed to return to the original starting point or to another parking position and automatically turn itself off. Two means of approaching the designated spot are shown in Fig. 1–13. The drill, or head, may follow a right-angle pattern in moving from *A* to *B* as shown at the left or it may move at a 45-degree angle until in line with the new position and then move directly to it along one axis.

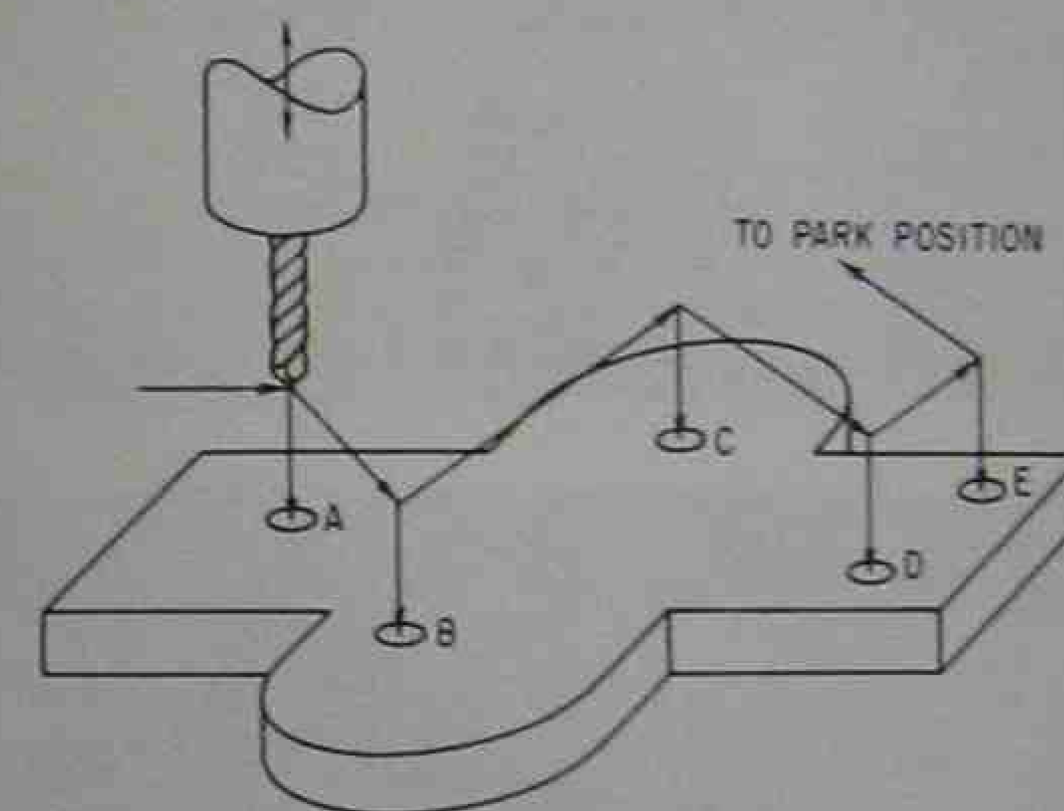


Fig. 1–12. Point-to-point controlled motions for drilling holes with a simple type of numerically controlled drill press.

Most point-to-point machines also have the capability to mill straight cuts, that is, to move along one set of ways at a controlled feedrate.

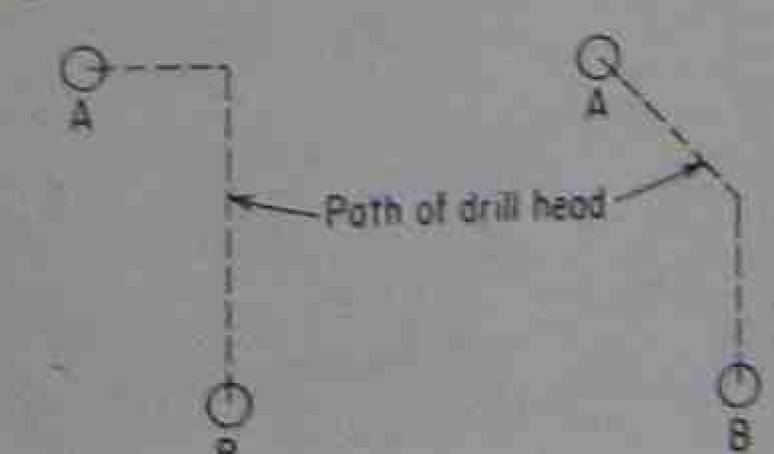
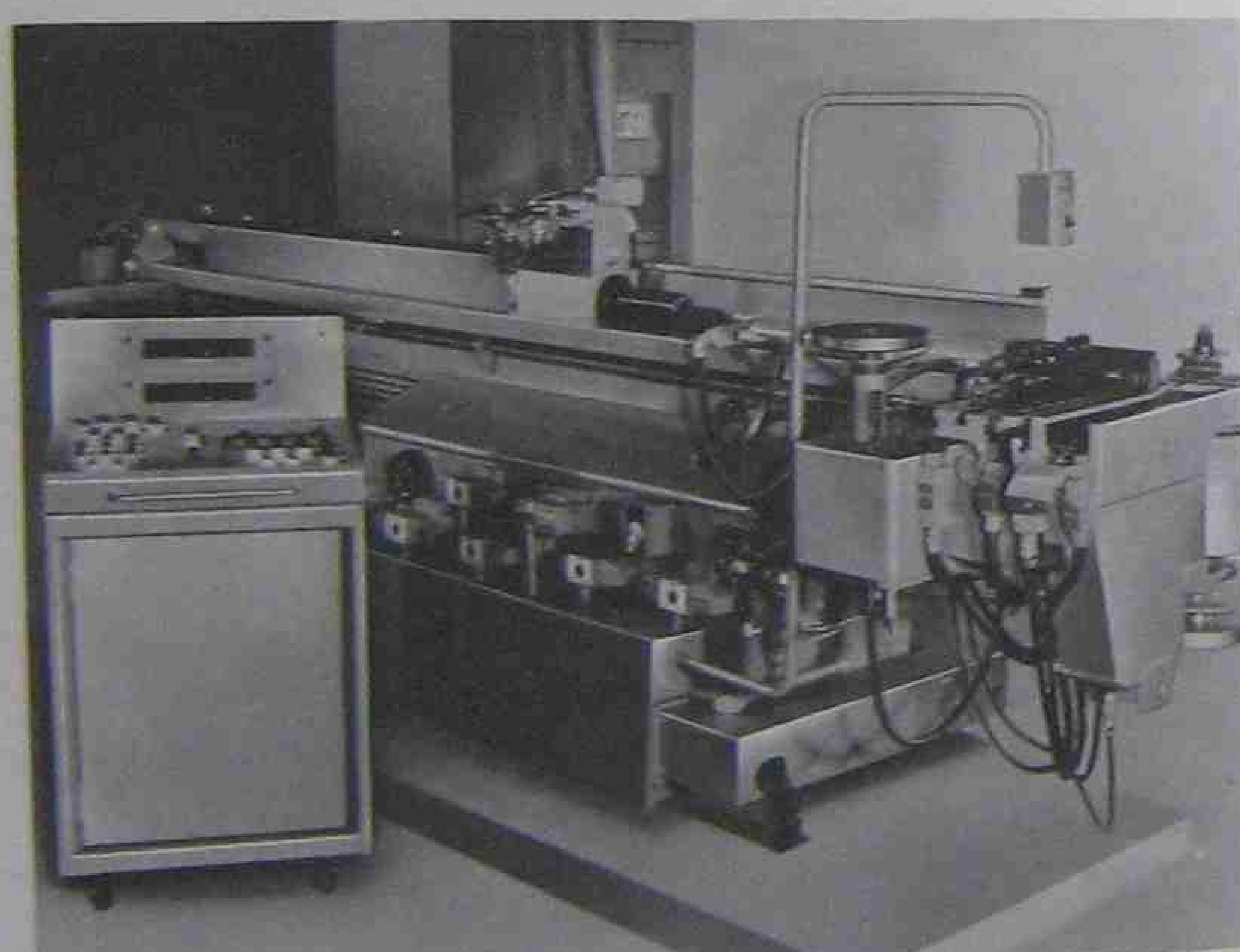


Fig. 1–13. Two means of approaching a point are shown above. (Left) A right-angle pattern is utilized. (Right) Here, the cutter moves at a 45-degree angle and, when in line, moves directly to the prescribed point.

Courtesy of Warner and Swasey

Fig. 1–14. The turret punch shown automatically positions the sheet material, indexes to the proper punch, and then performs the punching operation from prescribed tape data. The machine is capable of 235 punches per minute. An example of point-to-point or positioning numerical control. This particular model also has contour nibbling capability. Other models are equipped with a contour laser cutting capability.



Courtesy of Datex Corp.

Fig. 1–15. Positioning or point-to-point control is used with this automatic tube bender. The tube, after being fed into the machine, is positioned and then bent to the proper angle from coded tape instructions.

Courtesy of Kaltenbach, Inc.

Fig. 1–16. This cut-off saw, operating on a point-to-point principle, can handle various sizes of material and parts automatically, in addition to programmed cutting operations.

Courtesy of E. G. Heller's Son, Inc.

Fig. 1–17. A numerically controlled steel-fabricating machine. This machine automatically feeds, measures, handles, and punches holes in steel beams.

The point-to-point concept is by no means restricted to drilling or even cutting applications. Figures 1–14 and 1–15 show a numerically controlled turret punch press and a tube bender, respectively. With regard to the punch press, the work-piece is automatically positioned under the selected punch which then automatically punches the proper size and shape hole. Indexing of the turret is automatic and prescribed by the tape. Operation of the tube bender consists of positioning the tube and then bending it to the proper angle.

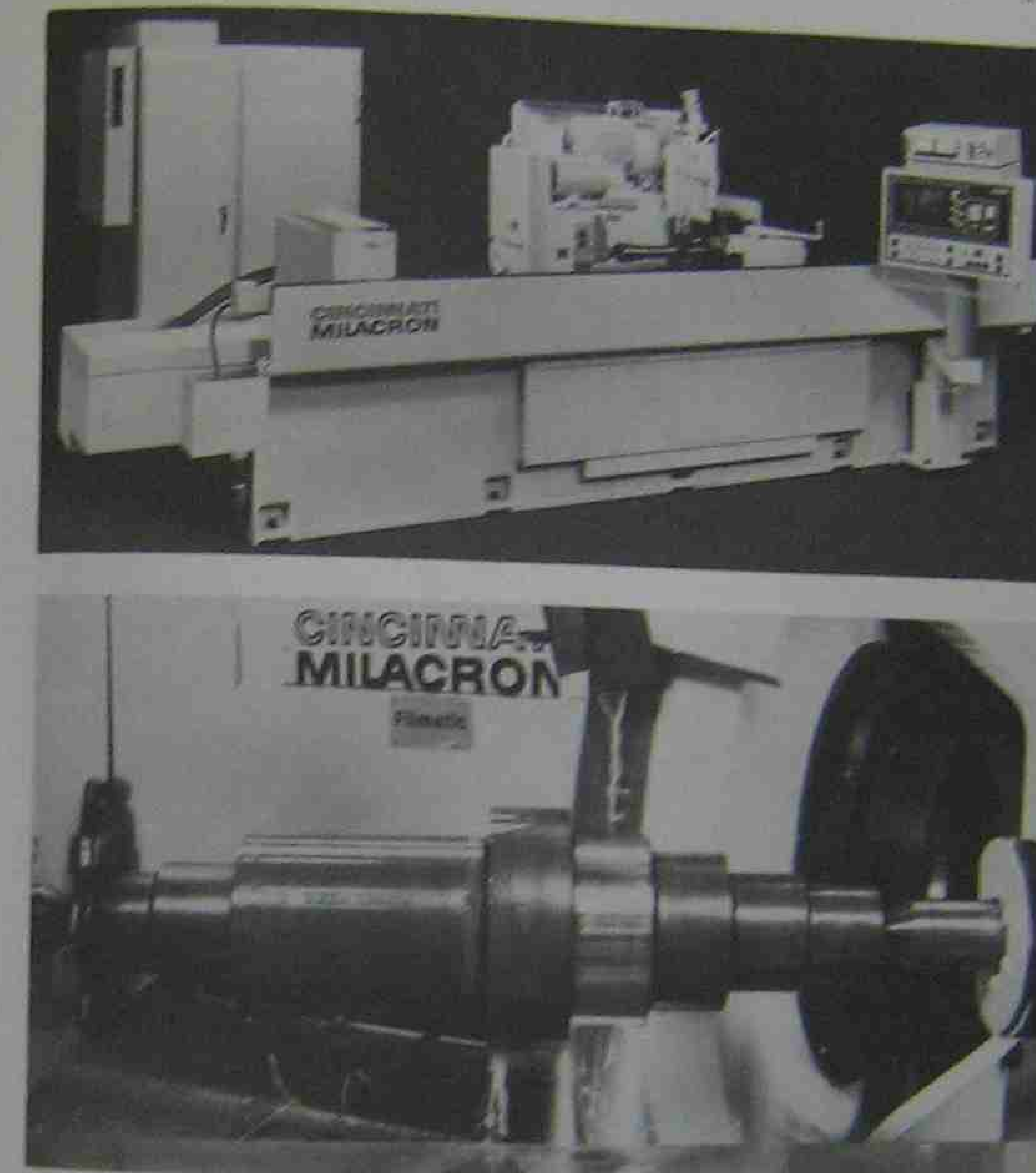
Numerical control has also been applied to cut-off saws and metal-fabricating machines. Figure 1–16 shows a CNC operated machine that can be programmed in accordance with the length of the parts to be cut, the widths of the parts, the number of pieces desired, and rapid traverse and feeds; most significantly, the parts are handled automatically in accordance with the input program. The steel-fabricating machine shown in Fig. 1–17 operates in much the same way, only, in this case, holes are punched instead of metal being cut.

Another application is shown in Fig. 1–18. Grinding has never been a very popular NC application, chiefly owing to the fact that a grinder is usually performing its grinding function a relatively high percentage of the time. NC has been most beneficial with applications wherein a high percentage of the machine cycle time is spent in nonproductive motions, such as moving the cutting tool from one position to the next, or in applications requiring a good deal of set-up time, e.g., drilling or milling operations. However, CNC, which allows for quick adjustments and rapid traverses, has improved the prospects for grinding as an NC application.

Another point-to-point application is shown in Fig. 1–19. In this case the control system adjusts the amount of feed and the bending die moves to adjust for the amount of bend required.

What is Continuous-Path or Contouring Control? Unlike point-to-point numerical control whereby the movements or the path taken between operations is relatively unimportant, the route when considering a continuous-path application is of consequence since the work-piece is being affected throughout the entire movement. In the sketch shown in Fig. 1–20, the cutting tool is moving along the perimeter of the part and is cutting metal continuously as it moves along its prescribed path. The entire travel must therefore be controlled to close accuracy both as to position and velocity. The control system or director, consequently, is more complicated and more costly.

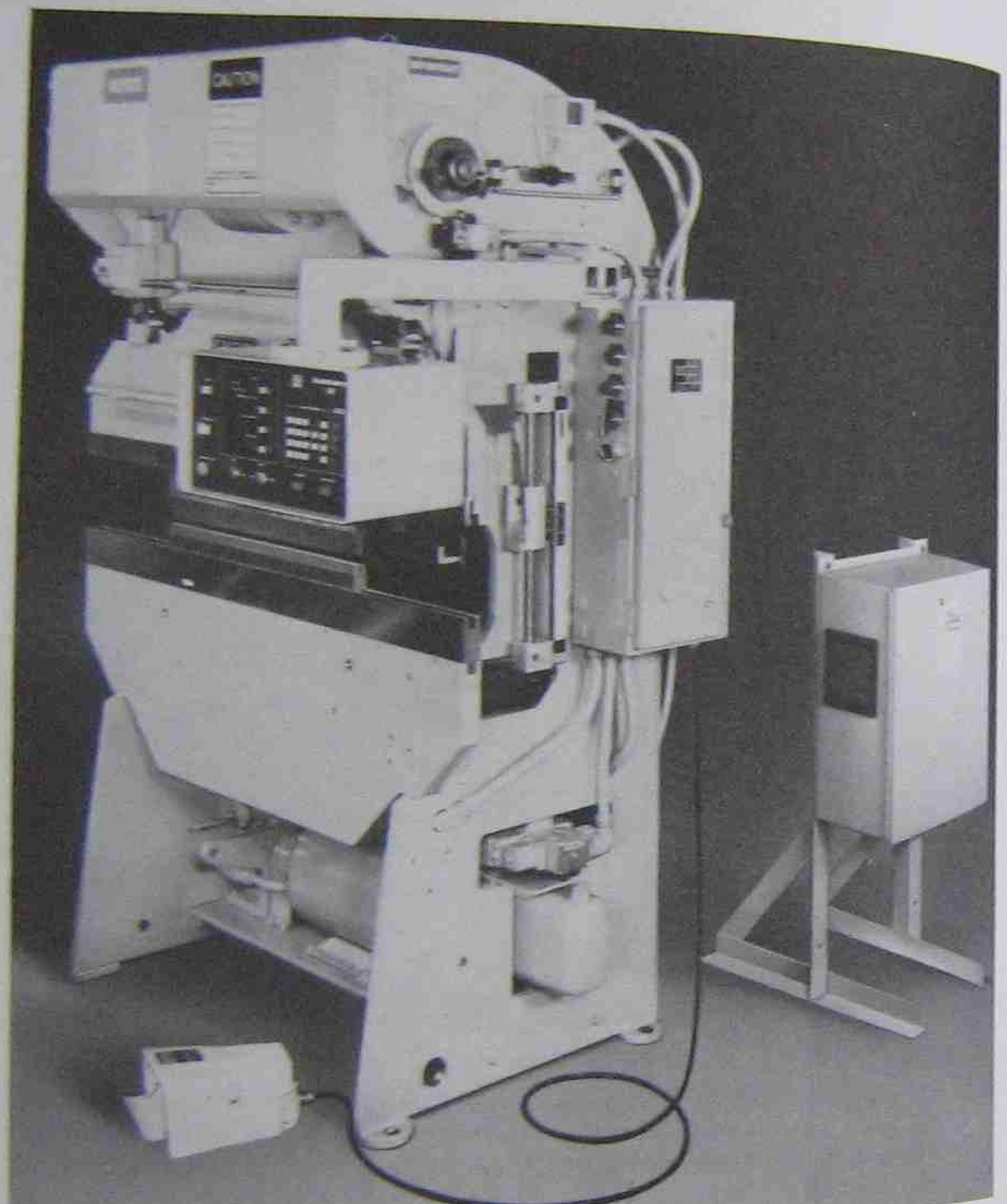
To date milling machines, machining centers, and lathes have proven to be the most popular applications of continuous-path numerical control. The machine tool shown in Fig. 1–8 is of the continuous-path variety as is the large skin milling machine shown in Fig. 1–3. The adjective skin refers to the machine's designed purpose, which is to machine or sculpture large wing skin panels. The two large pocketed aluminum sheets shown in Fig. 1–21, which are being machined on the contour skin milling machine, are to be assembled into the wing structure of an Air Force supersonic jet aircraft. Continuous-path milling may also be utilized for the machining of smaller and more complicated parts such as those shown in Fig. 1–22.



Courtesy of Cincinnati Milacron

Fig. 1–18. Numerical control grinding is another application that, although never very popular, should now become more so owing to CNC controls. The bottom photograph shows a closeup of the step grinding operation.

Although not as numerous as numerically controlled milling machines the contour lathe is becoming increasingly popular. Set-up time as well as accuracy, and particularly finish, are improved significantly when compared to conventional-type tracing lathes. Tapers, plus arcs and other curved cuts, in



Courtesy of Hurco Manufacturing Co., Inc.

Fig. 1–19. The numerical control pressbrake shown has two axes of control; one for gauging (feed) and one for the depth of the stroke, which determines the angle of the bend. Figure 1–19b shows a closeup of the feedback slide and operating mechanism.

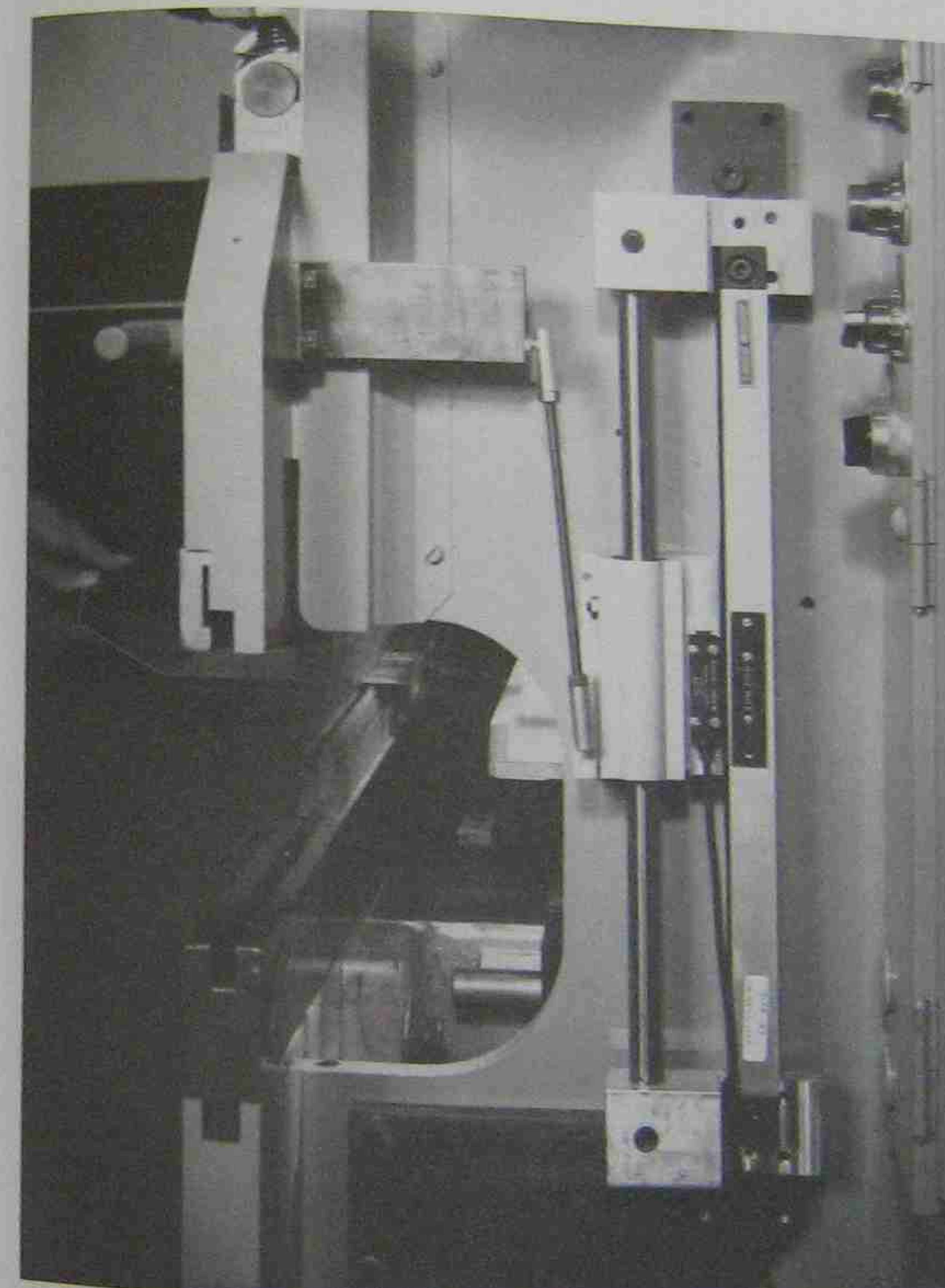


Fig. 1–19b.

addition to automatic thread cutting are possible from tape commands. Figure 1–23 shows a large numerically controlled contouring engine lathe.

As with point-to-point control, continuous-path control has numerous applications other than metal cutting. A salient example is the numerical control fabric-cutting machine shown in Fig. 1–24. The director or control system is essentially the same as that used with a milling machine or lathe, only, in this case, the motions of a cloth-cutting knife are controlled instead of a metal-cutting tool.

Figure 1–25 illustrates the use of numerical control with electro-discharge machining (EDM). In this case an electrically charged wire cuts a highly accurate path through metal. The path is controlled by a numerical control contouring system. A close-up view of a part being cut is shown in Fig. 1–26.

Another continuous-path application is described in Fig. 1–27. The carriage equipped with the multiple torches is controlled over a path describing the outline of the parts to be cut. Although not nearly as numerous as numerically controlled contour milling machines and machining centers, this development has been applied to torch cutting large steel plate required in shipbuilding and heavy steel fabrication.

Special Note: It is both possible and practical to utilize continuous-path controlled machines for performing point-to-point operations whereas point-to-point systems are restricted, in a practical sense, to point-to-point operations. It should also be noted that almost all of the newer CNC systems being manufactured are of the continuous-path type.

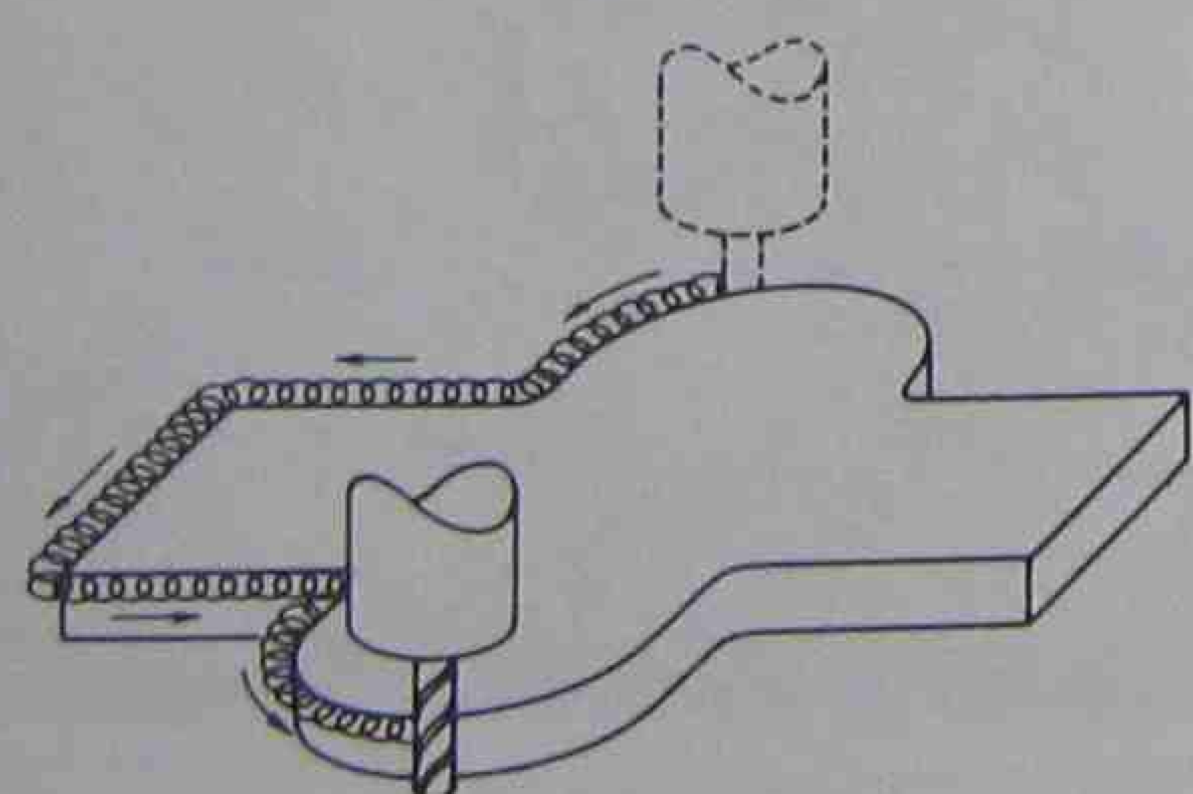
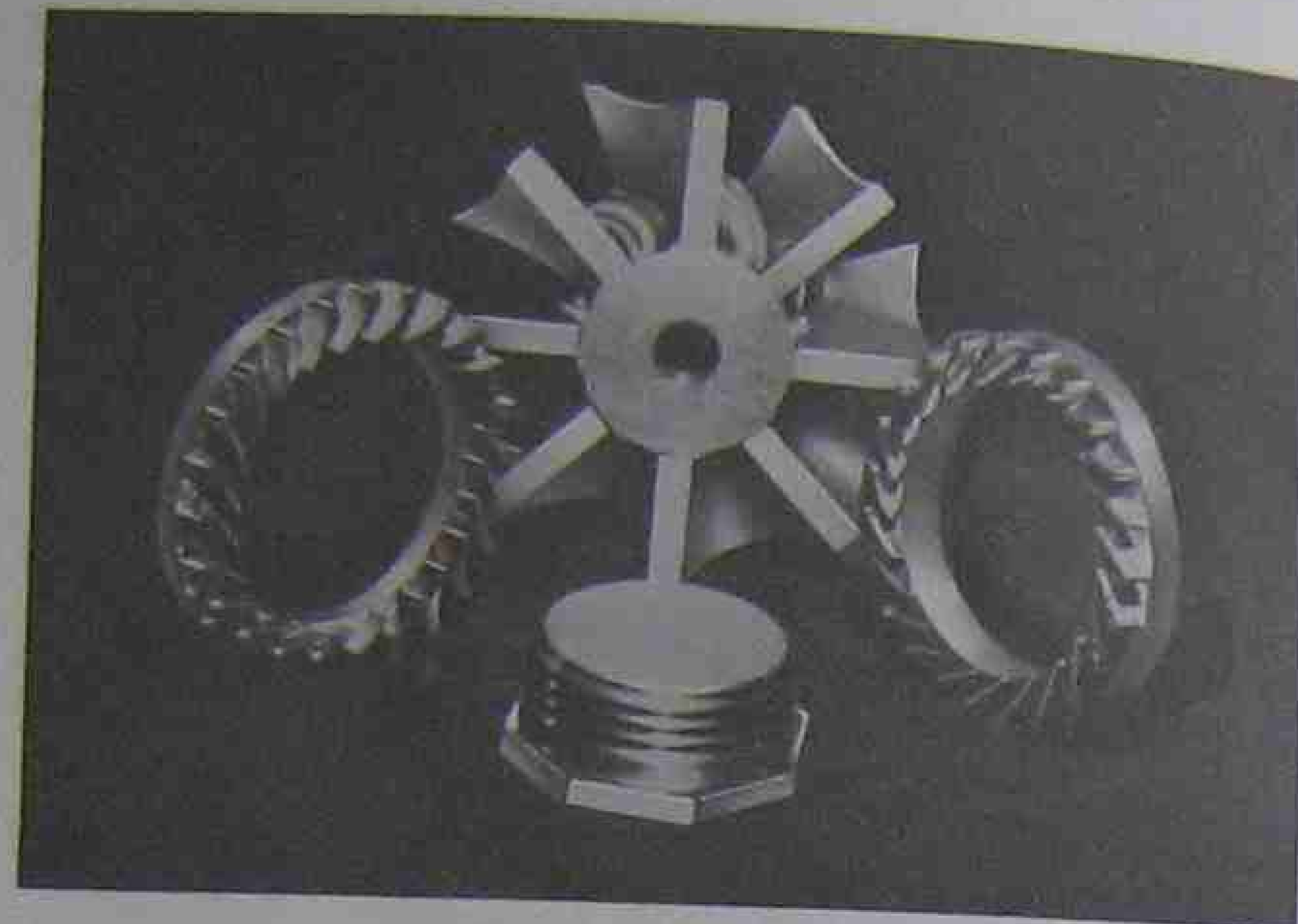


Fig. 1–20. Path of a milling-cutter tracing the perimeter or profile of a part under continuous-path numerical control.

Courtesy of Republic Aviation Corp.

Fig. 1–21. The two large airframe structures shown are being machined simultaneously on a large, continuous-path skin milling machine. The parts are for use on the wing panels of an Air Force supersonic jet fighter.

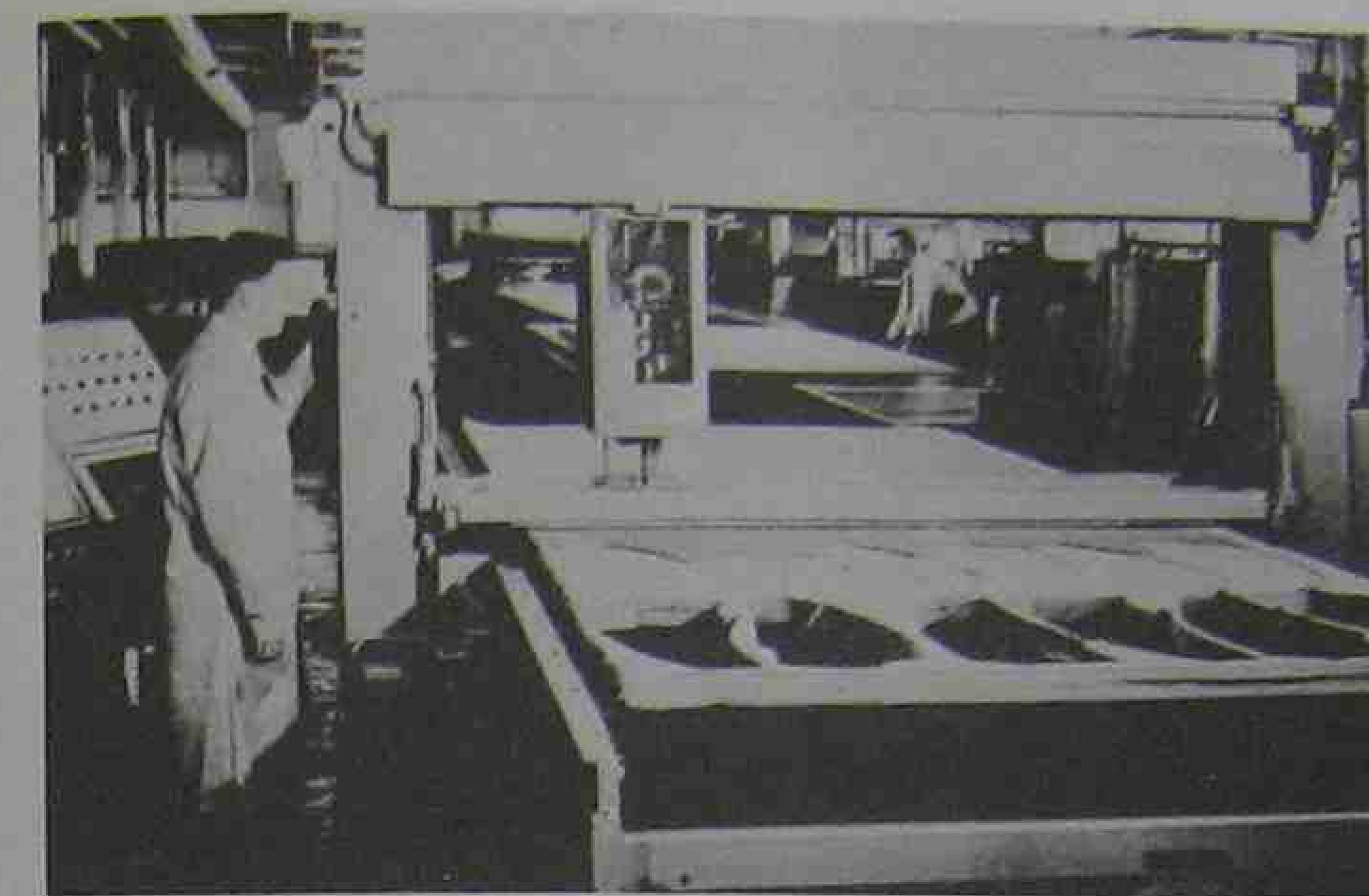


Courtesy of Thompson Ramo Wooldridge, Inc.

Fig. 1–22. Examples of complex parts produced on numerically controlled contour milling equipment.

Courtesy of Lodge and Shipley

Fig. 1–23. The numerically controlled engine lathe here shown is capable of making full contour cuts by coordinated motions of the saddle and cross-slide. "Gearless" thread cutting may also be performed by automatic synchronization of the spindle rpm and the rate of travel of the saddle. All operations are performed automatically from tape instructions. The slant bed arrangement, which is characteristic of NC lathes, offers improved rigidity and easier access to the workpiece and tooling than with conventional, horizontal bed lathes.



Courtesy of Cincinnati Milacron

Fig. 1–24. Numerical control is also being applied to fabric cutting. The cutting knife, capable of cutting through a stack of material, is directed by a numerical control unit.

Courtesy of Elox, Division of Colt Industries

Fig. 1–25. The electro-discharge (EDM) machine shown is used primarily for cutting small punch-die sets to very close accuracies and very fine finishes. The control system is shown at the right side of the machine. A closeup of a part being cut is shown in Fig. 1–26.

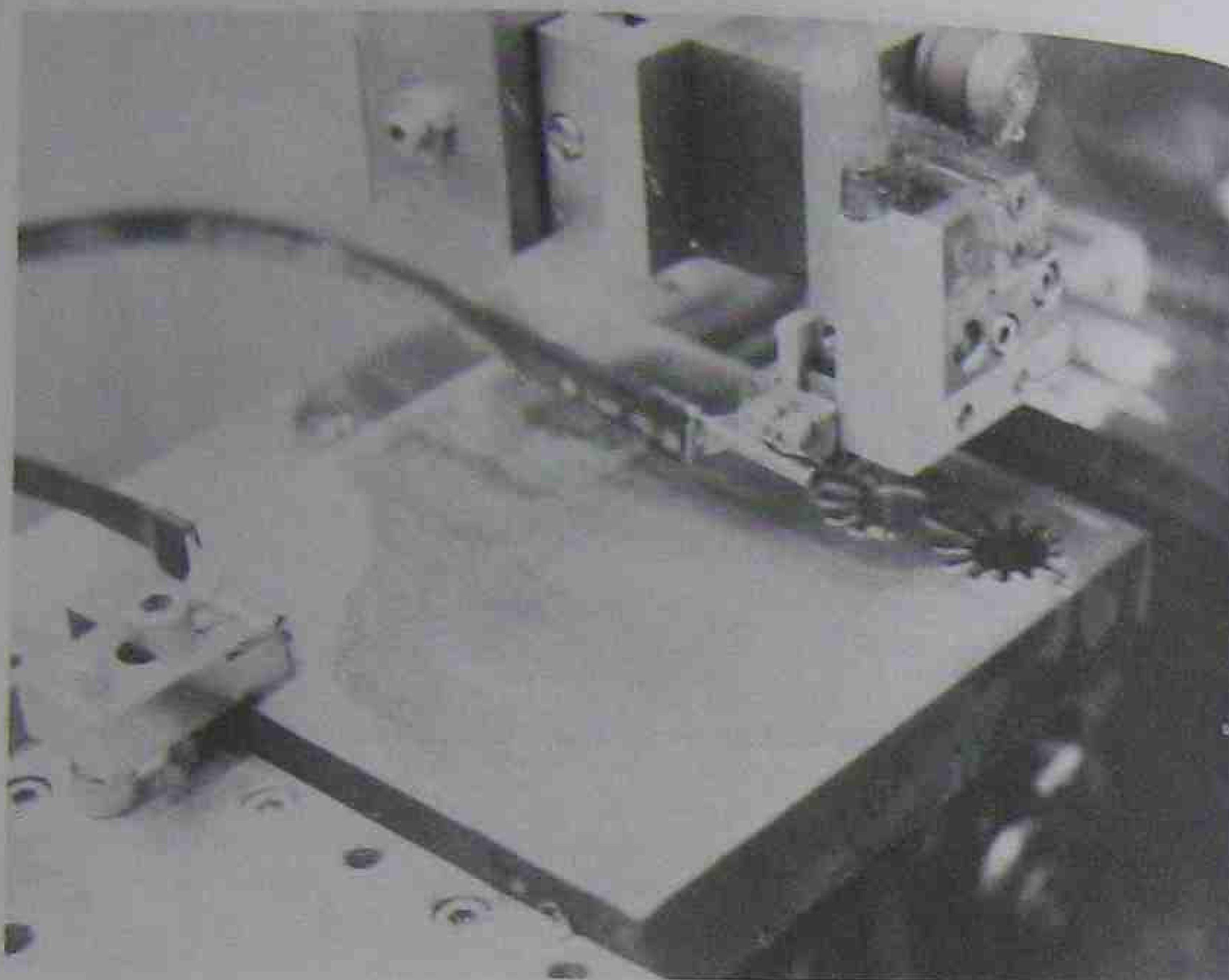
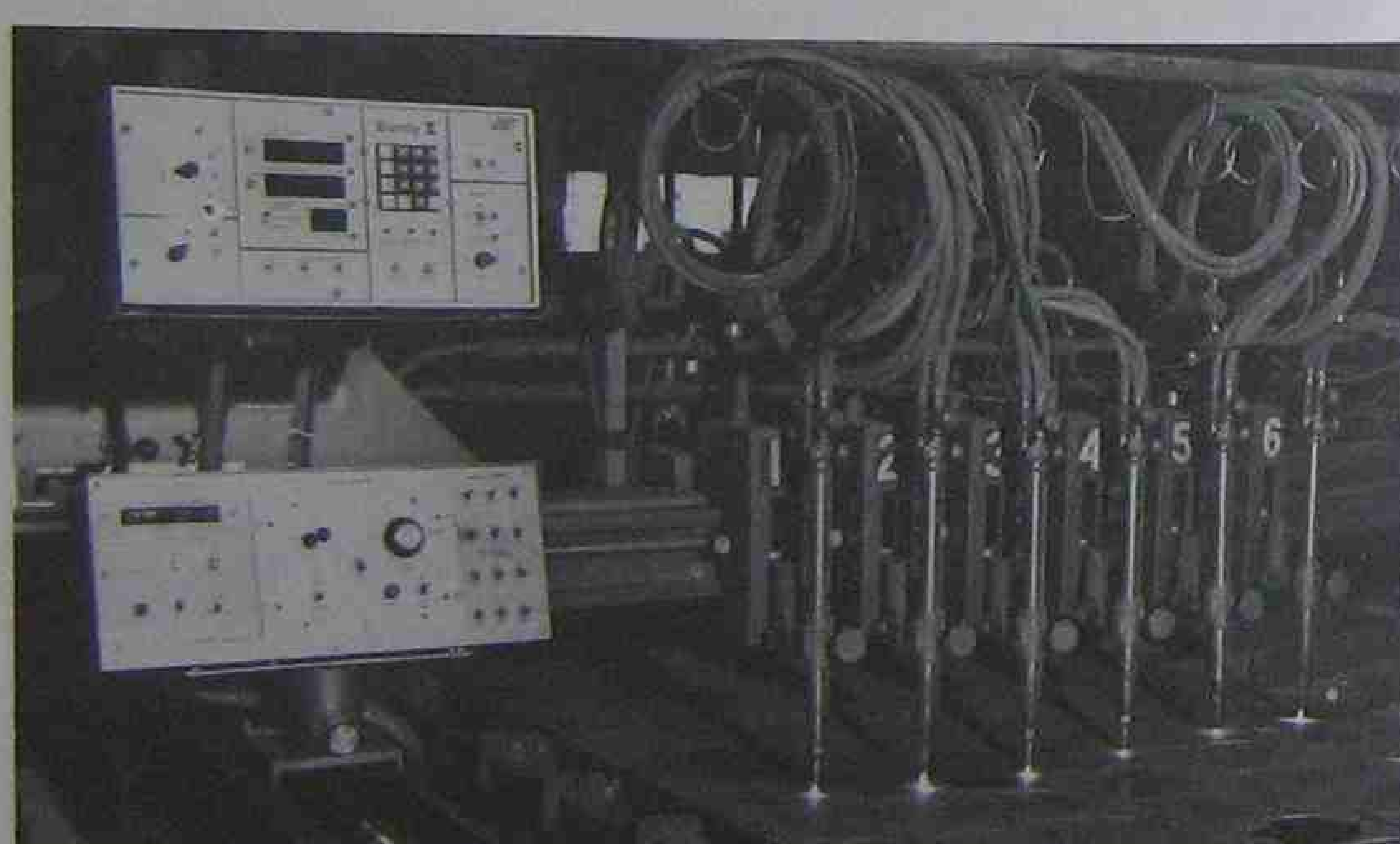


Fig. 1–26. Closeup view of a part being cut by a wire EDM numerical control machine.



Courtesy of Cleveland Machine Controls, Inc.

Fig. 1–27. This continuous-path torch cutting machine is capable of cutting six parts simultaneously from large sheet steel. Shipyards and heavy steel fabricators find this application profitable.

QUESTIONS TO CHAPTER 1

1. The first successful demonstration of a numerical control machine was at MIT in: 1947, 1950, 1952, 1955.
2. Since numerical control is an automated process, it is generally reserved for large quantities of the same part. (True, False.)
3. List as many potential point-to-point applications as you can think of.
4. List as many potential contouring applications as you can think of.
5. By 1979 approximately *20,000, 60,000, 100,000,* or *282,000* numerical controlled machines were shipped in the United States. (Circle correct quantity.)
6. What percentage of NC dollar shipments in 1979 consisted of lathes?
7. Numerical control could never be suitable for large quantity production. (True, False.)
8. Because of the power requirements of a control system, the size and cost are almost directly proportional to the size of the machine it controls. (True, False.)
9. Since point-to-point numerical control is less complicated than the contouring type, MIT decided to develop this concept first. (True, False.)
10. Which of the following NC applications is the most popular?
    a. Grinding
    b. Turning
    c. Broaching
11. The major components which comprise a numerical control machine-tool installation, in addition to the director (control system), are:
    a. ______________
    b. ______________
    c. ______________
    d. ______________
    e. ______________

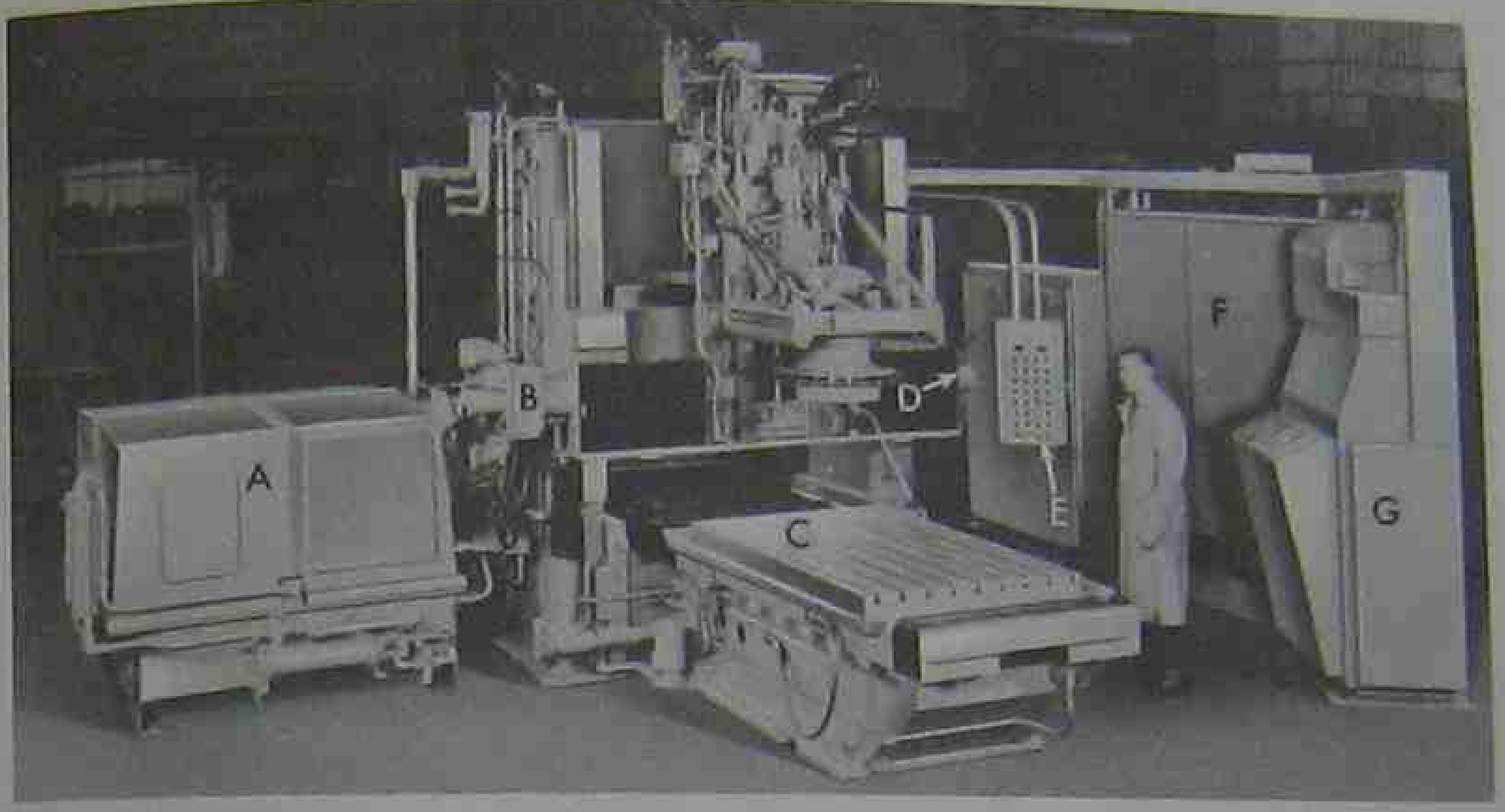
CHAPTER TWO

How the Numerical Control Installation Operates

It would be impossible in one chapter to fully cover all the details and theory of operation of any one numerical control system to say nothing of the many types of numerical control systems now being installed throughout industry. A typical installation will therefore be considered and some variants of this will be described with the intent of giving those considering the purchase and operation of numerical control equipment an overall view.

There are generally six major elements that comprise a numerical control installation. These are:

1. The control system
2. The machine tool
3. The drive unit or units
4. The feedback or servo components (assuming a closed loop system)
5. The electrically operated control equipment such as starters, relays, etc. (magnetics)
6. The manual controls, consisting generally of buttons, dials, or switches used by the operator.



Courtesy of Sundstrand Machine Tool Div., Sundstrand Corp.

Fig. 2–1. Major components comprising one type of numerical control installation: (A) hydraulic pump and reservoir, (B) drive unit, (C) machine tool, (D) feedback device, (E) manual controls, (F) electrical cabinets, and (G) control system.

GENERAL OPERATION

Figure 2–1 shows the layout of a three-axis numerical control machining center with the major components noted. Although of an older vintage, the operating principles of newer NC machines are the same. Had this assembly incorporated an electrical drive system instead of hydraulic drives, an electrical supply and amplifier unit would have been used in place of the hydraulic pump and reservoir.

Again referring to Fig. 2–1, tape instructions are read into the control system shown at the right.[1] Following electronic processing in the control system, electrical commands are amplified and sent to the drive units on the machine tool as well as to the electrical control cabinets. Commands sent directly to the drive units dictate the lengths of movements together with the proper feed rates. The commands that are directed to the electrical cabinets call for such operations as the starting and stopping of spindle motors; turning coolants on and off; adjusting spindle speeds; actuating tool changes; and operating other electrical devices. These command signals are transmitted at a relatively low voltage level and are used to actuate switches and relays that control much higher levels of power.

A feedback device that may be attached to either end of the lead screw "reports" the position of the moving machine tool element back to the control cabinet.

Manual controls which allow the operator to move the machine at rapid

[1] With most control systems, instructions may also be inserted manually via keyboard.

Courtesy of Jones & Lamson

Fig. 2–2. The operator's panel and the control system are incorporated in the headstock of the lathe shown (just to the left of the operator).

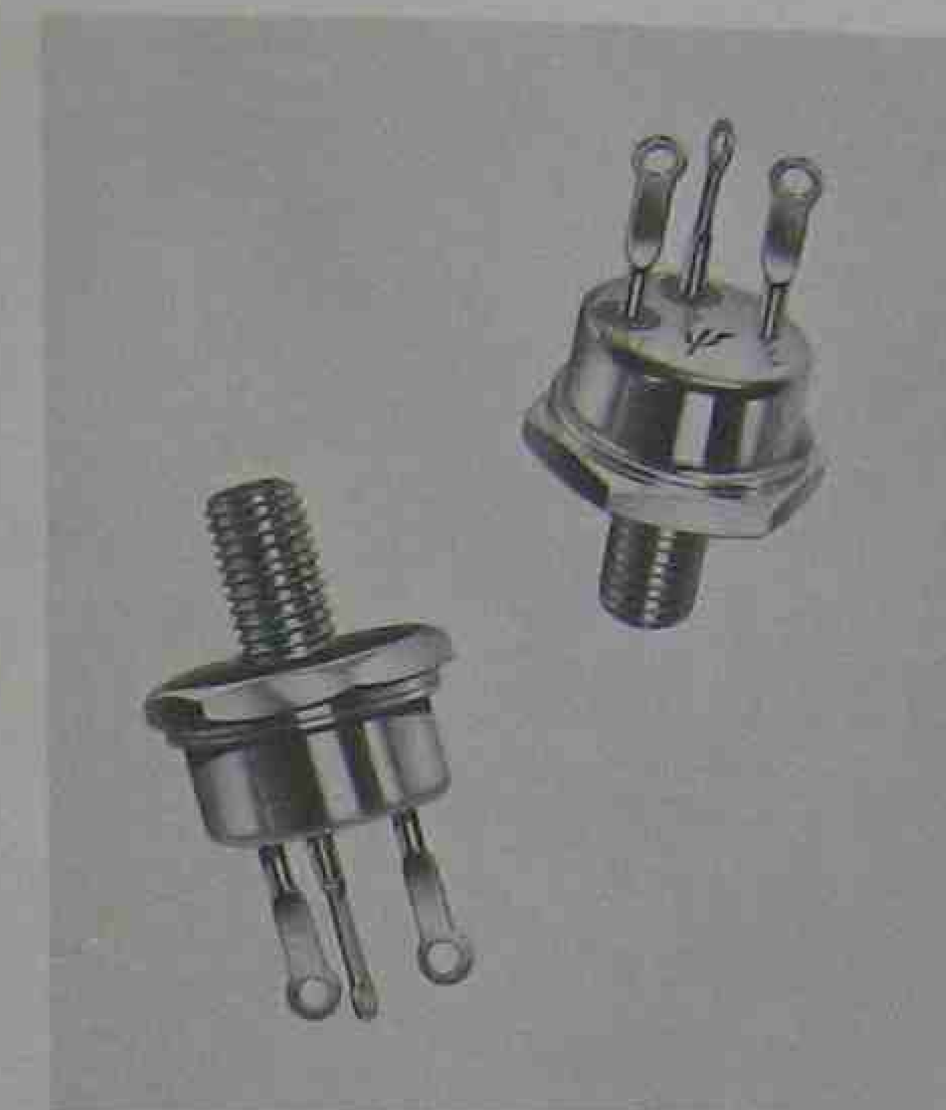
and jog traverses as well as control other functions such as starting and stopping the spindle motor, may be assembled as a pendant (as shown in Fig. 2–1); a stand; or a separate cabinet. On smaller machines manual controls, and even the control system, may be incorporated into one of the major machine tool components such as the head or bed. (See Fig. 2–2.)

THE CONTROL SYSTEM

Reference has been made to "first," "second," "third," and "fourth" generation control systems. First generation systems are those that utilized vacuum and gaseous tubes in order to control and amplify the electronic data. Second generation systems refer to units that are entirely, or almost entirely, constructed of *solid state* components, such as transistors. (See Fig. 2–3.)

A third generation, known as integrated circuitry, followed. This concept combined into one relatively small device the functions that would be performed by a number of separate components such as solid state diodes or amplifiers. The integrated circuit device was, therefore, a *complete circuit* which had been miniaturized. Figure 2–4 dramatizes the reduction in both size and cost for a comparable circuit which was originally designed as a vacuum tube device, then transformed to solid state configuration, and then into integrated circuit form.

It was not until approximately 1971 that the control system changed dramatically and a "fourth generation" system began appearing. The new concept was called CNC, which stands for Computer Numerical Control. Instead of wired or specially designed circuits developed to perform NC functions and

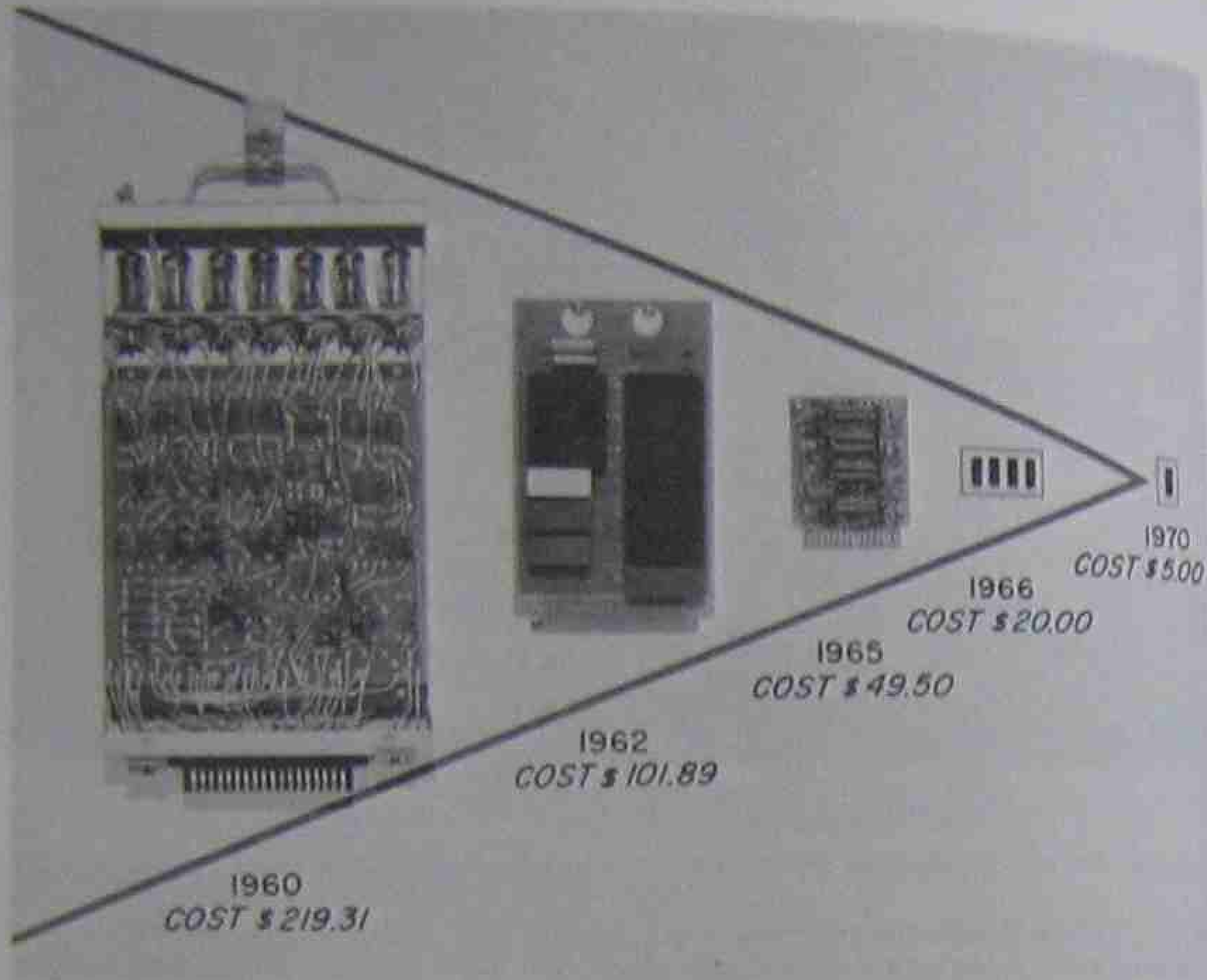


Courtesy of Thompson Ramo Wooldridge, Inc.

Fig. 2–3. These transistors measure approximately one inch long by one-half inch in diameter. Each replaces a vacuum tube about five times its size. These, however, are considerably larger than "third" and "fourth" generation components which involve integrated circuits, minicomputers, and microcomputers.

tailored to meet the electronic requirements of a particular machine tool, a general purpose minicomputer was used. The particular machine tool and system requirements were programmed into the computer, thus reducing, or eliminating, the special circuitry required to match a particular machine tool. The machine tool could then be operated from the stored program in the computer. This "softwire" approach offered some distinct advantages over the former "hardwire" systems. Those functions normally found in the hardwire systems could be extended, such as the number of digits in a command, or the amount of cutter compensation, or the extent of tool offset.[2] New features could be added readily that would have been impossible or extremely costly to incorporate previously, such as the ability to change (edit) a program while in the control system and the storage of a number of programs, canned cycles, programmable subroutines, maintenance diagnostics, or lead screw error compensation. At first the minicomputer systems were somewhat more costly than hardwire systems, however, as the cost of minicomputers dropped so did the cost of the systems. The machining instructions, which are punched into the tape and which are the same as would be required of a hardwire system, are read into the minicomputer where they are stored. Unlike the hardwire systems,

[2] See Numerical Control Terms, Definitions, and Standards (Chapter 10) for an explanation of cutter compensation and tool offset.



Courtesy of Industrial Controls Div., Bendix Corp.

Fig. 2–4. The progressive reduction in both size and cost of a vacuum tube design to an integrated circuit is dramatized above.

wherein the tape is read each time a part is machined, the machining instructions need be read into the minicomputer only once. After the instructions have been stored in the memory of the minicomputer they may be "played" as often as required. The tape reader is used for the initial input and remains idle while the parts are being machined.

While the minicomputer systems were a great improvement over the hardwire systems, these are being replaced by microcomputer systems. The microcomputer, which is often referred to as a "computer on a chip," is more compact, less costly, easier to maintain, and somewhat better suited to a shop environment. A microcomputer control system is shown in Fig. 2–5.

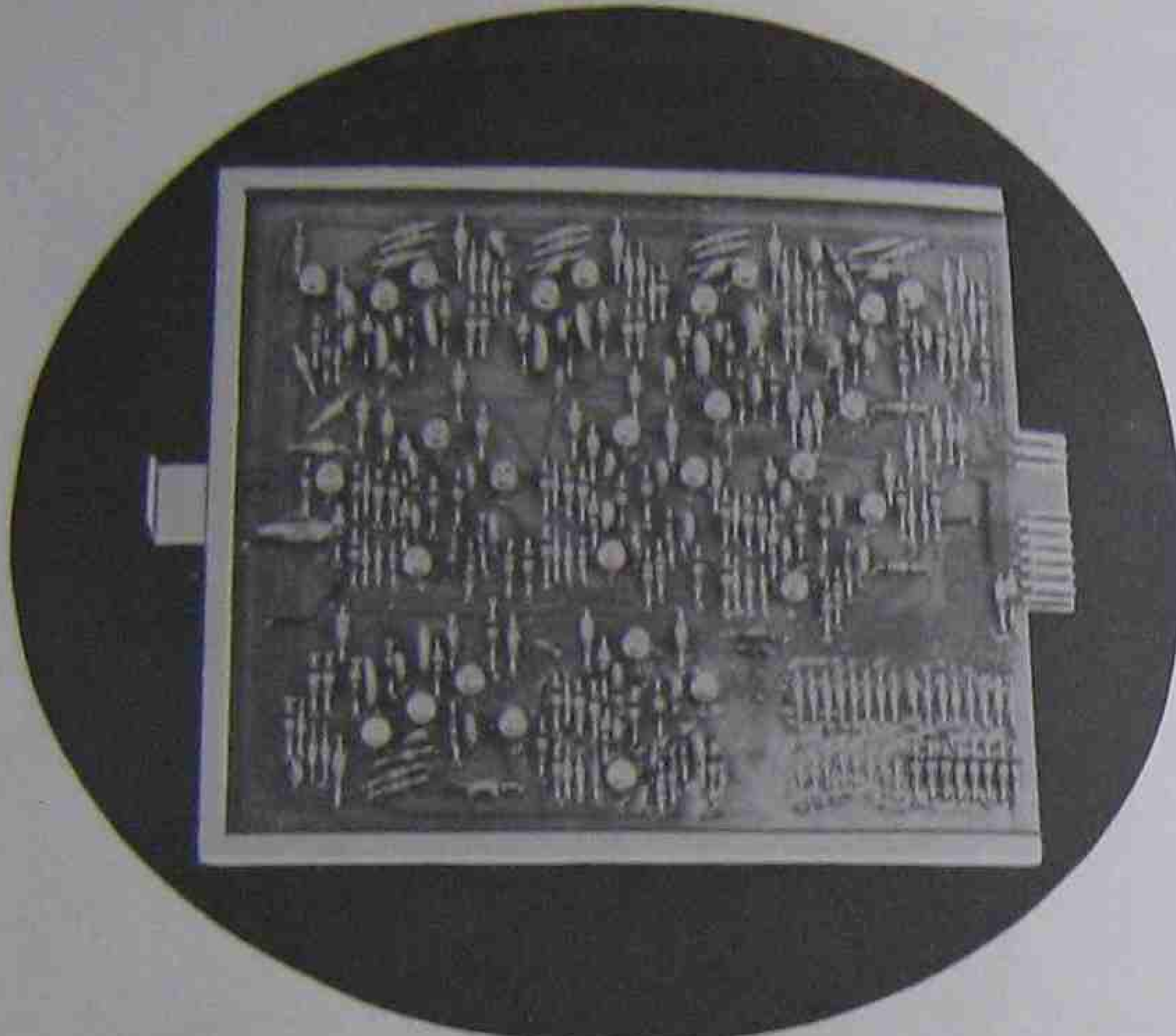
Circuit Boards Almost since the beginning of NC, control systems have utilized the concept of modular or circuit board construction. A *module* consists of numerous electronic circuits and components arranged with respect to a printed circuit pattern which is laid out on the board. The advantage of modular construction lies in its relative ease of maintenance since a faulty board is far easier to detect than any individual component. Once the faulty board has been located, it may be readily replaced and the machine tool put back in operation with a minimum of downtime. The module may then be checked, away from

Courtesy of Bendix Industrial Controls Division

Fig. 2–5. The CNC system shown consists of several compact microcomputers together with the other requirements of a control system. This particular system may be procured as a stand-alone unit or in modules for installation as an integral part of a machine tool.

the machine tool site, in order to find and replace the faulty component(s). Figure 2–6 shows an example of one type of module board.

Most control systems contain duplicate module boards which may be interchanged between control systems of the same manufacture or within the same control system since the circuitry of any one axis is usually identical to



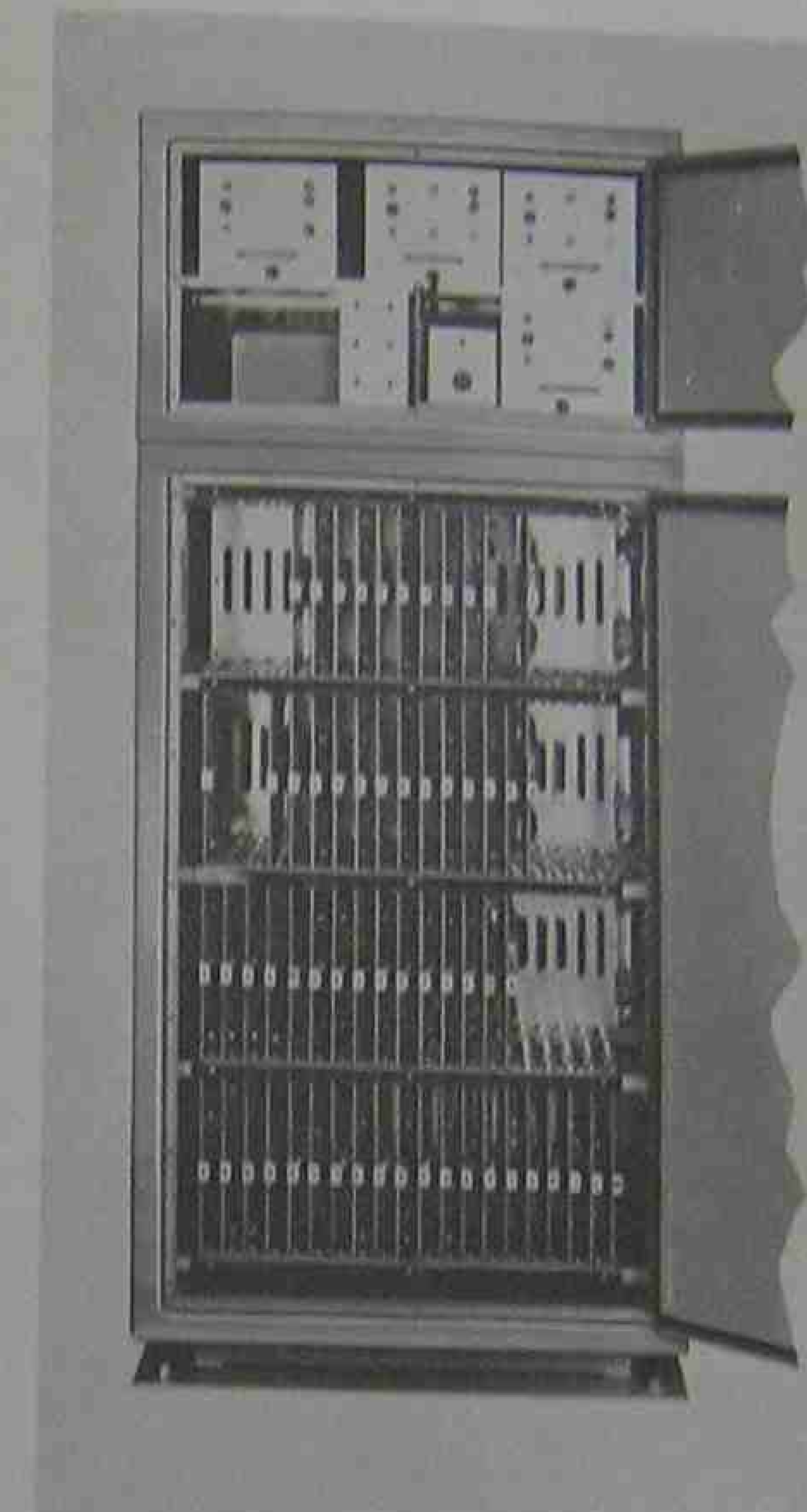
Courtesy of The Bunker Ramo Corp.

Fig. 2–6. Printed circuit module board. The size of the individual boards and the total number in a system will depend on the manufacturer.

that of another axis. Figure 2–7 shows the rear view of a hardwired NC cabinet containing 60 circuit boards. This may be compared to the system shown in Fig. 2–8 which contains 9 boards.

The Tape Reader The link between the coded punched information on the tape and the electronic signals of the control system is the tape reading mechanism. Prior to CNC systems, tape readers were generally of the mechanical or photoelectric type, depending on the reading speed required. For slower speeds—characteristic of point-to-point systems—mechanical readers that insert *feelers* or metal prongs into the holes were utilized although photoelectric slow speed readers were also common. A slow speed reader would be classified as being capable of reading from approximately 20 to 100 characters (rows across the tape) per second. High speed readers commonly used with hardwire contouring systems and ranging in reading speeds from 300 to 500 characters per second were almost always of the photoelectric type wherein holes are detected by light passing through to a photoelectric cell.

In addition to variations in reading speeds, differences exist as to whether



Courtesy of The Bunker Ramo Corp.

Fig. 2–7. Rear view of a hardwire modularized control cabinet.

the tape is handled on a reel (Fig. 2–9a) or allowed to collect in a *tumble box* (Fig. 2–9b). Because of the relatively long lengths required with contouring operations, it is almost mandatory that reels be utilized. Usually less tape is needed in point-to-point operations and hence it may be handled satisfactorily without reels. However, the use of reels may be worthy of consideration in this instance also. Readers having reel mechanisms (also known as spoolers) usually incorporate a rewind feature which automatically rewinds the tape from a coded auxiliary instruction recorded *on* the tape. The cost of tape readers for hardwire systems varied considerably, from a few hundred dollars for the slower mechanical type without spoolers to several thousand dollars for the higher speed photoelectric variety.

Courtesy of McDonnell Douglas Industrial Control Products

Fig. 2–8. The CNC system shown in the cabinet above contains 9 circuit boards, which is considerably less than the 60 shown in the hardwire system of Fig. 2–7. Also, the system shown in Fig. 2–7 has considerably less capability than that of the system shown above.

CNC, or softwire, systems negate the requirement for costly high speed readers since the tape for the machining of a part need be read into the control system only once for an entire run of the same part. Rather than re-running the tape for each piece, the machine tool operates from the program stored in the computer. It should also be noted that almost all tape readers used with CNC systems are of the photoelectric type since they have proven to be more reliable than the mechanical type and there is now little cost differential between the two types of readers.

The lower cost photoelectric readers are just one contribution to much lower cost control systems. Other contributing factors to lower cost are the

elimination for the need of an air conditioner and the lower cost of the electronic components. For example, a three-axis hardwire control system that once cost $35,000 can now be procured for approximately $10,000. And the newer microcomputer systems has far greater capability and reliability.

Tape material generally used is of paper or Mylar[2] film, the latter being the more expensive. Mylar is recommended where repetitive runs are expected. Some companies have adopted a practice of preparing a paper check out tape which may be kept as a "master" and then duplicating the punched data onto a Mylar tape for handling production runs. The tape is punched from plain material. The small sprocket holes common to standard tape, are produced as

Courtesy of Rocketdyne, A Division of North American Aviation, Inc.

Fig. 2–9a. This high-speed tape reader incorporates features such as reel handling mechanisms and automatic rewind, which were common to most contouring control systems of the hardwire type.

[2] Registered trade name of E. I. du Pont de Nemours & Co.

Courtesy of Hughes Aircraft Company, Industrial Systems Div.

Fig. 2–9b. This PTP system incorporates a combination reel and tumble box tape reading arrangement. The tape is fed from the reel and allowed to collect below. The tape may then be rewound onto the reel after the part has been machined.

the characters are being punched. These are used in the preparation of duplicate tapes and by some control systems.

Data Storage and Control Following the tape reading operation the numerical data are distributed to electronic storage sections that record the distance movements along the various axes. If it is assumed that the control system is of a three-axis type, the letter addresses (X, Y and Z) will direct the numerical data to the proper storage sections.

Figure 2–10 illustrates diagrammatically one general type of hardwire numerical control system which may be applicable to both point-to-point and contouring operations except that there is usually no *buffer* storage section with most point-to-point systems.

Referring to Fig. 2–10 it will be noted that there are two storage sections: one entitled "buffer storage" and the other "active storage." The function of the buffer storage section is to accept data from the reader *while* the machine is operating under the instructions contained in the active storage section. While such an arrangement is not usually called for in point-to-point systems it is extremely advantageous with contouring since the time required to read

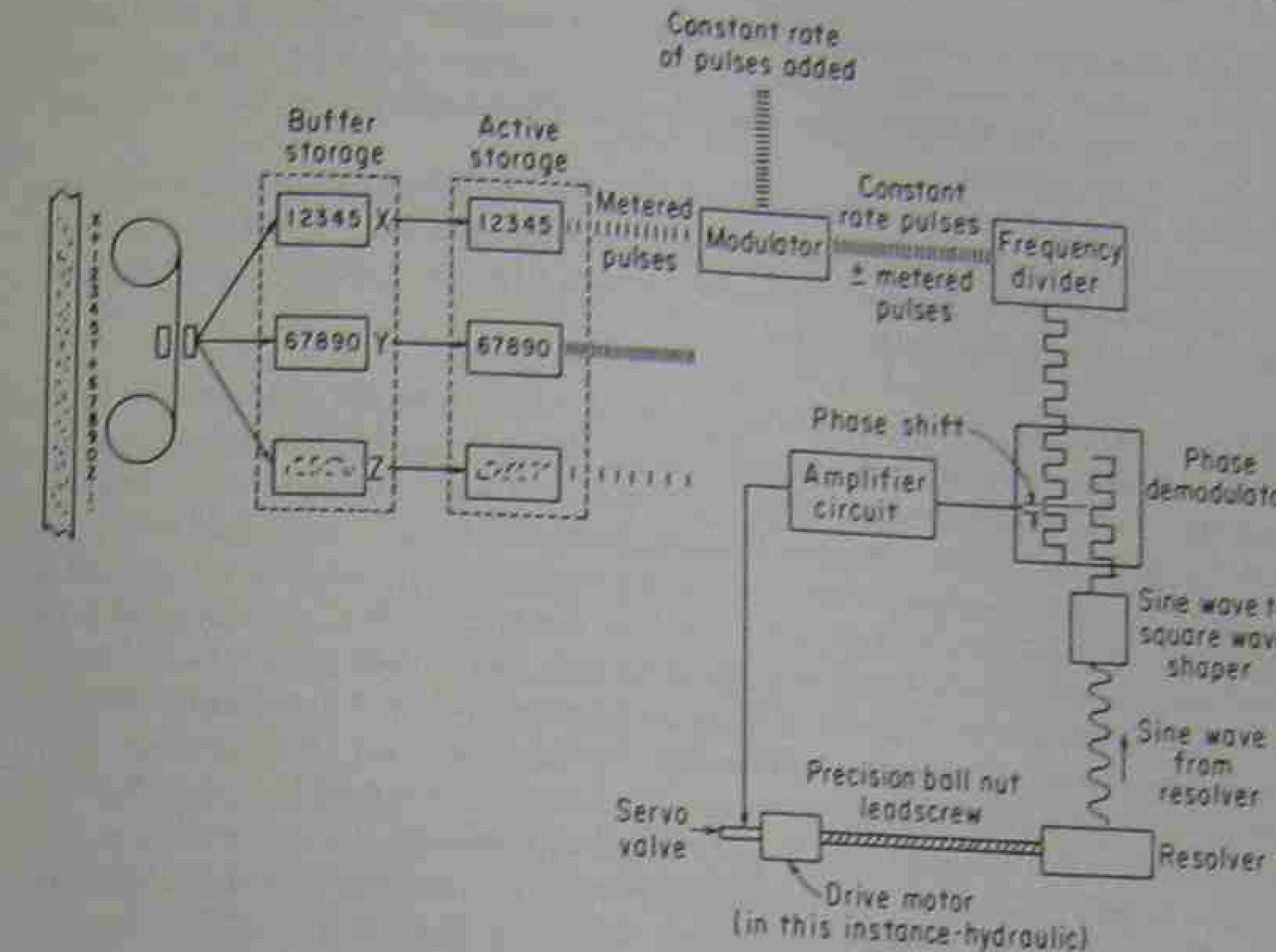


Fig. 2–10. Flow of data in a numerical control system.

contouring information into the control system via the tape reader is relatively long, even with high speed readers. When it is realized that a curved line is, in reality, often coded as a series of very short segmented straight lines (see Chapter 4) and that each straight-line segment requires a block of tape information, it is evident that without buffer storage the feedrate motion along the path would have to be interrupted prior to each straight-line motion in order to read the tape block for the forthcoming motion. The feedrate would therefore be delayed by an amount equal to the reading speed of the reader if buffer storage were not incorporated. The motion along the path could be made relatively smooth but only at the expense of the feedrate and consequent machining time for the part. An additional storage or buffer unit enables the control system to operate on active storage during the time period that the data for the next movement are being read into the buffer storage. The time required to transfer data from buffer to active storage is measured in millionths of a second and occurs at the precise moment that the straight-line path movement is completed for the block of information in active storage. The reason for the buffer storage not being a serious concern with point-to-point equipment is that the tape may be read while a depth movement, such as drilling, is being carried on[4] or read after an operation is performed and immediately prior to

[4] With two-dimensional systems where the depth motion is not under dimensional tape control.

the next dimension movement. Since there are far fewer blocks to be read with point-to-point systems than with contouring systems, the tape reading delay is not serious and would seldom justify the additional cost of buffer storage in the control system.

CNC systems may, or may not, require buffer storage depending on the speed of data transmission of the mini- or microcomputer. In either case it is far less expensive to handle the buffer storage requirements in a CNC system than with a hardwire control system.

Pulse Generation The distribution of pulses to the assigned axes may be handled in conjunction with the actual storage operation. The number of pulses and the rate at which they are distributed depends on the distance to be moved and the feedrate. If, for example, it were desired to move 1.2345 inches in the direction of the X axis and each pulse represented a distance of 0.0001 inch it would be necessary that 12,345 pulses be generated. Also, assuming it was desired to travel this distance at a rate of 10 ipm, the 12,345 pulses would have to be metered out in 0.12345 minute or approximately 7.4 seconds. With a contouring system, wherein movements in the directions of numerous axes must be closely coordinated, the train of pulses is measured in order that the required number for each axis is completed at precisely the same time as the others. If, for example, it were desired to provide component motions along the X and Y axes at the same rate and thus move the head of a machine along a 45-degree path, the number of pulses per period of time for each motion component would be equal. If, however, it were required to move along a path whereby the X-axis component were twice that of the Y-axis, then the number of X-axis pulses would be twice those of the Y-axis pulses over the same period of time. Metering of pulses to a number of axes so that the result is a straight-line movement is called *linear interpolation*. Metering of pulses so that the resultant movement is a circular arc is called *circular interpolation*. Since a point-to-point operation requires only that the head (or table) be positioned at specific points, without serious consideration as to the path of travel, the coordination of pulses is not nearly as important as with contouring.

Pulses can be generated and coordinated in one of several ways; two of the more common are the pulse rate multiplier and the digital differential analyzer—the latter also being referred to more popularly as a DDA. The pulse rate multiplier draws pulses from a timed stream of pulses in proportion to the axes' movements. The pulses for all axes are generated by a single digital clock which ensures that the relationship of the pulses remains constant since the train of pulses for the different axes comes from the same source. If the clock should change its rate, for example, the relationship of the pulse rates for the various axes would not change. This is most important in order to ensure an accurate straight-line movement. The DDA operates on a different principle in that pulses are fed to a register, or "reservoir," and are allowed to "overflow" the register at a reduced rate determined by the relationship of the movement of the axes. It is because of this overflow principle that DDA's are sometimes called overflow counters. Again, as with the pulse rate multiplier, the important consideration is that the pulses are generated by a single source for all axes

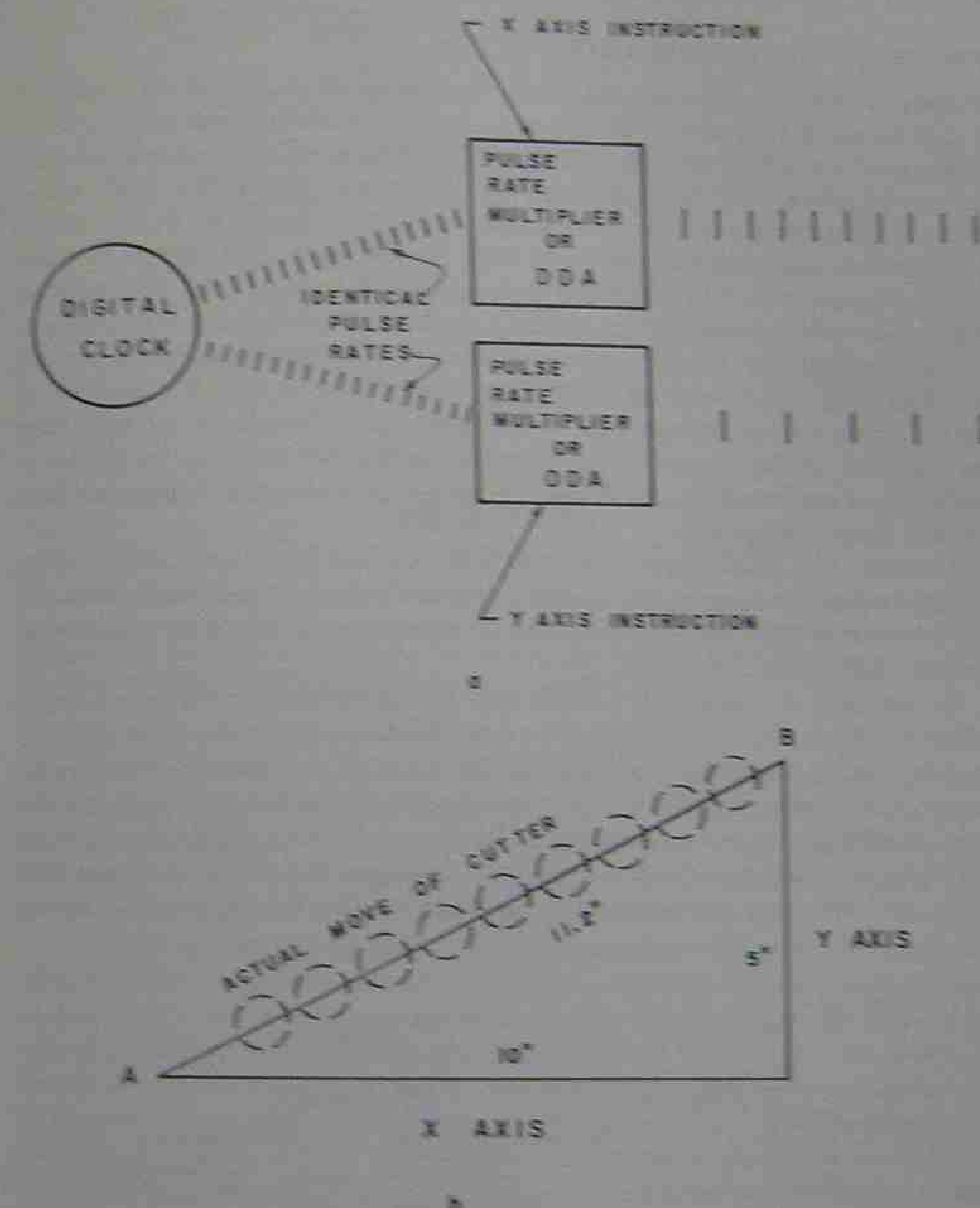


Fig. 2–11. Whether pulse rate multipliers or DDA's are used, the concept is much the same. Pulses from a single source, such as a digital clock, are fed to the pulse rate multipliers or DDA's shown in a (above). These latter units regulate the emitting pulse rate in conformance to the X and Y distances to be traveled over a fixed period of time. If it were required to move along the resultant line shown in b (above), the rate of the pulses for the X-axis movement would have to be twice that of the rate for the Y-axis movement.

An illustration of the pulse rate concept for both the pulse rate multiplier and the DDA is shown in Fig. 2–11.

THE SERVO OR FEEDBACK LOOP

Probably one of the more notable distinctions of an NC installation is whether it is of an *open loop* or *closed loop* arrangement. With an open loop system, pulses are transmitted directly to a motor that rotates a fixed amount for each pulse it receives. This type of motor is called a *stepping* motor, and, when attached directly to a 5 pitch lead screw, rotates 1.8° for every pulse, which is equivalent to a movement of 0.001 inch for the slide. The system is quite accurate despite the fact that there is no means of feeding back the location of the slide to the control system, which is typical of a closed loop arrangement. An illustration of the open loop concept is shown in Fig. 2–12. There are two drawbacks, however, with an open loop system. One is that the feedrate is limited due to the pulse rate that the stepping motor is capable of accepting and the other is that its power is limited to the smaller machine tools and to applications not requiring much load, or torque. It is also less expensive than a closed loop arrangement.

A closed loop arrangement includes a device that *feeds back* information as to the actual position of the machine element being moved. The command instruction may then be compared with the actual position and an adjustment made to correct the difference.

An example of a relatively simple and commonplace servomechanism is outlined in Fig. 2–13. The control of the home heating system shown is initiated via a setting on the thermostat. This is the *input* to the system and may be compared to tape instructions. The thermometer in the room records the existing condition just as a measuring, or *feedback*, device records the position of a slide on a machine tool. A difference between the input or thermostat setting and the room temperature as recorded by the thermometer causes a

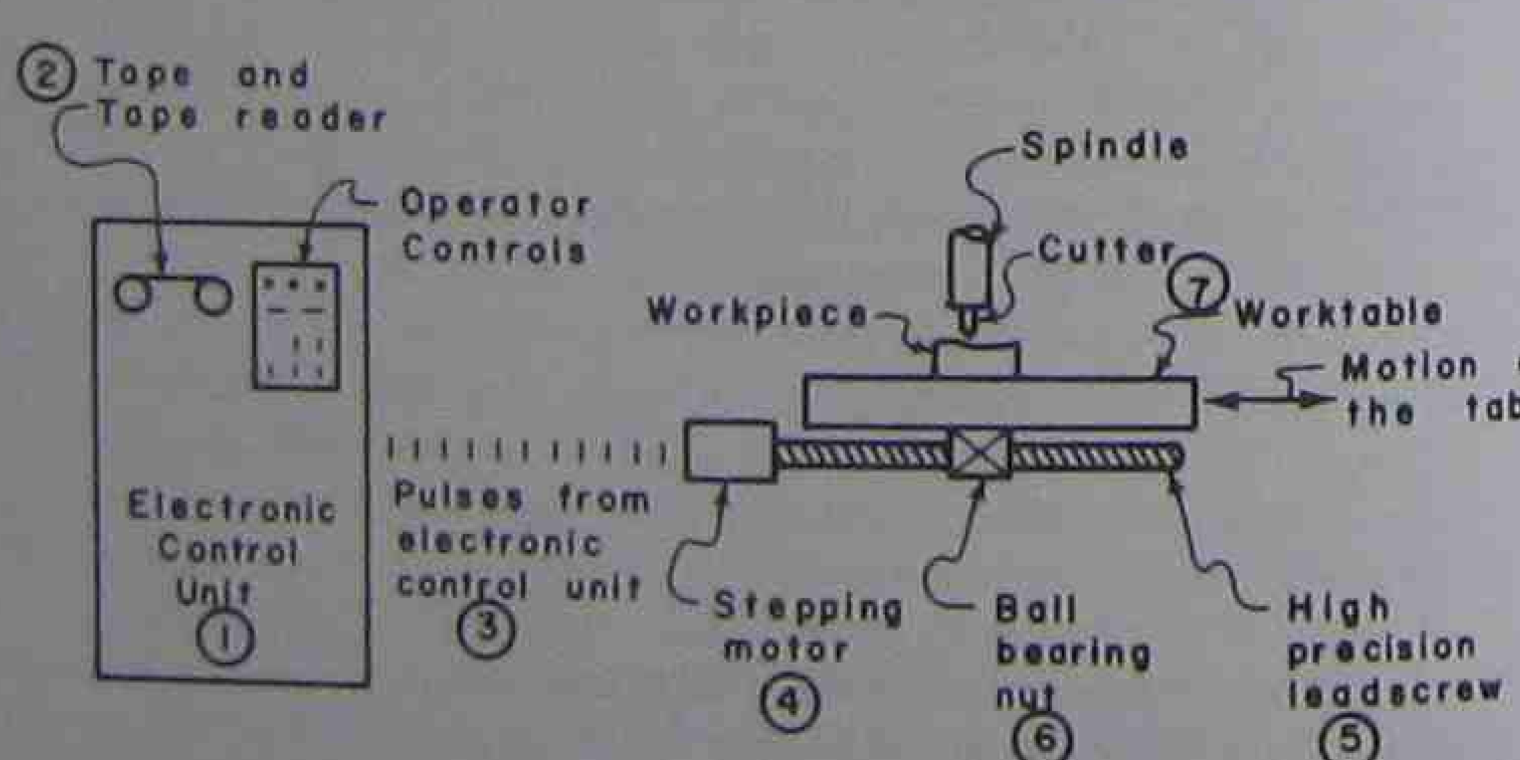


Fig. 2–12. Pulses are generated by the electronic control unit (1) in accordance with the instructions punched on the tape (2). These pulses (3) drive a stepping motor (4) which rotates a high-precision lead screw (5). The lead screw moves a ball-bearing nut (6) which is attached to the moving part of the machine tool, such as the worktable (7) or saddle.

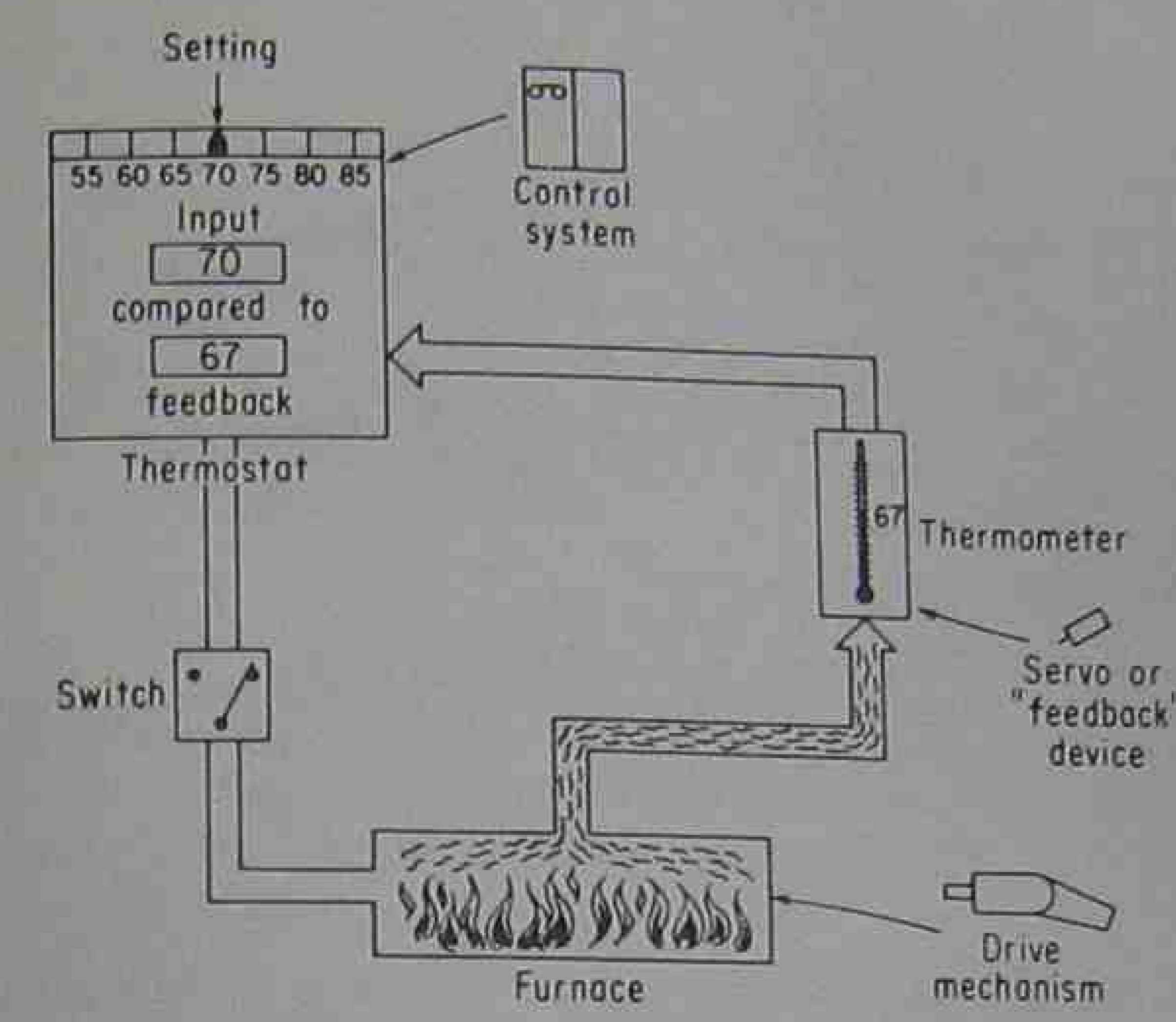


Fig. 2–13. The operating principle of the home heating system here shown schematically is similar to the feedback or servo loop principle employed with numerical control. The furnace may be compared to the drive motors, the thermometer to the linear measuring devices and the thermostat to the electronic control system.

switch to close which actuates a furnace which in turn raises the temperature of the room. When the temperature reaches the desired 70°F, as recorded by the thermometer, the switch opens and the furnace stops. The thermostat in this instance may be compared to the control system and the furnace to the drive mechanisms attached to the lead screws of a machine.

A diagram, describing the closed loop arrangement for an NC machine, is shown in Fig. 2–14. Input commands from the tape are compared to the location of the table via the position feedback signals. These signals are usually generated by either a resolver or encoder, which is attached to the lead screw (explained more fully later in this chapter). When there is an "unbalanced" condition in which the table is not where it is instructed to be, a voltage is fed to the drive motor. So that this voltage does not cause the motor to run too fast or cause the table to overshoot, a voltage-generating tachometer is also attached to the lead screw that counteracts the voltage to the drive motor and keeps it in line with that required for the specified feedrate of the table. When the table reaches its prescribed location, as determined by the resolver or encoder, the voltage to the drive motor stops, as do the drive motor, the lead screw, and the table. Tape commands that apply to functions such as turning the spindle motor on and off and selecting a tool from an automatic tool changer are fed directly to the electrical controls, where the power is amplified and fed to the controlled mechanism.

Again referring to Fig. 2–10, which describes a common type of servo arrangement and considering only the X axis since the others operate in iden-

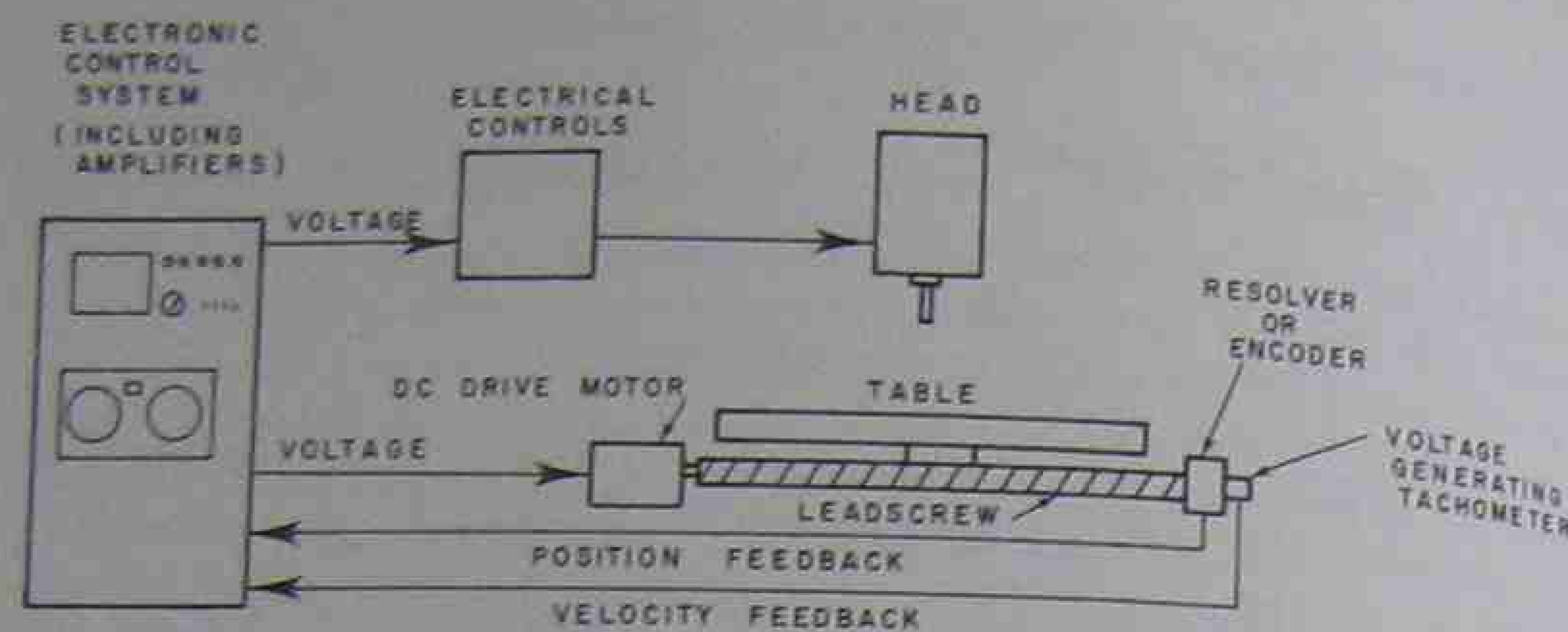
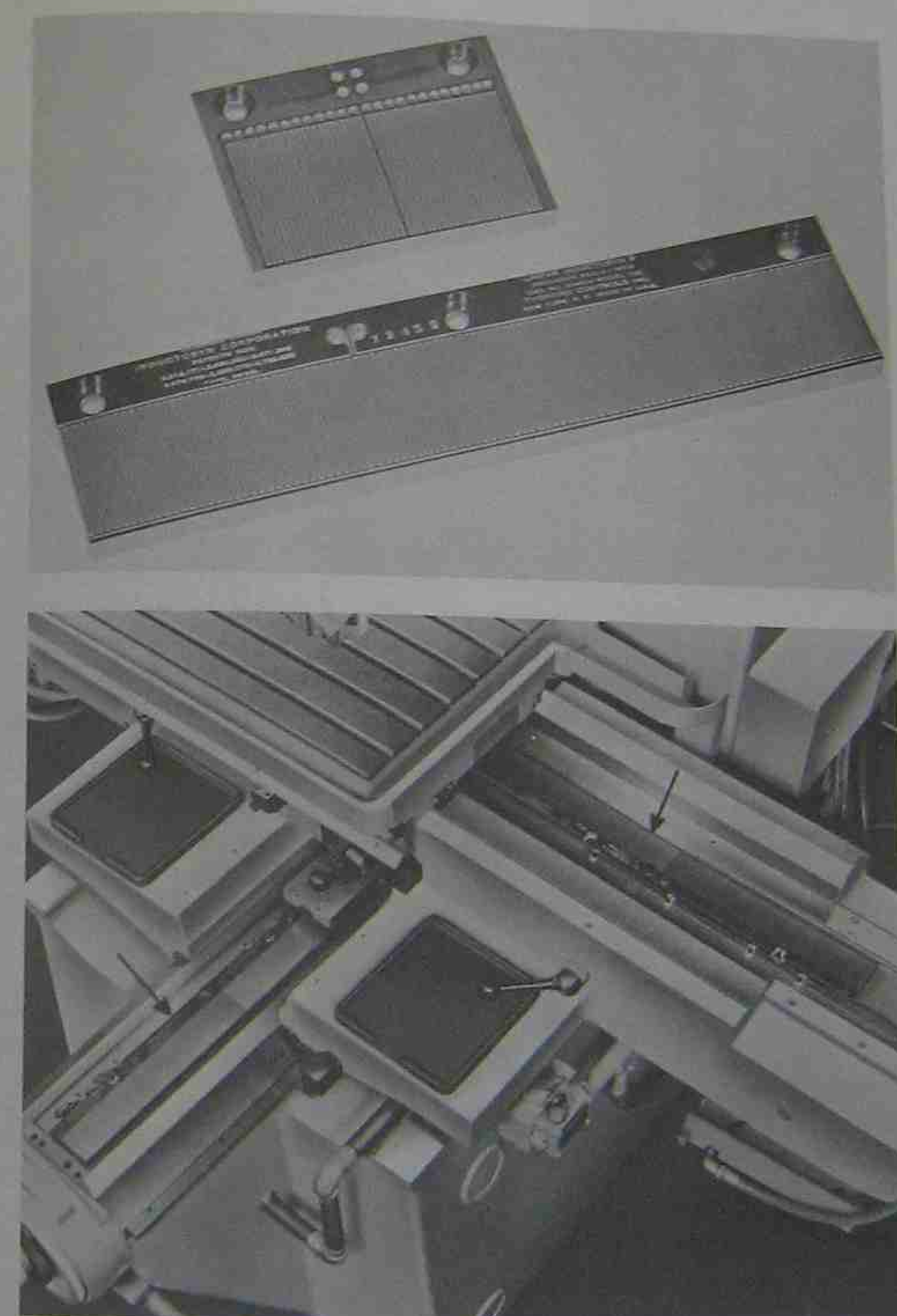


Fig. 2–14. The arrangement of a closed loop system is shown. Actually there are two feedbacks: one for the position of the table and another for the velocity, or feedrate, of the table. Signals are also sent to an electrical control section that handles such functions as turning the spindle on and off, or adjusting the spindle speeds, or changing a tool with an automatic tool changer.

tical fashion, the metered pulse train is fed to a *modulator* where it is superimposed on a train of pulses of constant rate. The modulator *adds* the command pulses to the train of "constant rate" pulses if the sign of machine tool motion is *positive* and *subtracts* them if the sign is *negative*. The resulting pulse pattern which contains the carrier pulses plus or minus the command pulses is passed through a frequency divider which reduces the number of pulses and converts the output to a square wave. The square wave is next fed to a demodulator

Courtesy of Kearfott Division, General Precision, Inc.

Fig. 2–15. This unit, which measures only 1½ inches overall, plays a very important part in the servo loop of a numerical control system. It is a resolver and feeds back the position of the machine tool slide to the control system for comparison with the input command.



Courtesy of Farrand Controls, Inc.

Fig. 2–16. (Above) Flat type feedback device consists of two elements, one of which passes over the other when table or slide is moved. (Below) Two pairs of the longer elements (see arrows) are shown in place on a machine tool.

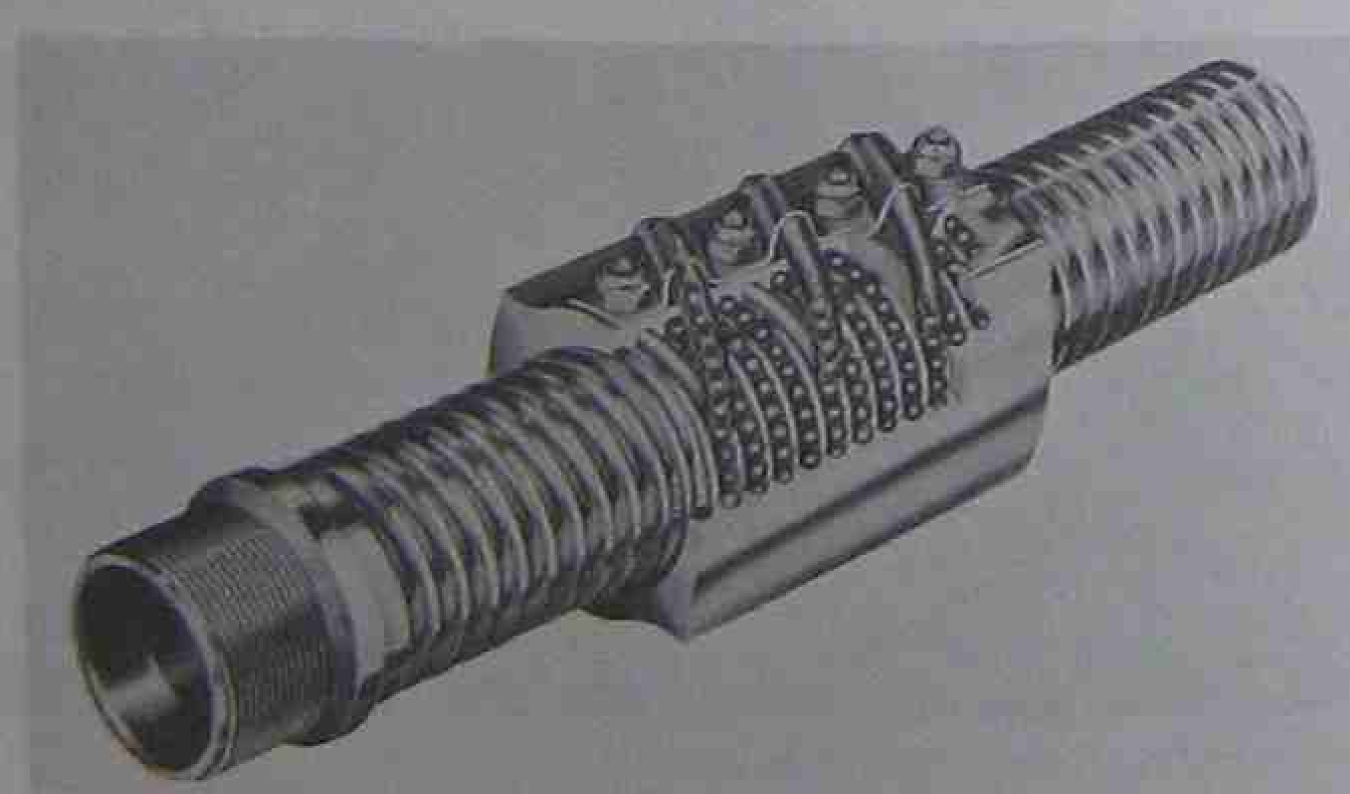
which compares this input with the input from a rotary resolver connected to the machine tool leadscrew. The output of the demodulator is a d-c voltage, the magnitude of which is proportional to the phase difference between the two inputs. This voltage is then amplified and fed to the coil (torque motor) of a hydraulic servo valve which controls the flow of fluid to the motor driving the lead screw or to a d-c electric motor.

The feedback unit or rotary resolver is a surprisingly small device to be playing such a significant role. The one pictured in Fig. 2–15 measures approximately 1½ inches in overall length. Another feedback device is the linear Inductosyn and in actuality operates as a resolver laid out in a flat pattern. The device consists of two elements which slide over each other without contact. One section may be compared to the stator of a resolver and the other section to the rotor. A close-up of the two elements with a view showing their attachment to a machine is provided in Fig. 2–16. An advantage of the Inductosyn is that it measures the actual position of the slides and does not depend on the accuracy of the lead screw, as does the resolver. Inductosyns, however, are more expensive than rotary resolvers, especially for long lengths of slide travel.

Another type of feedback device that is gaining popularity is the digital *encoder*. The encoder feeds back pulses to the control system which are usually generated by a rotary photo arrangement. The digital form of feedback is well suited to the CNC system, which is, essentially, a digital computer. It is for this reason, also, that the resolver's analog signal is often converted to a digital feedback form.

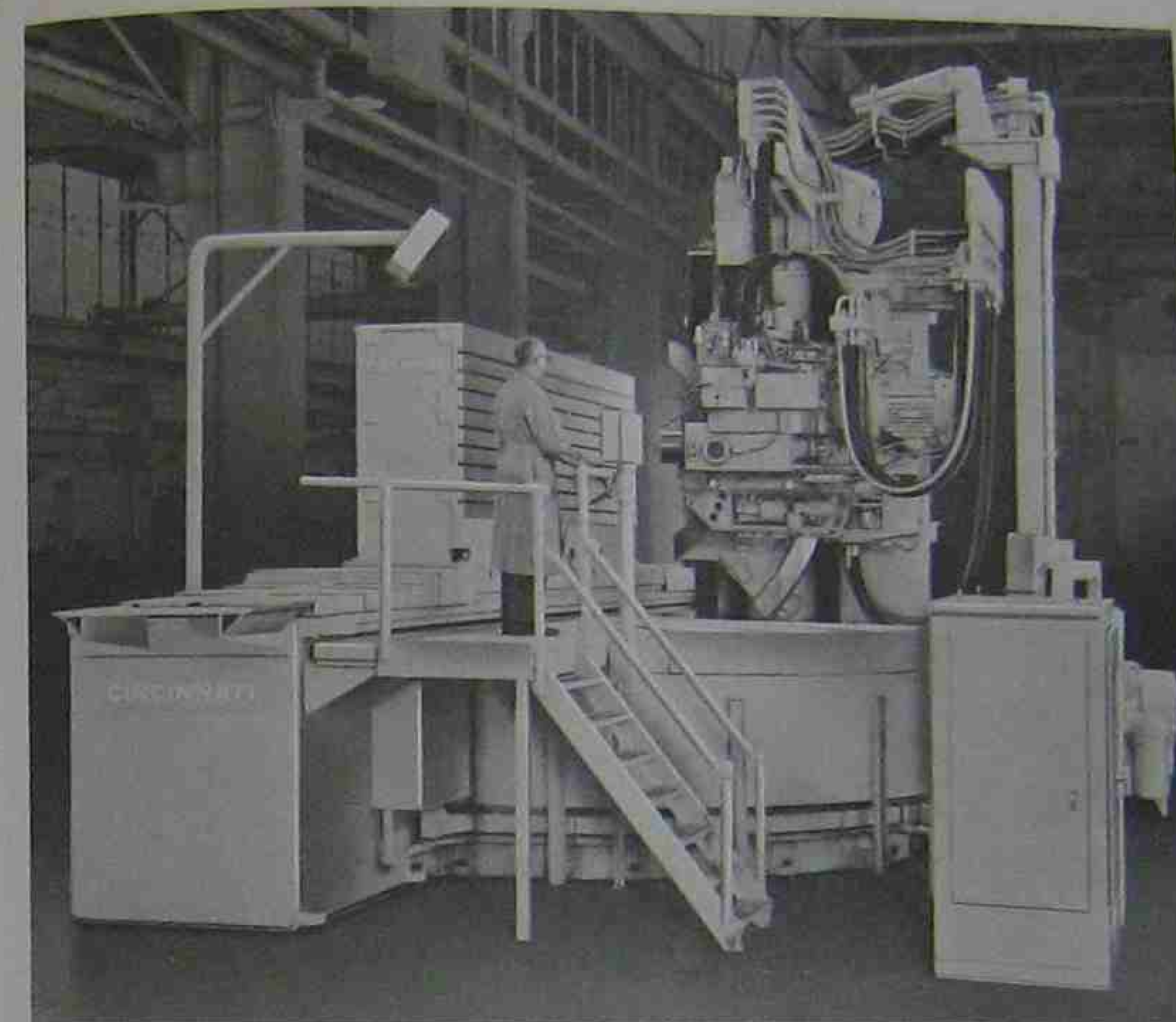
THE DRIVE MECHANISM

By far the most popular means of moving a machine tool table, head, or other member is through a precision recirculating ball-nut lead screw such as that shown in Fig. 2–17. The two chief reasons are accuracy and low friction which



Courtesy of Saginaw Steering Gear Division, General Motors Corp.

Fig. 2–17. This precision recirculating ball-nut lead screw provides a high degree of accuracy and low frictional resistance for a numerical control drive.



Courtesy of Cincinnati Milling Machine Co.

Fig. 2–18. Hydraulic drive mechanisms are used to operate this large five-axis milling machine. Smaller machine tools, such as point-to-point drills, small and medium size machining centers, or contouring lathes are generally operated by electric drives.

allows for rapid servo response. Other driving means include hydraulic cylinders, rack and pinion drives, or worm gears.

The "muscle" for driving the lead screw may either be hydraulic or electric power. The selection will depend on several factors, the principal one being the size of the load to be moved. (See Fig. 2–18.) While hydraulic drives generally have offered superior response, d-c electric motors having low rotor inertias are reported to be offering equivalent performances in ranges below 10 horsepower. Where power requirements are greater, as with the equipment shown in Fig. 2–18, hydraulic drives are predominant. Electric drives have also worked well on numerically controlled contour lathes where, again, the moving members do not require the amount of power necessary to move the elements of very large milling machines. In the hydraulic drive arrangement shown in Fig. 2–19, the hydraulic motor is attached to a recirculating ball-nut lead screw and is operated by a servo hydraulic valve. The feedback package containing the resolver is behind the hydraulic motor and geared directly to

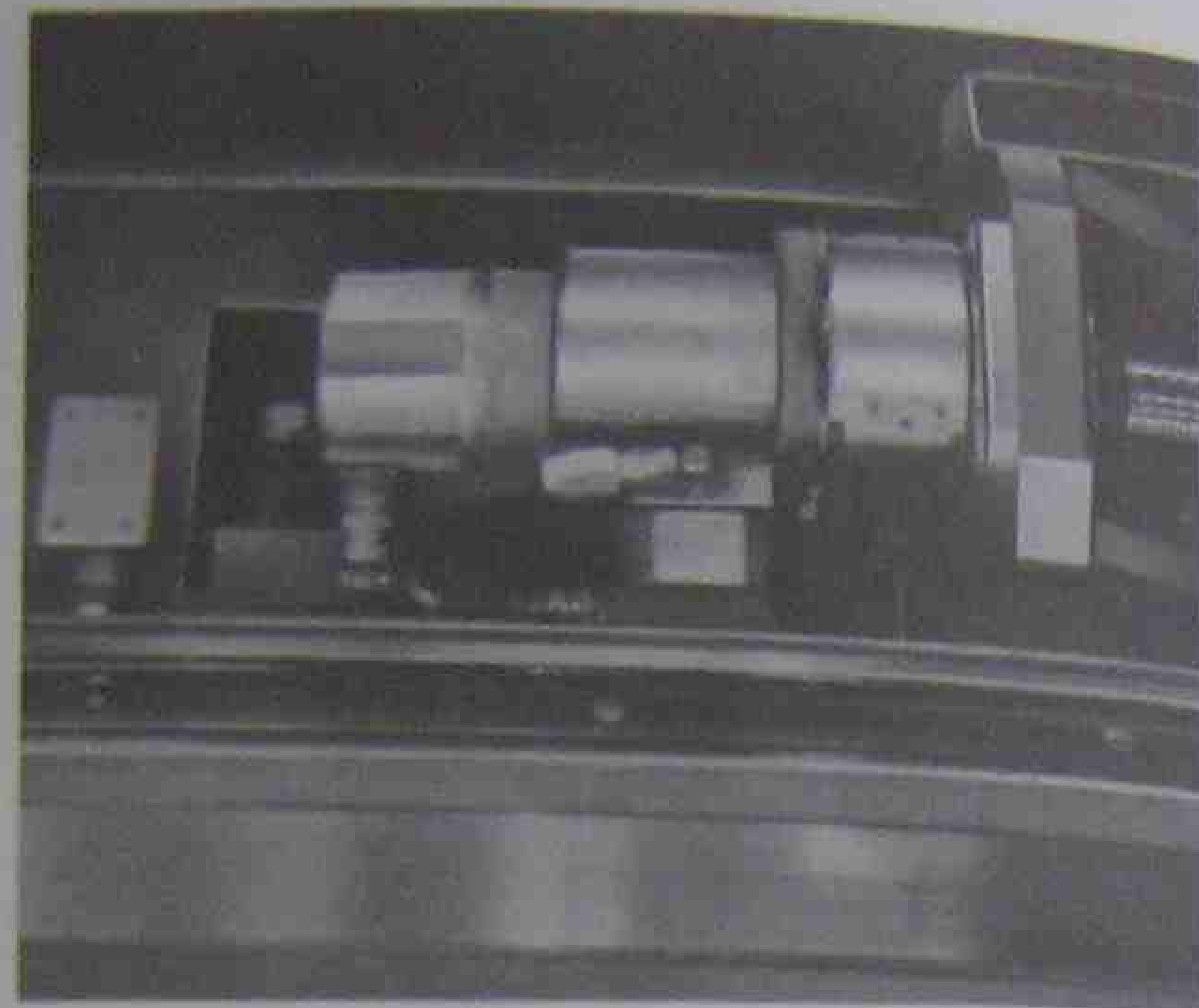


Fig. 2–19. This hydraulic drive mechanism comprises a hydraulic motor, servo valve, and feedback package. The recirculating ball-nut lead screw to which the drive motor is attached is also shown.

Courtesy of Pegasus Laboratories, Inc.

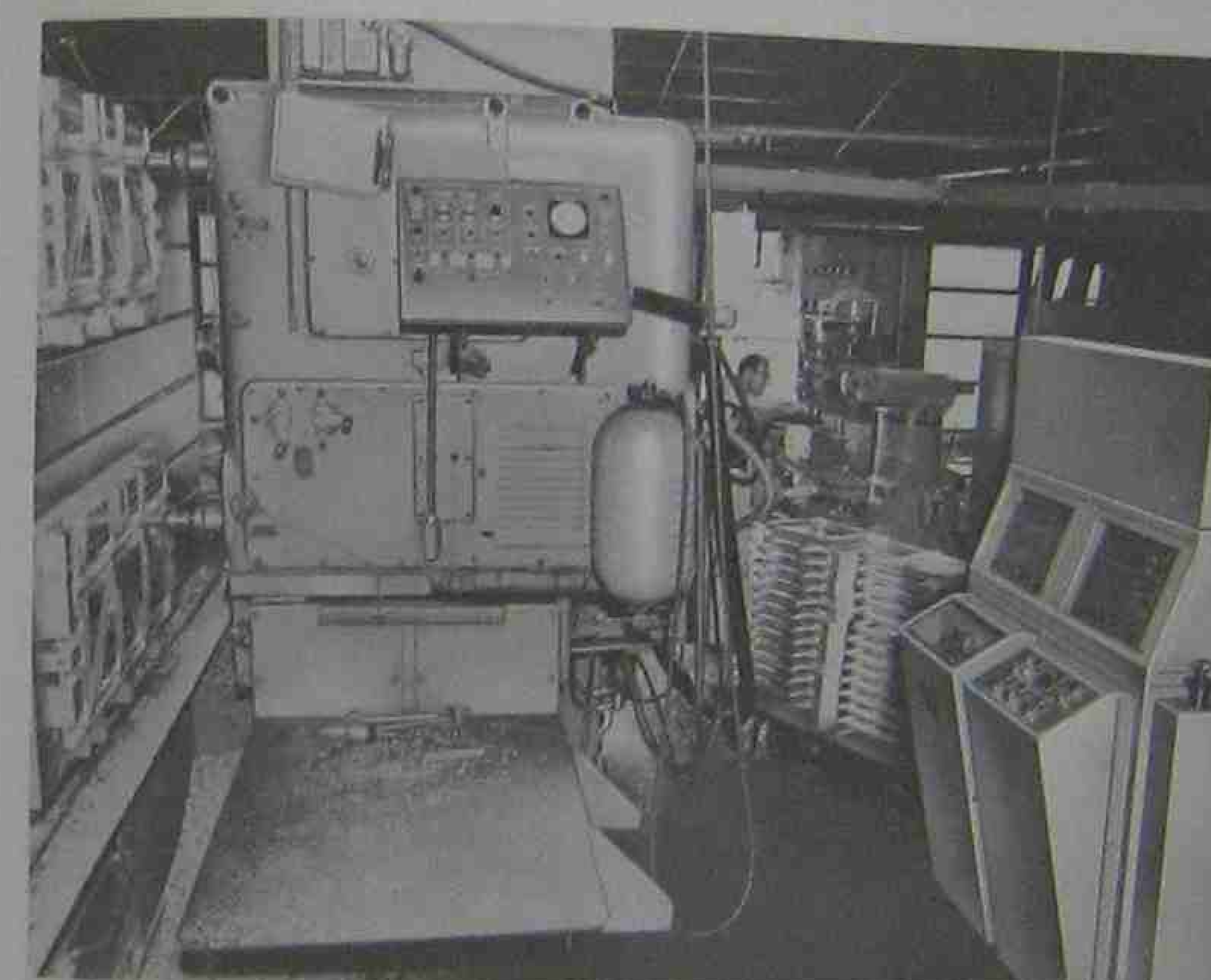
Fig. 2–20. This hydraulic servo valve regulates the amount of fluid to be passed to the hydraulic motor. This regulation depends on the distance to be moved and the velocity as called for on the punched tape. This unit measures about four inches across.

the shaft which rotates the lead screw. A close-up view of a valve similar to the one used in the drive assembly of Fig. 2–19, is shown in Fig. 2–20.

MACHINE TOOL

While early numerical control machines were modifications of conventional machines, most of the present day NC machines are especially designed for numerical control machining. Rigidity, vibration, friction along the ways, and mechanical response are some of the major design considerations which, although serious with conventional machines, are paramount when an NC machine is considered.

Two essential requirements of numerical control operation are low friction[5] between moving members and accurate response to commands. The first requirement is generally satisfied by properly designed slides and ways and the second by anti-backlash recirculating ball-nut lead screws and a well-balanced



Courtesy of The Bunker Ramo Corp.

Fig. 2–21. This retrofitted machine was originally designed and built over 39 years ago as a tracer control profiler. Chief features of the conversion were the addition of the electronic control system, feedback packages and adaptation of the valves to match the signals of the control system.

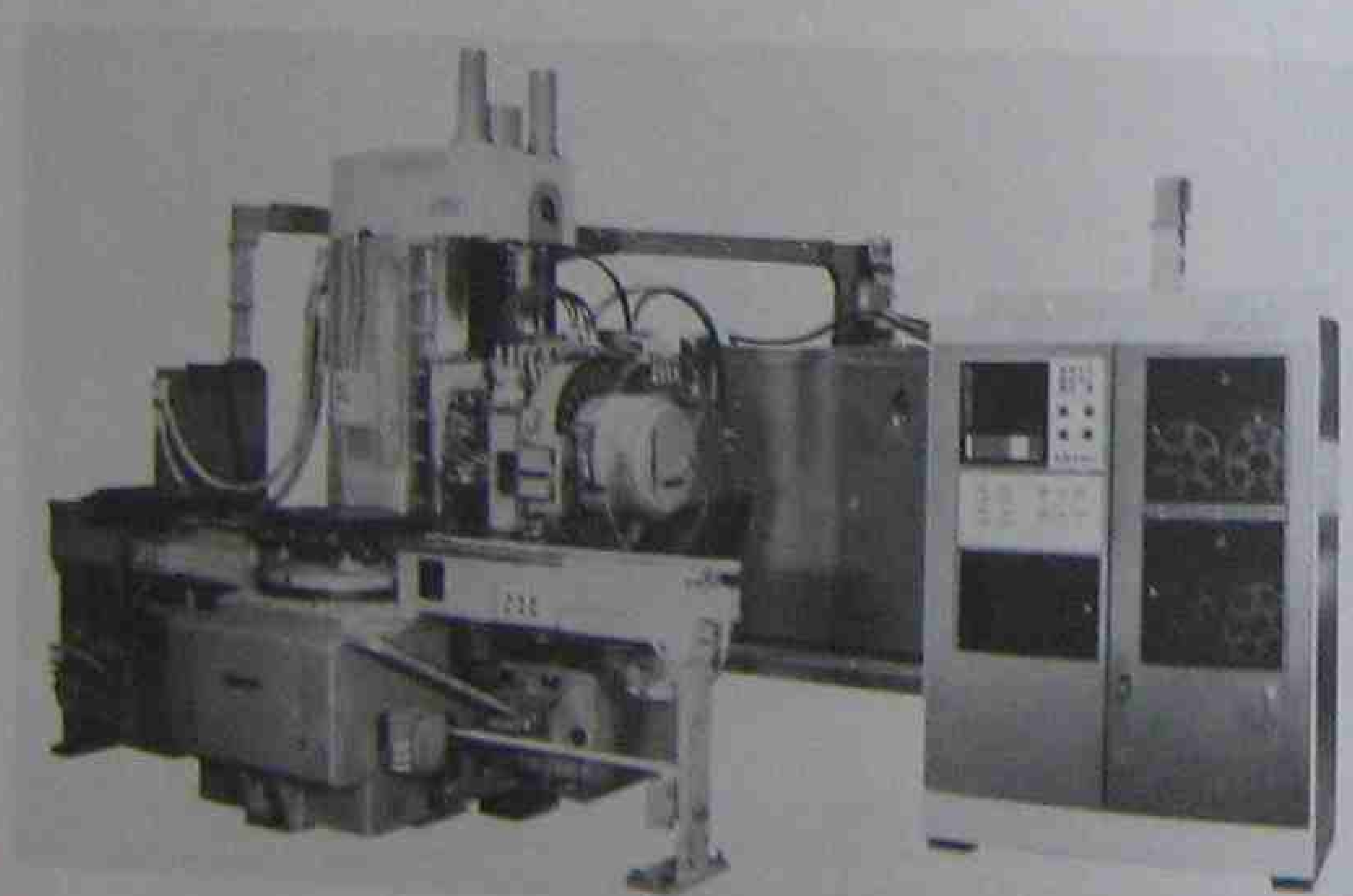
---

[5] Too low a coefficient of friction may also cause problems.

servo system. The extent of the requirement for accurate response will depend among other things on whether the machine tool is of the point-to-point or contouring variety; the feed rates sought, and the type of control system.

Retrofit The term "*retrofit*" which means *retroactively fitting* control systems and other required numerical control gear to a conventional piece of equipment in the field has been given serious consideration since the first numerically controlled machine was demonstrated in 1952. As a matter of fact the first numerically controlled machine was a retrofitted tracer profiler. (Refer to Fig. 1–2.) Since that time many machine tools have been retrofitted, most of which have proven successful.

There are basically three types of retrofits. One involves the adding of a numerical control package, including control system, drives, and feedback units, to a medium or large size machine not originally designed for numerical control. This usually involves special engineering design and, unless the machine is of high original cost, is generally not economically practical. Age of the equipment appears to have little bearing on whether a machine is suitable although costs to convert will most likely be higher for older equipment. Figure 2–21 shows a retrofitted three-axis profiler. The machine tool was originally built in 1942 as a tracer profiler. Large vertical boring mills dating back as far as 1909 have also been retrofitted. Retrofitting of conventional equipment, however, has practical economic and operating limitations:



Courtesy of C. O. Haffacker Company

Fig. 2–22. The machining center shown to the left was originally built with a hardwire control system. It has been rebuilt and retrofitted with a late model CNC control system. The hardwire version is shown in Fig. 6–6.

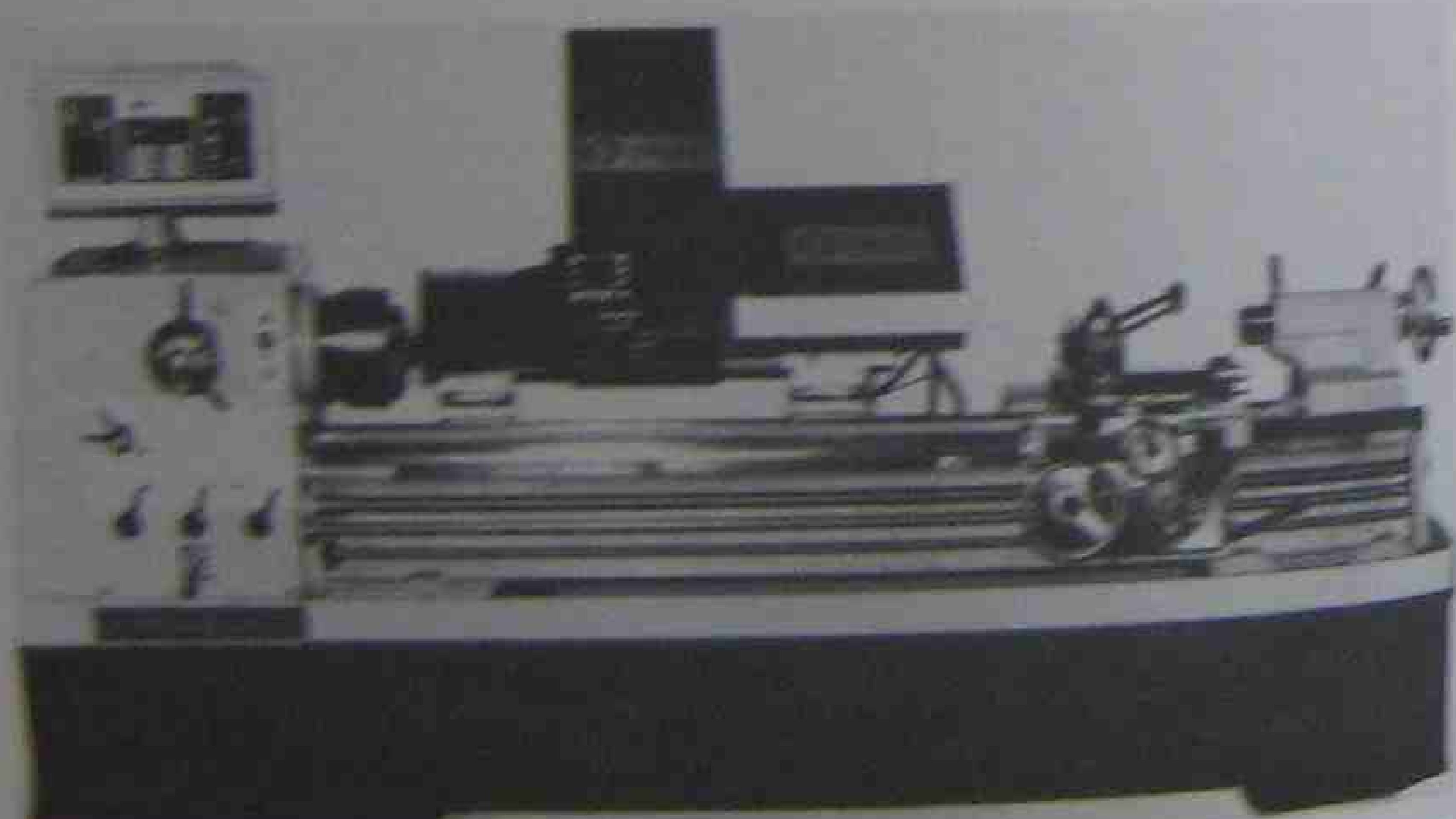
Courtesy of Anilam Electronics Corp.

Fig. 2–23. The small conventional milling machine shown has been retrofitted with a CNC system. The complete retrofit can usually be made within a day.

1. To make an original retrofit feasible, the initial cost of the machine, plus installation, should be relatively high. Large machines such as profile millers or vertical boring mills are most suitable. It would not be practical, for example, to adapt a $25,000 contouring control system to a machine that originally cost in the neighborhood of $10,000, particularly in view of special "one shot" engineering costs which may add another $20,000 to the overall cost.
2. The machine tool to be retrofitted should be in reasonably good condition. If extensive rebuilding is required, a new numerically controlled machine may be in order.
3. Some types of machine tools lend themselves better to numerical control retrofitting than others. The most suitable are those that were originally designed for automatic tracer operation—especially hydraulically operated tracer units. In these cases much of the equipment such as hydraulic valves and motors can be utilized.

The second type of retrofit involves the replacing of a hardwire control system with a CNC system. This is proving to be a reasonably popular route since, in many instances, the machine tool is in reasonably good shape while the control system has become obsolete. Even if the machine were to require some rebuilding, the economics can be attractive. One such retrofit is shown in Fig. 2–22.

The third type of retrofit involves smaller conventional machines, such as knee mills. A typical installation is shown in Fig. 2–23. In this case a preengineered retrofit package, consisting of the control system, drive units, feedbacks, spindle actuator, and ball-bearing lead screws, is included. The complete retrofit can usually be accomplished within a day's time at the customer's site.



Courtesy of Dataflow Division, Miník Tronn Inc.

Fig. 2–24. An add-on retrofit arrangement that includes a slide, saddle, turret, and NC system is shown attached to a conventional lathe.

An add-on type retrofit is shown in Fig. 2–24. In this case a complete saddle and turret arrangement is attached to the bed of a conventional lathe. The headstock, tailstock, and bed of the conventional lathe are still usable.

Some advantages of retrofitting include:

1. Less initial expense. With large machine tools costing on the order of $200,000 capital investment savings of 50% or more are not uncommon.
2. Shorter lead time. Generally a machine may be retrofitted in much less time than required to deliver a new machine tool. Quite often the machine may be retrofitted on-site; however, this will depend on the amount of mechanical rework required.
3. The same foundation may be utilized, thus saving installation costs.

The key disadvantage is that the retrofitted machine will probably lack the up-to-date features of a new machine.

QUESTIONS TO CHAPTER 2

1. A ______________ is a rotary device for feeding signals back to the control system in order to "close" the servo loop of a numerical control installation.
2. Buffer storage has been particularly helpful to numerical control "buffs" operating point-to-point equipment. (True, False.)
3. Second generation machines refer to those that do not require computer programming. (True, False.)
4. Most control systems operate on an a. open; b. closed; or c. plastic, loop. (Select one.)
5. Fourth generation refers to CNC control systems. (True, False.)
6. CNC systems were introduced in approximately 1965, 1968, 1972, 1978. (Select one.)
7. CNC systems are far superior to their hardwire predecessors. Name at least two advantages of the CNC system.
8. Ball-nut precision screws are highly accurate, however, wear excessively, due to high frictional coefficients. (True, False.)
9. Owing to accuracy requirements, the feedback of an NC system can only be taken directly from the sliding portions of a machine, such as the table. (True, False.)
10. A tape reader that is classified as "slow" has which of the following reading speeds:
    a. 60 characters per minute
    b. 500 characters per second
    c. 60 characters per second.
11. In the servo system shown in Fig. 2–13, if the ambient temperature were to drop below the 67°F mark, the furnace would most likely:
    a. Stop
    b. Explode
    c. Get much hotter
    d. Continue operating in the same manner.

12. A hydraulic valve regulates:
    a. Temperature of the fluid
    b. Amount of fluid
    c. Consistency of the fluid
    d. Cleanliness of the fluid, in microns.
13. There are two feedbacks in an NC servo system. One is a position feedback and the other is a ______________ feedback.
14. If a stepping motor that is tied directly to a lead screw rotates 1.8° per pulse, how far would a 5 pitch lead screw move a machine tool table if the motor received 254 pulses.
15. Hydraulic drives are far more popular than electric drives. (True, False)
16. The term *retrofit* means:
    a. retrofing it
    b. retroactively fitting
    c. retooling for roll fits.
17. a. A machine being considered for retrofit has the following history and characteristics:
    1. Large vertical boring mill
    2. Purchased in 1904
    3. Original cost, $100,000
    4. Cost to retrofit, including rebuilding, $150,000
    5. Comparable new numerical control machine: $550,000
    6. Delivery on new machine is 14 months. Delivery of the retrofit machine is 8 months.

    Would this machine be a reasonable consideration for retrofitting?

    b. Another machine has the following characteristics and history:
    1. Horizontal knee mill
    2. Purchased in 1960
    3. Original cost, $22,000
    4. Cost to retrofit, $70,000
    5. Comparable new numerical control machine: $92,000
    6. Delivery is the same for a retrofitted or new machine.

    Would this machine be a reasonable consideration for retrofitting?

CHAPTER THREE

Tape—The Controlling Medium

While it is possible, with most numerical control systems, to insert instructions manually, the general practice is to prepare *complete* instructions beforehand and store these data on some permanent medium. This capability accounts for the relatively higher efficiencies associated with numerically controlled equipment. During the past two and a half decades, which cover the approximate life span of numerical control in this country, numerous means have been adopted for storing machine instructions. These have ranged from a five-inch-wide punched plastic tape through motion picture type film, to computer cards. As the number of control systems and machine applications increased, it became evident that a standard type of instruction-storage medium would be required.

In order to best satisfy user requirements and expand the utilization of numerical control, manufacturers and users working through technical committees cooperated in the development of common "ground rules." The specifications developed by these committees are not intended to dictate the means of accomplishing requirements but are offered as recommended performance

Special Note. The dimensions used in this chapter are in inches. Millimeters could be substituted for inches and meters for feet.

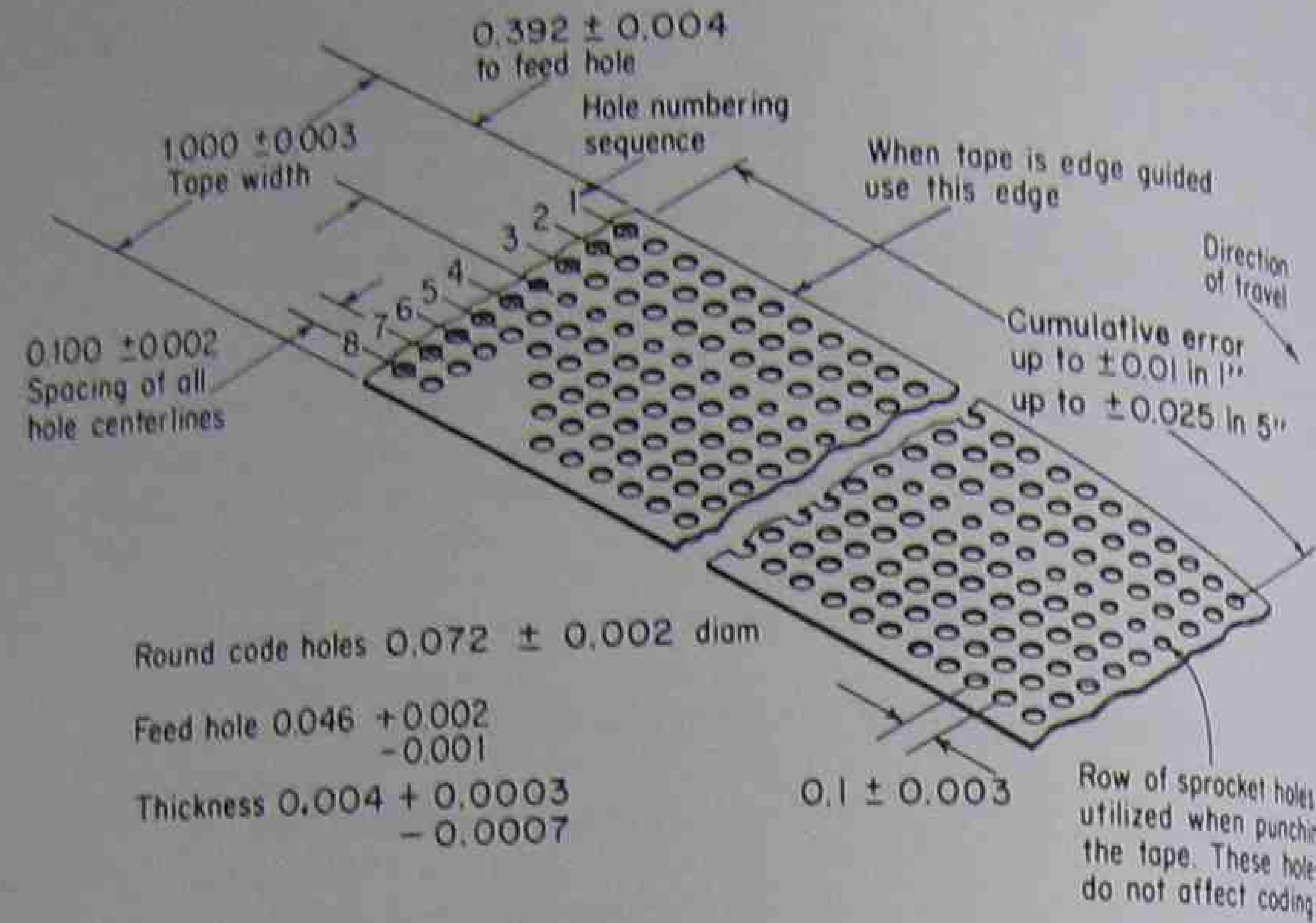


Fig. 3–1. A one-inch-wide punched tape having 8 rows of holes as described above has been selected as a numerical control standard for the controlling medium. Prior to this standardization, many different inputs were being marketed, ranging from five-inch-wide plastic tapes to punched cards. The development of a standard tape enables a user to "feed" many different numerically controlled machines having different control systems with the same stock size tape and utilizing the same type tape-punching device.

guidelines in order to improve operational compatibility between numerous control systems and machine tools. The organization that has thus far been most instrumental in the development of numerical control standards is the EIA[1] (Electronics Industries Association). One of the most significant agreements proposed by this group has been the type, size, and hole configuration of the storage medium. This is defined in EIA standards Nos. RS 227-A, 244-B, 274-D. (See Chapter 10.) Practically all of the control systems now being built will accept a storage medium conforming to these standards, which specify that the storage input medium shall be a punched tape, one inch wide and having eight rows of holes in an ordered fashion running along the length of the tape. Requirements for the size and spacing of the holes as well as tolerances for the thickness and width of the tape have also been established. (See Fig. 3–1.)

At the control system the tape is usually handled on reels. (See Fig. 3–2.) Reels are considered a necessity with hardwire contouring systems because of the length of tape generally required. Most CNC (softwire) systems having tape input also utilize reels. The tape reading mechanism with CNC systems, however, is less complex and less costly than with the hardwire systems since

[1] Other organizations have been active in the numerical control standards area. See Chapter 10.

much longer sections of tape data can be input in a single pass with CNC than with the hardwire system. Depending on the data storage capacity of the CNC system, up to a complete part program may be input in a single continuous run of the tape through the tape reader. These stored data are then fed to the machine tool as required. This, in contrast to the hardwire system which must receive and digest an integral instruction, via the tape reader, for each motion of the machine tool. This requires that the tape reader make many starts and stops while reading and processing the tape data.

Tape Coding Figure 3–3 describes the characteristics of the one-inch standard tape starting with the most basic definition—the *bit*. Referring to Fig. 3–3:

A *bit* is defined as a condition of one of two possible states (i.e., either "on" or "off"). With respect to the tape, it means either the presence or absence of a hole.

A *character* is a collection of holes positioned on one line across the tape representing either a number, letter, or symbol.

A *word* is an ordered set of characters which may be used to cause a specific action of a machine tool. Thus, in the example (Fig. 3–4), x + 12345 would

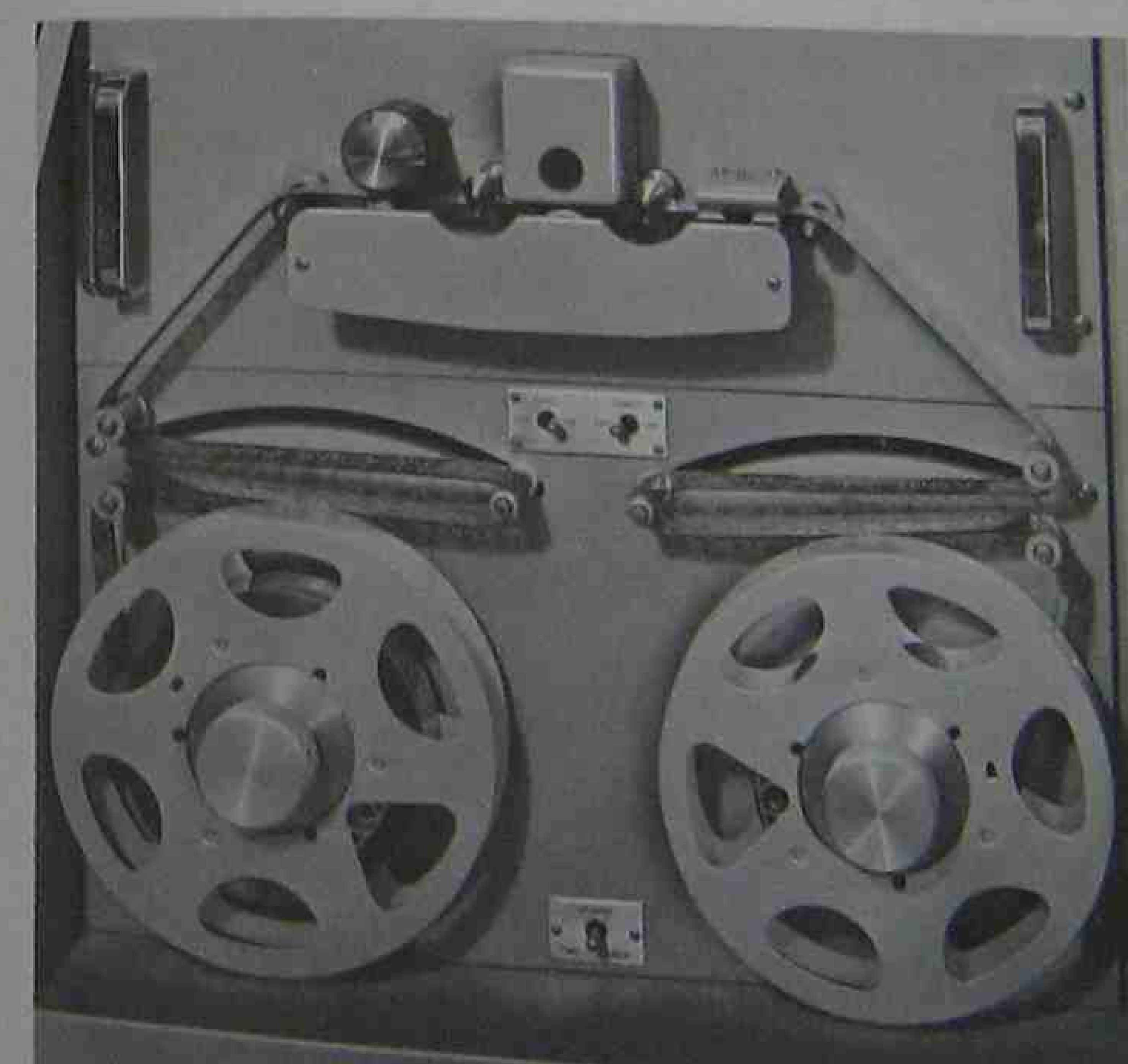


Fig. 3–2. The reels shown above, for a hardwire control system, handle the punched tape as it passes through the photoelectric tape reader. A special code punched at the end of the tape run actuates a signal that automatically stops the operation and rewinds the tape.

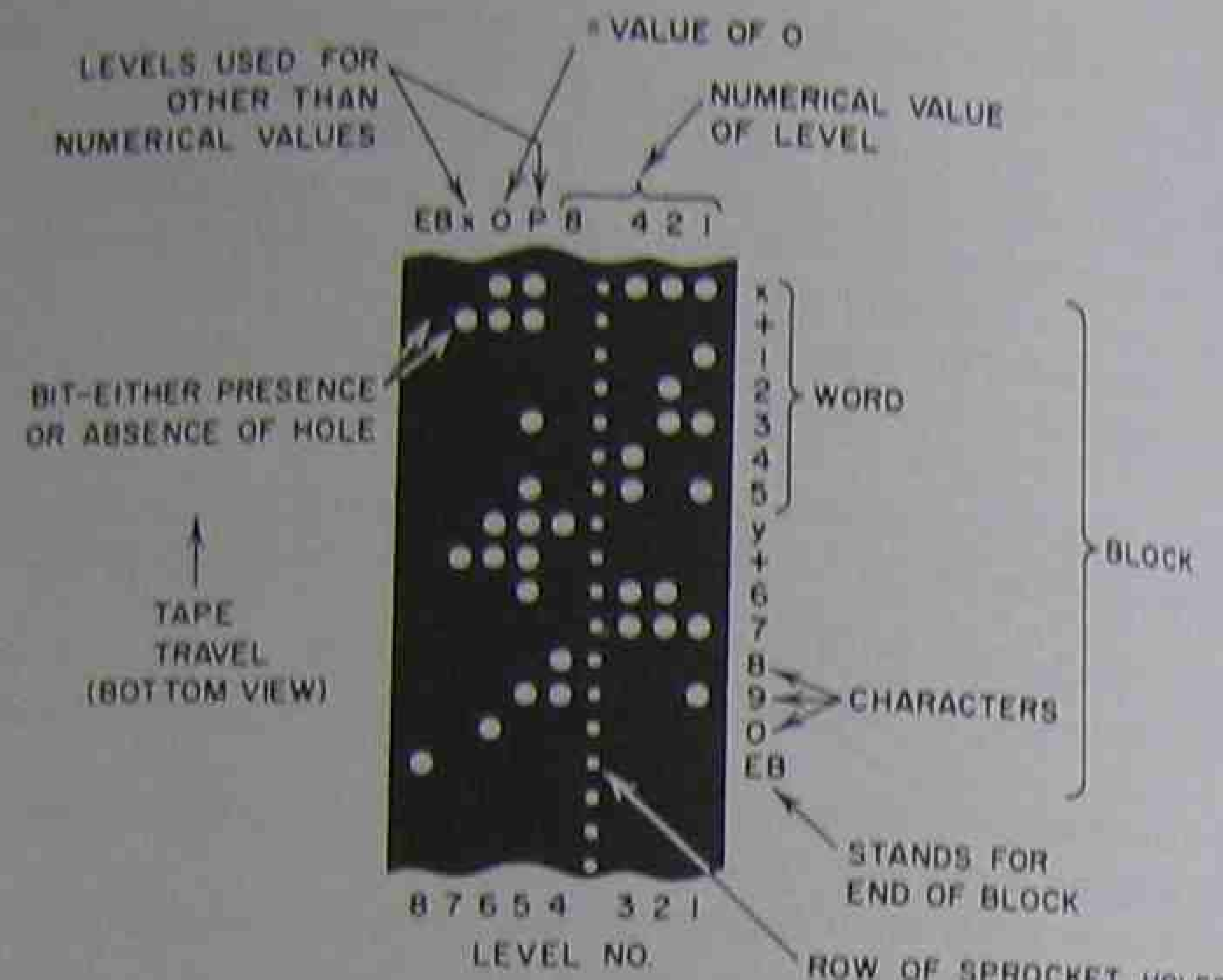


Fig. 3–3. The Binary Coded Decimal format per RS-244-B illustrates the arrangement and definitions of coded machine instructions. Machine motions corresponding to the above tape codes are shown in Fig. 3–4.

represent movement of the machine head 1.2345 inches in the positive x direction; *y* + *67890* would represent the instruction for moving the head 6.7890 inches in the positive *y* direction. In punching the tape, the decimal point is generally ignored in hardwire systems and may be included or excluded in many CNC systems.

A *block* is a word or group of words considered as a unit and generally

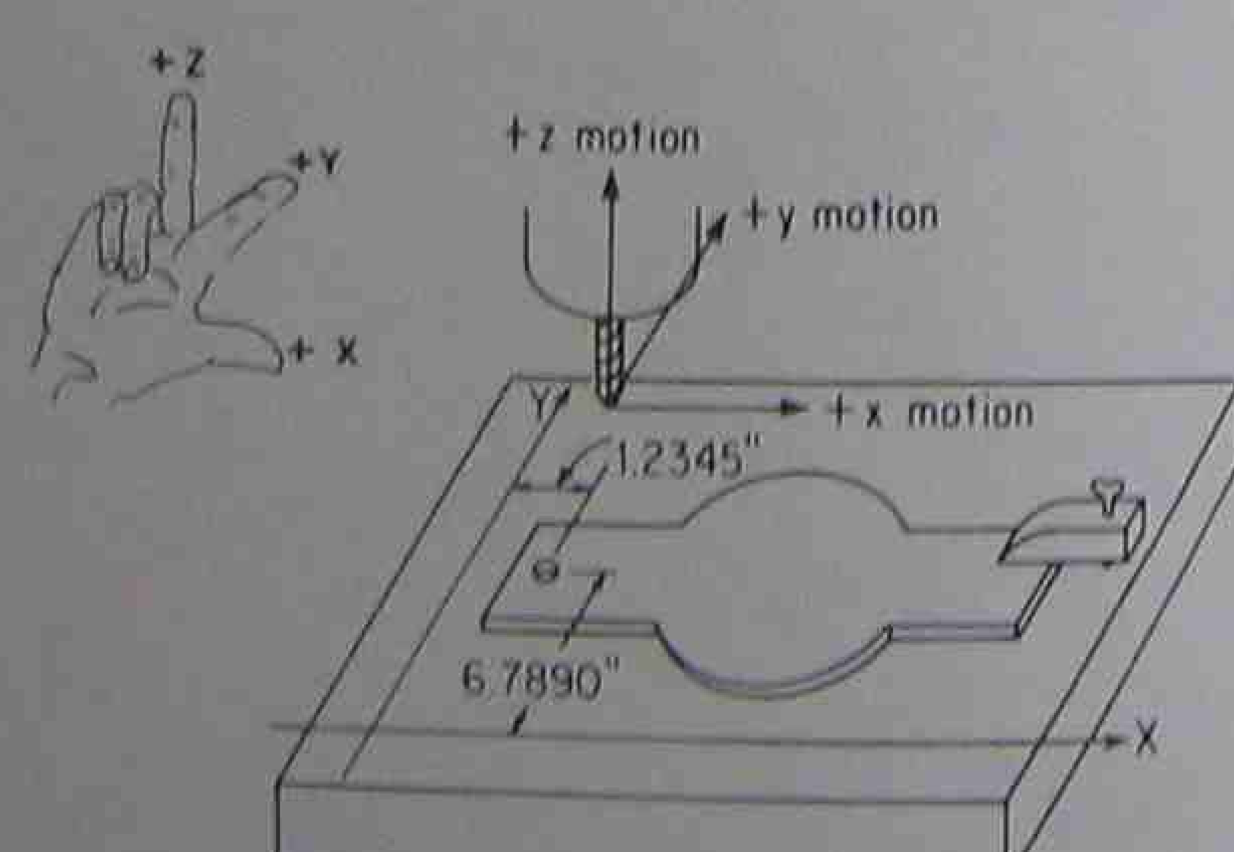


Fig. 3–4. The hole shown in the part fastened to the stationary table is 1.2345 inches from the Y axis in the positive *x* direction and 6.7890 inches from the X axis in the positive *y* direction. This conforms to the *x* and *y* words shown on the tape illustrated in Fig. 3–3, also the motions conform to the "right-hand rule," as shown. (Also see Chapter 10, p. 292.)

offering one complete instruction for a specific machine movement. Blocks are separated by an *end of block* character. The tape is read one block at a time when passing through the tape reading head in a hardwire control system and read multiblock with a CNC system. The number of blocks that can be read and stored depends on the storage capacity of the CNC system. Movement of the tape with a hardwire system will depend on the length of time required to perform the operation in a block.

The *level, channel,* or *track* all mean the same thing and refer to the lines of holes running *along* the tape.

The fifth or *P* level is referred to as the *parity* level with a hole punched whenever the number of other holes in the particular character is an even number (odd number with RS-358-B). Thus, a hole is punched in the *P* level when the number 3 is expressed since the value of 3 is a combination of levels 1 and 2 and constitute two holes, or an even number. Parity serves as a form of safety check, since occasionally when preparing a tape, a punch may not pierce clean through resulting in a possibility of reading an incorrect character thereby causing part-damage. Also assuming the parity hole is punched; and another is missed, then the even number of holes in the character would cause the machine to automatically stop. Parity checking is also a common practice in computer data processing.

It will be noted that the number of holes across the tape (i.e., in a character) in Fig. 3–3 is odd. This conforms with the EIA standard RS-244-B and was adhered to for many years. However, with the increasing prevelance of the computer in the NC field and particularly with the emergence of CNC systems,

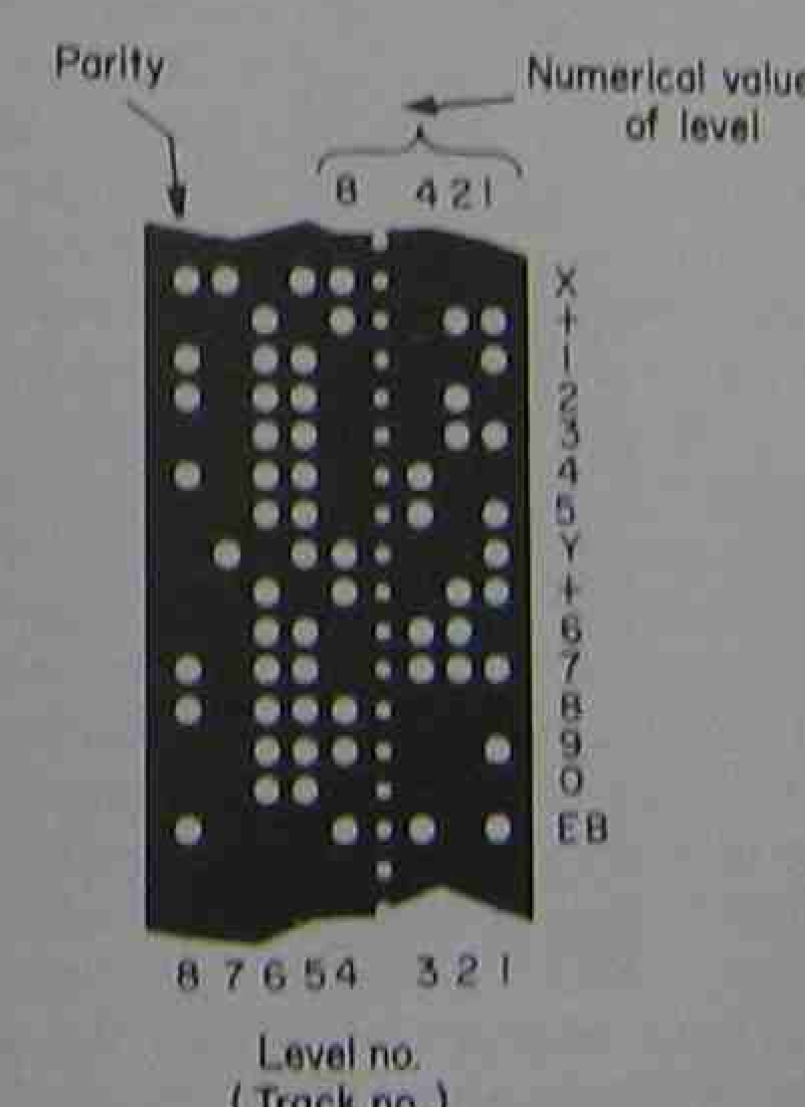


Fig. 3–5. The difference between RS-244-B and RS-358-B is in the character coding. The formation of the blocks is identical. The block formation shown above is therefore the same as shown in Fig. 3–3.

the trend is toward the EIA standard RS-358-B. While the EIA has not disowned RS-244-B it has noted that "users and builders of numerically controlled equipment should be aware that the flexibility and expandability of the code (RS-358-B) . . . make it more applicable than the RS-244-B code for the newer, advanced numerical control systems." In either event most CNC systems can handle either code. It should be pointed out that the only change from RS-244-B to RS-358-B is in the character coding. The formation of the words and the block are unchanged. The tape, considering RS-358-B, for the identical block of data as shown in Fig. 3–3 would appear as shown in Fig. 3–5. It will be noted that, with RS-358-B, there are more holes across the tape for comparable characters, in most cases. Another significant difference with RS-358-B is that the number of holes across the tape is now *even* instead of odd, as with RS-244-B. This is therefore considered an even parity as compared to the odd parity of RS-244-B. The numerical portion of the characters (i.e., levels 1 through 4) does not change. The change is noticeable in the balance of the levels (tracks) on the left-hand side of the tape, (i.e., track nos. 5, 6, 7, 8).

In addition to numerical values and address characters such as x, y, or z, the 8-level BCD (Binary Coded Decimal) tape may also express alphabetic and symbolic characters. Figure 3–6 describes the character coding for the letters of the alphabet plus symbolic codes in addition to the numeric characters. As will be described later in this chapter and throughout the text, most of these alphabetic and symbolic characters have an assigned numerical control function.

In addition to the dimension words which specify distance movements in the various axes, a block may contain other characters. These characters, which are specified by EIA standards, make up other *words* which consist of a group of numbers headed by a letter *address* character. These words may precede or follow the dimension words. An example of a block containing functions in addition to dimension words is shown in Fig. 3–7. These functions and their application to a machine tool are also shown. Although it is technically feasible with most hardwire and CNC systems to mix the order of the words, the standards recommend that there be a consistency which follows the general order shown in Fig. 3–7. This is covered in further detail later in this chapter under the sections "Format Classification" and "Format Detail." Also, for interchangeability, the standards recommend that words not be repeated in any one block, although this is technically feasible with many CNC systems.

The *sequence number* may be inserted as the first word of any or every block. Its purpose is to identify a block or group of blocks so that the operator may locate the block in order to edit it via the readout on a CNC system or, with a hardwire system, to locate a position on the tape without having to read the tape itself. With a hardwire system the sequence number is read and displayed (via number lights) at either the control or operator's console as the tape passes the reading head. With a CNC system the sequence number is displayed along with the remaining data in the block. Also, with a CNC system a number of blocks may be displayed at the same time via a CRT (cathode ray tube) or similar readout device. The display is drawn from the memory of the CNC computer.

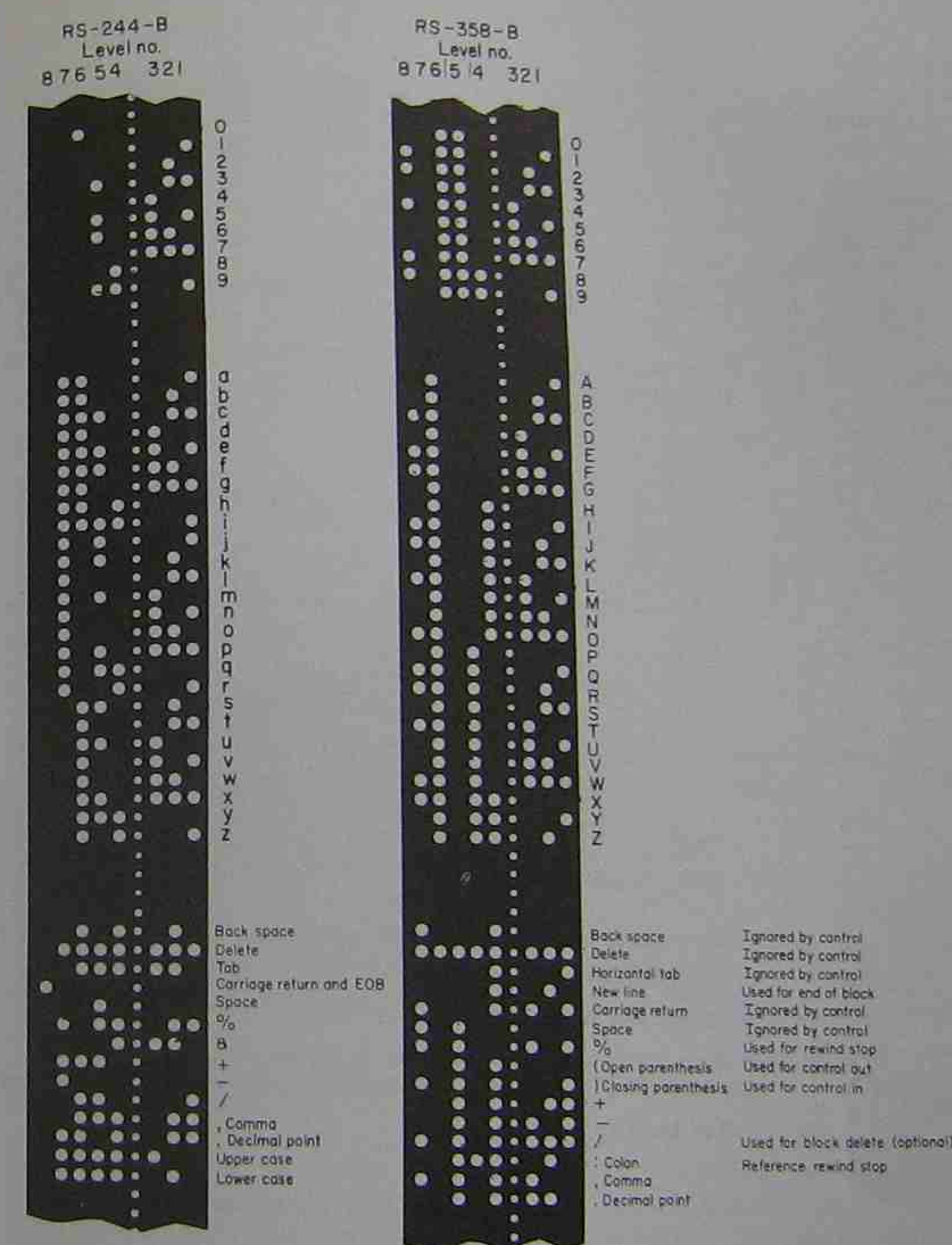


Fig. 3–6. Character coding for both RS-244-B and RS-358-B numerical control eight-level tape. (Tape preparation machines are shown in Figs. 4–1 and 4–2.) It will be noted that the RS-244-B letters are described as lower case, per the standard. Upper case letters are therefore shown in the text in order to reflect the latest standard, which is RS-358-B.

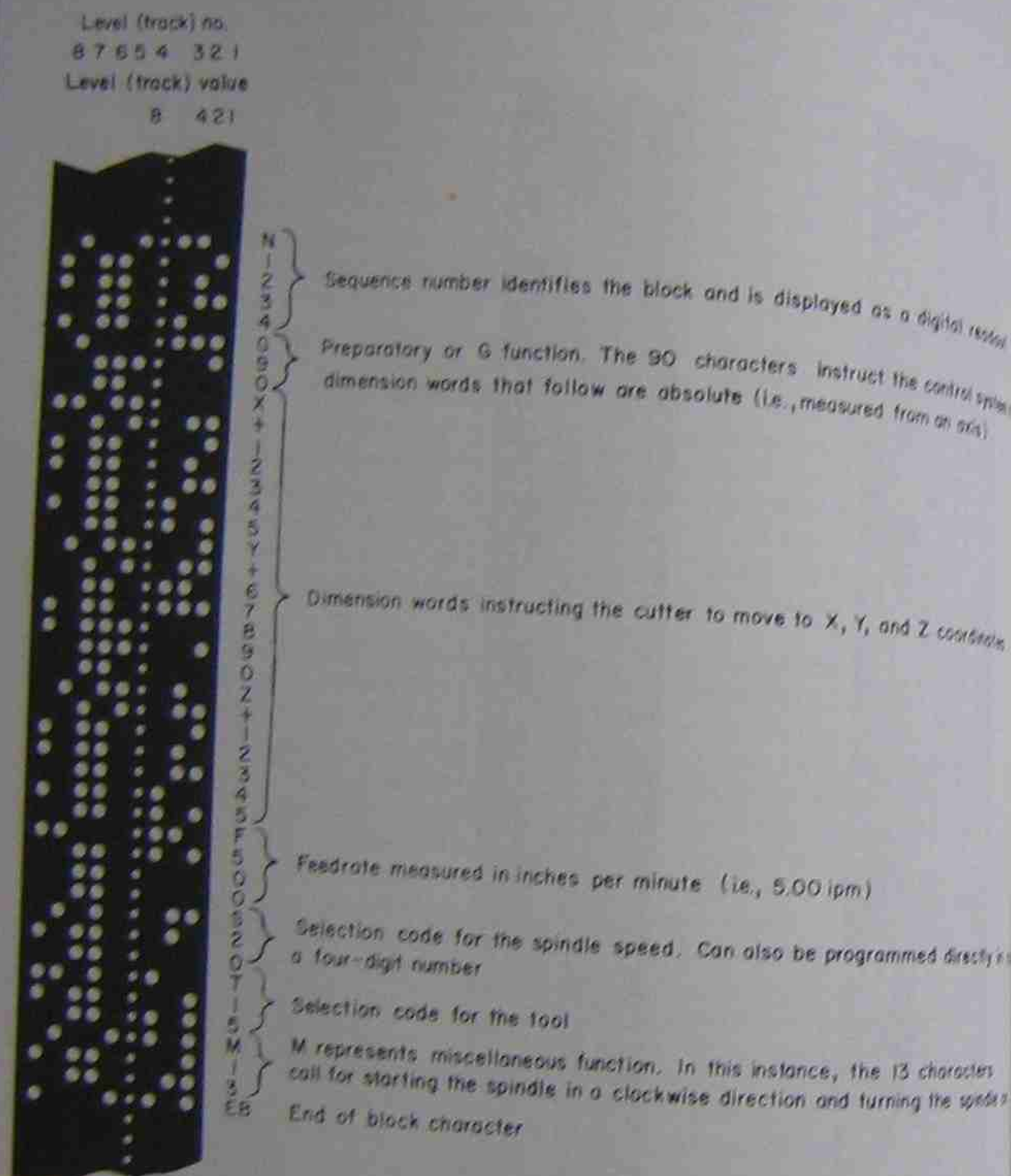
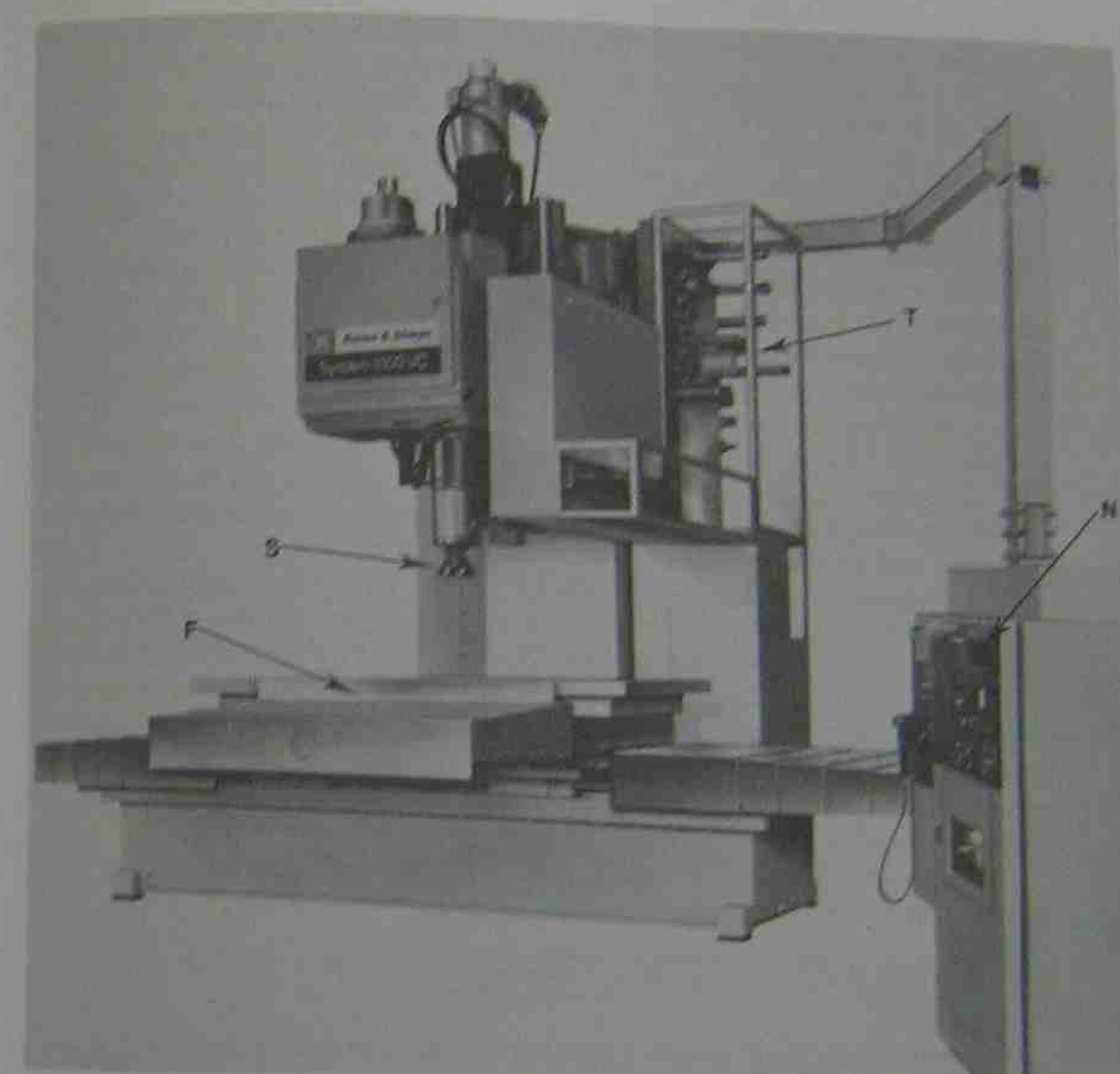


Fig. 3–7. The block of tape instructions shown above would cause the part to move to the absolute coordinate positions of $X = 1.2345''$, $Y = 6.7890''$, and $Z = 1.2345''$ at a feedrate of 5 ipm. At the same time the spindle is automatically adjusted to a speed



Courtesy of Brown & Sharpe Manufacturing Co.

represented by code number 20 and a tool is selected that matches code number 15. The initiation of the spindle in a clockwise direction and the turning on of the coolant is signaled by the miscellaneous word M13. The character EB ends the block. A three- or four-digit S or T code could have been used, depending on the machine tool being controlled. Letters, representing the functions N, F, S, and T, are shown in the photograph.

The *preparatory* or G function precedes the dimension words and "prepares" the control system for the information that is to follow in the block. Thus the G08 word (also referred to as *code*) instructs the machine to accelerate up to the specified feed rate, which is also noted later in the same block. By the same consideration, a G09 word would cause the machine to decelerate over the distance noted in the block. The G word also has the capability of changing the meaning of other words that follow in the block. For example, a G70 word means that the dimension words that would follow (e.g., X, Y, and Z words) are in inches. A G71 word would mean that the dimension words that would follow are in millimeters. Other G functions that have been standardized are shown in Table 3–1. Many of these functions will be explained further in the next chapter which covers manual part programming.

Also, it should be pointed out that the EIA standards are recommended standards and manufacturers are not obliged to follow them, although it is generally to their benefit and they most often do. Even the term *RS* which is a prefix to the EIA standards appropriately stands for *Recommended Standard*. A manufacturer, for example, may find it more suitable to utilize one of the G codes for a function other than that noted in the standards. This is more an exception than the rule however, and a more common occurrence is when a manufacturer utilizes one of the "unassigned" codes for handling a particular operation that is not covered in the standards.

The *feedrate* word may be expressed in either *inches per minute* (ipm) or *inches per revolution* (ipr). If expressed in inches per minute, a G94 code (word) would be required in the block. If expressed in inches per revolution, a G95 code would be required.[2] Inches per minute is the expression most used with milling machines and machining centers, whereas inches per revolution is more popular with turning machines (lathes). *Inches per revolution* is also popular with the Z-axis motion on machining centers, for such operations as drilling or tapping.

Older hardwire systems normally required a three-character coded number derived from velocity and distance. The formula:

$$\text{Feedrate Number} = \frac{\text{Velocity in ipm along the path of motion}}{\text{Distance along the path}} \times 10$$

The feedrate number assigned to a block would therefore be directly proportional to the velocity and inversely proportional to the length of travel described in the block. For example, if it were required to move a distance of 5.0000 inches at a feedrate of 50 inches per minute, then the feedrate number would be $\frac{50}{5} \times 10 = 100$. This feedrate coding is still carried in the EIA standards and is followed by a G93 code.

It is recommended that the S or *speed* word be noted directly in revolutions

[2] It should be pointed out that most systems can be set, electronically, to accept one of two, or several, alternatives. For example, a switch can be set on one of the circuit boards so that, if there were no G word expressed, such as G94 for ipm programming, the system would automatically assume ipm programming.

Table 3–1. Preparatory (G) Functions

Word (Code)	Explanation
G00	Used for denoting a rapid, or other traverse, rate with point-to-point positioning.
G01	Used to describe linear interpolation blocks and reserved for contouring. (Refer to Chapter 4 for explanation of linear and circular interpolation.)
G10 } G11 }	Used with some hardwire systems to express blocks of other than a *normal* dimension; a normal dimension being 9.9999 inches of movement in any one block. G10 would designate a *long* dimension of up to 99.9990 inches and G11 would designate a *short* dimension of up to 0.9999 inches.
G02 } G03 }	Used with *Circular Interpolation*. (Refer to Chapter 4.)
G04	A calculated time delay during which there is no machine motion (dwell). The amount of time is usually expressed as an X word. For example, a time delay of 1.5 seconds would be expressed as G04X1.5. Whereas the X word would usually refer to a dimension, the G04 code changes the meaning. The time delay can also be expressed as an F word, assuming it is preceded by the G04 code.
G05 } G07 }	Unassigned by the EIA. May be used at the discretion of the machine tool or system builder. Could also be standardized at a future date.
G06	Parabolic Interpolation.
G08	Acceleration code. Causes the machine to accelerate at a smooth exponential rate (assuming the control system and the machine tool have this capability).
G09	Deceleration code. Causes the machine to decelerate at a smooth exponential rate (assuming the control system and the machine tool have this capability).
G10 } G11 } G12 }	Normally unassigned for CNC systems (See explanation above of G10 and G11 for hardwire systems.)
G13 } G14 } G15 } G16 }	Used to direct the control system to operate on a particular *set* of axes. An example would be a machine tool having two heads which are not to operate simultaneously. Refer to Fig. 3–8. The set of axes for head *A* consists of motions in the *V* and *W* directions in addition to the *X'* motion of the table. A three-axis control system would therefore be sufficient to machine a part in these three simultaneous dimensions. If it were required to operate head *A*, a G13 code would precede the *X*, *V*, and *W* words. The same three-axis control system could also operate head *B* by preceding the *X*, *Y*, and *Z* words with a G14 word. Codes G15 and G16 may be utilized for other sets of axes, if required.

Table 3–1. Preparatory (G) Functions (Continued)

Word (Code)	Explanation

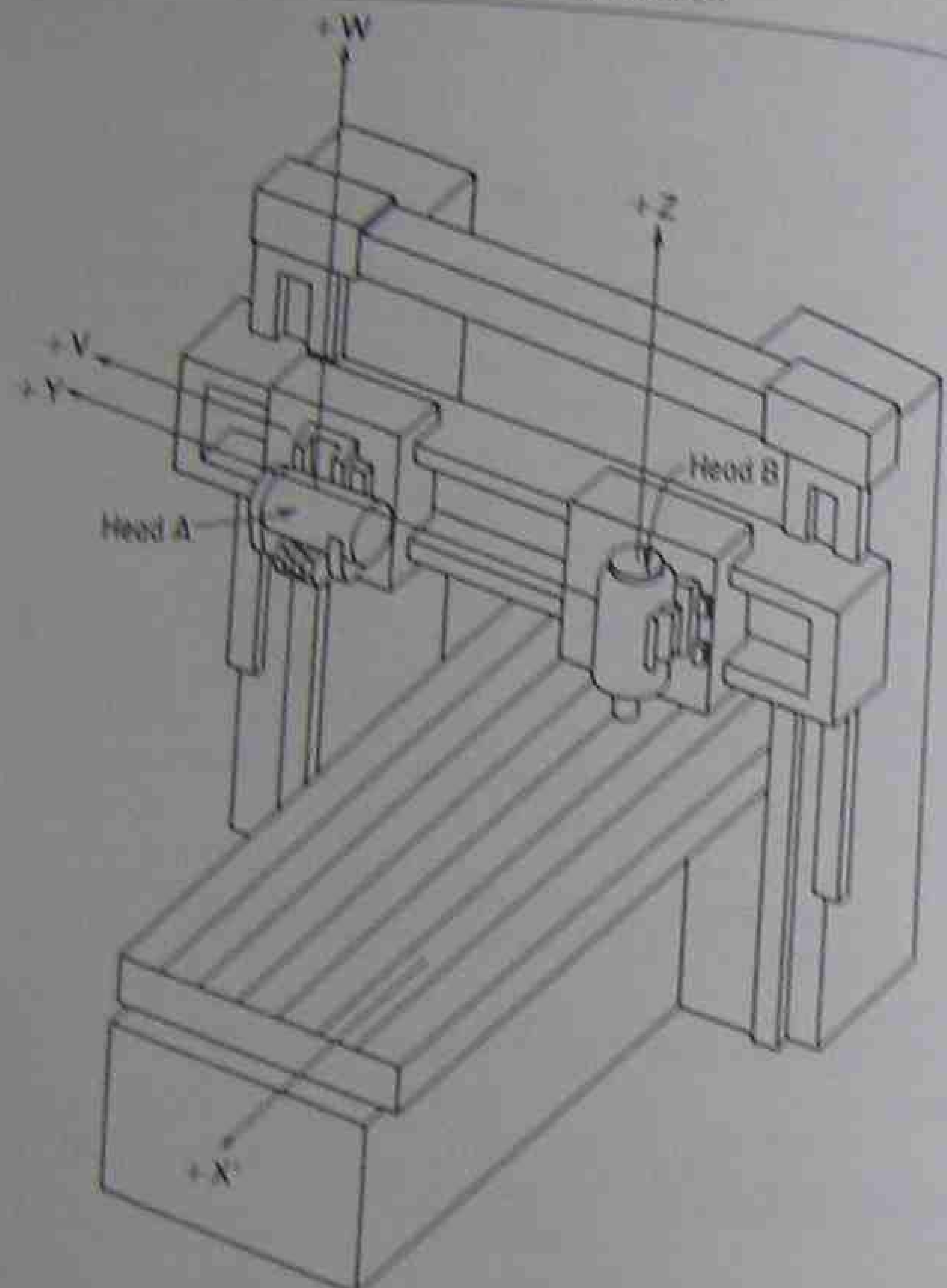


Fig. 3–8. The G13 through G16 codes are generally used to direct the control system to operate a set of axes. In the example above, one three-axis system can operate the two motions of either head Y and Z, or V and W, plus the third motion of the table X'. In this instance the heads are not operated simultaneously.

Word (Code)	Explanation
G17 G18 G19	Used to identify, or select, a coordinate plane for such functions as circular interpolation or cutter compensation.[3] If the sample part shown in Fig 3-9 were to be cut in the X–Y plane, a G17 preparatory code would be used. G18 would denote the horizontal or X–Z plane and cutting in the Y–Z plane would require a G19 preparatory code. The details of programming Circular interpolation are discussed in Chapter 4

[3] Refer to Chapter 10 for definition of cutter compensation.

Table 3–1. Preparatory (G) Functions (Continued)

Word (Code)	Explanation

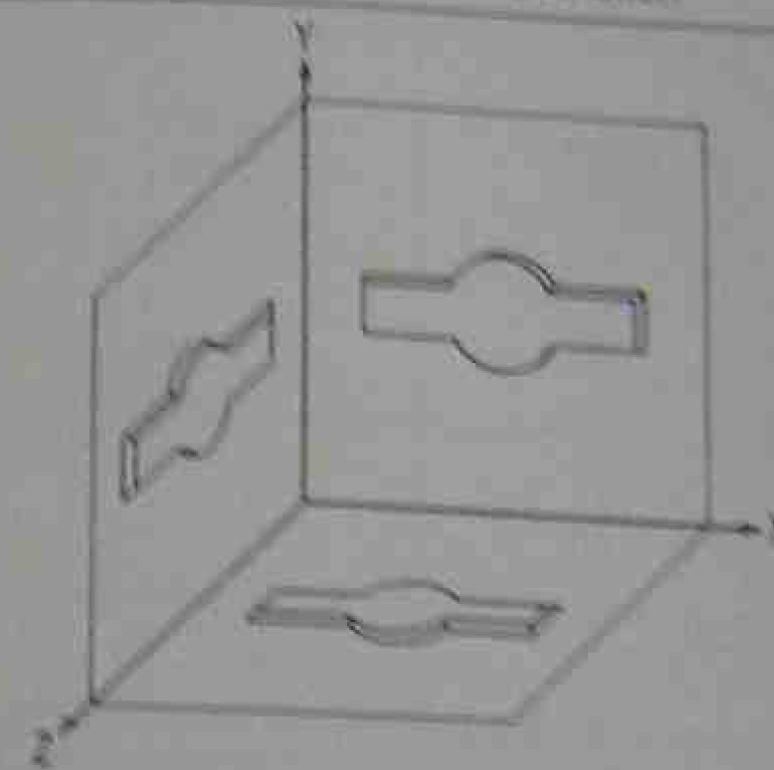
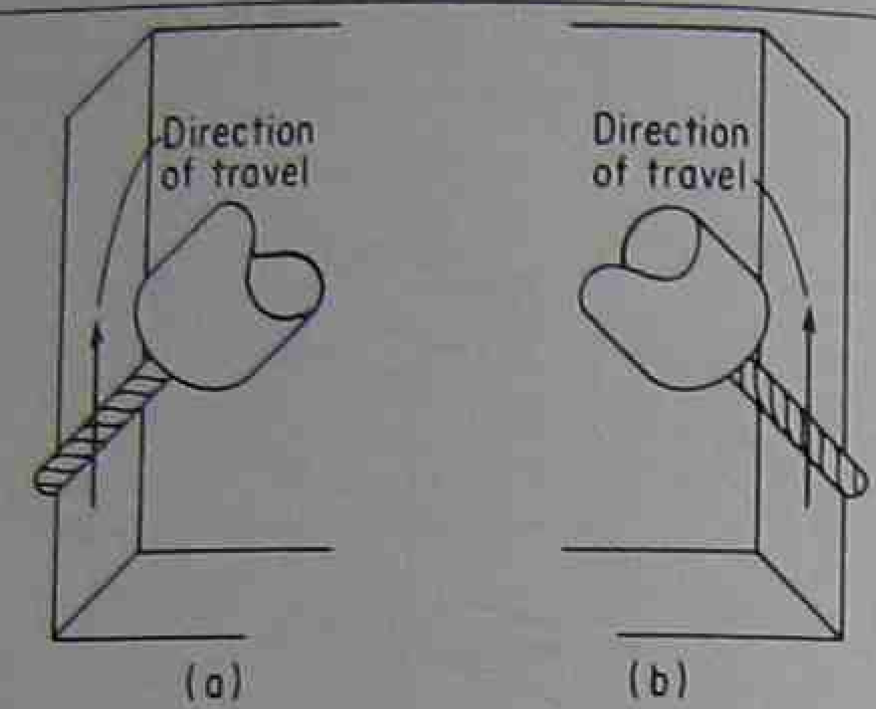


Fig. 3–9. The G17, G18, or G19 preparatory codes describe the plane selected for functions such as circular interpolation or cutter compensation. If the sample part (here shown positioned in the three planes) were to be machined in the X-Y plane, the G17 code would precede the other word instructions; G18 would refer to the X-Z plane and G19 to the Y-Z plane.

Word (Code)	Explanation
G20 through G32	Unassigned according to EIA standards; however, may be assigned by the control system or machine tool builder
G33 G34 G35	A mode selected for machines equipped with thread cutting facilities and generally referring to lathes. The G33 code is used when a constant lead is sought. Code G34 is noted when a constantly increasing lead is required and G35 is employed to designate a constantly decreasing lead.
G36 through G39	Unassigned
G40	A command which will discontinue any cutter compensation
G41	A code associated with cutter compensation wherein the cutter is on the left side of the work surface, looking in the direction of cutter motion (See Fig. 3–10.)
G42	A code associated with cutter compensation wherein the cutter is on the right side of the work surface. (See Fig. 3–10.)

Table 3–1. Preparatory (G) Functions (Continued)

Word (Code)	Explanation
	 **Fig. 3–10.** (a) The G41 preparatory code used with cutter compensation denotes that the cutter is on the left side of the work. (b) The G42 preparatory code used with cutter compensation denotes that the cutter is on the right side of the work.
G43 *G44*	Used with *cutter offset* and refers to the displacement normal to the cutter path to adjust for the difference between the actual and programmed cutter radii or diameters. G43 refers to an inside corner and G44 refers to an outside corner. *Offset* differs from *compensation* in that an offset distance is parallel to an axis, whereas compensation is the distance normal to the surface being cut regardless of whether the surface is parallel to an axis. (See Fig. 3–11.)
G45 through *G49*	Unassigned.
G50 through *G59*	Reserved for adaptive control. See Chapter 10 for definition of adaptive control.
G60 through *G69*	Unassigned.
G70	Inch programming.
G71	Metric programming.
G72	Three-dimensional circular interpolation—CW. (Refer to Chapter 4 for description of circular interpolation.)
G73	Three-dimensional circular interpolation—CCW. (Refer to Chapter 4 for description of circular interpolation.)
G74	Cancel multiquadrant circular interpolation.
G75	Multiquadrant circular interpolation.

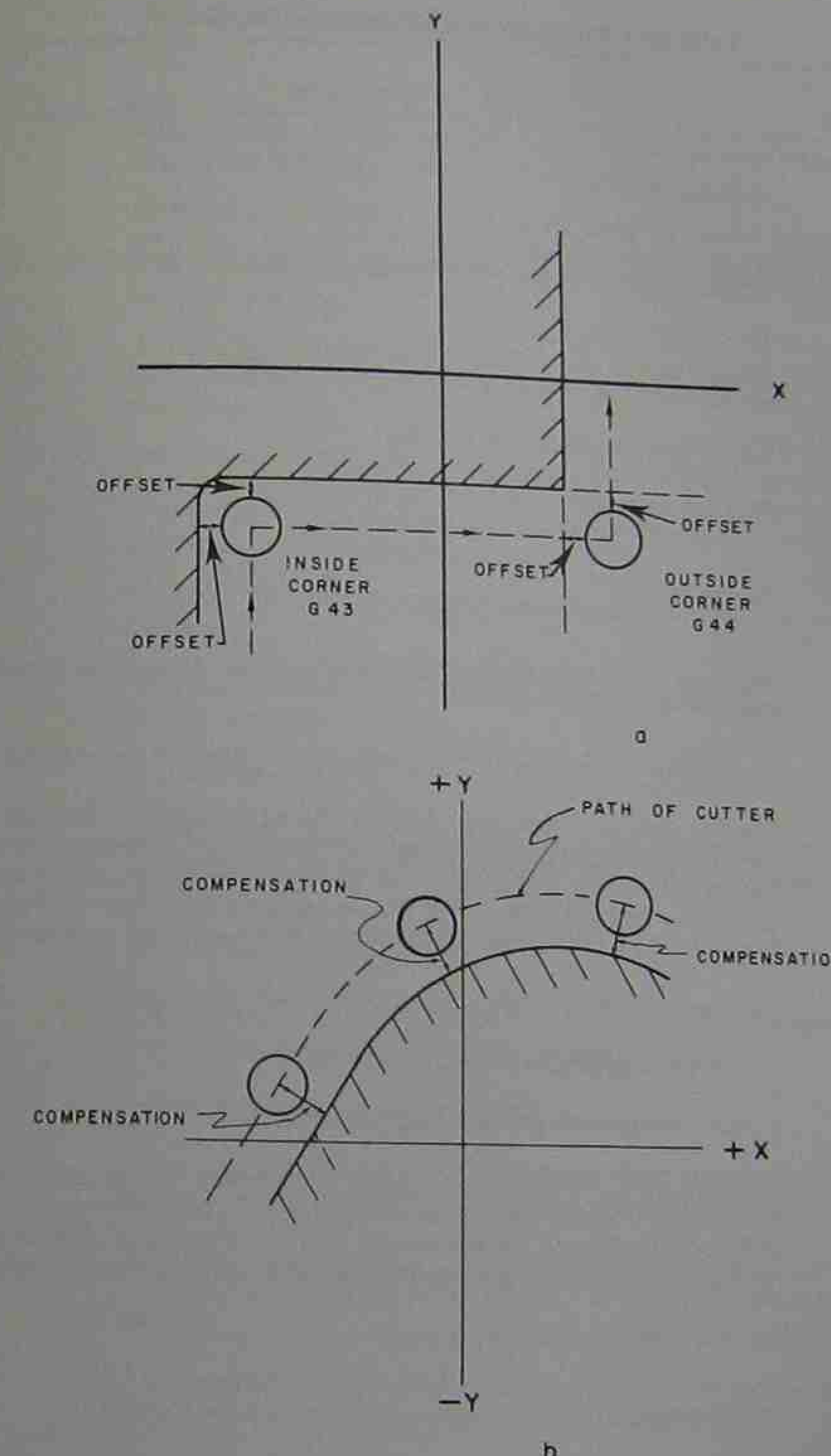


Fig. 3–11. *Cutter offset*, as illustrated in *a* above, denotes distance parallel to an axis. G43 is used when the cutter is moving toward an inside corner; G44 when the cutter is moving past an outside corner. Illustration *b* describes *cutter compensation* in which the compensated distance is normal to the work surface. Cutter offset is used with systems that are restricted to movements that are parallel to the axes (e.g., point-to-point systems). Cutter compensation is used with contouring systems. When G41 and G42 are used with contouring systems, G43 and G44 are not required.

Table 3–1. Preparatory (G) Functions (Continued)

Word (Code)	Explanation
G76 through G79	Unassigned.
G80	Command that will discontinue any of the fixed cycles G81 through G89.
G81	Drill, or spotdrill, cycle.[a]
G82	Drill with a dwell. An example would be spotfacing.[a]
G83	Intermittent, or deep hole, drilling.[a]
G84	Tapping cycle.[a]
G85 through G89	Boring cycles.[a]
G90	Absolute input. Note that the data input is to be in absolute dimensional form.
G91	Incremental input. Note that the data input is to be in incremental form.
G92	Used to preload registers to desired values. A common example would be to preload the axis position registers. This feature is helpful with manual part programming since, in effect, axes can be shifted and a series of operations repeated by using the same dimension words. This is explained further in Chapter 4, which covers manual part programming.
G93	Inverse time feedrate. (Explained previously in this chapter.)
G94	Inches (millimeters) per minute feedrate.
G95	Inches (millimeters) per revolution feedrate.
G96	Constant surface speed per minute. The spindle speed code units are surface feet (meters) per minute and specify the tangential surface speed of the tool relative to the workpiece. The spindle speed is automatically controlled to maintain the programmed value. Used principally with lathes.
G97	Spindle speed in revolutions per minute.
G98 G99	Unassigned.

[a] Explained, with illustrations, in Chapter 4.

Table 3–2. M or Miscellaneous Functions

Word (Code)	Explanation
M00	Automatically stops the machine. The operator must push a button in order to continue with the remainder of the program.
M01	Noted as an optional stop and acted upon only when the operator has previously signaled for this command by pushing a button. When the control system senses the M01 code via the tape reader, the machine will automatically stop.
M02	At the completion of the workpiece, this end of program code stops the machine after completion of all commands in the block. May include rewinding of tape.
M30	Noted as an end of tape command and goes slightly further than the M02 code in that this code will rewind the tape (assuming the control has this facility); and also automatically transfer to a second tape reader if incorporated in the control system.
M03	Start spindle rotation in a clockwise direction—looking out from the spindle face.
M04	Start spindle rotation in a counter-clockwise direction—looking out from the spindle face.
M05	Stop the spindle in a normal and efficient manner.
M06	Command to execute the change of a tool (or tools) manually or automatically, not to cover the selection of a tool as is capable with the T words.
M07	Code to turn a coolant on.
M08	Also a code for turning a coolant on. This may differ from M07 in that M07 may control flood coolant, and M08 mist coolant.
M09	Automatically shuts the coolant off.
M10 M11	M10 applies to the automatic clamping of the machine slides, workpiece, fixture spindle, etc. M11 is an unclamping code.
M12	An inhibiting code used for the synchronization of multiple sets of axes, such as a four-axis lathe having two independently operated heads (turrets).
M13	Combines clockwise spindle motion and coolant on in the same command which will cause both to occur simultaneously.
M14	Combines counter-clockwise spindle motion and coolant on in the same command.
M15 M16	Rapid traverse or feed motion in either the + (M15) or − (M16) direction.
M17 M18	Unassigned.

Table 3–2. M or Miscellaneous Functions (Continued)

Word (Code)	Explanation
M19	Oriented spindle stop. Causes the spindle to stop at a predetermined angular position.
M20 through *M29*	Unassigned.
M31	A command known as *interlock bypass* for temporarily circumventing a normally provided interlock.
M32 through *M39*	Unassigned.
M40 through *M46*	Used to signal gear changes if required at the machine; otherwise, unassigned.
M47	Continues program execution from the start of the program unless inhibited by an interlock signal.
M48	Cancel M49.
M49	A function which deactivates a manual spindle or feed override and returns the parameter to the programmed value.
M50 through *M57*	Unassigned.
M58	Cancel M59.
M59	A function which holds the rpm constant at its value when M59 is initiated.
M60 through *M99*	Unassigned.

Table 3–3. Other Address Characters

Address Character	Explanation
A	Angular dimension about the *X* axis. See Fig. 3–12.
B	Angular dimension about the *Y* axis. See Fig. 3–12.
C	Angular dimension about the *Z* axis. See Fig. 3–13.
D	Can be used either to express an angular dimension around a special axis (see Fig. 3–12), a third feed function, or a tool function for the selection

Table 3–3. Other Address Characters (Continued)

Address Character	Explanation
	of tool offset. Since it is recommended that only one type word be expressed in any one block, it may be desirable, on occasion, to move combinations of different axes at different feed rates for each axis.
E	This address also may be used for expressing an angular dimension around a special axis or a second feed function.
H	Unassigned.
I *J* *K*	Used with circular interpolation. (See Chapter 4.)
L	Not used.
O	Used in place of the customary "Sequence Number" word address *N* for a secondary head.
P	A third rapid traverse code or tertiary motion dimension parallel to the *X* axis.

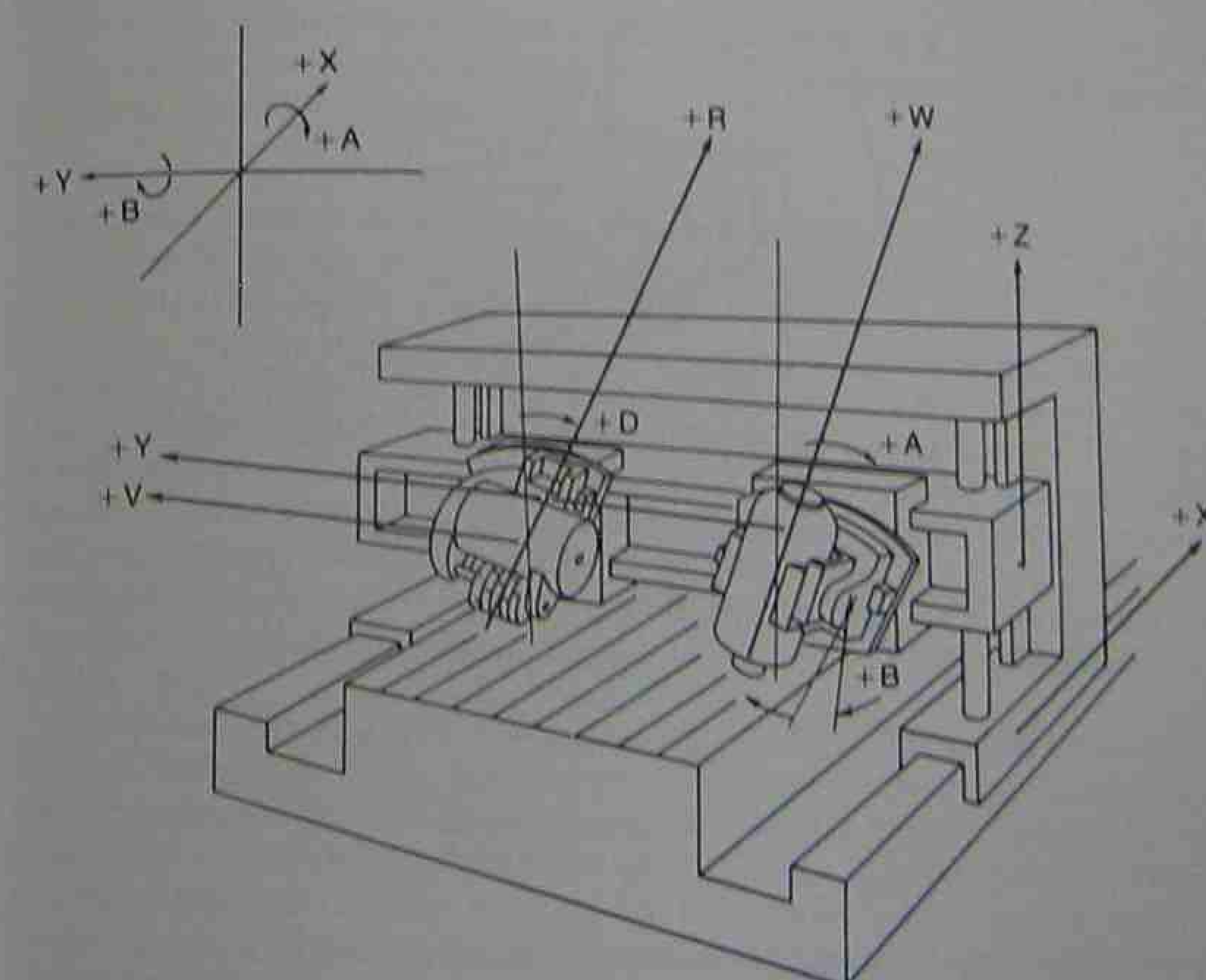


Fig. 3–12. The dual headed milling machine shown above requires a number of axes designations over and above the normal X, Y and Z motions. Twist motions of the heads are designated as *A*, *B*, and *D* while the cross motions of each head require separate identities, namely, *Y* and *V*.

Table 3–3. Other Address Characters (Continued)

Address Character	Explanation
Q	Second rapid traverse code or tertiary motion dimension parallel to the *Y* axis.
R	First rapid traverse code or tertiary motion dimension parallel (unless set at an angle, see Fig. 3–12) to the *Z* axis or the radius for constant surface speed calculation.
U	Secondary motion dimension parallel to the *X* axis.
V	Secondary motion dimension parallel to the *Y* axis. Figure 3–12 notes the cross motion of the left hand head as "*V*" which is a parallel motion to the cross motion of the right hand head noted as *Y* motion.
W	Secondary motion dimension parallel (unless set at an angle, see Fig. 3–12) to the *Z* axis.
end of record	Used to *stop* rewind motion and generally represented by the character percent.

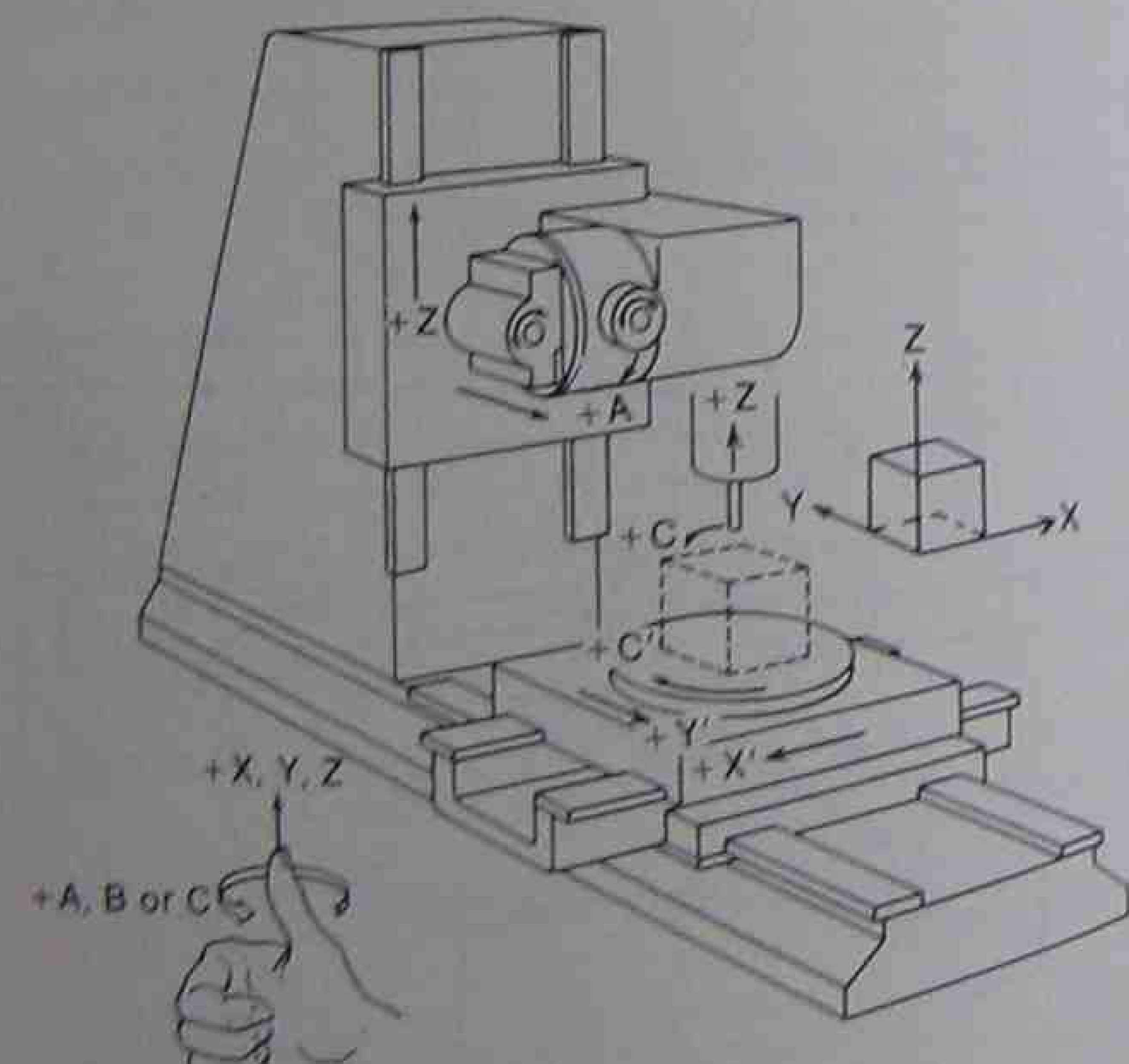


Fig. 3–13. The C′ motion is the rotary motion of the table about the Z axis while the C motion of the head is the circular traversing of the cutting head about the Z axis. Thus, A + C′ motion is equivalent to A + C motion and vice versa. According to the right hand coordinate system rule (see inset) a C′ motion is in a negative direction. This however is equivalent to a movement of the cutting head in a positive direction and is so used to indicate a positive cutting motion on the part.

per minute and as a four-digit word, although many manufacturers use a two- or four-digit coded word which refers to "steps" over the speed range. An alternative to the rpm coding is surface feet (meters) per minute (sfm). This capability is particularly useful with lathes wherein the spindle speed is automatically adjusted to maintain a constant sfm as called for by the S word. With most CNC systems this is controlled as a function of the distance the cutter is from the centerline of the spindle.

The *T* or *tool* word calls out the code number of the cutting tool to be automatically selected by the tool changer. If the code number and the tool offset are both designated in the T word, then the tool offset digits shall follow the associated tool code digits. When tool offset is selected by a different word, the D address is recommended. It is also recommended that the D word immediately follow the T word.

The *M* or *miscellaneous* function comprising two digits is contained in the latter part of the block and is assigned to auxiliary functions which do not relate to dimensional movements of the machine. One of the common auxiliary functions would be to turn the coolant on or off. The M functions (or words) that have been standardized are shown in Table 3–2.

Other Letter Address Characters In addition to the N, G, X, Y, Z, F, S, T, and M words that have been discussed, other letters have been assigned specific functions as shown in Table 3–3.

Other Tape Formats In addition to the character arrangement shown in Fig. 3–7 which describes a *word address* format, since each word (or set of numeric characters) is headed by a letter address (X, Y, Z, etc.), another format, which had been popular with hardwire point-to-point control systems, is *tab sequential*. This format is illustrated by the block shown in Fig. 3–14. In this case, a "TAB" character was inserted between the words, and the letter addresses may or may not have been shown. The advantage of this method is that the printout which may be obtained on a typewriter type tape preparation unit (e.g., Teletype) will be displayed in even rows. The present standard considers only the word address format. If a columnized listing is desired, tab characters may be inserted between words but will be ignored by the control system.

In addition to the standardizations described thus far the Electronics Industries Association's recommendations include a relatively simple shorthand code arrangement for describing characteristics of a control system that has been found extremely helpful to part programmers, machine tool and system engineers, and procurement personnel at user companies. In accordance with the standards, this description is known as a "*Format Classification Sheet*" and includes the following:

I. *Type of machine* (e.g., Vertical Milling Machine, Horizontal Jig Borer, or Vertical Turret Lathe.)
II. *Format Classification Shorthand* (refer to Fig. 3–15) noted by a letter

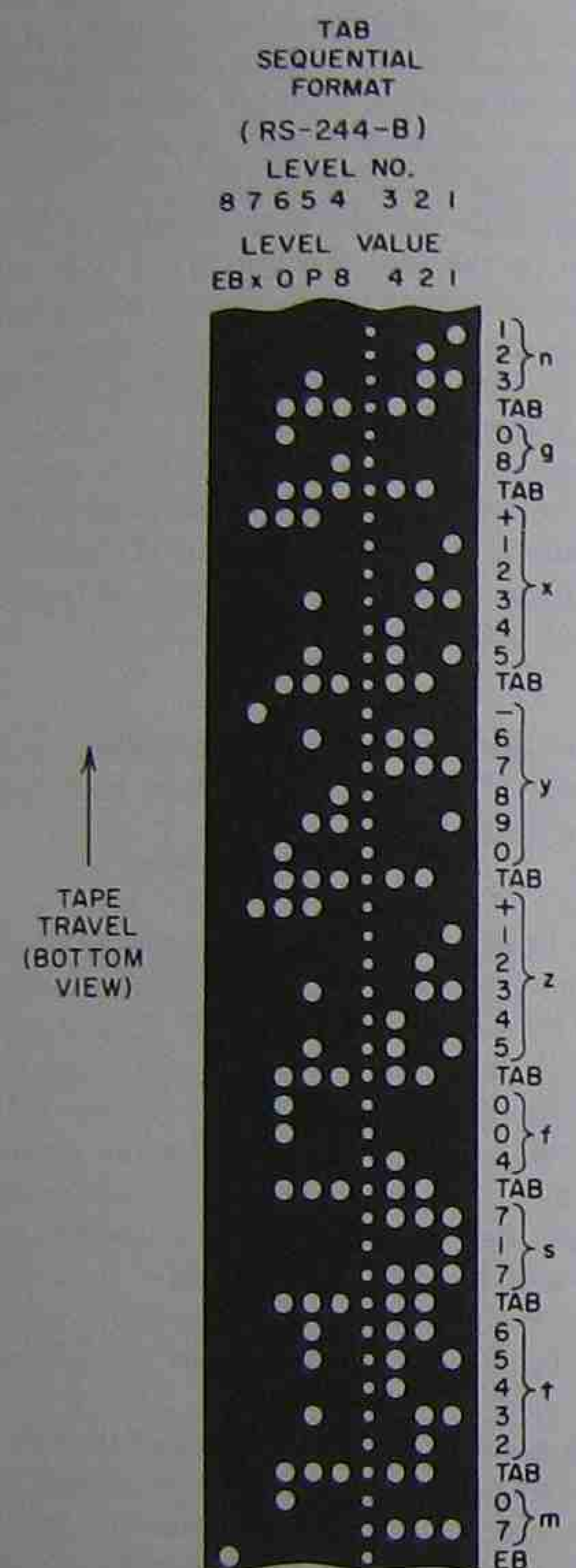


Fig. 3–14. The *Tab Sequential* format shown was popular with hardwire systems. The present standard considers only the word address format. The tab characters may be used to columnize the listing, which is typed as the characters are being punched, but will be ignored by the control system. The letter addresses, in any case, would have to be shown, according to present standards.

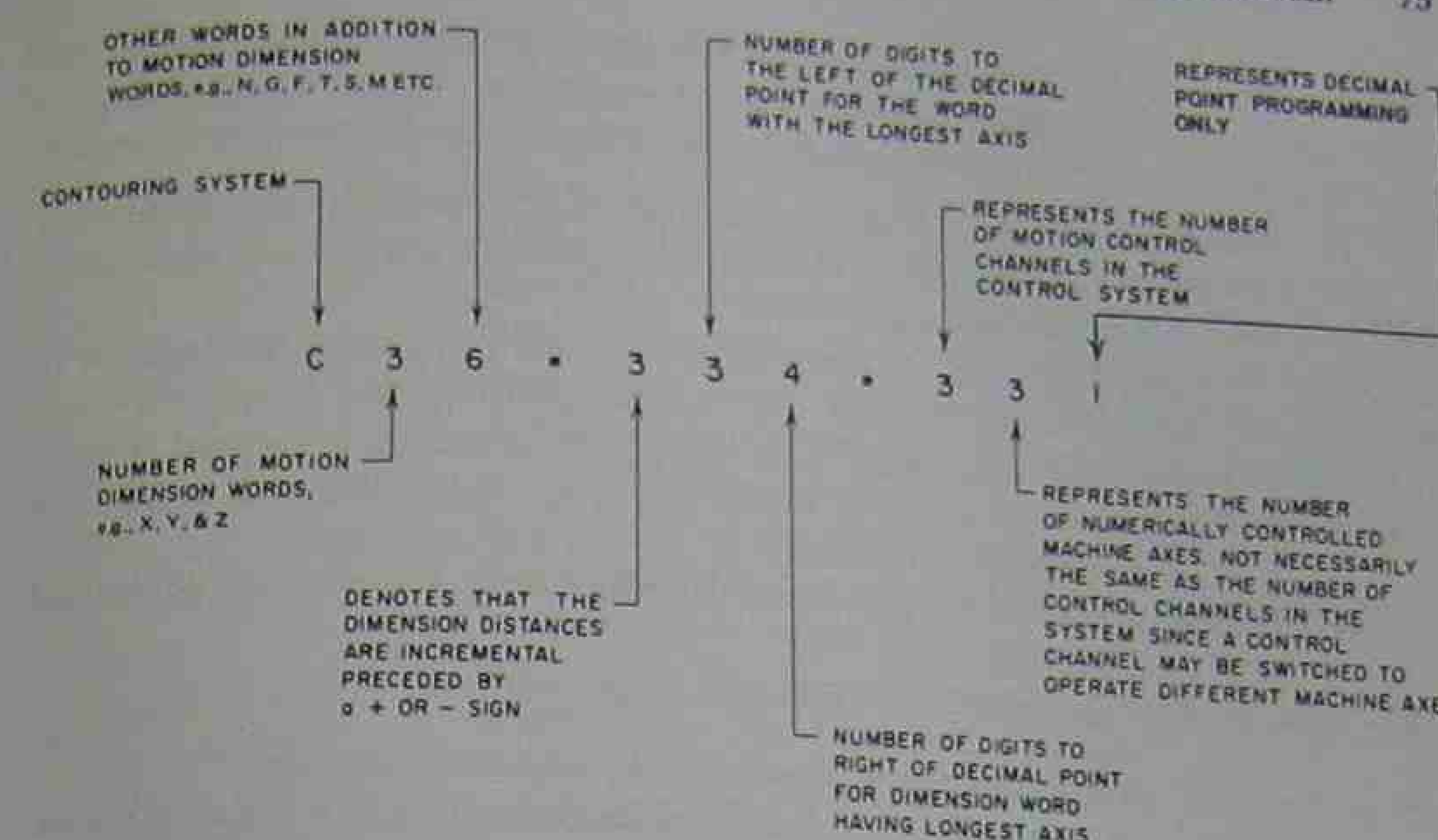


Fig. 3–15. The shorthand classification number shown above may be used to describe a numerical control system. Another set of letters and numerals which describe the system in greater detail is shown in the text.

followed by a series of numerals. The letter designation may be one of the following:

P—for variable block format positioning system
C—for variable block format contouring system
D—for variable block format dual system, i.e., contouring/positioning

The numerical designations which follow the first letter designation are as follows:

1. A digit representing the number of motion dimension words.
2. A digit representing the number of other words in addition to the motion dimension words. This digit is to be followed by a decimal point. If there are more than nine words and a second digit is required, subsequent characters will be moved ahead, i.e., the fourth character notation will then become the fifth character.
3. A digit representing the type of decimal data required by the system as follows:

   1 Represents coordinate dimensions, no algebraic signs
   2 Represents coordinate dimensions; + and – signs used
   3 Represents incremental dimensions; + and – signs used
   4 Represents coordinate or incremental dimensions depending on mode of operation; + and – signs to be used with incremental dimensions for contouring mode

5 Represents coordinate or incremental dimensions, depending on mode of operation; + and − signs used in both cases. Incremental dimensions for contouring mode.

4. The next and fourth numerical digit represents the number of digits to the left of the decimal point in the normal dimension word for the longest axis.
5. The next digit of the shorthand code represents the number of digits to the right of the decimal point in the dimension word for the longest axis. This digit is followed by a decimal point.
6. The next and sixth numerical digit represents the number of motion control channels in the control system.
7. The next digit represents the number of numerically controlled machine axes.
8. The last character shall be a digit wherein:

   1 represents decimal point programming only
   2 represents the full field (both leading and trailing zeros)
   3 represents leading zeros required
   4 represents trailing zeros required
   5 represents combinations of the above.

III. The *Format Detail* which describes the precise words used together with their order on the tape offers further information regarding the control system. This is probably one of the most significant and compact pieces of information when considering a control system. Guidelines are as follows:

1. Each word's letter address shall be listed in proper order.
2. Each dimension word address shall be followed by two digits, preceded by a sign character where + and − characters are used. The first digit in the dimension word shall indicate the number of digits in the word which are to the left of the decimal point, and the second shall indicate the number of digits to the right of the decimal point.
3. Each additional word address shall be listed in its proper place in order of words. A single digit shall follow the address to indicate the number of digits in the word.

   As an example, the *Format Detail* describing the control system for a machine similar to that shown in Fig. 3–7 would be as follows:

   N4G2X+34Y+34Z+34F32S2T2M2*

   Using the corresponding outline:

   N4 Represents the sequence number—in this instance 1234.

   G2 Represents the preparatory function with two digits—in this case G90.

X+34 Represents the X dimension measured in absolute ± values; 34 indicates that there could be three digits to the left and four digits to the right of the decimal point. However, in this case, zeros leading the "1" may be dropped and therefore the value 12345 is shown on the tape rather than 0012345. This is called *leading zero suppression* and applies to all words.

Y+34 Represents the Y dimension measured in absolute ± values, in this instance, corresponding to the tape, −67890 inches (one digit to the left and four digits to the right of the decimal point).

Z+34 Represents the Z dimension. In this case represented by Z + 12345 in the tape example.

F32 Represents the feedrate number in direct ipm. The number could go up to 999.99 ipm.

S2 Represents a two-digit coded speed number. If the speed were to be noted directly in rpm, the format would most likely be S4.

T2 Represents a two-digit tool code number.

M2 Denotes the auxiliary function having two digits.

* Denotes the end of the block.

IV. A *listing* of the preparatory, miscellaneous, tool, and other functions describing their ranges and references to the EIA code assignments. Also to be noted are functions, with their codes, that do not have a specific EIA assignment. An example would be any code between G60 and G69.

V. Other qualities unique to the system such as interlocks, available range of speed and feed codes (e.g., F function range 1–300; S function range 60–2800), maximum allowable dimensions of each axis (e.g., X, 60 inches; Y, 24 inches; Z, 12 inches), and any other information such as horsepower, tool offsets, and cutter compensation.

QUESTIONS TO CHAPTER 3

1. According to RS-244-B, words are made up of characters. (True, False.)
2. According to RS-244-B, the "parity" channel helps to ensure an odd number of holes in any one character. (True, False.)
3. According to RS-358-B the number of holes across the tape should be even, odd. (Circle correct answer.)
4. System and machine tool builders are required to adhere strictly to EIA standards. (True, False.)
5. With a CNC system the F, or feedrate word, is most commonly described
   a. as a ratio of ipm feed divided by the distance moved
   or
   b. directly in ipm
   or
   c. by the formula of the "magic three."

6. The difference between the M00 word and the M30 word is that the M00 word stops the machine at the end of the program and rewinds the tape and the M30 word stops the machine so that the operator may change a tool or reorient a part and then continue with the machining of the part. (True, False.)
7. The word that identifies a block is called a __________ __________.
8. An M09 word automatically turns the coolant on. (True, False.)
9. An M38 word *may* signal a bell advising the operator it's time for lunch. (True, False.)
10. The most popular tape format for contouring systems is tab sequential. (True, False.)
11. The "Ultra Conservative" machine has the following format detail: N3X + 2.3*Y* + 2.3*
    a. The sequence number has __________ digits of readout.
    b. How many miscellaneous codes are there?
    c. How many axes of motion are there?
    d. Does the machine have a tool changer?
    e. The machine is of the vertical type. (True, False, Cannot tell.)
12. One of the significant differences between a character in the RS-244-B code and the RS-358-B code is that the RS-244-B standard has an __________ parity check, while the RS-358-B has an __________ parity check. (Insert *even* or *odd* in the blank spaces.)
13. A bit, in tape language, is the combination of holes in tracks 2 and 5. (True, False.)
14. The terms *level*, *channel*, and *track*, all have the same meaning and refer to a line of holes running along the length of the tape. (True, False.)
15. The word address letter for the velocity of the table of a numerically controlled machine is: __________
16. The following block was manually programmed by an inexperienced part programmer and typed by a less experienced typist. Circle the errors.

    N5463G193*X* + 10642*Y*9436*F*30*M*15.3*E*0*B*

17. Describe the difference between cutter offset and cutter compensation.

CHAPTER FOUR

Manual Part Programming

The term *Manual Part Programming* refers to the preparation of a numerical control part program without the assistance of a computer. *All* of the detailed instructions for operating the numerical control machine are listed, in precise order, on a form, which is called a *manuscript*. Calculations are usually prepared with the aid of pencil and paper, and a hand or desk calculator. The instructions and data on the manuscript are then transferred to a tape (see Chapter 3). The tape is usually prepared by a device similar to a typewriter. Each strike of a key produces a character (a line of holes across the tape). Examples of these tape preparation units are shown in Figs. 4–1 and 4–2. A description of manual part programming and a simple point-to-point part program are described in this chapter.

CARTESIAN COORDINATE SYSTEM—THE BASIS FOR NC PART PROGRAMMING

During the early development of numerical control it became apparent that conventional blueprint notations were neither sufficiently consistent nor did they offer the proper mathematically machine-related information to be successfully utilized with numerical control. Since most machine movements are generally at right angles to each other and it was required that movements be

Fig. 4–1. One of several types of manual tape preparation units.

calculated from some fixed reference, the logical recourse was to utilize the relatively familiar Cartesian coordinate system. Essentially, this system provides a convenient means of locating points from two to three fixed lines, or planes through these lines, which are positioned at right angles to each other. Each line is described as an *axis* and is designated by a capital letter. Figure 4–3 describes a point *A* located at a specific position with respect to the two axis lines, *X* and *Y*. If we were to consider this area as a grid pattern, then the location of the point could be described as 3 units to the right of the *Y* axis and in the *x* direction and 4 units up from the *X* axis and in the *y* direction. It may be seen, therefore, that the location of any point or a series of points might be conveniently described in this manner. The location of points may

Courtesy of Numeridex

Fig. 4–2. The tape preparation system shown can edit programs in addition to preparing tapes. It can also be tied into a remote computer part programming service (see Chapter 5).

also be described in tabular form, e.g. (again referring to Fig. 4–3):

Point	x Dimension	y Dimension
A	3	4
B	5	6
C	6	7
D	7	6
E	6	5

The dimension may be expressed in inches, feet, miles, or any other distance expression. In the U.S. and some other English-speaking countries, the inch is common. The metric system is standard with most European and Scandinavian countries and is becoming more prevalent in the U.S. and other English-speaking countries.

In addition to describing the position of point *A* as it rests on a plane surface, it is also possible to describe the point with respect to its axis relationship in space. This may be accomplished by adding another line, or axis, designated as *Z*, which is perpendicular to the surface that is formed by the *X* and *Y* axes. (See Fig. 4–4). Three coordinate planes are then formed by the three intersecting axes, viz: *XY*; *XZ*; and *YZ*. The coordinate location of *A'* may therefore be stated as 3 units in the *X* direction; 4 units in the *Y* direction and

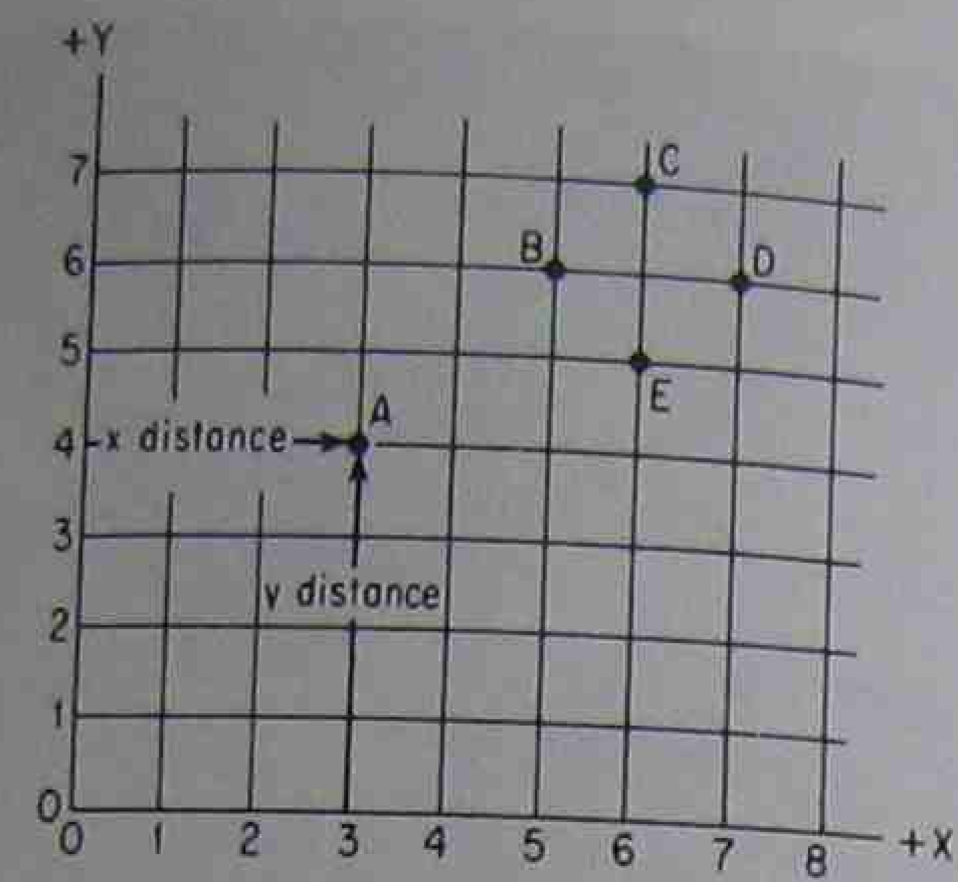


Fig. 4–3. The coordinate system shown is utilized to express the position of points with respect to X and Y lines (axes). In the example shown, point A is three divisions to the right of line Y and four divisions up from line X.

1 unit in the Z direction. The tabulation describing the location of point A′ would appear as follows:

Point	x	y	z
A′	3	4	1

For convenience, the dimensions shown in Figs. 4–3 and 4–4 are all positive; however, the Cartesian coordinate system also provides for negative dimensions. Figure 4–5 describes x dimensions to the left of the vertical Y axis

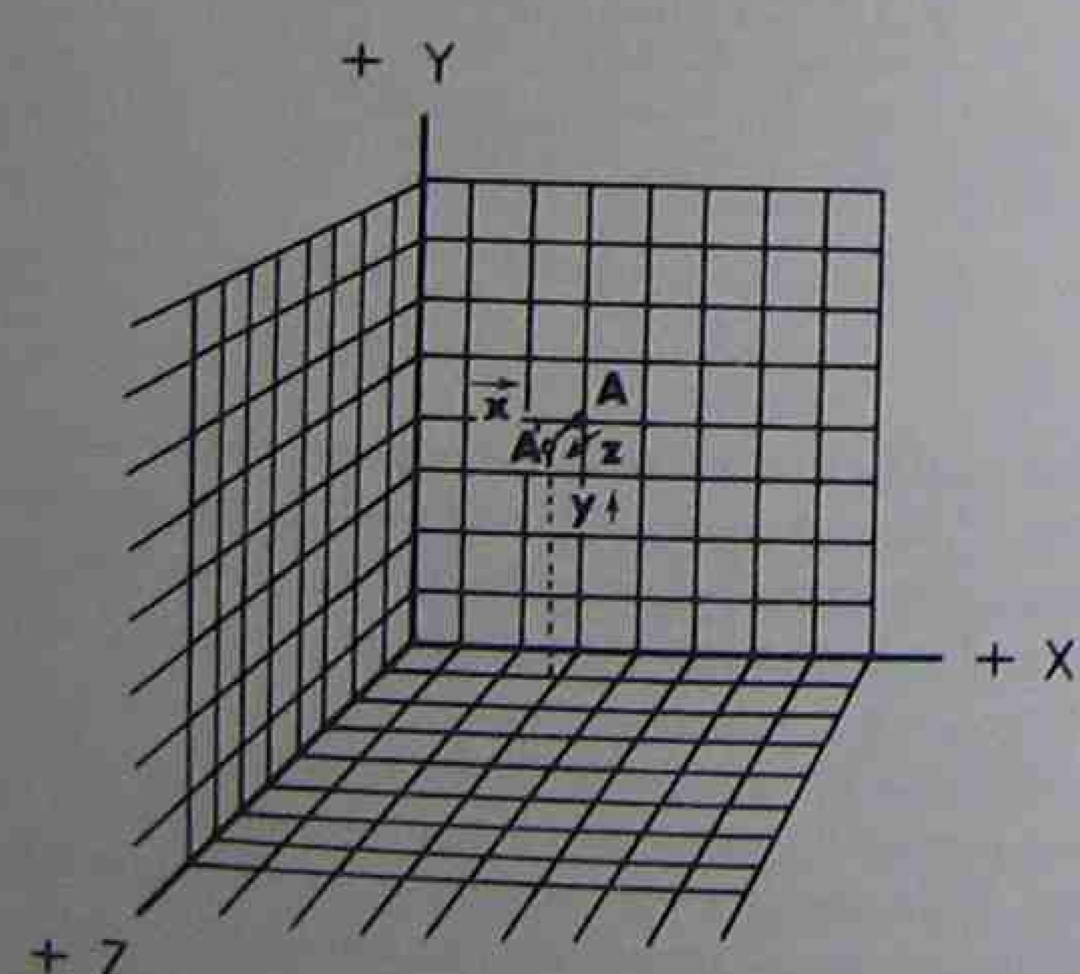


Fig. 4–4. In this illustration point A′ is located in space with its position described as a perpendicular distance from each of three planes formed by the X, Y and Z axes.

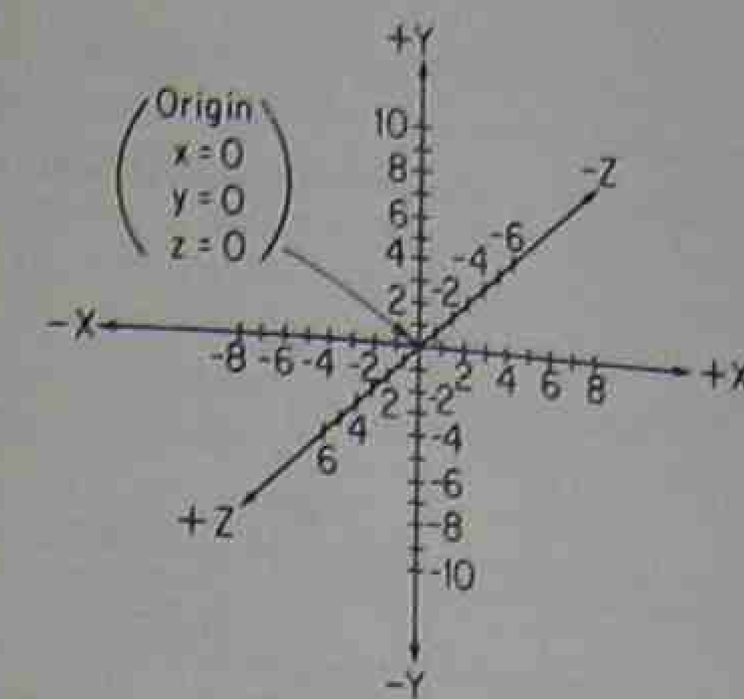
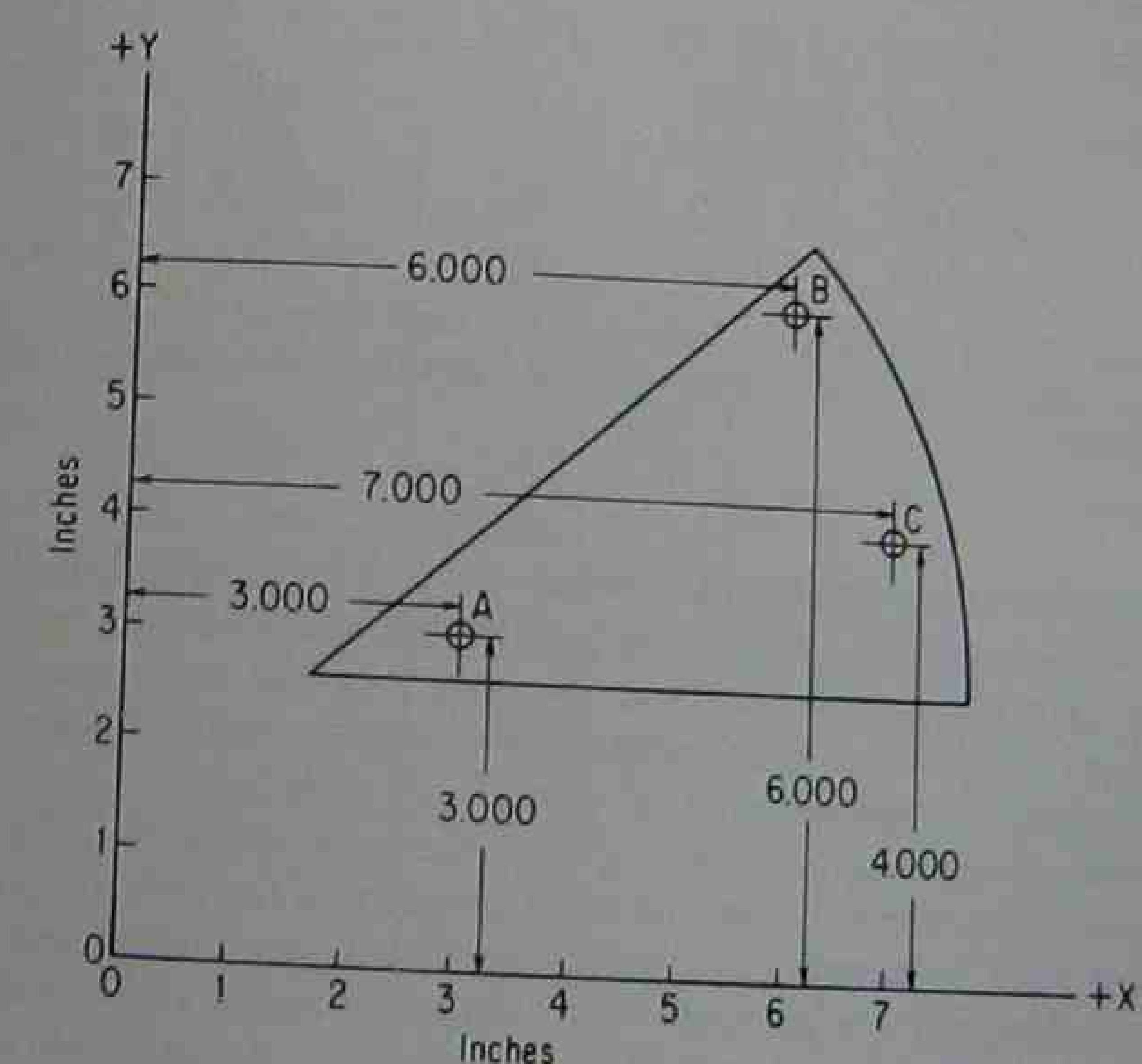
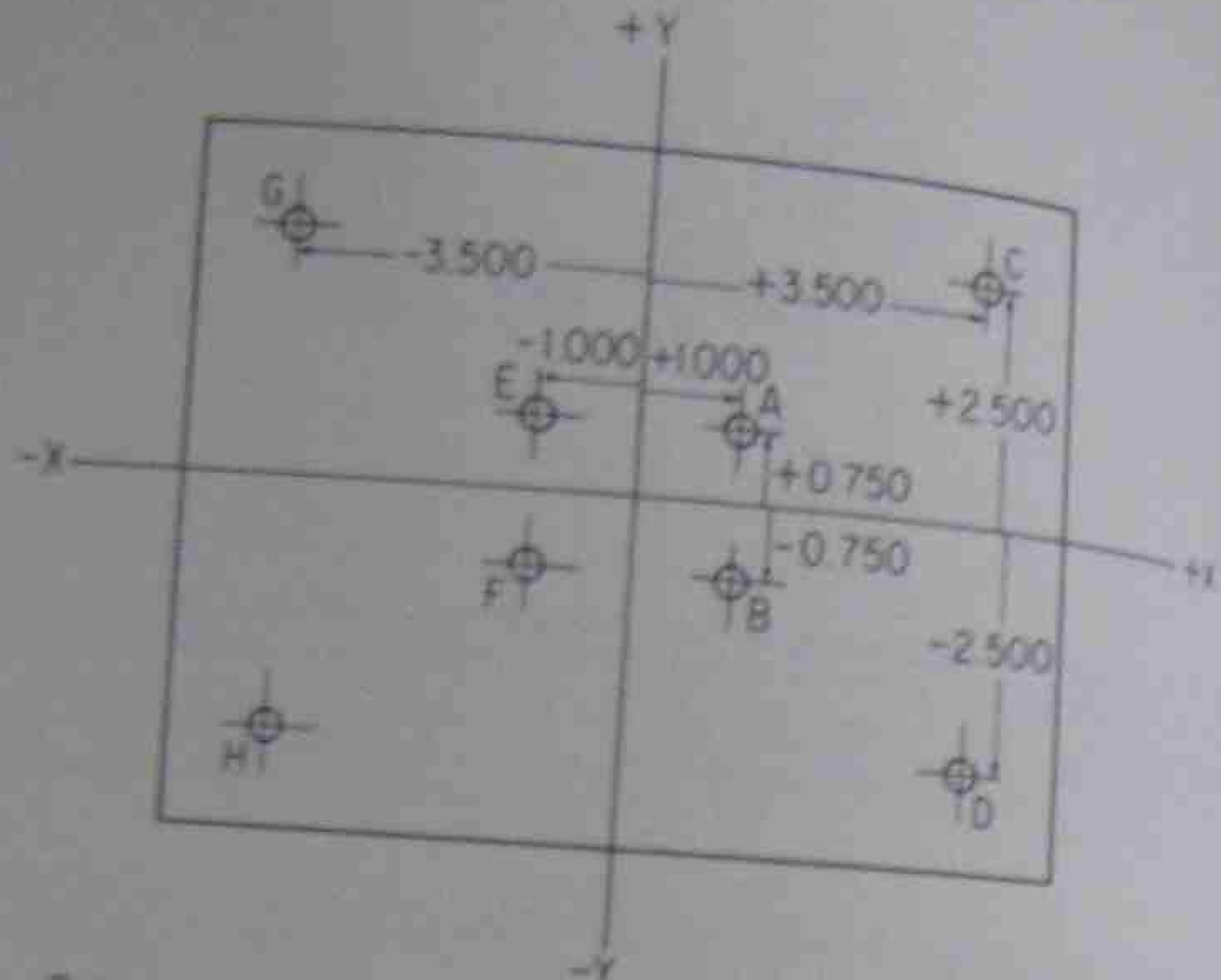


Fig. 4–5. Negative x values would hold for points to the left of the Y axis and negative y values would hold for points below the X axis and negative z values would hold for points to the rear of the X axis in the −Z direction.



Point	x	y
A	+3.000	+3.000
B	+6.000	+6.000
C	+7.000	+4.000

Fig. 4–6. The locations of the hole centers are shown as distances from the X and Y axes. The locations of the hole centers (points) may also be described in tabular form as shown.



Point	x	y	Point	x	y
A	+1.000	+0.750	E	-1.000	+0.750
B	+1.000	-0.750	F	-1.000	-0.750
C	+3.500	+2.500	G	-3.500	+2.500
D	+3.500	-2.500	H	-3.500	-2.500

Fig. 4-7. Parts having symmetrical hole patterns lend themselves well to numerical control machining and may best be programmed by positioning the origin at some central point on the part.

as being negative and y dimensions below the X axis as also being negative. Negative Z axis dimensions would lie beyond the plane formed by the X and Y axes. For ease of programming and in order to avoid error, most two-axis programming is performed in the upper right hand portion of the coordinate system since here all directions (X, Y, and Z) are positive.[1]

In order to describe the machining operations for a part in a form usable for preparing a tape, it is necessary that the distance of the specific points from the reference axes be noted. With point-to-point programming, the points to be described (located) are generally the centers of holes and the coordinate points of those centers are described in tabular form. An example showing a hole pattern and its associated tabular description is shown in Fig. 4-6.

[1] The sections are also referred to as quadrants. Considering only the X and Y axes, in Fig. 4-5, the upper right hand section is designated as the first quadrant; the upper left section as the second quadrant; the lower left as the third; and the lower right section as the fourth quadrant.

With parts having symmetry it may be more advantageous to locate the origin, or zero point, on the axis of symmetry of the part itself. The inconvenience of having to deal with negative values is outweighed by the reduction in arithmetic since identical numerical values may be used with only the sign having to be changed. Figure 4-7 shows an example of such a hole pattern. In this instance it was possible to describe the location of 8 holes with half as many different numerical figures as would be required if the part were wholly positioned within one quadrant.

The Manuscript By definition, the manuscript for manual programming is a

MANUSCRIPT
PTP
XXXX MACHINE

Part No. 1243
Part Name: Plate

Date: x/x/xx
Prep By: ABC
Checked By: DEF

Sequence No. (N)	X		Y		Spindle Speed (S)	Aux. Code (M)	Comments
001	+	1.000	+	.750	630	M13	Use ½" HSS
002	+	1.000	-	.750			Drill, set
003	+	3.500	+	2.500			depth @ 0.25"
004	+	3.500	-	2.500			
005	-	1.000	+	.750			
006	-	1.000	-	.750			
007	-	3.500	+	2.500			
008	-	3.500	-	2.500		M30	Tape will
							automatically
							rewind after
							last operation.

Fig. 4-6. Detailed, step-by-step machining instructions are listed on the "manuscript" shown above. The case shown refers to a relatively simple single-spindle drill press. The instructions shown refer to the pattern illustrated in Fig. 4-7. The hole configuration on the tape for a portion of the above manuscript would be as shown in Fig. 4-9. While generally accepted machining practice might require that spot drilling precede the drilling instructions here shown, this has been omitted for the sake of brevity.

planning chart or list of instructions which describes the detailed and precise step-by-step operation of a machine tool under numerical control. The information must be in such a form that it can be transferred directly to tape which requires that *complete* and *accurate* instructions be provided. An example of a simple manuscript form for point-to-point manual programming is shown in Fig. 4–8. The "write-in" figures refer to the pattern described in Fig. 4–7. Each horizontal line comprises a "block" of information, as explained in Chapter 3. The feed of the depth motion may be controlled as a third axis motion having full servo-control as with motions in the direction of the *X* and *Y* axes or it may be manually set by the operator.

Auxiliary or *M* codes may be peculiar to a particular machine or control system but where possible, are generally standard. The spindle speed and depth feed will remain unchanged unless changed by the operator or by a code on the tape. The "*M13*" code noted in the "auxiliary code" column is the standard

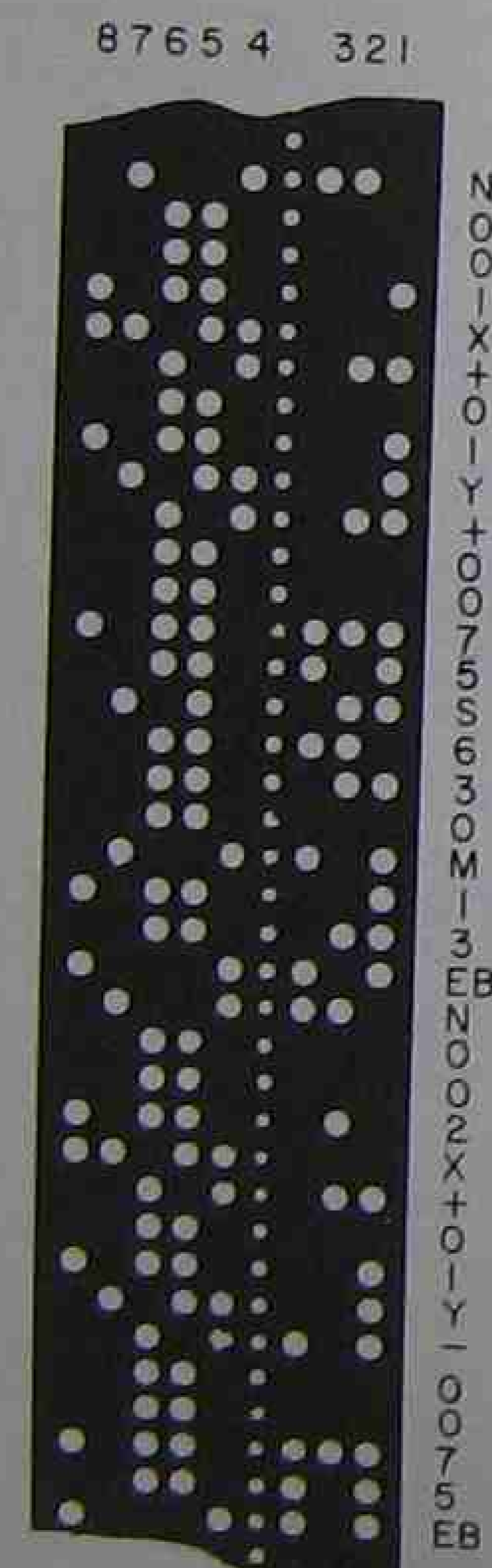


Fig. 4–9. Coded tape instructions for describing a portion of the motions noted on the manuscript in Fig. 4–8 and shown by the drill pattern in Fig. 4–7. The character coding shown conforms with RS-358-B.

EIA instruction for turning on the coolant and starting clockwise spindle rotation, the "*M30*" code is the EIA standard instruction for stopping the operation and automatically rewinding the tape back to the start of the first block. Figure 4–9 is a display of a portion of the tape which would be processed from the manuscript shown in Fig. 4–8. As with a number of control systems, zeros which follow significant figures need not be described on the tape, thus conserving tape, tape reading time, and tape preparation time. This feature is called "*trailing zero suppression*." Zeros preceding the significant figures, however, must be shown in this instance.

POINT-TO-POINT VS. CONTOUR PART PROGRAMMING

The part programming examples described in Figs. 4–6 and 4–7 involve point-to-point operations, in these cases the drilling of holes. The drill is directed to move to a spot, designated by its *X* and *Y* coordinates, and then to perform an operation at that point, such as the drilling of a hole. The machine is then instructed to move on to the next point and drill another hole, etc. This repetitive cycle continues until all of the holes are drilled. Point-to-point operations may also involve boring, reaming, tapping, or punching holes with a punch press. In moving from one point to another the path that the tool takes is not terribly important since the tool is moving in air, as long as it does not take too much time in moving about. (See Fig. 1–12.) The part programmer need only calculate the coordinates of the holes to be drilled and, if the machine has Z-axis control, the depths to which the holes are to be drilled. If, therefore, there are not too many holes involved, a point-to-point part may be programmed manually. If, however, a goodly number of holes are involved, the mere task of preparing the tape and using a tape-punch typewriter can be quite a chore, especially for an inexperienced typist, which many part programmers are. In such cases computer-assisted part programming may be in order. (See Chapter 5.)

Contour part programming, on the other hand, usually requires a good deal of calculation and is generally not nearly as well suited for manual part programming as with many point-to-point operations. As described in Chapter 1, continuous-path or contouring numerical control requires that two or more motions of a machine be simultaneously and precisely coordinated. Numerical control applications which fall into this category include machining centers, lathes, drafting machines, seam welders, profile millers, and torching equipment. Because of the necessity to control the resultant-path motions of the machine elements, programming requirements are generally more complex and, more significantly, far more detailed and time-consuming to prepare than point-to-point programs. With contour milling, each incremental movement must be "described."

Although practically all lathes, and most machining centers and mills now being manufactured, have the capability to move along a straight line or a *circular* arc with one block of information, many older hardwire NC machines required that the circular arc be broken into short straight-line increments or segments. Even with the newer systems arcs that are not circular must be

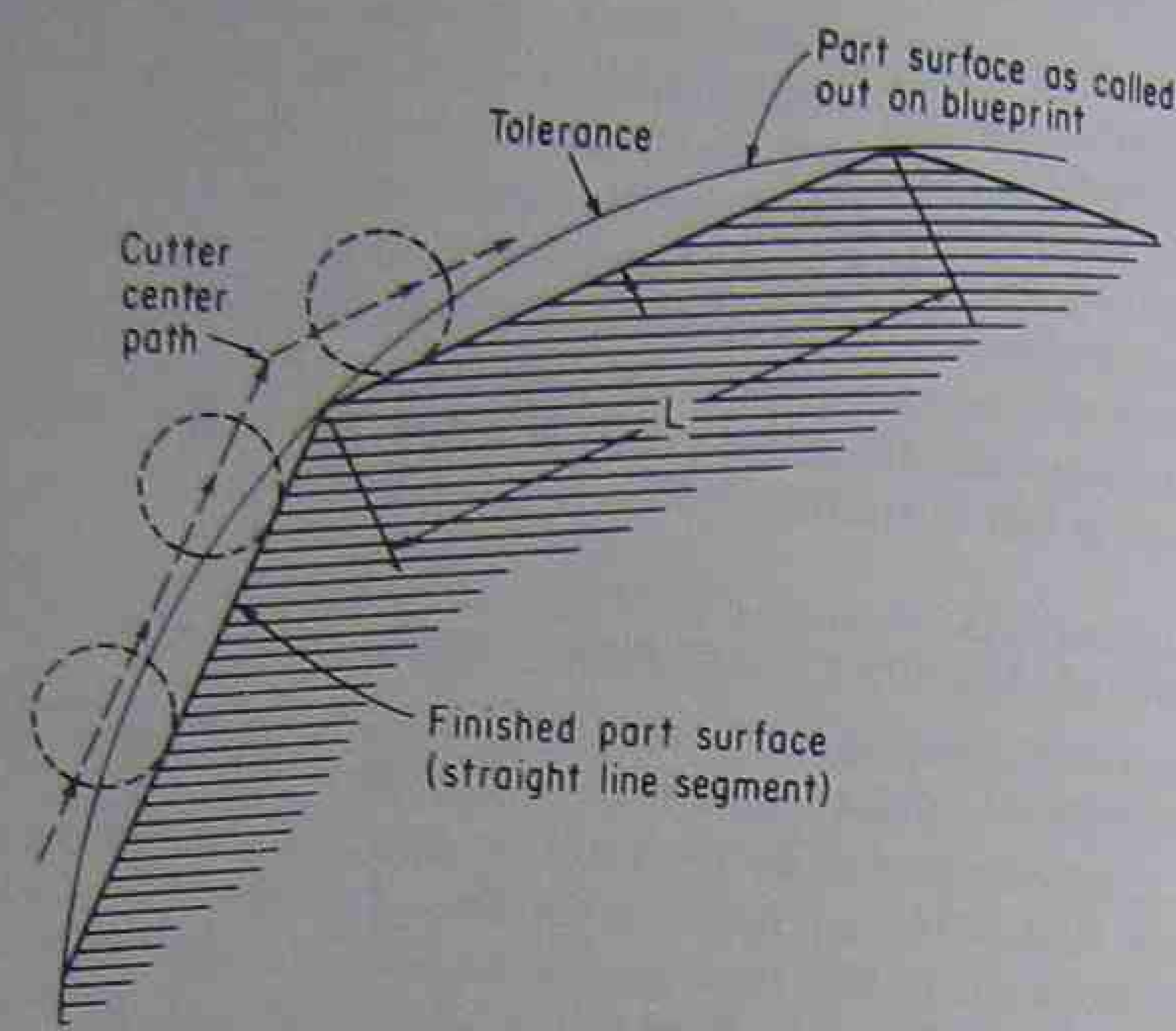


Fig. 4–10. Older contour machines move only in straight-line increments along circular arcs as here shown in exaggerated form. The straight-line cuts are calculated to be sufficiently short so that the maximum distance between the straight line and its related arc does not exceed the tolerance requirement. Two other methods of straight-line approximations of curves are shown in Fig. 4–11.

broken down into straight-line segments.[2] Figure 4–10 shows, in exaggerated form, a portion of an arc that has been broken into two straight-line segments. The part surface, as cut, will actually consist of a number of straight-line cuts that approximate, within tolerance, the curved surface. The length and number of these lines or chords will depend on the tolerance required on the part together with the extent of curvature of the arc. To illustrate, the length of the straight-line segment "L" shown in Fig. 4–10 is dependent on the finished part tolerance required; the broader the tolerance, the longer "L" may be and vice versa. The break-up of a curve into these straight-line segments is one of the major time-consuming requirements in contour programming. Note that in Fig. 4–10 the straight lines lie wholly within the arc. Figure 4–11 illustrates two other approaches to approximating a curve with straight-line segments, i.e., either positioning chords outside the arc or averaging the straight-line chord segments through the curve. Approximation of an arc by one of these straight line methods can be quite accurate, since in most instances the length of each chord can be made as small as .0002 inch.

A second significant calculation required with contouring entails describing the path of the center of the cutter rather than actual part contour dimensions since the tape instructions express machine movements and not the configu-

[2] Some control systems can move along parabolic arcs with single block data; however, these systems are rare.

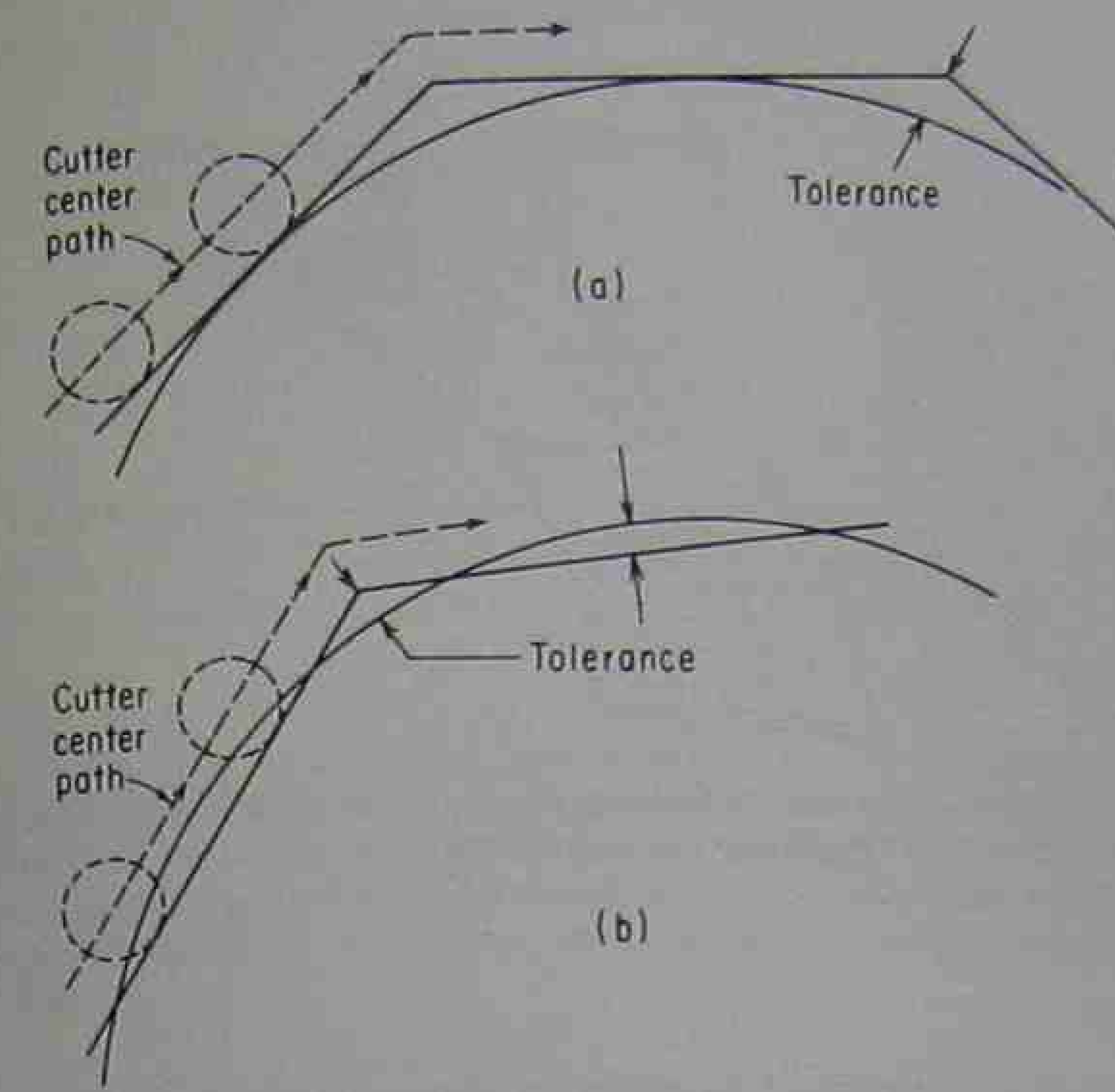


Fig. 4–11. In (a) the straight lines (exaggerated) are drawn as tangents to the curve and the finished part will be slightly oversize although within tolerance. Averaging the straight lines through the arc as shown in (b) is the preferred practice.

ration of the part. As an example, the calculated line of travel around the circumference of the part shown in Fig. 4–12 would be *offset* from the perimeter of the part by an amount equal to the radius of the cutter. The problem becomes more complicated as the number of "simultaneous movements" increases. Consider, for example, the required cutter offset calculations of a ball end mill that is traveling (from the standpoint of the X, Y, and Z axes) in three directions at the same time. A cut of this type may be necessary when machining a die surface or the contoured surface of an airframe structure. As an example in

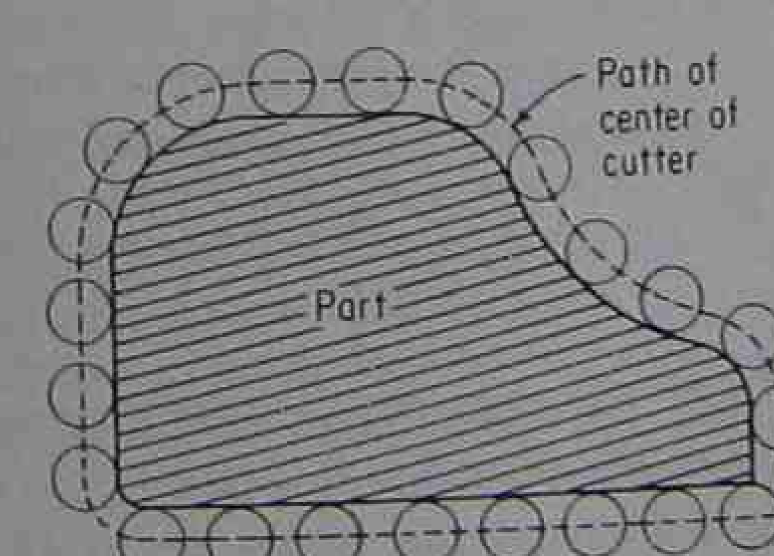


Fig. 4–12. The instructions noted on the tape would describe the path of the center of the cutter as shown by the dashed line around the perimeter of the part which is offset from the contour of the part by an amount equal to the cutter radius.

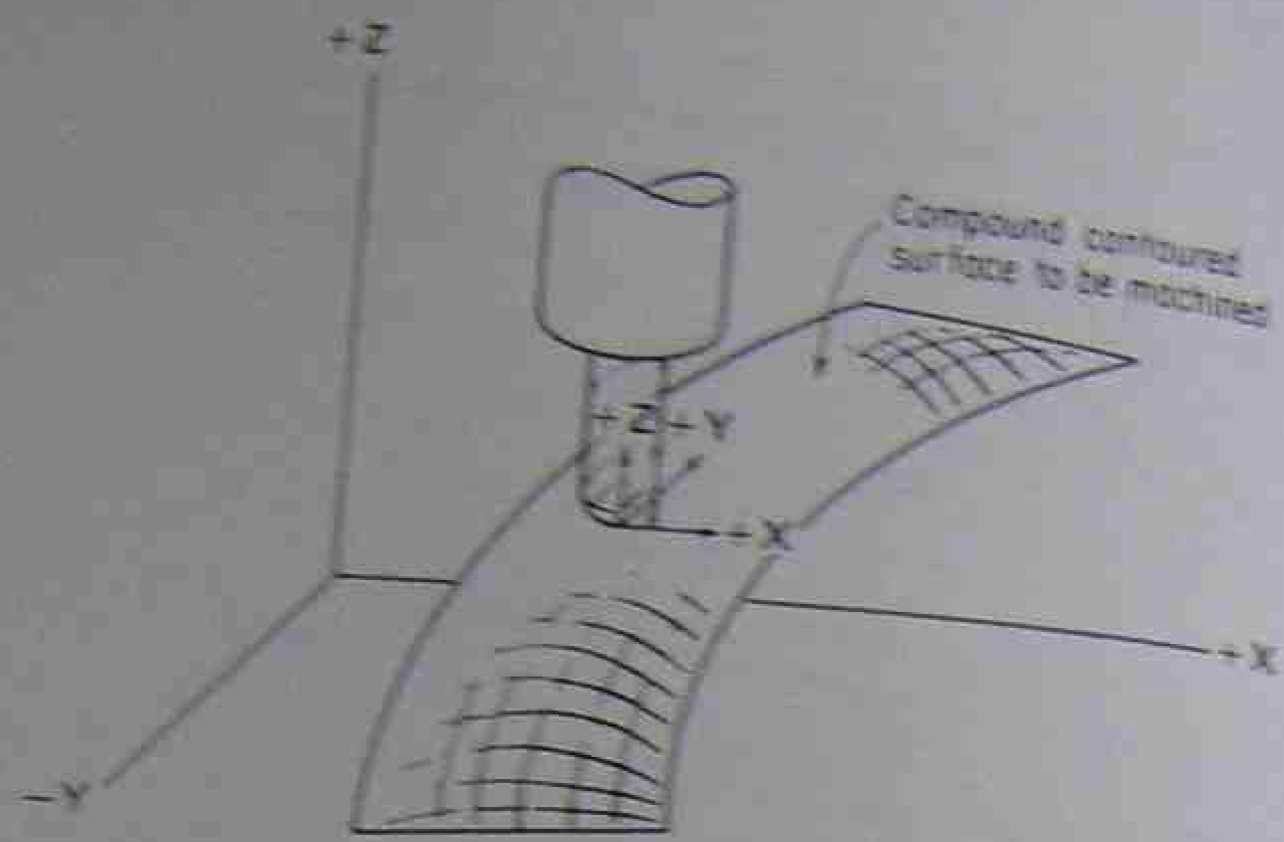


Fig. 4–13. With three-dimension contouring, the three straight-line components along the X, Y, and Z axes which determine the path of travel of the tip of the cutter must be calculated.

Fig. 4–13 the cutter is required to move in three directions simultaneously, i.e., its movement is the resultant of x, y, and z components. Thus, the x, y, and z components are calculated for various points along the line of travel of the tip of the cutter.

LINEAR INTERPOLATION

By definition, linear interpolation, a contouring feature, is the ability of the control system to move components of a machine along a resultant straight-line motion. Considering a two-axis move in the XY plane, this would mean moving along both the X and Y axes at the same time (see Fig. 4–14). If the cutter were to move at a 45° angle, the distance movement along the X axis would be the same as the distance movement along the Y axis, in the same time period. The lead screws for both axes would be timed to turn at the same rpm and the speed along each axis would be the same. If, as another consideration, the cutter were to move along a 30° angle, the distance to move along the X axis would have to be greater than that moved along the Y axis, in the same period of time (1.73 times as great). The X-axis movement would therefore have to be faster than the Y-axis movement (1.73 times as fast). The ability of the control system to monitor the pulses in the correct proportion so that the resultant move is along a correct straight line is considered linear interpolation.

Also, unlike point-to-point part programming in which the coordinates of the points are described (i.e., the distances from the X, Y, and Z axes), contour moves are usually programmed *incrementally*. That is, the distances the cutter moves from one point to the next are described. Referring to Fig. 4–15, if we wanted to move the cutter to points A, B, and C, such as with point-to-point programming, we would describe the coordinate locations of these points. All signs would be positive since all points are located in the first quadrant. If,

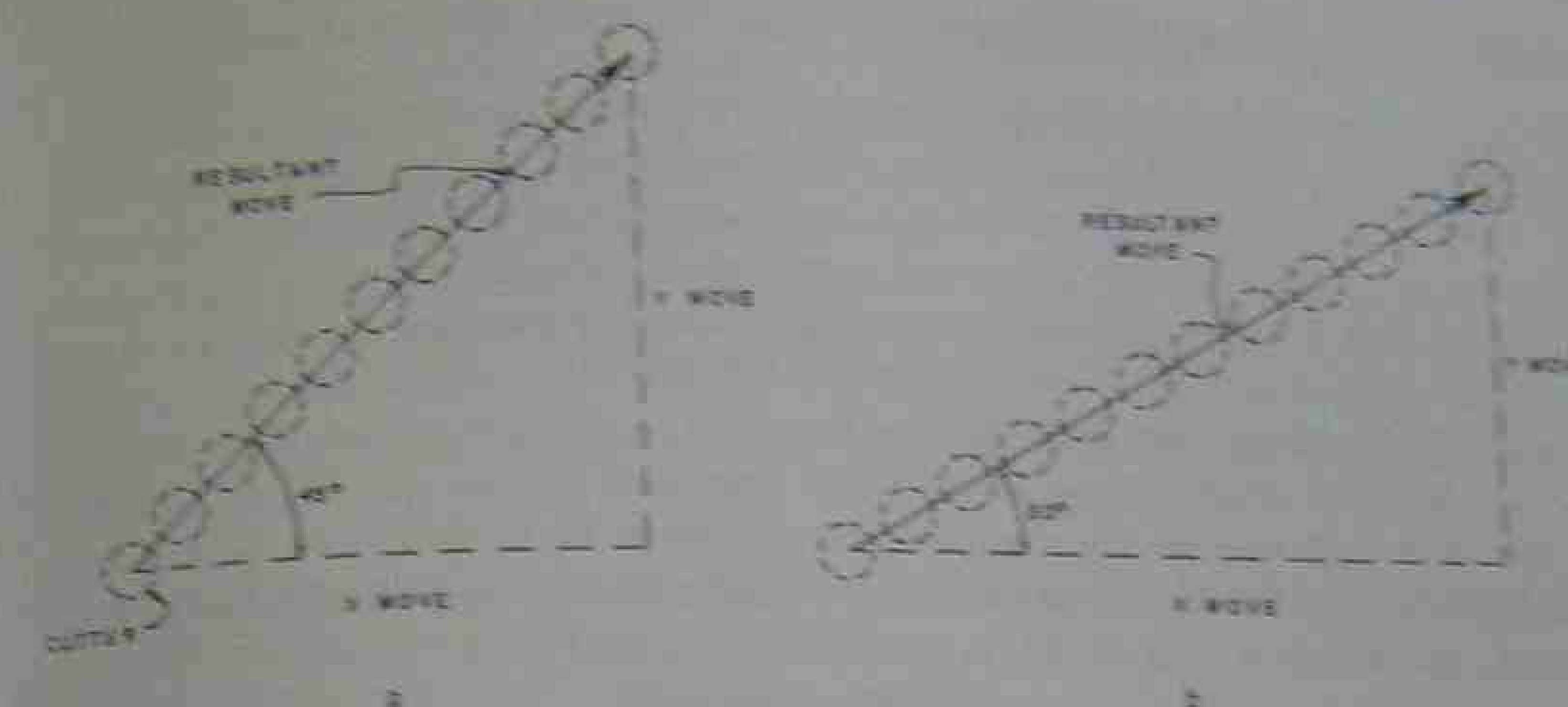


Fig. 4–14. The ability of a control system to move a cutter along a straight line by coordinating two or more axes movements is called linear interpolation. In *a* above the cutter is moved along a 45° angle line by making the X and Y movements equal. In *b* the X movement would have to be greater than the Y movement within the same time period. If the component X and Y movements are held steady, the resultant movement will be a straight line.

however, we wanted to move *along* line AB and then along line BC, in a contouring mode, we would probably note the X distance from A to B together with the Y distance from A to B. In this case both would be positive. In moving from B to C the X movement would be positive; however, the Y movement would be negative since the move is in a negative Y direction.

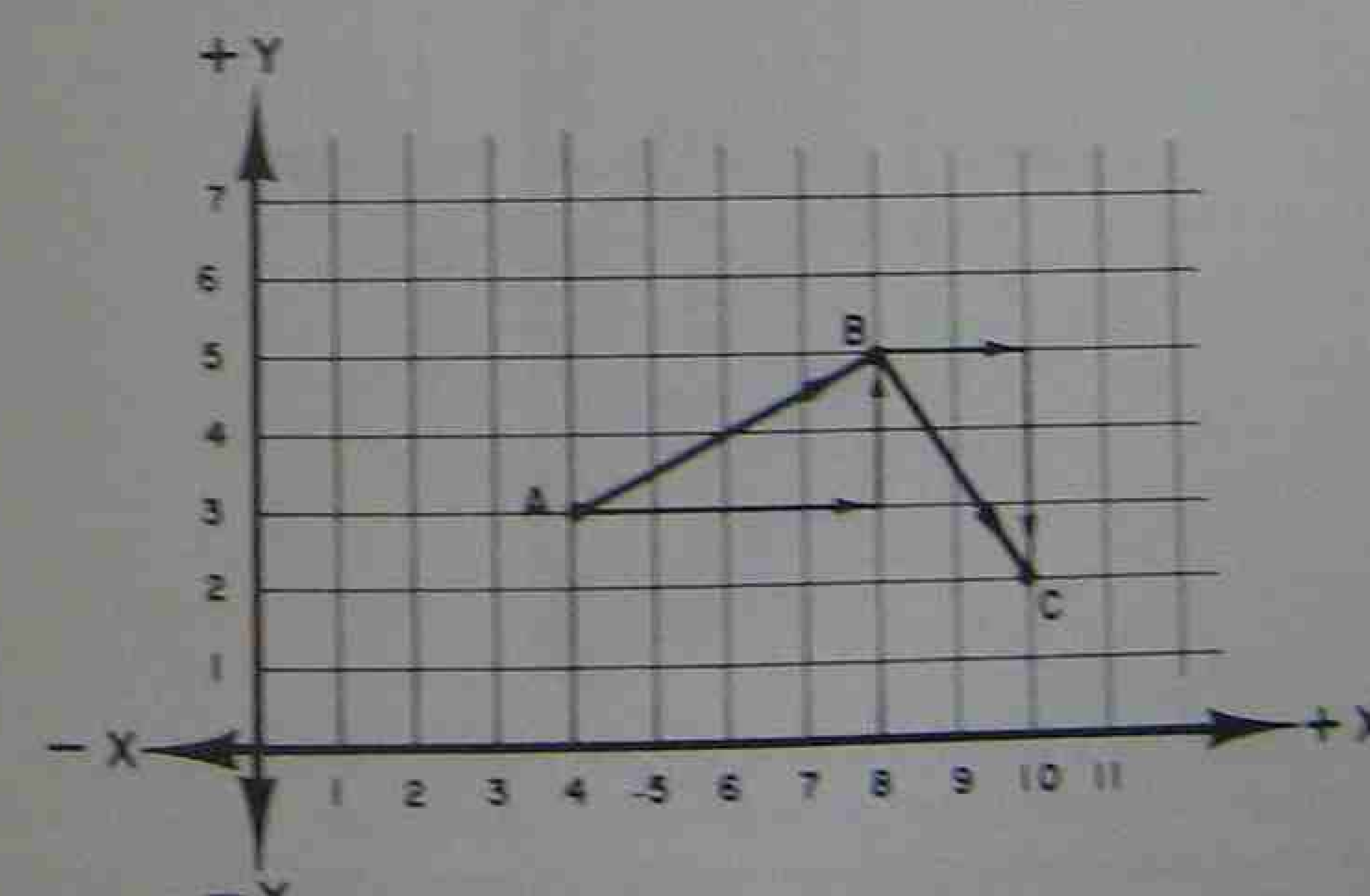


Fig. 4–15. In an absolute system the coordinates of points A, B, and C would be noted. In an incremental system the length of the movements from one point to the next, and parallel to the X and Y axes, would be noted.

CIRCULAR INTERPOLATION

Circular interpolation is a contouring feature that allows the part programmer to describe the movement around a *circular* arc in one block of data rather than by breaking the arc into short straight-line segments as would be required with linear interpolation. The arc is usually confined to the *XY* plane (plane formed by the *X* and *Y* axes) although some machining center and milling machine systems can perform this function in the *XZ* and *YZ* planes; and some of the newer CNC systems can perform this function in three axes simultaneously.

As with the general tape format that has been standardized (see Chapter 3) the tape instructions for circular interpolation have also been formalized. In programming, if it is desired to move a cutter along a circular arc from point *A* to *B*, as shown in Fig. 4–16, it would be required to:

1. Denote the *preparatory* code G02 which instructs the control system that the information following is to describe the path of a machine head in a *clockwise* and *circular* pattern instead of a linear interpolation mode. A G03 code would be noted if the cutter were to move in a counter-clockwise direction. This is necessary since any system incorporating circular interpolation will also include linear interpolation. In order to *switch* to a linear interpolation mode it is necessary that a G01 code be noted before the specific movement instructions changing the mode are inserted.
2. The *X* and *Y* movements which relate to the component lengths of the

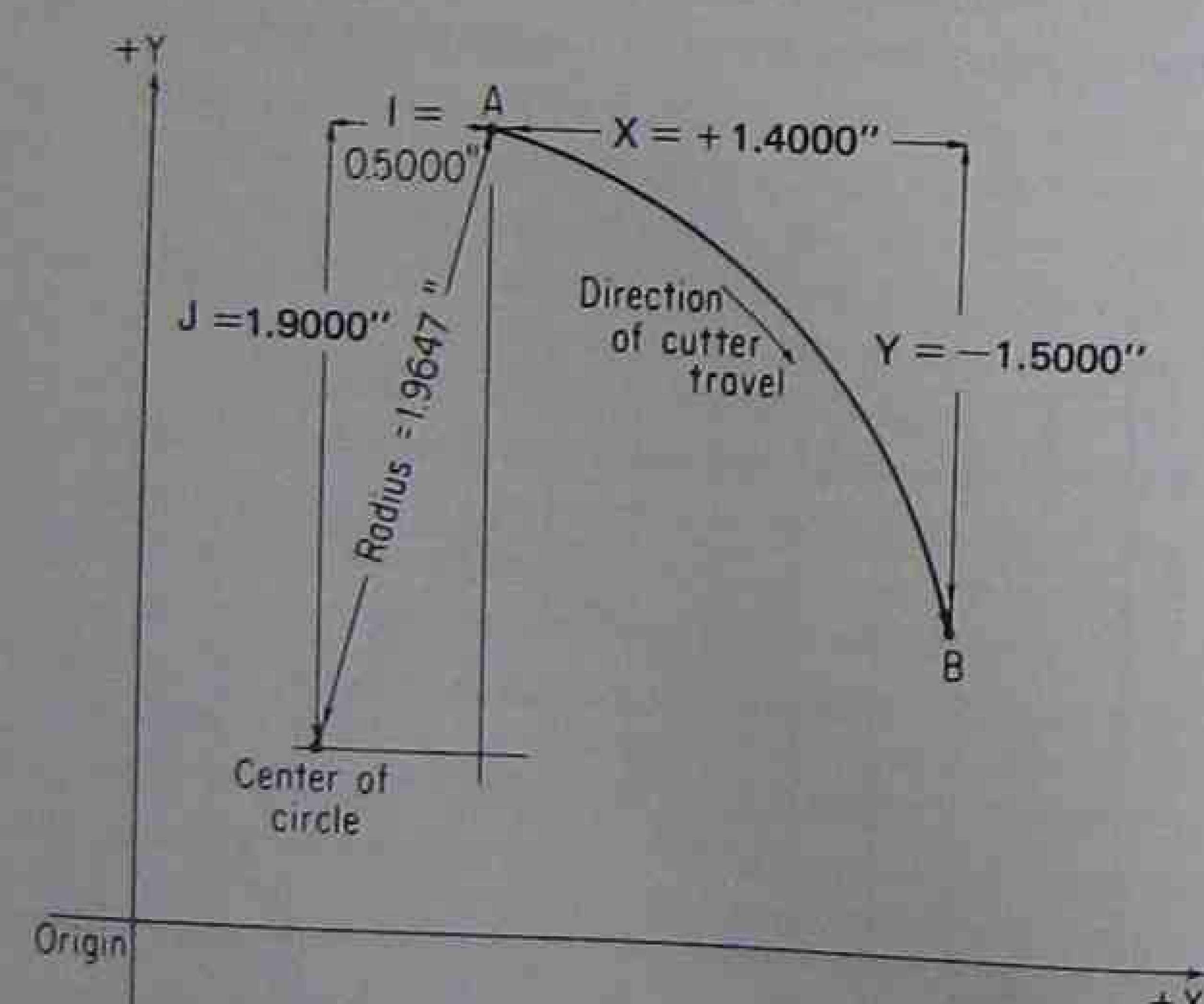
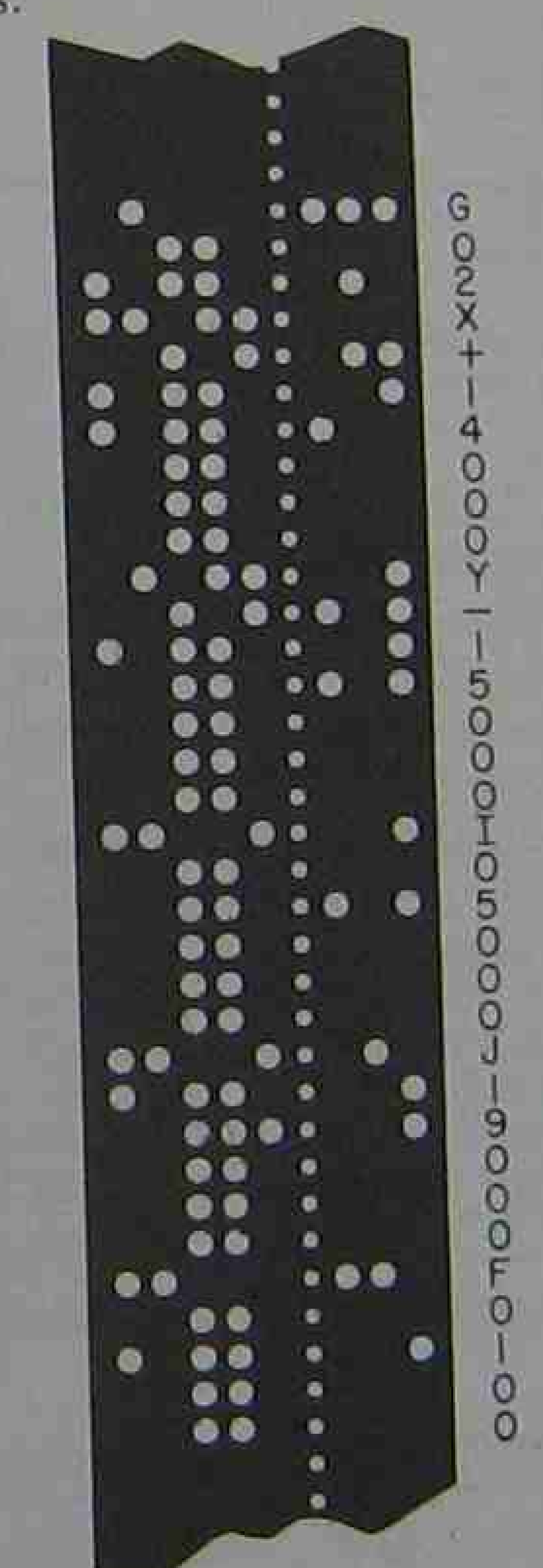


Fig. 4–16. When programming circular interpolation it is necessary to note the horizontal and vertical distances of the center of the circle with respect to the starting point. In this figure *I* and *J* are distances from the center of the circle to point *A*. With circular interpolation, the cutter will travel along the circular path in a continuously smooth curved motion.

arc are described next. In Fig. 4–16, the X movement is +1.4000" and the Y movement is −1.5000".

3. The *I* dimension, which is parallel to the X axis and represents the horizontal distance of point *A* from the center of the circle, is noted next. In Fig. 4–16 this dimension is 0.5000".
4. The *J* distance shown in the illustration of 1.9000" represents the vertical distance of point A from the center of the circle and is also noted on the tape.
5. As with linear interpolation the feedrate must also be described.

The coded instructions on the tape for describing the circular motion of Fig. 4–16 would be as follows:



With most control systems it is possible to drop the zeros either preceding or following the significant digits, however, these zeros are shown for the purpose of clarification.

Also, with most control systems, circular arcs must be divided into quadrants; that is so that no machine axis motion shall reverse during the interpolation of a block. The circular movement shown in Fig. 4–17 would therefore

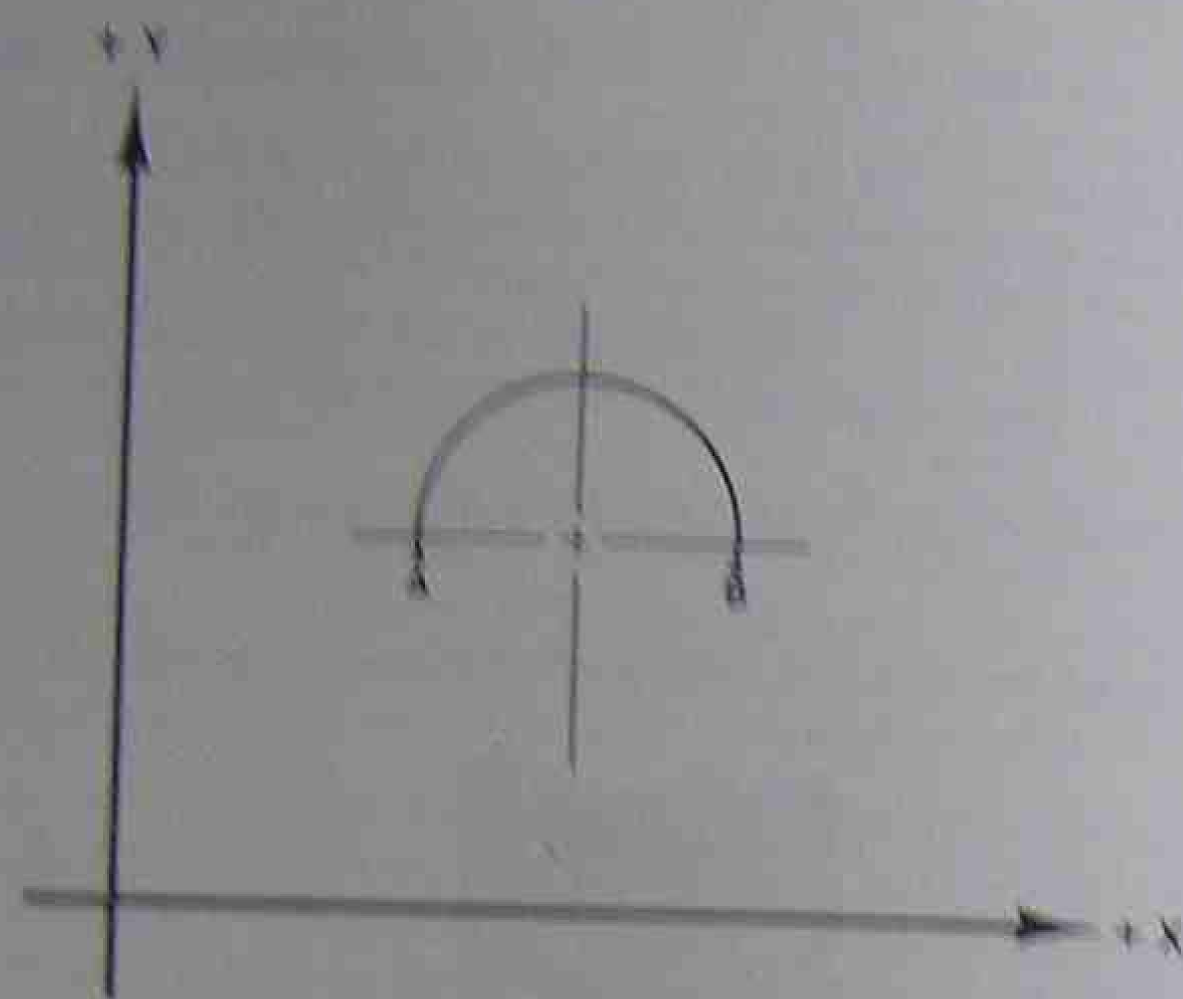


Fig. 4-17. Four blocks of circular interpolation data would be required to move a cutter from point A to point B since the cutter moves through four quadrants.

require four blocks of data. There are some systems that can move through the four quadrants with one block of data.

Circular interpolation, with lathes, is handled similarly except, in this case, I and K words are used instead of I and J words. This is because the plane of action for a lathe is XZ rather than XY, which is most common with machining centers and milling machines. See Fig. 4-18.

For an example of a circular interpolation move on a lathe see Fig. 4-19. If the tip (center of the tip) of the cutter were to move from point A to point B, the block would appear as follows:

N ____ G02 X+ ____ Z− ____ I ____ EOB

In moving from point B to point C the block would appear as follows:

N ____ G03 X+ ____ Z− ____ K ____ EOB

In the first instance I represents the offset of the start of the arc parallel to the

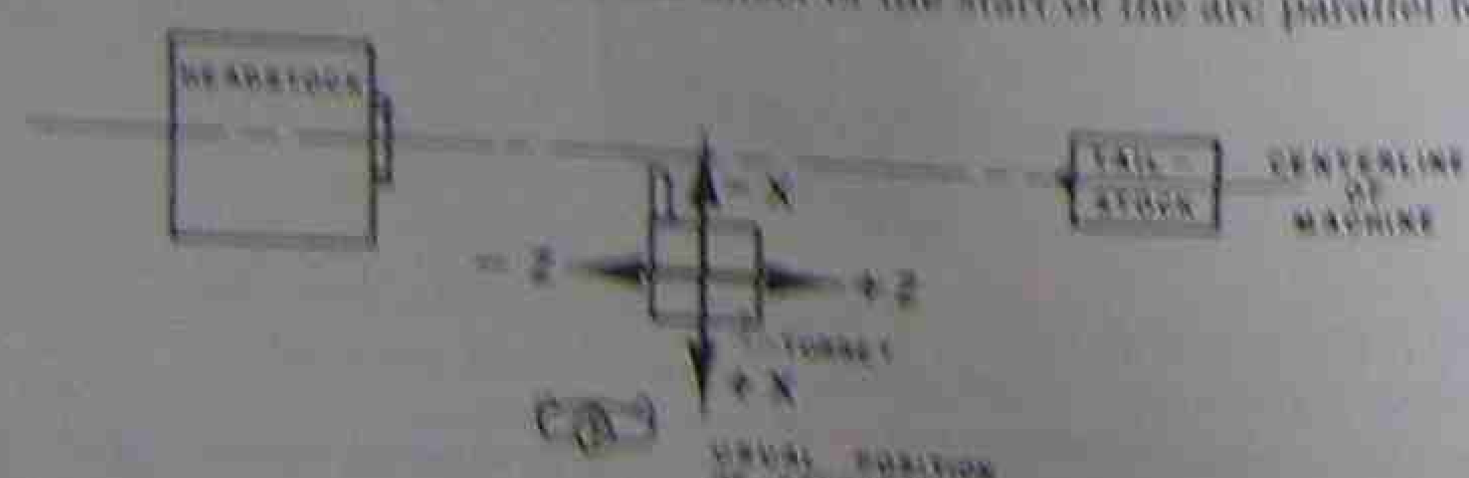


Fig. 4-18. Most lathes operate on a two-axis basis. In accordance with standard practice, the two axes used are X and Z.

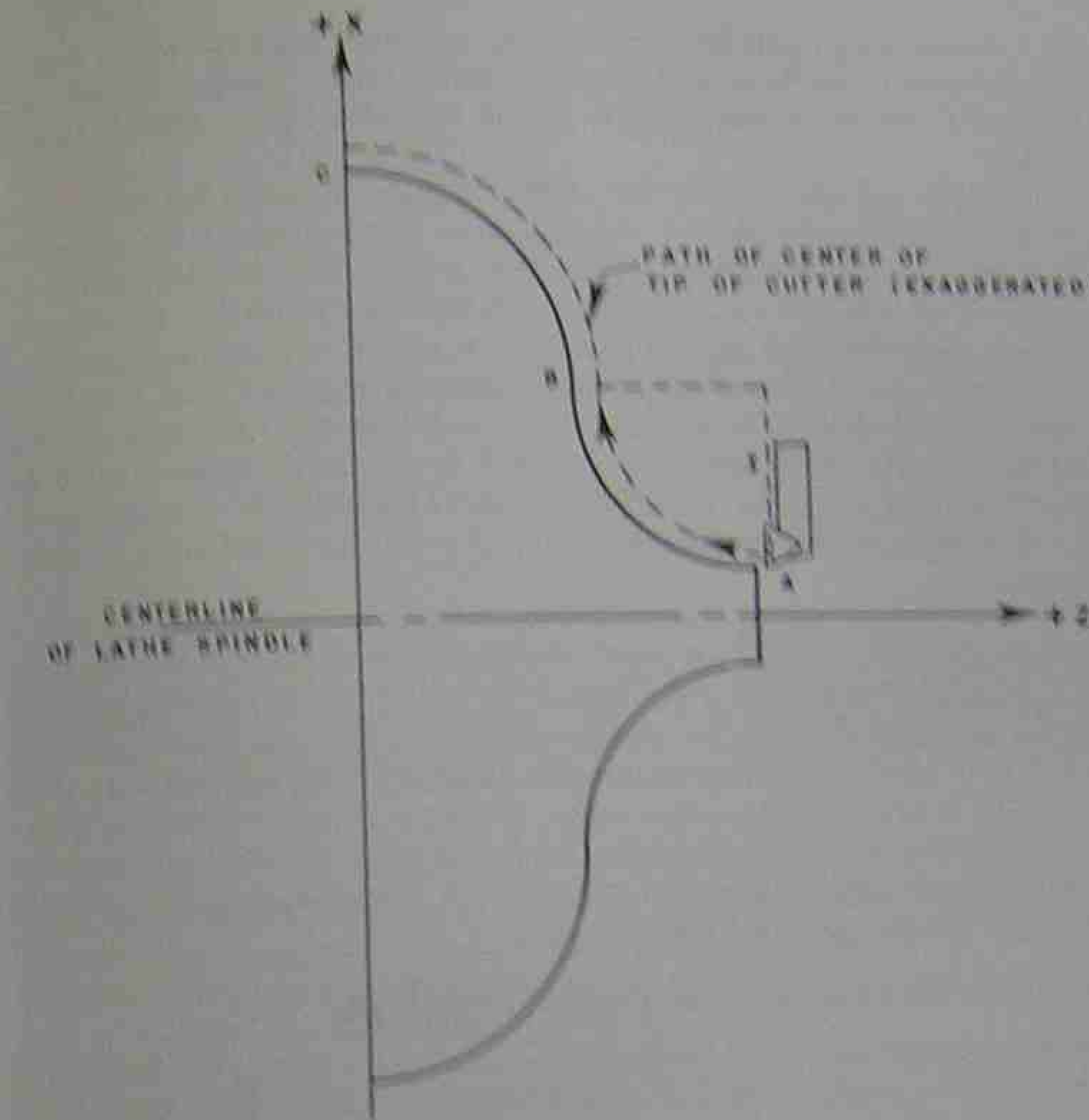


Fig. 4-19. In moving from point A to point B the center of the tip of the cutter moves clockwise (G02), and the offset of the start of the arc is parallel to the X axis and is described as an I word. When moving from point B to point C, the center of the tip of the cutter moves in a counterclockwise (G03) direction, and the offset of the start and the arc is parallel to the Z axis and is described as a K word. In this case the cutter is shown operating above the Z axis, which is common with NC lathes.

X axis. In the second instance the K represents the offset of the start of the arc parallel to the Z axis.

CNC Part Programming Capabilities

There are four major features of a CNC system that can contribute significantly toward reducing the manual part programming effort. While three of these features (all but subroutines) existed before the advent of CNC and can be found on some of the more expensive or later hardwire NC systems, it was not until CNC arrived that they became common, and more extensive regarding capabilities. One of these involves canned cycles, which may often be expressed in one block of data. A drilling cycle, in which the drill (or workpiece) is moved to a specified location and the drill is then moved at a rapid traverse

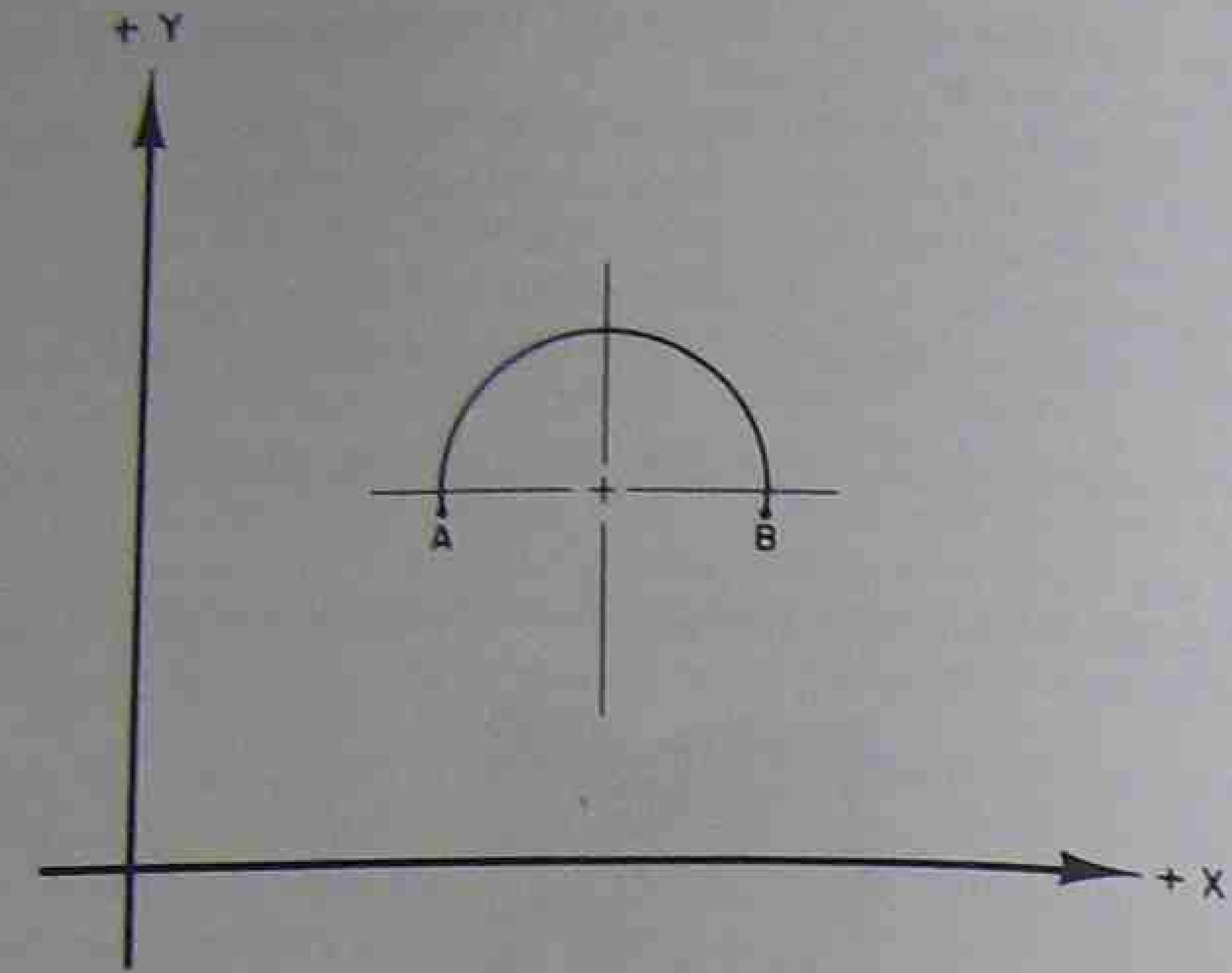


Fig. 4–17. Four blocks of circular interpolation data would be required to move a cutter from point *A* to point *B* since the cutter moves through four quadrants.

require four blocks of data. There are some systems that can move through the four quadrants with one block of data.

Circular interpolation, with lathes, is handled similarly except, in this case, *I* and *K* words are used instead of *I* and *J* words. This is because the plane of action for a lathe is *XZ* rather than *XY*, which is most common with machining centers and milling machines. See Fig. 4–18.

For an example of a circular interpolation move on a lathe see Fig. 4–19. If the tip (center of the tip) of the cutter were to move from point *A* to point *B*, the block would appear as follows:

N ____ G02 X+ ____ Z− ____ I ____ EOB

In moving from point *B* to point *C* the block would appear as follows:

N ____ G03 X+ ____ Z− ____ K ____ EOB

In the first instance *I* represents the offset of the start of the arc parallel to the

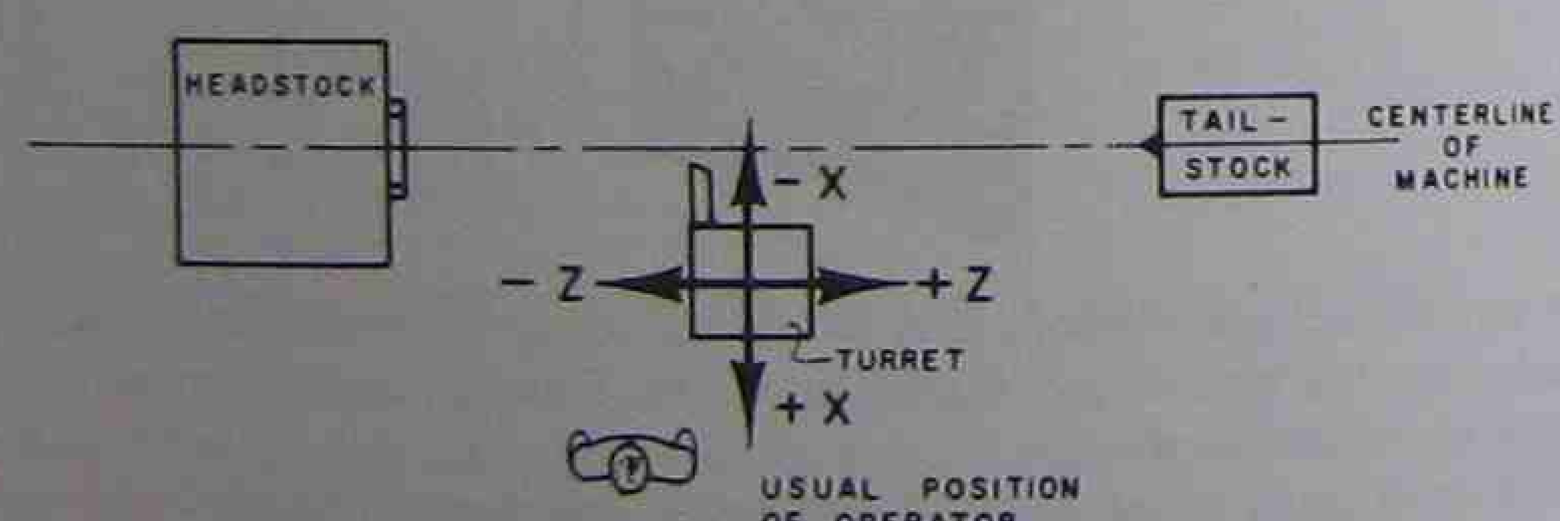


Fig. 4–18. Most lathes operate on a two-axis basis. In accordance with standard practice, the two axes used are *X* and *Z*.

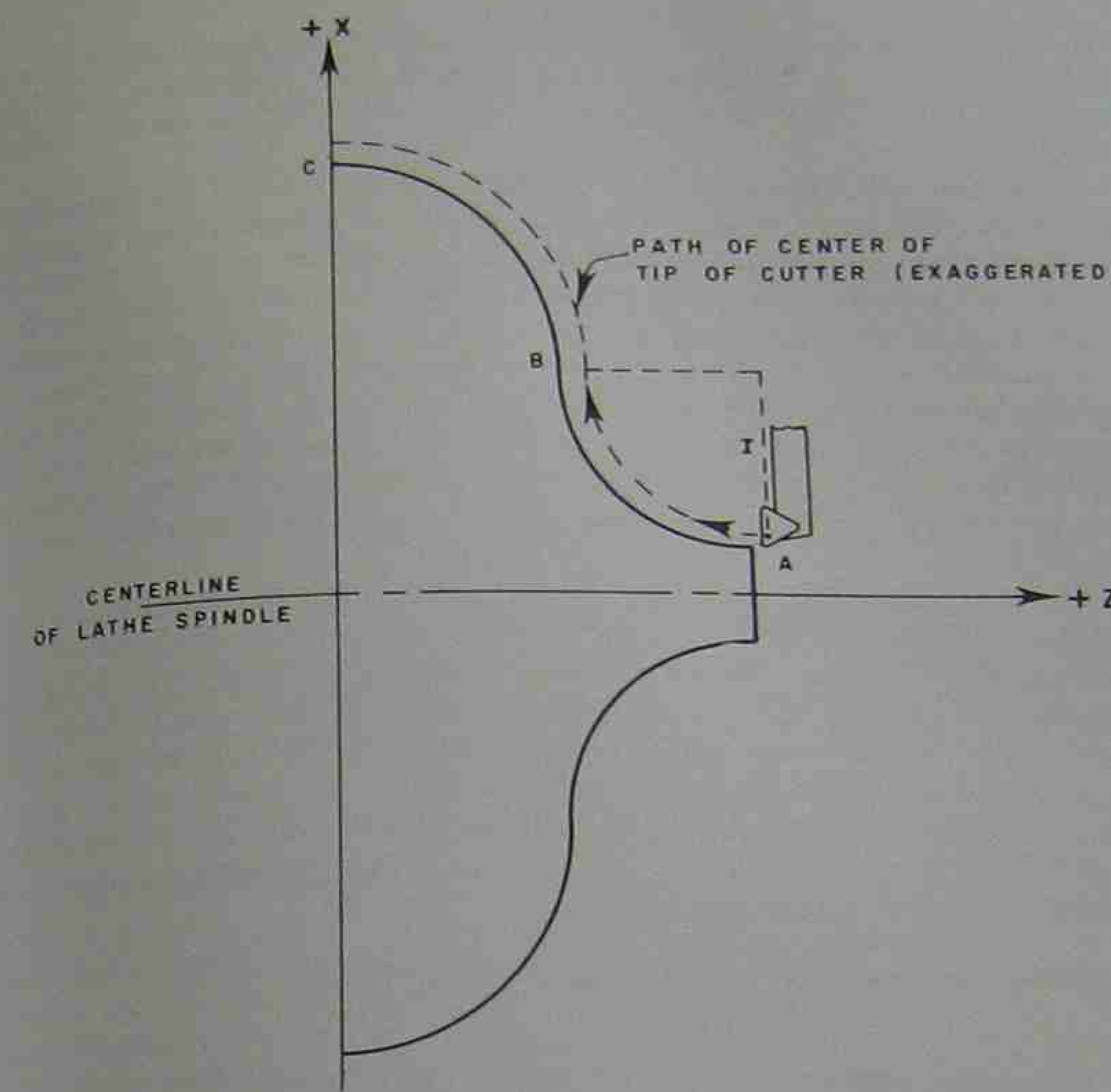


Fig. 4–19. In moving from point *A* to point *B* the center of the tip of the cutter moves clockwise (G02), and the offset of the start of the arc is parallel to the *X* axis and is described as an I word. When moving from point *B* to point C, the center of the tip of the cutter moves in a counterclockwise (G03) direction, and the offset of the start and the arc is parallel to the *Z* axis and is described as a K word. In this case the cutter is shown operating above the Z axis, which is common with NC lathes.

X axis. In the second instance the K represents the offset of the start of the arc parallel to the *Z* axis.

CNC Part Programming Capabilities

There are *four* major features of a CNC system that can contribute significantly toward reducing the manual part programming effort. While three of these features (all but subroutines) existed before the advent of CNC and can be found on some of the more expensive or later hardwire NC systems, it was not until CNC arrived that they became common, and more extensive regarding capabilities. One of these involves *canned cycles*, which may often be expressed in one block of data. A drilling cycle, in which the drill (or workpiece) is moved to a specified location and the drill is then moved at a rapid traverse rate to the material, then drill the material, and then move out of the hole at a rapid traverse rate, is an example of a canned cycle. The words, and their sequence in the block (see Chapter 3), are specified by the control system builder.

A second and somewhat more complex feature is called the subroutine, which may also be referred to as a subprogram. The subroutine is a section of a part program that is prepared by the part programmer and then stored for use at various points in the part program. A bolt hole pattern, as an example, which is to be drilled at various locations on a part may be preprogrammed as a subroutine and then *called* as many times as required. This subroutine concept can be extended by what are known as *parametric subroutines*. In this case specific numerical values are assigned to a letter or symbol when the subroutine is called, such as specific feedrates for the letter "F." The subroutine may be called a number of times and each time a different value for F may be assigned.

The third major CNC part programming aid is *cutter compensation*, which is not a new feature. It has been a not-too-common option on hardwire systems for a number of years. The main reason for its unpopularity with hardwire systems had been its high cost. Another drawback with the hardwire system had been its limited range (approximately ±0.0999"). CNC systems can handle up to a thousand times this amount (i.e., 99.999"). The fourth feature, which is the G92 code and involves the preloading of registers, allows the programmer to shift axes in order to reduce calculations. A more-detailed description of these CNC programming features follows.

CANNED CYCLES (DRILLS, MACHINING CENTERS)

Canned cycles for drills and machining centers are involved principally with Z-axis motions, such as with drilling and boring operations. They are initiated by a G preparatory code, and the complete operation is described in a single block. The G codes normally fall in the 80 series; however, there are some 70 series G codes for canned cycles. As an example, Fig. 4–20 illustrates a canned drilling cycle.[3] The drill cycle starts at point *A*. The drill then moves in an *XY* direction from *A* to *B* at a rapid traverse. Then, again at a rapid traverse, the drill moves to point *C*; then, at a programmed feedrate to point *D*; then back to point *C* at a rapid traverse. The drill could then be programmed to move to point *E* and repeat the feed portion of the cycle by noting the *XY* coordinates of point *E* in a succeeding block. Formats for the two blocks would appear as follows:

N ____ G81 X ____ Y ____ C ____ D ____ F ____ EOB
N ____ X ____ Y ____ EOB

[3] This canned cycle and those shown in Figs. 4–21 through 4–28 are not intended to adhere to any particular CNC manufacturer's format because, while the G number has been standardized in that G81 applies to drilling, G82 applies to drilling with a dwell, G83 applies to peck drilling, etc., the format and the words within the block may vary with each manufacturer. Some manufacturers have also found it more suitable to depart from the standards and apply different cycles to standardized G80 series code numbers.

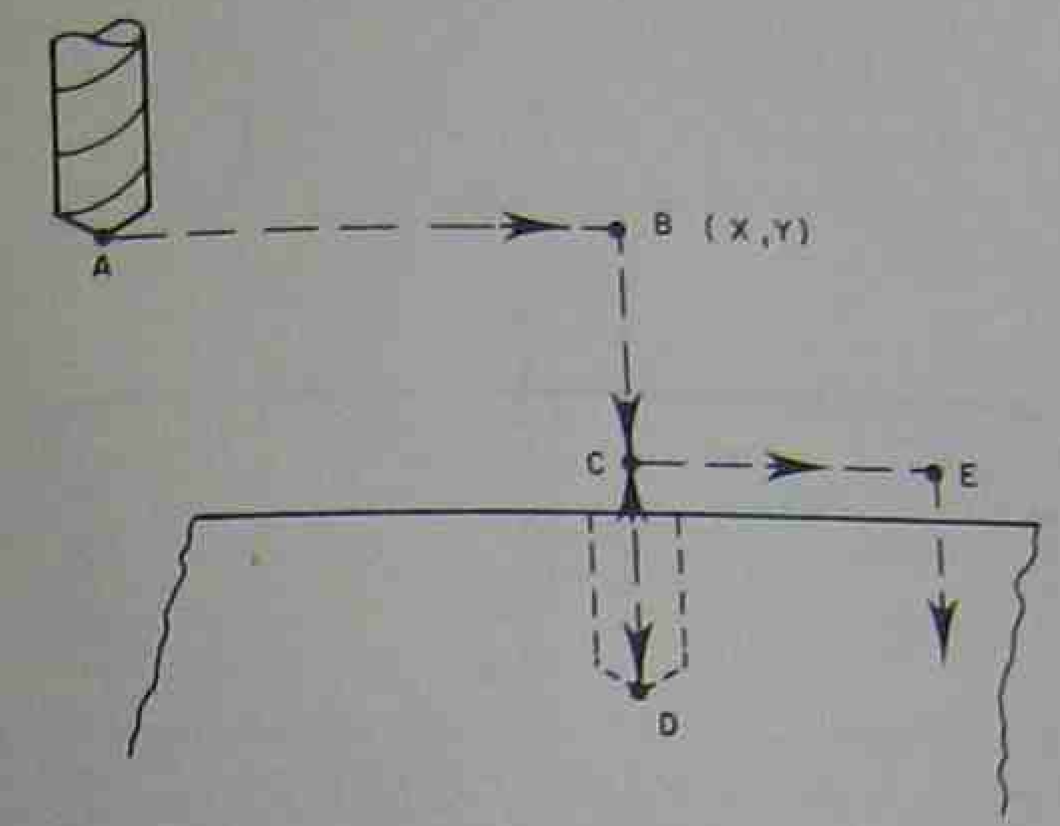


Fig. 4–20. G81, which is a canned cycle *drilling* code, would move the drill point from position *A* to *B* and then down to *C* at a rapid traverse; then from *C* to *D* at the programmed feedrate; then back to *C* at rapid traverse. *X* and *Y* coordinates may be noted in the next block which would move the drill to point *E* and the drill cycle would be repeated. This repeat capability is typical of canned cycles.

The *C* and *D* words would be expressed as *Z* ordinate values. F would be the feedrate in moving from C to D. Other canned cycles in the G80 series are described in Figures 4–21 through 4–28. A canned cycle is cancelled by a G80 code or another G80 series code, such as G81, G82, etc.

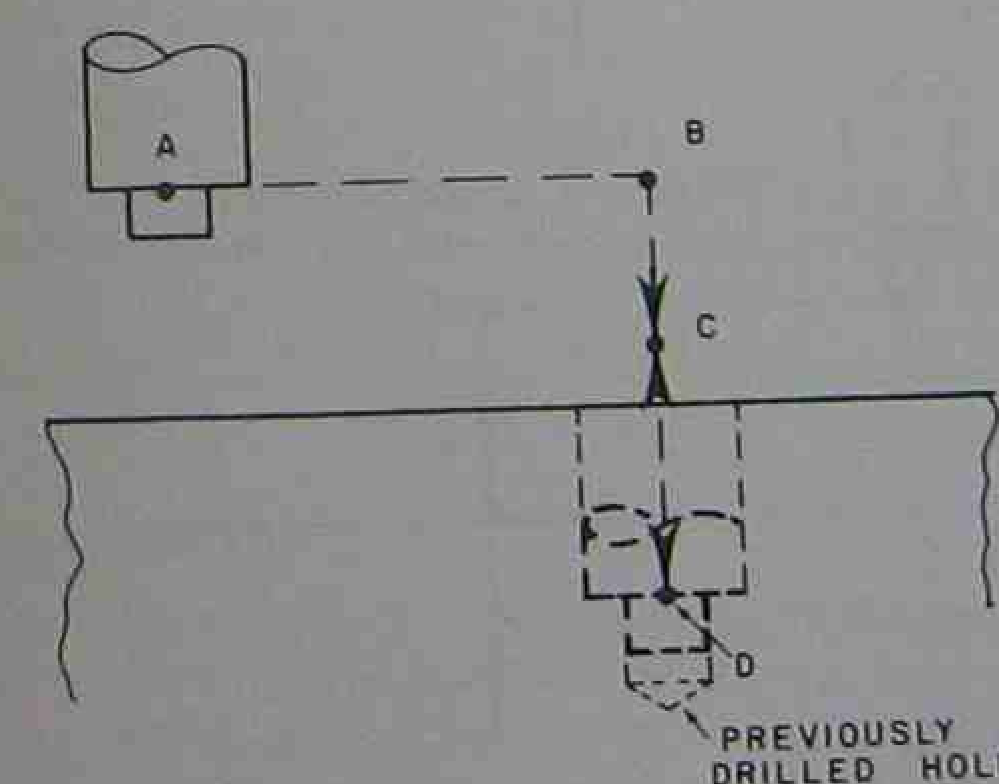


Fig. 4–21. The G82 code is very similar to the G81 code except that there is a dwell period at point *D*. The dwell is used to even out the bottom of a counterbored hole or a spotface. The time in seconds for the dwell is noted in the block. The block might appear as follows:

N ____ G82 X ____ Y ____ C ____ D ____ T ____ F ____ EOB

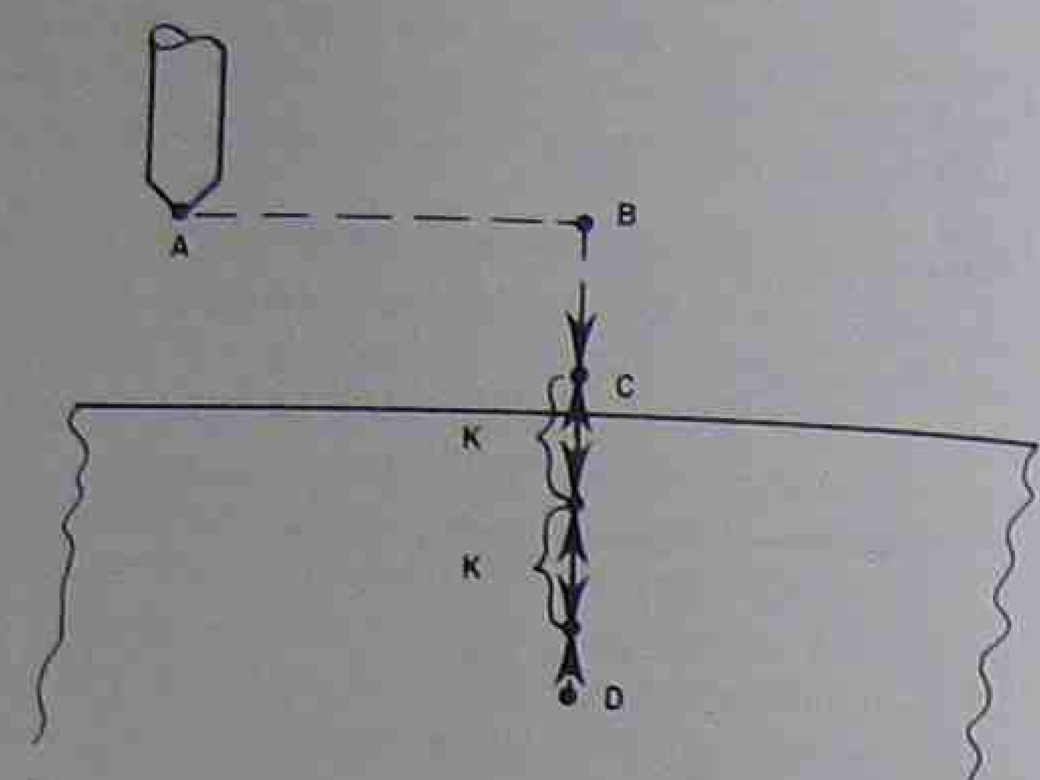


Fig. 4–22. The G83 code is used for *deep hole*, or *peck*, drilling. In this case the drill would move from point *A* to point *B* and then to *C* at rapid traverse; then feed the incremental distance *K*; then rapid back to *C*; then move down at a rapid traverse the distance *K*; then feed another *K* distance; then rapid back to *C*; then move down at a rapid traverse the distance *K* + *K*; then feed to *D*; then rapid back to *C*. The block might appear as follows:

N ____ G83 X ____ Y ____ C ____ D ____ K ____ F ____ EOB

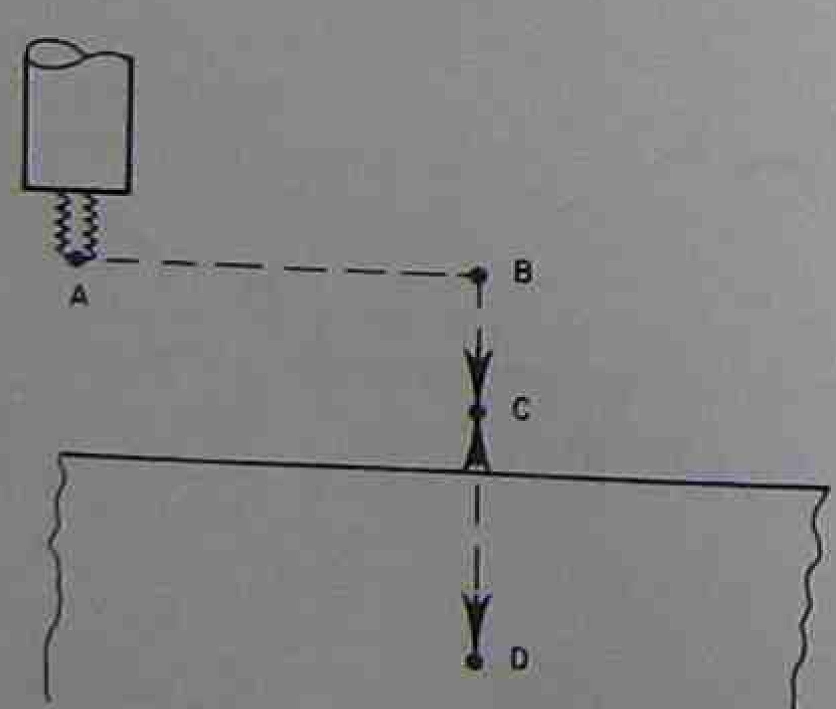


Fig. 4–23. The G84 code applies to the *tapping cycle*. The end of the tap moves from point *A* to point *B* and then to point *C* at rapid traverse; then feeds to point *D*; then reverses and feeds back to point *C*. The block looks very much like that of a drilling cycle, i.e.,

N ____ G84 X ____ Y ____ C ____ D ____ F ____ EOB

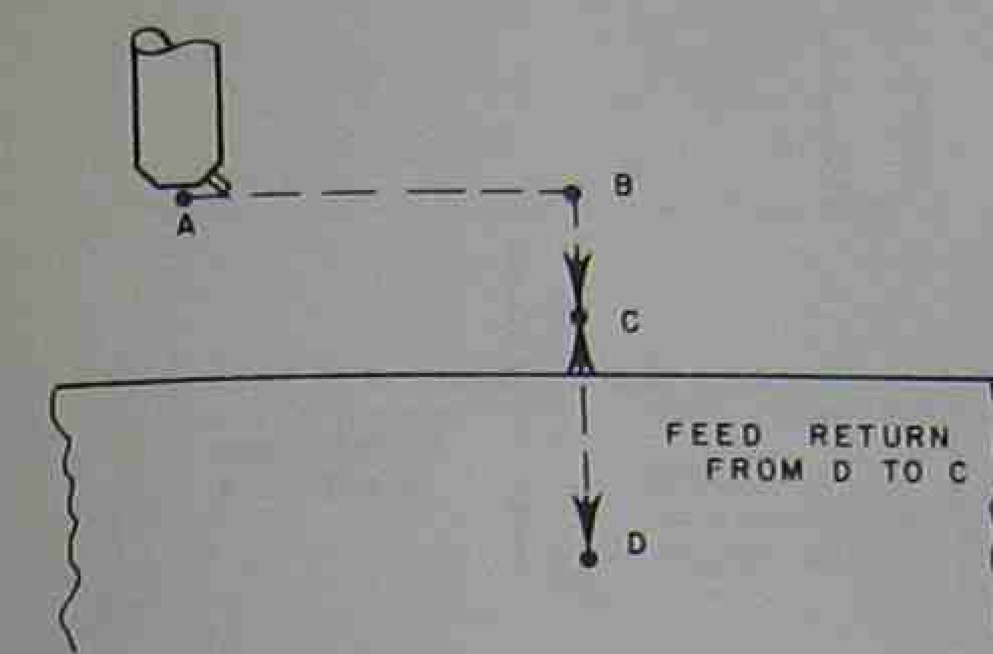


Fig. 4–24. The G85 code applies to boring. The boring tool moves from point *A* to point *B* and then to point *C* at rapid traverse. The tool then feeds to point *D* and then, at the same feedrate, moves back to point *C*, still rotating. The block might appear as follows:

N ____ G85 X ____ Y ____ C ____ D ____ F ____ EOB

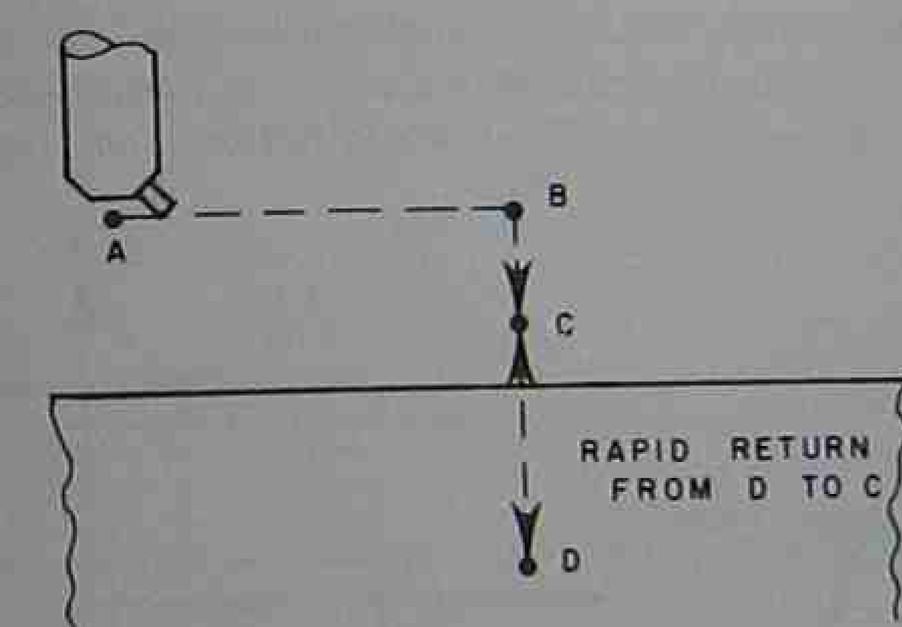


Fig. 4–25. The G86 code applies to boring as does G85 except in this case the spindle stops at *D* and returns to *C* at rapid traverse. The block would appear the same as with G85, i.e.:

N ____ G86 X ____ Y ____ C ____ D ____ F ____ EOB

CANNED CYCLES (LATHES)

While the canned cycles for drills and machining centers are somewhat standardized, at least to the extent of the G codes, there is little standardization for lathe cycles, and each manufacturer has his own format. Formats may even differ for a single manufacturer and may depend on the particular manufacturer's model. Most CNC lathe systems do, however, have some form of a canned cycle for rough turning and for threading. Also subroutines, which are described in the next section of this chapter, may be programmed by the user to suit his particular requirements; and, once developed, these subroutines may be reused, like canned cycles, any number of times.

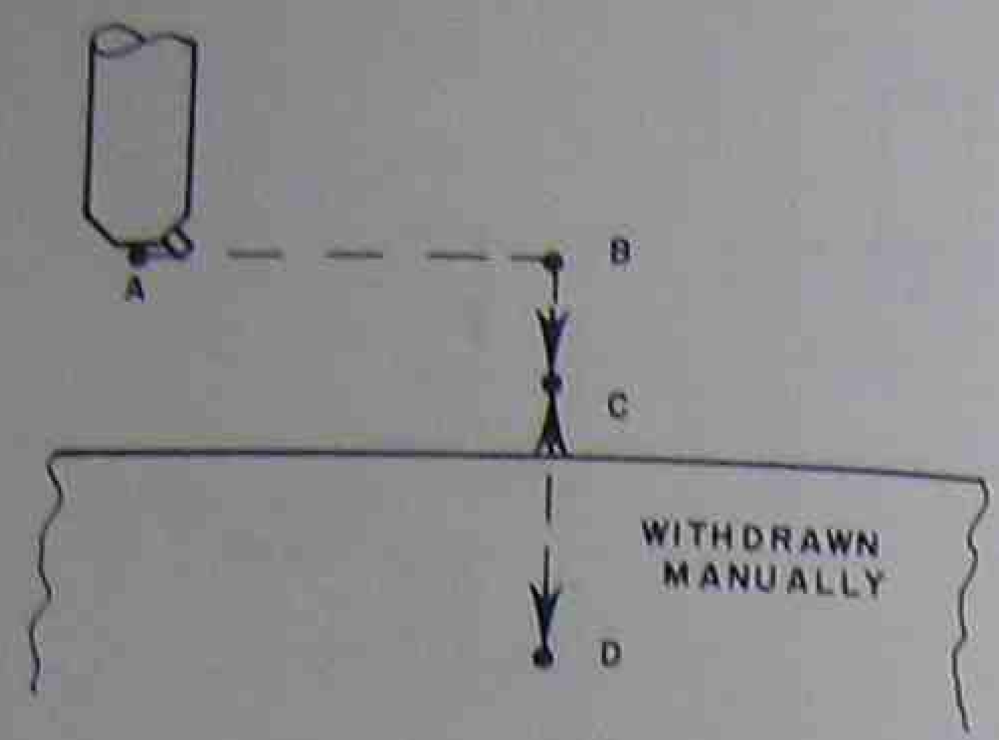
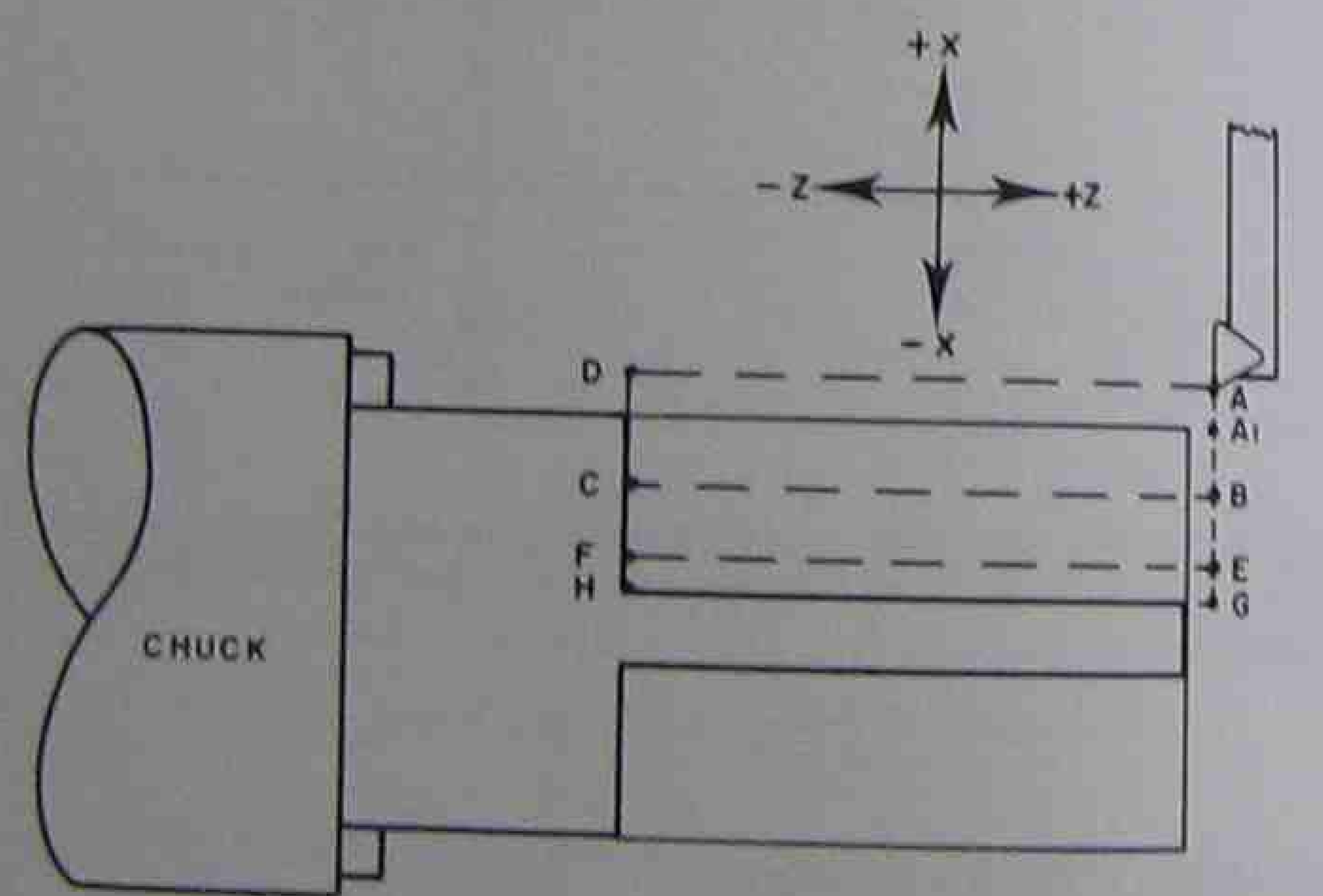


Fig. 4–26. As with G85 and G86, the G87 code also applies to boring, however, in this case, the spindle stops at *D* and is withdrawn manually. The block would appear the same as with codes G85 and G86, i.e.,

N ____ G87 X ____ Y ____ C ____ D ____ F ____ EOB

The format noted below, which is associated with the sketch shown below is not intended to describe the canned cycle for any particular manufacturer's system but rather to describe the *kind* of capability generally available with CNC systems:



In the sketch shown above the tip of the cutter is programmed to move to point *A*, which is a specified distance, in the +*X* direction, from the workpiece. The cutter is then programmed to move from point *A* to point *B* at rapid traverse; and then to point *C* at a prescribed feedrate; then to point *D*; and then back to point *A*, at rapid traverse. Next the cutter moves to point *E*, at rapid traverse; then to point *F* at the prescribed feedrate; then to point *C* and then to point *B*. Next the cutter moves, for the finish cut, to point *G*; then to point *H*; then to *F*, and then back to *A*.

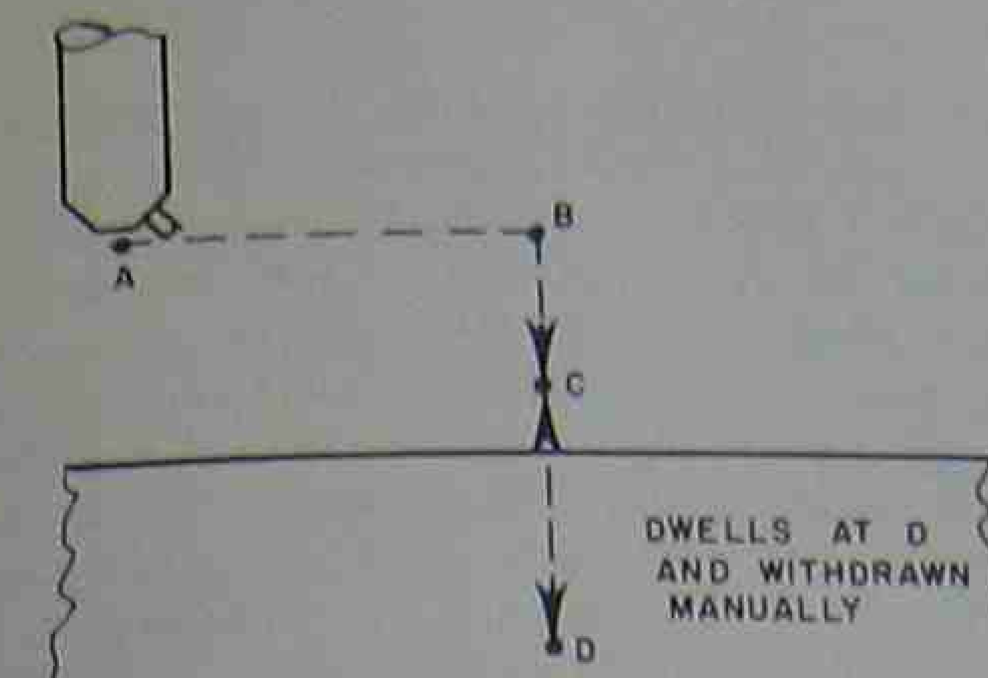


Fig. 4–27. G88 also applies to boring, however differs from the other boring cycles in that there is a dwell at *D* and a manual withdraw. The block would look the same as with the other boring cycles, except for the *T* word which is expressed in seconds:

N ____ G88 X ____ Y ____ C ____ D ____ T ____ F ____ EOB

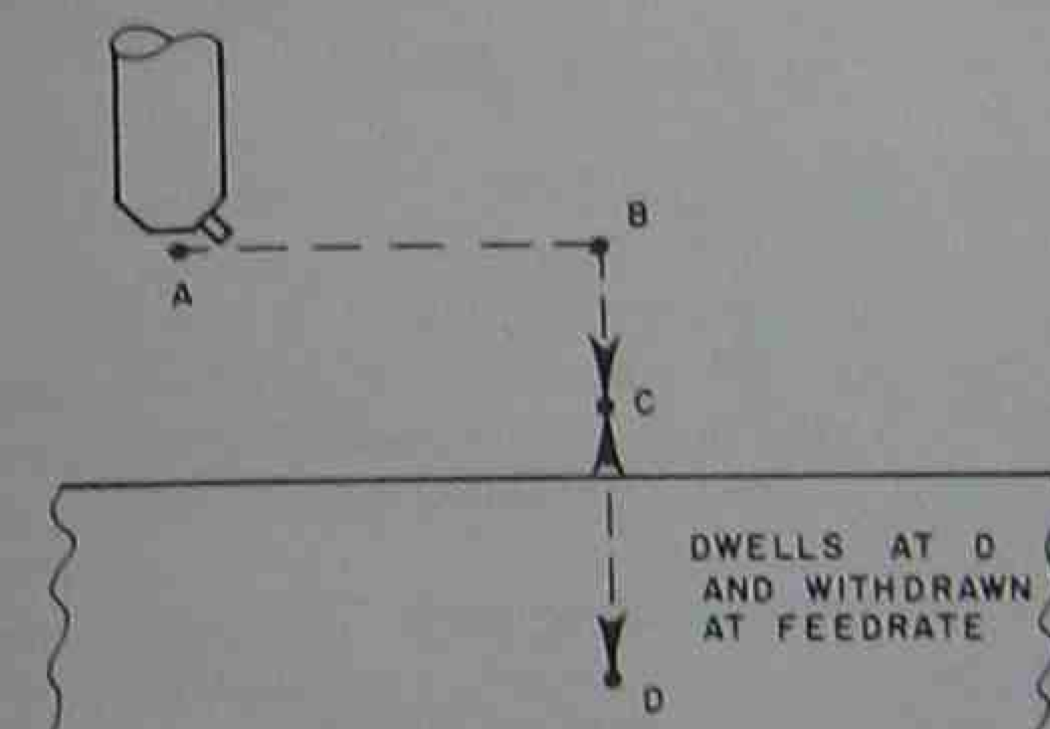


Fig. 4–28. The G89 code is similar to the boring codes of G85, G86, G87, and G88, except in this case the cutter dwells at *D* and withdraws under feed. This is probably the most popular boring code. The block might appear as follows:

N ____ G89 X ____ Y ____ C ____ D ____ T ____ F ____ EOB

All of these moves might be programmed in one statement as follows:

G ____ X– ____ Z ____ N ____ F ____ X(F) ____

The G word would note the type of cutting cycle, e.g., whether rough cutting passes, or threading, or grooving. X would describe the depth of the cut. In this case, the distance from *A'* to *B* and the same distance from *B* to *E*. N would indicate the number of passes. F the cutting feedrate. And X(F) the depth of the final cut, which is the distance from *E* to G.

A canned threading cycle would be very similar, only, in this case, the lead would have to be included in the statement.

SUBROUTINES

A subroutine is a set of commands, or blocks, that are identified and stored in the CNC system and may be activated at any point in the program by "calling" for this set of blocks by noting their assigned identity. The subroutine may also be repeated by noting the number of times it is to be repeated in the "call" block. There is no standard format for describing, or calling for, a subroutine. Each of the CNC system manufacturers has his own format. The example shown below, therefore, does not conform to any particular manufacturer's format; however, it is typical. Consider the illustration in Fig. 4–29. A one-eighth inch diameter end mill starts at the origin and moves to a point over what is to be the lower section of a circular groove. The cutter is next programmed to move down into the material and then in a circular pattern forming the groove. After the first groove is machined the cutter moves to a position for cutting the second groove, and repeats the cutting cycle. The cutter next moves to a third location and cuts a third circular groove. The cutter then moves back to the starting point. The following, which does not conform to any one system manufacturer's format, is an example of what the program might look like. Each block has been numbered, in parentheses, for explanation purposes.[4]

(1) N0010X01Y004325

Subroutine:

(2) N0020M92
(3) N0030Z-0025
(4) N0040G03X00565Y00565J00565
(5) N0050X-00565Y00565I00565
(6) N0060X-00565Y-00565J00565
(7) N0070X00565Y-00565I00565
(8) N0080Z0025
(9) N0090X02M93

(10) N0100M94N0020R2

Line (1) moves the cutter from the starting point to a location over the start of the groove. Line (2) notes that the subroutine instructions are to start with the next block because of the *M92* word. Lines (3) through (8) describe the blocks for machining the groove. Line (3) drops the cutter down into the material. Line (4) starts the circular cutting motion, in a counter-clockwise direction, in the fourth quadrant. Lines (5) through (8) complete the circle and direct the cutter up and out of the groove. Line (9) moves the cutter to the right over the location for machining the second circular groove and then lines (3) through (8) are repeated. Line (9), which is also repeated, moves the cutter to a location for machining the third circular groove. The *M93* word, noted in line (9), ends the subroutine. The blocks are stored in the CNC system and not acted upon

[4] The word format for this hypothetical CNC system is X24; meaning two digits to the left of the decimal point and four to the right. The system also has a *trailing* zero suppression feature, which means that zeros following significant figures in the word need not be shown. X01 in line (1) would therefore be X01.0000 and Y004325, also in line (1), would be Y00.4325.

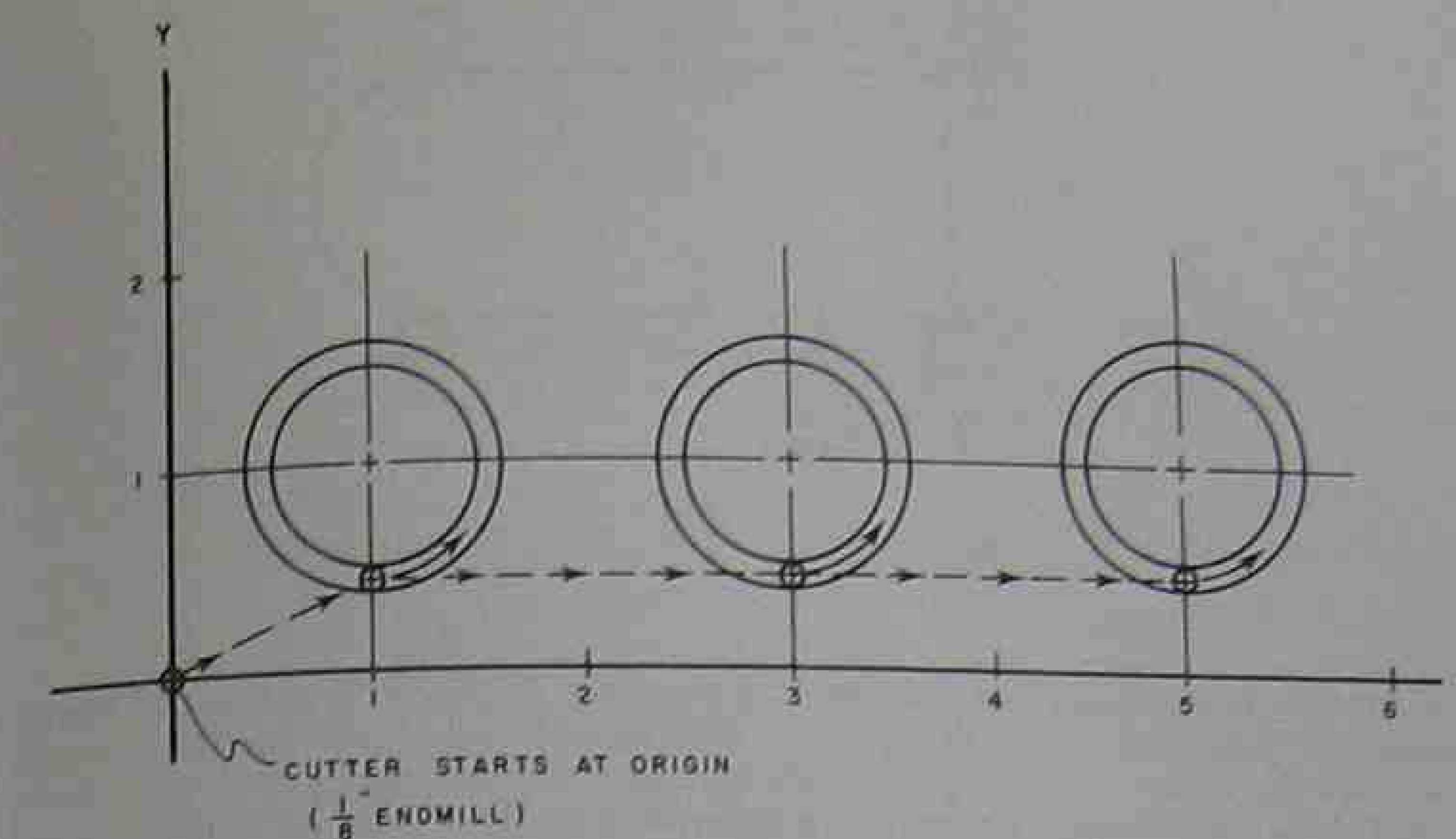


Fig. 4–29. The circular groove pattern shown can be repeated a number of times by use of the subroutine capability of the CNC system. Blocks of data need be described for only one pattern.

until "called" by line (10). The "call" word is M94. N0020, noted in line (10), specifies that the subroutine is to start with sequence number N0020. (M92 is a command for all subroutines and may be repeated in a program; therefore, the sequence number must also be shown in order to identify the start of this particular subroutine.) R2 in line (10) notes that the subroutine is to be repeated twice. The subroutine may be called for at any point in the program and as many times as desired.

A *parametric*, or variable, subroutine would be written the same as that shown above, however, instead of noting the numerical value in a word such as X038, a letter, or punctuated number or letter, might be noted. For example X(Q) might be noted instead of X038. Numerical values could then be assigned to Q as required. In the example shown in Fig. 4–29, if it were required to move the cutter to different depths, the Z value in lines (3) and (8) might be noted as Z(Q). When the subroutine is called, a numerical value may be assigned to Q. If the value of Q, for example, were to be one-half inch, the call statement might appear as follows:

N0220M94N0020R2Q005

CUTTER COMPENSATION

Cutter compensation may be used for slight adjustments to accommodate for the resharpening of a cutter or up to major adjustments to account for the radius of a cutter. It normally applies to two-axis contouring operations and automatically adjusts the center-line path of the cutter to be either closer or farther from the edge of the workpiece. Figure 4–30 illustrates the principle. Assume the center of the cutter (end mill) is to move from point *A* to point *B* and then around the perimeter of the part in a counter-clockwise direction. The data on

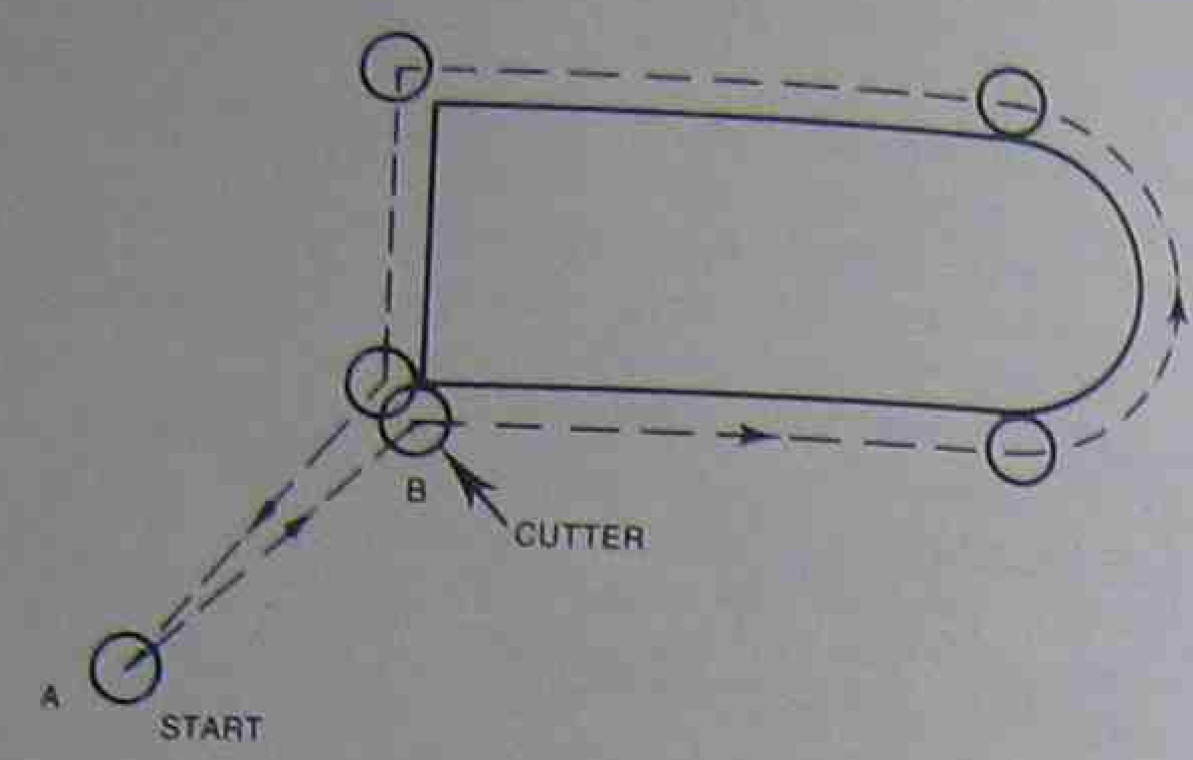


Fig. 4–30. The part programmer describes the geometry of the part and the CNC system calculates the path of the center of the cutter. Data as to the amount of the compensation and whether the cutter is traveling to the left or right of the part must also be considered. CNC system builders handle this feature in different ways, and there are restrictions and special considerations. These should be checked carefully in the part programming manual covering the system.

the tape would describe the coordinates of the part and the CNC system would calculate the points that the center of the cutter is to move through. The amount of cutter compensation, in this case the radius of the cutter, may be inserted manually or by a block on the tape. The part programmer must also note, on the tape, whether the cutter would be moving to the right or left of the part. If the cutter is to move to the right of the part, as is shown in Fig. 4–30, a G42 code would be used. If the cutter were to move in a clockwise direction and be to the left of the part, a G41 code would be used. A G40 code cancels cutter compensation. The use of the cutter compensation feature varies with every CNC system, and there are also limitations and special considerations. System requirements should therefore be carefully reviewed and applied when writing programs and operating the equipment.

G92 CODE—PRELOAD OF REGISTERS

Figure 4–31 illustrates one of the ways in which the G92 code may operate. Rather than program all eight holes from the original *X-Y* axes, the holes in Pattern #1 may be programmed and the same values used for Pattern #2 by shifting the axes (i.e., utilizing axes *X′-Y′* for Pattern #2). This is accomplished by first programming Pattern #1 with respect to the original axes, as follows:

N014X005Y005
N015X005Y01
N016X01Y01
N017X01Y005

then followed with a G92 block, i.e.,

N018G92X-01Y-005

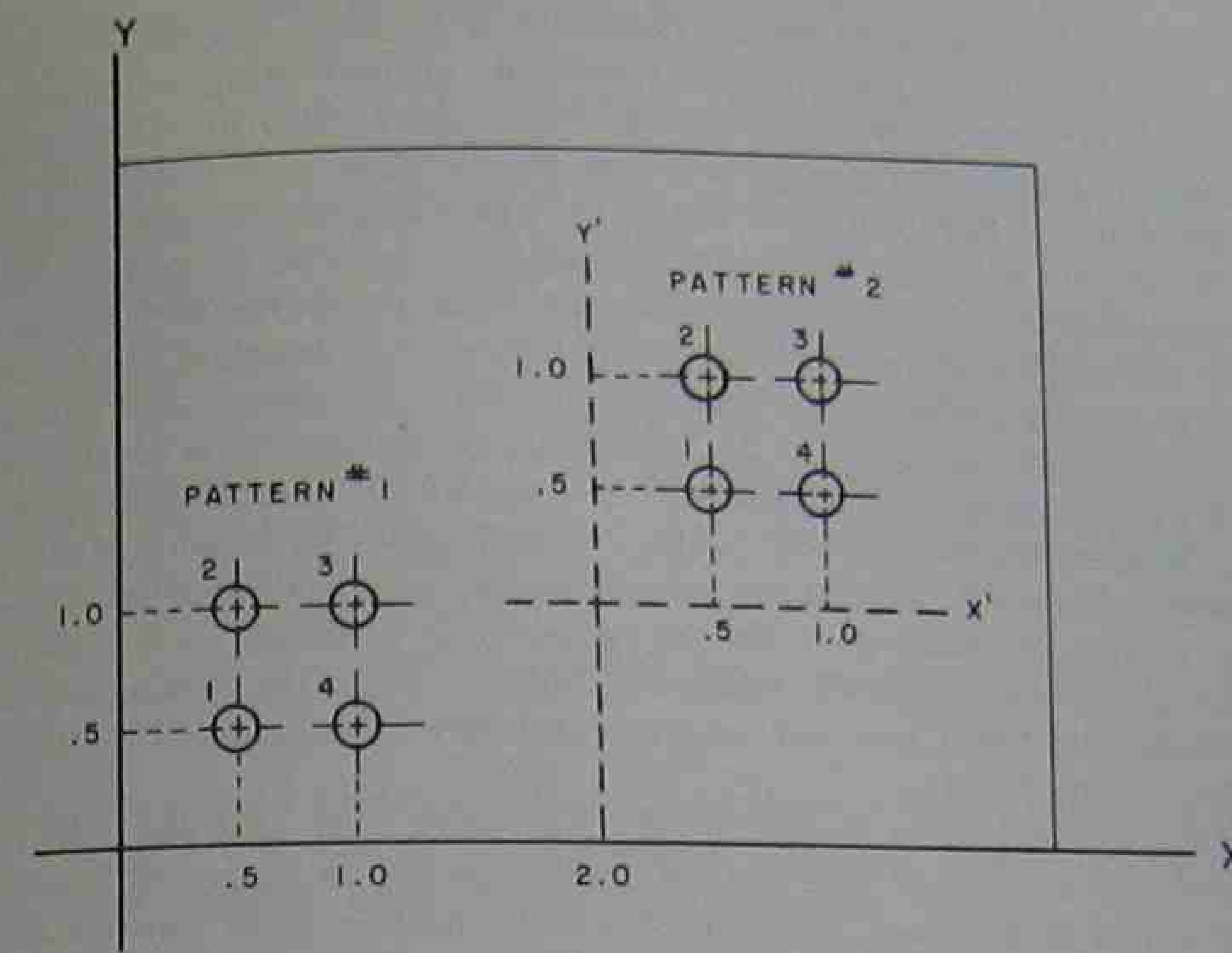


Fig. 4–31. The G92 code allows the part programmer, in effect, to shift the original axes to a more convenient location. Pattern #2, which is identical to Pattern #1, can be programmed with the same *X* and *Y* coordinate values by utilizing the shifted *X′Y′* axes and programming Pattern #2 with respect to the shifted axes.

which is followed with the same block dimensions as for Pattern #1, i.e.:

N019X005Y005
N020X005Y01
N021X01Y01
N022X01Y005

What has happened, in effect, is that the last hole to be drilled in Pattern #1 (i.e.: hole #4) which had coordinates X = 1 and Y = .5 with respect to the original axes, has been changed to have coordinates X = −1 and Y = −.5 with respect to the *X′Y′* axes of Pattern #2. The blocks that follow will be programmed with respect to the new *X′Y′* axes. G92 does not initiate any motion. It changes the registered location of the last position of the cutter.

When is Manual Part Programming Suitable?

Whether a part should be programmed manually or with external computer assist depends chiefly on the complexity of the part, on the type of control system the part is to be programmed for (whether hardwire NC or softwire CNC), and, to some extent, on the complexity of the NC machine tool. If the control system is of the hardwire type, there is very little, if any, computation that can be handled within the system and even relatively simple parts are better programmed with the aid of a computer. If, however, the control system is of the softwire (CNC) type, which practically all now being manufactured are, the need for external computer aid is lessened since the CNC system incorporates a computer and can handle a good deal of the computations required. There is, however, a practical limit to the part programming capability of a CNC system, and parts that might be considered complex are usually better programmed with the aid of a separate and external computer (see Chapter 5).

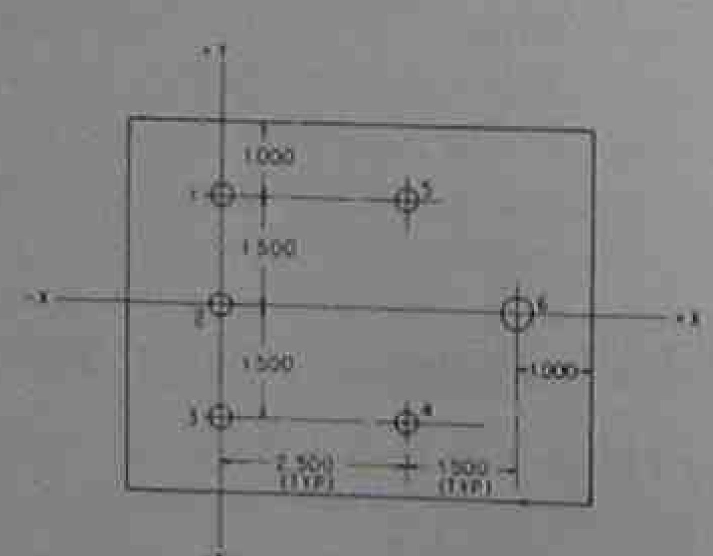
The complexity of the machine tool, while not as important as the complexity of the parts when considering manual vs. computer-assisted part programming, should, nevertheless, be considered since the number of codes and special part programming requirements increase as the number of features on the machine tool increases. For example, a machining center with a tool changer and rotary table would require the part programming of tooling codes and tool offsets, and an auxiliary code for the rotary table position. Lathes also are usually equipped with a goodly number of features, including threading cycles, tool offsets, and turret and tool selection and may be considered reasonably complex NC machines.

Another consideration is the number of different parts that are to be programmed over a period of time. If production runs are long and relatively few parts are to be programmed, then even the more complex parts may justifiably be programmed manually with the assistance of the internal computer in the CNC system. If, on the other hand, a large number of different parts are to be programmed, then external computer assistance may be justified, even with simpler parts.

Computer-assisted part programming is covered in Chapter 5.

QUESTIONS TO CHAPTER 4

1. Considering the plane formed by the intersection of the *X* and *Y* axes, how many quadrants may be available to a part programmer?
2. According to the Cartesian coordinate system, all dimension values must be positive. (True, False.)
3. The following hole pattern is to be drilled on a relatively simple, two-axis, numerically controlled drill press.



Holes 1, 2, 3, 4, and 5 are ½ inch in diameter. The diameter of hole 6, is 1 inch.

A partially completed manuscript is shown below. Fill in the blanks. Spot drilling, in this instance, is not necessary.

If there is no change in the movement, the ordinate need not be repeated.

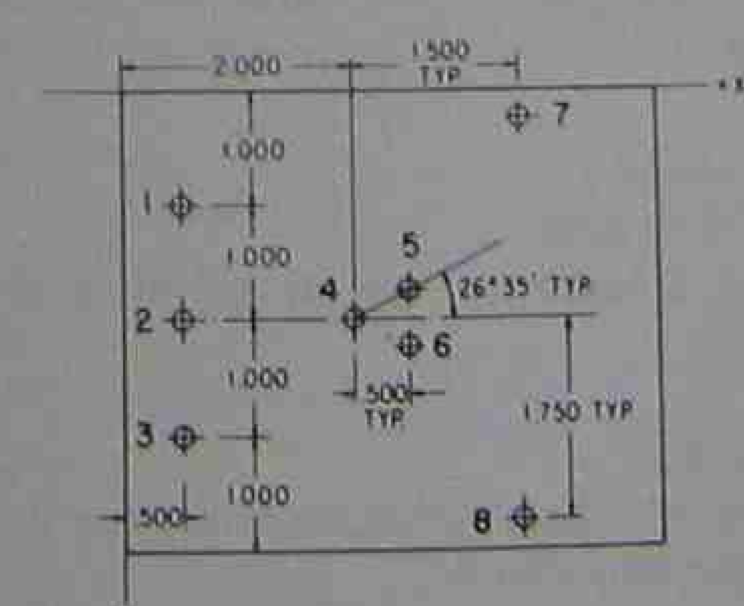
MANUSCRIPT PTP XYZ MACHINE — Part No. / Part Name / Date / Prep By / Checked By

Sequence No. (N)	X		Y		Aux. Code (M)	Comments
001		0	+			Use ½" HSS drill
002						Manually set depth
003		0	-	1.500		@ [illegible] and drill [illegible] holes
004	+					Machine will stop. Change
005			+	1.500	M00	to 1" HSS drill and drill [illegible]
006	+			0	M30	Machine will stop and [illegible]. Change
						to ½" HSS drill, take out
						completed part and put
						in new stock

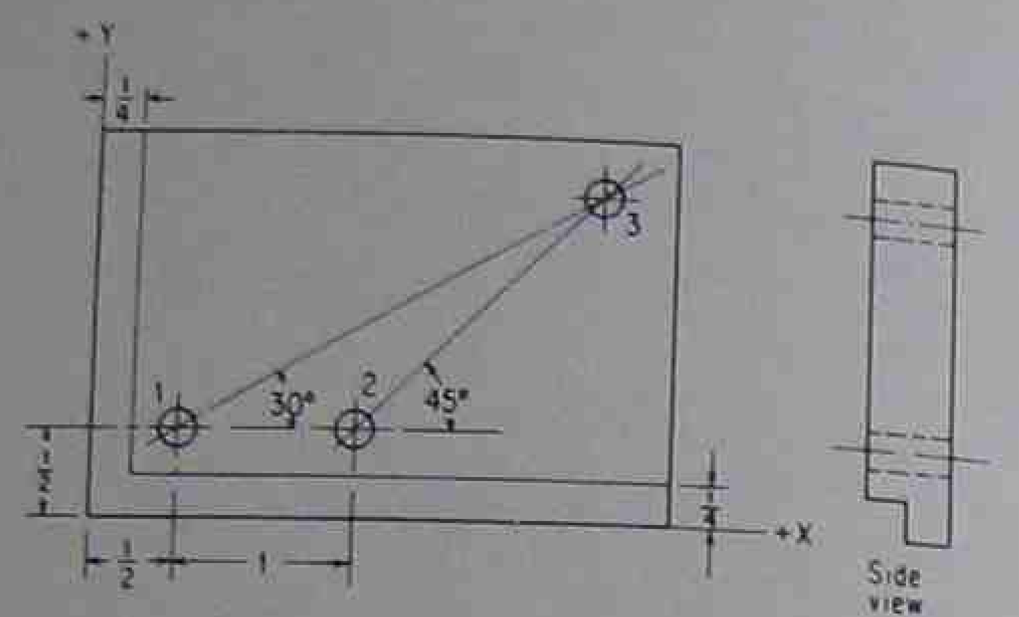
4. Complete the manuscript following, for the part shown below. The origin (x = 0; y = 0) is located in the upper left-hand corner. All holes are ½ inch in diameter. Spot drilling is not necessary.

MANUSCRIPT PTP XX MACHINE — Part No. / Part Name / Date / Prep By / Checked By

Sequence No. (N)	X		Y		Aux. Code (M)	Comments
001	+	.500	-	1.000		Use ½" HSS drill
002			-			Manually set depth
003			-	3.000		@ 2.500 inches and
	+	2.000	-	2.000		drill 8 holes.
005	+	2.500	-			
006	+	2.500	-			
007	+	3.500	-	.250		
008	+	3.500			M 30	



5. a. Note the x and y coordinates of the three holes shown in the part following:

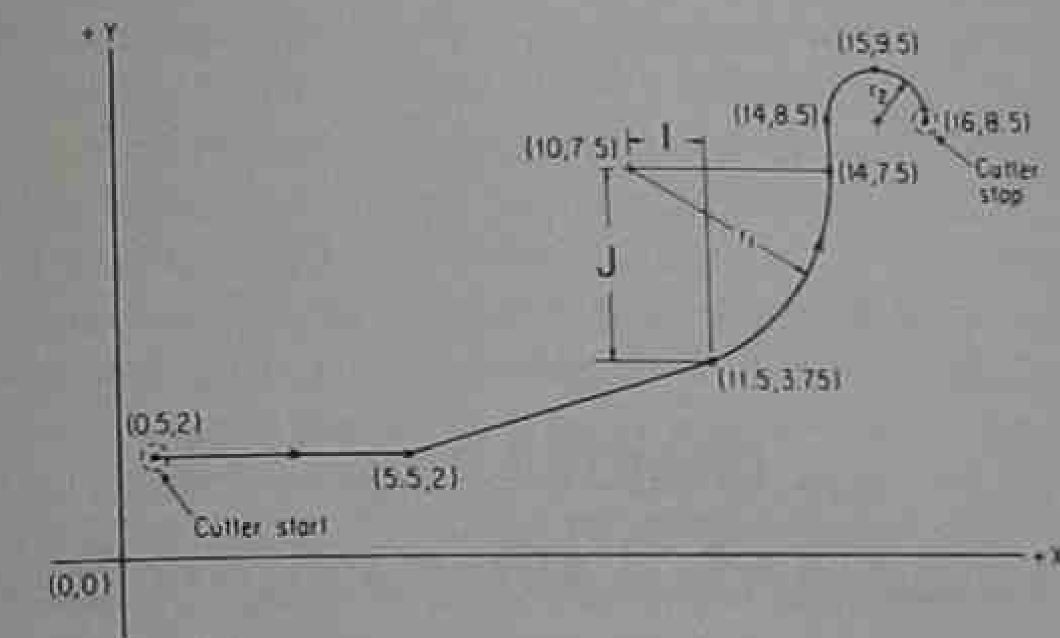


b. What would be the x and y coordinates if the 45° angle were changed to 40°?

6. Circular interpolation reduces the number of straight-line segments required to be calculated when a machine is moving about a circular arc. (True, False.)
7. The closer the tolerance required with linear interpolation, (a) the (greater), (less) will be the number of line segments. The larger the radius of the arc, (b) the (greater), (less) will be the number of line segments, if the same tolerance is to be held.
8. The most popular method of developing line segments is to:
   a. Run straight lines *outside* the curve.
   b. Run straight lines *inside* the curve.
   c. Run straight lines both *inside* and *outside* the curve.
9. With most control systems how many blocks would be required to move around a complete circle (i.e., 360°) when utilizing circular interpolation?
10. When using circular interpolation with a lathe, the letter offset parallel to the *Z* axis is _____.
11. If the *X* and *Y* axes movements are timed to be equal, the resultant linear movement would be at what angle to the *Y* axis.
12. Canned cycles have been strictly standardized, and the word format is exactly the same for most CNC systems. (True, False.)
13. In the first column are shown the various canned cycles. The second column represents a list of preparatory codes. Match the functions in the left-hand column with the codes in the right-hand column.

a. Drill plus dwell	1. G89
b. Deep hole drill	2. G81
c. Boring, spindle rotating on withdraw at feedrate	3. G85
d. Drill	4. G84
e. Tapping	5. G82
f. Boring, spindle rotating on withdraw at feedrate plus dwell	6. G83

14. A subroutine is acted upon as it is read by the reader. (True, False.)
15. Parametric subroutines are used exclusively for describing the path around the outside of a part. (True, False.)
16. When using cutter compensation, what is the code word if the cutter is to travel to the left side of the part?
17. Most CNC systems can handle up to (a) ±0.001" (b) ±0.9999" (c) ±99.9999" (d) 999.999" of cutter compensation.
18. Cutter offset and cutter compensation mean approximately the same thing. (True, False.)
19. Fill in the missing X, Y, I, or J words in the program listed below. The sketch applies to both linear and circular interpolation. Coordinates are shown in parentheses adjacent to their corresponding points. The system follows an incremental format.



N	G	X	Y	I	J	
001	01	X + 5.0000				
002			Y + 1.7500			
003		X + 2.5000		I 1.5000		
004			Y + 1.0000			
005		X + 1.0000	Y + 1.0000	I 1.0000		
006		X + 1.0000			J 1.0000	

CHAPTER FIVE

Computer-Aided Part Programming

WHAT IS COMPUTER-AIDED PART PROGRAMMING?

Computer-aided part programming may be defined as a means for assisting a part programmer with the preparation of instructions for an NC machine. The instructions are usually in the form of a punched tape; however, they may be on a floppy disc or magnetic tape or transmitted directly from the computer. Punched tape is, however, by far the most common media and probably represents about 98% of all input media, not counting those control systems that normally handle direct manual data input (MDI).

The punched tape that would be prepared by the computer is very similar to that which would be prepared manually. The computer acts as an *aid* in preparing the tape and, when properly administered, can reduce the time and cost for preparing a correct tape considerably. The computer, in this case, is not part of the CNC system, but rather external to it.

WHEN IS COMPUTER-AIDED PART PROGRAMMING SUITABLE?

Whether a computer should be used for assisting in the preparation of NC tapes depends on a number of factors, namely,

1. The complexity of the parts to be programmed. With contour parts this would refer principally to the geometry of the parts. If three-, four-, or

five-axis simultaneous contour machining is required, there is generally little question that computer assistance is required. Even in cases of two-axis contouring that would require trigonometric calculations, computer assistance should be carefully considered.
2. The number of parts to be programmed. If production runs are short and the quantity of different parts to be programmed is consequently large, and even if the parts are relatively simple, it would probably be wise to consider computer assistance. There are no hard figures on this; however, if a company ran three parts a year on an NC machine, and the parts were fairly complex, then it might pay to laboriously toil through the three parts via manual part programming. If, however, a company ran 50 parts a year over an NC machine, even if the parts were relatively simple, computer assistance should be considered. This would also depend, of course, on the number of NC machines within the company. Whether a company should use a computer would, in the final analysis, depend on the *total* number of parts programmed for *all* of its NC machines. If the number is high, and even if the parts are relatively simple, computer assistance should be seriously considered.
3. The parts are simple; however, the tapes are long due to redundant operations. Keeping in mind that part programs prepared manually must be *typed* manually, and considering the typing ability of most part programmers, there is generally a good deal of room for typing errors. The computer almost never makes an error in automatically typing the tape. What, therefore, is a *long* tape? This would depend on the typing ability of the person typing the tape; however, anything over 20 feet might be considered susceptible to an unacceptable number of errors for the average tape punching typist. Again, as with the number of parts programmed, if the tapes were to be long, but there were few of them, then manual programming might be in order regardless of the typing ability of the typist.
4. The NC machines are complex. Even with relatively simple part geometry, such as parts associated with a lathe, the different words to be considered and their relationship to the machine's functions can be quite numerous with complex NC machines, such as tool-changing machining centers and many lathes, especially four-axis lathes.
5. The total number of *different* machines in the shop. One of the key advantages of computer-aided part programming is that the rules for programming one machine can be the same as those for programming a similar machine. In writing a part program that is to be processed by a computer, the part programmer need not be too concerned with the tape coding and the exact tape format for the particular machine since the computer prepares the precise formats required. While the part programmer using computer assistance must understand the operations of the machine and have an understanding of the manual part programming requirements, he or she need not be as intimately familiar with the detailed coding and format of every machine as would be required if the parts were to be manually programmed.

Courtesy of University Computing Company

Fig. 5–1. The APT system may be utilized with computers ranging from mini to large scale. The arrangement shown consists of a minicomputer, keyboard and printer, keyboard and CRT, plotter, and high speed tape punch.

APT and COMPACT II

Unquestionably the two most popular computer-assisted part programming languages are APT and COMPACT II. It is estimated that at least 75% of the companies and organizations utilizing computer-assisted part programming, external to the CNC system, utilize one of these two languages. APT stands for Automatically Programmed Tool and was developed by MIT under Air Force sponsorship during the very early days of numerical control (circa 1955–1960). Implementation and further development were to be followed up by the Aerospace Industries Association and the Illinois Institute of Technology Research Institute. Advancements and refinements of APT are continuing under the sponsorship and guidance of an organization known as CAM–I.[1] APT systems are offered in varying degrees of sophistication by a number of companies and are suitable for a large variety of computers ranging from mini (see Fig. 5–1)

[1] CAM–I stands for Computer-Aided Manufacturing–International, Inc. and is comprised of approximately 100 company members, both within the United States and abroad. The companies contribute funds for specific computer-related projects that are expected to improve manufacturing techniques and increase productivity. CAM–I is located at Suite 1107, 611 Ryan Plaza Drive, Arlington, Texas 76011.

Courtesy of Control Data Corporation

Fig. 5–2. This large scale computer, which is used quite frequently for numerical control calculations, is capable of performing several million calculations per second.

to large scale (see Fig. 5–2). The system is also offered on a remote time share[2] basis whereby a user needs only a telephone terminal, such as a Teletype unit. (See Fig. 5–3.)

COMPACT II is a proprietary and commercially available system offered only by Manufacturing Data Systems, Inc.[3] The system is based on a language called SPLIT, which was introduced in approximately 1960. MDSI began offering COMPACT II, on a remote time share basis, in 1969. The system is now also being offered for installation on in-house minicomputers. Both APT and COMPACT II are discussed, in additional detail, in this chapter.

Computers

Before going into detail about the APT or COMPACT II languages it is probably appropriate to discuss the computer. To begin, there are two very

[2] Time-sharing is a practice in the computing field which enables a number of users to run different programs on the same computer almost simultaneously. Although the computer is generally operating on only one program at any one instant, it has the capability of switching from one program to another with lightning speed thus being able to satisfy a number of users, at different locations at any one time.

[3] Manufacturing Data Systems, Inc. (also known as MDSI) is located at 4251 Plymouth Road, P.O. Box 986, Ann Arbor, Michigan 48106.

Courtesy of General Electric Company

Fig. 5–3. Computer-assisted part programming may be accomplished on a remote time-share basis via telephone lines. The Teletype installation shown ties in directly with a large-scale distant computer. Preparation of the tape is almost instantaneous, since the user has direct time-share access to the computer.

broad categories of computers, namely, analog computers and digital computers. An analog computer is usually designed for a special purpose and, although sufficiently accurate for its requirement, its accuracy is limited. It might be compared to a slide rule, or the speedometer of a car. The numbers and values are usually directly related to a controlled variable source such as a voltage. A digital computer, on the other hand, can be made as accurate as we would want to make it and is therefore used for numerical control part programming. Most computers are of the digital type. The digital computer works on only two signals, either on or off. These signals are combined at a very rapid rate to describe numbers. The two-signal limitation mandates a numbering system that can only work with two signals; and the system used is called a binary system, which is based on powers of two. If we wanted to express the number 25, for example, the binary arrangement would be 0 0 1 1 0 0 1. This is explained below:

Line #1	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Line #2	64	32	16	8	4	2	1
Line #3	0	0	1	1	0	0	1

Line #1 notes the power to which 2 is raised. Line #2 notes the corresponding

numerical value. Two to the one power, for example, is 2; two to the two power is 4; two to the four power is 16, etc. We need not stop at 2^6 as shown but could continue to higher numbers, depending on the computer. The number 25 is described by setting signals, or circuits, in the computer, noted by 1's and 0's in Line #3, that correspond to 1 + 8 + 16 (reading from right to left), which equals 25. By the same reasoning the number 72 would be expressed as 1 0 0 1 0 0 0. The binary system for a computer works very much like the binary system for the holes on a numerical control tape, only in this case the number can be larger and extend beyond the number 8, as shown on Line #2 above. As on a tape each 1 and 0 is called a bit. There are 8 bits to a byte and usually 1, 2, or 4 bytes make up a word.

A computer may be addressed in this binary, or so-called computer machine, language; however, it is awkward to do so and assemblers and compilers have been developed. An *assembler* is a computer program that converts letters and numbers to the binary code that the computer understands. Therefore, instead of writing 1 0 0 1 0 0 0 for the number 72, we need only write 72. The assembler converts this number to 1 0 0 1 0 0 0, which the computer understands. In the same way the computer also converts letters of the alphabet to binary notations. A *compiler* goes quite a bit further and allows the programmer to address the computer in mathematical notation. For example, the programmer could write 2 + 3 + 4 = X. The compiler, in this case a FORTRAN compiler, would convert the numbers to binary notation and perform the calculation. Both APT and COMPACT II utilize a FORTRAN compiler in their part programming systems. This is illustrated in Fig. 5–4.

The two features that distinguish a computer most are its speed and its internal storage capacity. Minicomputers, costing as low as two or three thousand dollars, have calculating speeds measured in micro (millionths) seconds and have internal storage capacities ranging from 16,000 to 128,000 words. Large scale computers, costing perhaps several million dollars, have calculating speeds of nano (billionths) seconds and internal capacities ranging to several million words. Both APT and COMPACT II can be handled on a broad range

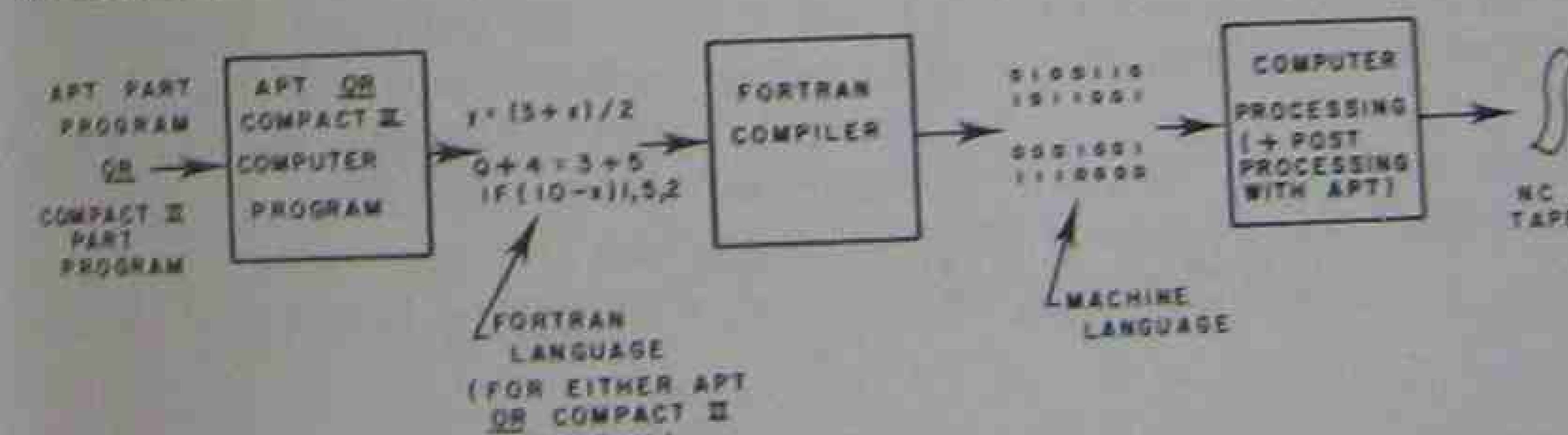


Fig. 5–4. Shown above is a rough flow pattern of the computer software programs associated with both the APT and COMPACT II part programming systems. It should be pointed out that the computer could handle either system only if the computer were primed with the proper computer program to handle the system. And then it would also depend on whether the particular computer could handle either the APT or COMPACT II system. The postprocessor, noted in the third box from the left, is a program software package that must be prepared for each machine/control-system combination. COMPACT II uses a link arrangement which is "tailored" to the particular machine/control-system combination.

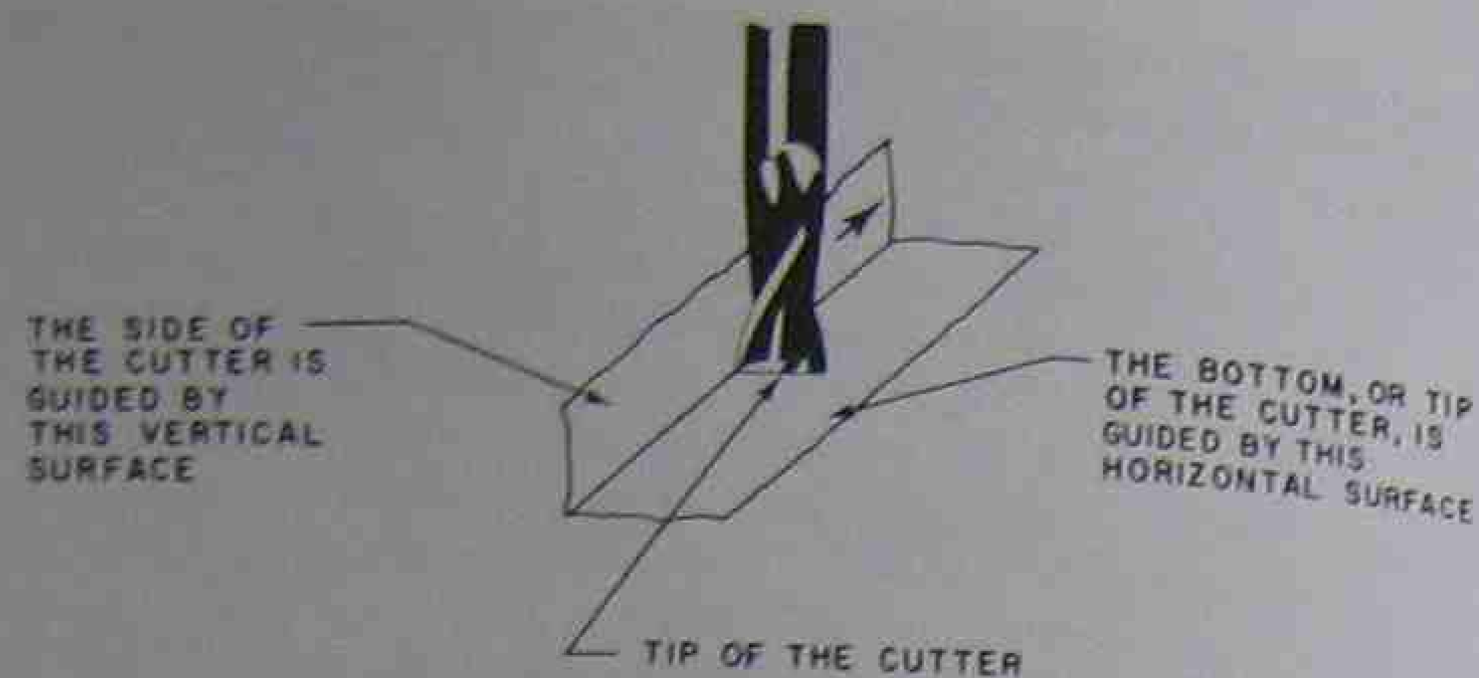


Fig. 5–5. The cutter is guided by two surfaces or planes. These may be imaginary surfaces, constructed by the part programmer to guide the cutting tool in space, or they may be the actual surfaces of the part.

of computers, from mini to large scale. However, some of the capabilities of the systems are sacrificed at the lower end of the minicomputer range.

Fundamentals of APT

MOVING THE CUTTER

APT part programming is based on a three-dimensional concept; that is, the computer "looks at" the movement of the cutter (presumably a milling cutter) as though it were being guided in space by two surfaces or planes, usually normal or perpendicular to each other. In a three-axis move the cutter would move in the *X*, *Y*, and *Z* axes simultaneously while being guided by the two surfaces.[4] (See Fig. 5–5.) Even though the cutter may be moving in two dimensions (*X*- and *Y*-axis movements only) the computer still maintains the three-dimensional space concept.

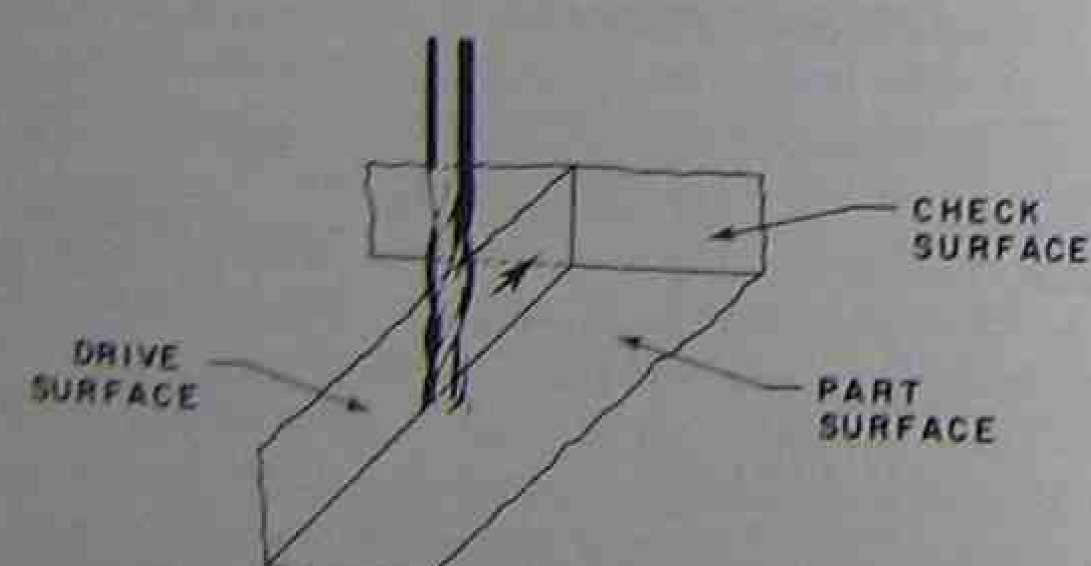


Fig. 5–6. The path of the cutter is guided by the *part* surface and the *drive* surface. To check the motion a third surface is used, and this is appropriately called the *check* surface.

[4] APT is also well suited for handling four- and five-axis motions. A four-axis motion would consist of three linear and one rotary motions and a five-axis motion would consist of three linear plus two rotary motions. All motions would be simultaneous.

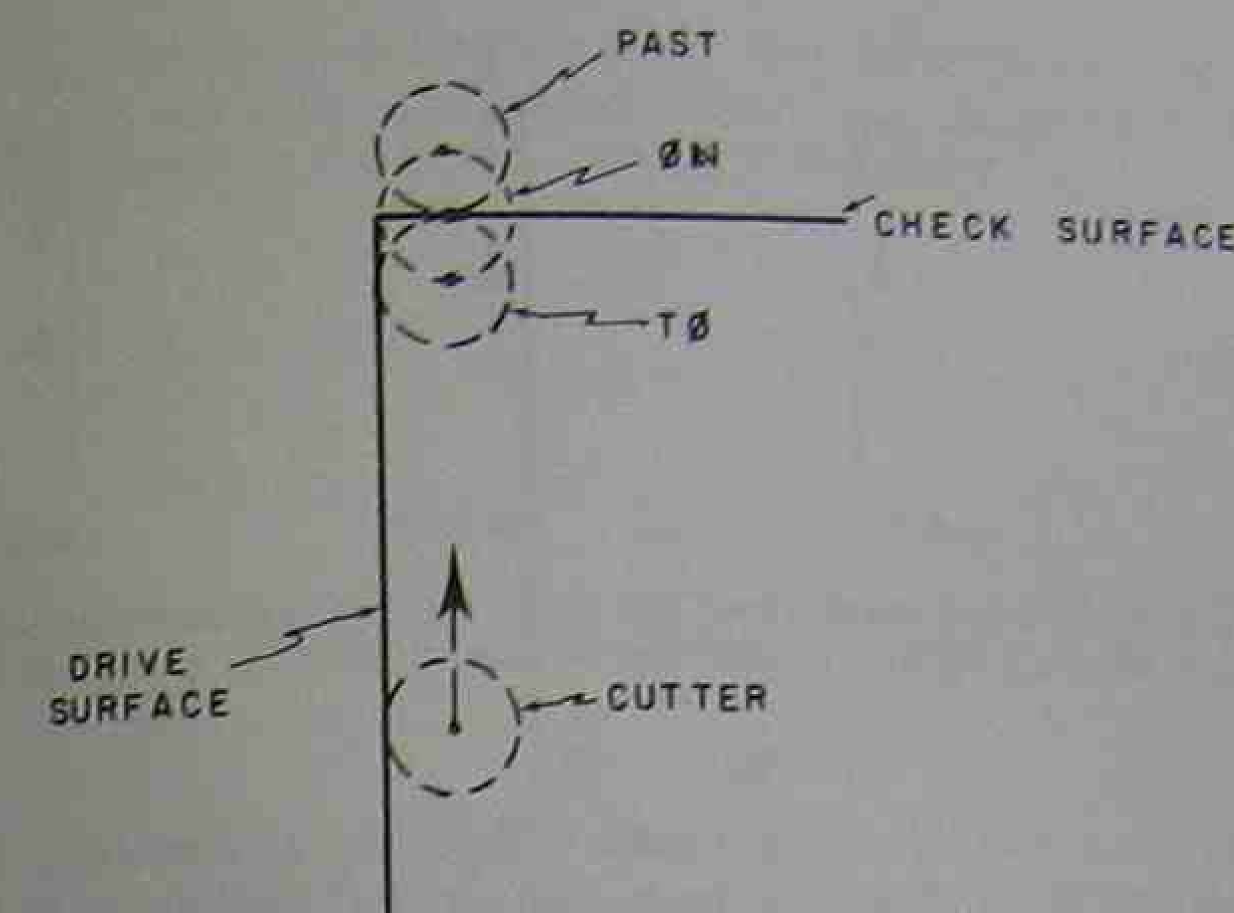
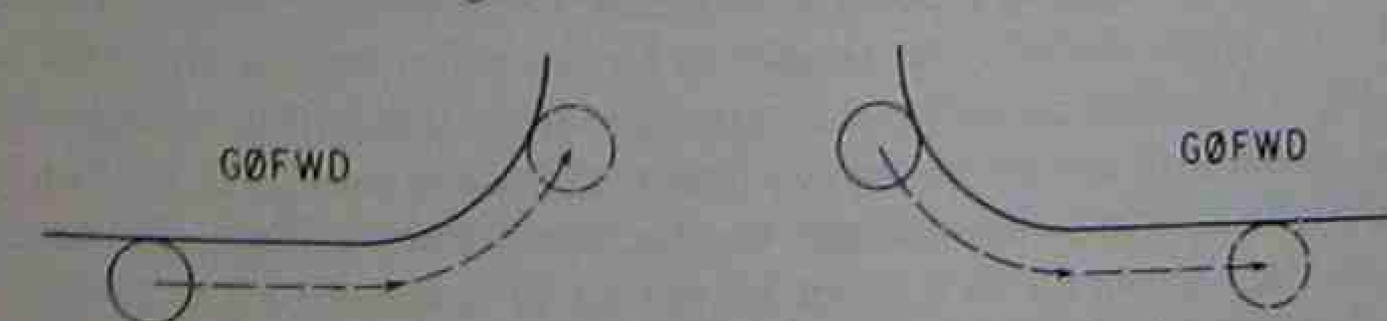


Fig. 5–7. The cutter may be directed to move TØ the check surface, ØN the check surface or PAST the check surface, as shown above. In order for the cutter to move TØ or PAST the check surface the computer would have to "know" the diameter, or radius, of the cutter. This is accomplished by the statement in the program CUTTER/1, for a cutter of 1-inch diameter which would precede the statement instructing the cutter to move TØ or PAST the check surface. It will be noted that, when looking down on the *drive surface* and the *check surface*, they appear as lines.

The cutter will move along the two surfaces until it is redirected, or stopped by a third surface. These three surfaces are defined as the *drive* surface, the *part* surface, and the *check* surface. (See Fig. 5–6.) The cutter may be directed to move either TØ, ØN,[5] or PAST the check surface as shown in Fig. 5–7. Once the cutter is at the check surface it may be directed to GØLFT, GØRGT, GØFWD, or GØBACK. Directing the cutter is like steering a car, and the programmer would determine which instruction he should use by looking in the direction that the cutter is traveling. (See Fig. 5–8.)

Where a cutter, or machine, moves from a straight line onto a circular section, or vice versa, the instruction GØFWD is used. In this instance, the straight line would be tangent to the circle; for example:



[5] It is customary practice in writing a computer program to place a slash (/) through the letter "O" in order to distinguish it from the numeral zero. Similarly, a horizontal line is applied to the letter "Z," which appears as "Ƶ," in order that there be no mistaking it for the numeral "2."

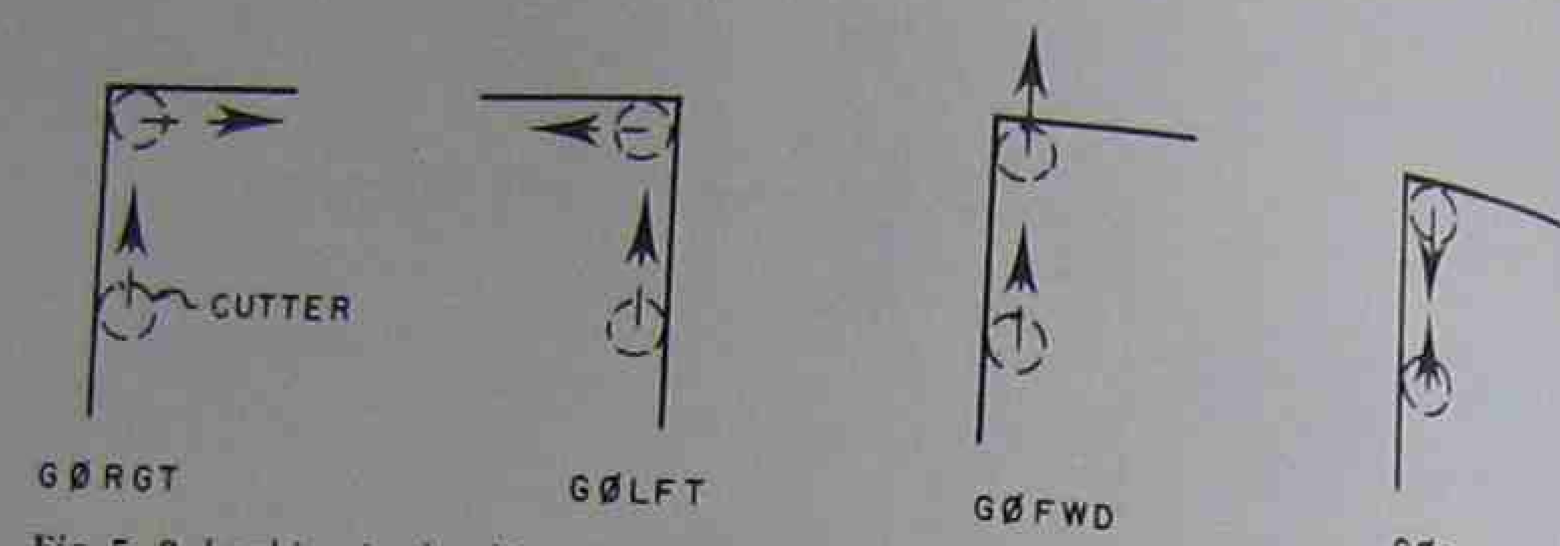
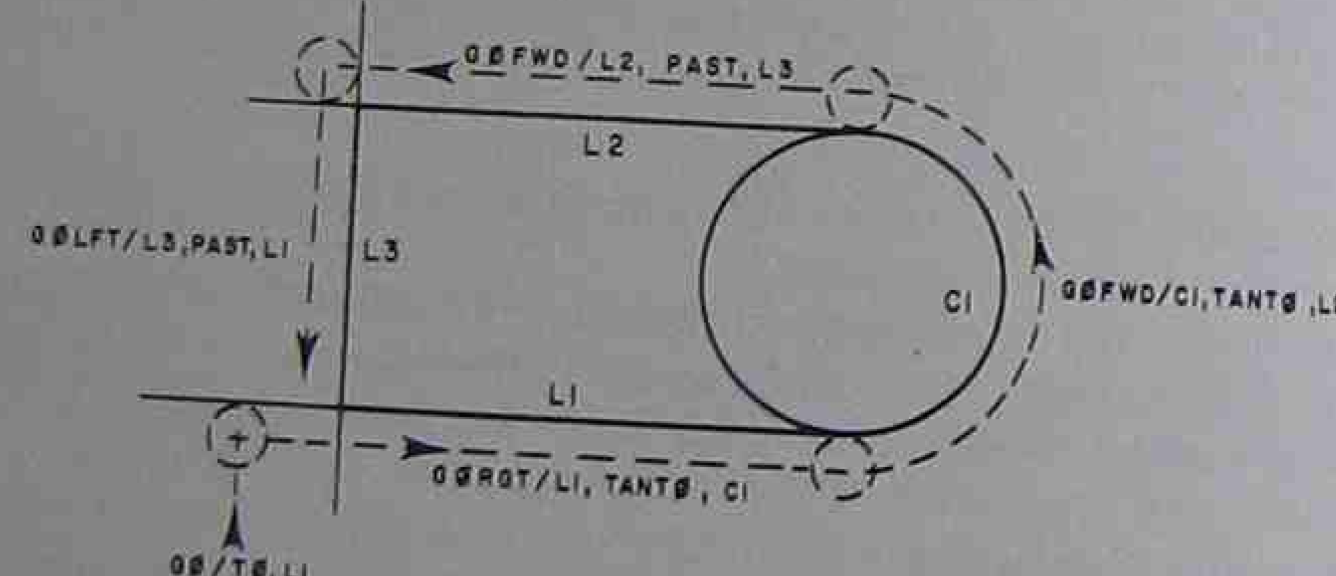


Fig. 5–8. Looking in the direction of travel, the cutter may be instructed to make a right turn (GØRGT), a left turn (GØLFT), continue forward (GØFWD), or reverse direction (GØBACK).

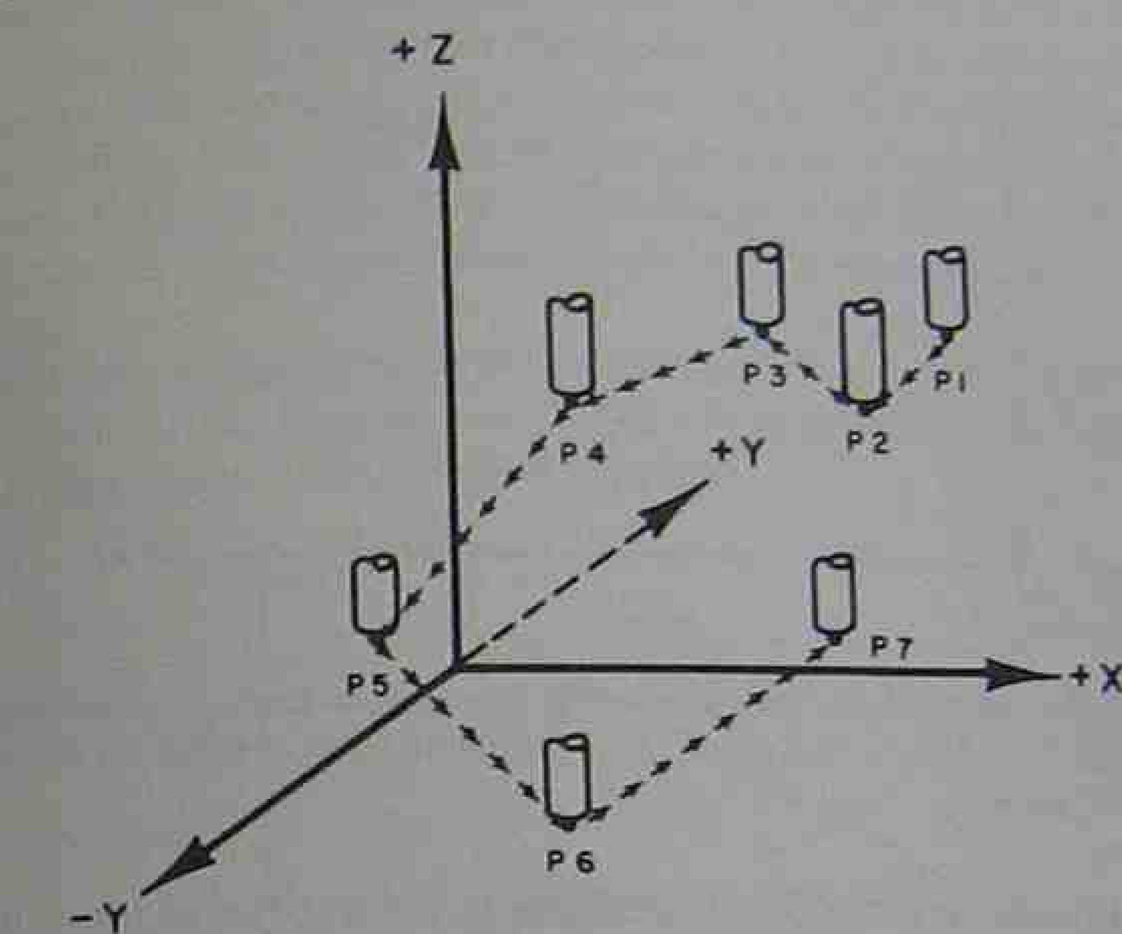
In order to move a cutter along a path with changing directions the programmer must list instructions for each change of direction. In the sketch shown below the cutter is required to make a right turn along the line that is labeled L1. Then to move around the arc C1 and then along the line L2. The cutter is then to pass line L3 and make a left turn and move along L3 until past L1.



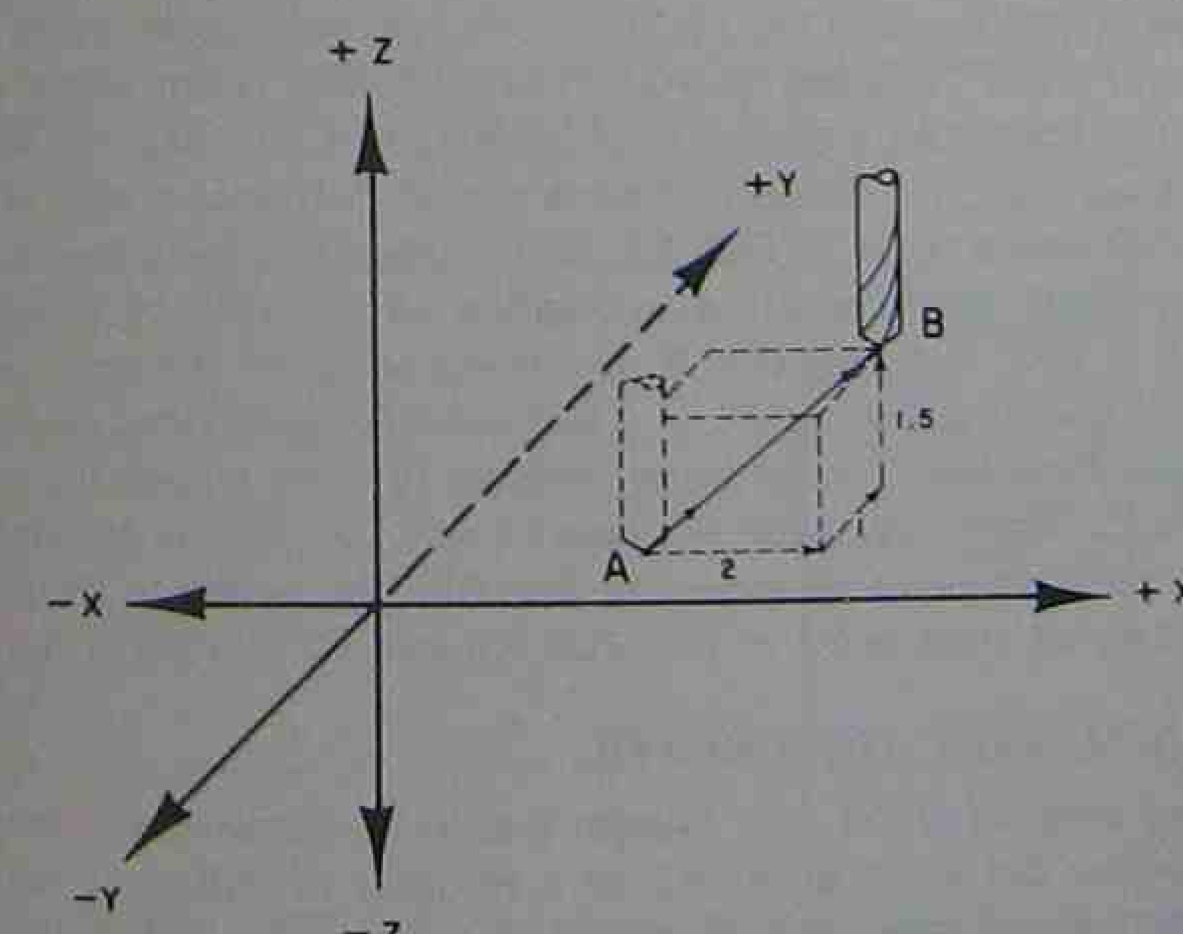
The APT motion statements are shown along the path of travel. When the cutter is moved along L1 and then along C1, the "TØ," "ØN," or "PAST" condition could not apply since L1 is *tangent* to C1 and the motion statement is therefore GØRGT/L1,TANTØ,C1, which means "go right along a line which is labeled L1 until reaching a point where line L1 is tangent to a circle which is labeled C1." Also, since C1 is tangent to L1 the command is GØFWD rather than GØLFT. The same consideration applies when the cutter reaches the top part of the circle and moves along the line that is labeled L2. It will be noted that a slash (/) separates the general instruction GØRGT from the details of *where* to go, i.e., along L1 to a point where L1 is tangent to C1. Also, it will be noted that the words are separated by commas and that there is no punctuation at the end of the statement.

In drilling and other point-to-point operations, such as boring and tapping, the cutter need not be guided by surfaces as used with contour milling operations. It is sufficient to either direct the cutter to a specific point or describe

the incremental distance that the cutter is to move. In the sketch below the cutter may be moved from the point labeled P1, to P2, to P3, to P4, etc. by the statements GØTØ/P1, GØTØ/P2, GØTØ/P3, GØTØ/P4, etc.



Movement of the cutter an *incremental* distance is illustrated in the sketch below. In moving from point A to point B the cutter moves 2 inches in the $+X$ direction, 1 inch in the $+Y$ direction, and 1.5 inches in the $+Z$ direction. The APT statement for moving from A to B would be GØDLTA/2,1,1.5



APT GEOMETRY STATEMENTS

Before a cutter can be guided around the geometry of a part or directed from one point to another it is necessary that the geometry elements, such as lines,

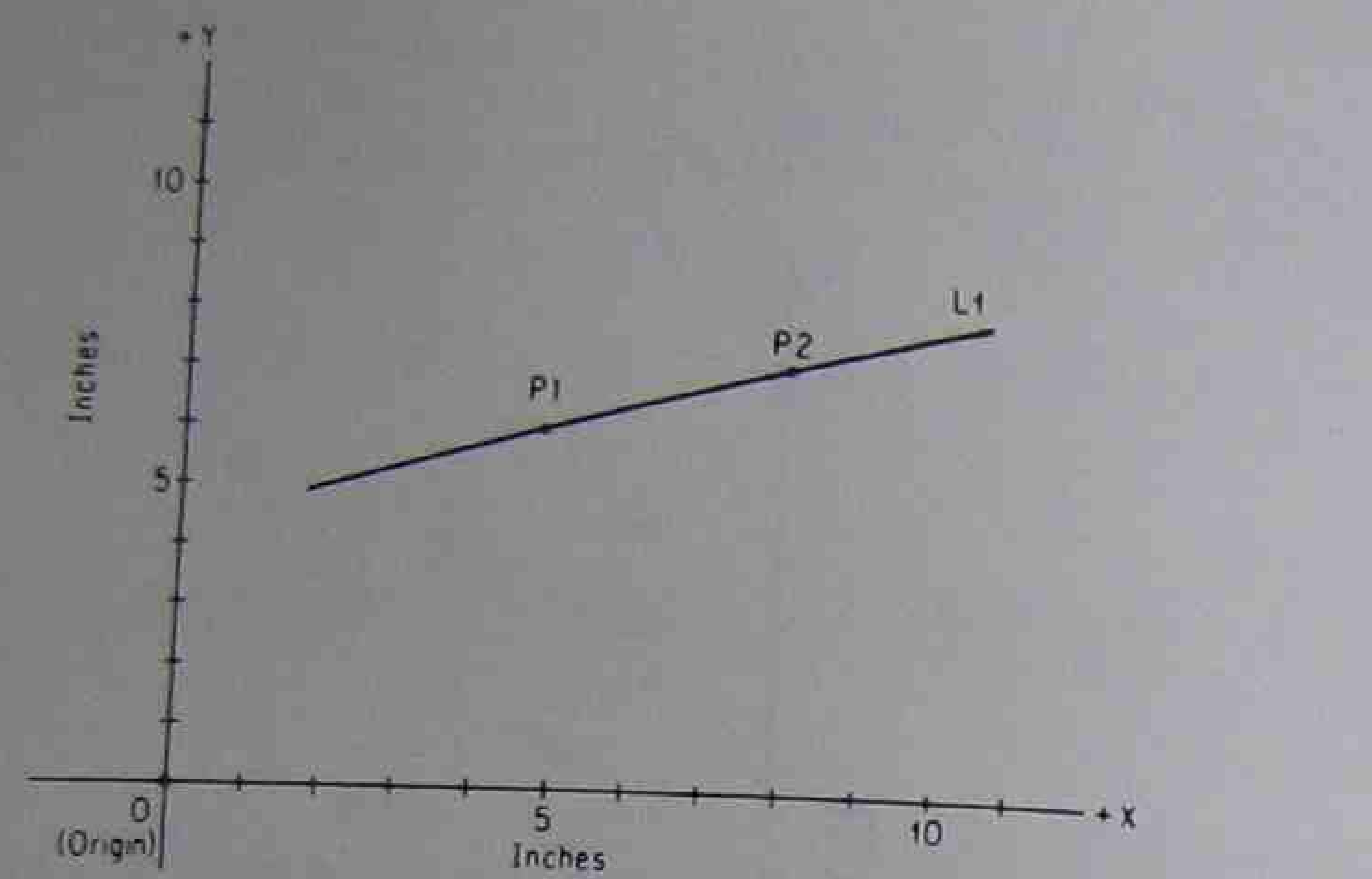


Fig. 5–9. P1 and P2 are symbols which identify two points having coordinates $X = 5$; $Y = 6$; $Z = 0$; and $X = 8$; $Y = 7$; $Z = 0$. L1 is the identifying symbol for a line running through these two points.

circles, and points, be described. It is fruitless to tell a cutter to move to a point noted as P1 unless the computer knows the precise location of P1. P1 is a *symbol* and must first be described before it can be used in a motion statement such as GØTØ/P1. A *symbol* can be any combination of letters and numbers as long as there are no more than six figures and at least one figure is a letter. CAT1 would be a legitimate symbol. MØUSE2 would also be a legitimate symbol. BUNGALØW4 would not be. It would, however, make more sense to label a point as P1, or PT1, rather than CAT1 or MØUSE2. A line might logically be labeled L1 or LN1. And a circle could be labeled C1 or CI1 or CIR1. After the symbol is assigned, the identification is noted. A line is noted as LINE; a point as PØINT; and a circle as CIRCLE. Every effort has been made to make the computer input words as close to English as possible. A point having coordinates $X = 5$, $Y = 6$, and $Z = 0$, and being assigned the symbol P1 (see Fig. 5–9) would be described as P1 = PØINT/5,6,0.[6] Referring, again, to Fig. 5–9, a line could be described as going through the two points P1 and P2 as L1 = LINE/P1,P2. Before using the symbols P1 and P2 in order to describe the line L1 it would have been necessary to describe the symbols P1 and P2 in geometry statements such as P1 = PØINT/5,6,0 and P2 = PØINT/8,7,0.

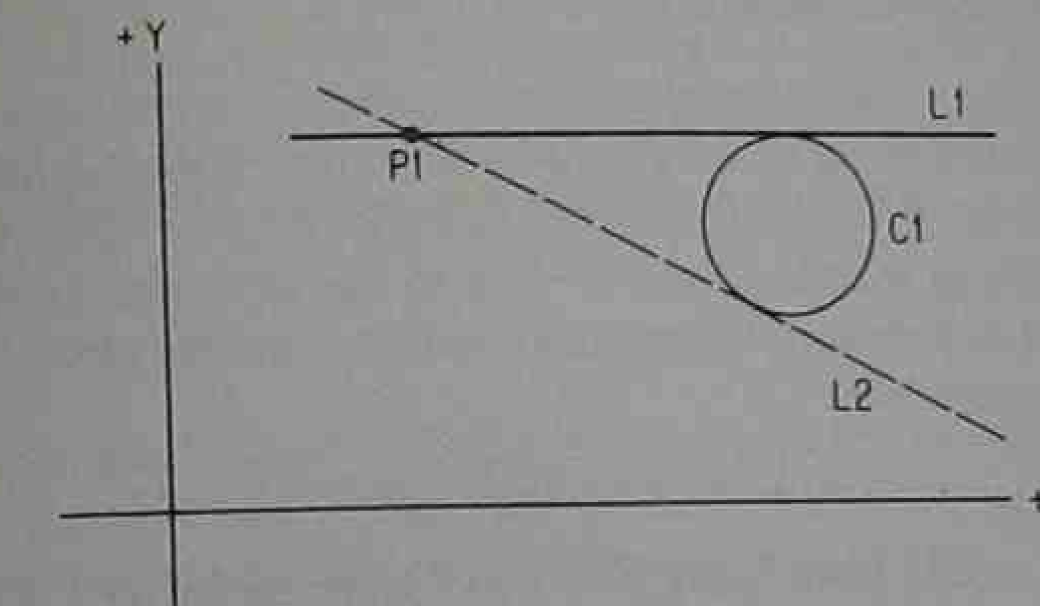
MORE APT GEOMETRY STATEMENTS

One of the key benefits of the APT system lies in its versatility, wherein it is possible to define the same geometry in a number of different ways. This relieves the part programmer of a good deal of arithmetic since he may select

[6] A slash mark (/) separates the description of the geometry (PØINT) from factors relating to it, such as the coordinates $X = 5$, $Y = 6$, $Z = 0$. Commas separate the values of the coordinates.

the most suitable and most readily obtainable definitions that fit the particular blueprint dimensioning scheme. With certain schemes it would be almost impossible to hold to a single definition, such as defining a point only by its coordinates. A point may, therefore, also be described as the intersection of two lines (see Fig. 5–10, left), or the intersection of two circles (see Fig. 5–10, right). In the latter instance, there are two intersection points and they may be distinguished by describing one of them as being larger or smaller in the x or y direction, than the other. It is not necessary that the coordinates of P1 and P2 be known in order to use these symbols in subsequent geometry or motion statements.

As with the description of the point there are a number of different ways of describing a *line*. One way is to have it pass through a defined point and also be tangent to a circle, as shown below:



In this instance, there are two lines that could meet this requirement (namely,

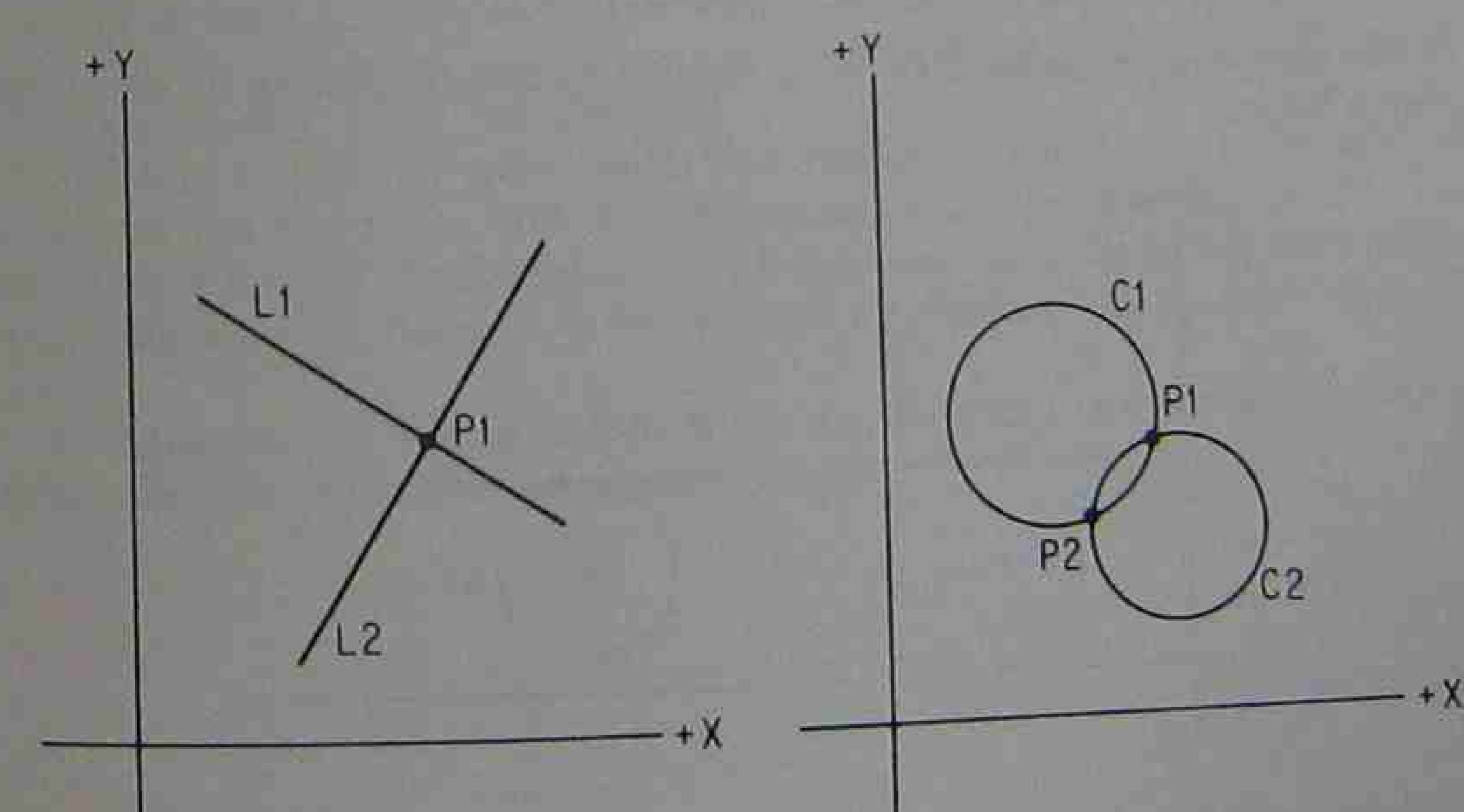


Fig. 5–10. (*Left*) The APT statement describing the point where the two lines intersect would be: P1 = PØINT/INTØF,L1,L2. (*Right*) P1 would be larger in the X direction than P2. That is, its x ordinate would be larger. P1 would also be larger in the Y direction. P1 could therefore be described as either: P1 = PØINT/XLARGE,INTØF,C1,C2; or P1 = PØINT/YLARGE,INTØF,C1,C2. Both C1 and C2 would have had to have been defined previously, in APT statements.

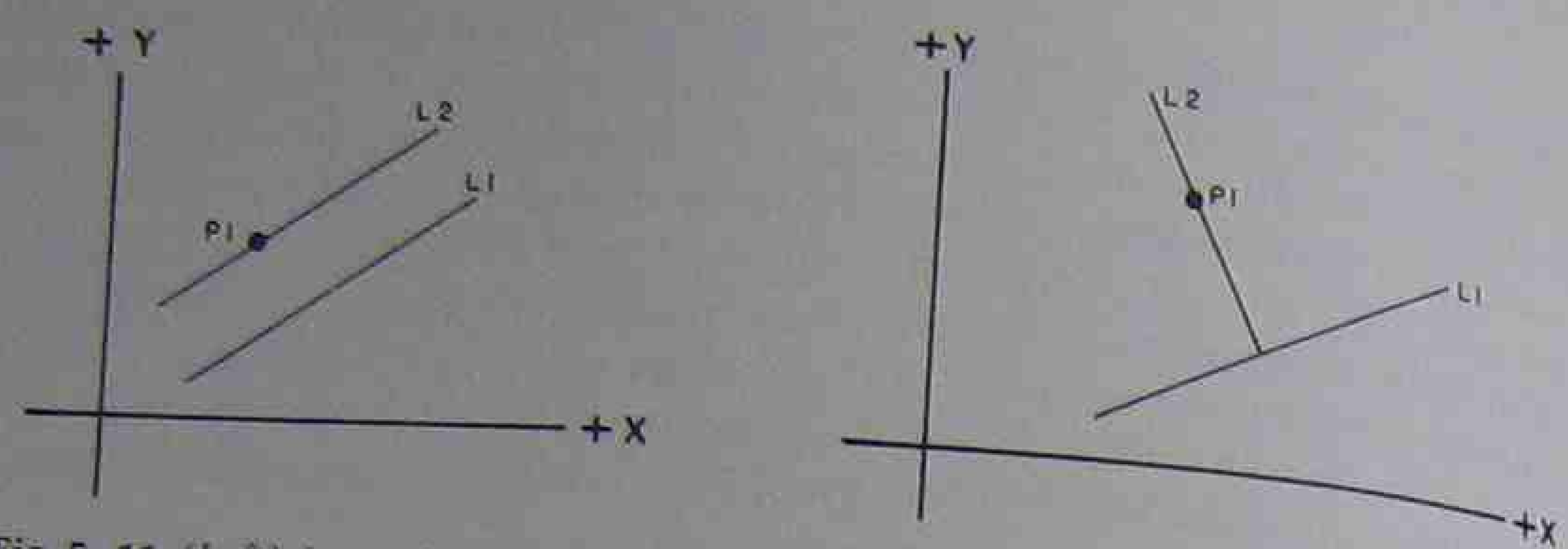


Fig. 5–11. (*Left*) Assuming L2 is parallel to L1 and passes through P1, the APT statement describing L2 would be: L2 = LINE/P1,PARLEL,L1. (*Right*) Assuming L2 is perpendicular to L1 and passes through P1, the APT statement describing L2 would be: L2 = LINE/P1,PERPTØ,L1. In both cases, L1 and P1 would have had to have been described in an APT statement, previously, in the program.

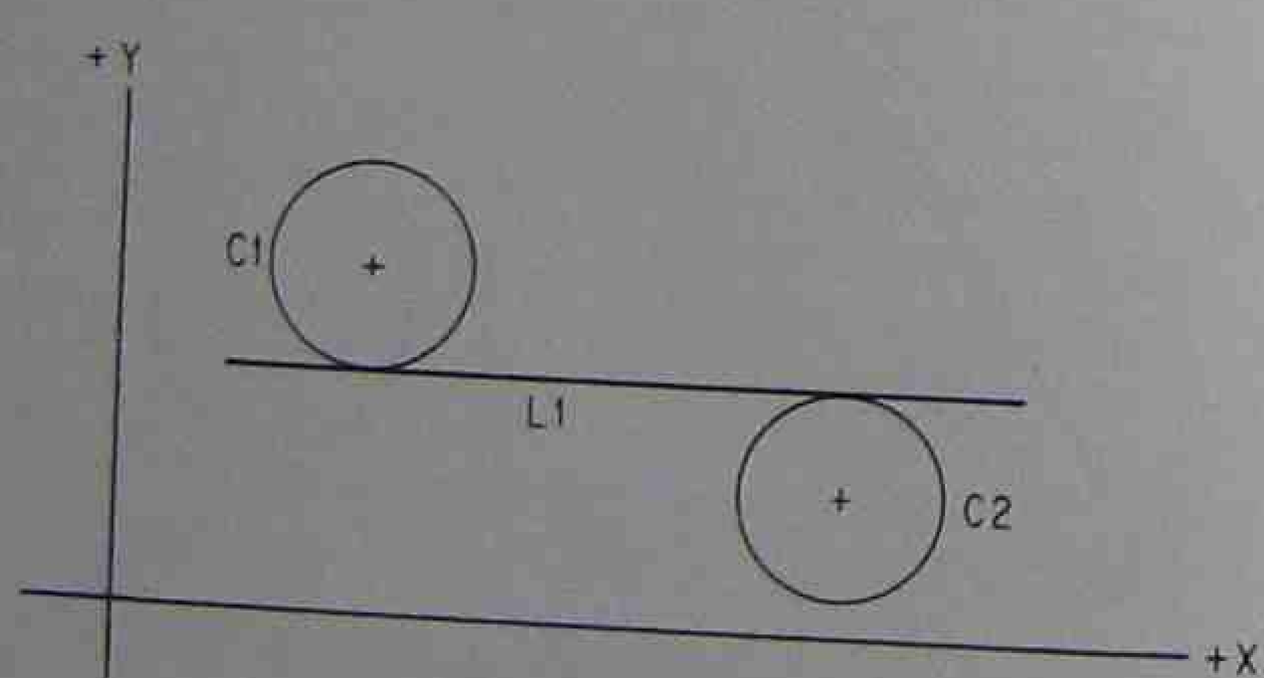
L1 *and* L2), and there must be some way, therefore, of distinguishing between the two and specifying the line desired. Clarification of this ambiguity is accomplished by noting whether the line lies to the *left* or *right* of the center of the circle, looking in a direction from the *point toward the circle*. The geometry statement for L1 would be:

L1 = LINE/P1,LEFT,TANTØ,C1

where P1 and C1 would have been previously defined in APT statements. The statement for L2 would be:

L2 = LINE/P1,RIGHT,TANTØ,C1

A line may also be defined as being tangent to two circles, as shown in the sketch below:



L1 would be defined as a line tangent and to the right of the center of the circle C1 and tangent and to the left of the center of circle C2—*looking from circle C1 to circle C2*. The APT statement would be as follows:

L1 = LINE/RIGHT,TANTØ,C1,LEFT,TANTØ,C2

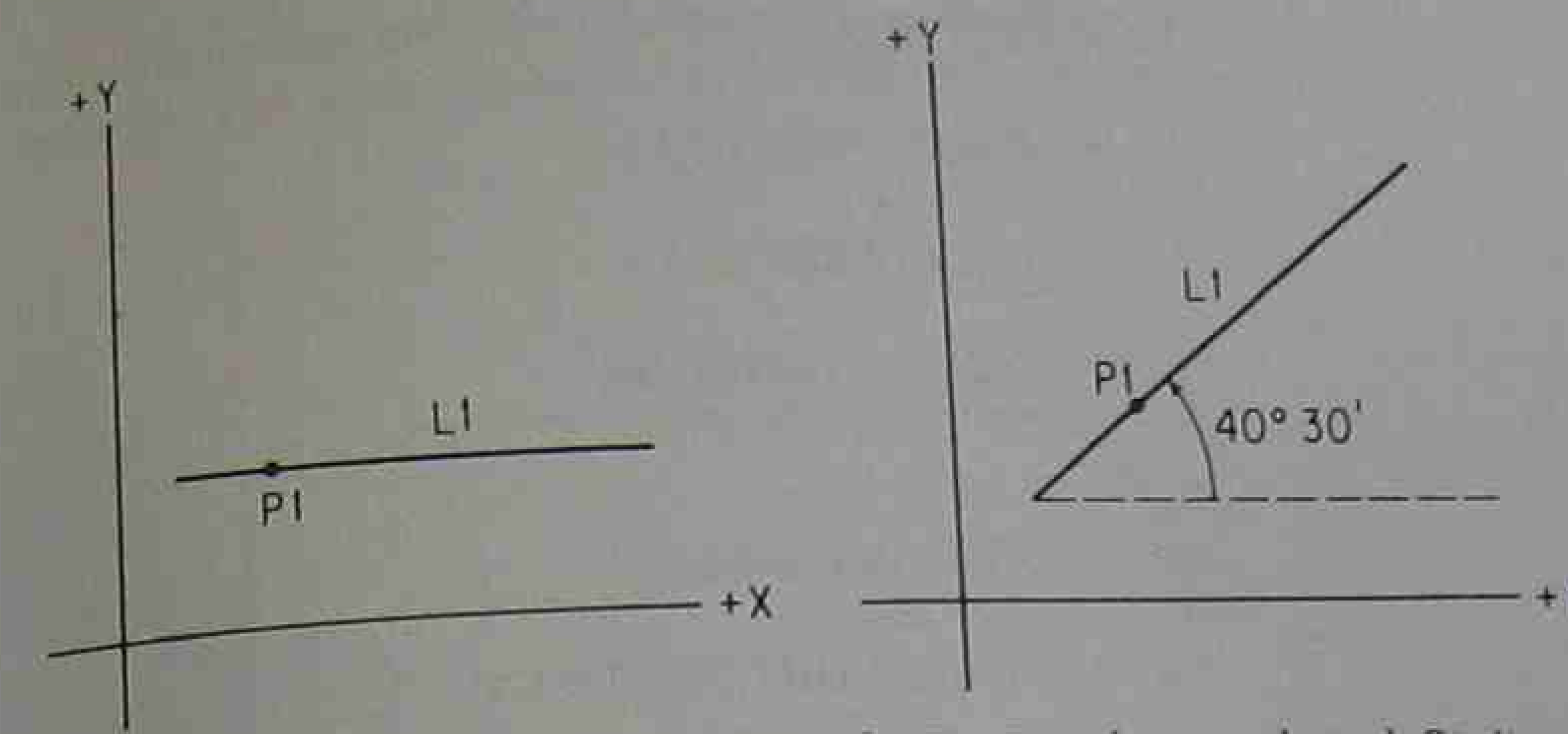


Fig. 5–12. (*Left*) Assuming L1 is parallel to the X axis and passes through P1, it may be defined as: L1 = LINE/P1,ATANGL,0. (*Right*) If L1 is at a +40°30′ angle with the X axis (positive angle measured in a counterclockwise direction), and it passes through P1, then it may be defined as: L1 = LINE/P1,ATANGL,40.5. Degrees are not shown, and minutes and seconds are noted as a decimal part of a degree—hence 30′ = .5 degrees.

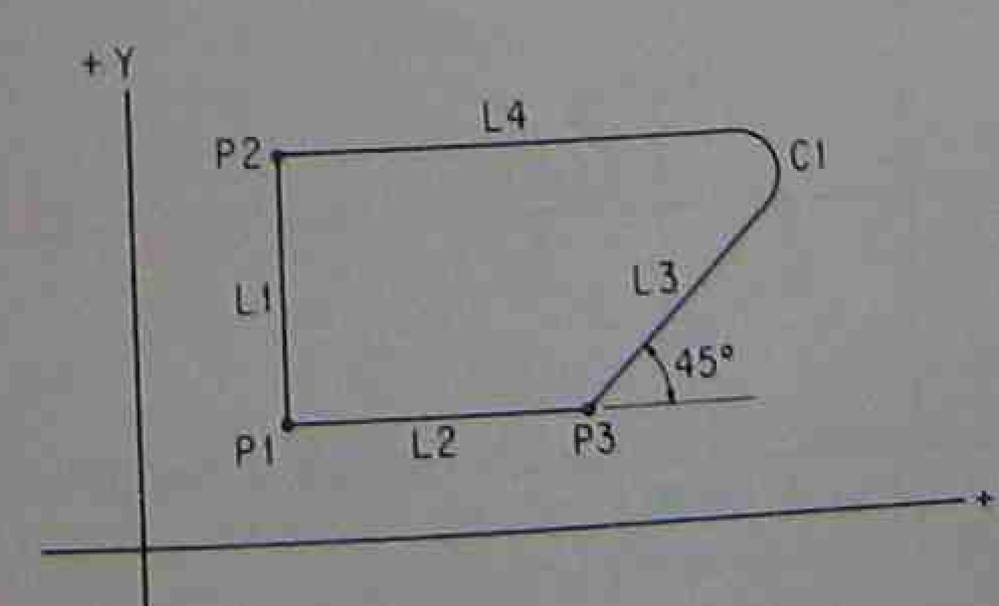
or the line could be defined as being tangent and to the right of the center of circle C2 and tangent and to the left of circle C1—*looking from circle C2 to circle C1*. In this instance, the APT statement would be:

L1 = LINE/RIGHT,TANTØ,C2,LEFT,TANTØ,C1

Two other ways of defining a line would be through a point and *parallel* to another line; or through a point and *perpendicular* to another line. Examples are shown in Fig. 5–11, left and right.

Probably one of the most useful line statements involves a point and the angle that the line forms with the horizontal, or X axis. (See Fig. 5–12.) Horizontal and vertical lines, common with engineering designs, may be conveniently expressed as being at 0 degrees and 90 degrees, respectively, with the X axis.

Considering the sketch below, there are, therefore, a number of ways in which the line segments may be described:



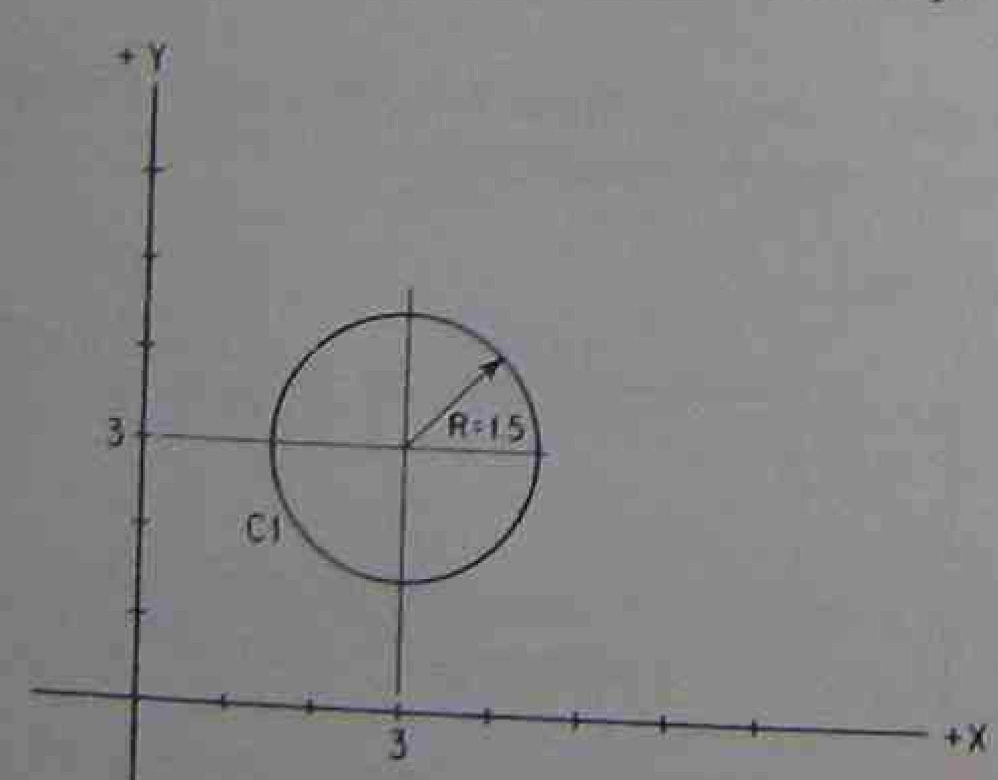
L1 = LINE/P1,P2
or
L1 = LINE/P2,PERPTØ,L2
or
L1 = LINE/P1,PERPTØ,L4
or
L1 = LINE/P1,ATANGL,90

L3 = LINE/P3,ATANGL,45
or
L3 = LINE/P3,RIGHT,TANTØ,C1

L2 = LINE/P1,ATANGL,0
or
L2 = LINE/P1,PARLEL,L4
or
L2 = LINE/P1,PERPTØ,L1

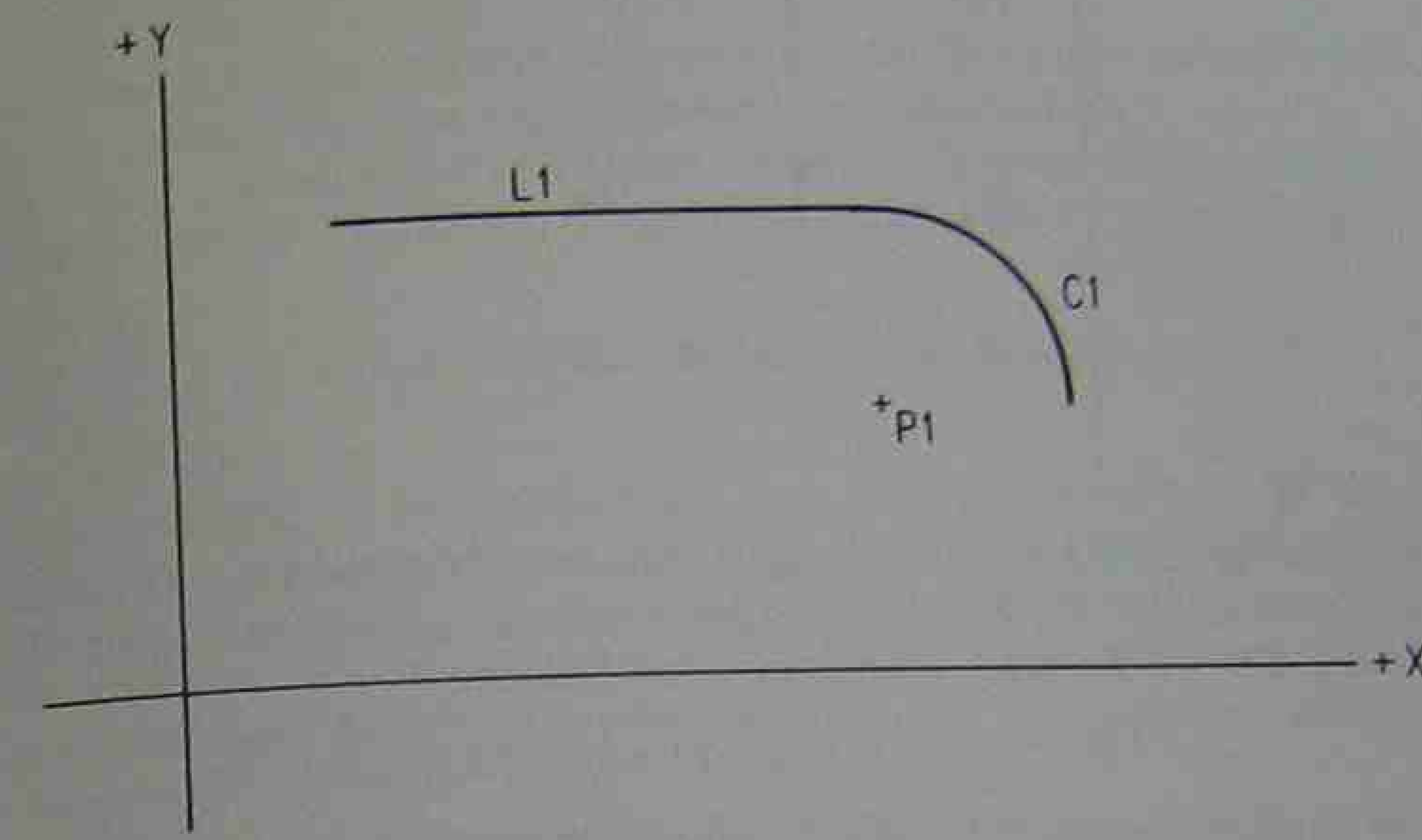
L4 = LINE/P2,ATANGL,0
or
L4 = LINE/P2,PERPTØ,L1
or
L4 = LINE/P2,PARLEL,L2
or
L4 = LINE/P2,LEFT,TANTØ,C1

The most direct and common way of describing a *circle* is to specify the center by *x* and *y* coordinates and the radius. An example is shown below:



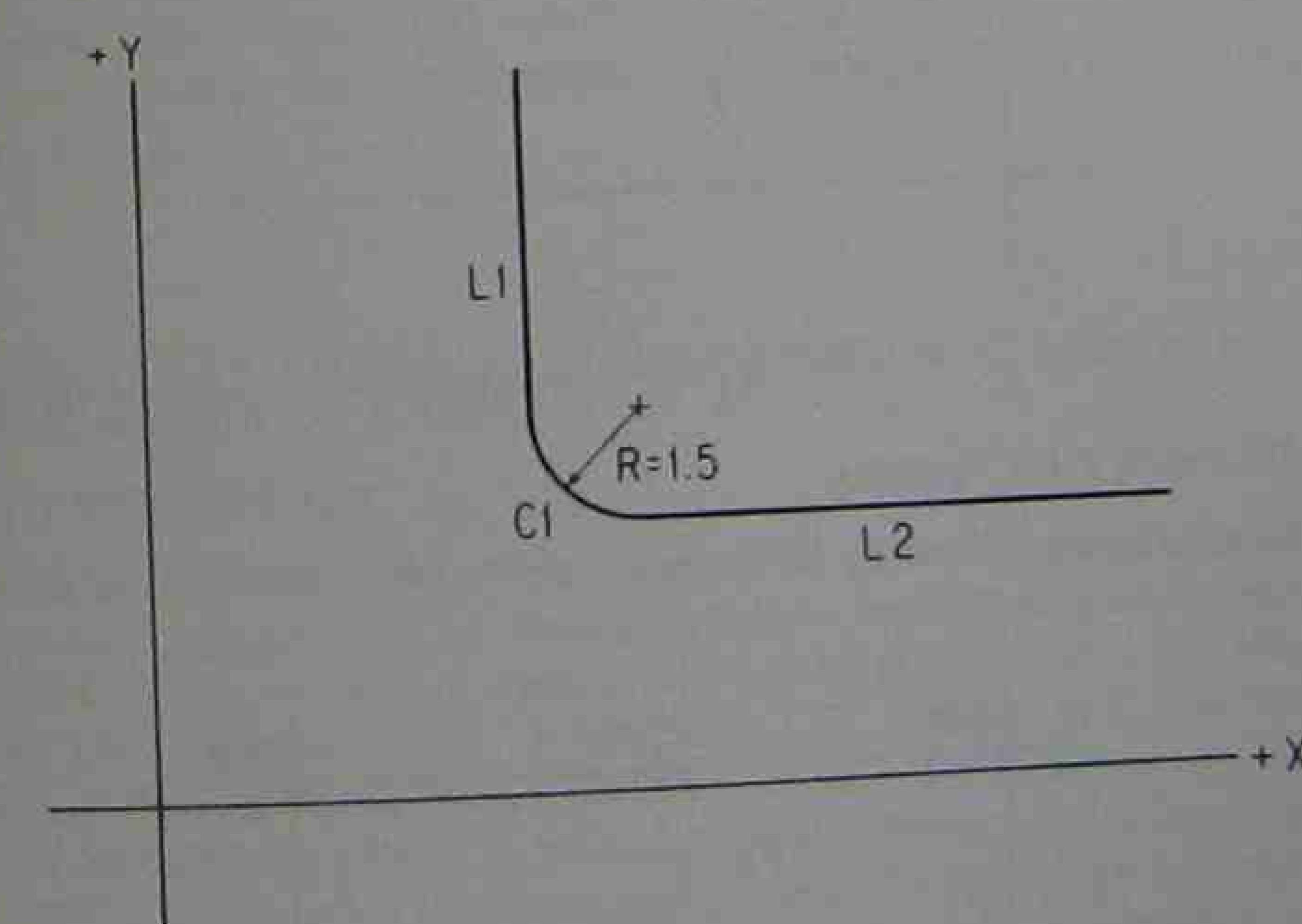
C1 = CIRCLE/3,3,1.5

At least two other methods for describing a circle are available. One specifies the center, and a line tangent to the circumference. See below:

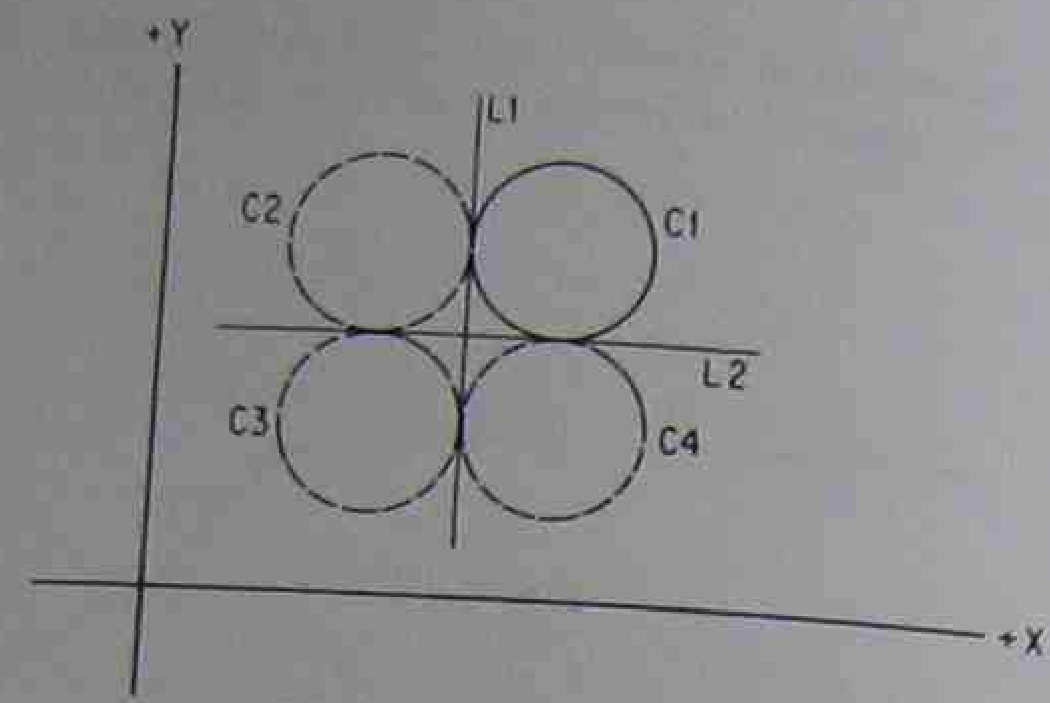


C1 = CIRCLE/CENTER,P1,TANTØ,L1

Another, and extremely convenient, method allows the programmer to describe a circle as being tangent to two nonparallel lines. The center, in this case, need not be specified. An example is shown below:



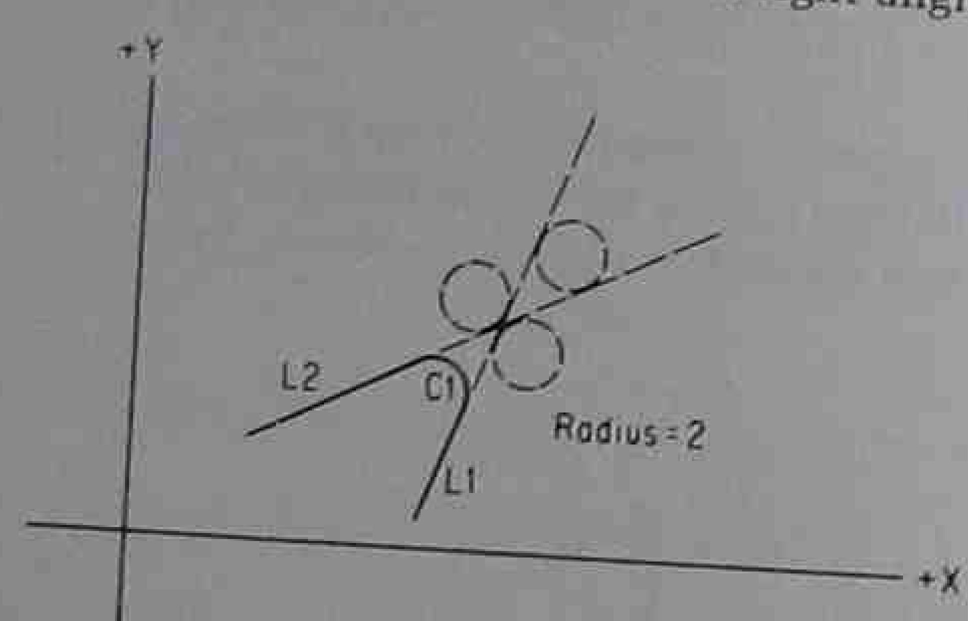
In reality, there are *four* possible circles that are tangent to L1 and L2 above, and it is necessary, therefore, to eliminate three. If circle C1 is the one that is to be described, then circles C2, C3, and C4 as shown in the following sketch, require elimination.



By denoting that C1 is on the larger *X* side of L1, C2 and C3 are eliminated, and by denoting that C1 is on the larger *Y* side of L2, C4, and again, C3 are eliminated. The APT statement would be as follows:

C1 = CIRCLE/XLARGE,L1,YLARGE,L2,RADIUS,1.5

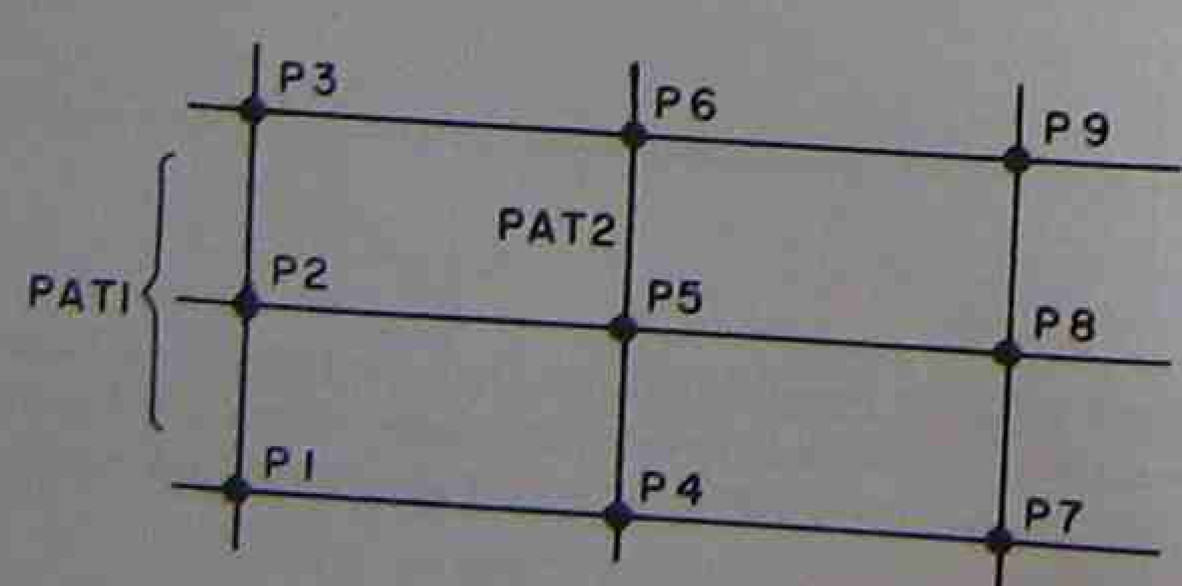
Another example in which L1 and L2 are not at right angles is shown below:



In this instance C1 = CIRCLE/XSMALL,L1,YSMALL,L2,RADIUS,2

POINT-TO-POINT PATTERNS

These may also be described in APT. Consider the pattern below, in which the rows of points are parallel:



PAT 1 is comprised of P1, P2, and P3 which fall in a line and are equally spaced. The APT statement would be:

PAT 1 = PATERN/LINEAR,P1,P3,3

meaning that PAT 1 is a linear pattern with the first point being P1 and the last being P3, and having a total of 3 equally spaced points.

Next PAT 1 is repeated two times, at equal spacing and in parallel lines in order to form PAT 2. PAT 2 would then include PAT 1 in its statement, as follows:

PAT 2 = PATERN/PARLEL,PAT1,V3,3

where:

V3 is a vector, or line, describing, essentially, the distance of the lines parallel to the line formed by P1, P2, and P3. The last numeral "3" denotes that PAT 1 is repeated twice for a total of three patterns.

The motion statement would be GØTØ/PAT 2 and would result in moving the machine about PAT 2 in accordance with the sequence in which PAT 2 was described.

Prior to the motion statement a CYCLE statement would be described that would perform an operation, such as drilling to a specific depth, at each of the points in the pattern. A drill cycle statement might appear as follows:

CYCLE/DRILL,RAPTØ,.1,FEDTØ,1,RTRCTØ,.1,IPM,10

which would instruct the machine to move the tip of the drill to within .1 inch above the work surface at a rapid traverse; then to drill a hole 1 inch deep at 10 ipm and then to retract at rapid traverse to a location .1 inch above the surface of the work. This cycle would repeat at each of the point locations in the pattern.

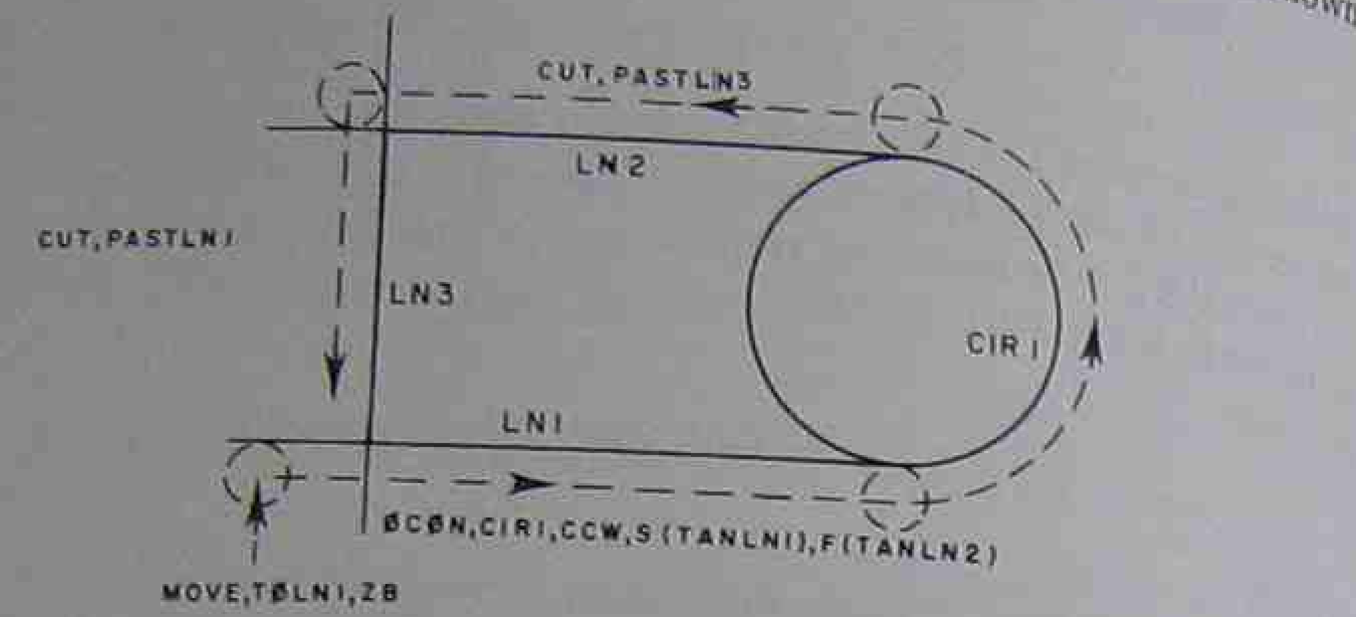
Fundamentals of COMPACT II

MOVING THE CUTTER

COMPACT II has been used primarily as a two-dimensional program, although it does have the capability for handling three-, four-, and even five-axis work.

Like APT, the words are very similar to English. A line, for example, is noted as LN. A point as PT. A circle as CIR. Unlike APT, the identifying symbols are specified. LN must be used for line. PT must be used for point. CIR must be used for circle. A move at rapid traverse would be noted by the word MOVE. If the cutter were removing metal, the instruction would be CUT.

A cutter, presumably a milling cutter, may be moved to a line, on a line, or *past* a line. The instruction for moving *to* a line when cutting metal is CUT,TØLN__. For moving on a line the instruction is CUT,ØNLN__. And for moving *past* a line it is CUT,PASTLN__. (See Fig. 5–13.) CUT statements are also used for moving the cutter about the geometry. In the illustration below the cutter moves to the line LN1 and then around the perimeter as was shown earlier in this chapter with the APT example.



The cutter starts out with a rapid traverse move to LN1 and to a height where Z is zero, noted by ZB. The statement for this move is MØVE,TØLN1,ZB. The cutter is then instructed to cut along LN1 and then contour on the outside of the circle CIR1 in a counter-clockwise direction, starting at the point where CIR1 is tangent to LN1 and finishing where CIR1 is tangent to LN2. All this is expressed in the statement ØCØN,CIR1,CCW,S(TANLN1),F(TANLN2). Next the cutter is instructed to move along LN2 and past LN3 by the statement CUT,PASTLN3. And finally the cutter is instructed to move along LN3 and past LN1 via the statement CUT,PASTLN1.

A cutter may be moved from one point to another by either noting the points by their identifying symbols, e.g., MØVE,PT5, or by noting the incremental

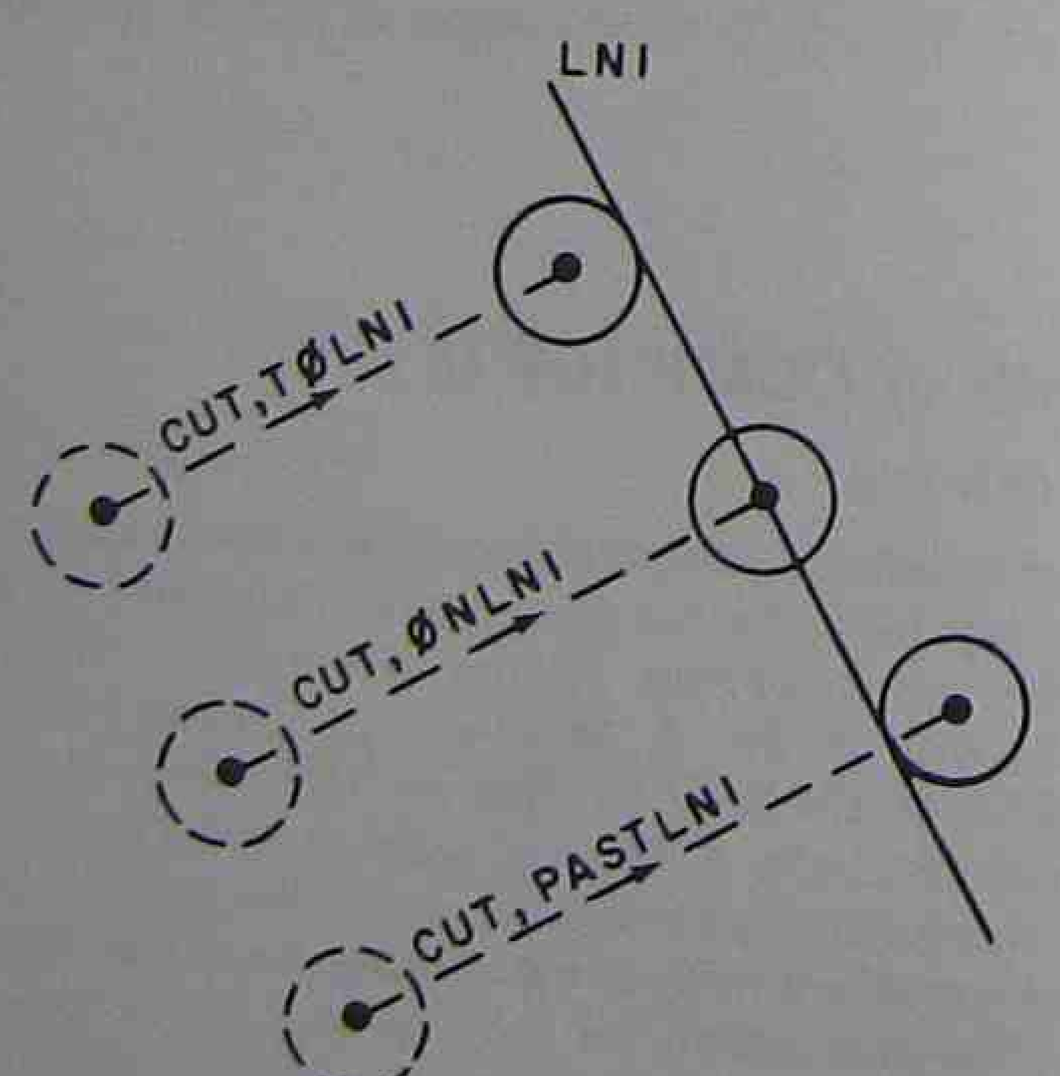


Fig. 5–13. A cutter may be brought *to, on,* or *past* a line by noting the instructions shown above. Prior to these instructions the program would have included a statement noting the diameter or radius of the cutter.

move, e.g., MØVE,3X,5Y,2Ƶ, which would mean move 3 inches in the +X direction, 5 inches in the +Y direction and 2 inches in the +Ƶ direction, from the present position.

Drilling and other point-to-point operations such as boring and tapping may be handled in one statement. For example, the statement DRL,PT2,1A would instruct the drill to move, at a rapid traverse, to a point with the identifying symbol PT2, and then move into the workpiece at a previously defined feedrate for a total distance move of 1 inch and then move out of the workpiece and back to PT2 at a rapid traverse.

COMPACT II GEOMETRY STATEMENTS

Like most computer-aided part programming systems the geometry of COMPACT II is identified by symbols. The symbol for a point is PT, which would be followed by an identifiable number, such as 1, 2, 3, etc. The statement for defining a point with an identifying number of 1 would be DPT1. If the point were defined by its X, Y, and Ƶ coordinates, the statement would be

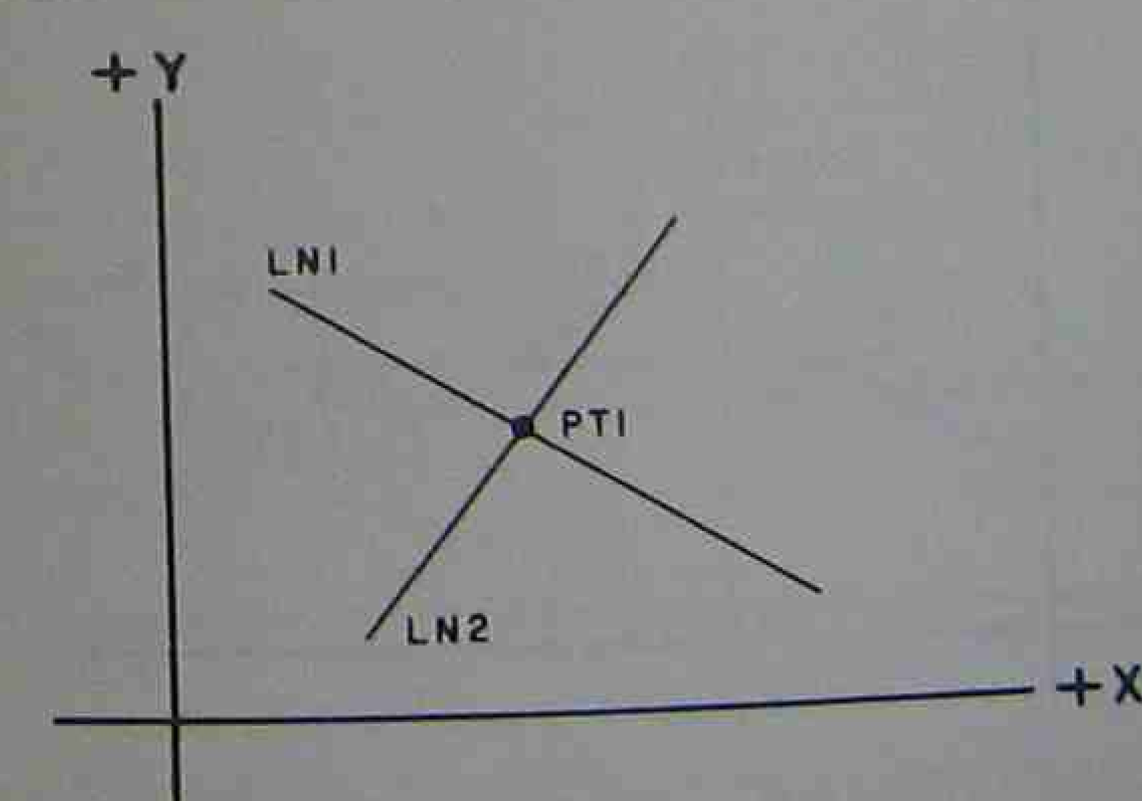
DPT1,3XB,5YB,2ƵB

This means that the point identified as PT1 would lie 3 inches in the positive X direction, 5 inches in the positive Y direction and 2 inches in the positive Ƶ direction from a set of X, Y, Ƶ axes that are called the Base, which is what the *B* stands for in the above statement. Lines and circles may be defined similarly. A line, for example, may be defined by describing the two end points of the line, e.g., DLN5,PT1,PT2. A circle may be defined by noting its center point and the radius, e.g., CIR1,PT1,2R. In all cases the identifying symbols must be described before being used in a statement. Thus PT1 would have to be described before being used in the statement CIR1,PT1,2R. This general rule holds true for almost any system.

MORE COMPACT II GEOMETRY STATEMENTS

Two other ways of defining a point are by

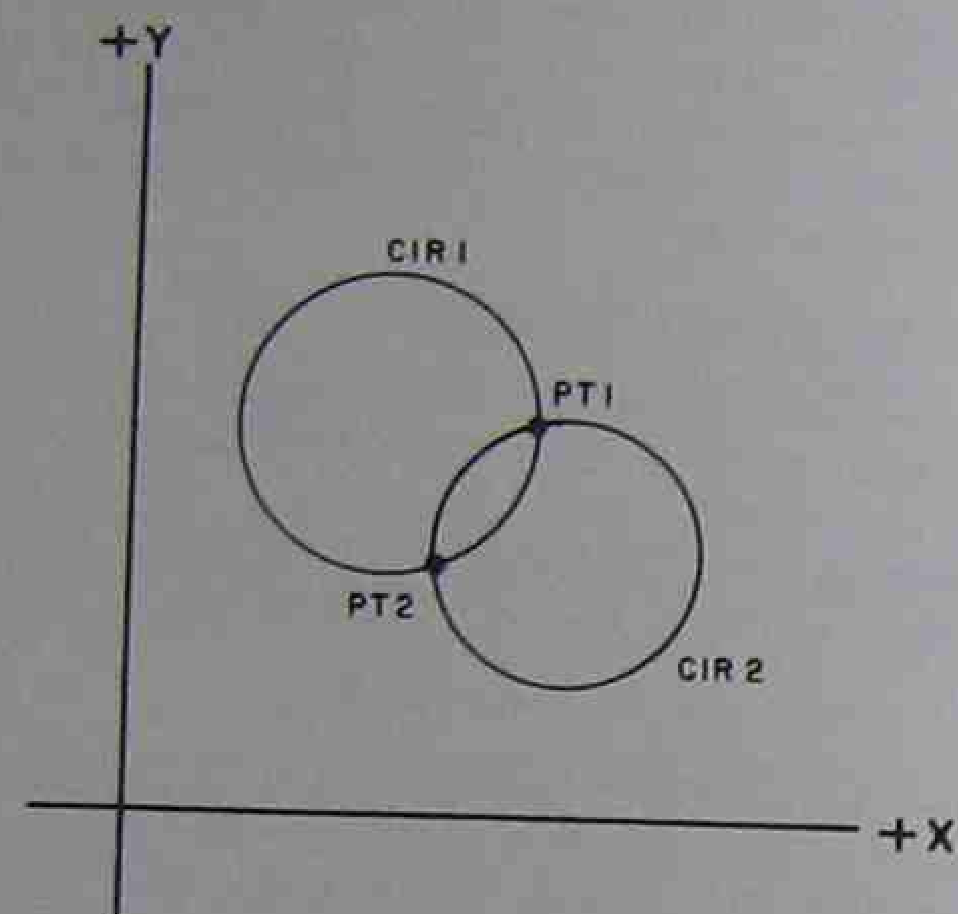
a. the intersection of two lines (shown below):



DPT1,LN1,LN2,3ƵB

(3ƵB describes the distance of the point from the *XY* plane, in the positive Z direction);

b. the intersection of a line and a circle or by the intersection of two circles, e.g.,

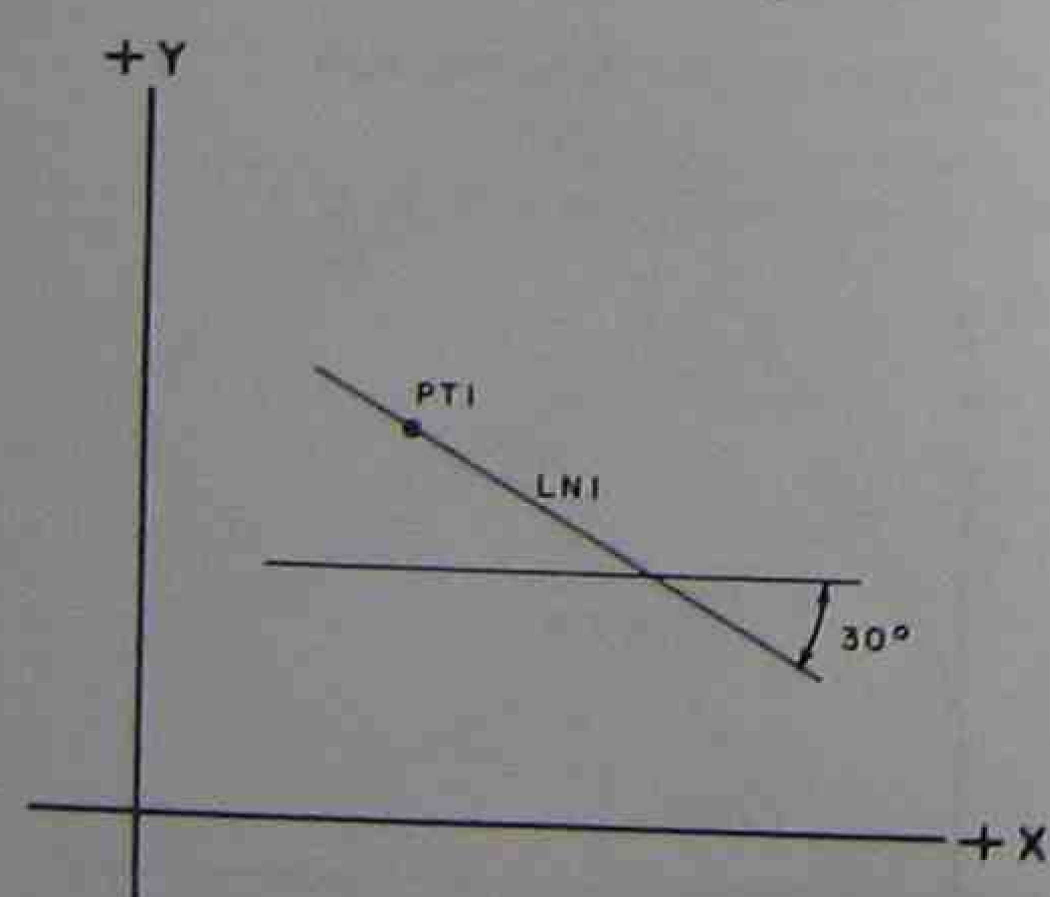


DPT1,CIR1,CIR2,XL,3ƵB

(the XL distinguishes PT1 from PT2 in that PT1 is larger in the *X* direction than PT2. The statement for PT2 would be DPT2,CIR1,CIR2,XS,3ƵB).

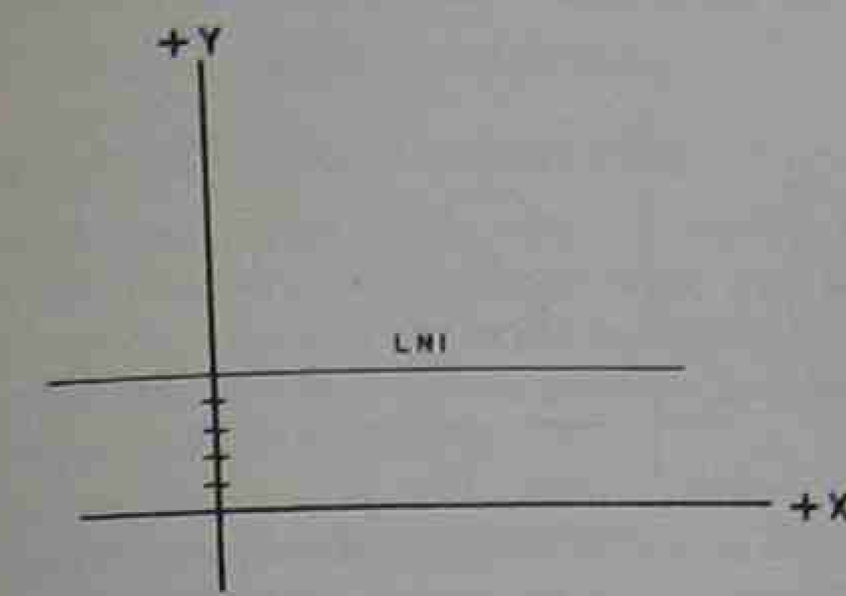
A line may be defined a number of ways, e.g.,

a. as passing through a point and being at an angle with the *X* axis:



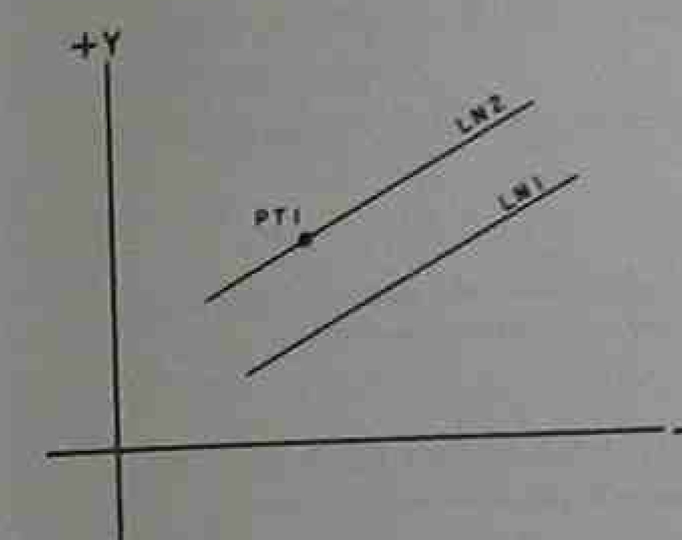
DLN1,PT1,30CW

b. as lying a distance from the X or Y axis and being parallel to it:

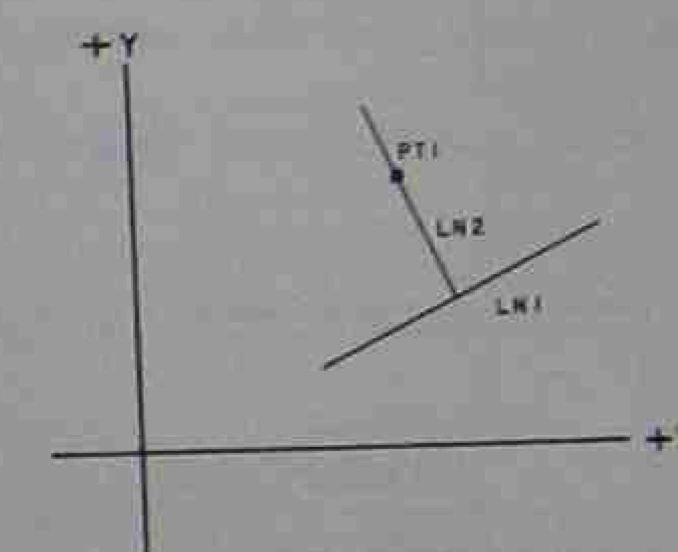


DLN1,5YB

c. as passing through a point and either being parallel or perpendicular to another line:

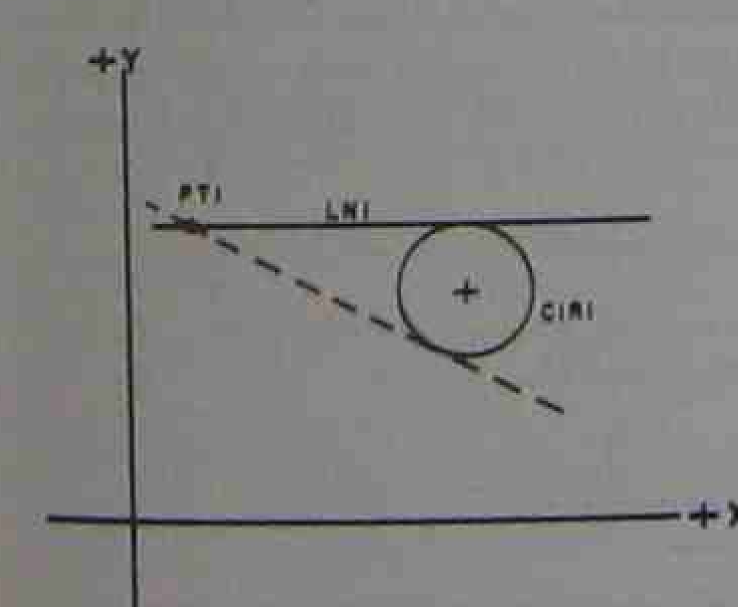


DLN2,PT1,PARLN1

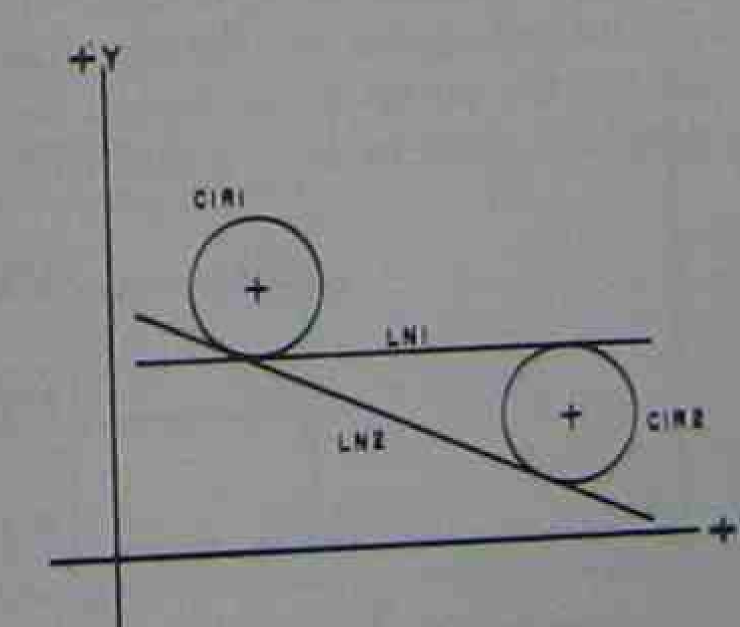


DLN2,PT1,PERLN1

d. as passing through a point and being tangent to a circle, or as being tangent to two circles:



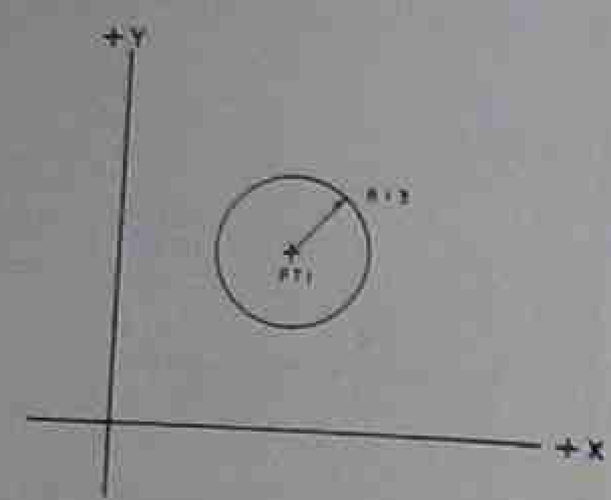
DLN1,PT1,CIR1,YL



DLN1,CIR1,YS,CIR2,CRØSS
DLN2,CIR1,YS,CIR2

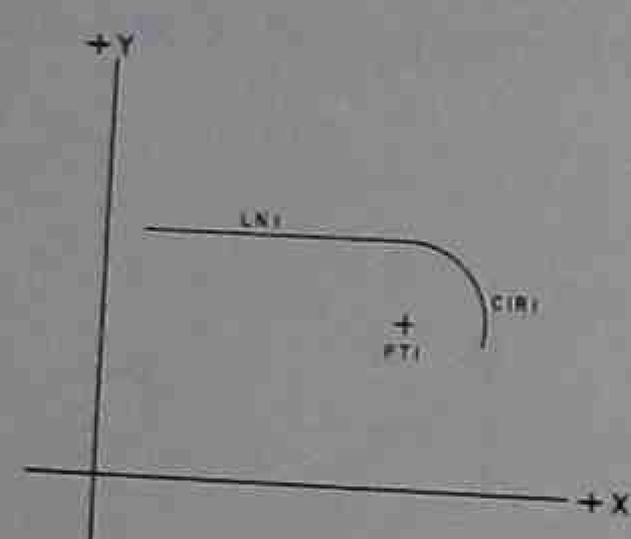
A circle may be defined a number of ways, e.g.,

a. by the center of the circle and its radius:

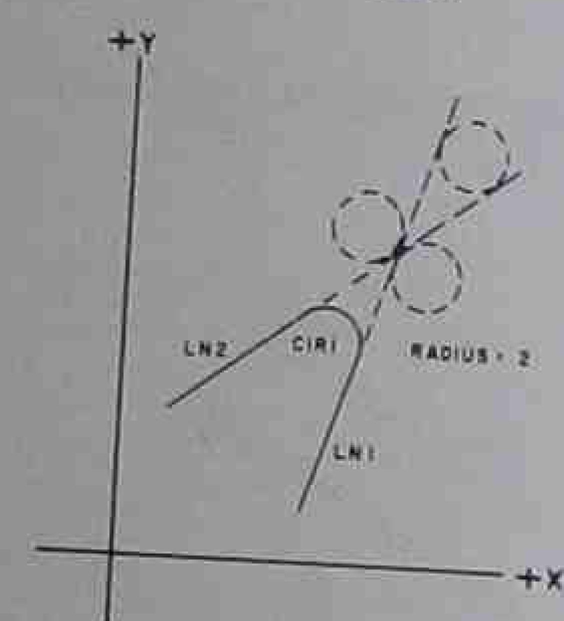


DCIR1,PT1,3R

b. by the center and tangent to a line or tangent to two lines:



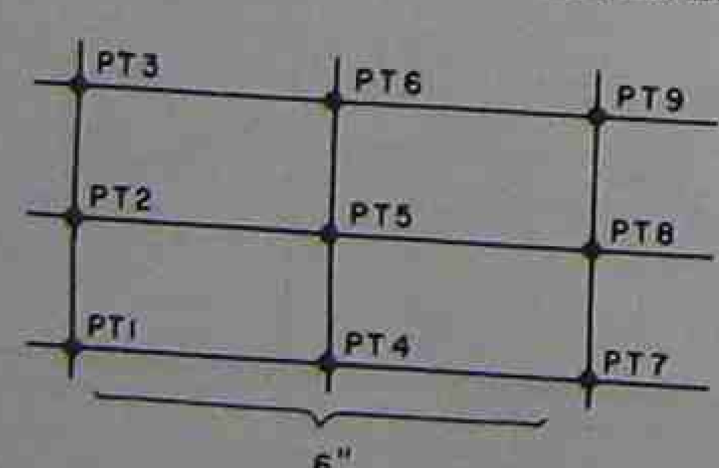
DCIR1,PT1,LN1



DCIR1,LN1/2XS,LN2/2YS,2R

POINT-TO-POINT PATTERNS

Patterns, or sets as they are called in the COMPACT II system, may be described and operations performed at each of the points in the pattern. In the illustration below the SET is described for the three points from PT1 through PT3. This set, which is labeled SET1, is then repeated, in the positive *X* direction, two times. The rectangular set that is formed is then called SET2. The drill is then directed to the points in the pattern by a drill statement which notes how far the drill is to move in the −Z direction. The statements are also shown below.



DSET1,S(PT1),F(PT3),3EQSP,NØMØRE
DSET2,SET1,3EQSP,6LX,NØMØRE

Drilling instructions, such as the feed rate and the gage length of the tool, would be contained in a previous statement, e.g.,

MTCHG,.02IPR,5GL

WRITING AN APT PART PROGRAM (MILLING)

Consider the part shown in Fig. 5–14, which is similar, in form, to that described earlier in this chapter. The perimeter of the part is to be machined in one pass. The material is to be held above the worktable by a 1-inch riser block. (See Fig. 5–15.) The symbols that are assigned to the geometry segments are also shown in the top view of Fig. 5–15.

The starting point for the end mill cutter is shown in Fig. 5–16 and its location is noted by the symbol *SP*. This is at the tip and center of the cutter. The cutter moves from SP normal to the line L1 (extended). At the same time the cutter is moving down, to a position .1″ below the bottom of the part. The cutter next "makes a right turn" and moves along L1 to a point where L1 is tangent to C1. The cutter continues to move around C1 and on to line L2. After moving along L2 and past the extension of L3, the cutter "makes a left turn" and moves along L3 until it passes the extension of L1. The cutter then moves back to SP and stops.

The complete part program is shown below. The numbers in parentheses

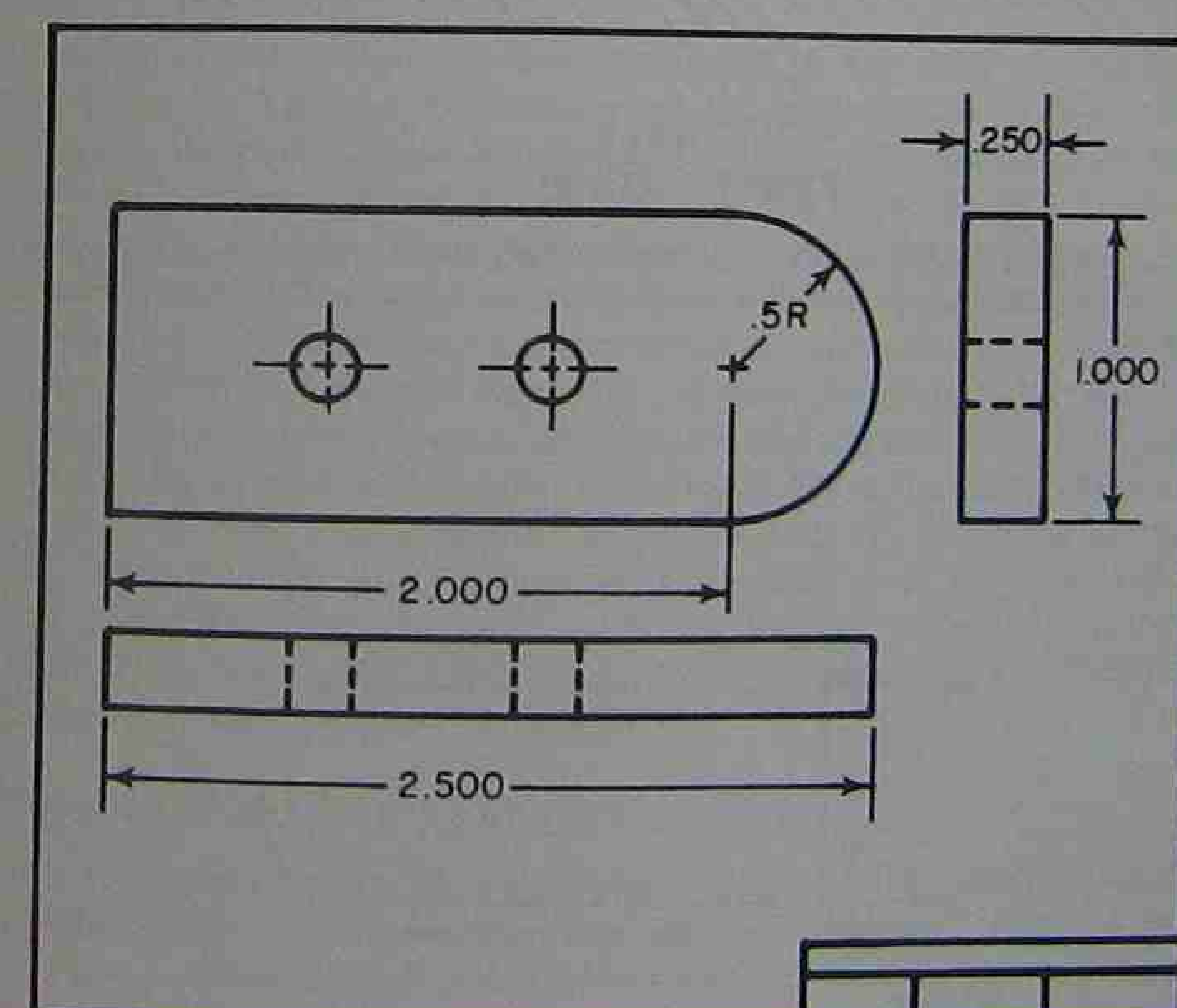


Fig. 5–14. Engineering drawing describing part to be machined. The two holes shown are for holding the part and are drilled prior to NC machining.

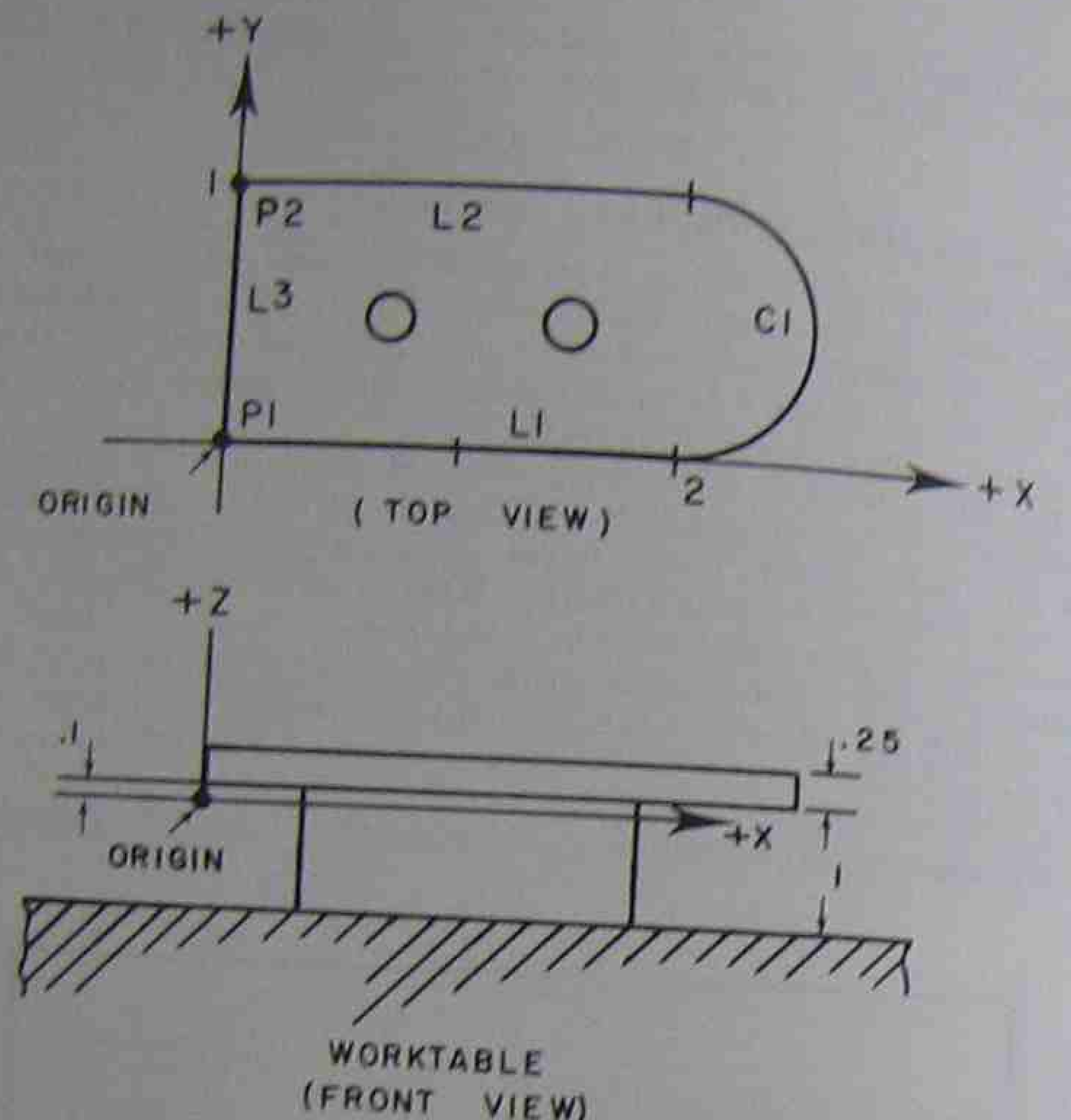


Fig. 5–15. The part described in Fig. 5–14 is held above the worktable by a 1″ riser block The origin, or point where X = 0, Y = 0, Z = 0, is below the left-hand bottom corner of the part.

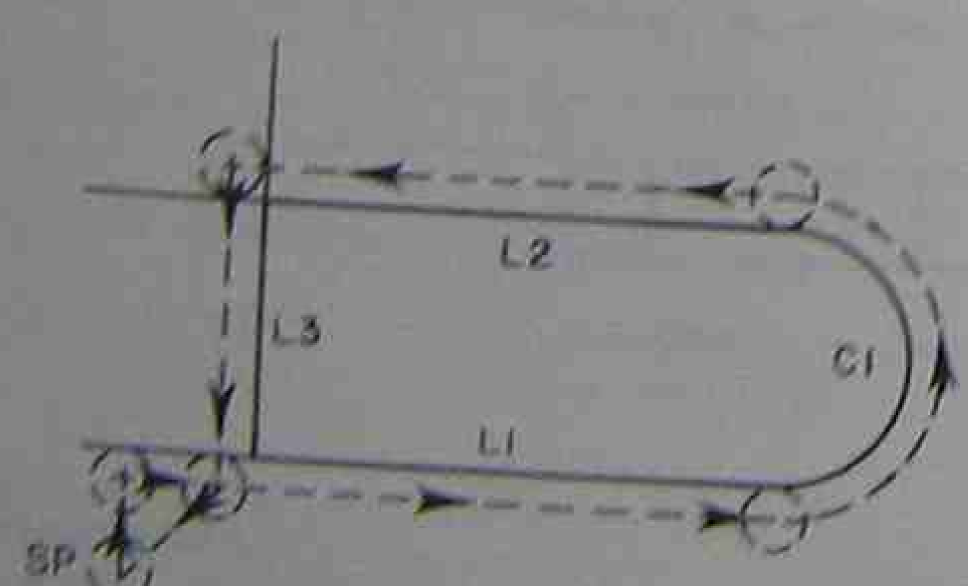


Fig. 5–16. The path of the center of the cutter is shown as it moves about the perimeter of the part.

shown at the left side of the program are for reference purposes and are not part of the program.

```
(1)              PARTNØ FLAT PLATE NØ 12345678
(2)              MACHIN/MØDEL PTX
(3)              CLPRNT
(4)              CUTTER/.25
(5)              FEDRAT/10
(6)    SP  =     PØINT/−.5,−.5,1
(7)    P1  =     PØINT/0,0,0
(8)    L1  =     LINE/P1,ATANGL,0
(9)    C1  =     CIRCLE/2,.5,.5
(10)   P2  =     PØINT/0,1,0
(11)   L2  =     LINE/P2,PARLEL,L1
(12)   L3  =     LINE/P2,PERPTØ,L1
(13)             FRØM/SP
(14)             GØ/TØ,L1
(15)             GØRGT/L1,TANTØ,C1
(16)             GØFWD/C1,TANTØ,L2
(17)             GØFWD/L2,PAST,L3
(18)             GØLFT/L3,PAST,L1
(19)             GØTØ/SP
(20)             FINI
```

Referring to the above, the first statement in any APT part program is PARTNØ, line (1). This may be followed, on the same line, by any identification form, such as the part number or part name. Line (2) calls out the postprocessor for the machine/control combination that is to machine the part. The postprocessor is that part of the computer software program that tailors the tape instructions for the particular machine/control combination. Every different machine/control combination requires its own postprocessor even though the APT processor, which is the main computer software program, is the same for all machine/control combinations. Line (3) notes that the computer is to print out the coordinates of all the straight-line moves of the cutter. Line (4) notes that the cutter is to have a diameter of .25 inches. Line (5) describes the feedrate in ipm. Lines (6) through (12) describe the geometry of the part. Lines (13) through (19) are motion statements and describe the path of the cutter. Line (20) ends the part program. Preparation of the machine tape is automatic after this program is input to the computer.

WRITING A COMPACT II PART PROGRAM (MILLING)

Consider the same part as with the APT part programming example (Fig. 5–14). Also, consider the same setup and tool path as illustrated with the APT example, Figs. 5–15 and 5–16; however, with COMPACT II, the symbols for the geometry elements are different. L1 with the APT program would have to be LN1 with COMPACT II. L2 would be LN2. C1 would be CIR1. (See Fig. 5–17).

The program for machining the part along the same path as with the APT

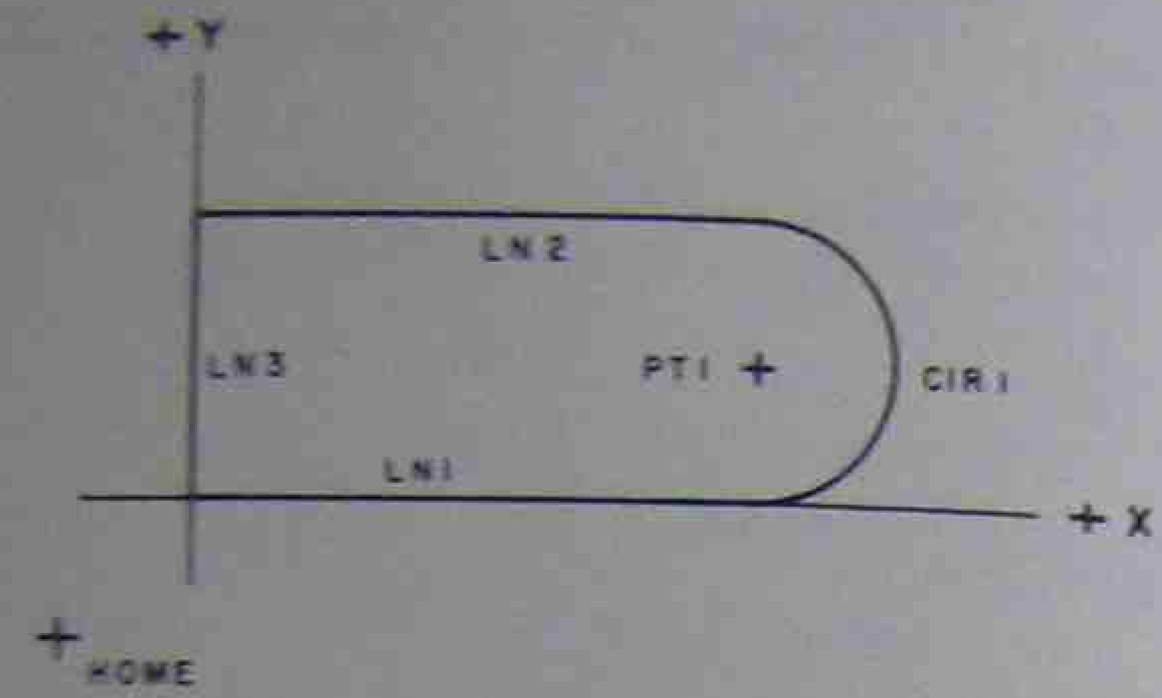


Fig. 5–17. Unlike APT, the symbols for the geometry of COMPACT II are specific. PT, for example, must be used for a point, LN for a line, and CIR for a circle.

program is as follows. The number shown in parentheses before each line is for reference and is not part of the program.

```
(1)  MACHIN,MØDEL QTY
(2)  IDENT,FLAT PLATE NØ 12345678
(3)  SETUP,-.5LX,-.5LY,4LZ
(4)  BASE,XA,YA,ZA
(5)  DLN1,YB
(6)  DLN2,1YB
(7)  DLN3,XB
(8)  DPT1,2XB,.5YB,ZB
(9)  DCIR1,PT1,.5R
(10) MTCHG,TØØL1,.25TD,3GL,10IPM
(11) MØVE,TØLN1,ZB
(12) ØCØN,CIR1,CCW,S(TANLN1),F(TANLN2)
(13) CUT,PASTLN3
(14) CUT,PASTLN1
(15) HØME
(16) END
```

Referring to the above, the first statement in any COMPACT II part program is MACHIN, ________, line (1). This is followed, on the same line, with the code for the machine/control combination that the part is being programmed for. In the program above it is MODEL QTY. This corresponds to the MACHIN statement in APT. Line (2) notes the identification of the part. Line (3) describes the location of the center of the face of the spindle with respect to the zero location on the machine. This zero location may be *fixed* on some machines and *floating* on others. In Fig. 5–18 it is floating and therefore coincides with the zero location for the BASE which is noted in line (4). XA, YA, and ZA, noted in line (4), mean that the zero point, or origin of the BASE, is zero inches in the X direction, zero inches in the Y direction, and zero inches in the Z direction with respect to the zero location. Lines (5) through (9) describe the geometry of the part. Line (10) notes that the tool change is manual, i.e.,

MTCHG; the tool diameter is .25"; the gage length of the tool is 3"; and the feedrate is to be 10 ipm. Lines (11) through (14) are motion statements. Line (15), also a motion statement, directs the cutter to the HOME position, which is the same as the original starting position. The word END, line (16), is the last statement in a COMPACT II program.

It should be pointed out that both the APT and the COMPACT II programs could have been written more concisely. The programs have been written in a fashion that it is felt best illustrates the principles of the systems.

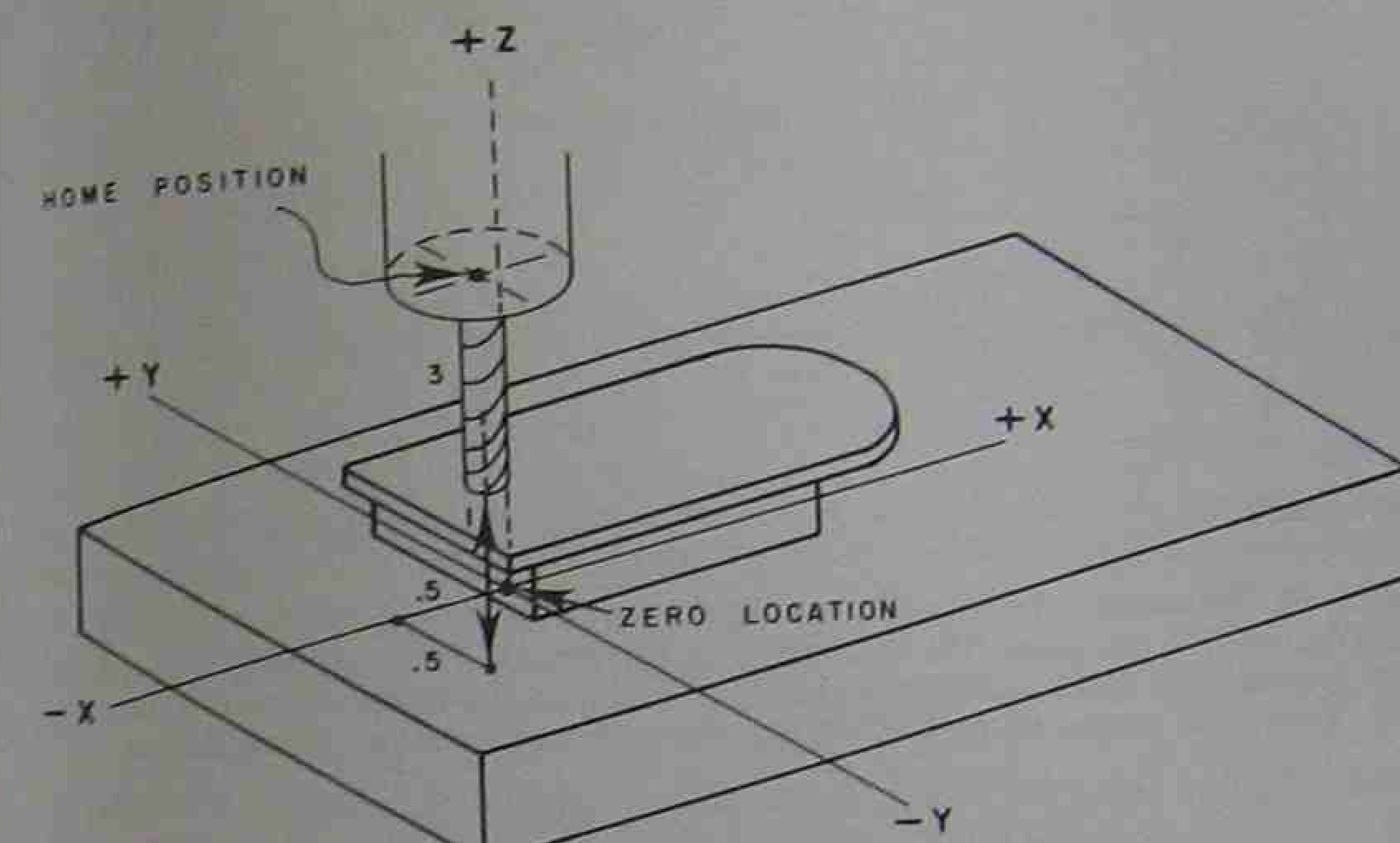


Fig. 5–18. In a free floating zero system, as shown above, the zero location can be set at any point within the range of the machine. The setup location, or HOME position, may then be set with respect to the zero location. In this case the HOME position, which is at the center and face of the spindle, is −.5" in the X direction, −.5" in the Y direction, and 4" in the Z direction. The 3" distance to the tip of the tool is noted by 3GL (3" Gage Length) in the MTCHG statement. With a fixed zero system, the zero location is set and is usually located at the lower left-hand corner of the table.

WRITING AN APT PART PROGRAM (TURNING)

Since APT was developed initially for contour milling machines, and is still considered the most suitable for simultaneous three-, four-, and five-axis machining, it is not very well suited, in its basic form, for the simpler two-axis geometry inherent with turning operations. What is required is that preprocessors or special routines be developed that utilize the basic APT system; for example, special routines have been developed for roughing, boring, and threading operations. There are a number of commercially available turning systems that use the APT system as their base. One such turning system is called the United Lathe Module, which is offered by McDonnell Douglas Automation Company. A part program utilizing this system is shown below. Figure 5–19 shows the setup and Fig. 5–20 offers a closeup description of the

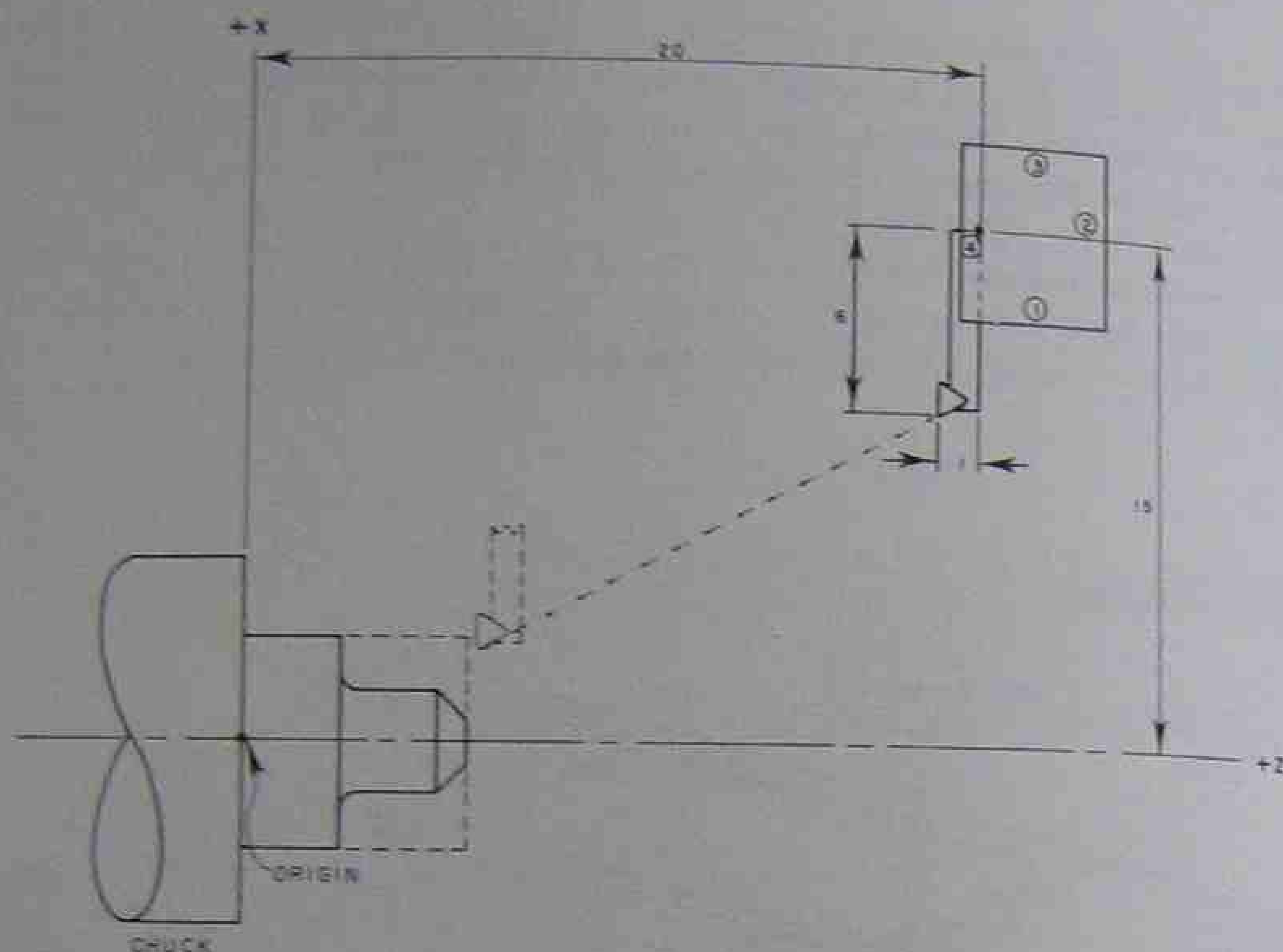


Fig. 5–19. The setup is described by noting the distance of the tool from the origin or from the absolute zero point. The location of the tip of each of the cutting tools is defined with respect to the fixed locations on the turret so that as the turret rotates the computer will "know" the precise location of the tool tip with respect to the *X* and *Z* axes.

geometry profile of the part, together with dimensions as described by the *X*- and *Z*-axis intercepts.

```
(1)  PARTNØ LATHE EXAMPLE
(2)  MACHIN/ MØDEL LATHE
(3)         T1 = TØØL/FACE,1,XØFF,−1,YØFF,−6,RADIUS,.031
(4)  BLANK1 = SHAPE/FACE,3.5,TURN,2
(5)    PART1 = SHAPE/FACE,3.5,TAPER,3.5,.5,ATANGL,−45,TURN,1$
                FILLET,.25,FACE,1.5,TURN,2
(6)             FRØM/(20−1),(15−6)
(7)             LATHE/RØUGH,BLANK1,PART1,STEP,.1,STØCK,.05,$
                SFM,300,IPR,.01,T1
(8)             LATHE/FINISH,PART1,SFM,400,IPR,.005,T1
(9)             END
(10)            FINI
```

Line (3) describes the tool. In this case the tool is located on face 1 of the turret and its tip is −1 inch "off" in the *X* direction and −6 inches "off" in the *Y* direction, when considering *XY* rather than *XZ* axes. The radius of the tip of the cutting tool is also noted in this statement. Line (4) describes the dimensions of the rough material, or blank. Lines parallel to the *X* axis are

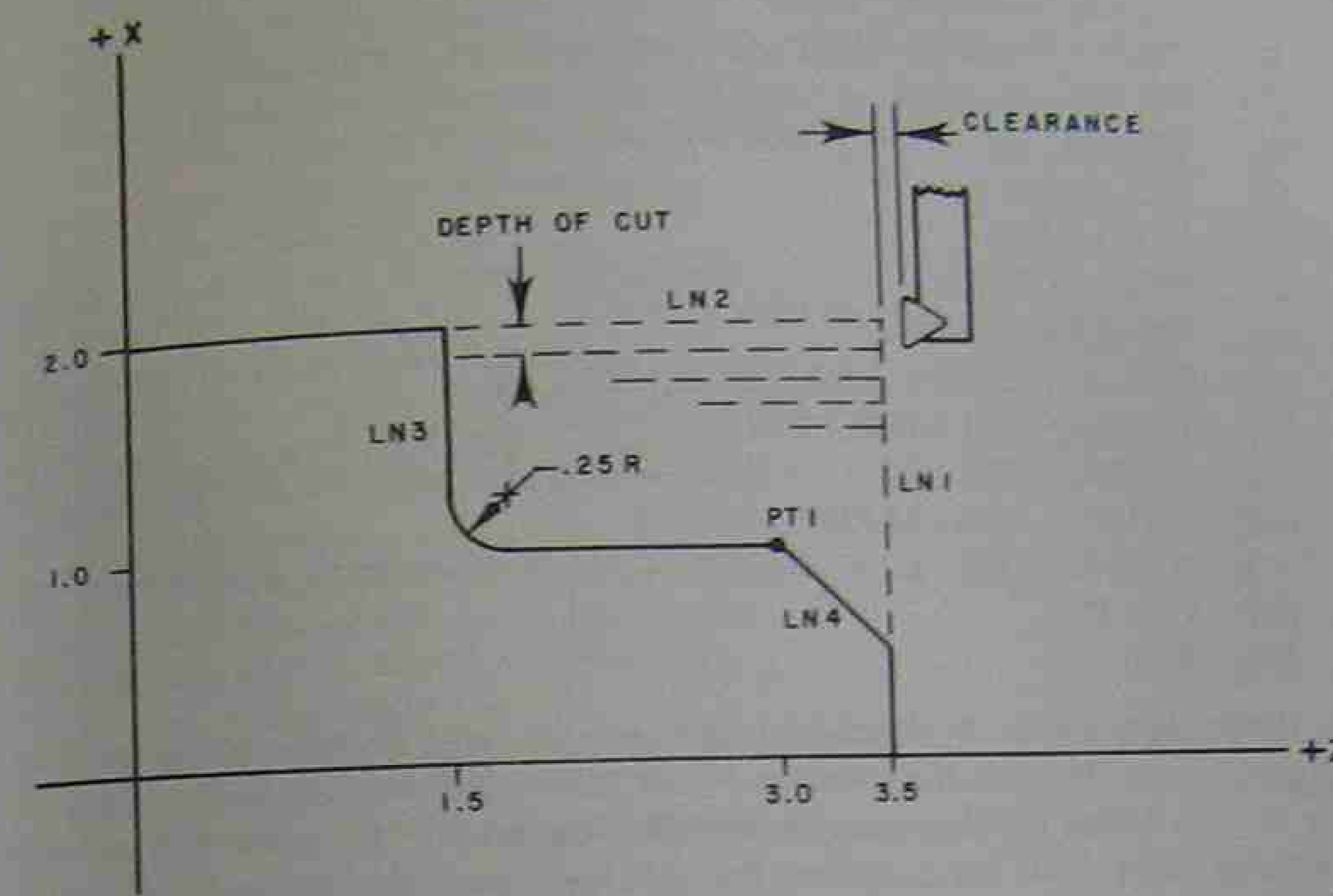


Fig. 5–20. Profile of the part shown in the chuck in Fig. 5–19. The symbols which identify the lines, namely, LN1, LN2, LN3, and LN4, are used with the COMPACT II part programming system.

noted as FACE lines, and lines parallel to the *Z* axis are noted as *TURN* lines. The FACE line in this case is 3.5 inches along the *Z* axis and parallel to the *X* axis. The TURN line is located 2 inches above the *Z* axis and parallel to it. Line (5) describes the shape of the finished part. The term *FILLET* used in this statement to describe a circle that is tangent to the line described by TURN,1 and the line that is described by FACE,1.5. The $ sign means that the statement is continued on the next line. It should be pointed out that these geometry elements must be contiguous and must be described in sequence. Line (7) describes the roughing operation and notes that the material to be roughed out lies between BLANK1 and PART1; that the STEP, or depth of cut, is to be .1 inch; that .05 inch is to be left for the finish cut; that the speed is to be 300 sfm and the feedrate is to be .01 ipr; and that the tool to be used is noted by the symbol T1. Line (8) describes the finish cut, which is to be along the contour described by PART1.

WRITING A COMPACT II PART PROGRAM (TURNING)

A program for the same part, as with the APT example, is written in COMPACT II (FasTurn) language and shown below. The setup is as shown in Fig. 5–19 and the profile of the part is as shown in Fig. 5–20. In this case LN1, LN2, LN3 and LN4 have been identified in Fig. 5–20 for this program since they form the intersections of the starting and finishing points and are required.

```
(1)  MACHIN,MØDEL TURN
(2)  IDENT,SAMPLE PART
(3)  SETUP,15X,20Ƶ
(4)  DLN1,3.5ƵB
(5)  DLN2,2XB
(6)  DLN3,1.5ƵB
(7)  DPT1,1XB,3ƵB
(8)  DLN4,PT1,45CW
(9)  DPB1,LN1,S(LN4);LN2,F(LN3),NØMØRE
(10) DPB2,LN4,S(LN1);2D;.25R;LN3,F(LN2),NØMØRE
(11) ATCHG,TØØL1,GLX-1,GLƵ-6,.031TLR,300FPM,.01IPR
(12) CUT,PB2,MB1,MAXDP.1,STK.05
(13) ATCHG,TØØL1,GLX-1,GLƵ-6,.031TLR,400FPM,.005IPR
(14) CUT,PB2
(15) END
```

Referring to Fig. 5–20, lines (1) and (2) are the same as required for a milling program. Line (3) notes the location of the cutter with respect to the origin (cf. Fig. 5–19). Lines (4) through (8) describe the geometry for the starting and finishing lines. Line (9) describes the outside boundary for the material that is to be roughed out. One line of the boundary is LN1 and this *starts* the boundary where LN4 intersects LN1, thus the notations LN1,S(LN4) in line (9). The same reasoning applies to the notations LN2,F(LN3) in which LN3 *finishes* the boundary where it intersects LN2. Line (10) describes the finished part boundary. The notation *2D* describes the line that is parallel to the Ƶ axis and is equivalent to a 2-inch *diameter*, or a 1-inch distance, from the Ƶ axis. The notation *.25R* describes a circle that is tangent to the *2D* line and LN3. As with the APT program, the lines and circles described in this manner must be tangent and must be described in sequence. Line (11) calls for the proper tool; notes the cutter location; notes the tool radius; and specifies the speed and feedrate. Line (12) instructs the cutter to machine the area between the material MB1 and part boundaries PB2, to a maximum depth of .1 inch and to leave .05 inch of stock for a finish cut. Line (13) specifies the same tool only, in this case, the speed and feedrate are changed for the finish cut. Line (14) is the instruction for cutting along the profile of the part, and thus the finish cut. And line (15) ends the program.

PART PROGRAMMING SYSTEMS

A list of companies offering part programming systems and services is shown below. Source: Numerical Control Lathe Language Study and Numerical Control Language Evaluation sponsored by the U.S. Army Communications Research and Development Command, Fort Monmouth, N.J. Copies of this report may be obtained from the Numerical Control Society, 519 Zenith Drive, Glenview, Illinois 60025.

Cincinnati Milacron, Inc., Machine Tool Group
4701 Marburg Avenue, Cincinnati, Ohio 45209

Digital Systems Corporation
317 Monroeville Mall, Monroeville, Pennsylvania 15146

Encode, Inc., Perkins Way
Dexter Industrial Green, Newburyport, Massachusetts 01950

General Electric Co., Information Services Business Division
401 N. Washington Street, Rockville, Maryland 20850

Ingersoll Milling Machine Company
Rockford, Illinois 61101

Manufacturing Data Systems, Incorporated
4521 Plymouth Road, Ann Arbor, Michigan 48105

Manufacturing Software & Services
6761 Bramble Avenue, Cincinnati, Ohio 45227

McDonnell Douglas Automation Company
P.O. Box 516, St. Louis, Missouri 63166

Olivetti Corporation of America
500 Park Avenue, New York, New York 10022

Structural Dynamics Research Corporation
5729 Dragon Way, Cincinnati, Ohio 45227

Threshold Technology, Inc.
1829 Underwood Boulevard, Delran, New Jersey 08075

A. S. Thomas, Inc.
355 Providence Highway, Westwood, Massachusetts 02090

United Computing Corporation
22500 S. Avalon Boulevard, Carson, California 90745

University Computing Company
1930 Hi Line Drive, Dallas, Texas 75247

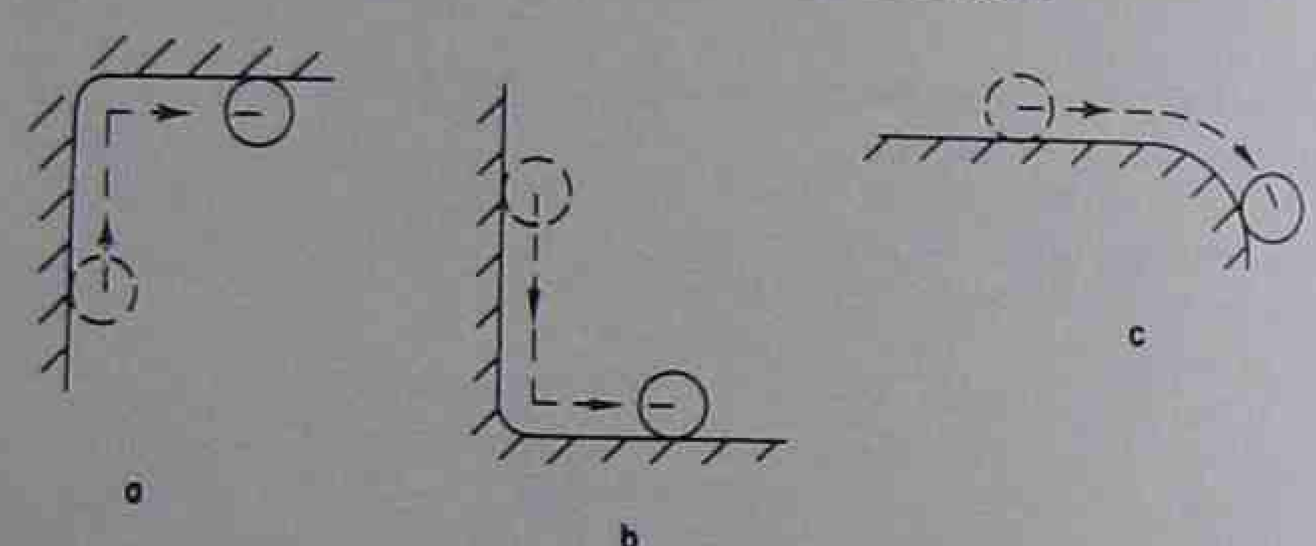
Weber N/C Systems
11611 West North Avenue, Milwaukee, Wisconsin 53226

Westinghouse Electric Corporation, Industry Systems Division
200 Beta Drive, Pittsburgh, Pennsylvania 15238

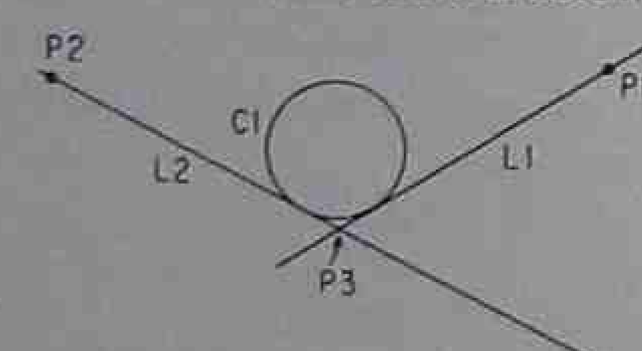
QUESTIONS TO CHAPTER 5

1. Computer-aided part programming refers to the assistance offered by the computer within the CNC system. (True, False.)
2. The most common media for transporting the data and information to the numerical control system is:
   a. Floppy disc
   b. Magnetic tape
   c. Punched tape
3. The punched tape prepared by the computer is completely different from that prepared manually. (True, False.)
4. Computer-aided part programming is suitable when (answer true or false for each of the items noted below):
   a. there are a goodly number of complex parts
   b. when there are a few simple parts
   c. when there are many simple parts
   d. when the production runs are short
   e. when the production runs are very long and the parts are simple
   f. when the parts are simple with respect to geometry, however, require lengthy tapes.
5. COMPACT II unlike APT is a proprietary system and offered only by CAM–I. (True, False.)

6. In order to be able to use computer-aided part programming it is necessary to have a large scale computer on site. (True, False.)
7. The computer operates on a (select one):
   1. Quadruple System
   2. Planery System
   3. Binary System
   4. Alphabetic System
8. In order to converse more readily with a computer, assemblers and compilers have been developed. (True, False.)
9. An assembler is an electronic piece of hardware that assembles the data for a numerical control program. (True, False.)
10. Name the three surfaces involved in an APT move.
11. Two of the three surfaces in APT appear as lines when viewed directly from above. What are these surfaces?
12. A cutter may move *to*, *on*, *past*, and *way past* a check surface. (True, False.)
13. Note the APT statements for the illustrations below.

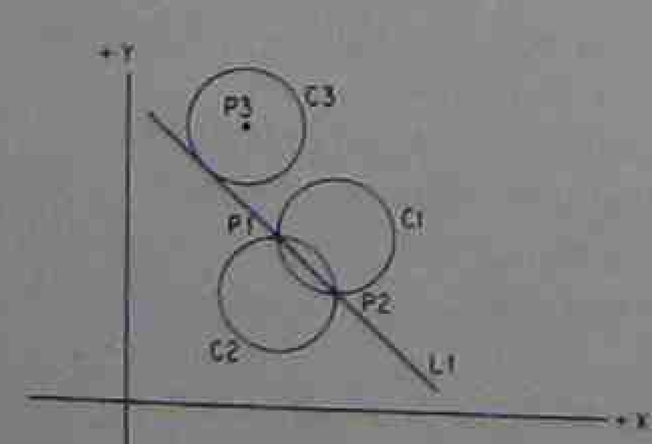


14. a. Considering P1, would the line, L1, shown below, be to the *right* or *left* of the circle?



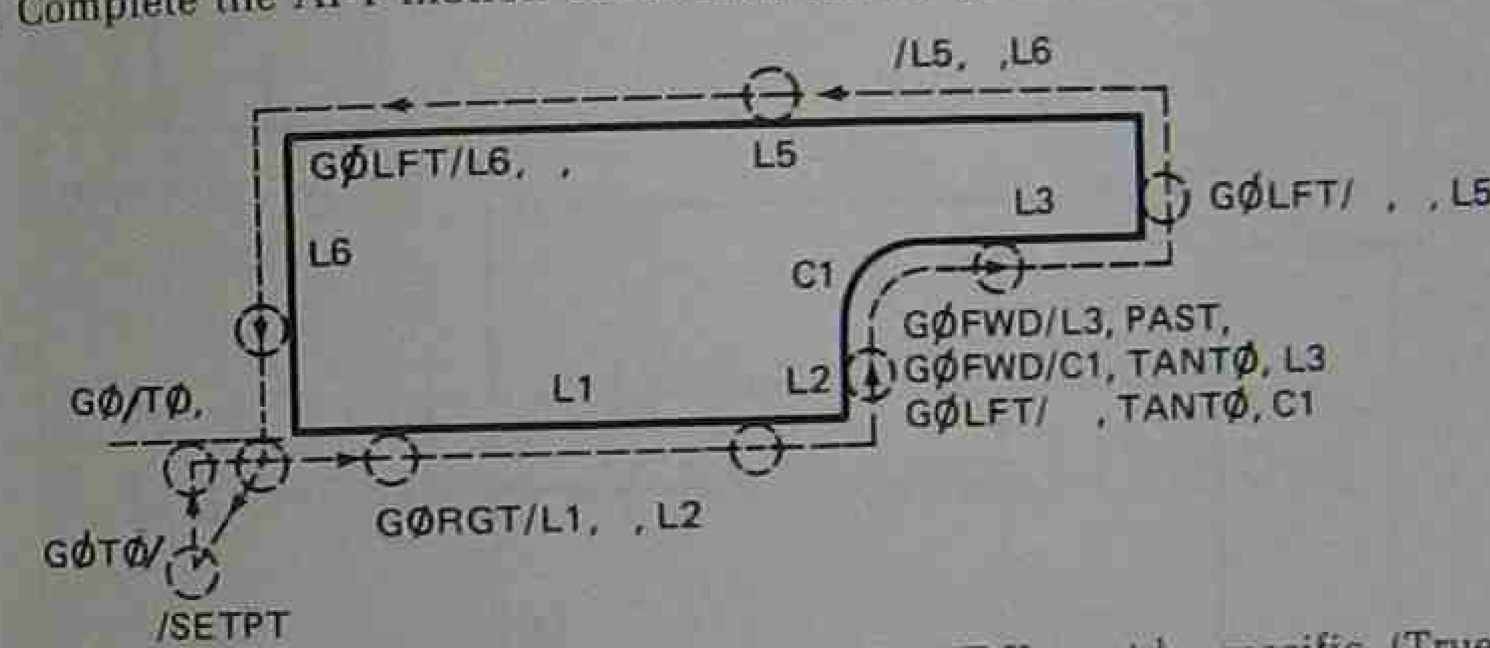
b. Considering P2, would L2 be to the *right* or *left* of C1?
c. Complete the statement below, to correctly describe P3:
P3 = PØINT/ ,L1,

15.

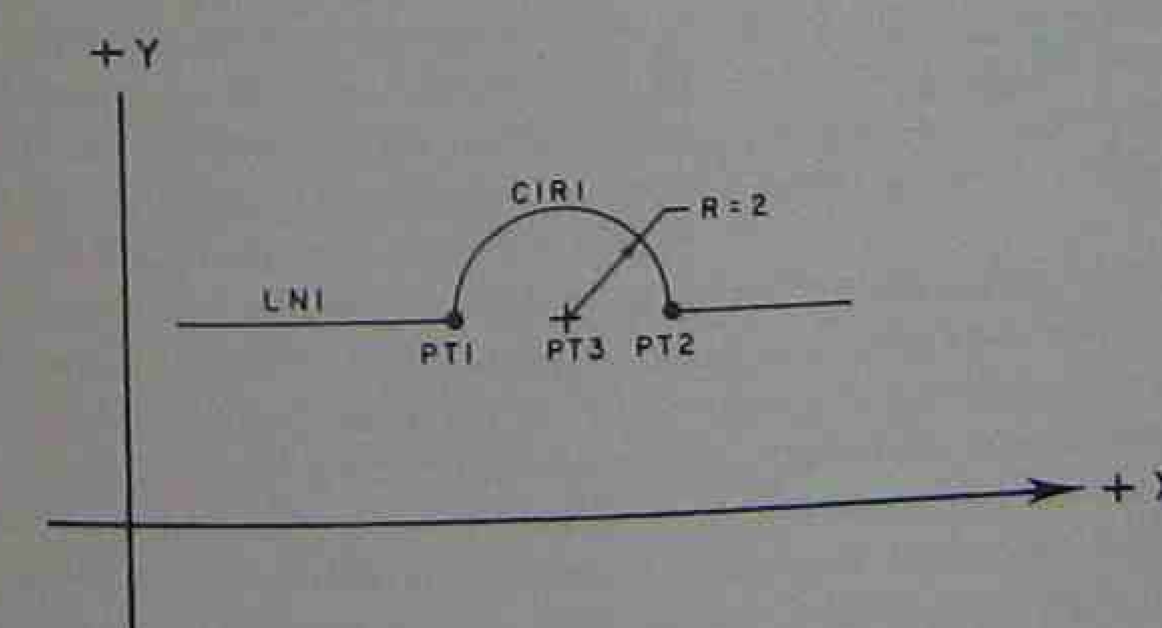


Referring to the above sketch, complete the following statements: (This is not a program—merely individual statements.)
P1 = PØINT/X ,INTØF, , ,
P2 = PØINT/Y , , , ,
L1 = LINE/ ,
C3 = CIRCLE/CENTER, , ,L1

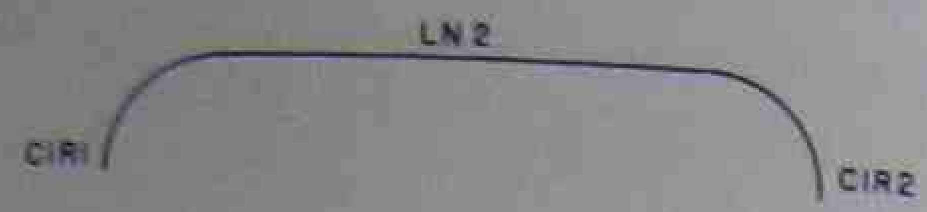
16. The following APT statements were prepared by an inexperienced part programmer and were key punched by an even less experienced key punch operator. Circle their errors and omissions where they occur. (This is not a complete program—merely separate statements.)
   a. L1 = LIN/P1,PARLEL,L2,
   b. L1 = LINE ,P2/PERPTØ,L2
   c. PT1 = PØINT/2,3
   d. L3 = CIRCLE/5,6,3
   e. FEEDRATE/15
   f. ØUTØL/.0005
   g. CUTTER,2 DIA
17. Complete the APT motion commands shown in the sketch below:



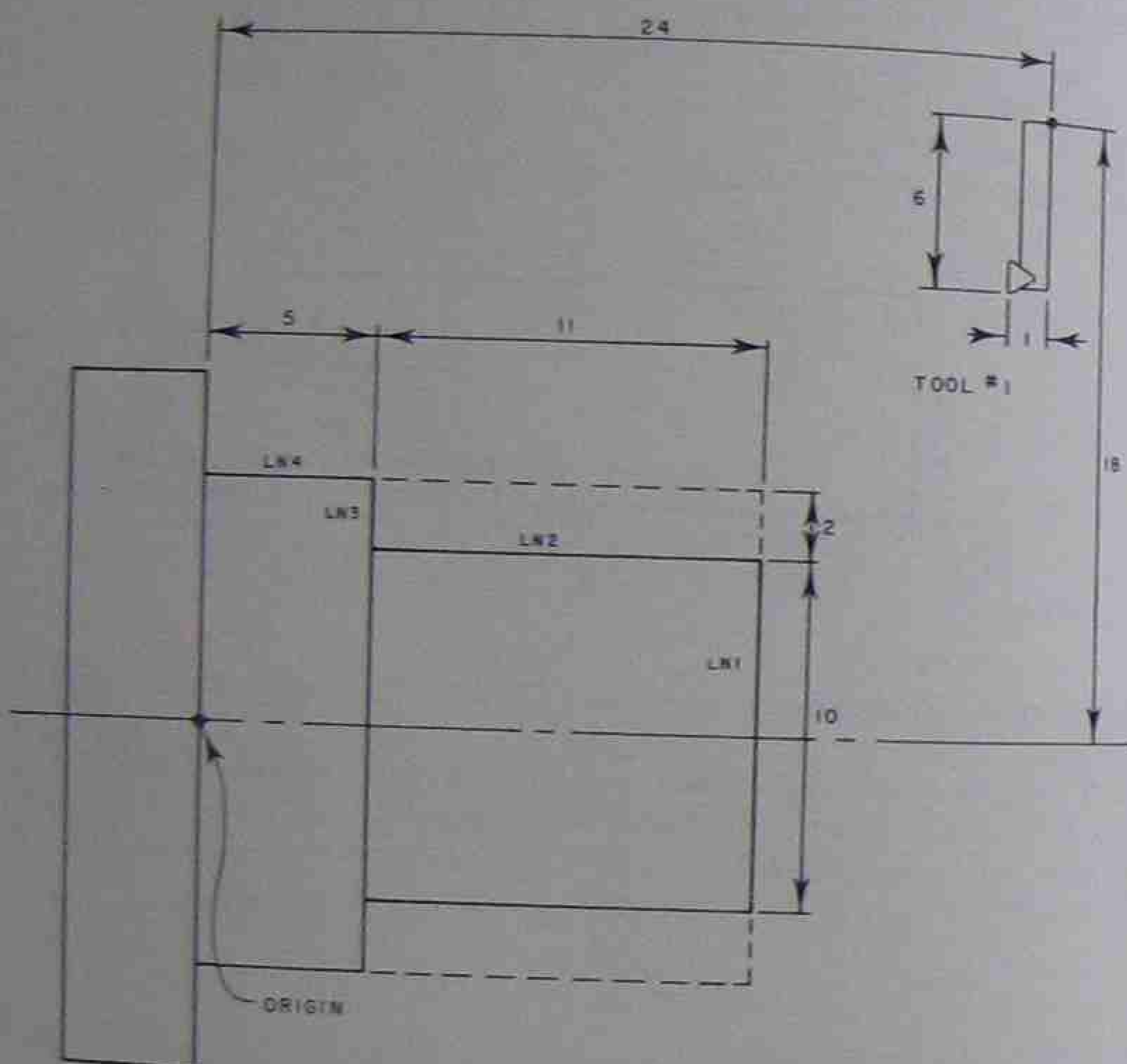
18. Unlike APT the symbols used with COMPACT II must be specific. (True, False.)
19. A point lies on coordinates X = 1.5 and Y = 6.7 and is 8.5 inches above the XY plane. What is the COMPACT II geometry statement for the point?
20. Referring to the sketch below, LN1 and PT3 have been defined. Write the COMPACT II geometry statements for CIR1, PT1, and PT2.



21. Referring to the sketch below, write the statement for LN2, using COMPACT II language. CIR1 and CIR2 have been defined.



22. Write both the APT (United Lathe Module) and the COMPACT II (FasTurn) part programs for turning the part shown below. The radius of the tool tip is .031". The speed is to be 300 fpm for the roughing cuts and 400 fpm for the finish cut. The maximum depth of the rough cuts is to be .1" and the stock for the finish cut is to be .05". The feed is to be .01 ipr for the roughing cuts and .005 ipr for the finish cut.



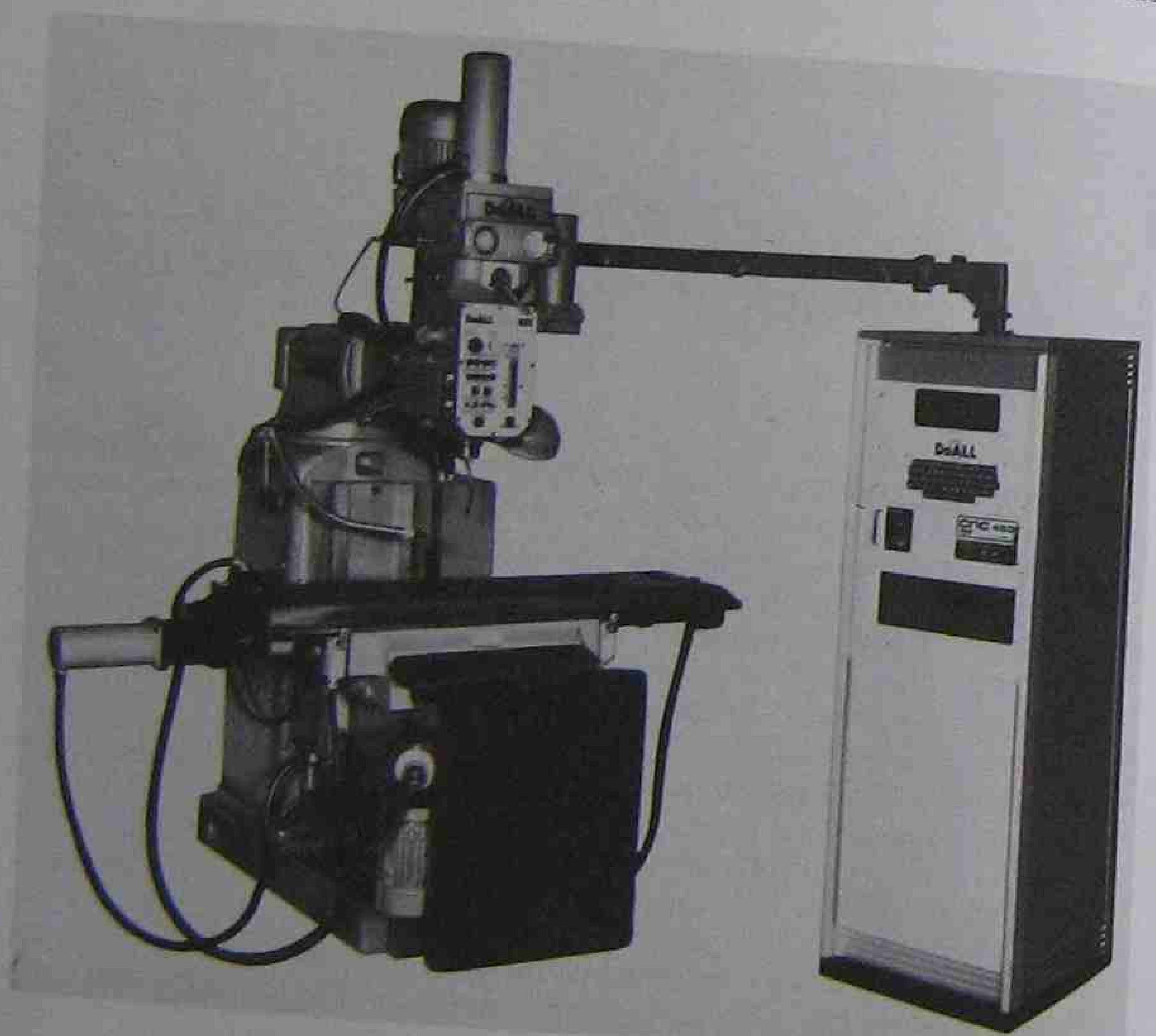
CHAPTER SIX

Advantages of Numerical Control

Before considering, in detail, the numerous advantages of numerical control, it would probably be helpful to briefly review the essential differences between a numerically controlled machine and its conventional counterpart. Although there may be some significant differences in dynamic design characteristics and, to some extent, in appearance, the functional performances remain the same. Numerical control machines are expected to drill, bore, ream, and tap holes the same as their conventional counterparts. Numerical control lathes must turn parts using similar metal-cutting techniques as conventional lathes. And numerical control punch presses punch holes in sheet metal with a single stroke as do conventional punch presses.

Early numerical control machines were essentially conventional machines that had been modified to accept electronic controls and drive mechanisms. They therefore looked very much like conventional machine tools. As the technology advanced and functions were combined the numerical control machines began to take on a distinctive appearance. *Machining centers* were developed that can perform numerous functions including drilling, boring, milling, reaming, and tapping in addition to changing cutting tools and workpieces automatically. These multifunction capabilities required greater rigidity of the machine tool members, which resulted in numerous design innovations.

NC lathes have also undergone some radical changes. The conventional horizontal bed, for example, has become slanted, which adds rigidity and better operator accessibility. The lower cost NC machines, however, still retain something of a conventional appearance. Low cost NC milling machines, for example, even if equipped with devices for automatic drilling, tapping, and boring, still look pretty much like conventional milling machines. (See Fig. 6–1.) But it is not so much the comparison in appearance between numerical control and conventional machines that is significant but rather their capabilities. The addition of electronic controls to a basic machine tool does not radically change its inherent capabilities such as horsepower and speed ranges. What has resulted, essentially, is that a numerical control machine tool may now perform up to its highest mechanical capabilities whereas the human operator and a conventional machine could not possibly compete with the more precise and repetitious electronic controls. Parts machined by numerical control may now be produced in less time, with less tooling, and with greater accuracy. A study, in detail, of the numerous advantages of numerical control follows.



Courtesy of DoAll Company

Fig. 6–1. A lower cost NC milling machine is not too unlike a conventional milling machine in appearance.

REDUCTION OF FLOOR-TO-FLOOR PRODUCTION TIME

As noted previously, numerical control cannot normally enhance general machining characteristics; and the equipment cannot cut any faster than is consistent with conventional practice. What actually results with a numerically controlled piece of equipment is that the machine is kept cutting a greater percentage of the time. It is not an unusual circumstance for a conventional machine tool to actually produce chips less than 20 percent of a working shift. The attainment of anything much higher would probably be considered quite noteworthy. The 80 percent of unproductive time may be attributed to:

1. Handling and setup of the work-piece, cutting tools, and fixtures
2. Constant review and checking of blueprints and/or operation (process) sheets
3. Checking part tolerances between operations
4. Operator fatigue
5. Intermittent visits to the Methods or Engineering Departments by the operator or foreman
6. Trips to the tool crib.

While the reduction of handling and setup times is probably the most significant item of savings noted above, the savings in the other areas also may be quite substantial but usually vary more widely since they depend on the overall efficiency of plant operations.

Numerical control enables the machine to *perform more work* by keeping it effectively useful a greater percentage of the time. It is not uncommon for a numerically controlled machine to log in chipcutting time of from 60 to 70 percent. Theoretically, the only non-productive time that could be attributed to a numerical control installation, excluding tape prove out, is that required for setting the holding fixture on the table, assuming one is required: loading and unloading the work-piece; setting the cutting tool; and inserting the tape for the first piece. Reductions in floor-to-floor machining time[1] of from 50 to 80 percent are considered quite normal when compared to machining practices for conventional equipment. This statistic has been proven time and again by users of numerical control equipment.

The percentage of savings will depend on the type of equipment—whether it be a mill, drill, lathe, or punch press—as well as the complexity of the parts being produced and, to a large extent, the efficiency (or inefficiency) of the old method. Table 6–1 describes the percentage of savings realized by users of NC equipment when they compared their results to previous conventional machining methods. The figures shown were the result of a survey made by the Numerical Control Society. It should also be noted that this survey was made when hardwire NC systems were common and the figures are all conservative. CNC (softwire) systems are proving to be more efficient, especially with tape prove-out and setup operations, and these percentages could justifiably be increased by at least 5 percent (i.e., 60.0 percent for drilling of aircraft parts

[1] The total time to produce a quantity of pieces of the same part, including setup time.

Table 6–1
Average Percentage Savings—NC vs Conventional Machining

	Drilling	Boring	Turning	Three-Axis Milling	Five-Axis Milling	Punching
Aircraft parts	60.0	56.6	65.9	74.2	74.6	75.0
Average of all NC users	53.1	45.7	51.7	66.4	72.8	64.9

Source: Numerical Control Society

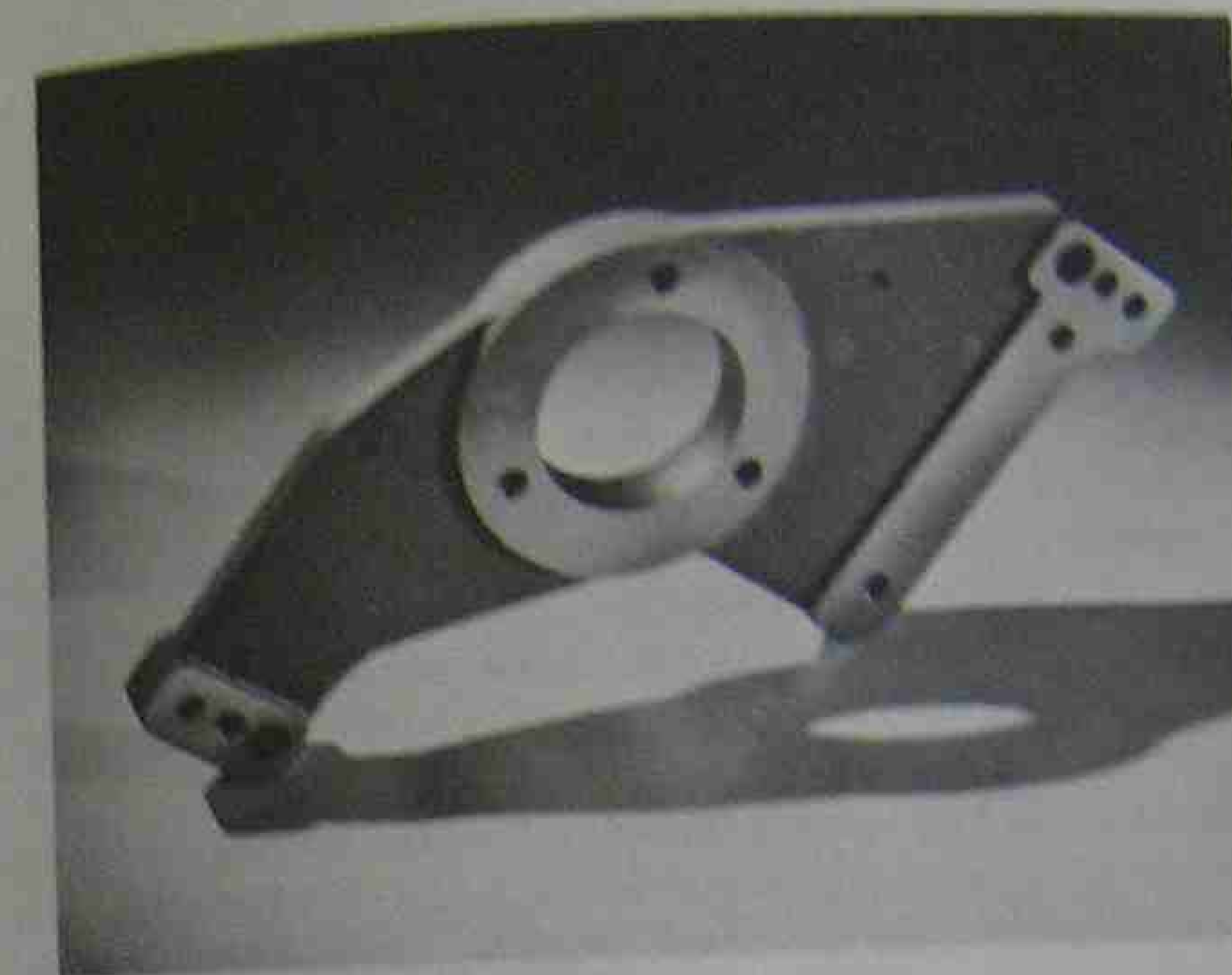
would become 65 percent). The savings that might be realized with a machining center that is capable of drilling, boring, and milling would depend on the proportion of these operations for the parts to be machined. A common breakdown for machining centers would be drilling = 50 percent; boring = 17 percent; three-axis milling = 33 percent, which results in a savings (for all uses) of 56.2 percent. Considering CNC, the figure would be approximately 61 percent and, furthermore, considering an automatic tool changer the overall percentage figure for machining center savings would reasonably approach 70 percent. This means that, if a batch of parts took 100 hours to manufacture by conventional means, the NC machining center route would take 30 hours.

As noted in Table 6–1, the largest percentage savings have been realized with milling equipment, since often the most complicated parts are handled by these machines. Drills and associated types of equipment are usually a close second with turning equipment a close third. This has been the overall general experience but it is certainly not categorical. Some operations may benefit considerably more by numerical control turning equipment than by other forms of numerical control equipment. To a large degree, savings depend on the types of parts being machined. Complex parts, for which an operator was required to "feel his way through," have generally resulted in the highest savings when switched to numerically controlled machine tools. This accounts for the higher savings shown with aircraft parts, which are generally more complex than the average part. We may assume, therefore, that in rare cases with extremely simple parts, numerical control may not be justifiable. Also, when savings are exceptionally high, it may be, though not necessarily, a reflection on the conventional method. If exceptionally old and outmoded equipment was utilized prior to numerical control, the difference realized may be quite outstanding.

EXAMPLES OF SAVINGS IN FLOOR-TO-FLOOR MACHINING TIME (POINT-TO-POINT DRILLING AND STRAIGHT CUT MILLING)

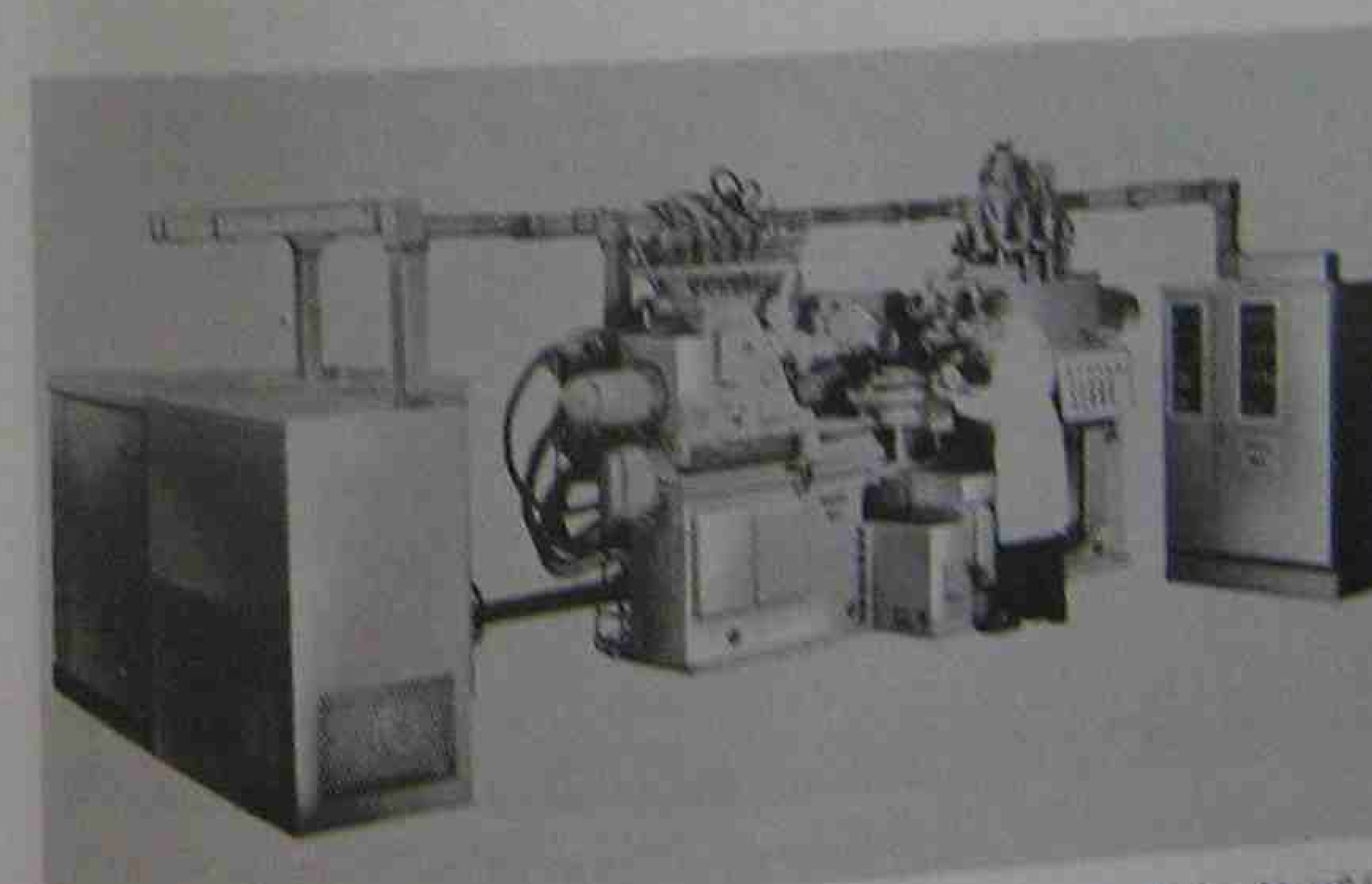
The bracket shown in Fig. 6–2 was being machined on a conventional radial drill with the aid of a standard-type drill jig. The operation was transferred to a relatively inexpensive single-spindle numerical control drill, which resulted in a floor-to-floor machine time saving of 47 percent. The lot quantity in this instance was 13 pieces.

More impressive savings may be expected from costlier point-to-point numerically controlled machine tools that employ features such as automatic tool changing, work-piece positioning, and automatic spindle speed changing. Such a machine, Fig. 6–3, produced the part shown positioned on the work-table



Courtesy of Pratt & Whitney Co., Inc.

Fig. 6–2. This bracket required 0.240 hour to machine using a conventional drill jig with the required bushed hole pattern. With a relatively inexpensive numerically controlled drill press, this time was reduced to 0.139 hour. Included is the setup time—amortized over 13 parts.



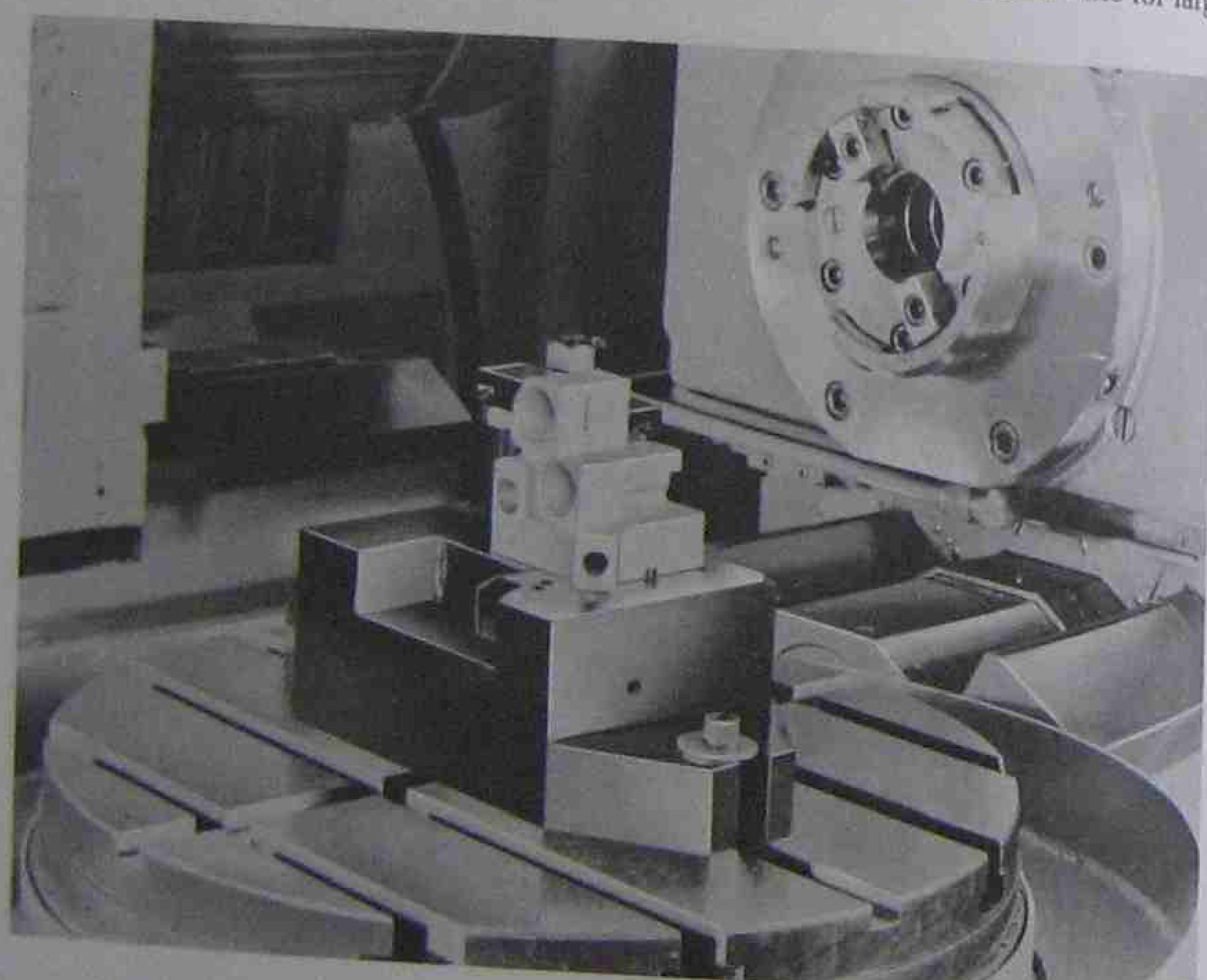
Courtesy of Hughes Industrial Systems Div., Hughes Aircraft Co.

Fig. 6–3. In addition to positioning the cutting tools, this "machining center" is capable of automatically selecting the proper tools, adjusting speeds, and rotating the work piece.

in Fig. 6–4. Machining this part by conventional methods would have required a little over an hour and would have involved eleven setups on five different machine tools. By utilizing a numerically controlled machine tool that incorporated automatic features such as tool changing, part positioning, and automatic spindle speed coding, it was possible to reduce the machining time on this part to 18 minutes or a 70 percent reduction. Savings were almost entirely attributed to the elimination of part handling and setup time on each of the five conventional machine tools and to the uninterrupted machine cutting cycles on the numerically controlled machine.

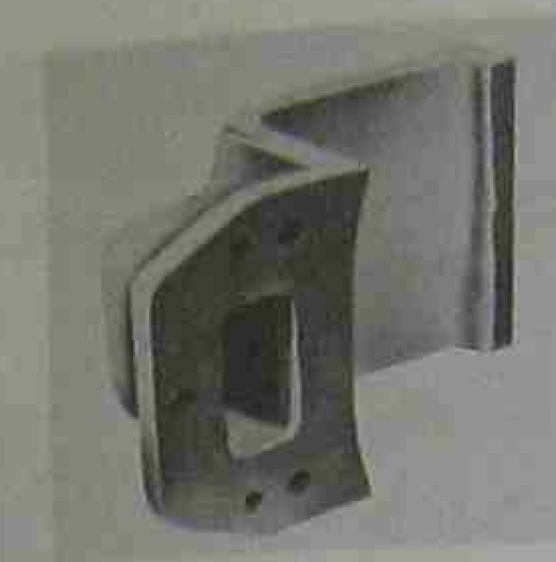
Other examples of savings in floor-to-floor machining time are shown in Fig. 6–5. These parts were produced on the machine shown in Fig. 6–6 which has similar capabilities to those of the machine shown in Fig. 6–3—such as automatic tool changing, part positioning and spindle speed changing.

Savings in Sheet Metal Fabricating (Point-to-Point) Although by far the lion's share of numerical controls has been applied to cutting tool applications, other manufacturing processes have also benefited. The punch press similar to that shown in Fig. 1–14 has the capability of reducing the fabrication time for large



Courtesy of Hughes Industrial Systems Div., Hughes Aircraft Co.

Fig. 6–4. Utilizing the machine shown in Fig. 6–3, sixteen separate machining operations are performed on the aluminum casting shown on the work table in two setups and utilizing one holding fixture. Conventional methods required five different machines, five fixtures, and eleven setups.



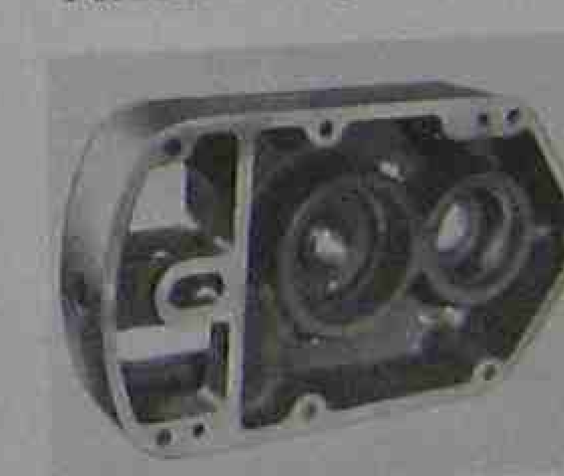
SWITCH BRACKET
(Lot size—15)
Conventional .95 hrs.
Numerical Control .21 hrs.
Percent Saving 78

MANIFOLD
(Lot size—30)
Conventional .73 hrs.
Numerical Control .36 hrs.
Percent Saving 51

GEAR HOUSING
(Lot size—10)
Conventional 12.95 hrs.
Numerical Control 1.00 hrs.
Percent Saving 92

CYLINDER BLOCK
(Lot size—10)
Conventional 4.05 hrs.
Numerical Control .40 hrs.
Percent Saving 90

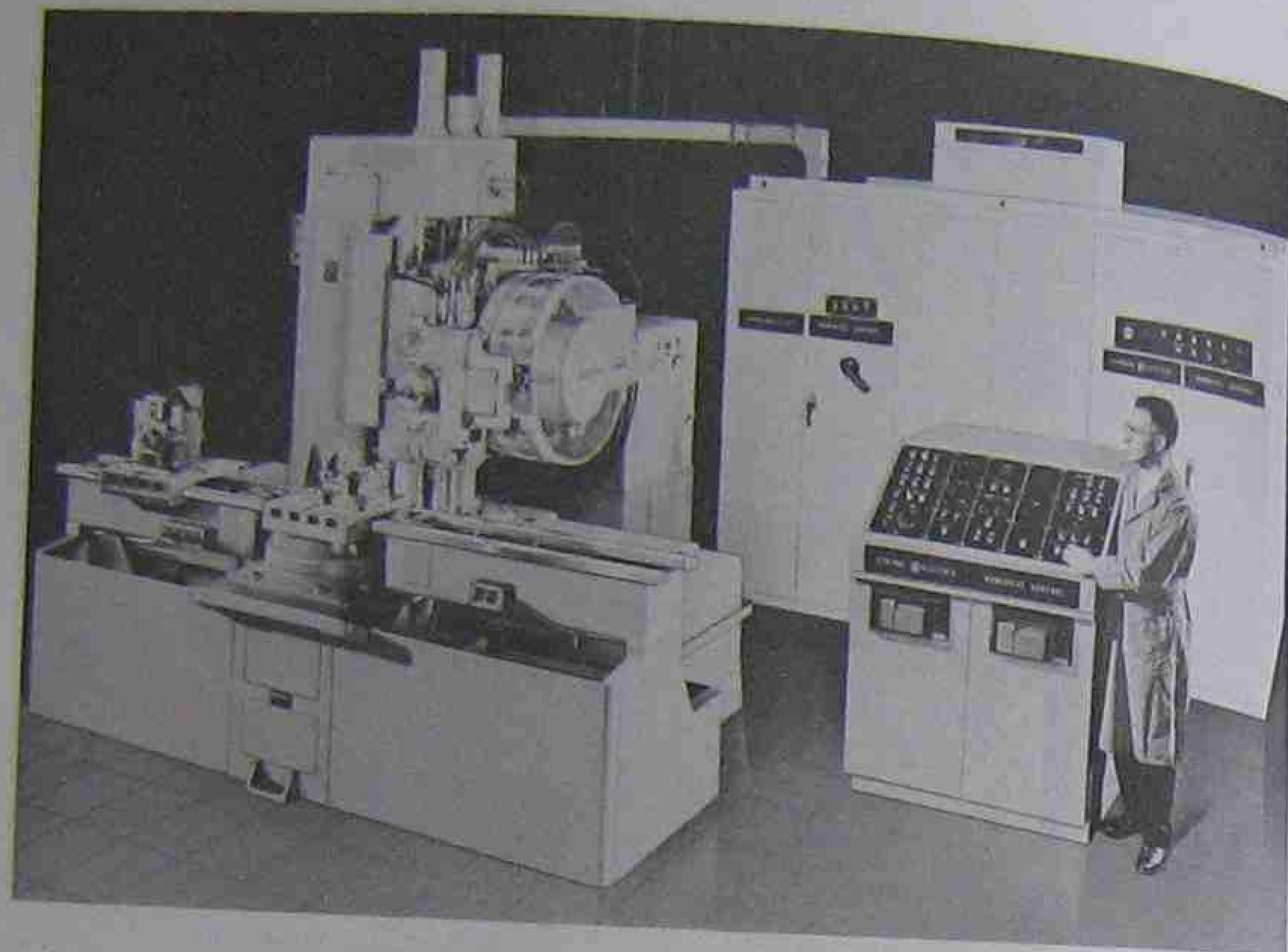
TABLE BRACKET
(Lot size—10)
Conventional 3.78 hrs.
Numerical Control .70 hrs.
Percent Saving 81



MOTOR BASE
(Lot size—15)
Conventional 2.16 hrs.
Numerical Control .65 hrs.
Percent Saving 70

Courtesy of Kearney & Trecker Corp.

Fig. 6–5. Comparisons of numerical control versus conventional machining methods in terms of floor-to-floor machining time are shown above. Setup time has been amortized over the lot sizes shown.

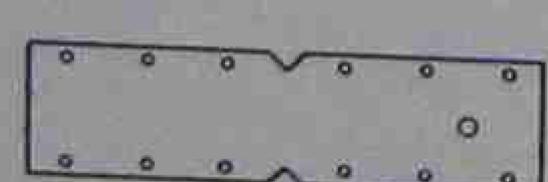


Courtesy of Kearney & Trecker Corp.

Fig. 6–6. Like the machine shown in Fig. 6–3, this automatic "machining center" has the capability of automatically changing the cutting tools and rotating the work piece to the required working position.

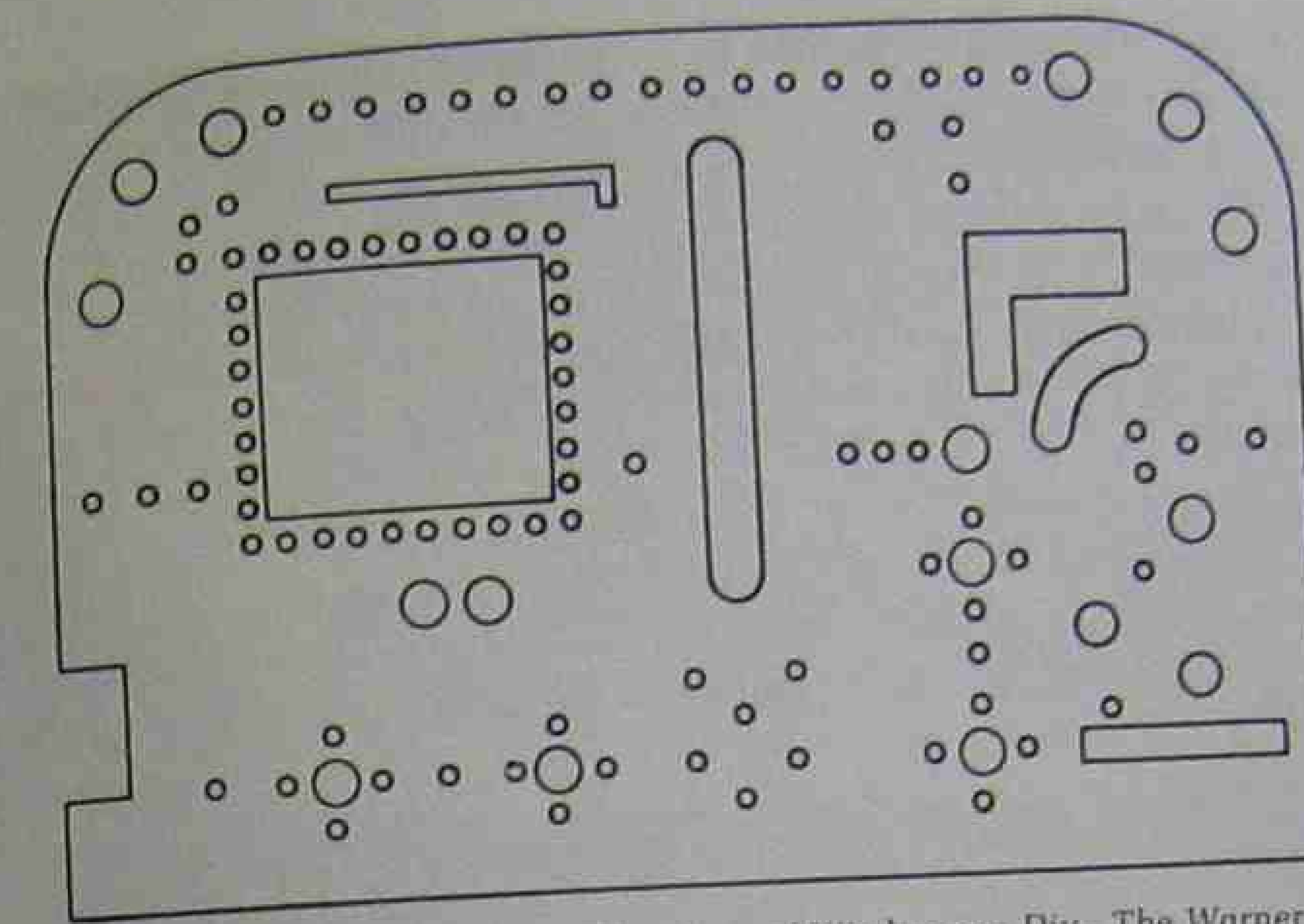
sheet metal parts by as much as 92 percent. Savings in floor-to-floor punching time, including setup, vary with relation to the *number of strokes* per part and the number of parts constituting a lot. This variance may range from approximately 35 percent savings for relatively small parts (See Fig. 6–7) to over 90 percent for a larger, more complicated part such as that shown in Fig. 6–8. It is probably worth re-emphasizing that these parts are required in relatively small lot quantities that would not justify a special stamping die incorporating all of the required hole patterns.

Examples of Savings in Machining Time (Contour Milling) As described earlier, most often the greatest savings are realized with the more complicated



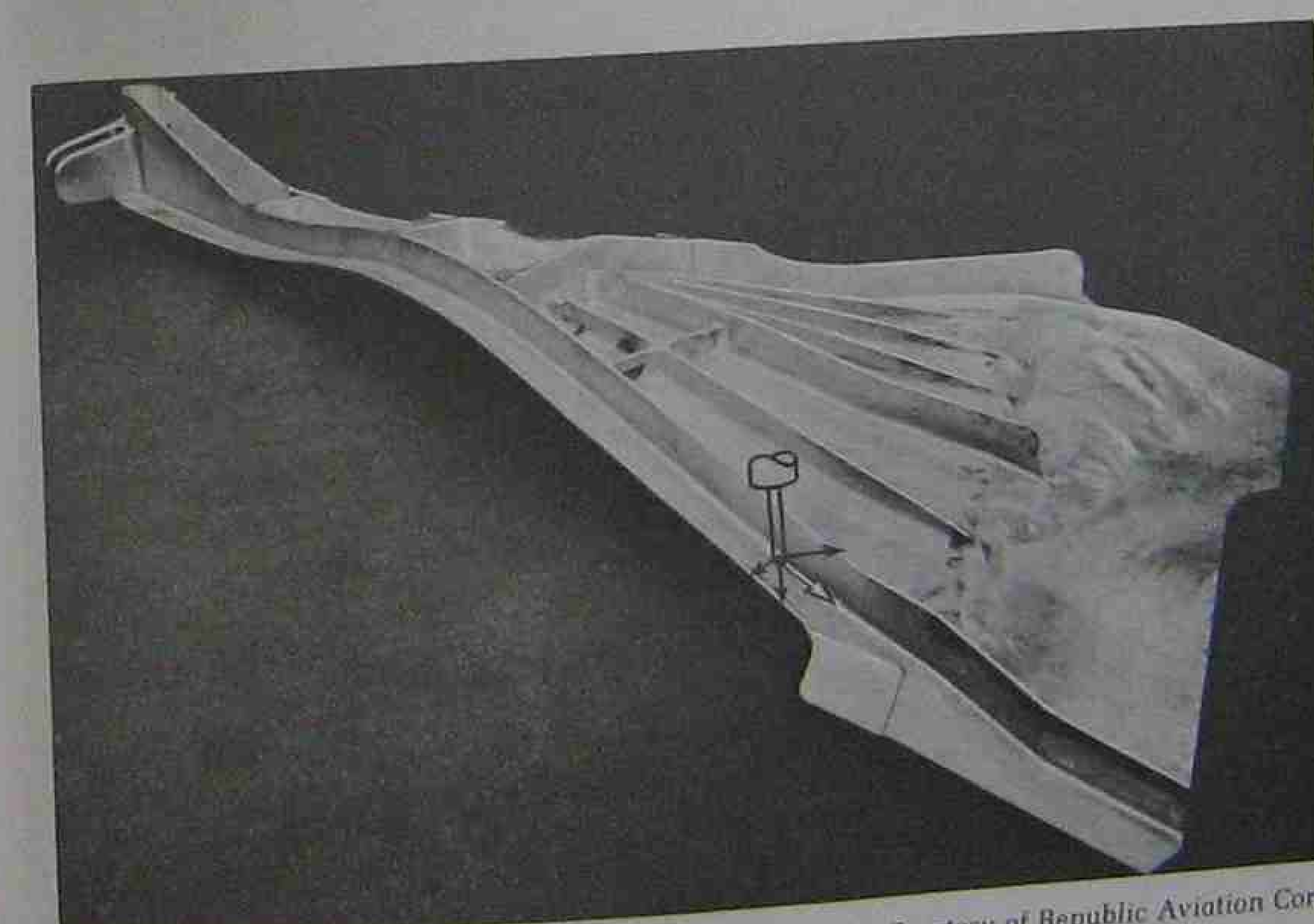
Courtesy of Wiedemann Div., The Warner & Swasey Co.

Fig. 6–7. The sheet metal part shown above measures 7¾″ × 22³⁄₁₆″ × ³⁄₁₆″ and requires 15 punches or strokes on a numerical control machine. Total time for one piece, including loading and unloading, is 45 seconds. This compares with approximately a minute and a half for conventional manually operated punching machines, assuming the operator is performing at peak efficiency.



Courtesy of Wiedemann Div., The Warner & Swasey Co.

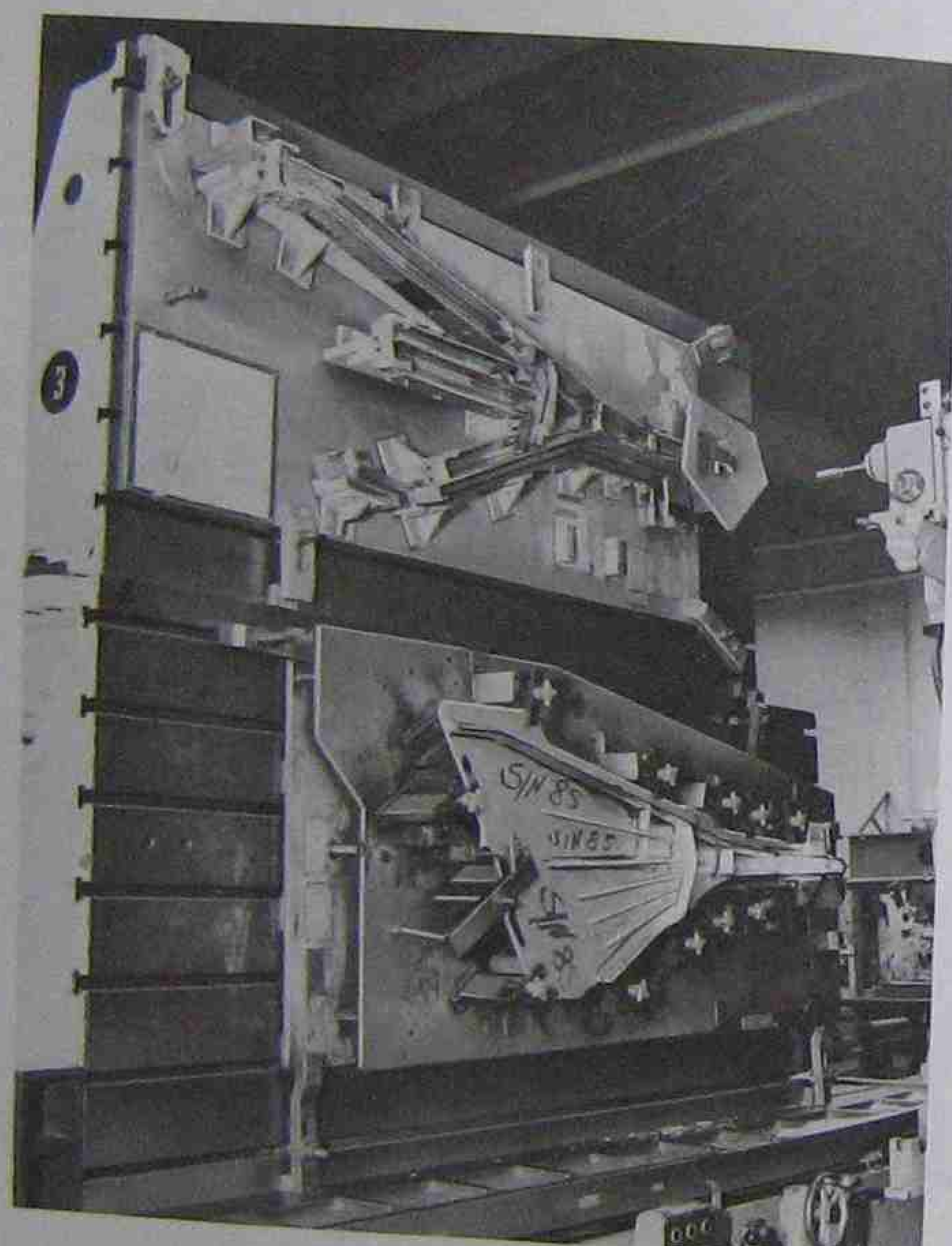
Fig. 6–8. The above part which measures 6′ × 9′ × ⅜″ and requires 180 punches took 13.4 minutes to fabricate on a numerically controlled turret punch (similar to that shown in Fig. 1–14). A conventional manually controlled punch press would have required approximately two hours.



Courtesy of Republic Aviation Corp.

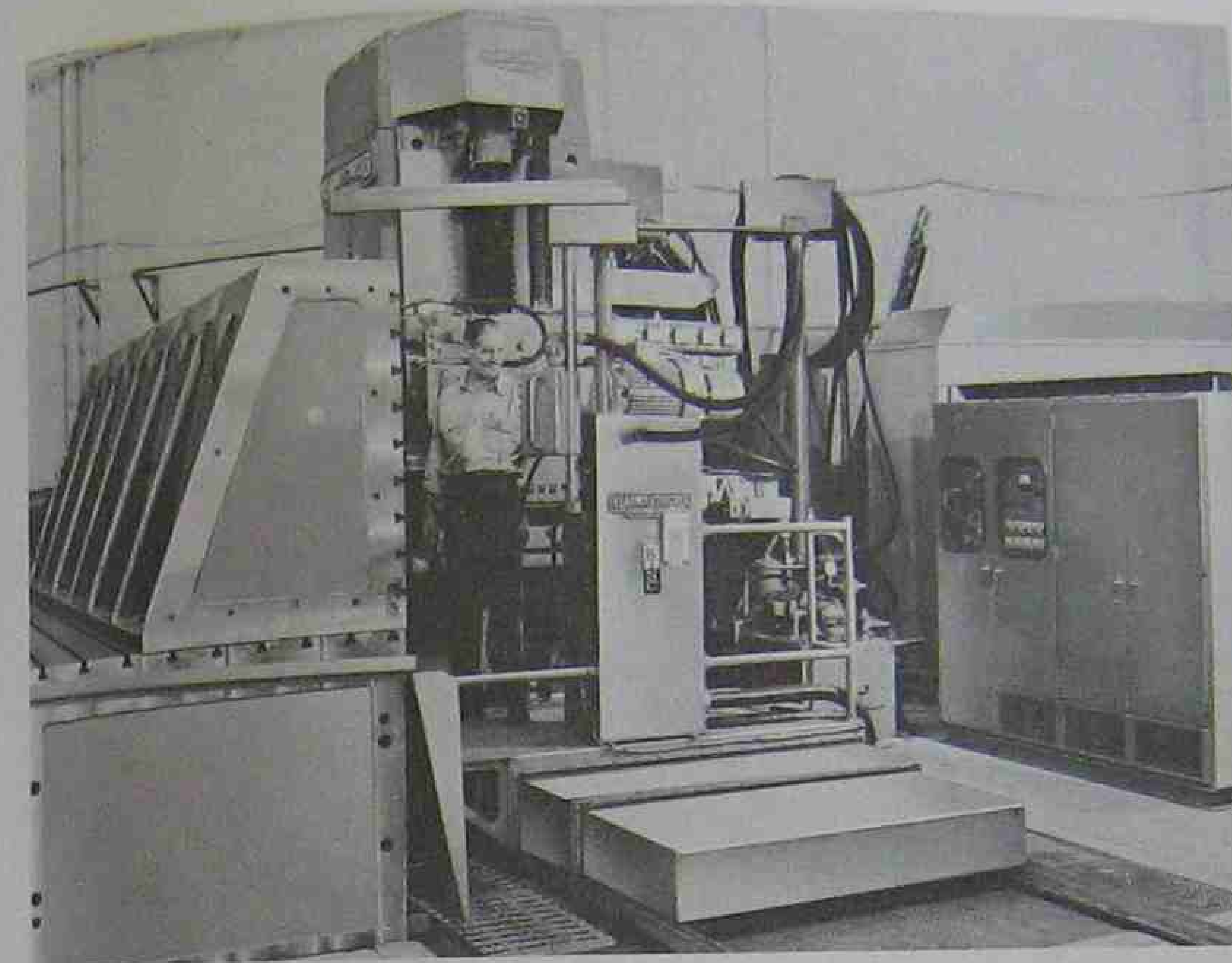
Fig. 6–9. The complex contoured part here shown measures approximately six feet in length and required 50.00 hours to machine using the conventional method shown in Fig. 6–10. When transferred to a numerically controlled profiler (Fig. 6–11) the machining time was reduced to 14.40 hours. The superimposed cutter outline shows the three cutting motion components and their resultant.

parts. This is particularly apropos of those parts requiring continuous-path or contour milling such as drop hammer or stamping dies or relatively large aircraft components. An example of a complex airframe part is shown in Fig. 6–9. In order for a milling cutter to "sculpture" the compound surfaces and numerous webs it is necessary that the motion of the cutting tool be the resultant of three simultaneous, coordinated movements. Prior to numerical control this part required 50.00 hours machining time on the conventional type tracing profiler shown in Fig. 6–10. The part to be machined, shown in Fig. 6–9, is located at the lower portion of the angle plate while the tracing model is positioned directly above it. A stylus, which is equivalent in diameter to the cutting tool, follows a contoured path on the model which results in like motions of the cutting tool located below. Immediately following the transfer



Courtesy of Republic Aviation Corp.

Fig. 6–10. In the conventional method of machining the part illustrated in Fig. 6–9, the path of the cutter is determined by the outline of the model shown above it on the angle plate.



Courtesy of Republic Aviation Corp.

Fig. 6–11. When the part shown in Fig. 6–9 was transferred to the numerically controlled machine shown above, savings of 71 percent in machining time resulted. The part, here obscured from view, is positioned on the vertical face of the angle plate at the left of the machine.

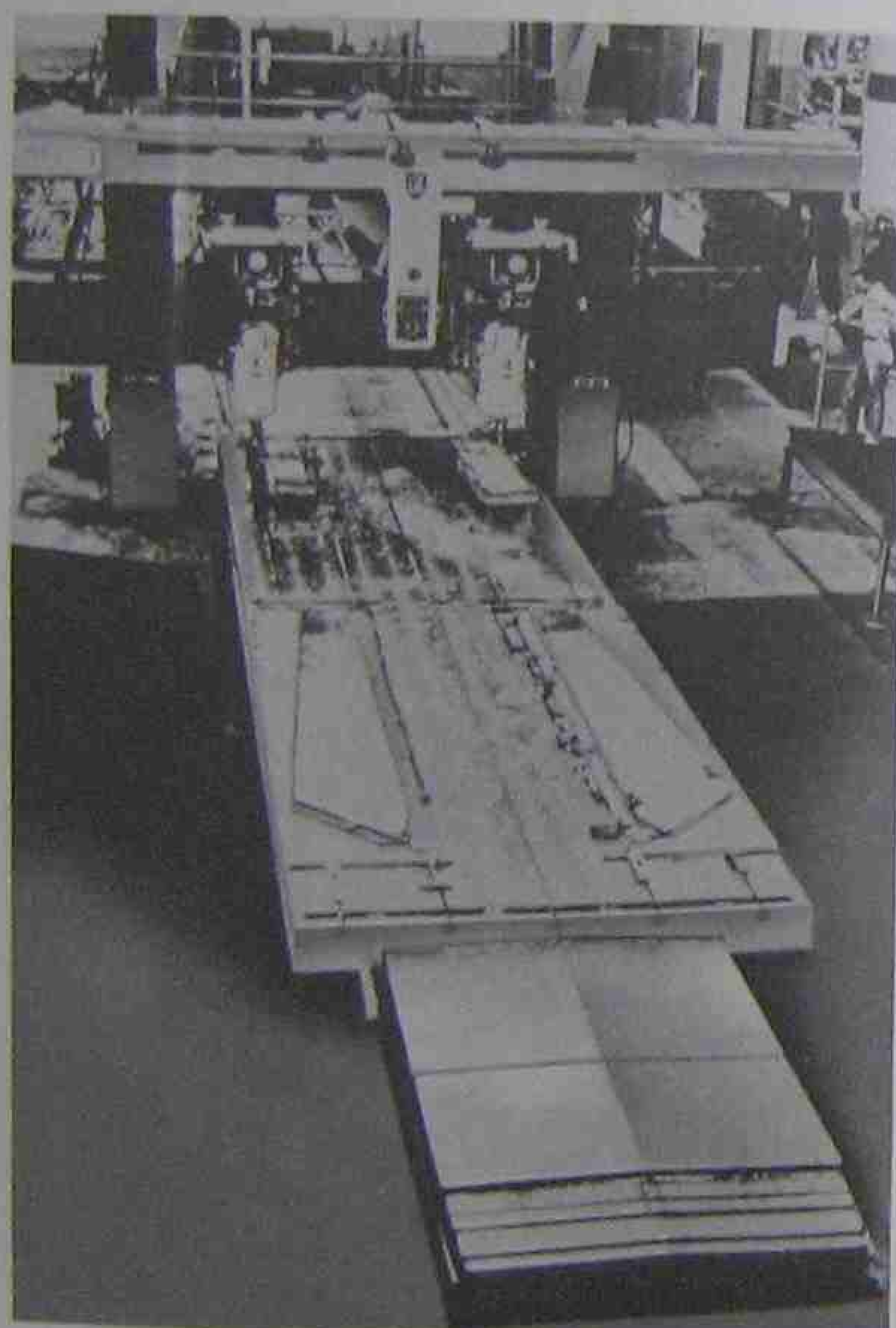
Courtesy of Republic Aviation Corp.

Fig. 6–12. When this part was transferred to a numerically controlled profile milling machine (shown in Fig. 6–11) only 27 percent of the time taken to mill it by the conventional tracing method was required. The part is brought to the machine as a rough forging and is machined to the required tolerances.

of the machining of this part to the numerically controlled machine shown in Fig. 6–11, the floor-to-floor machining time was reduced from 50.00 hours to 14.40 hours—a savings of 71 percent.

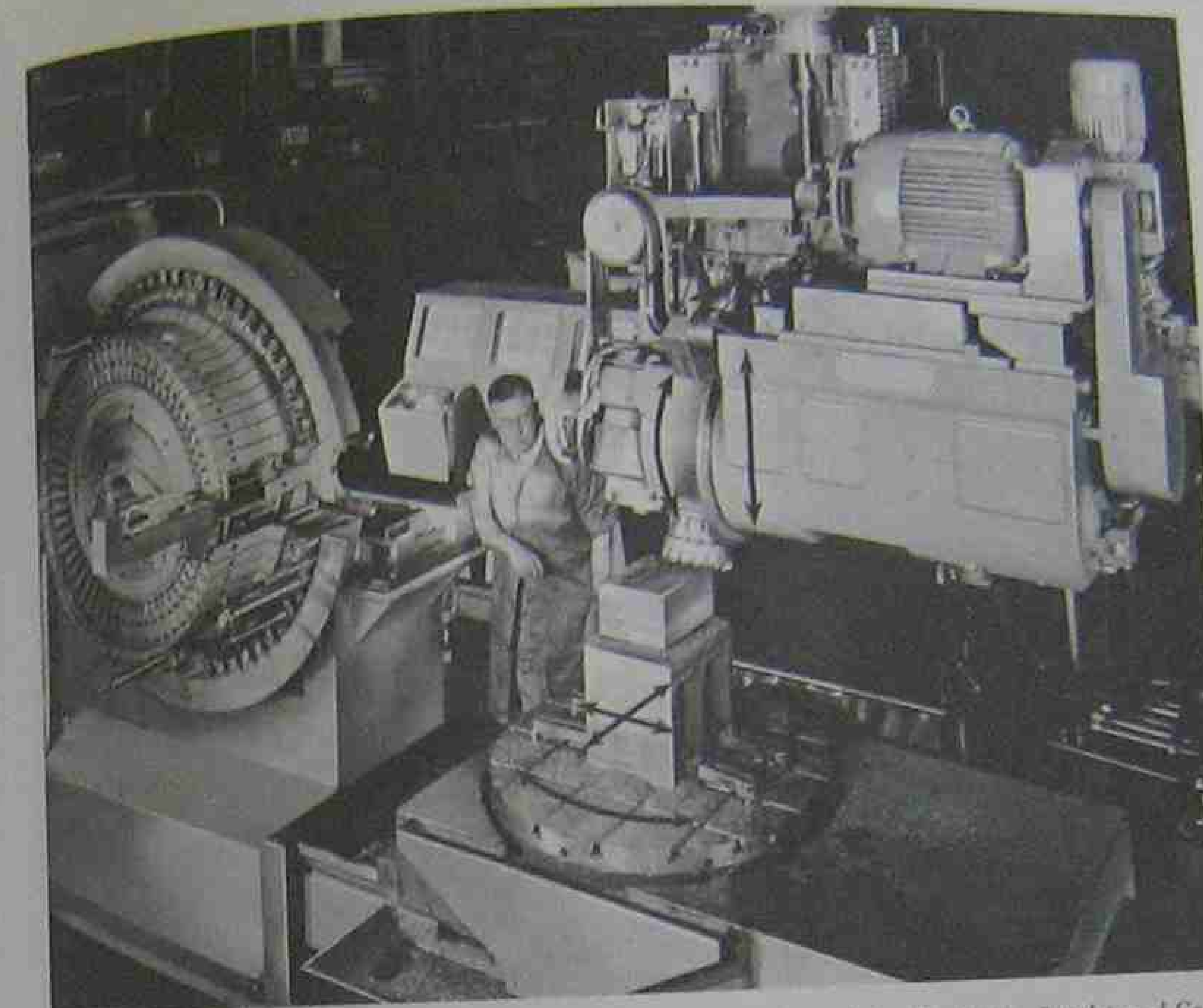
Another contour part that resulted in typical savings is shown in Fig. 6–12. When transferred from a tracer controlled machine to the numerically controlled profiler shown in Fig. 6–11, machining time was reduced from 8.06 hours to 2.16 hours—a savings of 73 percent.

Transfer of the two large airplane skins from conventional machining methods to the skin milling machine shown in Fig. 6–13 reduced the hours required from 30.00 to 2.20—a 93 percent savings. It should be pointed out that, in this instance, two parts are machined simultaneously by the two heads.



Courtesy of Republic Aviation Corp.

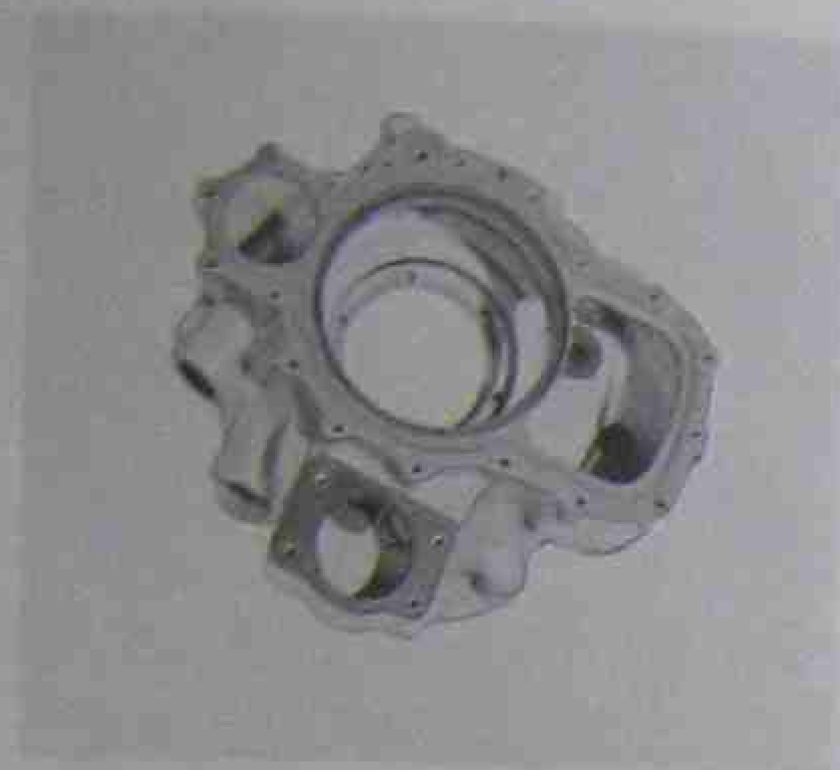
Fig. 6–13. Using two heads operating simultaneously, this large numerically controlled milling machine reduced the machining time of the parts shown by 93 percent.



Courtesy of Sundstrand Machine Tool Div., Sundstrand Corp.

Fig. 6–14. This machining center incorporates the full range of automatic features and capabilities noted in the "machining centers" shown in Figs. 6–3 and 6–6 in addition to contouring capability. The all-angle rotary table and head, which are tape controlled, enable machining on all but one surface (bottom) of an irregularly-shaped part without changing the setup.

Further savings in milling the contoured parts noted above could have been achieved had the machine tools been equipped with facilities for automatically changing cutting tools. Such a contour machining center is shown in Fig. 6–14. As with most continuous-path numerical control machining, this equipment has the capabilities of performing such point-to-point operations as drilling, boring, reaming and tapping, in addition to contour milling. Another feature of this machine which has a considerable effect in reducing setup time is the capability of being able to machine on five sides of a cube-shaped part without changing the setup. This is accomplished by the machine's ability to rotate the table onto which the part is mounted as well as being able to rotate the cutting head (denoted by the superimposed arrows shown on the machine tool in Fig. 6–14). All cutting operations in five planes can be performed automatically and simultaneously on parts such as those shown in Fig. 6–15. Automatic tool and spindle speed changing is performed from coded instructions on the numerical control tape. Also, if desired, the work-piece can be remotely positioned on a pallet and automatically positioned onto the table. (See Fig. 6–16.)

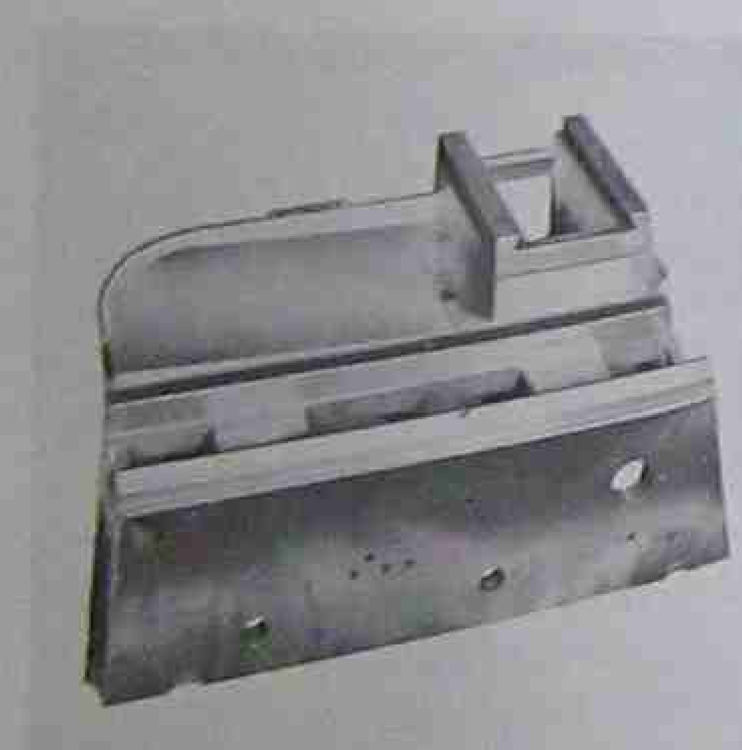


ALUMINUM HOUSING
Milled
Drilled
Tapped
Counterbored

Conventional Machining	12.0 hrs.
Numerical Control Machining	1.9 hrs.
Percent Saving	84

CAST IRON FEED BOX
Bored
Drilled
Tapped
Reamed
Counterbored

Conventional Machining	32 hrs.
Numerical Control Machining	6 hrs.
Percent Saving	81

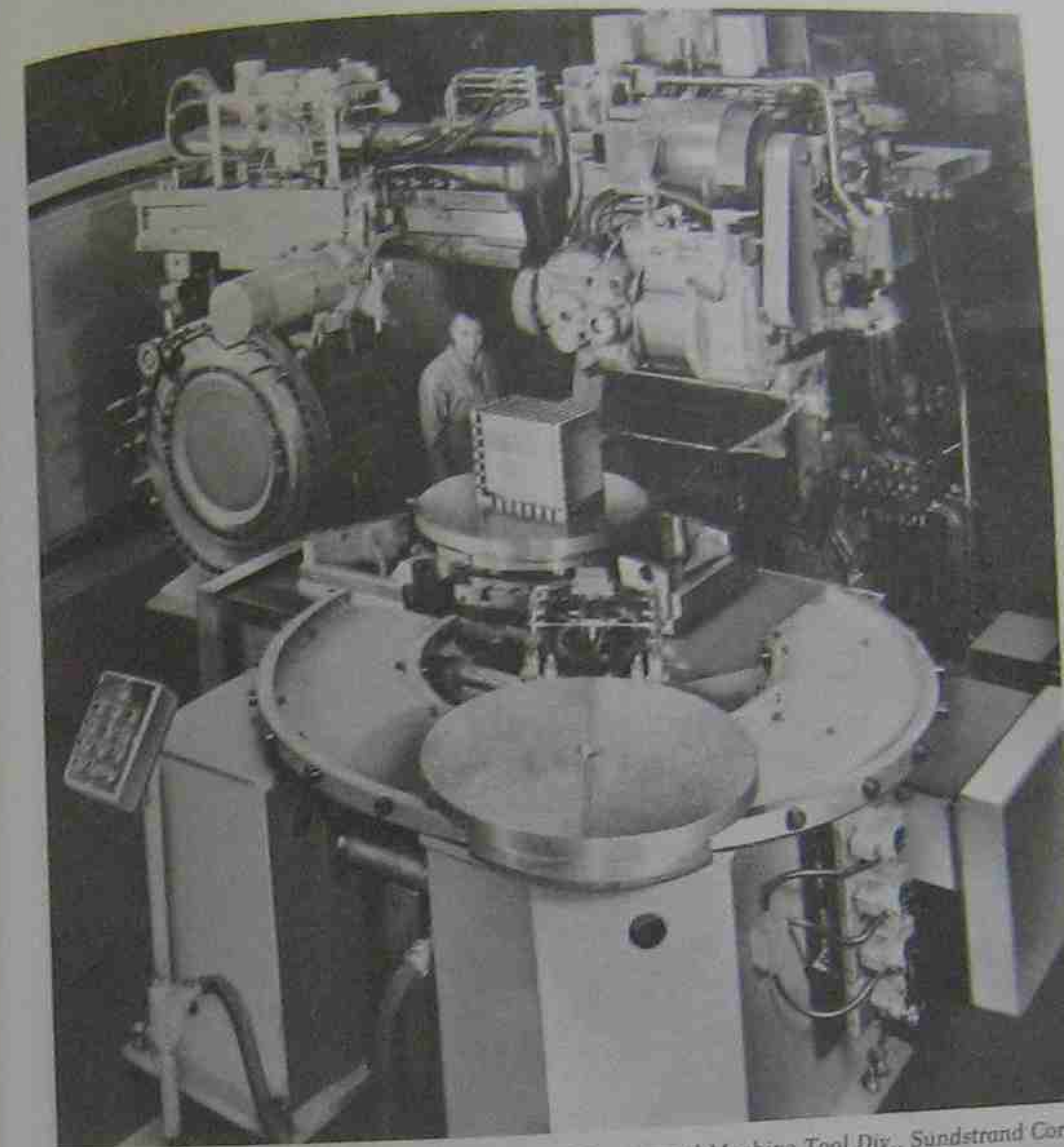


CAST IRON BASE
Drilled
Tapped
Bored
Milled

Conventional Machining	20.4 hrs.
Numerical Control Machining	7.4 hrs.
Percent Saving	64

Courtesy of Sundstrand Machine Tool Div., Sundstrand Corp.

Fig. 6–15. These parts were machined on the numerically controlled "machining center" shown in Fig. 6–16. The aluminum housing incorporates both PTP and contouring cuts.



Courtesy of Sundstrand Machine Tool Div., Sundstrand Corp.

Fig. 6–16. The setup for the next part may be made on the circular pallet, shown in the foreground, while the machine is cutting. A code on the tape automatically moves the completed part off the work table and moves the new part into position.

This enables the machine tool to continue operation while the setup is being made. Examples of savings realized when machining complex shapes with this machine are shown in Fig. 6–15.

Examples of Savings in Machining Time (Turning) Although not as popular at first as numerically controlled mills and drills, lathes have rapidly gained popularity and now vie with machining centers for the number of units sold per year. Savings have proven to be quite substantial even when compared to tracing type lathes which normally require only a simple flat pattern template. Again, the most substantial portion of the savings may be attributed to a reduction in setup and machining times, for although tracing lathes may be considered "automatic," the same condition exists as with tracer milling equipment in that without numerical control, numerous adjustments and checks must be made by the operator during the cutting cycle. This is particularly true for roughing cuts and for threading operations.

The numerically controlled lathe shown in Fig. 6–17 was responsible for reducing the cost to machine the die punch shown in Fig. 6–18 from $953 to $533. This is particularly significant since, in this instance, only one part was machined. Another example is shown in Fig. 6–19. The large double-end electric motor shaft which measures 4¾ inches in diameter and 60 inches long was completely machined in 36 minutes on the numerically controlled lathe shown in Fig. 6–17, including 10 minutes handling time. Manual preparation of the tape required approximately 4 hours. On a conventional lathe, the method used previously, it took approximately 8 hours to machine the part. The top view in Fig. 6–19 shows the shaft prior to the numerically controlled lathe operation. The lower view shows the completed shaft. Previous operations included facing, center drilling and spotting for the steady rest.

Courtesy of R. K. Le Blond Machine Tool Co.

Fig. 6–17. This early numerically controlled contouring lathe being operated at an Army R&D center eliminates the necessity of a tracing template. The resultant savings in total machining costs are due chiefly to a reduced setup time and an uninterrupted cutting cycle.

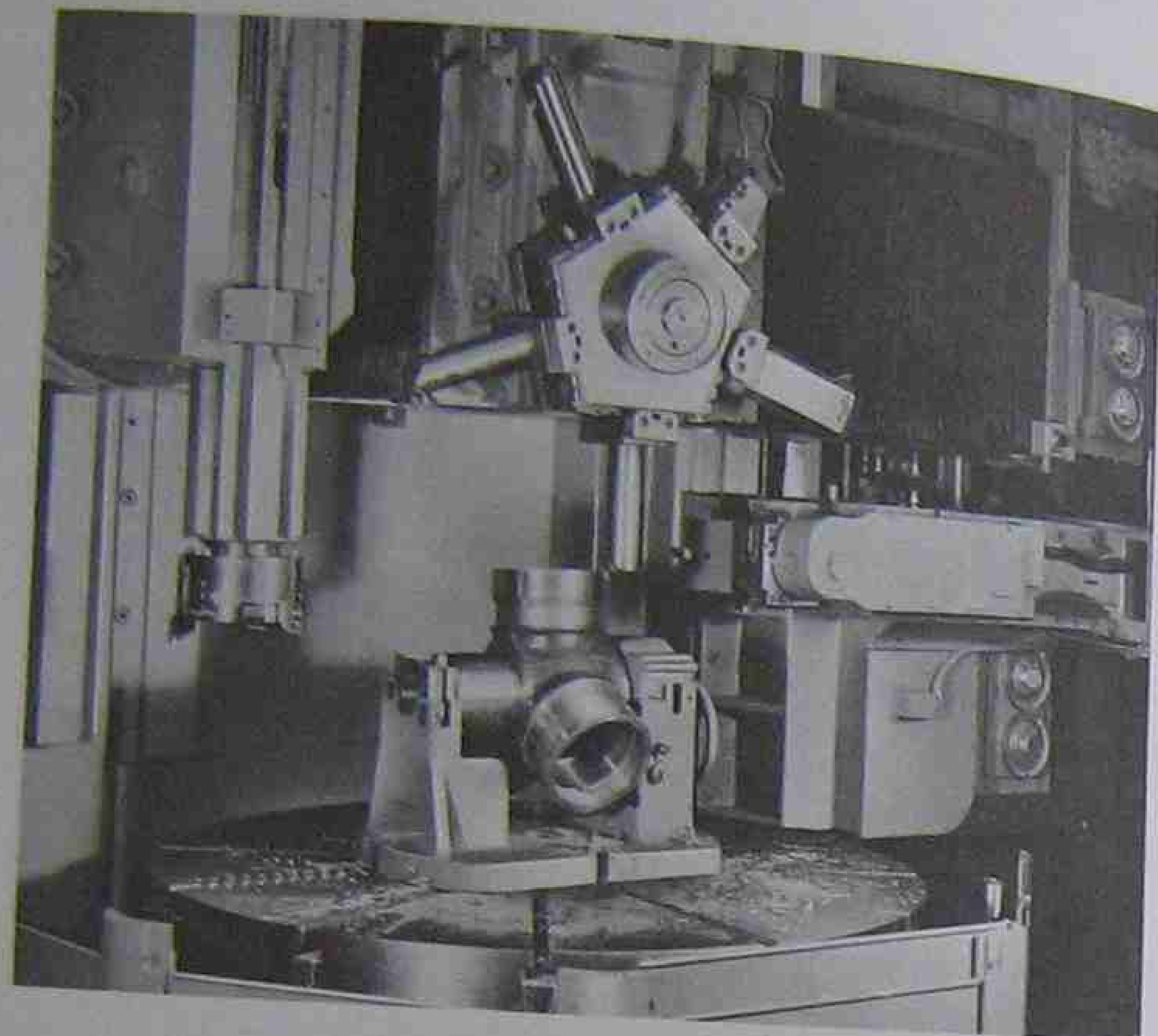
Courtesy of R. K. Le Blond Machine Tool Co.

Fig. 6–18. This part was machined at a cost of $533 on the numerically controlled lathe shown in Fig. 6–17. This compares with $953 on a conventional tracing lathe.

Numerically controlled vertical turret lathes (boring mills) have also gained prominence in the turning field. The part shown on the machine tool table in Fig. 6–20 required twenty-two handling operations to produce utilizing conventional equipment. When transferred to the numerically controlled machine shown in Fig. 6–20, the number of operations was reduced to three.

Courtesy of R. K. Le Blond Machine Tool Co.

Fig. 6–19. This 60-inch long electric motor shaft was machined in 36 minutes on the lathe shown in Fig. 6–17. This compares with approximately 8 hours by conventional means. Tape preparation time, without the aid of a computer, took approximately 4 hours.



Courtesy of The Bullard Co.

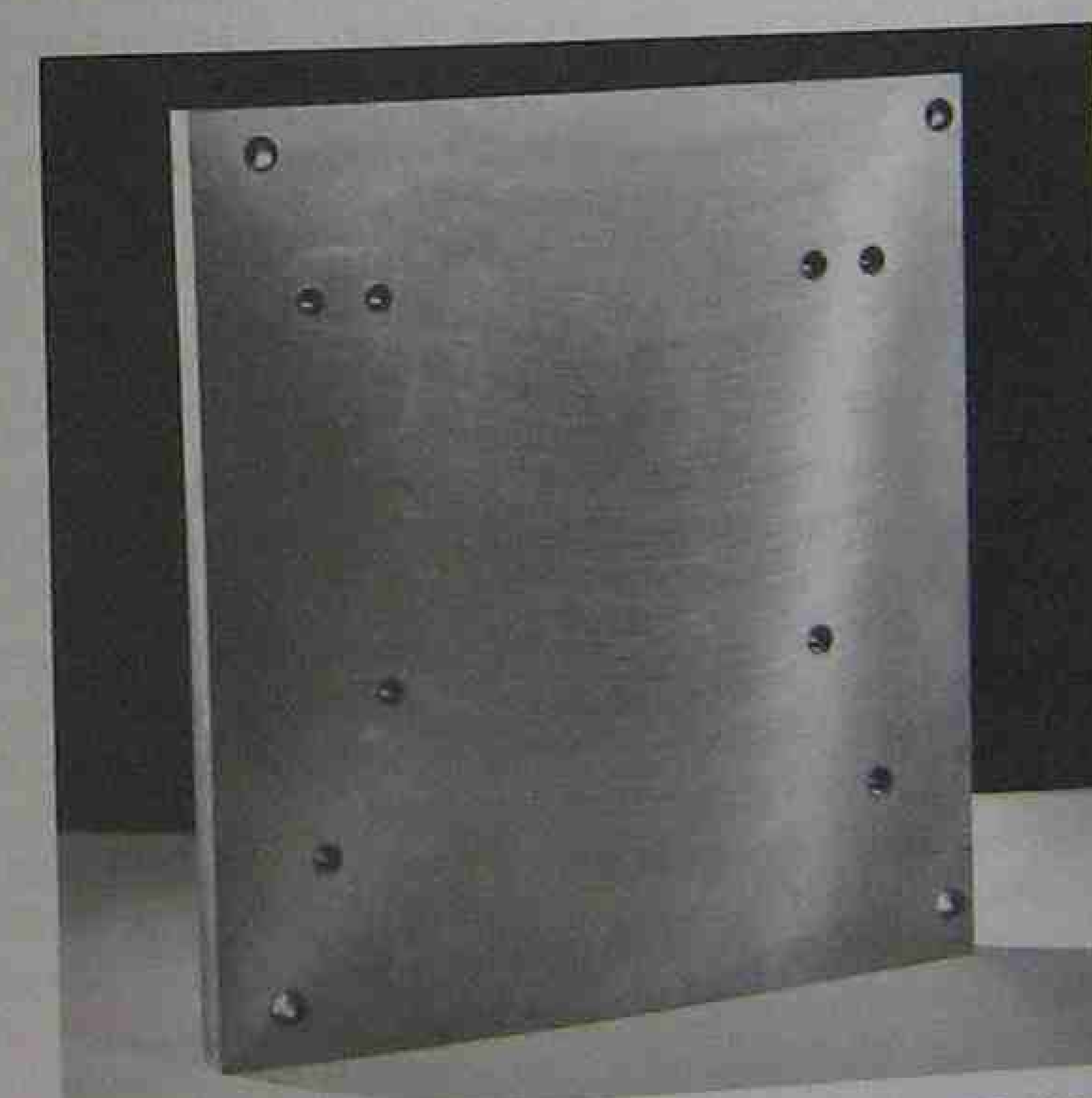
Fig. 6–20. Machining operations were reduced from twenty-two to three when the part shown on the numerically controlled vertical turret lathe was transferred from a conventional method of manufacturing.

SAVINGS IN TOOLING COSTS

The cost to prepare a numerical control tape may be compared to the cost of preparing a piece of material for manual machining. This usually entails scribing measurement lines on a semi-finished part or raw material which has been "painted" with a blue substance, and which a machinist uses as a visual guide. Of course, material may be machined directly without marking, by utilizing calipers, micrometers, scales, and measuring indicators on the machine tool. This, however, is admittedly time consuming and affects the overall cost of the part regardless of whether the scribing of the material or the numerous manual checks are considered to be in the tooling or machining category. If the cost to prepare a part for machining without the aid of guide tools, such as a drill jig or tracing template, were compared to the cost for preparing a tape, numerical control would most likely have a distinct edge even for machining a single piece assuming computer assist is used. This has been proven time and again when it was found more economical to machine single templates, which were to be used on tracing machines, by numerical control rather than by manual means. We might conclude therefore that, while manual machining methods (such as with standard drills, lathes, and knee mills) may be more economical than using tracing equipment on single pieces or exceptionally small lot sizes, numerical control is generally more economical from a tooling standpoint than either manual or tracing means or when utilizing a drill jig. As with the savings realized when comparing machining hours, the savings when comparing the preparation of a tape to the manufacture of hard tooling such as a box drill jig or a tracing model or flat template increases with the complexity of the part since, in turn, it affects the complexity of the guide fixture.

Dollar savings in tooling costs when switching to numerical control, while perhaps not as substantial on the average as savings in machining costs, have nevertheless proven quite significant. The cost to prepare a tape may range from 10 to 70 percent of the cost to prepare comparable hard tooling, with perhaps the mean running around 50 percent.

Examples of Savings in Tooling Costs Consider the example shown in Fig. 6–21. Manual *preparations* for machining this part required .65 hour on a jig borer and .62 hour on a radial drill for a lot of 13 pieces or a total of 1.27 hours. Tape preparation required .80 hour—a savings of 37 percent over the manual method which did not employ guiding fixtures of any kind. Figure 6–22 shows the steps required by both the manual and numerical control methods. Although numerical control requires one additional step, the sum of steps 1 and 2 for numerical control requires less time than the *preparations* required for



Courtesy of Pratt & Whitney Co., Inc.

Fig. 6–21. Preparation time for drilling, countersinking and tapping ten holes and drilling and counter-boring two holes in the above part was reduced by 37 percent when the job was transferred from conventional means to a simple numerically controlled drill press.

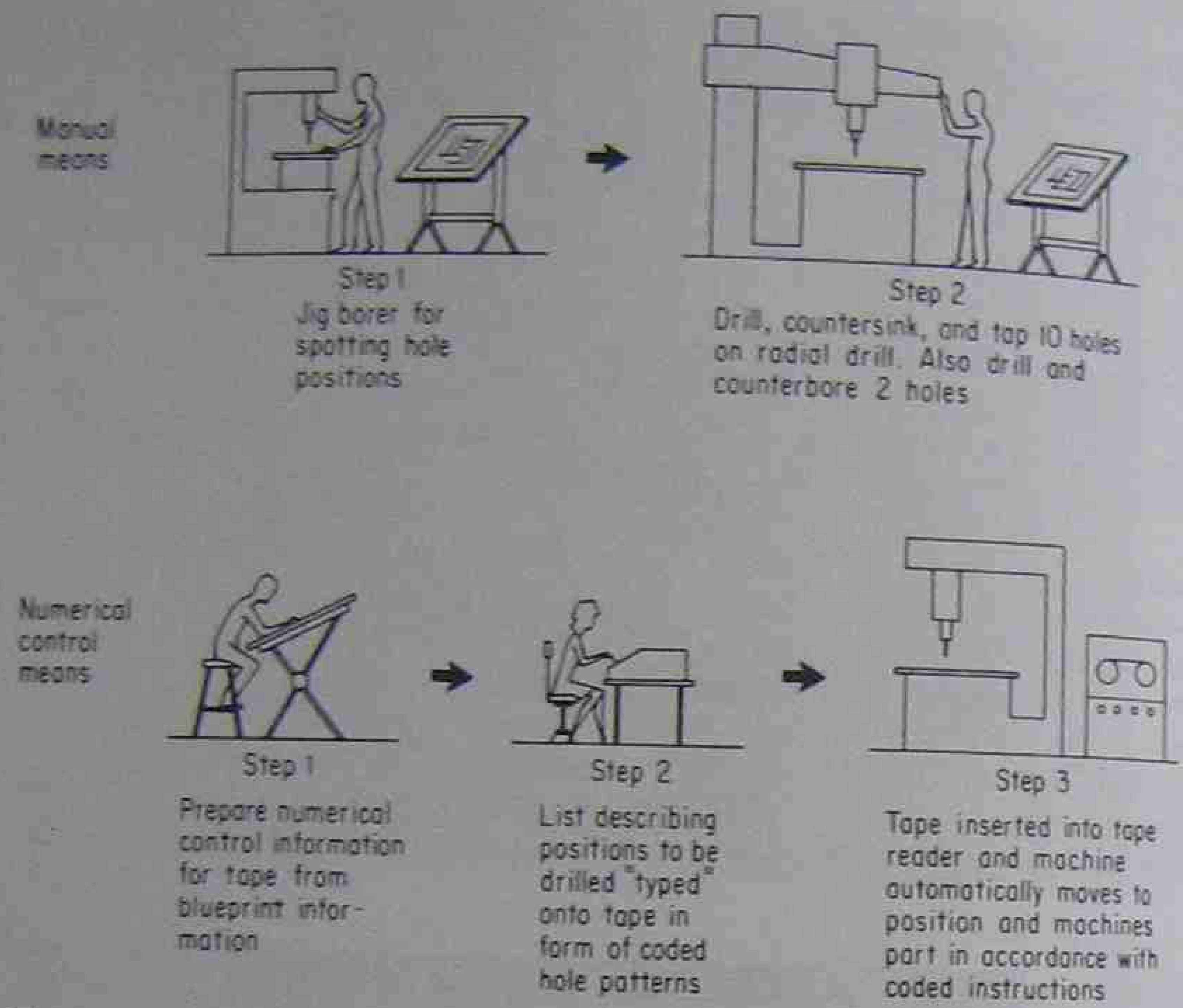
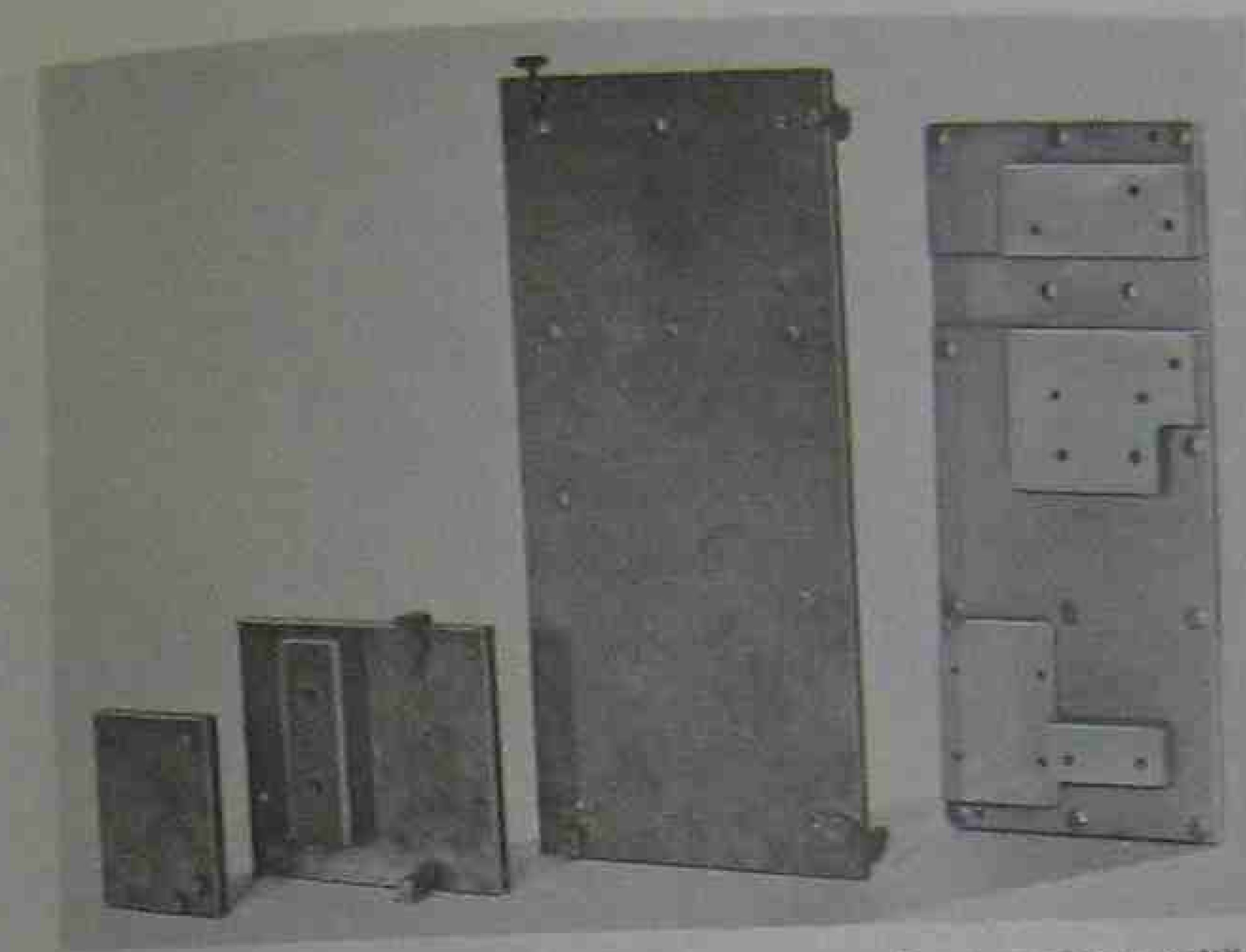


Fig. 6–22. In this illustration, the tooling time which may be considered as the preparation time required for machining by manual means has been replaced by Steps 1 and 2 under the numerical control method. By performing part of the required preparation away from the machine site, the numerically controlled drill will have a far greater utilization factor. Also only one machine will be required instead of two.

machining the part in both steps under the manual method. Furthermore, the total time to produce the part is less with numerical control and only one machine is required instead of two.

Figure 6–23 shows three drill jigs which were used to machine the part shown at the right-hand side of the illustration. The cost of these three fixtures amounted to $1,094.35 for labor (at $21.60/hr.) and $238.70 for material for a total tooling cost of $1,333.05. Time required to prepare the tape amounted to 1.66 hours. If the same hourly rate is used as with the tool fabrication shop (i.e., $21.60/hr.), the cost to prepare the tape would amount to $35.97[2]—a savings of approximately 97 percent. It is because of savings of this magnitude that drill jigs have become almost extinct in those shops using numerical

[2] Actually the hourly charge would be less since shop overhead, when considering equipment, is generally higher than assigned office costs. In this instance the only equipment necessary for the numerical control method would be a relatively inexpensive tape preparation machine (ranging from $1,000 to $4,000—depending on manufacturer and model).

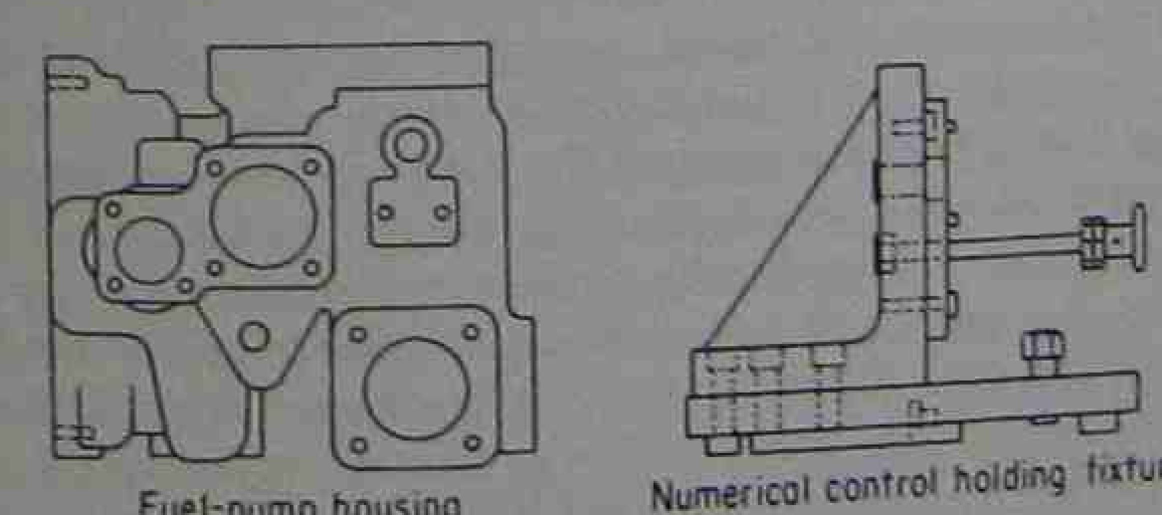


Courtesy of Pratt & Whitney Co., Inc.

Fig. 6–23. Manual tape preparation time required to replace the three drill jigs on the left was 1.66 hours or, when calculated at $21.60 per hour, amounted to $35.86. Design and fabrication cost of the drill jigs amounted to $1,301.85, or approximately 37 times as expensive.

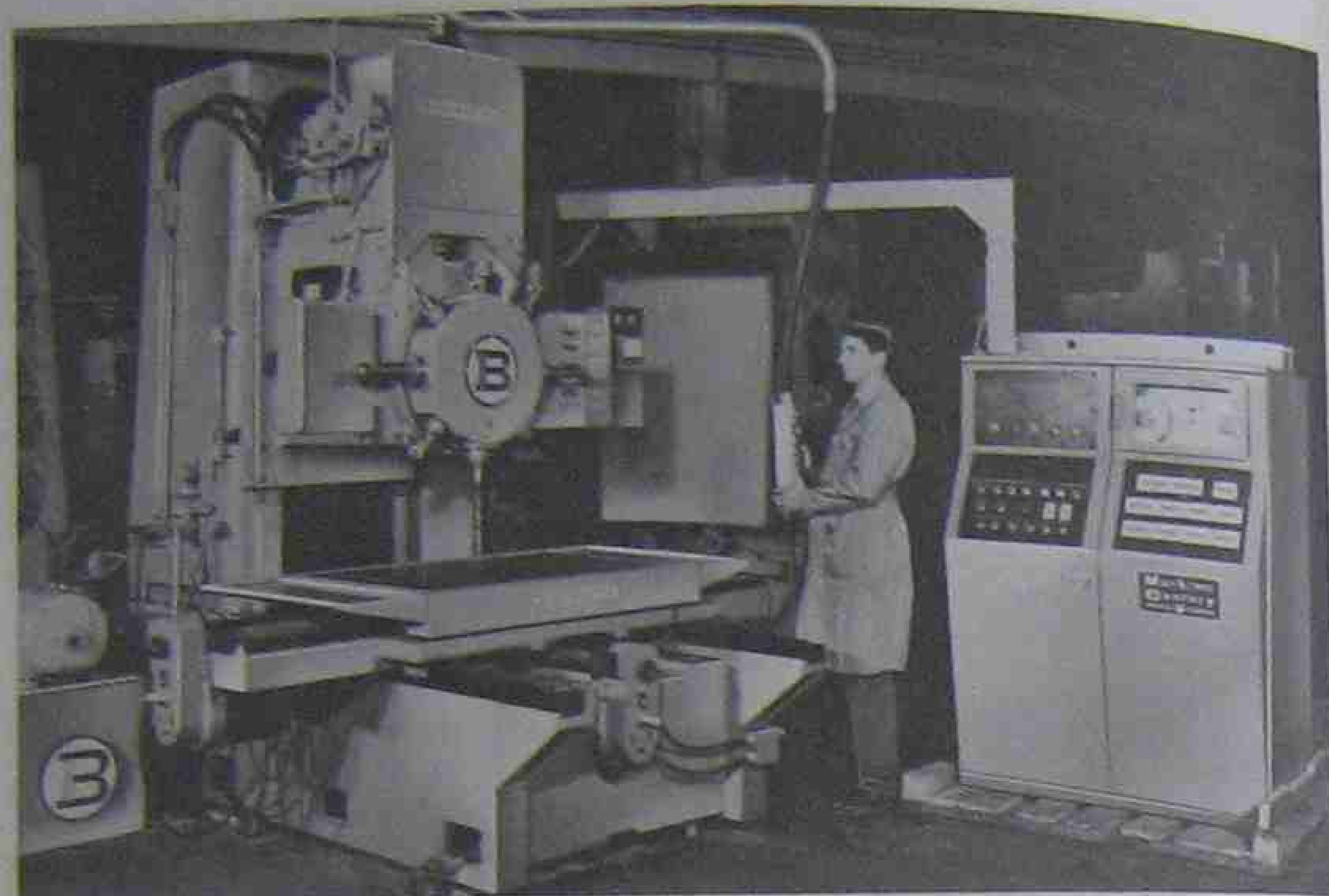
control. Machining time is also reduced; however, the savings, in this instance is not outstanding (e.g., .78 hr./pc. by conventional drilling techniques vs. .68 hr./pc. numerical control or a savings of approximately 13 percent).

Another example of savings in tooling costs is shown in Fig. 6–24. Three drill jigs plus supplementary items were necessary under conventional machining methods. The cost was $11,411 and the time to design and fabricate the fixture was approximately 7 weeks. Machining of the part on the numerically controlled turret drill press shown in Fig. 6–25, resulted in the necessity for only one holding fixture (also shown in Fig. 6–24) costing $1,247 and requiring only 2½ weeks to design and manufacture.



Courtesy of Metalworking Magazine

Fig. 6–24. This holding fixture was the only hard tooling required for machining the fuel pump housing shown, with numerical control. This fixture costs $1,247 as compared to three drill jigs costing $11,411 which were required for conventional drilling.



Courtesy of Burgmaster Corp.

Fig. 6–25. This turret drilling machine automatically indexes to the proper drill from coded instructions on the tape. An example of tool cost savings realized in conjunction with this machine is shown in Fig. 6–24.

The preparation of tape is also generally less expensive than the cost to fabricate relatively simple, flat templates required when tracing lathes are used. A template for the part shown in Fig. 6–18 required ¾ *hour* to design and 16 hours to fabricate. Tape preparation required approximately 3 hours to plan and less than ¾ hour to process on the small, manually operated tape punching machine similar to that shown in Fig. 6–26; a savings of approximately 78 percent.

Still another example of savings in tooling costs can be illustrated by comparing the tape shown in Fig. 6–27 with the tracing model shown in Fig. 6–28, which may also be seen in the upper portion of the angle plate of the machine described in Fig. 6–10. In this particular instance, planning and preparation of the tape required only about one-half the time required to plan and fabricate the tracing model. In addition, engineering changes can be made much more readily on the tape and storage of the tape is, quite obviously, far easier than with the model. Periodic inspections are also required of the model whereas no inspection is required of the tape after it has been successfully proven out.

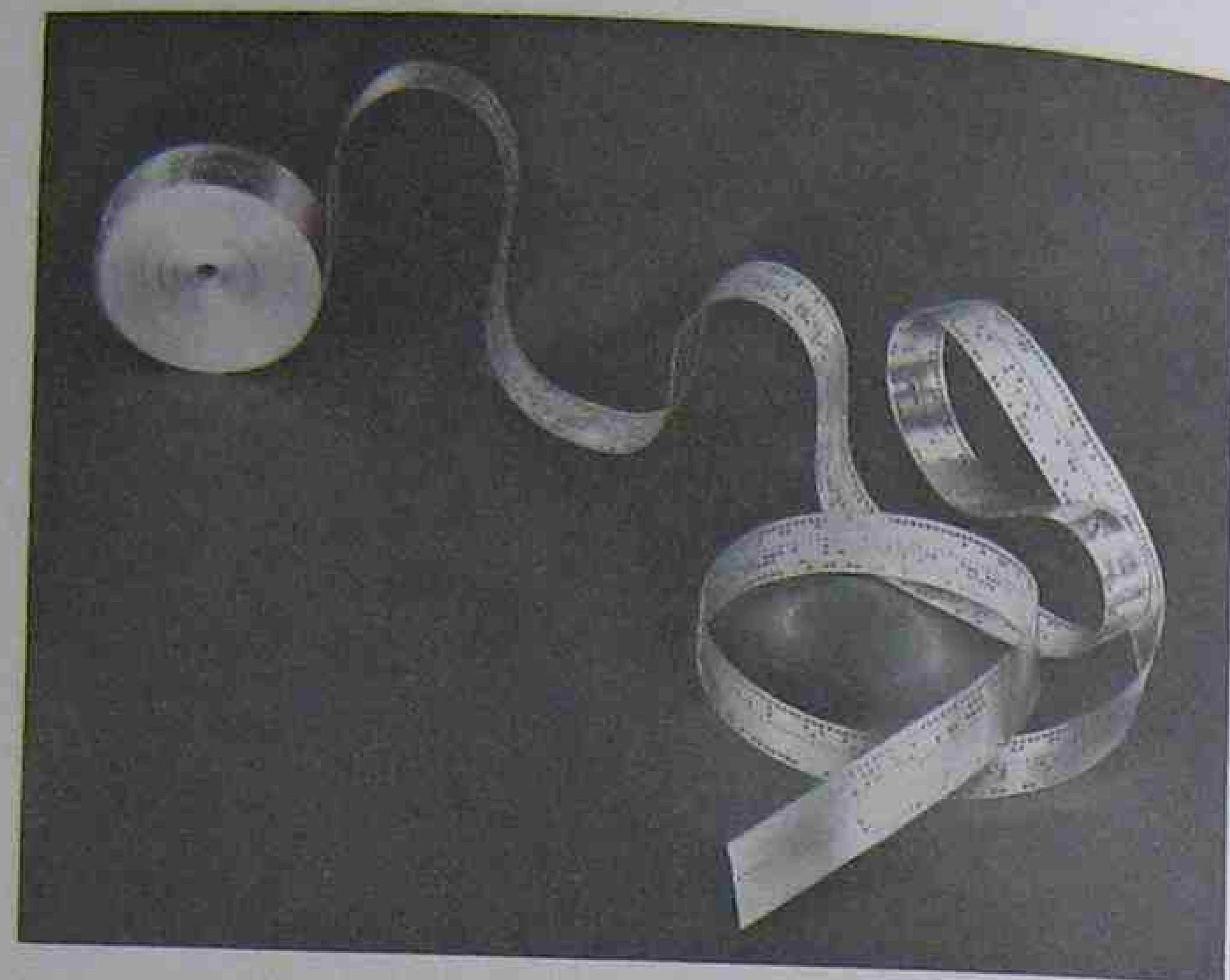
OTHER NUMERICAL CONTROL BENEFITS

Accuracy and Repeatability If we are to define *accuracy* as the ability of a machine tool or other piece of equipment to position to an indicated value and *repeatability* as the ability of a machine tool to repeat its accuracy, then numerically controlled equipment is far superior to its predecessors. Heretofore,

Courtesy of the Teletype Corporation

Fig. 6–26. The device shown here is a common type for preparing machine control tapes. The keyboard is similar to that of a standard typewriter. The punched tape produced is shown at the left of the unit.

accuracies were limited primarily to the operator or, in the case of tracing machines, to the accuracy of the model or template. Because of the human element that interceded between the blueprint and the machine it was impossible to expect maximum part accuracy. Since in numerical control, the data which are transmitted to the moving elements of a machine tool are precisely the same as shown on the blueprint, intermediate sources of human error are eliminated and the accuracy of the part will depend almost solely on the accuracy of the machine itself. Inherently, the electronic control system is far more accurate than the mechanical machine tool it operates. Most numerical control machines are guaranteed to hold a *positioning* accuracy of ±.001" on any one axis and repeat to within .0005". It would take an expert machinist to perform this well and consistently with conventional equipment. Also, if there is a will and a requirement, NC machines can be made far more accurate. The machine in Fig. 6–29 is capable of holding part accuracies of five millionths of an inch (±.000005"). A second NC machine is being installed at this

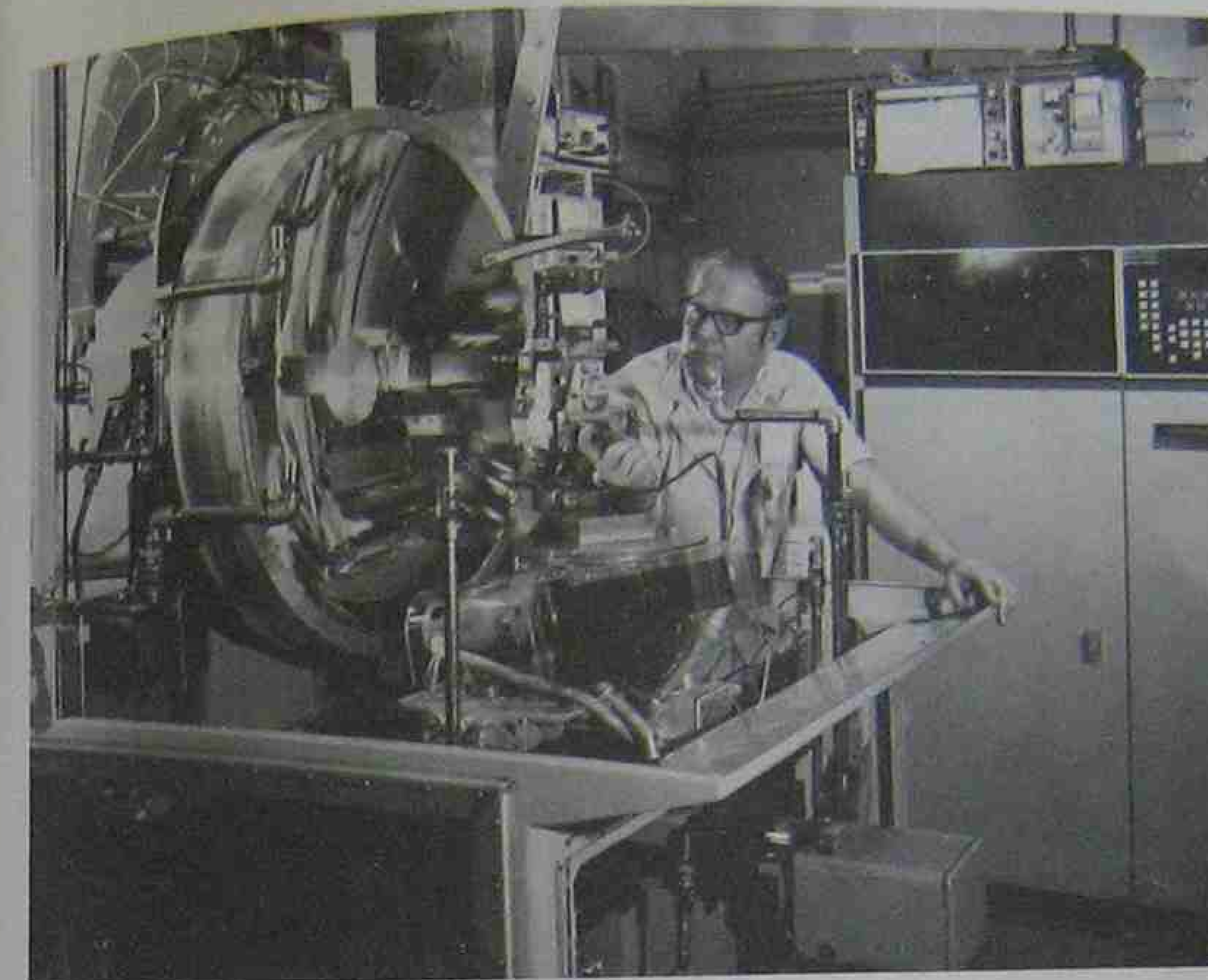


Courtesy of Republic Aviation Corp.

Fig. 6–27. This 4-inch diameter roll of tape replaced the six foot tracing model shown in Fig. 6–28. In addition to saving approximately one half the cost of the tracing model, engineering changes were made far more readily with the tape and the storage problem was also much simpler.

Courtesy of Republic Aviation Corp.

Fig. 6–28. The tracing model shown above and also in Fig. 6–10 was replaced by the tape shown in Fig. 6–27.



Courtesy of Lawrence Livermore National Laboratory

Fig. 6–29. The NC machine shown above is capable of holding part accuracies of five millionths of an inch (±.000005").

facility that will have the capability of holding a positioning accuracy of one-half millionth of an inch (±.0000005"), and a third machine is being planned that will hold one-ten millionth of an inch (±.0000001"). These machines will be used at the Lawrence Livermore Laboratory to machine optical reflectors that must conform in accuracy to wavelengths of light.

A question may justly arise as to the value of high accuracy. If requirements are such that high accuracy is not essential, then certainly economics do not warrant its achievement. Accuracy, however, is closely allied with repeatability when considering the successful mating or interchangeability of parts after machining. Accuracy may also be a prominent factor when attempting to keep the weight or the amount of lost material to a minimum. If, for example, the closest that a tolerance could be held on a milling machine was ±.005 inch and if a part did not allow negative tolerance, then the *nominal* dimension sought would have to be .005 inch larger than required. If, however, it was possible to maintain ±.002 inch with a numerically controlled machine tool, then the nominal path could be .003 inch closer to the ideal surface thus saving the difference in weight or being able to start with less material in the case of expensive metals.

Almost Total Elimination of Scrap Since the machining cycle with numerical control equipment is almost entirely automatic, except for changing to the prescribed cutting tool, chances of operator error are remote, providing, of course, the operator is capable and has been properly trained. At many numerical control installations the common practice of scrap *allowances* has been abandoned. As an example, out of a total of over 2000 pieces of the relatively complicated part shown in Fig. 6–9, only two were scrapped. Also in these two instances the cause of the damage was attributed to a faulty design condition in the electronic control system which has since been corrected. It should be remembered, however, that no numerically controlled machine is completely foolproof, and improper setups, incorrect tool selections, and other human errors can occur.

Lower Capital Investment If it is assumed that the productivity of a numerically controlled machine is three or four times that of a conventional machine, then, in addition to operating savings, fewer machines are required to produce the equivalent number of parts. Although a numerically controlled machine may cost considerably more than its conventional counterpart, its higher productive capabilities alone, not considering other benefits such as reduced tooling costs and shorter lead time, will generally far more than outweigh its higher cost. As an illustration, consider the cost of a numerically controlled boring mill at approximately $300,000. A comparable manually operated machine might run in the neighborhood of $200,000. If the output of the numerically controlled machine is three to four times that of the conventional machine, then initial numerical control investment alone is justifiable—not considering operating savings such as in the areas of manpower, tooling, accuracy, and floor space.

Less Floor Space If it is assumed that one numerically controlled machine has the capability of producing the work of 3, 4, or even 5 conventional machines, space requirements are far less with numerical control equipment. The reduction of area may not be directly proportional to the reduction of the number of machines since a numerically controlled machine is usually somewhat larger and requires associated units such as electrical and electronic equipment cabinets. Figure 6–30 offers a reasonable comparison of floor space requirements for a numerically controlled machining center as shown in Fig. 6–6, and comparable conventional equipment. In this particular instance numerical control requires approximately one-half the space. However, there are many cases where only one-third or one-quarter of conventional equipment space is required. This is particularly true with the newer CNC machines where control systems can be included as an integral part of the machine tool (see Fig. 1–11).

Savings in Facility Requirements Because of the fewer numerically controlled machines required to sustain equivalent conventional production, less power, lubricant, coolant, and, possibly, air are required. Although the numerically controlled equipment will be operating a greater percentage of the time, its

movements will be far more efficient than with manual- or tracer-operated equipment.

Reduced Inspection Time Because of inherently better accuracies and particularly the repeatability attainable with numerical control, it is possible to take fuller advantage of statistical quality control methods. The probability of follow-on pieces not conforming to within extremely close proximity of the first piece is small. Therefore, if inspection of the first piece indicates a satisfactory result, it is very reasonable to assume that the remaining pieces will be acceptable. Should a discrepancy exist, it can normally be traced to an operator error such as inserting an incorrect diameter tool or failing to target correctly or to the inadvertent movement of the holding fixture or wear of the tool. The more automatic features incorporated on a piece of numerically controlled equipment, such as tool changing, therefore tend to reduce the chance of error on follow-on pieces. Prior to numerical control it was necessary that 135 points be checked on the part shown in Fig. 6–9. With numerical control the number of points which had to be checked was reduced to 5, and these were points at which the operator was required to insert a different cutting tool. A check was therefore required not of the machine tool but of the operator.

If we consider that, quite often, inspection time may equal or exceed actual machining time, considerable savings can be realized in this area. It is seriously

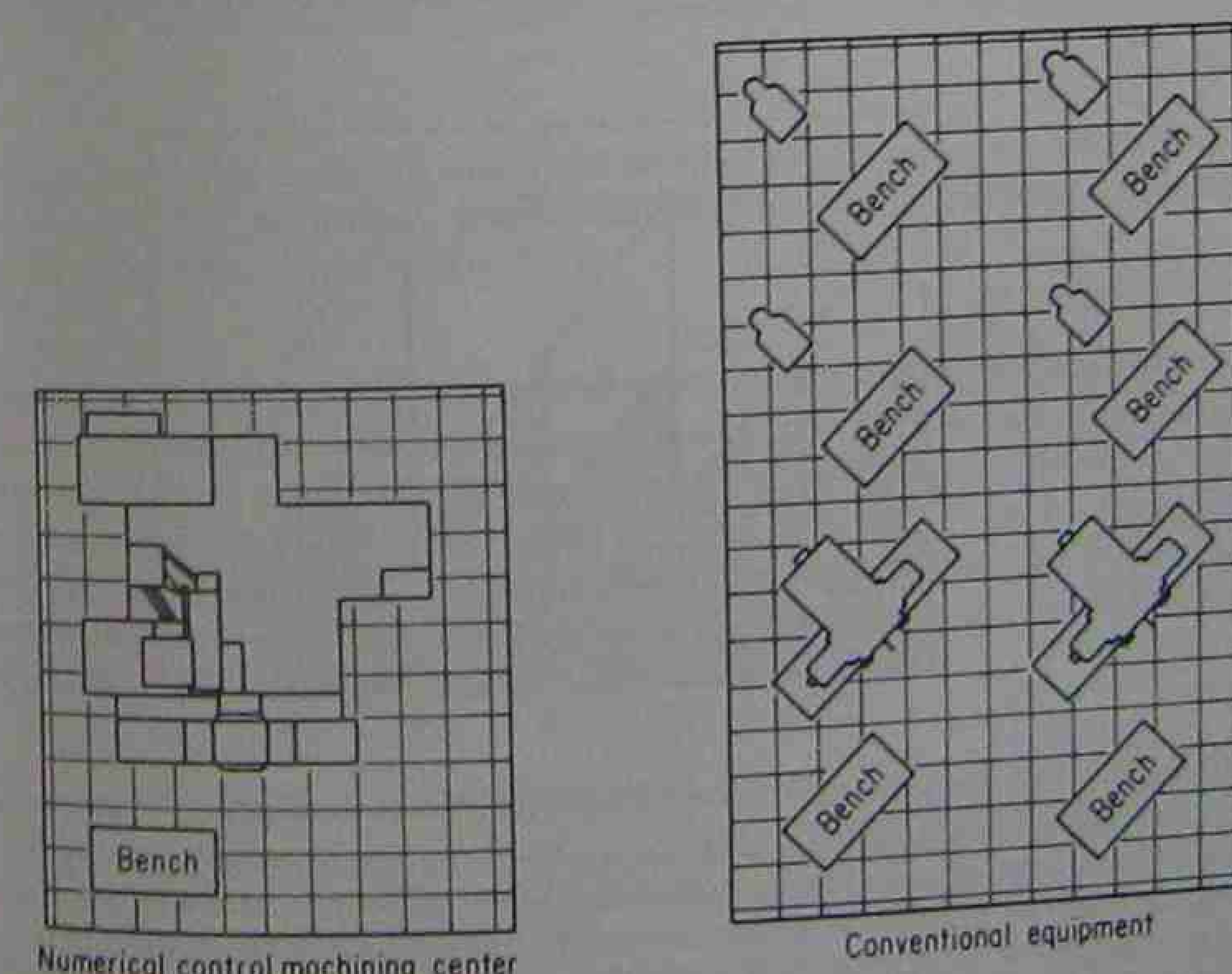


Fig. 6–30. A comparison of minimum floor area requirements of conventional and numerically controlled equipment for producing equivalent outputs usually results in favor of the numerically controlled equipment by a factor of at least two to one.

doubted, however, whether this contribution toward reduced overall manufacturing costs has yet to be fully appreciated or taken advantage of to any great extent.

Reduced Flow Time and Inventory In conjunction with reduced tooling costs another recognizable benefit of numerical control is the reduction of the time required to progress from the blueprint information to the completion of the first piece. It is not unusual, particularly with complicated parts, to reduce several months of flow time to weeks or even days. The possibility that several persons can effectively work on or program portions of a part may contribute significantly toward shorter lead time since, with conventional machining, there is a limit to the number of people that can simultaneously contribute to the preparation of a piece of hard tooling.

Aside from some of the obvious benefits of shorter delivery schedules and, consequently, happier customers, shortened lead time results in a favorable sustaining effect on inventory requirements, since it may not be necessary to stock certain items if delivery can be made within reasonably short periods.

Another important factor contributing toward lower inventory levels is reduced setup costs, since usually the average stock level maintained will depend on a number of factors, one of which is the total preparation and setup costs for a run. Another factor would be the rate of usage or sales. If, for example, the setup cost for a particular part was high in relation to the actual run time,

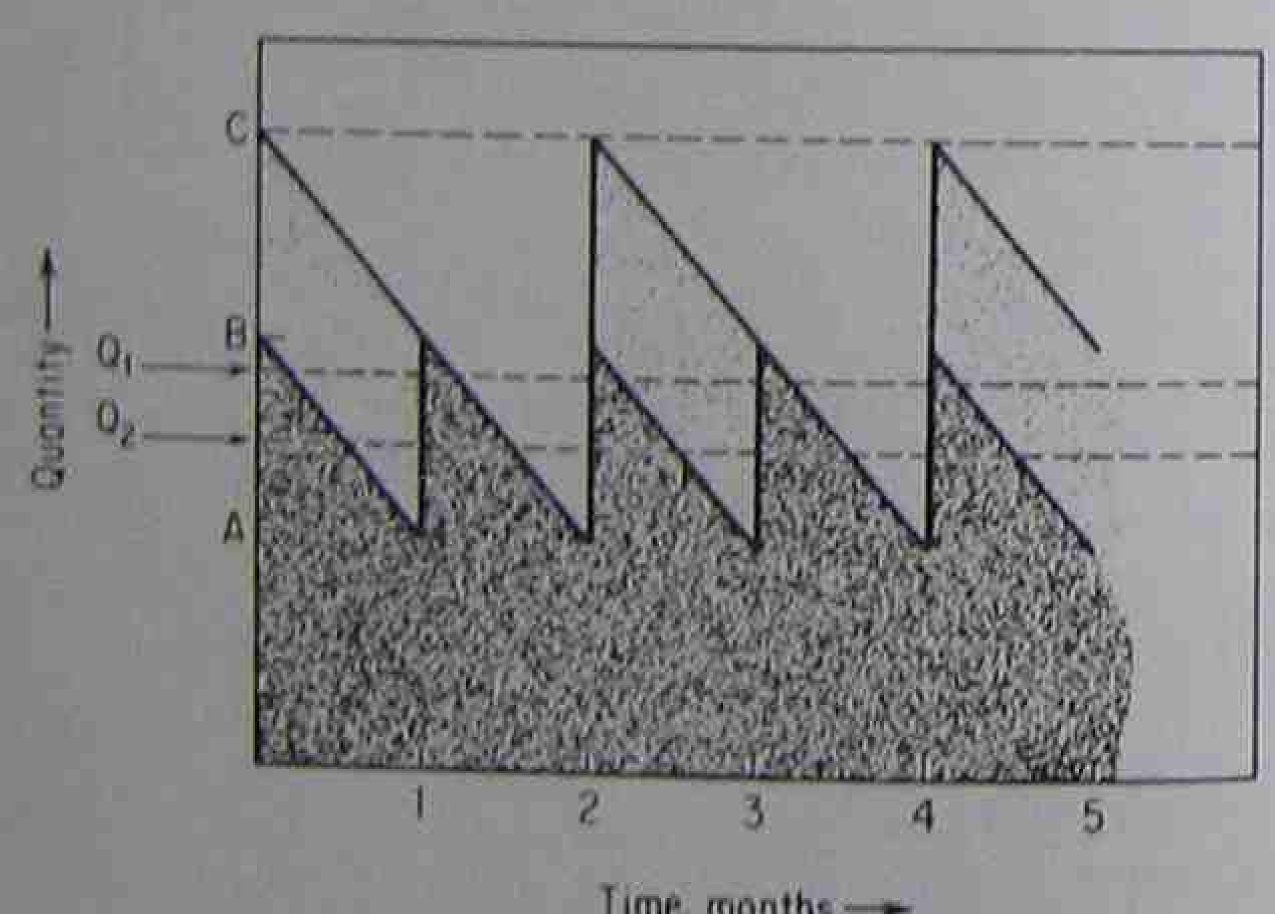
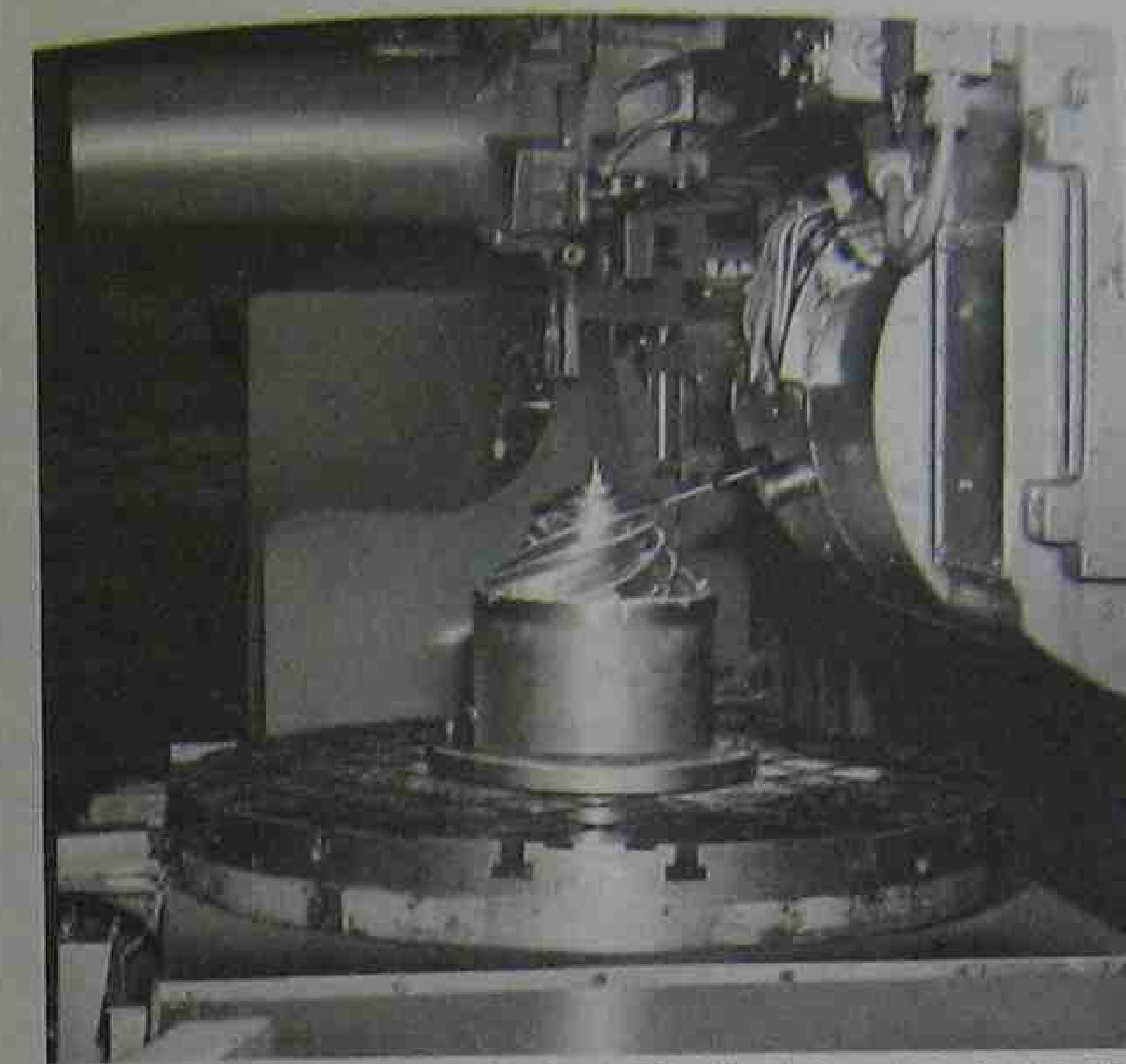


Fig. 6–31. If the minimum allowable inventory level is established at line *A* and a lot was produced every two months, assuming that the attrition rate of the finished parts was reasonably even, then the quantity represented by *A* to *C* would be required for each run. This establishes an average inventory level at Q_1. If, however, the manufacturing cost is lowered due to reduced setup time with numerical control, then it may prove just as economical to run a lot every month, thus reducing the average inventory value, and consequent investment costs, to the level Q_2.



Courtesy of Rocketdyne Div., North American Aviation Inc.

Fig. 6–32. From a practical standpoint, this rocket engine impeller shown on the work table would have been extremely difficult to machine prior to the advent of numerical control.

a large quantity of parts per run could most likely be expected. Since numerical control reduces setup time, smaller size lots may be considered. The effect on inventory is shown in Fig. 6–31.

Versatility in Design The latitude of controlled motions capable with numerically controlled machines has opened the door to part design concepts which were previously untenable with conventional equipment. For example, prior to numerical control it would have been impractical, if not impossible, to machine the part shown in Fig. 6–32 which was machined on the equipment shown in Fig. 6–14. Also, changes in engineering design are far more easily facilitated with numerical control.

Improved Production Planning and Control Since far greater control may be exercised over machining and other numerically controlled production operations than with conventional equipment, production planning and shop loading are also more predictable. In effect numerical control has removed many of the last-minute questions that arise on the shop floor and placed them back in the programming and other planning areas thus enabling the shop to become more efficient and thereby produce more work on a more reliable basis.

Less Skilled Operators Required This can be a moot point. If the NC machining involves prototype, very accurate, or complex parts, then a highly skilled machinist would be a sound investment. However, there is a good deal of machining that can be performed by nonmachinists or semiskilled operators utilizing numerical control machines, especially when machining larger quantities of identical parts. Indeed, the lack of skilled machinists is one of the major reasons for adopting numerical control.

QUESTIONS TO CHAPTER 6

1. Owing to the feedback arrangement inherent in most numerical control equipment, it is to be expected that the machine will cut metal a good deal faster than its conventional counterpart, thus resulting in the notable savings generally associated with numerical control. (True, False.)
2. In certain instances, conventional equipment may prove superior to numerical control equipment with regard to direct dollar savings. (True, False.)
3. Whether numerical control equipment would be more efficient with respect to its conventional counterpart will depend primarily upon (circle two):
   a. The size of the part
   b. The quantities being considered
   c. The skill of the operator
   d. Whether the day or night shift is being considered
   e. The complexity of the part
   f. Whether or not the area in which the numerical control equipment is operating is air conditioned.
4. While a reduction in floor-to-floor production time is significant with NC, by far the biggest contributing factor toward an NC justification lies in its reduction of floor-space requirements, especially in high rent districts. (True, False.)
5. The average percentage savings of NC vs. conventional milling is 66.4 percent (cf. Table 6–1). This means that a part that took 20 hours to machine conventionally would take approximately 11 hours, 13 hours, 7 hours, 3.5 hours to machine by NC. (Circle correct number of hours.)
6. Referring to Table 6–1, if six conventional lathes are required to handle the workload in a particular machine shop, how many NC lathes would be required to handle this same workload, assuming all of the work is suitable for NC machining.
7. Generally, the single, most significant reason for switching to numerical control has been:
   a. High accuracy
   b. Higher repeatability
   c. Better finishes
   d. Direct man-hour savings
   e. Lower inventories
   f. Reduced lead time
   g. Lower overall power-requirements.

8. Numerical control is not applicable in companies that manufacture in "mass production" quantities. (True, False.)
9. The degree of man-hour savings that may be attributed to numerical control equipment is generally proportional to the amount of human involvement in the operation. Separate each of the machines listed below into two categories: Those which appear to be good numerical control candidates, and those which appear to be poor numerical control candidates:
   a. Lathe
   b. Centerless grinder
   c. Drill
   d. Gear hobber
   e. Automatic screw machine
   f. Mill
   g. Tool grinder
   h. Surface grinder
   i. Inspection machine
   j. Punch press
   k. Drafting machine
10. Numerical control has often proved to be the best route even when machining one-of-a-kind. (True, False.)
11. The cost of preparing a tape usually ranges from: a. 10 to 20 percent; b. 5 to 6 percent; c. 3 to 8 percent; d. 10 to 70 percent; e. 5 to 98 percent of the cost of preparing comparable hard tooling. Which percentage is correct?

CHAPTER SEVEN

Justifying Numerical Control Equipment

Actually, there is little difference between the procedures for evaluating the procurement of numerically controlled equipment and any other capital expenditure. Considerations such as improved quality and reduced operating costs apply in numerical control justifications as in almost all capital-type procurements. If numerical control is to be considered somewhat unique, it is probably due to the fact that many of its advantages are not yet readily discernible. As more units are incorporated into different manufacturing environments, it is becoming apparent that there are numerous advantages connected with its operation which were originally unforeseen. This may be borne out by the fact that a second numerically controlled machine is generally much easier to justify than the first. The factors selected as examples in this chapter, however, represent direct economic savings that have been substantiated repeatedly in numerical control installations. For purposes of illustration, intangible benefits such as improved quality and decreased flow time, although significant, may be considered as bonus advantages and weighed in accordance with a particular operation.

SOME GENERAL CONSIDERATIONS

While there are many approaches to an economic analysis for evaluating a procurement, the following general considerations in the author's opinion are pertinent:

1. Any justification analysis is a prediction and is based on *anticipated* operations. In the great majority of cases it may be fruitless to attempt an elaborate and precise analysis. It is therefore suggested that the analysis be kept *simple* and deal with the major contributing factors. Normally, there is little basis for attempting to establish the precise dollar savings to be expected some years hence when such major factors as obsolescence, future work-loads, inflation, widely varying interest rates, and new product developments are extremely difficult to predict with any real accuracy (see Fig. 7–1). Also, a straightforward and readily understood justification analysis is generally more acceptable to management since, in the final examination, it is usually not the sole deciding factor but only contributes toward the decision. The decision itself must be based on such intangible factors as future competitive advantages, anticipated budget requirements, possible obsolescence of the equipment being considered due to other improved manufacturing techniques, and market trends.
2. When considering replacement of present equipment, rarely are sufficient funds available to "*take advantage*" of all the procurements that can be "*justified.*" Although savings may be predicted through the procurement of a certain piece of equipment, the dollar value of capital equipment purchases must necessarily be restricted in order to allot funds to other required operating expenditures such as taxes, fuel, rent, and salaries. Most often, therefore, the justification analysis must *clearly* demonstrate a reasonably large predicted savings. Should extensive calculations be required, it may indicate a marginal condition which cannot be tolerated in the light of overall budget requirements. Admittedly, there may be examples where a lengthy and detailed analysis will reveal a substantial subtle predicted savings; however, this is generally a rare case. Highly detailed analyses are probably more justifiable when comparing alternative equipment due to a required expansion or the establishment of a new manufacturing facility than when considering the replacement of present equipment.
3. Separate the tangible dollar savings expected from the intangible advantages such as improved quality, shorter flow periods, and improved customer relations, even though these intangibles may contribute heavily in the final decision. Attempting to place a price tag on nebulous, although perhaps significant factors, and including these in the basic economic justification may only tend to hide or distort the true economic picture.
4. Utilize as much existing *data* as possible. The difficulty in arranging a justification lies not so much in the formula but in acquiring the necessary back-up data. This may prove more difficult with smaller companies that have never established standard hour systems or time study analyses. Applicable and properly apportioned cost accounting data may also prove

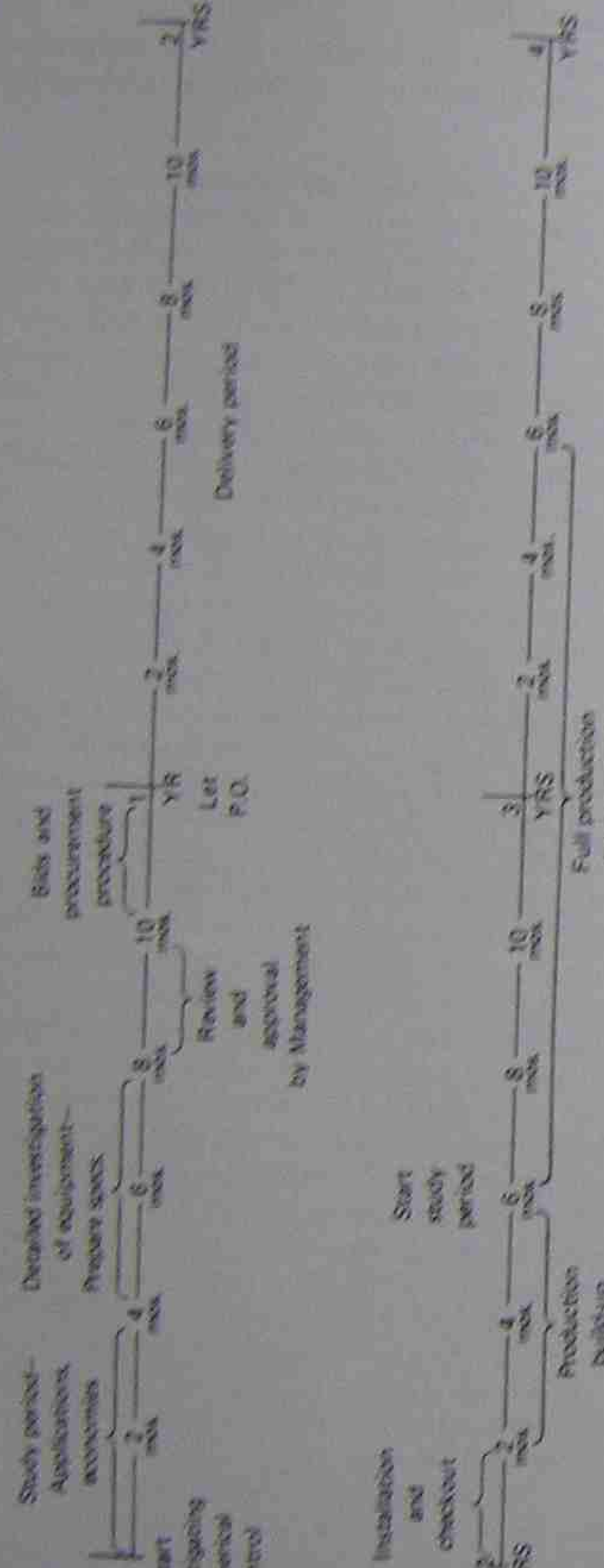


Fig. 7–1. A typical flow period describing the series of events leading from start of investigation to full operation is illustrated above. Delivery time will vary depending on the type, size, and general demand for numerical control equipment. It will be noted that the yearly justification period under study may be approximately three years away and a minuscule analysis would seem inappropriate in light of factors difficult to predict, such as: new product developments, future workloads, competition, and inflation. The "small shop" has an advantage in this instance, since investigation and procurement practices are generally less rigid, and formal procedures required in larger operations are obviated.

difficult to come by, especially when attempting to pro-rate or assign expenses to a particular operation. One difficulty may arise with overhead costs which are often apportioned across a relatively broad area of operation that includes many additional manufacturing operations other than the one under consideration. Time study figures, data from original proposal estimates, and shop loading records are possible sources of information.

5. While numerous texts and countless detailed formulas covering capital procurements have been proposed, it is generally agreed that no one formula is universally acceptable. Since no two companies have exactly the same operating characteristics, the exact justification formulas used will necessarily differ. The general approaches, however, need not be entirely different; and the extent of detail may depend on the amount of data available.

WHEN MAY NUMERICAL CONTROL BE A CANDIDATE?

Experience has shown numerical control is particularly well suited to a manufacturing operation when:

1. *The quantity of parts being produced in any single setup is relatively small.* The range may be from 1 to 200 pieces although theoretically there may be no restriction to the upper limit. Figure 7–2 illustrates the variations in the total cost of a hypothetical production lot as the total quantity or lot size varies when (A) numerically controlled equipment is used, (B) conventional equipment is used, and (C) special-pur-

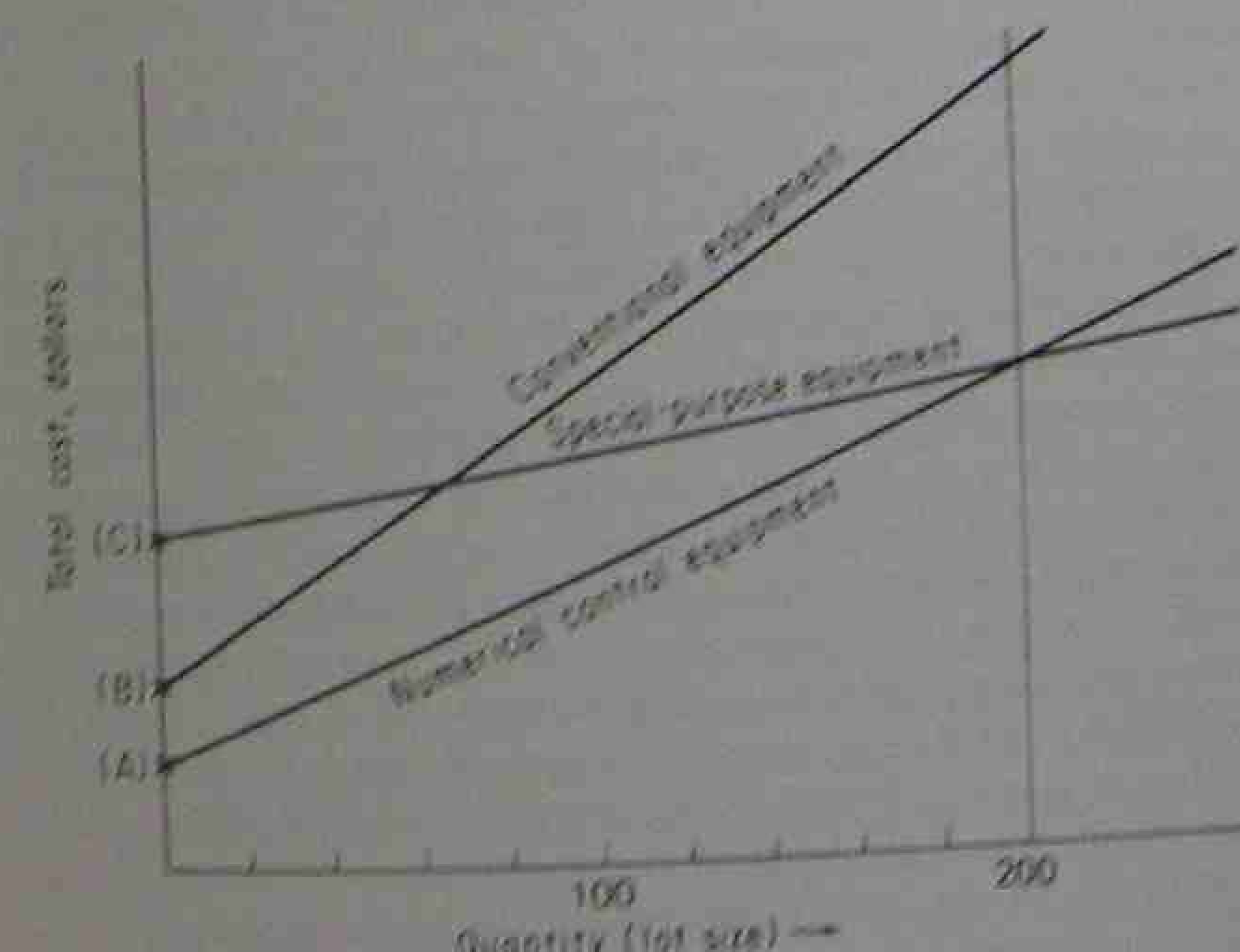


Fig. 7–2. The graph above illustrates setup and tooling costs for three different machining methods—Points (A), (B), and (C)—and their associated straight-line curves which represent the total cost for the particular method at varying quantities. The example above indicates that for quantities of from 1 to 200, numerical control is the most economical and that above 200 units special purpose equipment might be considered.

pose equipment is used. If, as an example, the costs to prepare for machining were considered alone, the totals would appear at the "B" quantity of production as noted for the three different means of manufacture.

a. The cost at Point (A) would include tape preparation and setup. If a holding fixture is required, its design and fabrication costs are included.
b. Point (B) which refers to conventional machining would include design and fabrication of tooling, if required, plus setup. In this instance, the tooling, or fixture, would probably be more complicated than with numerical control and would incorporate guides such as bushed drill holes or tracer templates. Manual preparation, such as scribing lines on stock material and machine adjustments, would also generally require more time than to prepare a tape.

When considering very low quantities, such as one or two units, it is possible that the conventional machining route could prove more economical than the numerical control route. This depends, almost entirely, on the complexity of the part. If the machining requirements are very simple, such as the drilling of a few holes in flat stock, then conventional machining may prove more economical than the numerical control route for the few pieces. If, on the other hand, a part is complex, either due to geometry or the number of tools required, then even one piece can normally be machined more economically via numerical control. The more complex the part, the more suitable it is for numerical control manufacture. A very simple part might therefore change the location of the conventional equipment line shown in Fig. 7–2, so that at very low quantities this route would be more economical than numerical control. Also, a good deal depends on how a company is organized to handle its numerical control operation. If the procedure is formal, as with most large companies, in which the planning for a numerical control part must progress through several departments (e.g. methods, tool design, programming, scheduling), then the more direct route would be to hand the engineering drawing to a machinist and have him prepare the pieces by conventional machining practice, keeping in mind that this would hold only for very simple parts, in very small quantities. It appears that, even though numerical control machines are replacing conventional equipment at a rapid rate, there will always be a requirement for conventional machine tools, even if only to prepare parts for proper positioning and alignment for machining on numerical control machines.

c. Point (C) would include the design and fabrication costs for elaborate tooling customary with mass production equipment such as a transfer line. Manual setup and adjustments would also be extensive with this type of arrangement.

If it is found that the total cost to machine a part conventionally after tooling and setup charges have been included is higher than with nu-

merically controlled machining, then the numerical control and con-
ventional lines will never meet and one piece would be cheaper to
produce by numerical control. Because of lower machining cost per
piece, with special-purpose equipment, a lot quantity is reached
whereby numerical control becomes more expensive. Numerically con-
trolled machine tools, because of their versatility, cannot usually com-
pete costwise with special-purpose, mass-production equipment and
were never intended to. Their forte is with small and medium size runs.
However, it is interesting to note that, according to one survey, ap-
proximately 80 percent of all machine parts produced are handled in
lots of less than 20 pieces. A wide field is therefore open to numerical
control.

2. *Parts are complex.* Because of multi-axis capabilities, wherein numer-
ical control makes many simultaneous machining motions possible,
parts may now be produced that heretofore were impossible. The part
shown in Fig. 6–32 requires five simultaneous contouring motions (X,
Y, Z, and 2 twist) and would have been extremely difficult to produce
without numerical control.
3. *Lots of the same part are to be repeated.* This is somewhat of an ideal
situation since the program and tape have already been prepared.
4. *Design changes are frequent.* The relative ease with which changes can
be made on tape when compared to changes in hard tooling weighs
heavily in numerical control's favor.
5. *Minimum lead time is a requirement.* Generally, the length of flow time
required from blueprint to first finished part is considerably less with
numerical control manufacturing. Reductions of from days or weeks,
to several months, are not uncommon.
6. *Tooling costs are high with respect to the overall manufacturing cost.*
7. *Inspection costs are high due to a requirement for measuring many
check points.* Because of extremely close dimensional repeatability in-
herent with numerical control, it is often only necessary to inspect a
first piece and spot check critical dimensions on succeeding pieces.
8. *Decentralized plants and/or an extensive sub-contracting complex re-
quire an interchange of tooling or manufacturing data.* Because of pre-
determined instruction data and general standardization of tape size
and formats, exchange of work between plants can now be handled far
more smoothly than under conventional manufacturing means.
9. *Scrapping a part would be costly.* Due to advances in forging and casting
techniques, the value of parts, prior to machining, is often significant.
The part shown in Fig. 6–9 which is being machined under numerical
control starts as a raw forging valued at approximately $2,000. It is not
uncommon, particularly in the aerospace industry that, either because
of invested manhours or special material, completed pieces attain a
value of $50,000–$60,000. It is also quite possible that one operator
error could scrap such a part causing a serious delay in a priority pro-
gram, in addition to the substantial monetary loss.
10. *Floor space is at a premium.* Greater productivity of numerically con-

trolled equipment (two to six times) when compared to conventional counterparts results in fewer machines being required. However, associated components, such as the hydraulic reservoir, electrical equipment, and control cabinets, are required with a numerical control unit so that the net saving in floor space is generally from 30 to 60 percent.

METHODS OF JUSTIFYING NUMERICAL CONTROL EQUIPMENT

As noted earlier in this chapter the approach toward justifying numerically controlled equipment is not wholly unlike methods used in justifying any capital procurement. Essentially the advantages and economies to be gained must be weighed against the investment.

Consideration for the procurement of manufacturing equipment usually occurs when:

1. *A new plant is being established and equipment procurements are essential.* In this instance a decision as to alternatives is required. If the lot sizes and variety of products are conducive, then numerical control would most likely have a clear edge.
2. *Facilities are to be expanded to meet increasing production.* Assuming that numerical control fits into the production scheme, it again should have a distinct edge. In this instance, as with an entirely new facility, the unit costs to produce the parts contemplated by different methods and with numerous types of equipment must be compared.
3. *Replacement of worn out equipment is in order.* Oftentimes there is an admitted reluctance toward change, so that it may be difficult to consider a replacement in the light of the latest manufacturing technology. It is, perhaps, not an altogether unusual occurrence for a shop operator or supervisor to request a machine "just like the one I've been using for the past 15 years—*only new.*" The goal should be to produce acceptable parts, as economically as possible. The best manufacturing means for accomplishing this end may be altogether different from that presently being used.
4. *Modernization of existing facilities, although adequate, is sought.* In this instance, neither the age nor condition of the present equipment is of consequence except as it affects trade-in value, the object being to manufacture identical or similar parts more cheaply by replacing present equipment, considering the cost of the new equipment. This has been the most common circumstance when considering numerical control.

While a number of acceptable approaches or formulas may be used for assembling and presenting justification data there are two which probably have been used more extensively than others when considering modernization. These are:

1. Payback
2. MAPI

PAYBACK METHOD

This method, which is sometimes referred to as the *payout* or *payoff* approach, is the most direct, aside from a rational guess, and, while not entirely accurate, has undoubtedly been used more often than any other. Simply stated, the yearly savings anticipated by operating the proposed equipment are compared against the investment, and the number of years required to cover the investment is calculated. As an example, consider the procurement and installation of a numerically controlled drilling machine costing $30,000. Yearly savings by operating a numerically controlled machine are estimated to be $15,000. The payback period would be calculated by dividing the total cost ($30,000) by the yearly savings ($15,000): $30,000/$15,000 = 2 years. If we are to consider income taxes, which are applied against savings, the savings would be reduced. However, depreciation is a cost which would reduce the income taxes and, effectively, raise the savings. Another consideration is the investment tax credit, which would reduce the investment, thereby reducing the payback period. A comprehensive formula considering all these factors is shown below:

$$\text{Payback Period} = \frac{\text{Total Investment} - \text{Total Investment} \times \text{Percent Investment Tax Credit}}{\text{Savings} - \text{Savings} \times \text{Tax Rate} + \text{Yearly Depreciation} \times \text{Tax Rate}}$$

If in the example above the investment tax credit were 10% and the tax rate were 46%,[1] the payback period would be 2.85 years. (See below.) The yearly depreciation is based on a 10 year life, and, if figured on a straight line basis, the yearly depreciation would be $3,000 per year:

$$\text{Payback Period} = \frac{\$30{,}000 - \$3{,}000}{\$15{,}000 - \$15{,}000 \times 0.46 + \$3{,}000 \times 0.46} = 2.85 \text{ years}$$

ADVANTAGES OF THE PAYBACK METHOD

1. It is simple, readily understood, and appreciated.
2. It has real significance when considering equipment that is highly susceptible to obsolescence, either by its own nature or due to the nature of the product.

DISADVANTAGES OF THE PAYBACK METHOD

1. It does not offer a ready indication of the return on investment, although this may be determined by further calculations which amount to the reciprocal of the payback period apportioned yearly depreciation of the equipment. If, in the example shown above, a 10-year service life were considered, the yearly depreciation cost would be $30,000/10 or $3,000 per year, calculated on a straightline basis. The resulting net savings would be $15,000 − $15,000 × 0.46 + $3,000 × 0.46 = $9,480 operating savings as noted above, less $3,000 which is the yearly depreciation expense, or $4,500 per year. The $9,480 figure may then be compared with the average

[1] Varies depending on the company's profits. Shown is the maximum amount. Latest IRS tax schedule is 16% on first $25,000 profit; 19% on the next $25,000; 30% on the next $25,000; 40% on the next $25,000; and 46% on profits over $100,000. These figures may vary from year to year as does the investment tax credit, and it is suggested that the latest tax schedule be consulted.

yearly investment of $27,000 ($30,000 − $3,000) which results in $9,480/ $27,000 or a return of 35.1 percent on the investment over a 10-year period.

2. It is difficult to compare alternatives with respect to the best investment return unless depreciation and a percentage return on investment are taken into consideration as noted above.
3. It offers no "*satisfaction*" during the payback years and therefore cannot be considered as an operating advantage, from an accounting viewpoint, until after the payback period.

MAPI METHOD

Probably the most popular method with numerical control justifications is the MAPI approach. This technique, developed by the Machinery and Allied Products Inst., 1200 18 St. N.W., Washington, D.C., fits well with a modernization analysis which is the most prevalent type when considering the procurement of numerically controlled equipment. Essentially the MAPI formula compares a first year's cost to produce parts by the existing method with that of a proposed method. The savings which would be realized with the proposed method are then compared against the first year apportioned cost of the new equipment. If a significant savings can be shown during the first year of operation, it can be reasonably assumed that in the following years justifiable savings will also be realized. The approach is similar to the payback method in that the yearly savings are calculated in the same manner. The distinction is that MAPI weighs the *first* year's savings against the apportioned first year's cost, utilizing the double declining balance method.[2]

ADVANTAGES OF THE MAPI METHOD

1. As with the payback method, the MAPI formula is relatively simple and easily understood and appreciated.
2. It is excellent when considering the replacement of present equipment.
3. It denotes when new equipment should be purchased since the factors evaluated in the analysis vary with time and a negative result at one period may become positive at a later time.

DISADVANTAGES OF THE MAPI METHOD

1. It does not denote the rate of return on investment although this can be calculated from the source information.
2. It emphasizes machine operating costs while many of the advantages of numerical control lie outside the machine tool area such as in: (a) Production Control, (b) Inventory Control, (c) Engineering, and (d) Quality Control. These savings, when they can be calculated, should be noted on line G of the form shown in Fig. 7–9 as described later in the chapter.

[2] Instead of deducting an equal amount of depreciation each year, as with the straight-line method, the percentage is doubled and then multiplied by the remaining value of the investment. As an example, consider an investment of $50,000 depreciated over a ten-year period. On a straight-line basis the depreciated value would be $50,000/10 or $5,000 per year, a depreciation of 10 percent per year. On a double declining balance method the first year's depreciation would be 10 percent × 2 or 20 percent of $50,000 = $10,000; the second year's depreciation would be 20 percent of $40,000 = $8,000; the third year's would be 20 percent of $32,000, etc., until the tenth year is reached.

3. It does not look beyond the first year although it is assumed that succeeding years will follow suit and savings will be realized if the first year indicates such is the case. This is not an entirely unreasonable assumption since accurate prognostications beyond the first year incorporate a good deal of speculation with any type of justification analysis.

EXAMPLE OF A MAPI JUSTIFICATION

Since an evaluation of intangibles is generally necessary when considering procurements and no two operations are wholly similar, it may be concluded that the answer to an economic formula should not be the sole deciding factor when considering a procurement nor should any one formula apply universally. An economic analysis based on a formula does, however, offer sound direction on which to arrive at a decision.

The example shown below, therefore, describes a general step-by-step analysis which may be followed in determining the economics of a possible procurement. Steps may be deleted, added to, or their order changed—depending on the particular application of the study.

Step 1 Select a representative sample of manufactured parts. Care should be exercised in order that the parts selected represent, as close as possible, the type of production *anticipated* during the year under study, which is the first year after the new equipment would be installed, should it be justified. While there is no rule as to the number of parts, experience has indicated that a carefully selected complement of twenty generally offers a fair cross section. Actual time studies, standards, estimates used when quoting, or general shop experience may next be used to determine the proportion of time which would be required by the different types of operations. A suggested work sheet format is shown in Fig. 7–3.

There are, of course, simpler though perhaps less accurate means of determining the proportion of work. One means would be to count the number of different types of machines operating at any one time. Another would be to collect the number of manhours devoted to particular types of machines over a period of time. This latter means is probably satisfactory assuming the work reviewed in the shop is representative of future type work. An advantage of the approach shown in Fig. 7–3, however, is that the data gathered in this study will be usable with the justification analysis itself, which is to follow.

Step 2 Although drilling operations weigh heaviest in the example shown in Fig. 7–3, which will more often than not be the actual case, applying numerical control to this area may not be the most fruitful choice, since the greatest savings realizable with numerical control may be in other manufacturing areas. If milling operations are extremely complex and require a large proportion of setup hours, an investment in numerically controlled mills might offer a better return than with numerically controlled drilling equipment, even though the

HOURS FOR AVERAGE LOT SIZE OF 15 PIECES INCLUDING SET UP TIME

Line No.	Part No.	Part Size LxWxH or LxDia	Number of Different Tools Required	Drill	Bore	Mill	Turn	Total
1	45C103	4X2X1	2	.20	—	.23	—	.43
2	63B208	6X2X2	3	.08	.13	.09	—	.30
3	48M2	7X3X1	2	.26	—	.02	—	.28
4	63P206	3X1X1	8	.13	—	—	—	.13
5	M-61	5X2	3	.06	—	—	.32	.38
6	0-59-X	20X4X3	2	.33	.09	.39	—	.81
7	45T100	7X6	4	.07	.05	.08	.12	.32
8	XXXX	XX	XX	XX	—	XX	—	XX
18								
19								
20								
Total Hours				3.02	.63	1.61	1.02	6.28
Percent of Overall Machining Requirement				48	10	26	16	100.0

Fig. 7–3. Tabulation of hours required to perform various operations on a *representative* sample of twenty different parts. According to the sample, drilling operations account for almost half of the shop requirement. Also shown is the approximate size of the parts which is used to determine the size of the machine and is described in Step 3.

percentage of drilling operations is significantly higher. If, however, the complexity of operations is reasonably uniform, the area of first approach might logically be the area of greatest activity. Since, in the example shown, drilling operations account for approximately one-half of the shop's activity, these will be selected as the first to be scrutinized.

Actually there are few numerical control machine tools that perform drilling operations solely. Most are capable of performing several operations such as drilling, milling, and boring. These are called *machining centers*, and this concept extends to the lower cost machines as well as to the larger tool changing and rotary table type machines. There are differences, however, in the structure of machining centers which would make one more suitable for, say, milling than another. The spindle, for example, would be more sturdy, and supplied with a heavier horsepower than normally required for drilling. Utilizing the chart format shown in Fig. 7–3 would point out the higher orders of activity.

Step 3 Next, the size of the machine should be determined. Generally, a machine tool buyer will purchase more capacity than is really required. And this is very expensive. The problem is that the normal buyer will recall those parts which were extraordinarily large and procure a machine to handle these few parts. Since the cost of a machine rises on almost a geometric scale with its size as, for example, a machine having a 60″ X-axis travel costing four times as much as a machine having a 30″ X-axis travel, it would seem prudent to investigate the most economic size required very carefully. One way of doing this is to plot the sizes of sample parts on a *probability-log* scale. Figure 7–4 shows a straight line that has been plotted on *probability-log* graph paper.[3] The straight line has been drawn through points that were plotted for the lengths of the sample parts described in Fig. 7–3. It is presumed that the length of the parts would lie along the X axis of the machine tool, which is generally the longest axis. The horizontal axis of the graph represents the percentage of parts that fall within a particular size category. Referring to Fig. 7–4, 95 percent of the parts are 30″ or less in length. We would have to add a few inches on both sides of the part for cutter allowance, say two inches on each side, which would mean that a 34″ X-axis travel would handle 95 percent of the parts having an X-axis dimension of 30″ or less. If we wanted to accommodate 99.9 percent of the parts, which is an impractical requirement, we would need a machine having an X-axis travel of 94″ (90″ + 2″ + 2″). A reasonable percentage of coverage is 85 percent which would require a machine having approximately 24″ of X-axis travel (20″ + 2″ + 2″). A similar graph could also be drawn for the Y and Z axes, although the X axis is usually the predominant consideration. The same type of analysis would apply to lathes, only, in this case, the considerations would be the diameter and lengths of the parts.

In plotting the points with respect to the horizontal axis it is required that the midpoints of the percentages be used. In the example shown there are twenty parts and, therefore, each point would be plotted at 5 percent increments (100% ÷ 20 = 5%). However, instead of starting at 5 percent and then plotting

[3] May be procured at most drafting supply stores.

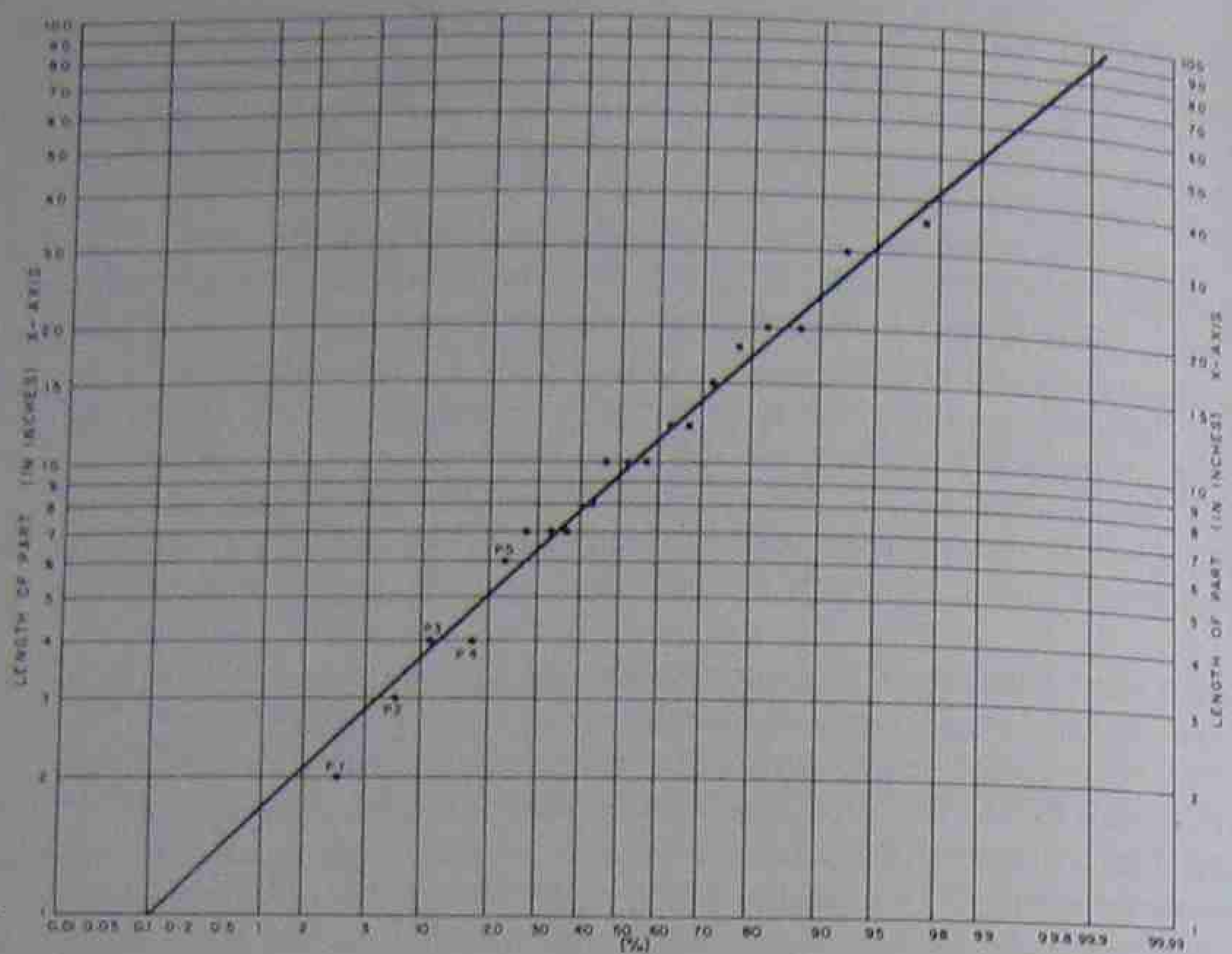


Fig. 7–4. The required size of machine tool can be determined from the chart shown above. If, for example, it were required to determine the X-axis travel that would accommodate 95 percent of our parts, we would read up from the 95 percent figure on the abscissa (horizontal axis) until we "hit" the plotted line, and then read horizontally to the ordinate (vertical axis). In this case the reading would be 30". If we then add 2 inches on each side of the part, the X-axis travel would be 34 inches and would accommodate 95 percent of our parts.

the next point at 10 percent, the next at 15 percent, etc.; the first point is plotted at 2½ percent; the next point at 7½ percent; the next point at 12½ percent; the next at 17½, etc. The lengths of the parts would also have to be arranged in ascending order with respect to the particular axis being considered. Referring to Fig. 7–4, P1 would be plotted at 2½ percent and 2"; P2 at 7½ percent and 3"; P3 at 12½ percent and 4"; P4 at 17½ percent and 4"; P5 at 22½ percent and 6"; etc.

Another determination to be made under this third step is whether the numerical control machine should have an automatic tool changer. This depends on two factors: the number of different tools required for a part and the number of pieces in the lot. If the quantities to be machined in a lot are quite large (100–200 pieces) and even if there are few tool changes to be made per part, then an automatic tool changer should be considered. Also, if there are relatively few pieces in a lot but there are a goodly number of different tools required for the average part (e.g., greater than 10), then an automatic tool changer should again be considered. The number of tools per part are noted in Fig. 7–3.

Step 4 Review the types and specific models of numerically controlled drilling machines available. These range from relatively inexpensive single-spindle drills costing approximately $20,000 to complete machining centers costing several hundred thousand dollars. While high cost equipment may be justifiable, the dollar category is generally governed by the amount of funding available. Therefore, having selected the type of machine that fits into the proper dollar category, the next step is to evaluate the features of the various qualifying machines and their associated control systems. A suggested analysis form is shown in Fig. 7–5. The features shown represent a small portion of those that should be considered. It is suggested that Chapter 10 be used as a reference for control system features.

Step 5 When the analysis has narrowed to a type of machine—or better—a particular machine, the next and most difficult task is to estimate the required time for machining the sample parts, utilizing the equipment selected. Undoubtedly the most accurate, though least economical, means would be to subcontract all or a number of the sample parts to a shop utilizing the numerically controlled equipment under consideration. Quite often the machine tool supplier will offer this service at a nominal cost. More often, however, satisfactory results have been obtained by comparing in-house sample parts with similar parts which have been machined on numerically controlled equipment (see Chapter 6). As an example, run-time comparisons may be made based on the number of holes and different tools required for the in-house sample and for a similar part that has been machined on comparable numerical control equipment.

Although it would be natural to suspect that machine tool and control system manufacturers might exhibit *exceptionally* favorable cases, this has not been the rule. While there are cases in which savings as high as 95 percent have been experienced, normal expectant savings of from 40 to 75 percent have been *consistently* documented by both users as well as suppliers and are considered reliable. Generally, machine tool and systems builders have been careful not to describe exceptional cases, since it is realized that the best sales future lies in satisfied customers and repeat orders. It is also illogical to claim exceptionally high savings when, in the majority of cases, average savings are generally sufficient to justify numerical control. Figure 7–6 describes a suggested simple format for comparing conventional and numerical control direct machine hour costs covering the twenty parts originally selected.

A guide that might be used for determining percentage savings is detailed in Fig. 7–7. What is shown are the productivity increases reported by users of numerical control equipment via a survey made by the Numerical Control Society. Although the survey was made some time ago (approximately 1970) the figures have proven to remain reasonably accurate. If anything, they might be considered on the conservative side, in light of present-day CNC control systems. It would not be unreasonable to add, as a very conservative minimum, a 10% factor to these figures to account for improved CNC capabilities at the machine tool site. As an example, the average productivity increase for all manufacturers for drilling machines is shown as 113 percent in Fig. 7–7. This

COMPARISON ANALYSIS FORM

Feature (machine)	Model X	Model XX	Model XXX
1 Spindle Speeds	100 – 2500 rpm Infinitely variable	80–2000 rpm 20 steps	150–4000 rpm 40 steps 3 ranges
2 Feed (Depth) Z axis	0–110 ipm	50–80 ipm	2–90 ipm
3 Range X axis / Y axis	22 in. / 18 in.	20 in. / 16 in.	24 in. / 20 in.
4 Size of Table	23 x 19 in.	19 x 15 in.	23 x 19 in.
5 Horsepower of spindle	2	3	4
6 Rapid traverse rate	180 ipm	190 ipm	200 ipm
Feature (Control System)	**Model X**	**Model XX**	**Model XXX**
Design	Microcomputer	Microcomputer	Minicomputer
Tape Reader	Standard	Optional	Standard
Tape Punch	Optional	Optional	Optional
Maintenance Diagnostics	Optional	Standard	Standard
Cutter Compensation	Optional	Standard	Standard
Approximate Price	$xx.xx	$xx.xx	$xx.xx

Fig. 7–5. The format above may be used to evaluate the comparative features of machine tools and their associated control systems. The specifics noted are hypothetical and not intended to represent any particular model machine or control system.

CONVENTIONAL VS. NUMERICALLY-CONTROLLED DRILL HOURS FOR AVERAGE LOT SIZE OF 15 PIECES

	Part No.	Conventional	Numerical Control
1	45C 103	.05	.03
2	48M-2	.26	.13
3	63P 206	.11	.04
4	M-61	.06	.03
5	0-59-X	.19	.06
6	XXX		
7	XXX		
8	XXX		
9	XXX		
10			
11			
12			
13			
14			
15			
	TOTAL HOURS	2.01	.88

Percent savings of numerical control compared to conventional = 56 %

or 2.28 conventional machines would be required to do the work of 1 numerically-controlled machine.

Fig. 7–6. A direct comparison of the machine time required for both conventional and numerically controlled drilling operations is shown above. The numerical control time may be determined by actually running the parts on numerical control equipment like or similar to that considered or by comparing in-house parts to examples of similar parts that have been run on the numerically controlled equipment. This latter approximation has been found to be quite reliable and machine time savings, including set-up time, of from 40 to 75 percent are not at all uncommon.

Manufacturer of	Average productivity increase from NC machines (%)					
	Drilling	Boring	Turning	Three-Axis Milling	Five-Axis Milling	Punching
Aircraft and parts	149	125	193	287	293	300
Missiles and ordnance	115	106	93	258	420	350
Average for all manufacturers surveyed	113	84	107	197	267	185

Source: Numerical Control Society

Fig. 7–7. The figures shown are the percentage productivity increases when comparing numerical control equipment to conventional counterparts. A percentage increase of 149 percent would amount to a productivity increase ratio of 2.49.

means that a numerical control machine could produce 2.13 times (100 percent + 113 percent) the amount of identical parts as its conventional counterpart in a specified time. Looking from a different perspective, the numerical control machine could produce the parts in 47 percent of the time it takes the conventional machine (1/2.13 = 47 percent). This also means a savings of 53 percent (100 percent − 47 percent). If we up-date the figure to account for CNC by considering a 10 percent increase, the percentage savings would be 58.3 percent (53 percent + 5.3 percent).

Step 6 Next the direct machining dollar costs for the yearly period must be compared. This may be done as follows:

a. Determine the yearly load that could be handled by the numerically controlled machine. Assuming there are at least three conventional machines producing parts, on a full-time basis, similar to the twenty selected, it would be safe to surmise in this instance that one numerically controlled machine could be kept busy.

b. The number of operator hours required per year for the numerically controlled machine is calculated next. Assuming a two-shift operation the figures are as follows:

  240 working days per year × 16 hours per day = total number of paid operator hours required per year = 3,840 hours.

c. Calculate the total direct dollar man-hour cost to operate the numerically controlled machine for the yearly period, including fringe benefits such as social security and hospitalization insurance. This could amount to:

\$8.00/hr. (direct hour wage)	+	\$2.50 (direct hourly fringe benefit)	=	\$10.50/hr.

The total direct yearly cost to operate the numerically controlled drill (assuming one full-time operator is required for each shift) would be:

3,840 hours × \$10.50/hr. = \$40,320[4]

d. Next, the direct cost of equivalent production on existing machines is calculated. Assuming the same wage applies as with numerical control and one operator per machine is required, the cost would be 2.28 times as much, or \$91,930. This is explained as follows:

Based on the 20-part sample, 2.01 hours would be required by conventional equipment as compared to .88 hours by the numerically controlled drill. Or 2.01/.88 = 2.28 conventional machines required to do the work of the one numerically controlled machine. Therefore:

2.28 (efficiency factor with numerical control)	×	\$10.50/hr. (hourly cost including fringe benefits)	×	3,840 hrs. (total hours two shifts)	=	\$91,930

Step 7 If possible, determine the effect of overhead costs. This may be extremely difficult. Thus, in the example shown it may be impossible to say what portion of the overhead expenses such as heat, light, taxes and insurance would be saved when reducing two machines to one. Probably none. How great a reduction is necessary, therefore, before overhead expenses *are* actually affected? This will depend entirely on the type and size of operation. A general rule might be to ignore overhead costs when considering a change that will effect a small percentage of the shop (e.g., less than 5 percent). A larger change should be carefully considered on the basis of the *actual* savings—will less electricity and heat actually be required, for example, by moving from two- to one-shift operation?

Step 8 After the direct machine-hour savings, probably the next most significant tangible economy will be realized in tooling. If during the year under study 20 new and different parts are to be manufactured, the cost to design and fabricate the hard tooling for these parts must be estimated and compared to the cost for tape preparation plus the design and fabrication of simple holding fixtures, where required. In the example it is assumed that the majority of representative parts can be hand-programmed economically, however it might be expected that several of the more complicated parts would be programmed with the assist of sub-contracted or on-site computer time.

---

[4] While it is reasonable to expect some shut-down time on both numerically controlled and conventional equipment due to preventive and normal maintenance and less than an 8-hour shift might be considered, it is most often difficult to achieve full operator efficiency on a job to which an operator is temporarily transferred. The 8-hour shift period is therefore applied to both conventional and numerically controlled machines over the full year in the example described.

ANALYSIS OF CONVENTIONAL VS. NUMERICAL CONTROL TOOLING COSTS

	CONVENTIONAL			NUMERICAL CONTROL					
	COST, DOLLARS			COST, DOLLARS					
Part No.	Fixture Design	Fabricate Fixture	Total	Part Programing	Punch Tape	Computer	Holding Fixture Design	Holding Fixture Fabrication	Total
45C103	70	120	190	20	2	—	NOT REQ.	—	22
63P206	200	300	500	100	10	—	50	75	235
P-36X	620	900	1520	140	—	200	160	150	650
10-493	90	150	240	70	8	—	NOT REQ.	—	78
Total			$2450						$985

Fig. 7–8. Conventional versus numerical control tooling costs are compared above. Since tool fabrication costs involve material as well as labor, only the final dollar figures are shown (i.e., labor hours have been extended by multiplying by the hourly rates).

The cost of the present hard tooling for the twenty parts considered to be representative of *anticipated* work will most likely be on record and it now remains to estimate the cost of preparing tapes and holding fixtures. The latter should offer little difficulty although estimating programming and tape preparation time may prove to be a novel experience.

Several of the twenty parts may next be selected and described in a format such as is shown in Fig. 7–8. Although in this instance only four parts from the original twenty have been described, the number chosen for the tooling analysis may be more, depending on the range and variance in complexity of the parts originally selected. The dollar figures shown may be estimates derived from user and supplier literature or by actually programming the selected parts. One user, after approximately four years of numerical control practice, has determined that the time required to program a point-to-point part manually is closely proportional to the number of holes, and averages approximately *five minutes per hole* when the part is to be machined on a turret-type numerically controlled drill press.

If it is decided to program the sample parts and an inexperienced person performs the calculations, the time recorded should be multiplied by a decimal factor in order to arrive at a realistic figure, since in actual practice experienced programmers will perform this function. This factor may be as low as .35.[5] In other words an experienced programmer could program a part in about one-third the time required of a novice. Experience has also shown that the cost to prepare the "tooling" for numerical control (i.e., tapes and holding fixtures, if required) has *averaged* approximately 50 percent or less of the cost to design and fabricate equivalent hard tooling. This figure may therefore be used as a safe "rule of thumb." (See also Chapter 6.)

According to the analysis of Fig. 7–8, tooling costs for the four numerical control parts under consideration average 40 percent of that of conventional tooling. If tooling for 20[6] different new parts is considered, the estimated cost would be five times that shown for the four parts.

Numerical control "tooling" costs would be:	5 × $ 985 = $ 4,925
Conventional tooling costs would be:	5 × $2,450 = $12,250

Step 9 Consider other direct *dollar savings* resulting from:

a. Reduced inspection costs
b. Scrap reduction or elimination
c. Less power consumption
d. Reduced finished goods inventory

---

[5] This factor is derived from the *Learning Curve* theory which states that the time required to do a job decreases on a fixed-formula basis with relation to the number of times the job is performed.

[6] Although it is very possible that more than the 20 different parts used in the sample will be machined during the anticipated yearly period and tooling costs might consequently be higher, this figure was chosen in order to be consistent with the sample lot size.

e. Savings in assembly time due to closer accuracy and repeatability of detail parts
f. Less tool storage costs

(*Please refer to Chapter 6 for a detailed analysis of the various areas of savings.*)

Step 10 Assemble data into a final format which will present a comprehensive and economic picture. Figure 7–9 is essentially the same format as used by

Line	Type of Cost	Column 1 Cost of one year's operation with present conventional equipment	Column 2 Cost of first year's operation with one numerically controlled machine
A	Direct manhour costs (8 hours per day x 2 shifts per day x 240 working days per year x 2.28 conventional machines – col. 1 or 1 numerically-controlled machine – col. 2) x the direct hourly wage rate – including fringe benefits.	$91,930	$40,320
B	Tooling costs Hard fixtures for conventional vs. tapes & simple holding fixtures for n/c.	$12,250	$ 4,925
C	Maintenance	$ 500	$ 800
D	Scrap and rework	$ 4,000	$ 400
E	Inspection	$ 6,000	$ 800
F	Tool storage and preservation	$ 800	$ 80
G	Other savings	–	–
	Total operating cost for year	$115,480	$47,325

First year operating savings with numerical control = $115,480 – $47,325
Before taxes = $68,155
After taxes = $36,804
(54 % of $68,155)

Fig. 7–9. The simple format shown above has proven to be useful for collecting data on the year's operation being considered. In the example shown the adoption of numerical control would result in a sizable savings—even after consideration of income taxes, which reduce the savings by 46%.

government agencies in evaluating equipment procurements under "Modernization Programs," and has proven helpful for numerical control analyses. It is based on the MAPI concept.

First the direct man-hour costs as calculated in Step 6 are inserted on Line A of Fig. 7–9. Estimated tooling costs as described in Step 8 for both conventional and numerically controlled operations are next shown on Line B. Dollar estimates for other tangible costs should also be shown. Typical figures are displayed in the format.[7] Since savings, in effect, are earnings and subject to a 46 percent IRS tax (see footnote on p. 183), the $68,155 operating savings which resulted from a subtraction of the year's numerical control cost of $47,325 from the yearly conventional cost of $115,480, are reduced to $36,804.[8]

Step 11 Savings are next compared against the cost of the investment. The total investments is first computed:

1. Cost of new numerically controlled machine	$63,000
2. Installation	1,000
3. Total installed cost	$64,000
4. Disposable value of present equipment (2 machines—fully depreciated)	−4,000
5. Total investment	$60,000

The first year's apportioned cost, which is to be compared with the first-year's savings, may be calculated by multiplying the total dollar investment by the appropriate percentage factor shown in Fig. 7–10.

According to present IRS rulings, manufacturing equipment may be depreciated over a 10-year life, thus the chart in Fig. 7–10 calls for multiplying the total investment of $60,000 by 20.0 percent to obtain a first year depreciation investment cost of $12,000. As will be noted, this method of depreciation is more stringent than the straight-line method. It should be pointed out that methods of depreciation and the determination of service life on equipment are not held as fast rules by the IRS, and latitude is allowable providing the proposed depreciation percentages and service-life estimates can be substan-

[7] Although Line G of the format (Other Savings) may prove significant and should be carefully scrutinized, no figures are shown in the example because of the wide-range nature of these types of savings.

[8] As noted on p. 183, the 46% tax applies to the savings, or profits that would be realized by the numerical control equipment, thus reducing the effective savings. However, depreciation is a cost which reduces the dollar amount of taxes and thereby helps to improve the savings. There is also an investment tax credit that may be considered, and this reduces the overall investment. In order to demonstrate the principles of the analysis, these considerations have not been included in the example in the text (except for noting the after tax savings). If they were considered, the revised savings would equal savings − savings × tax rate + depreciation × tax rate. The revised total investment would equal total investment − total investment × percent investment tax credit. Using the figures in the example, the revised first year's savings would equal $68,155 − $68,155 × 0.46 + 0.2 × $60,000 × 0.46 = $42,324. The revised total investment would be $60,000 − $60,000 × 0.10 = $54,000. (Depreciation is figured prior to the application of the investment tax credit.)

Service Life (Years)	% Factor
5	40.0
6	33.3
7	28.6
8	25.0
9	22.2
10	20.0
11	18.2
12	16.7
13	15.4
14	14.3
15	13.3

Fig. 7–10. The *percentage factor*, when multiplied by the total dollar investment, results in a first year apportioned cost which is to be compared to the first year's savings. First year's operating savings less the total investment × percentage factor, equals the actual first year's savings. In accordance with general IRS regulations, a 10-year life has been selected which corresponds to a percentage factor of 20.0. The percentage factors are based on a standard accounting formula used for depreciation purposes. (Double declining balance method.)

tiated. Once a specific system is established, however, it generally must be adhered to.

RESULTS OF ANALYSIS

From the example described above it is estimated that one numerically controlled drill will perform the work of 2.28 standard drills, and it may therefore be assumed that two drills will be removed from the shop should the numerically controlled drill be procured—providing the work load remains the same.

Subtracting $47,325 from $115,480, a net operating savings of $68,155 for the first year is estimated should numerical control be put into operation, and is compared against the first year's investment cost of $12,000. Clearly, nu-

merical control holds the edge in this instance. Should production requirements remain unchanged, higher savings may be expected in succeeding years since the depreciation percentage factor will be reduced. This type of analysis, however, considers only the first year.

Should a comparison between the investment cost and savings result in a negative answer, wherein the cost is higher than the savings, another study may be undertaken at a later date, since factors such as maintenance costs, labor rates and production requirements change—in addition to improvements in numerical control equipment—which are expected to reduce costs as well as increase efficiency.

QUESTIONS TO CHAPTER 7

1. Because of the peculiarities associated with numerical control it is necessary that the procedures for evaluating its procurement be handled on a novel and entirely different basis than when considering conventional equipment. (True, False.)
2. Which one of the following standpoints makes justifying numerical control somewhat unique?
   a. The tape format is confusing
   b. More exact figures are required
   c. Depreciation life is generally longer
   d. Depreciation life is generally shorter
   e. Advantages are not always discernible
   f. The cost of hard tooling is difficult to figure
   g. Operators are not working a good portion of the time.
3. Intangibles may help a justification and, therefore, must be assigned a monetary value in order that they be afforded fair representation in the dollar justification. (True, False.)
4. In order to offer management a true picture of the economics involved in a numerical control justification, it is necessary that as much detail be included in the study as it is possible to accumulate. (True, False.)
5. Referring to the chart in Fig. 7–2, if, for purposes of demonstrating the chart, there were no setup or tooling costs involved in the production of a lot of 100 parts by a conventional method, and the cost to produce a unit was the same as that shown for the "conventional equipment," the approximate break-even point at which numerical control would be less costly would be:
   a. 10 pieces
   b. 50 pieces
   c. 100 pieces
   d. 500 pieces
   e. Never
6. In determining the proper size of NC machine it is always best to be on the safe side and to buy a machine that will be capable of handling all possibilities. (True, False.)
7. Referring to the graph shown in Fig. 7–4, what percentage of parts could be handled by a machine that had an X-axis travel of 24".

8. Referring, again, to the graph shown in Fig. 7–4, what is the largest size part that can be handled in order to accommodate 75 percent of the parts.
9. One of the most popular and readily understood methods of statistical justification is known as the:
   a. Crystal ball method.
   b. Lost cause method.
   c. Buy now–pay later method.
   d. Payback method.
   e. Cashflow method.
10. To date, most numerical control equipment has been justified on a basis of:
   a. Replacement for conventional equipment.
   b. Handling new and additional production requirements.
11. Two advantages of the Payback system are that it is ________________ ________________ and ________________.
12. The one most significant disadvantage of the Payback system is that it:
   a. Is too easy to calculate
   b. Does not offer a clear rate of return on investment
   c. Does not consider the future value of money
   d. Does not take into account the period *after* payback.
13. What is the percentage rate of return (return on investment) on a numerical control equipment investment of $40,000 if the depreciable life of the equipment is 10 years and the yearly estimated savings, not considering taxes, is $10,000?
14. Referring to Fig. 7–7, if 50 parts could be machined in 8 hours on a conventional lathe, how many parts could be machined on an NC lathe in the same time period? (Consider "Average for all manufacturers surveyed.")
15. After determining that there are sufficient funds for buying the numerical control equipment, which is the first step in a numerical control equipment justification?
   a. Checking sub-soil conditions for the foundation
   b. Reviewing parts and selecting a representative sample
   c. Checking to see that there is a sufficient supply of blank paper tape
   d. Reviewing the numerical control machines on the market.
16. In order to place a heavier weight on depreciation in the earlier years of equipment utilization and, consequently, make the justification more conservative, which is the best depreciation method to use of the five mentioned below:
   a. Double declining balance
   b. Inward approach
   c. Straight line
   d. Curved line
   e. Double straight line.
17. What is the yearly cost of operating three conventional machines on a two-shift basis; 8 hr./shift; @ $6.00/hr.? Consider 2000 man-hours per year, per shift, and fringe benefits amounting to 33⅓% of the direct hourly rate.
18. If one numerical control machine could do the work of the three noted in problem 17 and assuming the same cost and hourly conditions, what would be the yearly savings?
19. According to a careful sample of parts, one numerical control machine operating on two shifts would replace six conventional machines operating on one shift. The hourly labor figure, including fringe benefits, is $12.00 for both conventional and numerical control machines. The yearly tooling cost for the six conventional machines averages $20,000. Numerical control data processing costs would amount to $4,000/year/machine. Programming costs would amount to $6,500/year/machine, and numerical control hard-tooling costs would amount to $1,000/year/machine. In order to utilize the numerical control machine most efficiently, it is necessary that it operate on a two-shift basis, 2000 hr./yr./8 hr. shift. The conventional machines would be expected to continue operation on a one-shift basis. Other savings include a 30% reduction in inventory which normally runs at an average of $50,000 and is considered to have a 10% yearly investment value, and a $4,000 scrap savings. Installed cost for the NC machine is $160,000. A firm offer from the "Fair Play" Used Machinery Company for the six conventional machines is $20,000. Considering a 10-year life for the numerical control machine (refer to the text for the depreciation table), and not considering taxes, tabulate and determine the following:
   a. First year's savings (considering the investment) ________________
   b. Payback period ____________

CHAPTER EIGHT

Organizing for Numerical Control

SELECTION AND PROCUREMENT OF EQUIPMENT

After it has been determined that a particular numerical control application will most likely prove profitable (see Chapters 6 and 7), consideration must then be directed toward procuring the most suitable combination of machine tool and control system.

There are a number of factors that may rightly cause concern when entering or expanding a numerically controlled operation. These are usually due to the numerous *novel* considerations inherent in numerical control. Fortunately, the answers to these considerations are *reasonable, economic* and in most cases have been *proven* under actual manufacturing conditions. Like any advanced manufacturing process numerical control requires careful planning and a more extensive analysis than normally would be given to the procurement of a conventional piece of equipment.

Prior to numerical control, the selection of a standard type machine tool, such as a lathe, drill press, or milling machine was a relatively simple, albeit technical, responsibility. Although improved features such as heavier horsepowers and greater speed-and-feed ranges were being incorporated in conventional machines, the trend was comparatively conservative until the advent of

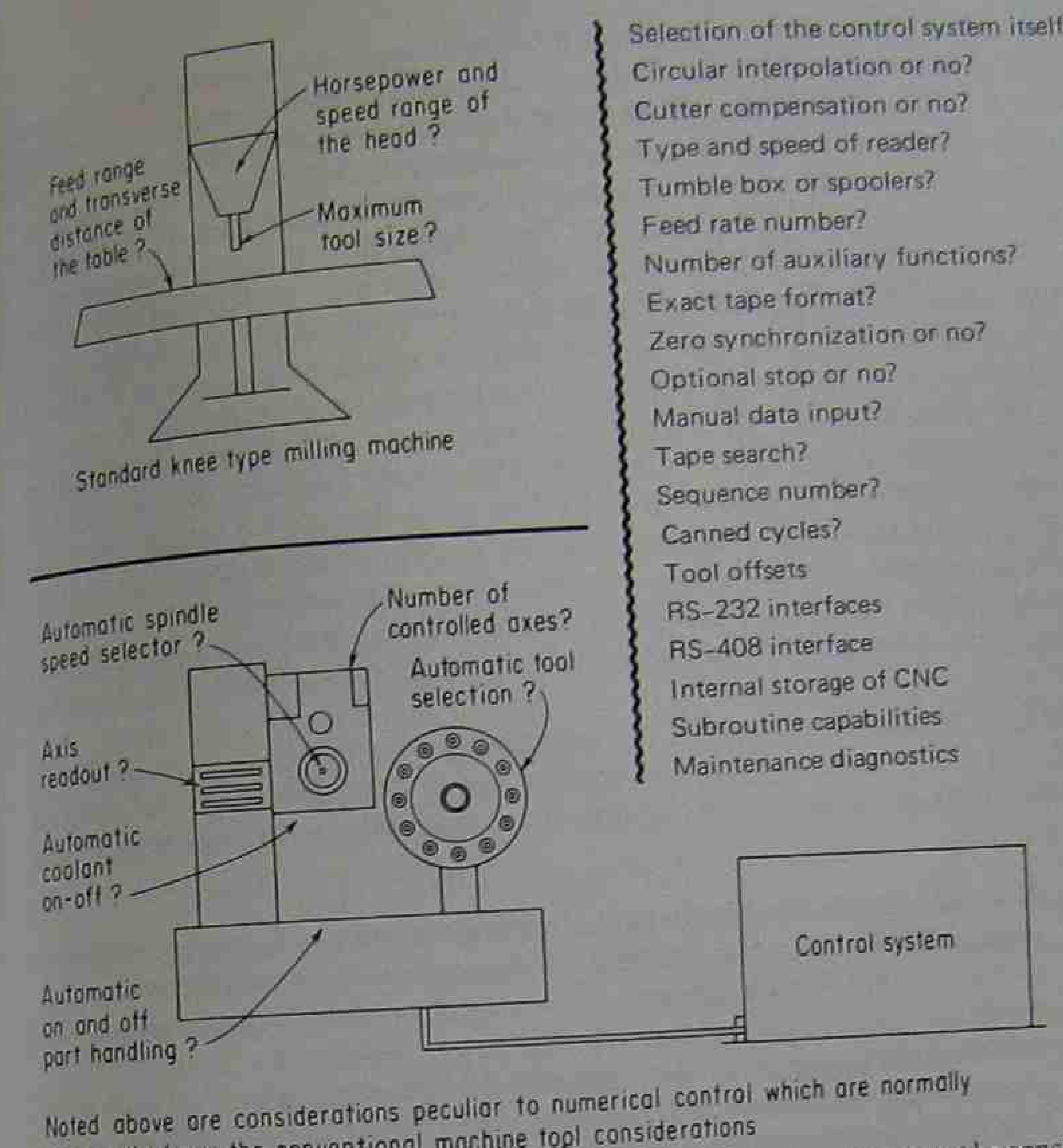


Fig. 8–1. A schematic comparison of some of the major features which must be considered in the purchase of a conventional versus a numerically controlled machine tool.

numerical control. Now, with numerical control it is necessary to become familiar with a machine tool that may be considerably more complex, as well as an electronic control system. While it is not necessary to learn circuitry and electronic theory, one must become sufficiently familiar with the features and operation of the control system to make a judicious selection of the optional features offered. Also, since the control system represents a large portion of the procurement expense, its selection and the features offered should be given the same serious consideration as the machine itself. Although the EIA[1] and AIA[2] have developed numerous standards, particularly with regard to tape input and format (see Chapter 3), a wide latitude for selection still exists. Figure 8–1 offers a comparison of some of the major features to be considered in a

[1] Electronics Industries Association.

[2] Aerospace Industries Association.

standard machine tool and in its numerically controlled counterpart. Because of the various ways of precalculating and storing numerous automatic functions which were previously left to the discretion of the operator, the analysis and economic evaluation of certain features may justifiably become involved unless there is a clear understanding of the precise operation of the feature. (These are explained in Chapter 10.) Since there are, at this writing, approximately 25 different manufacturers' systems available which could be conceivably applied to approximately 60 different manufacturers' machine tools, not to consider other than machine cutting tool applications such as EDM and flame cutting, the difficulty of making the proper decision may be first seen as overpowering especially in view of the multitude of options offered. Quite often, therefore, a purchaser will relinquish the decision for the selection of system features to the machine tool builder even though the builder may not have the same first hand knowledge of the particular shop operation as does the shop manager. It would, therefore, behoove a prospective purchaser of numerical control equipment to have a reasonable understanding of numerical control principles and the numerous features normally offered with a control system.

Other considerations equally as important as the machine tool, CNC software, and the electronic hardware are the various services, type and amount of instruction, guarantees, and computer programs offered by suppliers. These may include:

1. A formal electronic maintenance training course at the system builder's plant. This has been the practice for the more complicated systems such as contouring. The duration of these courses is usually from one to two weeks.
2. A formal part programming course offered by the machine tool builder—either at the customer's or supplier's plant. The duration of the instruction may vary from several hours for the simplest type of PTP drill press to several weeks for the more complicated contouring applications. (This subject is explained in greater detail further on in the chapter.)
3. On-site maintenance instruction during the installation and checkout of the equipment.
4. Ninety days to one year free service covering labor and materials.
5. Various instruction manuals covering:
   a. Programming
   b. Electronic and mechanical maintenance, including preventive maintenance practices
   c. Operator instructions
   d. Parts list with recommended spares.
6. Computer software which comprises the necessary postprocessor peculiar to the machine tool-system combination. This may be in the form of magnetic computer tape or computer cards, supplied by the machine tool builder, which when used in conjunction with a general program such as APT will result in a completely automatic programming system. Part programming and computer programming documentation which

describes the instructions for utilizing the above software should also be furnished. It is also prudent to investigate available sources of computer programming support if it is intended to utilize computer-assisted part programming. There are a number of minicomputer and remote part programming systems available (discussed in Chapter 9).

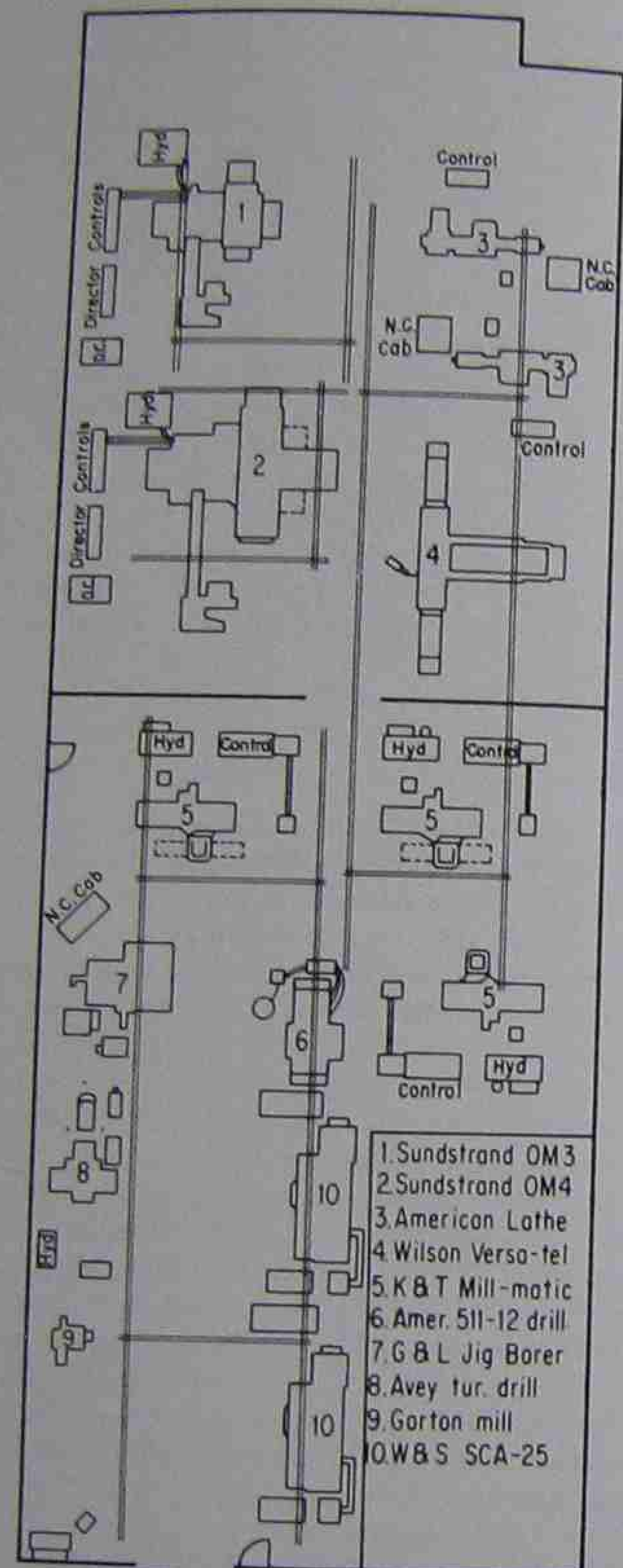
INSTALLATION

After a particular machine tool and its associated controls have been placed on order, some time may pass before delivery. The duration may vary from a few weeks for the simpler machines which possibly can be shipped directly from stock, to over a year for the more costly profilers or skin mills. As with the shipment of any type of engineered product (most numerically controlled machines are built to order), a normal flow time required for the procurement of materials, fabrication, assembly and test must be allowed the manufacturer. This waiting period can be put to extremely good value by the expectant user if a formal preparatory program is initiated within the plant. The number of people involved in such a program and the lead time required prior to delivery will depend on the type and complexity of parts which are expected to be machined by the purchaser as well as the complexity of the machine tool itself.

With regard to the installation of the equipment, which is normally the function of the Plant Engineering or Facilities Department, the lead time period may be advantageously utilized in the gathering of information covering:

1. Power requirements—Most domestic machine tools operate on 220 or 440 volts while their associated control systems operate on a voltage of approximately 115, via a stepdown transformer.
2. Detailed dimension drawings—Describing the overall space requirements, including required operator and maintenance areas. An example of the floor plan of a versatile numerical control shop is shown in Fig. 8-7. In this case various types of NC machine tools are located in one area. Other arrangements would have, for example, NC lathes located with conventional lathes, NC mills together with conventional mills, etc. This latter arrangement has the advantage of maintaining a specific type of operation, such as turning or milling, under specialized functional supervision and is best suited to operations having relatively high production rates involving simpler parts. In those environments where lot sizes are relatively small and parts are reasonably complex it is usually preferable to contain all of the NC machines in one area. This is especially true if machining centers, which combine a variety of operations, are involved. Putting all NC machines in one area also has the advantage of closer proximity to the part programming function since a single part programming area would normally service all NC equipment. It is also easier to service and maintain NC equipment if it is located in one area.

NC machines may also be grouped into "cells" so that like parts may be handled in sequential operations along a line of various types of NC and conventional machines. In order for this kind of arrangement to work parts must be grouped according to common characteristics. The process



Courtesy of Thompson Ramo Wooldridge, Inc.

Fig. 8–2. This layout shows an integrated numerical control machine shop with the milling machines, lathes and drilling machines arranged in one general area rather than spread throughout the plant.

for grouping these parts is called Group Technology. This is explained further in Chapter 9.

3. *Certified foundation drawings*—Where it will be necessary to excavate and pour concrete, "certification" or the "guarantee" by the machine tool builder that the prints are accurate, and any possible error which results in extraordinary costs will be the machine tool builder's responsibility, is common practice with reliable machine tool builders. Many machine tools, particularly of the smaller variety, do not require special foundations and, in this instance, certified prints are not a requisite.
4. *Facilities* required such as air and water.

The above are general considerations when procuring any machine; however, since it is normally expected that a piece of numerically controlled equipment will exhibit superior performance, some special considerations are in order, viz.:

1. Avoid locations near heavy stamping or press-type equipment. Excess vibration may cause electronic malfunctions or, more probably, cause soldered connections to separate.
2. Avoid locations which require that the machine tool be tied into a power transformer bank which supplies other heavy machine loads. Brief line voltage drops might develop due to instantaneous power requirements and may result in a loss of control system synchronization. This is generally not serious since most systems will automatically stop the machine tool in this event. However, undue delay and minor annoyance could result. Usually, an allowable variance for the control system is ±10 percent of the rated 110 volts.
3. Avoid locations in the proximity of heavy magnetic fields. While the delicate portions of most control systems are shielded against outside electrical interference, care should be exercised in avoiding this possible condition.
4. Owing to superior production rates and the higher than normal value of numerical control equipment, accessibility and material handling devices are worthy considerations. Setup time can be held to a minimum, especially when heavy tooling fixtures and parts are being manufactured, if overhead facilities, such as hoists, are strategically placed over the machine tool table. Small jib cranes attached to the more stable portions of a large type machine tool have been found to be extremely helpful in moving large parts on and off machine tool tables.

One means of monitoring a major numerical control (or any other) installation is shown on the chart in Fig. 8–3. This type of pictorial scheduling may also be utilized for other preparatory related numerical control activities. In this hypothetical instance the solid horizontal bars show the scheduled time required to perform specific phases. The cross-hatched bars positioned directly under the solid bars show the actual length of time required. As will be noted from the chart, receipt of the layout and foundation prints, power requirements, and any other pertinent information normally supplied by the equipment builder is scheduled for mid-January. Preparation of the "Request of Bid" for

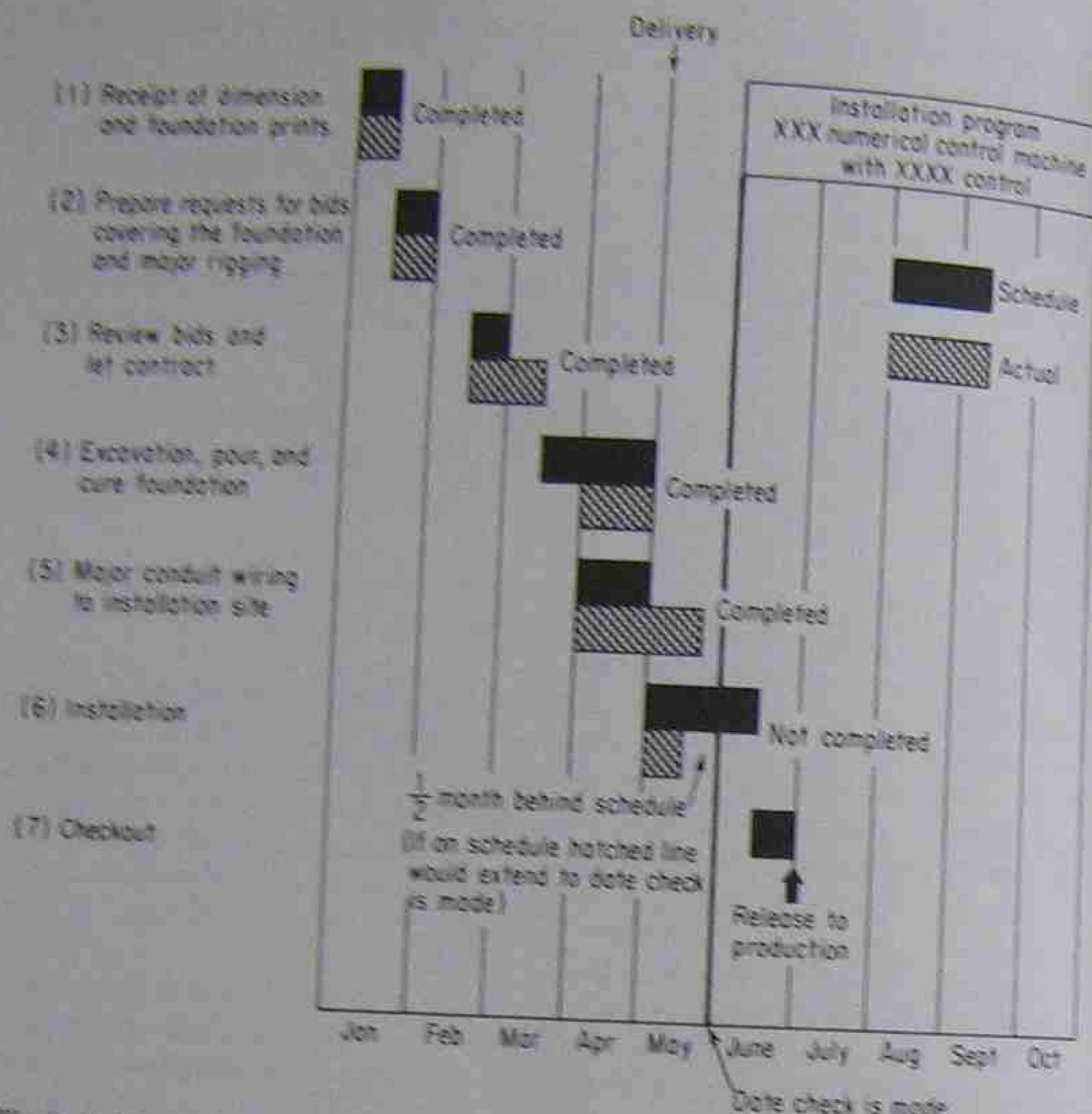


Fig. 8–3. A typical progress chart showing the status of a numerical control installation program. The actual performance is indicated by the cross-hatched horizontal bars which are drawn beneath the solid horizontal bars denoting scheduled periods for completion.

the installation contracting, which is related to the equipment builder's requirements as well as the peculiar soil and other environmental conditions, was scheduled over a two-week period and, according to the accompanying hatched bar, was initiated and completed on schedule. The third item, which entailed reviewing the bids and awarding the installation contract was initiated on time (mid-February); however, the contract was awarded one-half month behind schedule which resulted in a one-half month delay in initiating the "excavation, pour, and 'cure'" of the foundation. This last operation was accelerated, however, and was completed in one month instead of the one and one-half months scheduled, thus moving the project back on schedule. Also, as will be noted, "major conduit wiring" required approximately three-quarters of a month longer than anticipated.

Probably of more significance than describing the history of the program

is the providing of a means of quickly determining the present status. This may be readily accomplished by scribing a heavy vertical line at the date the status check is made and denoting the actual status via the hatched horizontal bar. The chart shown in Fig. 8–3 describes the project status as of June 1 (heavy vertical line located at June 1). According to the schedule, the installation phase (item 6) should be two-thirds accomplished and completed by the middle of June. If, as in the example, this phase was only one-third completed, the installation would be one-half month behind schedule and the hatched bar would be drawn one-third the length of the scheduled time which is one-half month short of the vertical line. The prime virtue of a visual aid of this type (commonly known as a Gantt Chart) is that it offers a ready means of identifying those elements of a program that are not on schedule in order that corrective action be initiated.

MAINTENANCE—GENERAL

Although the technical progression of numerical control equipment design during the past two decades has been outstanding, there is no one likely to declare that machine tools and electronic control systems will operate entirely trouble-free or can readily be maintained by non-technically trained personnel. Indeed, while component and overall design reliability have been improved considerably, the best user assurance of maximum up time lies in well trained on-site maintenance personnel particularly in the electronic area. Contrary to some past advertising proclamations, which may contend that little or no electronic experience is necessary for proper maintenance, it is generally agreed that the best qualified personnel have had some formal electronic training and experience. Electrical maintenance personnel are generally not qualified unless offered formal or extensive on-the-job electronic training. Some companies, utilizing a quantity of numerically controlled machines have established supervisors possessing an electrical engineering degree; however, these are exceptional cases. The image of the "model technician," and that usually required of control systems builders' field service technicians, is a man having a high school diploma, with a better-than-average academic record plus a two-year associate degree in electronics, usually from a state university or technical institute, plus approximately two year's directly related numerical control experience. The requirements for in-plant personnel are usually not as stringent and many installations are operating satisfactorily with maintenance personnel having lesser qualifications. These people, however, have generally been exposed to a formal training program, either on-site or at the equipment manufacturer's plant, the duration of which may run from several days to three or four weeks. A working knowledge of hydraulics, pneumatics and mechanics is also a valuable asset since it is often difficult to determine whether a malfunction lies in the electronic, electrical, mechanical, pneumatic or hydraulic portion of the numerical control package. Regardless of purported simplicity of the numerical control equipment being installed, a qualified and well-trained maintenance technician is an exceptionally fine investment as is the cost of sending technicians to the courses offered by the control system and machine tool builder. One practice that is practically useless is to send a supervisor to a course and expect this person to teach others on his return. The problem with this approach is that when the supervisor does return a good deal of work has probably accumulated and it will take some time to "clear" his desk. During such time he will probably forget most of what he had learned. Also, while he may be a good supervisor, he may not necessarily be a good teacher.

Reasonably good educational qualifications; general experience; and a formal training course alone cannot be expected to produce a service technician fully qualified to handle a particular control system. Additional training may be provided by having service technicians work with the supplier's field service personnel during the installation and checkout period of the equipment. This duration may run anywhere from two to six weeks with the more complex machines and offers in-plant personnel an excellent opportunity to follow the detailed step-by-step operations of a control system and its related machine tool.

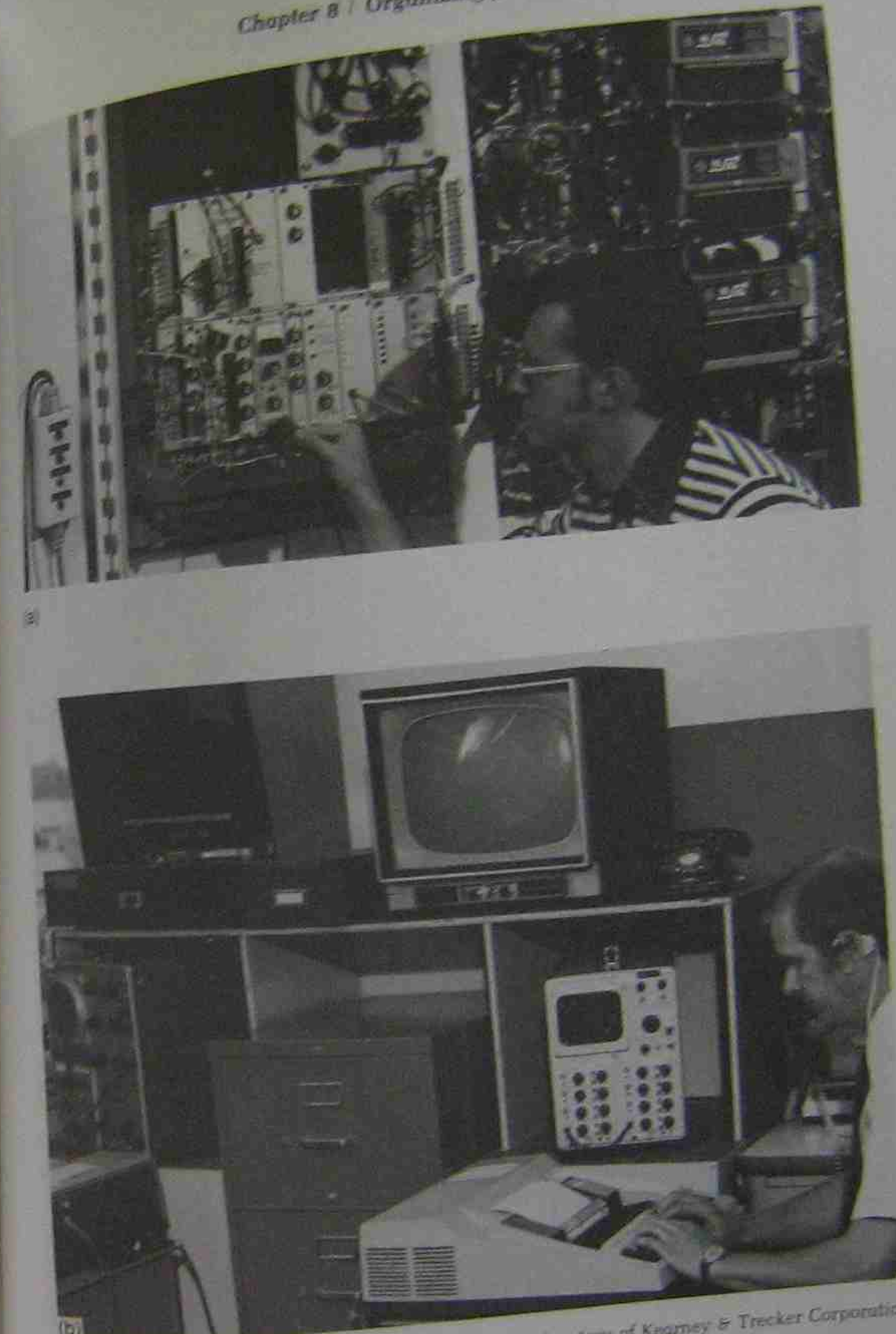
ESTABLISHING AN IN-PLANT MAINTENANCE PROGRAM

A numerical control maintenance "staff" may be comprised of anywhere from a part-time individual to a full-blown department of as many as 40 persons. One Long Island, N.Y. airplane manufacturer, having 37 large complex numerical control machines, most of which are contouring, has a staff of approximately 20 maintenance personnel (mechanical and electronic personnel covering a three shift 6-day week operation). Other facilities have been able to satisfactorily handle as many machines of a simpler nature with 2 to 5 men. Since most numerically controlled machine tools are now being utilized to maximum extent, namely, on two or three shifts, maintenance personnel must necessarily be spread over the full period of operation with the largest staff assigned to the normal day shift. On larger numerical control installations, the ratio between mechanical maintenance and electronic maintenance personnel has generally been split on an approximate 50–50 basis. At the smaller installations, one or two servicemen may be assigned to share both the electronic and the mechanical as well as the electrical responsibilities.

Another alternative is to rely on contract service. One company,[3] as an example, will service a numerical control system, including drives and servo feedbacks and including materials. The fee would depend chiefly on the complexity, age, and condition of the system. Also, certain machine tool builders offer a direct telephone tie-in with their diagnostic testing equipment. The machine tool builder may then analyze the malfunction remotely and recommend corrective action. Normally, however, this is only suitable with CNC equipment. See Fig. 8–4.

Whether personnel should specialize in either the electronic or mechanical requirements or combine these talents is sometimes a moot point. However, it has generally been found that with the more complicated equipment, particularly of the contouring nature, it is advisable for the maintenance personnel to specialize. On the simpler point-to-point hardwire configurations, it may be

[3] Honeywell, Inc., Fort Washington, PA.



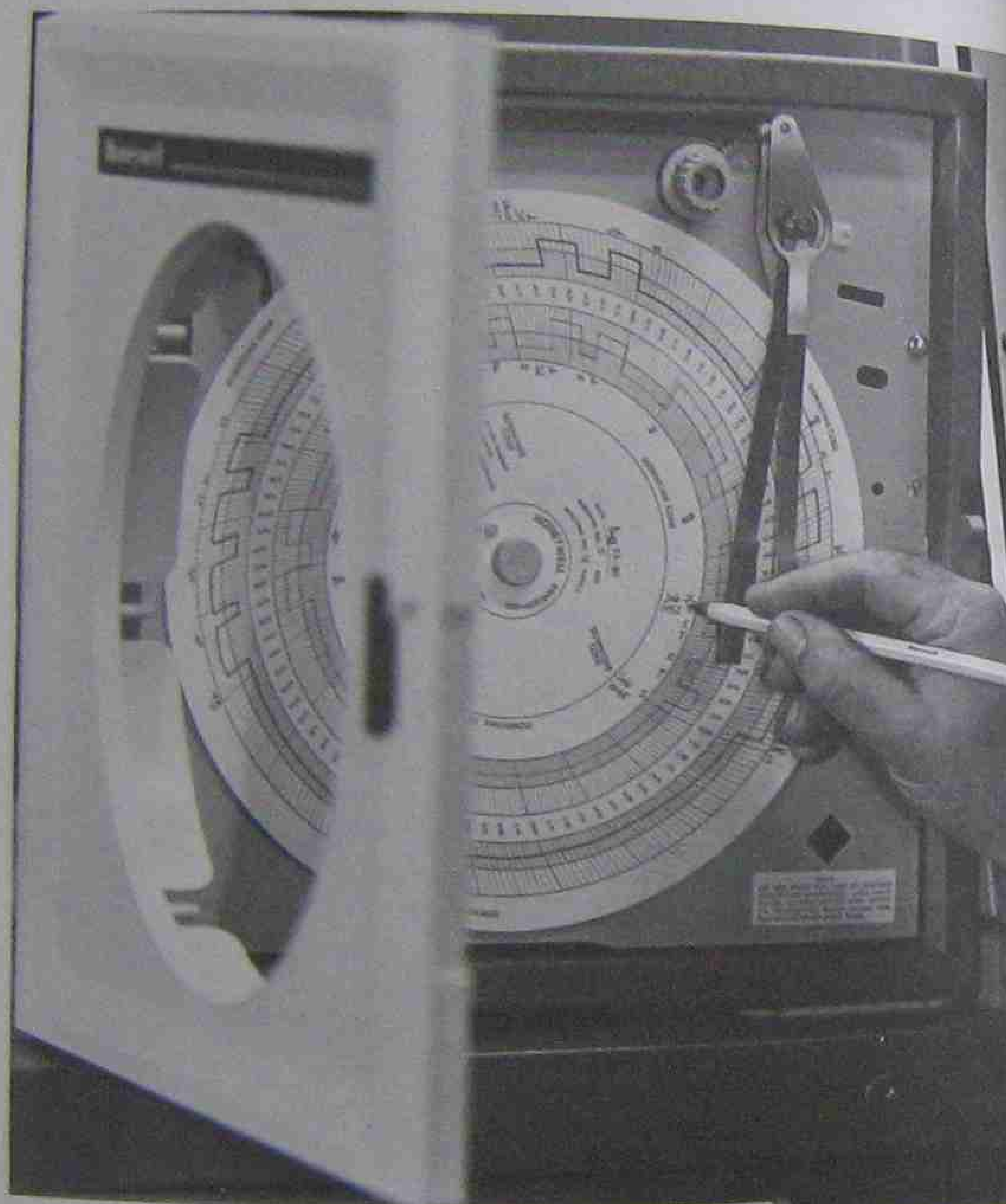
(a)

(b)

Courtesy of Kearney & Trecker Corporation

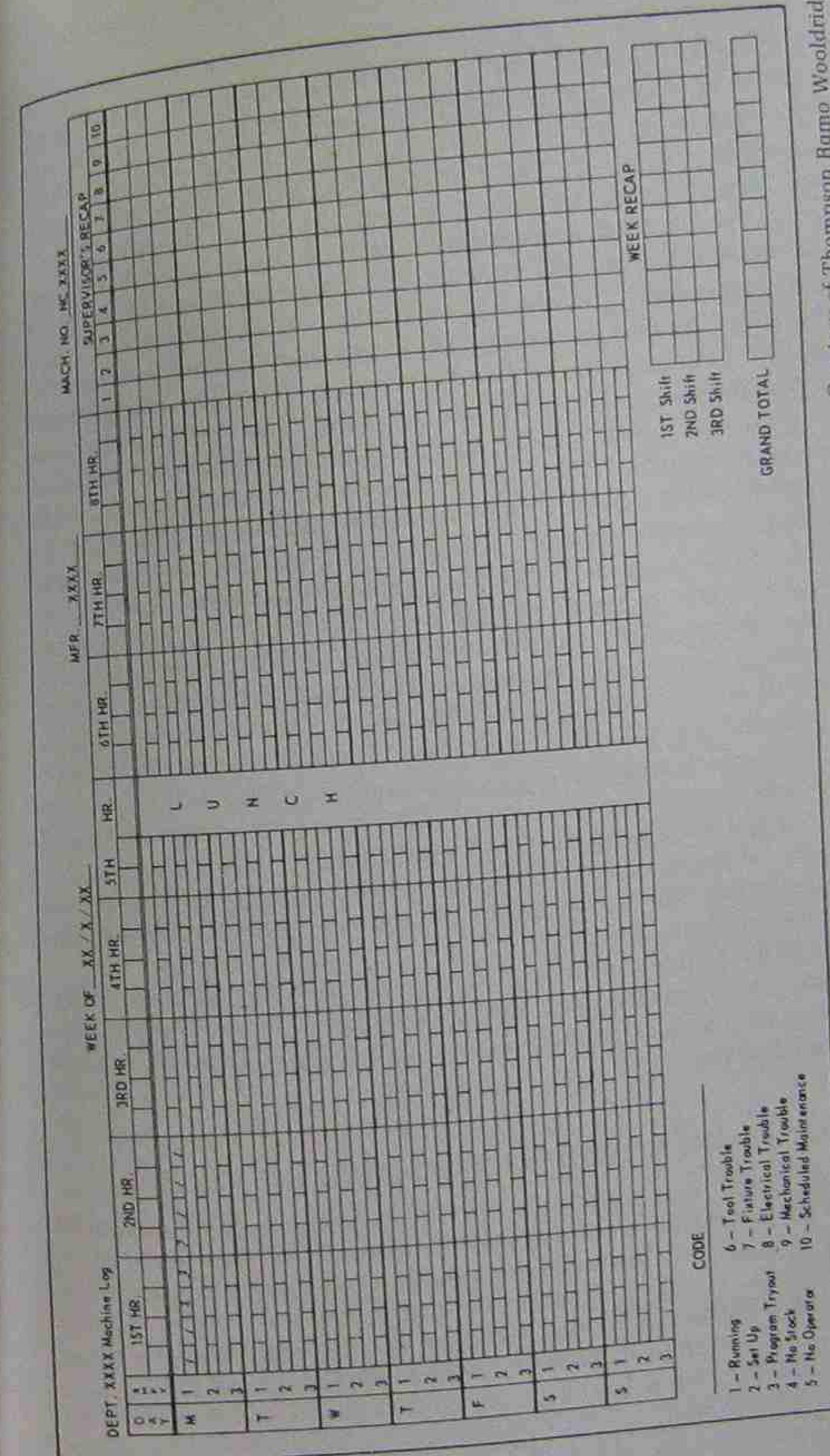
Fig. 8–4. The CNC electronic control system shown in *a*, which is located at the user's plant, may be analyzed at the machine tool builder's plant, shown in *b*, by the use of a remote telephone terminal line and a computer.

possible for one person to handle both the electronic as well as mechanical requirements. If a choice must be made between an individual having a heavy mechanical maintenance background and weak on electronics or an individual heavy on electronics and weak in the mechanical sense, it is recommended the latter be chosen. This is not meant to derogate mechanical maintenance responsibilities; however, because of the somewhat novel requirements in the electronics field when considering numerical control equipment, it has gen-



Courtesy of Honeywell Inc.

Fig. 8–5. The recorder shown automatically tracks uptime vs downtime on an NC machine. The cause for the downtime must be noted by the operator or supervisor. There are other "shop floor" recording systems that are tied directly to a central computer.



Courtesy of Thompson Ramo Wooldridge, Inc.

Fig. 8–6. This chart used in conjunction with the recorder shown in Fig. 8–5 can be utilized for keeping a record of a numerically controlled machine's operating and nonoperating times. The period covered in this instance represents three shifts over a seven-day week. The status of the machine is checked and recorded at 15 minute intervals. A code number (1 through 10) represents the existing condition, such as (1) running, (4) no stock, etc., and is noted in the appropriate box. These data may also be compiled via computer.

erally been found more difficult to obtain trained electronics personnel than mechanical maintenance talent.

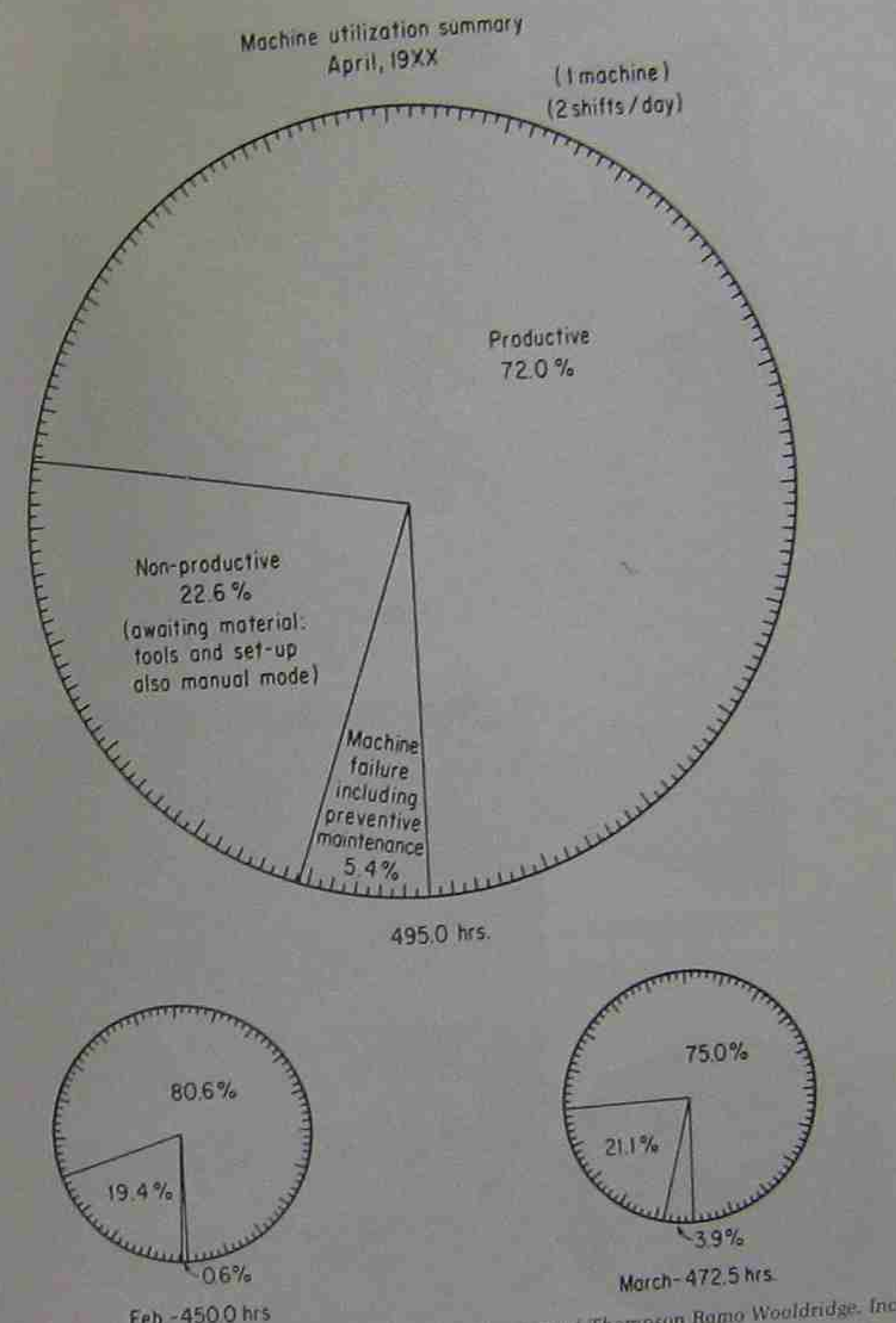
Although obtaining reliable and accurate down time statistical information has proven somewhat difficult, particularly when attempting to distinguish between machine or control system failures, surveys taken some years ago covering large contouring machine tools and controls indicated an average down time of approximately 10 percent, including planned preventive maintenance down time for both machine and control system. The split between the controls and the machine was approximately 50–50. It would be expected that, with the more reliable CNC systems, the total percentage downtime is now less than 10 percent, and the split between the controls and machine would weigh heavier toward the machines. At one plant in the Cleveland area, total downtime for both machine tool and contouring control system of a solid-state second generation configuration ran less than 3 percent on the average over a period of the first five months of operation, including planned downtime for preventive maintenance. While this is certainly a noteworthy record, other installations are reporting system downtime percentages of less than 2 percent for their solid-state control systems and even less for CNC microcomputer systems. In order to be assured of obtaining accurate data, time recorders may be attached to NC machine tools (see Fig. 8–5.) The responsibility for determining whether the downtime was due to machine or system failure or attributable to production problems such as waiting for material, or setup time, must necessarily be left to the operator or supervisor. A close-up of the chart for recording the condition at scheduled times during a shift is shown in Fig. 8–6. Used in a positive and constructive manner, careful scrutiny of the operation and follow-up action will almost certainly improve the percentage up-time of the machine by highlighting the causes of downtime.

A "utilization" summary chart similar to that shown in Fig. 8–7 has proven valuable for illustrating productive time; nonproductive time, other than machine failure; and machine downtime.

These types of shop floor data may also be compiled by a computer. In this case the operator, or supervisor, can assign downtime causes directly into a remote computer which can then compile the data and produce the required management information.

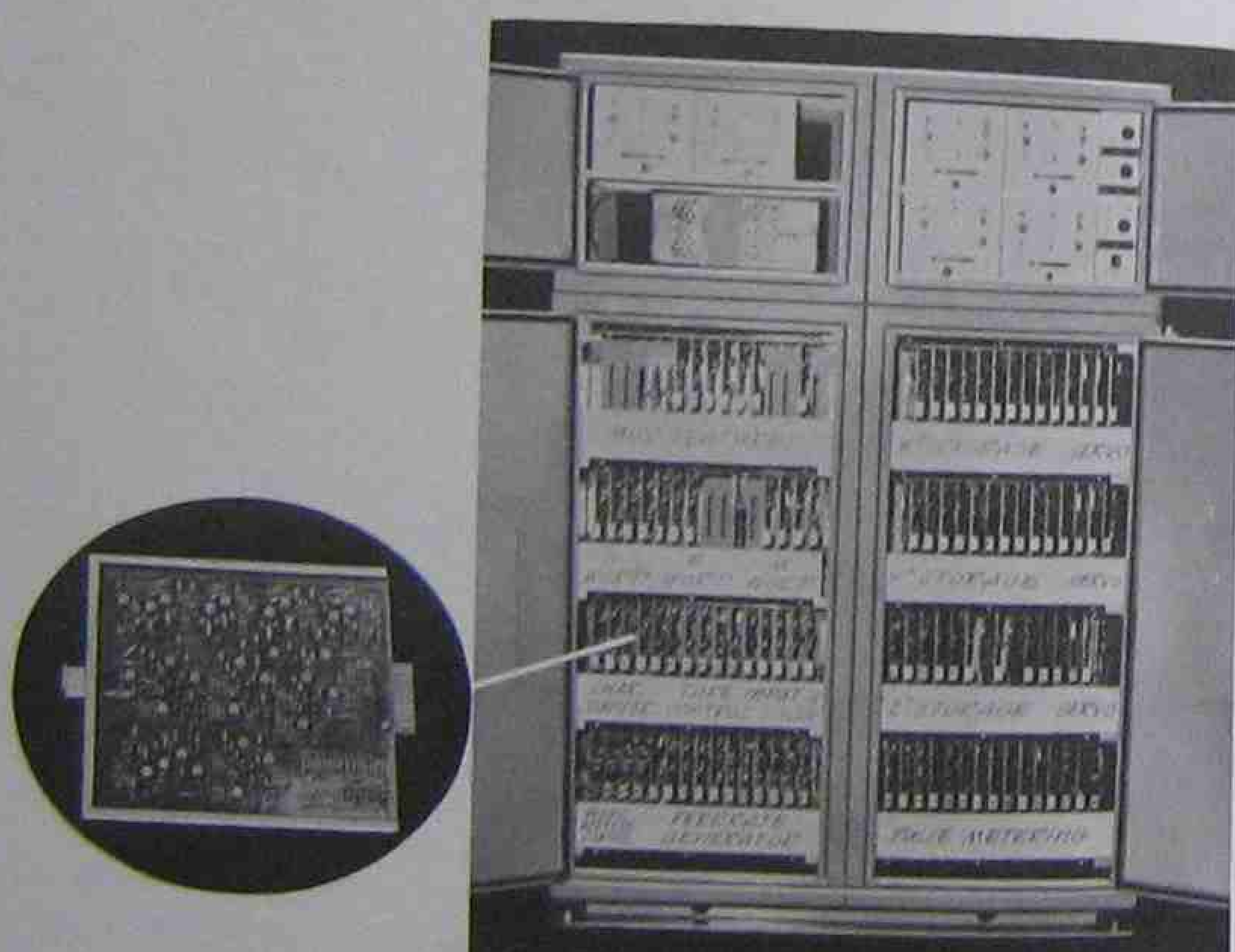
MAINTENANCE DESIGN CONSIDERATIONS (CONTROL SYSTEM)

A. Accessibility While it is admittedly difficult to obtain both an extremely compact yet highly accessible control system, the use of hinged sectional doors and racks plus modular circuit board construction has helped considerably. The photo shown in Fig. 8–8 demonstrates front and rear control cabinet accessibility of an early model hardwire control system. Similar access doors are also designed into other sides of the cabinet shown. Accessibility of a recently designed CNC system is shown in Fig. 8–9. It will also be noted that the CNC system has far fewer circuit boards than the hardwire system shown in Fig. 8–8. Alternatives for access should be carefully considered by the designer and purchaser of numerical control equipment since, quite often,



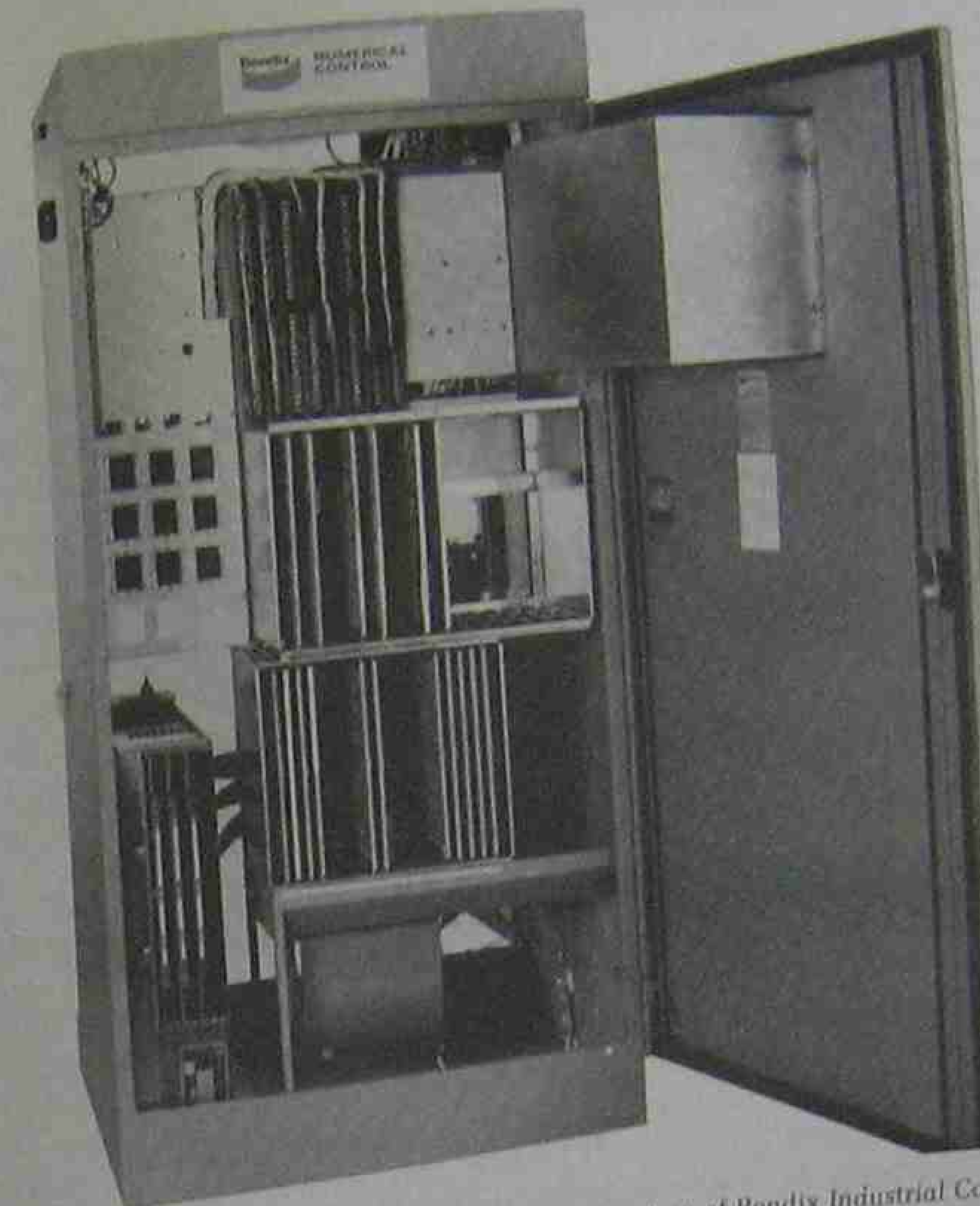
Courtesy of Thompson Ramo Wooldridge, Inc.

Fig. 8–7. These "management type" charts show monthly percentage apportionments of time attributed to: (1) productive time when the machine is operating under the tape mode; (2) non-productive time, when the machine is either being operated manually as in setup operations or when it is waiting for material, tools, tapes or instruction; (3) machine down time due to failure of either the controls or machine tool, and for preventive maintenance. (Figures are hypothetical.)



Courtesy of Bunker Ramo Corp.

Fig. 8–8. Maximum accessibility and solid state module board construction are important maintenance considerations. The placards which describe the various functional sections of the control system are for illustration purposes.



Courtesy of Bendix Industrial Controls Division

Fig. 8–9. Relatively few boards and high accessability as shown above are typical of most CNC systems.

control systems must be located in rather "cramped quarters" on the production shop floor; and normal access means are restricted or eliminated entirely.

B. *Microcomputers* As of this writing microcomputer control systems are proving to be the least costly and most reliable.

C. *Minimum Test Equipment* The ability to maintain the control system with a minimum amount of test equipment should also be sought. The system shown in Fig. 8–8 may be checked with three pieces of equipment, viz., a voltmeter, an ammeter, and an oscilloscope. This is made possible by the use of numerous built-in checking features and indicator test lights, which are an integral part of the control system.

D. *Standard Components* The use of standard "off the shelf" components is extremely desirable. Most systems builders have adhered to this practice since the small profit to be gained by the sale of specialized components and devices is far more detrimentally offset by any down time delays at the user's plant. It is to be expected that the user can obtain most necessary electronic pieces from electronic supply houses within reasonable proximity of his plant.

*E. **Plug-In Modular Components*** Although a fair amount of comment has been made regarding modular design, reiteration may be justified since this is one of the most important single advances in favor of the numerical control industry (see Fig. 8–8 photograph of rear). It has reduced the amount of troubleshooting time required by enabling maintenance personnel to readily detect faulty module boards and quickly insert replacements which may be carried in spares stock. The faulty module may then be analyzed for the exact cause of failure, away from the machine site, while the machine continues in operation.

*F. **Diagnostic Programs*** Software programs may be prepared for the CNC mini- or microcomputer that are capable of testing functions and logic boards. In effect, the computer can be programmed to check itself as well as the machine tool and to assist in locating the sources of error.

PREVENTIVE MAINTENANCE

Another aspect of a maintenance program, which is becoming increasingly popular, is the establishment of preventive maintenance procedures. Since a program of this nature can only be effective if handled on a statistical and regulated basis, the preventive maintenance instructions issued to the user by the equipment supplier must clearly define the requirements and be kept to a practical minimum. Normally there are daily checks which should last probably no longer than from three to ten minutes. Weekly or semi-weekly checks of the control system are also recommended. (See Fig. 8–10.)

One of the major obstacles in implementing a successful preventive maintenance program has been due to the demands for meeting production schedules which result in scheduled preventive maintenance down-time periods as shown in Fig. 8–10 being eliminated in favor of continuing production. This practice can be extremely detrimental to the overall efficiency of numerical control equipment. Preventive maintenance periods must, therefore, be scheduled into production operations and be offered the same consideration as that of a production run of manufactured parts.

PART PROGRAMMING

Undoubtedly, one of the most potentially disturbing considerations of the neophyte about to enter numerical control lies in the requirement for preparing the tape language. While electronic maintenance also presents a challenge, there is nothing essentially novel about the basic electronics, only in the rather new order and application. Programming, on the other hand, is almost always an entirely new experience and, despite the fact that a standard-type coordinate

March 14, 19xx

PREVENTIVE MAINTENANCE SCHEDULE

Week Beginning	Operation Frequency	Down Time Required	Mach. # 10-18800	Mach. # 20-467289	Mach. # 20-467374
April 1, 19xx	2 week	4 hours	Mon.	Tues.	Wed.
April 15, 19xx	"	"	"	"	"
April 29, 19xx	2 wk; 6 wk.	12 hours	Mon-Tues.	Tues-Wed.	Wed-Thurs.
May 13, 19xx	2 week	4 hours	Mon.	Tues.	Wed.
May 27, 19xx	"	"	"	"	"
June 3, 19xx	12 week	16 hours	Mon-Tues.	Wed-Thurs.	
June 10, 19xx	2 wk-6 wk-12 wk	12 hrs-24 hrs.	Mon.	Tues.	Wed-Thurs-Fri.
June 24, 19xx	2 week	4 hours	Mon.	Tues.	Wed.
July 8, 19xx	"	"	"	"	"
July 22, 19xx	2 wk; 6 wk.	12 hours	Mon-Tues.	Tues-Wed.	Wed-Thurs.
August 5, 19xx	2 week	4 hours	Mon.	Tues.	Wed.
August 19, 19xx	"	"	"	"	"
August 26, 19xx	12 wk.	16 hours	Mon-Tues.	Wed-Thurs.	
Sept. 2, 19xx	2 wk-6 wk-12 wk.	12 hrs-24 hrs.	Mon.	Tues.	Wed-Thurs-Fri
Sept. 16, 19xx	2 week	4 hours	Mon.	Tues.	Wed.
Sept. 30, 19xx	"	"	"	"	"
Oct. 14, 19xx	2 wk; 6 wk.	12 hours	Mon-Tues.	Tues-Wed.	Wed-Thurs.
Oct. 28, 19xx	2 week	4 hours	Mon.	Tues.	Wed.
Nov. 11, 19xx	"	"	"	"	"
Nov. 18, 19xx	12 wk.	16 hours	Mon-Tues.	Wed-Thurs.	
Nov. 25, 19xx	2 wk-6 wk-12 wk.	12 hrs-24 hrs.	Mon.	Tues.	Wed-Thurs-Fri.
Dec. 9, 19xx	2 week	4 hours	Mon.	Tues.	Wed.
Dec. 23, 19xx	"	"	"	"	"
Jan. 6, 19xx	2 wk; 6 wk.	12 hours	Mon-Tues.	Tues-Wed.	Wed-Thurs.
Jan. 20, 19xx	2 week	4 hours	Mon.	Tues.	Wed.
ETC.					

These cycles will be continuous at frequencies as shown.

John C. Smith, Supervisor

G.T.

Fig. 8–10. This is a typical preventive maintenance schedule which has been incorporated into the production planning program of a numerical control facility.

system is employed which was likely learned in high school and may have long since been forgotten, the new vocabulary consisting of "bits," "characters," "levels," "binary coded decimal," and "blocks," although familiar to most computer personnel is generally foreign to manufacturing people. The practice is to *train* programmers rather than attempt to hire them directly from outside sources, since well-trained programmers are scarce and the demand far exceeds the supply. Potential in-house programmers are normally available working in fields such as tool design and methods engineering. "First class" machinists have also proven to be quite versatile and "convertible" to part programmers. The degree of training required for a successful part programmer will depend on:

1. The *educational* and *experience* background of the trainee
2. Whether *point-to-point* or *contour* programming is involved
3. The *complexity* of parts to be machined
4. The amount of *computer* assistance available.

The basic requirements of a candidate for *any* type of part programming consist of:

1. A usable knowledge of shop mathematics including basic trigonometry
2. Thorough working familiarity with blueprints
3. Knowledge of machining practices involving cutting rates and sequence of operations.

These three requisites are necessary even for programming the simplest of parts which may consist of drilling several holes in a flat piece of stock. (See Fig. 8–11.)

In addition to the basic requisites, PTP part programmers who are required to prepare manuscripts[4] involving a multitude of points or angular motions, may benefit from a working knowledge of algebra and plane geometry as well as solid and descriptive geometry.

Contour part programmers, in addition to being highly proficient in the basic requisites, can benefit substantially by possessing an above average aptitude for mathematics up through and including analytic geometry, although this level may not always be mandatory and will depend almost solely on the complexity of the parts to be manufactured and the extent of engineering definition offered by the designer.

It should be pointed out that the *part programmer* need not understand the functioning of a computer or *computer programming* techniques in order to prepare the manuscript information that will be utilized by the computer. The numerous computer-supported part programming languages available generally negate this requirement although an understanding of computer operating principles may aid the part programmer in his coordinating activities with the computer programmer.

The details of both manual and computer-aided part programming are covered in Chapters 4 and 5.

[4] Form used for detailing machining instructions (see Chapter 4).

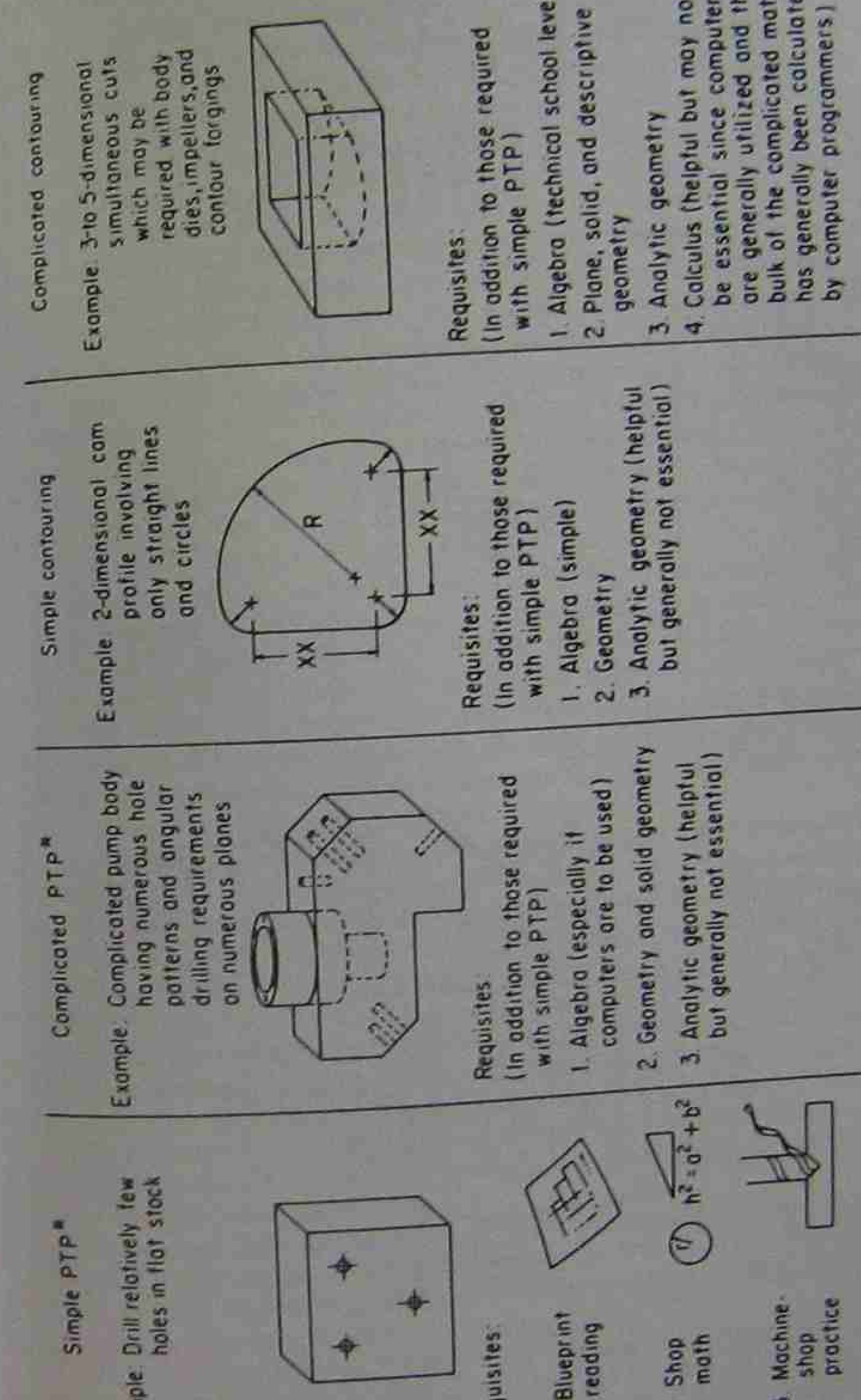


Fig. 8–11. Qualifications for personnel who are to program parts of varying degrees of complexity.

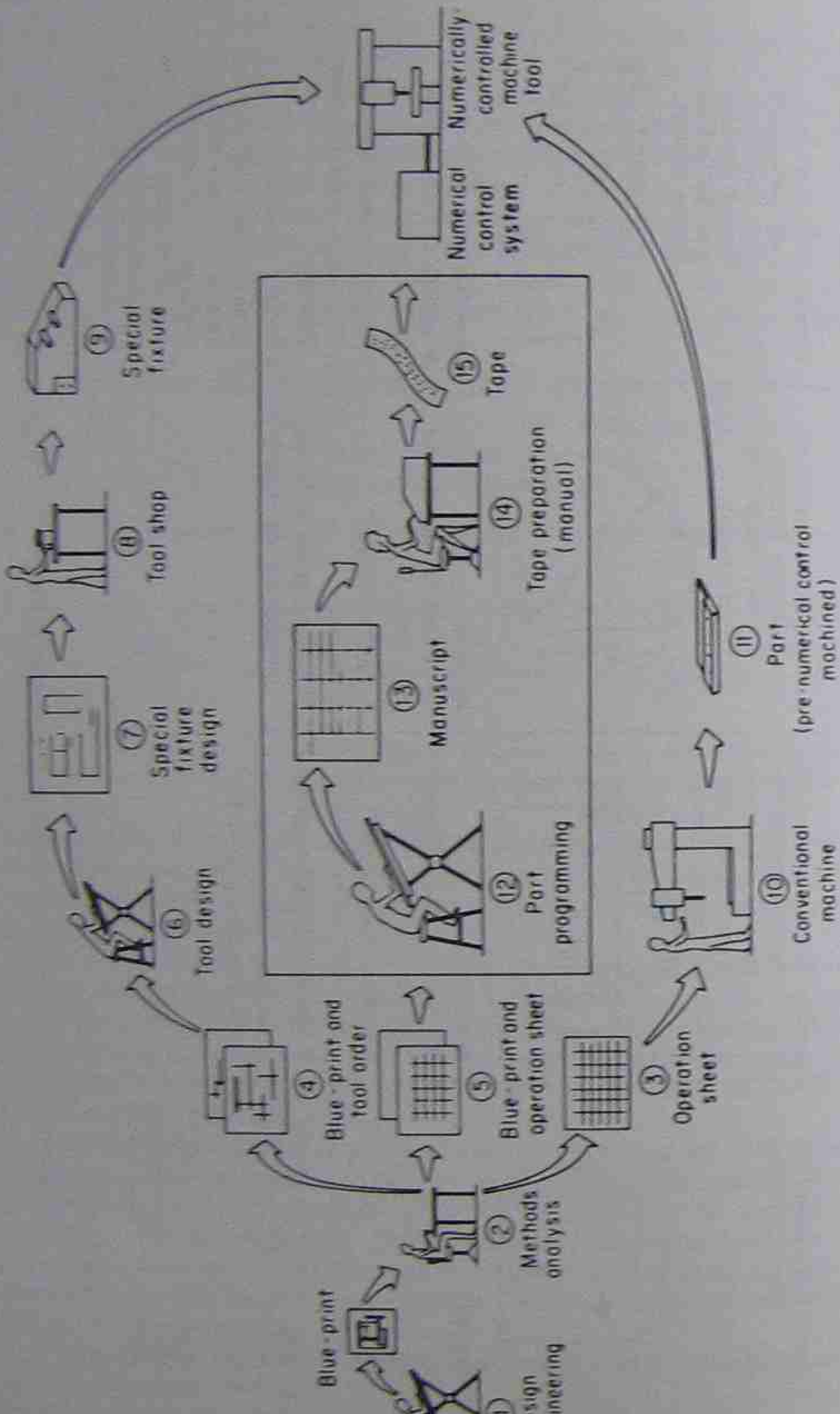


Fig. 8–12. Steps required for manually programming a tape. Also shown are the steps required when special holding fixtures are needed and when machining operations other than those which are numerically controlled are necessary such as tooling holes or banking surfaces for positioning a part on the numerically controlled machine.

TAPE PREPARATION

The input medium to the control system, which is generally in the form of a punched tape (see Chapter 3), may be prepared either manually via a special tape punching typewriter or automatically via a computer. The means sought will depend to a great extent on the number and, particularly, the complexity of parts required to be programmed. Where parts fall in the PTP category and are relatively simple, manual programming has been proven reasonably successful. This applies especially to CNC systems that have computing capability (see Chapter 4). Manual programming may also prove feasible with simple contour parts involving straight lines and circles on a two-dimensional plane. Where the quantity of different parts is heavy or complex, whether PTP or contouring, computer assist whether for hardwire or CNC controls has been used to exceptional advantage in reducing the number of manual calculations.

Figure 8–12 describes the functions required for manually preparing a numerical control tape. Whether these functions are carried on by different departments, or combined under one or a few individuals, depends on the size of the company. The circled numbers in the paragraphs below refer to the circled numbers shown in Fig. 8–12.

Engineering prints may be prepared in either conventional format or, preferably, in numerical control language ① and passed on to the methods analyst ②. Although the general practice thus far has been to convert conventional design formats, since numerically controlled machines are usually incorporated into an existing operation where conventional prints predominate, it is expected that, hopefully, engineering drafting practices will eventually conform to design numerical control coordinate language. The part programmer's task will then be considerably simpler.

In addition to numerical control operations, most parts require associated operations such as the drilling of tooling holes, deburring, heat-treating, or painting. In accordance with standard practice *all* operations are therefore generally listed on an *operation sheet* or *schedule* ③ together with appropriate accounting charge numbers and, perhaps, time study standards. Special holding fixtures may also be required, and the necessary *tool orders* may also be prepared by the *methods engineer* ④. Also, the responsibility for selecting the parts most suitable for numerical control may rest with the methods analyst. The form shown in Fig. 8–13 describes one means of analysis being utilized in order to determine whether a part presently being machined conventionally should be diverted to numerical control. Parts demonstrating the greatest potential savings would be selected.

Should a special holding fixture be required, steps ⑥ through ⑨ follow the release of the tool order from the methods department. Steps ⑩ and ⑪ would be the result of an operation sheet calling for machining operations other than and prior to numerical control machining.

The tape preparation function which is initiated by the receipt of the operation sheet and blueprint ⑤ is shown in steps ⑫ through ⑮. Following the methods analysis, the part programmer must detail every operation and all axes movements. With manual programming as in this case, a precise block-by-block description must be outlined on the manuscript ⑬ since the tape will

COST ANALYSIS

NUMERICAL CONTROL vs. CONVENTIONAL MACHINING

Prepared By: – ABC
Acct'g.: DE
Fact. Eng.: F.G.
Mfg.: H.I.J.

Present – Conventional

Mach.	Oper.	Dept.	Oper. Description	Std. Set-up Hrs.	Std. Run.Hrs./C.	Actual S.U. Hrs.	Actual Run.Hrs./C.
BAND SAW	1	MACH SHOP	CUT BLANK FROM STOCK	.02	.15	.02	.18
PROFILER	2	" "	MILL PERIMETER	.93	.36	.90	.34
BORING MILL	3	" "	DRILL AND BORE 8 HOLES	.30	.58	.41	.66
—	4	PAINT	SPRAY PAINT	—	.01	—	.01
—	5	STORES	TO STORES	—	—	—	—
			TOTAL	1.25	1.10	1.33	1.19

Tooling Costs:						
MILLING TEMPLATE	$540	Labor $:	10.00	8.25	10.65	9.50
DRILL JIG	1860	O.H. $:	10.00	8.25	10.65	9.50
TOTAL	$ 2400	Total $:	20.00	16.50	21.30	19.00

DECISION: N.C. ☒ CONVENTIONAL ☐

Numerical Control

Mach.	Oper.	Dept.	Oper. Description	Std. Set-up Hrs.	Std. Run.Hrs./C.	Actual S.U. Hrs.	Actual Run.Hrs./C.
BAND SAW	1	MACH SHOP	CUT BLANK FROM STOCK	.02	.15	.02	.18
MACH CENTER	2	" "	MILL PERIMETER, DRILL & BORE 8 HOLES	.25	.30	.29	.26
	3	PAINT	SPRAY PAINT		.01		.01
	4	STORES	TO STORES				
			TOTAL	.27	.46	.31	.45

Tooling Costs:						
TAPE (MILL & DRILL)	$720	Labor $:	2.16	3.34	7.44	3.60
		O.H. $:	2.16	3.34	7.44	3.60
		Total $:	4.32	6.67	14.88	7.20

Courtesy of Thompson Ramo Wooldridge, Inc.

Fig. 8–13. A hypothetical type of analysis form that may be utilized in order to determine the selection of parts most suitable for numerical control machining. In this example numerical control machining would clearly be the favored choice. (Figures are hypothetical.)

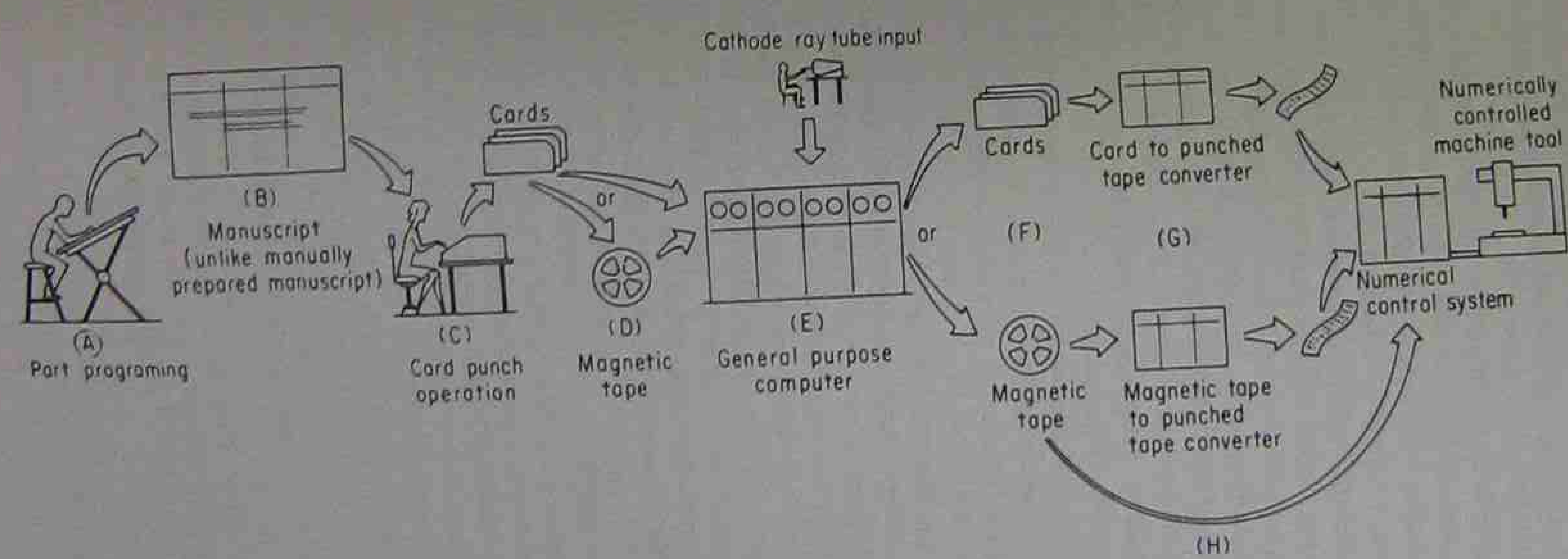


Fig. 8–14. The tape preparation steps shown above involve the use of a large scale general purpose electronic computer. As with manual programming, the part programmer (A) prepares a manuscript (B). This manuscript, however, differs considerably from that required for manual programming. Computer cards are then punched (C) directly from the manuscript information. These cards may be inserted directly into the computer or converted to magnetic tape (D) for computer insertion. The computer (E) produces cards or magnetic tape (F) which are automatically converted to punched tape (G) for the numerical control system. One development eliminates steps (A), (B), (C), and (D), by allowing a programmer to communicate directly with the computer via a cathode ray tube (CRT) and a light pen. Other arrangements of computer-assist part programming are described in Chapter 5.

be prepared directly from the manually calculated data on the manuscript (see Chapters 3 and 4). The tape, manually prepared by the desk typing device, may then be inserted directly into the control system.

The steps required when utilizing a computer are shown in Fig. 8–14. Since the essential difference occurs directly in the part programming area, comparison will be restricted to these operations shown in steps ⑫ through ⑬ of Fig. 8–12. Although step (A) in Fig. 8–14 and ⑫ in Fig. 8–12 are both designated as a "part programming" function there is a considerable difference in the amount of detailed description required when preparing a manuscript for manual tape preparation and when preparing one for computer input since the computer performs the bulk of the calculations which are required in manual programming. Although more steps are involved when preparing tapes via the computer route, the flow time and cost are considerably less. A job which might take several weeks to program manually may be reduced to days or even hours by utilization of a computer.

The flow procedure shown in Fig. 8–14 involves a large scale in-house computer. Computer assist is also available via small in-house minicomputers, or by a remote telephone line connection to a large-scale computer. These systems are described in Chapter 5.

Referring to Fig. 8–14, the manuscript (B) is passed to a "card punch" operator (C) who then types computer cards directly from the listed manuscript information. These cards may then be processed directly into the computer or converted to magnetic tape information, which is then read into the computer (E). The reason for transferring information onto magnetic tape is that data may be fed in and out of the computer much more quickly than with punched cards. The total computer time required, which is relatively expensive, may then be kept to a minimum. Most large-scale computers, therefore, now performing numerical control calculations utilize magnetic tape. A more common practice is for the part programmer to have direct, interactive access to the computer via a keyboard. This would eliminate steps (C), (D), (F), and (G).

One recent development also eliminates steps (A) and (B) by allowing the part programmer to "draw" the path of the cutting tool on the face of a cathode ray tube (CRT) which is tied-in directly to the computer. (See Fig. 9–2.)

GENERAL

The extent to which a manufacturing organization must prepare and adjust for numerical control will depend on a number of factors, viz.:

1. The type and complexity of parts to be manufactured, i.e., whether PTP or contouring.
2. The in-house talent and the ability to attract personnel with the necessary backgrounds, assuming that most in-house personnel will require specific training in numerical control and for the particular machine and control system to be utilized.
3. The availability of computer aid.
4. The extent of formal procedures already established covering areas such as: (a) methods; (b) planning; (c) design engineering; (d) industrial engineering; (e) tool design; and (f) production control.

Since, admittedly, numerical control has reallocated much of the methods and planning responsibilities from the shop floor to the drafting board, the ease with which a relatively small shop can adjust will be inversely related to its dependence on handling manufacturing operations at the machine tool sites. The extreme example, and unfortunately not too uncommon, is a shop where the operator or machinist is offered an "engineering sketch" and instructed to "machine the part per sketch." The entire responsibility for (1) planning; (2) tooling; (3) methods; and (4) machining is left to the operator or machinist. Although this approach has sometimes been found satisfactory for producing research or prototype parts, there are several notable drawbacks. First, the machine tool sits idle a large percentage of the time while the operator is performing other than machining functions. Actual cutting time may average as low as 5 or 10 percent in this instance. Secondly, we cannot generally expect one person to possess the equivalent capabilities of several specialists. The proper place to *plan* a job is in a planning department. Similarly, the proper location for *machining* a part is at the machine tool site. In order to realize the highest operating efficiency, as many associated machine tool responsibilities as possible should be allocated to areas apart from the machine tool site. This is particularly true of numerically controlled machines because of their relatively high cost. In order to keep a numerically controlled machine tool operating at peak efficiency, more careful planning is required than with conventional equipment. The fact that complete machining instructions can be prepared apart from the machine-site with numerical control engenders the suitability of more precise planning in other manufacturing associated areas such as production control, inventory and quality control. The long sought goal when an operator might remain at his machine without having to "check" with the tool crib, engineering, methods department, or other sundry sources may be realized with a properly organized and administered numerical control operation. Numerical control, as with any advancing technology, requires a greater degree of specialization. The numerous functions and talents required of a conventional machine operator are now assigned to different individuals such as the part programmer, card punch operator, computer programmer, tool setter, etc. The net result is a more accurate job, a greater percentage of the time, at less cost per piece. The supporting "team" behind the numerical control machine tool operator is described in Fig. 8–15. Although, at first glance, the numerous requirements may appear somewhat frightening to the uninitiated, actually the only truly novel requirement is the part programmer. The other functions generally exist either separately or combined under a few individuals as is the custom with smaller shops. The one key aspect with numerical control that may present a somewhat novel challenge is that the instructions to the operator and machine, in the form of tapes and tooling, must be *precise*. The operator has far less latitude to analyze the machining requirements and form his own judgments. Correcting errors at the machine

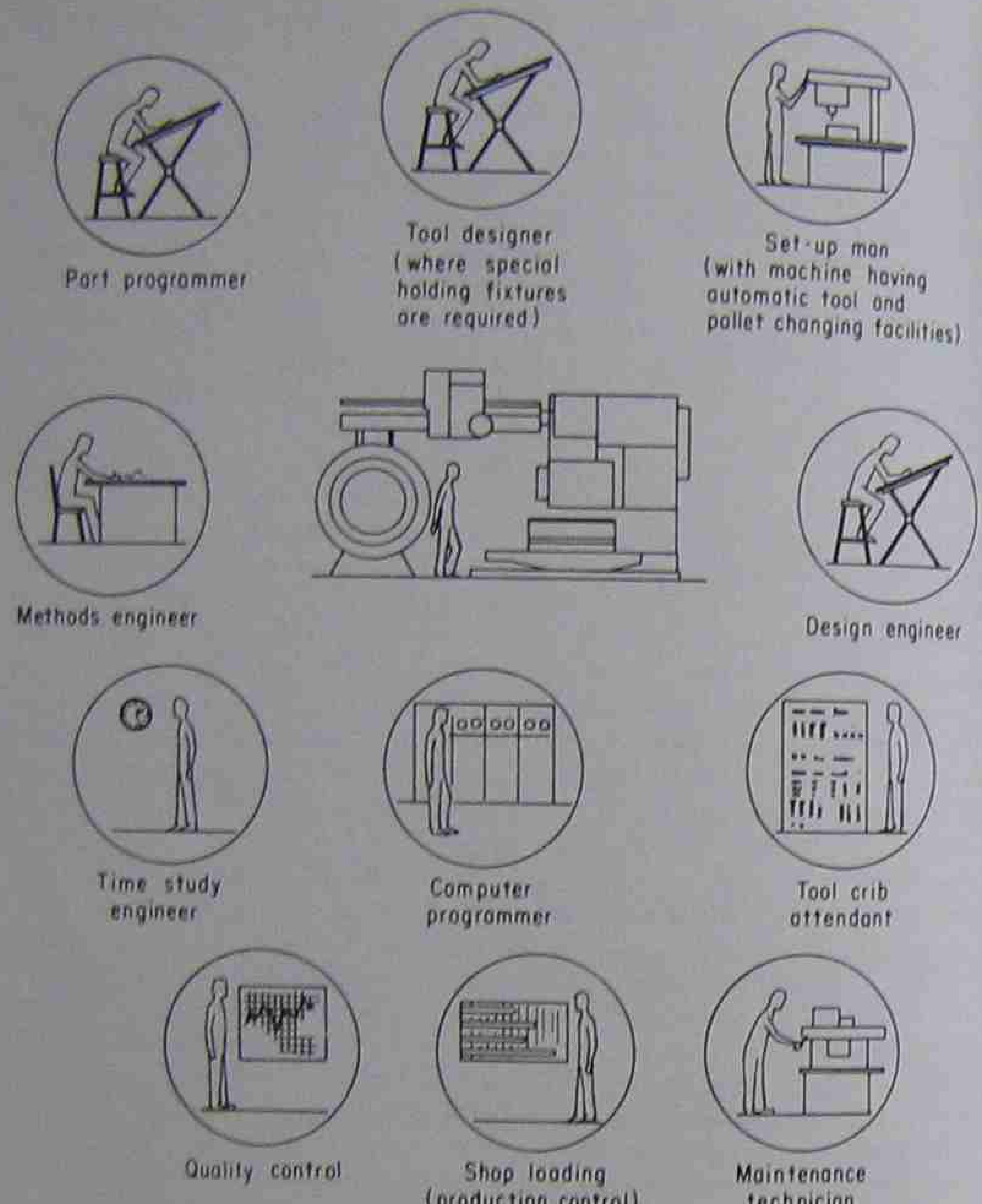


Fig. 8–15. The extent of successful utilization of a numerical control installation depends on the efficiency of the supporting "team." Tape data, tooling, and other requirements delivered to the machine tool site must be adequate and precise.

site, while less troublesome with CNC than hardwire systems, can prove a costly procedure, both in damaged parts and lost time.

QUESTIONS TO CHAPTER 8

1. A numerical control machine and its associated control system are considerably more complex than their conventional counterpart. The user must therefore become familiar with numerical control terminology. Next to the

terms listed below, note whether the term is applicable to the machine tool (MT); control system (CS); external computer used for part programming (C); or none of the three (N). Although a machine tool may perform an operation, if the operation is peculiar to a numerically controlled machine and initiated by the control system it is to be considered a (CS) feature. As an example, "circular interpolation" would be considered a (CS) feature, whereas "drive unit" would be considered an (MT) feature. (Suggestion: In addition to checking this chapter also check Chapters 3, 4, 5, and 10.)

1. Manual data input
2. Automatic tool changer
3. Automatic spindle lock
4. Tape search
5. Digital storage
6. Hydraulic pressure
7. Tab sequential format
8. Zero shift
9. Postprocessor
10. Sequence number
11. Off-line operation
12. Machine language
13. Magnetics
14. Feedback package
15. Track
16. Subroutine
17. APT or COMPACT II
18. Tool setting
19. Quantum
20. Backlash

2. Since a good deal of instruction and documentation normally accompanies the delivery of a numerical control machine, little in-plant preparation is required. (True, False.)
3. Preventive maintenance programs for NC equipment should be very flexible and arranged or rearranged to suit production schedules. (True, False.)
4. Certified foundation drawings are the machine tool builder's guarantee that the prints are accurate and that any error resulting in extraordinary costs will be borne by the machine tool builder. (True, False.)
5. Referring to Fig. 8–3; if a check were made of item 4, "Excavation, pour, and cure foundation," at the end of June, and it was found that this item was approximately 2 months behind schedule, where would the cross hatched bar be located?
   a. Where it is shown in Fig. 8–3
   b. At the end of June
   c. At the end of August
   d. Not shown at all.
6. Referring again to Fig. 8–3, had item 6, "Installation," been on schedule at the end of May, where would the cross-hatched bar then belong?
   a. The middle of June
   b. Approximately the third week in May
   c. At the end of May.
7. Referring once again to Fig. 8–3, the cross-hatched bar in item 3, "Review bids and let contract," shows that as of the end of May, this item:
   a. Was completed 2½ months ahead of schedule
   b. Is completed and was approximately 2 weeks behind schedule
   c. Was completed on schedule.
8. Since a numerical control system requires electrical power and exists in an electronic environment, it is beneficial to locate this equipment as near

to electrical transformers and generators as the plant-layout arrangement will allow, in order to reduce the power loss between the source and the control system. (True, False.)

9. The best user assurance of maximum uptime is:
   a. A maintenance insurance policy with the equipment manufacturer
   b. A direct phone line to the manufacturer's field service department
   c. Well-trained on-site maintenance personnel
   d. Good set of spare parts.
10. The best way to train maintenance personnel is:
    a. By means of a correspondence course
    b. By means of a formal training course offered at the equipment builder's plant, prior to installation of the equipment
    c. On-site, while the equipment is being installed
    d. On-site, following installation of the equipment.
    e. Send only the supervisor to the school and have him train others on his return.
11. Standard "off-the-shelf" components are highly desirable. (True, False.)
12. Of the four qualifications shown in Fig. 8–11, which one would best fit the requirements for programming the part described in Question 4 to Chapter 4?
13. The most practical means, at present, of acquiring part programmers is to:
    a. Place an ad in a newspaper
    b. Train them in-plant
    c. Consult an employment agency.
14. Since the part programmer must convert conventional drafting descriptions to the numerical control coordinate language, most companies are now describing these parts in the coordinate language system thus eliminating this requirement of the part programmer and thereby reducing the overall part-programming time considerably. (True, False.)
15. All operations involving a part are generally listed on:
    a. A manuscript
    b. A computer listing
    c. Tool orders forms
    d. Operation sheets
    e. Blue-prints.
16. Downtime for both machine and control system, and including preventive maintenance, normally should not exceed: a. 1%; b. 5%; c. 10%; d. 25%; e. 50%.
17. Complex contour part programming generally requires a working knowledge of analytic geometry. (True, False.)
18. Referring to Fig. 8–14, the steps in preparing a tape via a computer are automatic (i.e., require no further human recording or calculating) after:
    a. The manuscript is written
    b. The computer cards are punched
    c. The magnetic tape is prepared.
19. The manuscript for preparing a manual part program is generally (longer), (shorter), than a part program prepared for a computer.

20. Referring to Fig. 8–15, match the function with the personnel.

Personnel	Function
a. Part programmer	1. Arranges for automatic tool changers
b. Tool designer	2. Keeps equipment operating
c. Setup man	3. Keeps material coming to machine
d. Design engineer	4. Determines man-hour requirements
e. Tool crib attendant	5. Maintains stock of cutters, *et al.*
f. Maintenance technician	6. Maintains raw and finished stock levels
g. Shop loader	7. Prepares manuscript
h. Inventory controller	8. Specifies *all* operations
i. Time-study engineer	9. Designs special fixtures
j. Methods engineer	10. Designs parts

CHAPTER NINE

Future Trends

Although numerical control may not have experienced the phenomenal growth some had predicted, most will agree that it has enjoyed a reasonably rapid climb. (See Chapter 1.)

There are several factors that are expected to hasten the wider utilization of numerical control and the most significant of these are:

1. The use of the microcomputer as a control system. Microcomputers are less expensive, more reliable, and offer many more features than hardwire systems. This has tended to reduce the cost of many numerical control machine tools. The three-axis NC milling machine shown in Fig. 1–10, for example, sells for approximately $38,000 at the time of this writing. The OEM[1] cost of a hardwire three axis control system alone was $38,000 in the 1960–1965 era. And the machine shown in Fig. 1–10 has far more capability.
2. Machine tools that are designed for numerical control. Many earlier NC machines were conventionally designed machines with numerical control systems attached.

[1] OEM stands for Original Equipment Manufacturer. In this case the OEM cost is the cost to the machine tool builder.

3. Lower-cost computer-aided part programming systems. These software systems have been designed for the lower-cost stand-alone minicomputers. Some systems, consisting of both hardware and software, are selling for as low as $10,000–$20,000.
4. Computer graphics that allow a design engineer to prepare an engineering drawing utilizing a CRT[2] and then store the design, which may be used later by a part programmer to prepare a numerical control tape.

NUMERICAL CONTROL, COMPUTER-AIDED DESIGN, AND COMPUTER-AIDED MANUFACTURING

Numerical control, probably more than any other technological development, opened the way for computers to enter the realm of manufacturing. Numbers could be inserted directly into the manufacturing process and thereby control machining operations to very close accuracies. Numerical control machines, however, require a lot of numbers, and the computer, which has a large capacity for handling numbers, was brought into play. At first the computer was used as an intermittent aid for handling the more complex calculations. Then an automatic programming system was developed called APT (see Chapter 5). Using this system, it was possible to prepare one set of instructions; pass these instructions through the computer; and come out with a tape ready for input into the NC control system. Little or no calculation was required. APT was expanded to include automatic machining calculations involving pocketing, drilling cycles, and speed and feed determinations based on metal-cutting requirements and tooling characteristics.

Computer graphics, which is the ability to interact with a computer by developing "pictures" on a CRT, became popular with engineers as a visual aid in their design work. This is referred to as Computer-Aided Design, or CAD. (See Fig. 9–1.) Once the part design was in the computer the part programmer could use the part display to move a circle, which represented the cutter, about the part. (See Fig. 9–2.) This is an extension of the part programming capabilities of the computer and illustrates the partnership that can occur between the design and the manufacturing engineer, and may be considered as one form of Computer-Aided Manufacturing, or CAM, also referred to as CIM, which stands for Computer-Integrated Manufacturing. Actually, any direct involvement of the computer in the manufacturing process, including numerical control, may be considered a CAM operation. The development of schedules, manufacturing methods and processes, and overall planning and estimating, via the computer, may all be considered part of CAM.

GROUP TECHNOLOGY

Group technology, from an NC standpoint, involves the grouping of similar parts so that they may be manufactured more efficiently than if they were handled on an individual basis. The concept, therefore, is particularly appli-

[2] CRT stands for Cathode Ray Tube, and, when tied to a computer, it is capable of displaying engineering designs for development and modification. It is discussed in more detail later in this chapter.

Courtesy of International Business Machines Corp.

Fig. 9–1. Designs and graphical illustrations are "drawn" on the face of the cathode ray tube (CRT) shown in front of the operator on the left. As the operator draws the figure it is stored in the computer. This CRT-computer arrangement permits the operator to "call up" various predesigned and stored computations and figures. It also allows him to rotate the figures shown on the CRT in order to permit viewing at different angles.

cable to small and medium size runs encompassing a large variety of parts. If parts could therefore be handled in "batches," because of their similarity, lot sizes could be increased, which would reduce the amount of setup and handling time. If enough similar parts could be gathered, a manufacturing "cell" could be established whereby the parts could be processed on a continuous basis similar to a high production line. The cell would ideally be made up of numerical control machine tools. The cell concept differs from conventional practice wherein machines are grouped according to their operations, such as the lathe department, mill department, etc. With the cell concept a lathe might be grouped with a mill and drill; as an example, handling and transportation of parts from one department to another, as well as changing the setup for each part, would be eliminated.

In order to group parts it is necessary to code the parts. Once parts have been coded, in accordance with their physical characteristics, the code numbers may be entered into a computer. If it were then required to group all parts having certain characteristics so that they could be machined as a "batch," all that would be required would be to call for all parts that fit the code number and obtain a listing from the computer.

Courtesy of Rocketdyne, a Division of North American Rockwell Corp.

Fig. 9–2. Utilizing an electronic light pen, the programmer has the capability of moving the circular image (representing the cutting tool) around the part. A tape for the machine tool is also prepared by the computer almost simultaneously.

There are a number of coding schemes available. One popular scheme[3] utilizes a base of 12 digits. The first four digits classify the part as to its form; that is, whether it is, for example, round or cubic, or whether a round part, for example, requires machining from one or both ends. The second four digits describe the dimensions of the part. The ninth and tenth digits note the accuracy requirements, and the eleventh and twelfth digits describe the material. Other digits which could follow would describe, for example, lot size, time/piece, or machine tools involved.

Once a library of codes has been established a number of CAM functions can be performed by the computer. One function could be to determine the economic justification for certain machines. Another function could be to determine the loading requirements for a specified rate of production. A third, and very significant function, could be to have the computer prepare process sheets automatically. This would be accomplished by having the computer match the coded number of the part being considered with similar code num-

[3] Called MICLASS and offered by Organization for Industrial Research, Inc. located in Waltham, MA.

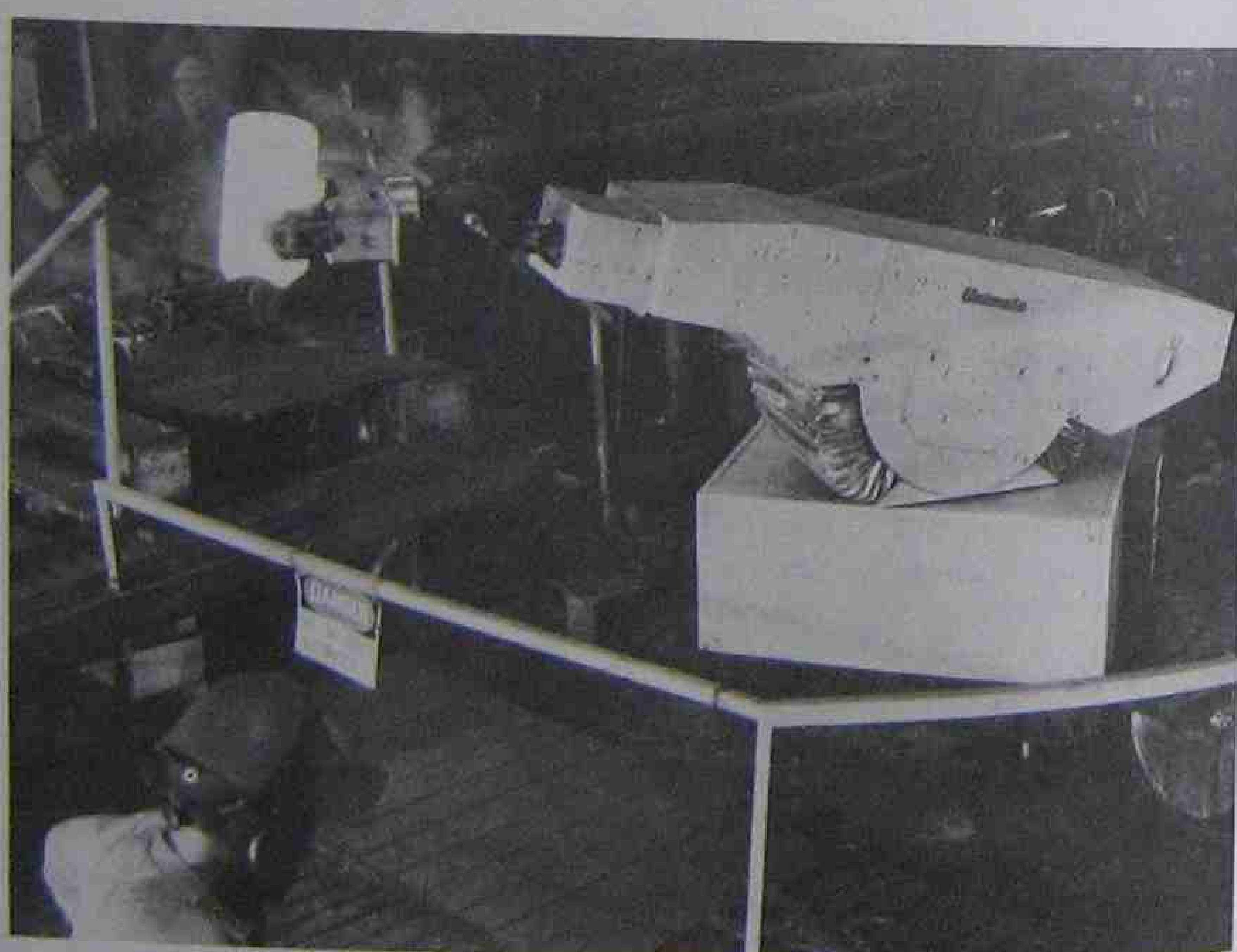
bers in the computer library for which operations and processes have been prepared and put into the computer.

Group technology has been slow in taking hold in the United States. Russia, Japan, and European countries have been involved with coding parts and developing systems for at least the past 20 years. However, the acceptance rate is improving in the United States and a number of large companies and Department of Defense installations have installed turnkey systems.

ROBOTS

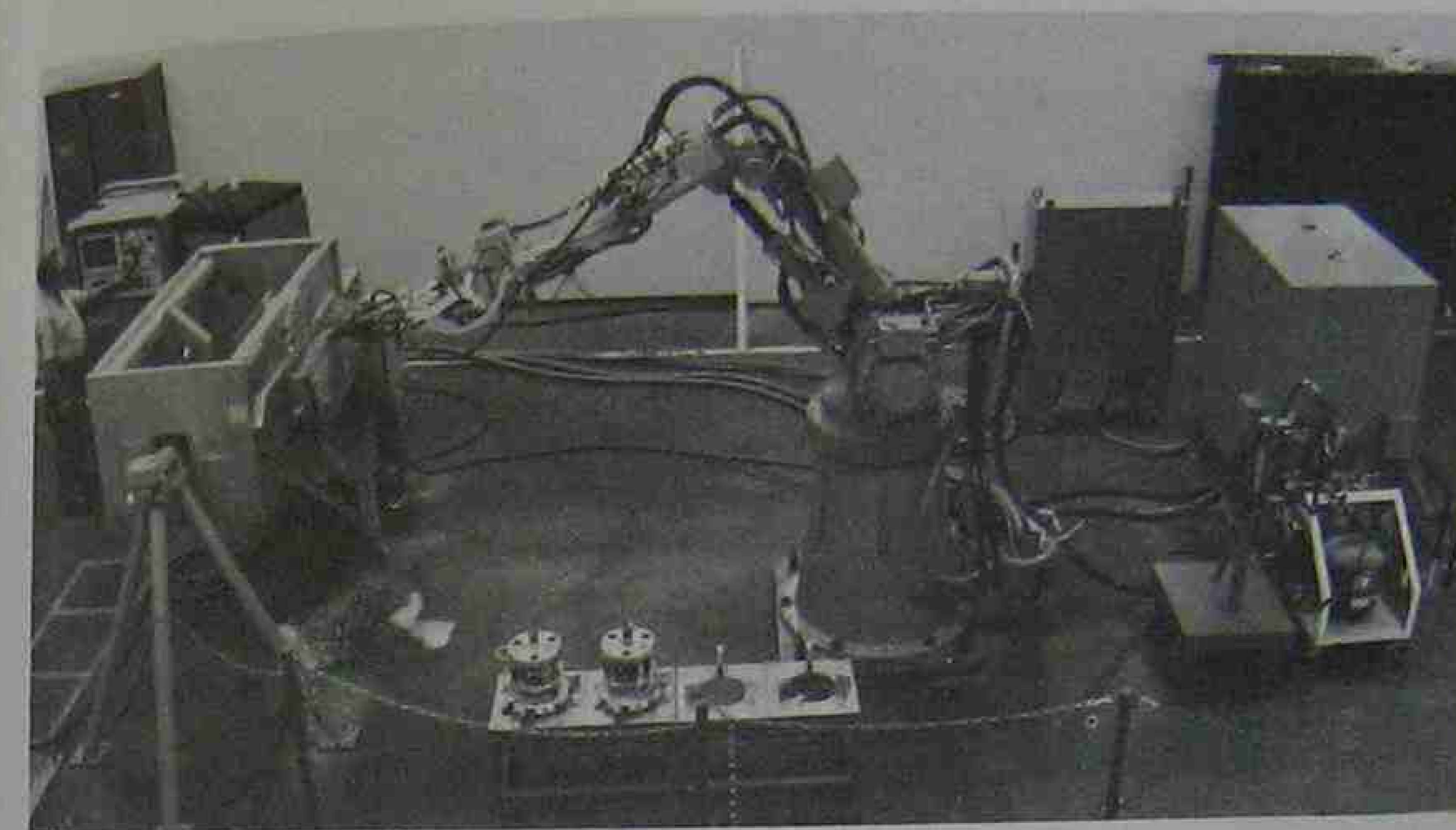
Although not considered a numerically controlled machine in the sense of a machine tool for cutting and forming material, the industrial robot follows the same principles of operation. Its operating components are instructed to move to specific locations, or along a prescribed path, and sensing devices, or servos, feed back the locations to a control system just as with the slides of a numerical control machine tool.

The robot is now enjoying reasonable acceptance and its future looks very bright. It has however been a long time in arriving. Unimation Inc., one of the pioneers, was offering fully automated units approximately 20 years ago and, until a few years ago, most robots were used only in hazardous or extremely uncomfortable operator environments. (See Fig. 9–3.) This is now changing



Courtesy of Unimation, Inc.

Fig. 9–3. Robot handling a white hot billet.



Courtesy of General Dynamics and the U.S. Air Force ICAM Office

Fig. 9–4. Computer-controlled robot performing drilling and routing operations on an aircraft sheet metal part. The fixture is of a rotary type so that a part may be loaded on one side while another part is being machined on the other side.

and robots are being used in areas that heretofore were reserved for humans. As of this writing, Japan is leading the field with over 10,000 units in operation, followed by the United States with approximately 3,000 units, West Germany with 850 units, and Sweden with 600 units.

Robots vary. There is the *pick and place* variety which is normally con-

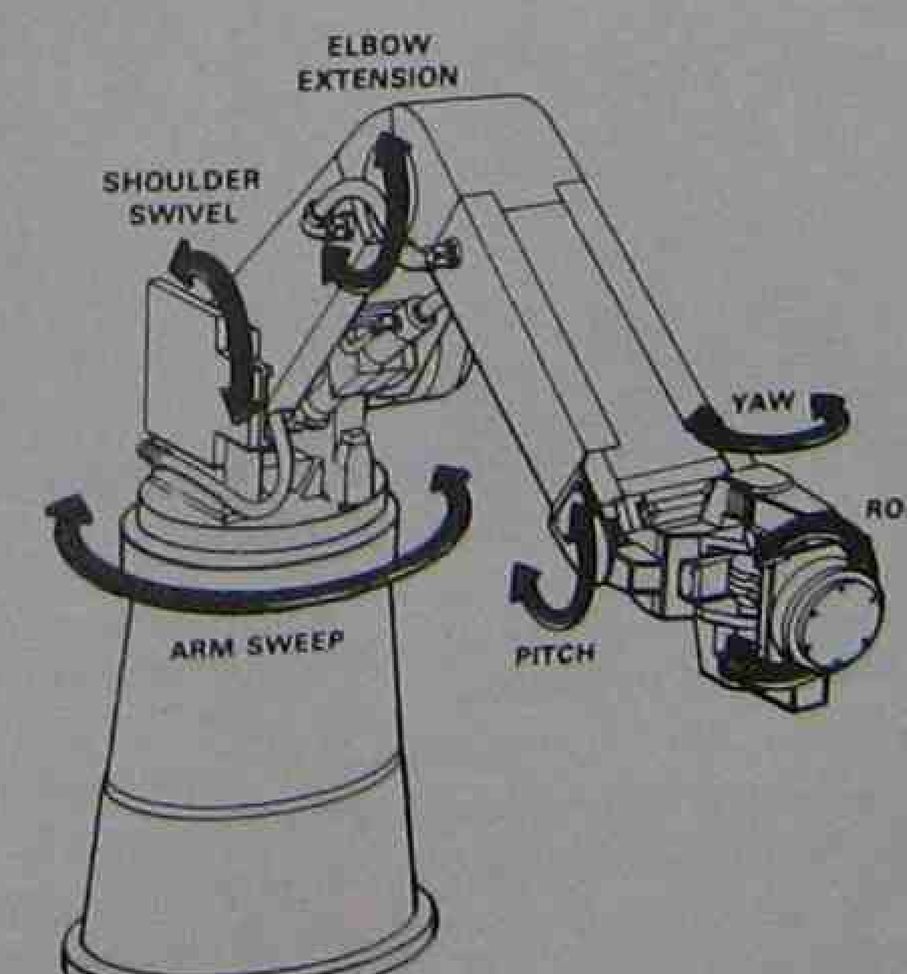
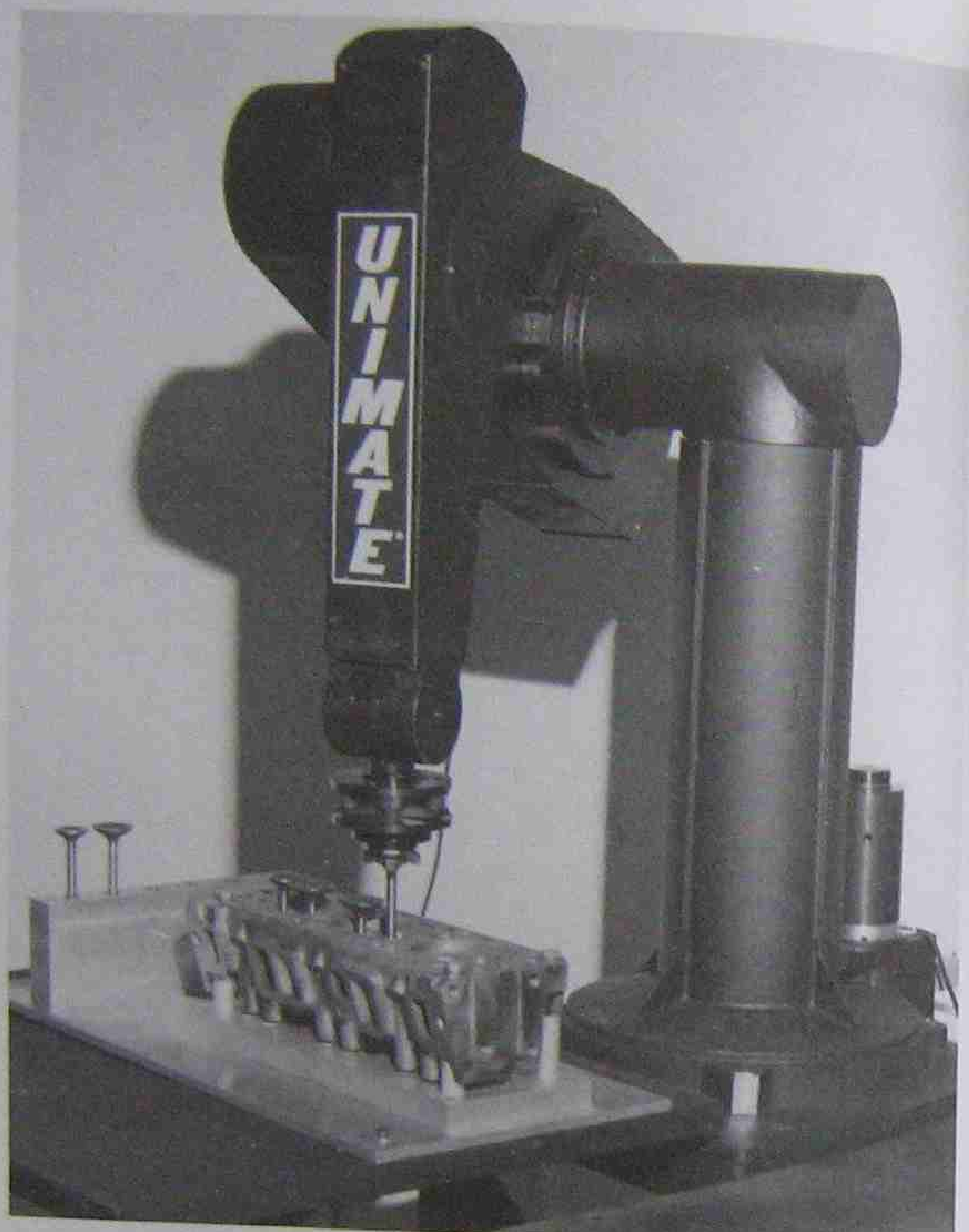


Fig. 9–5. As described, six axes of motion are possible with the robot shown in Fig. 9–4.



Courtesy of Unimation, Inc.

Fig. 9–6. The computer-controlled robot shown in *a* above has been programmed to pick up and insert valve stems into the manifold block shown attached to the fixture. Figures *b* through *d* show the sequence of operations.

trolled by mechanical stops. The number of movements is limited and the operations performed are relatively simple. This type of robot's price tag is also reasonable and in the range of a few thousand dollars. Then there are the *point-to-point* type robots. These are servo controlled and are capable of positioning at varying locations within their range of travel, similar to the ca-

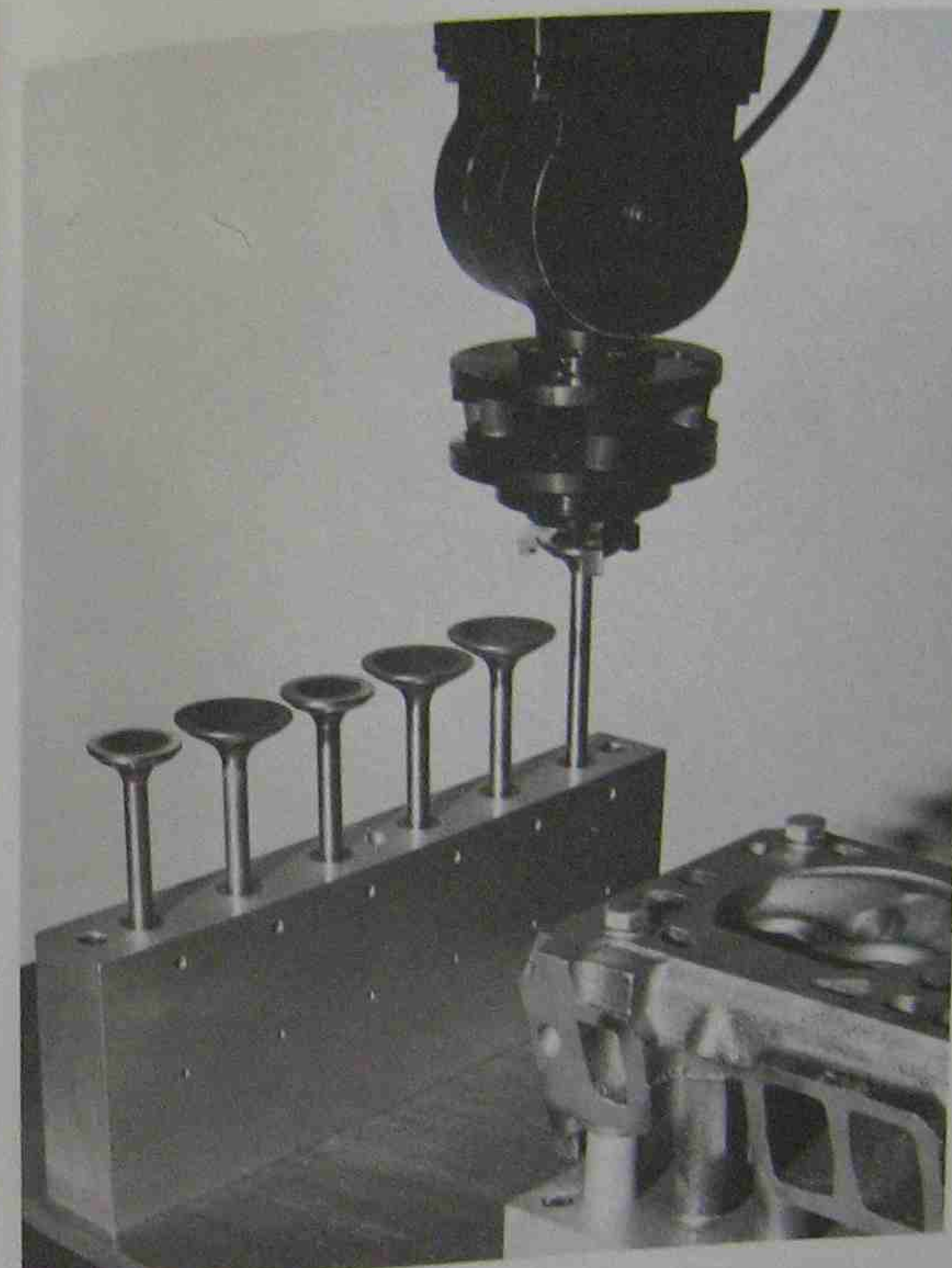


Figure 9–6b

pability of a point-to-point NC machine tool. And, thirdly, there is the *continuous path robot*. These are the type used for such applications as paint spraying and seam welding. In order to handle the data requirements for the movements some form of memory is required, such as a disc or drum or magnetic tape. The robot is "programmed" on the record–playback principle wherein the operator moves the machine to the locations, or along the required path, and

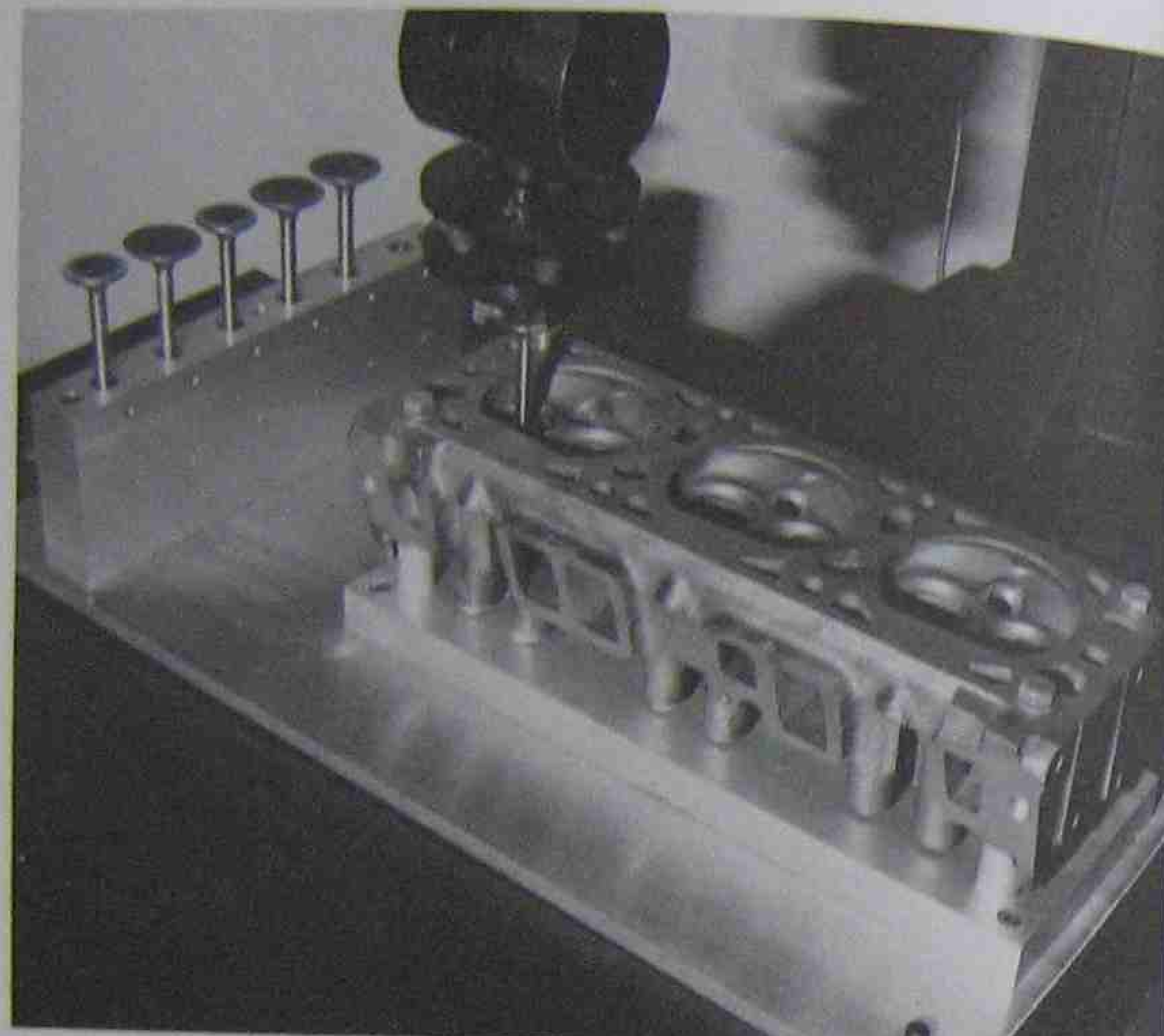


Figure 9–6c

the motions are recorded. Actually, the operator must move each of the motion elements in order to obtain the prescribed effective motion of the "hand." These last two types fall in the $50,000 price category.

As with numerical control, robots may also be controlled by computers. One advantage over the hardwire system is that motions of the hand can be prescribed and the individual motions of the elements are calculated by the computer. Another advantage is in the storage capability of the computer. And probably the most significant advantage is that the robot can be programmed without the operator having to run the machine through its motions. This could result in a significant cost saving since the machine is nonproductive while it is recording motions via the record–playback principle. It would also be expected that languages would be developed, such as APT, wherein verbal instructions could be input to the computer, such as PICK UP PART AND PUT IN SPINDLE.

Robots are also developing visual intelligence via cameras, lasers, and scanning devices. Prototype units have been built that can distinguish one part configuration from another and make the proper selection. A part can be identified and the proper program selected for processing the part. Compensation can be made automatically for parts that are not precisely oriented. The price

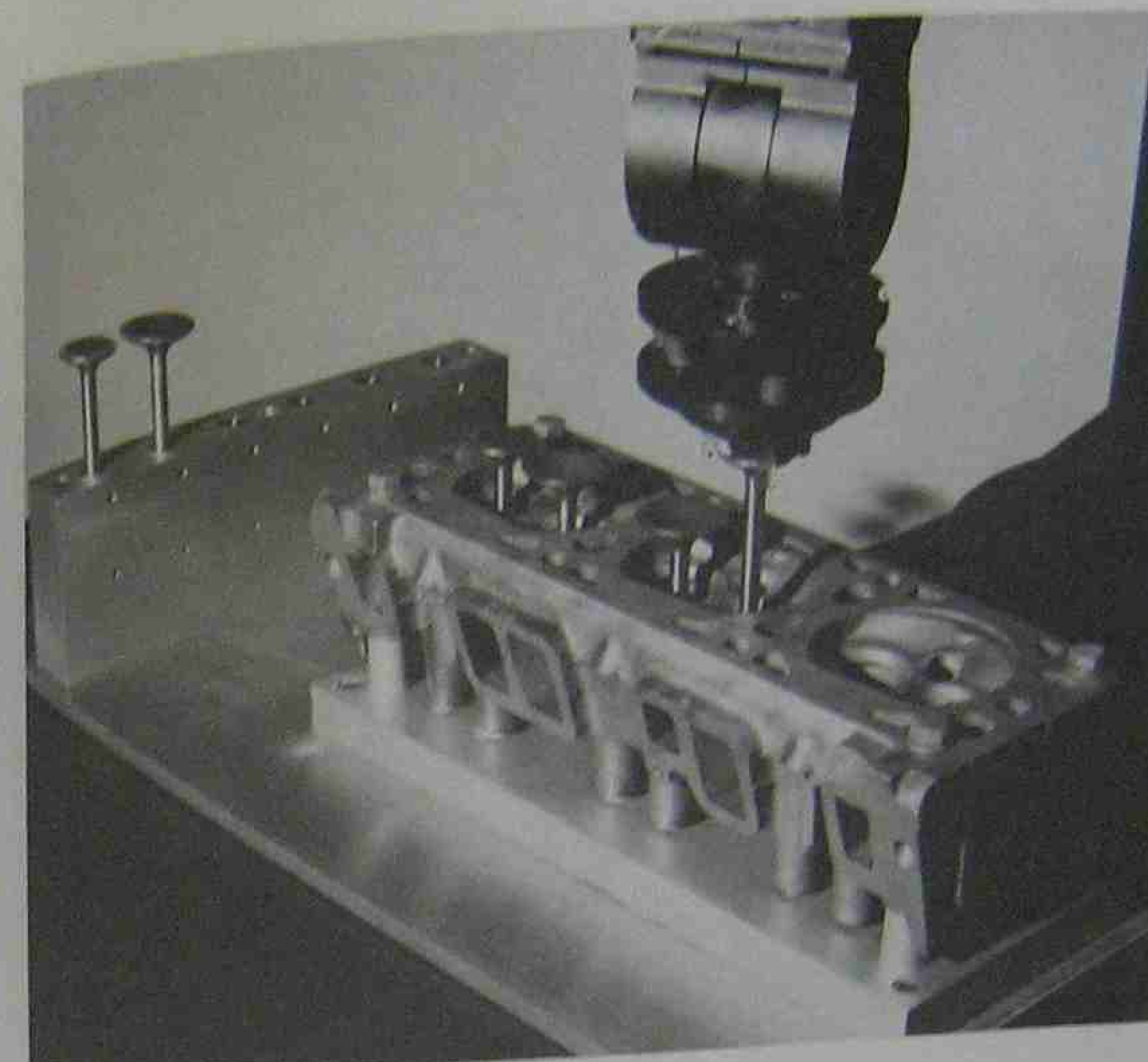


Figure 9–6d

tag for these more intelligent computer-controlled robots can range up from $80,000.

The photograph shown in Fig. 9–4 illustrates the use of a computer-controlled robot for the drilling and routing of an access door of a supersonic jet fighter. The different cutting heads required are shown in the foreground and are automatically selected from programmed instructions. Since the accuracy of the robot (±.050") is something less than required for this particular application, a jig is used to bring the cutting tool to within the desired accuracies. The six axes of motion for this robot are shown in Fig. 9–5. The robot shown in Fig. 9–3 is also capable of six axes of motion.

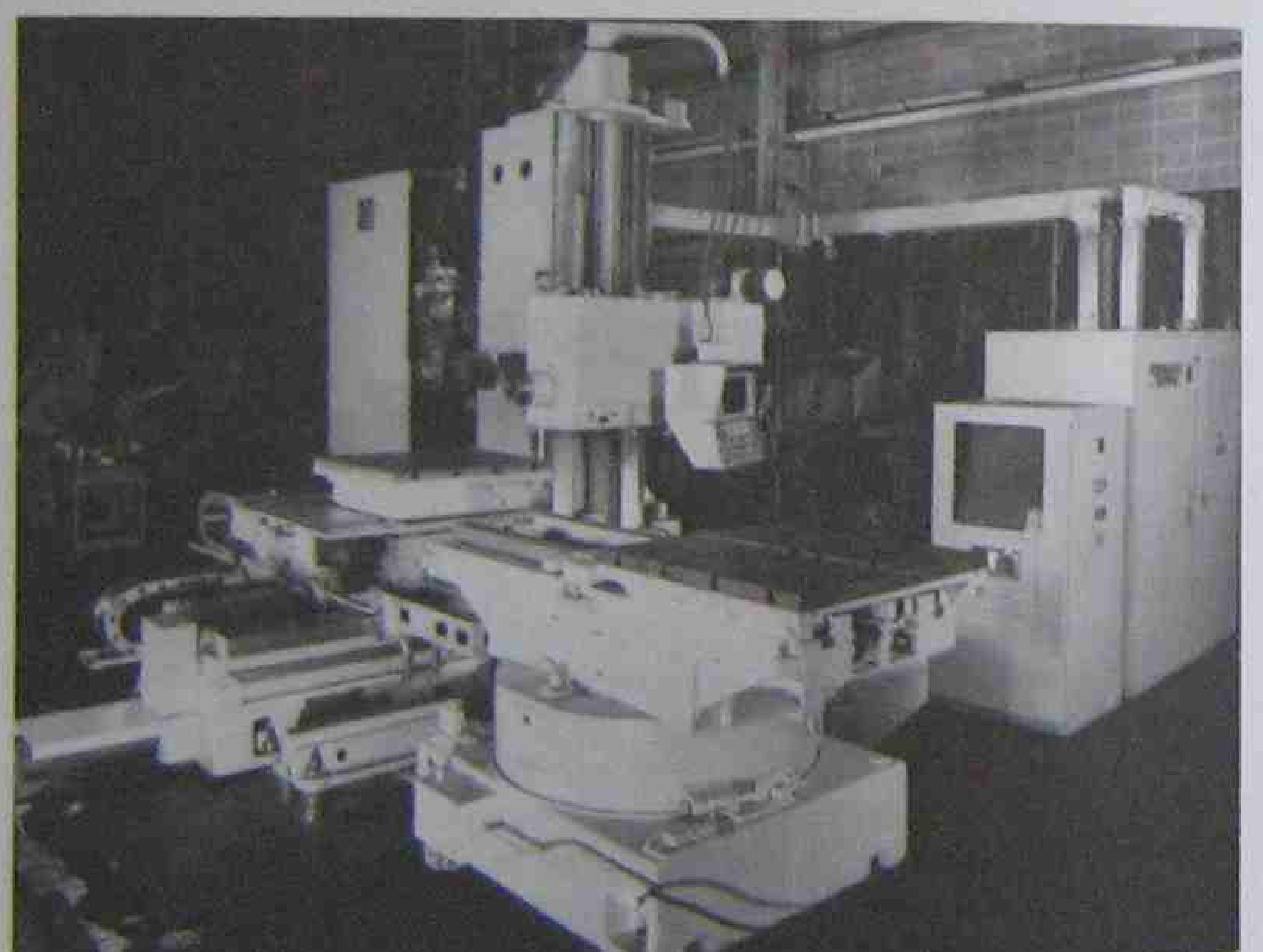
Robots, particularly "intelligent" robots, are expected to play a much heavier role in the CAM evolution. The computer-operated assembly robot shown in Fig. 9–6a, b, c, and d can select the proper valve and place it into a manifold section. Assembly robots have also been "taught" to select, pick up, and screw light bulbs into instrument panels.

AUTOMATIC PART HANDLING

Devices for loading and unloading parts at the machine tool site are expected to become more prevalant. This will be especially true for parts produced in

Courtesy of Warner & Swasey

Fig. 9–7. Multiple spindle lathe with automatic feed and handling device. Claimed to offer 2½ times the productivity of a single spindle NC machine.



Courtesy of Giddings & Lewis, Inc.

Fig. 9–8. While not a new development, automatic pallet changers are becoming increasingly popular, especially with larger machines. Shown above is a pallet changer tied in with a 25-hp spindle machining center.

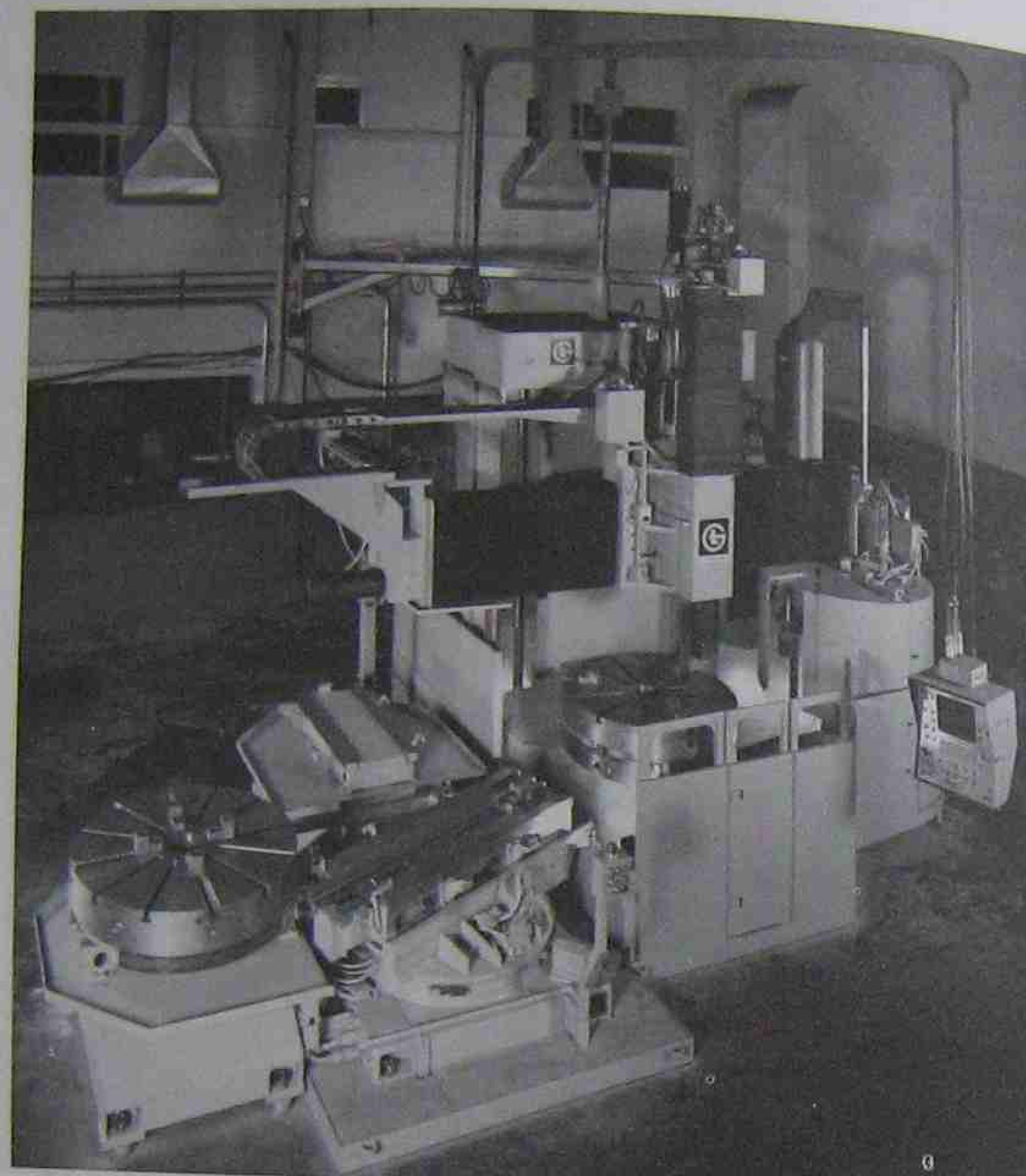
Courtesy of Olofsson Corporation

Fig. 9–9. Large multispindle heads are changed automatically with the machine shown above. Although, in this instance, the parts are not automatically changed as described in the other figures, 9–7 through 9–13, the machine is well suited to medium volume applications as are the other machines shown in Figs. 9–7 through 9–13.

the mid-range of lot sizes (i.e., several hundred to several thousand pieces). Examples of part-handling arrangements, at the machine tool, are shown in Figs. 9–7 through 9–13.

THE AUTOMATIC FACTORY

The term "automatic factory" is often referred to, but seldom in any detail. This is probably because no one has a really clear definition of precisely what a practical "automatic factory" should look like. A true automatic factory would probably be one that receives, inspects, processes, packs, and ships material without human intervention. Furthermore, all paper work, including receiving and packing slips, shipping instructions, and invoices might also be handled without human intervention or assistance. This may be, perhaps, an idealistic and unrealistic goal, at least for the forseeable future. However, since the first numerical control machine was installed and the computer became part of the manufacturing process, we have been moving toward this goal. The chronology shown below traces the significant milestones in numerical control



Courtesy of Giddings & Lewis, Inc.

Fig. 9–10. Vertical turning center with automatic chuck changer.

and CAM from its industrial beginning in 1952, through the present, and into the prophetic future.

The problem with the automatic factory is not so much in the hardware technology but rather in the software development and organization requirements. The machines—the microcomputer control systems, the handling systems, the tooling—are all available. The challange lies in determining the precise requirements, preparing sound economic justifications, preparing the necessary computer software, and assembling and coordinating the whole system.

Numerical Control Chronology

1952	First numerical control machine demonstrated at MIT. Vacuum tube—1st generation (Air Force contract)
1953	Air Force contract to MIT to investigate computer-aided part programming; start of APT programming concept (see Chapter 5)
1955	Air Force procurement of approximately 35 million dollars worth of numerically controlled contour mills
1956 1957 1958	Relatively dry commercial sales period, however, active development period
1959	Introduction of solid-state electronics (2nd generation) and modular printed circuit boards
	Announcement of APT program by MIT & Aerospace Industries Association
1960	Introduction of machining center & automated tool-changing devices
1961	Air Force contract for development of ADAPT (junior version of APT)
1962	Substantial growth in numerical control, particularly point-to-point equipment
	Owing to wide general interest, APT turned over to Illinois Institute of Technology (IITRI) for development and management
1963	Introduction of applications other than metal-cutting tools, such as coil winder, component insertion machine, welding machine, filament winding machine, two-axis inspection machine, wire wrapping machine
1964	General utilization of computer programs as aids to part programming such as CAMP II, SNAP, AUTOMAP, SPLIT, AUTOPROPS, AUTOSPOT
1965	Introduction of "low cost" control systems
	Continuing effort by aerospace and automotive companies in computer-aided design & graphics
1966	Introduction of integrated circuits (3rd generation) resulting in smaller packages and higher reliability
1968	Commerical introduction of Direct Numerical Control
1969	COMPACT II part programming system offered on a commercial time-share basis
1972	Minicomputers began replacing hardwire control systems
1975	Stand-alone minicomputer graphic systems for NC part programming began to appear
	Distributed Numerical Control appearing
1976	Microcomputers for CNC systems began to appear
	Group technology gaining favor in the United States
	Increase of robot popularity

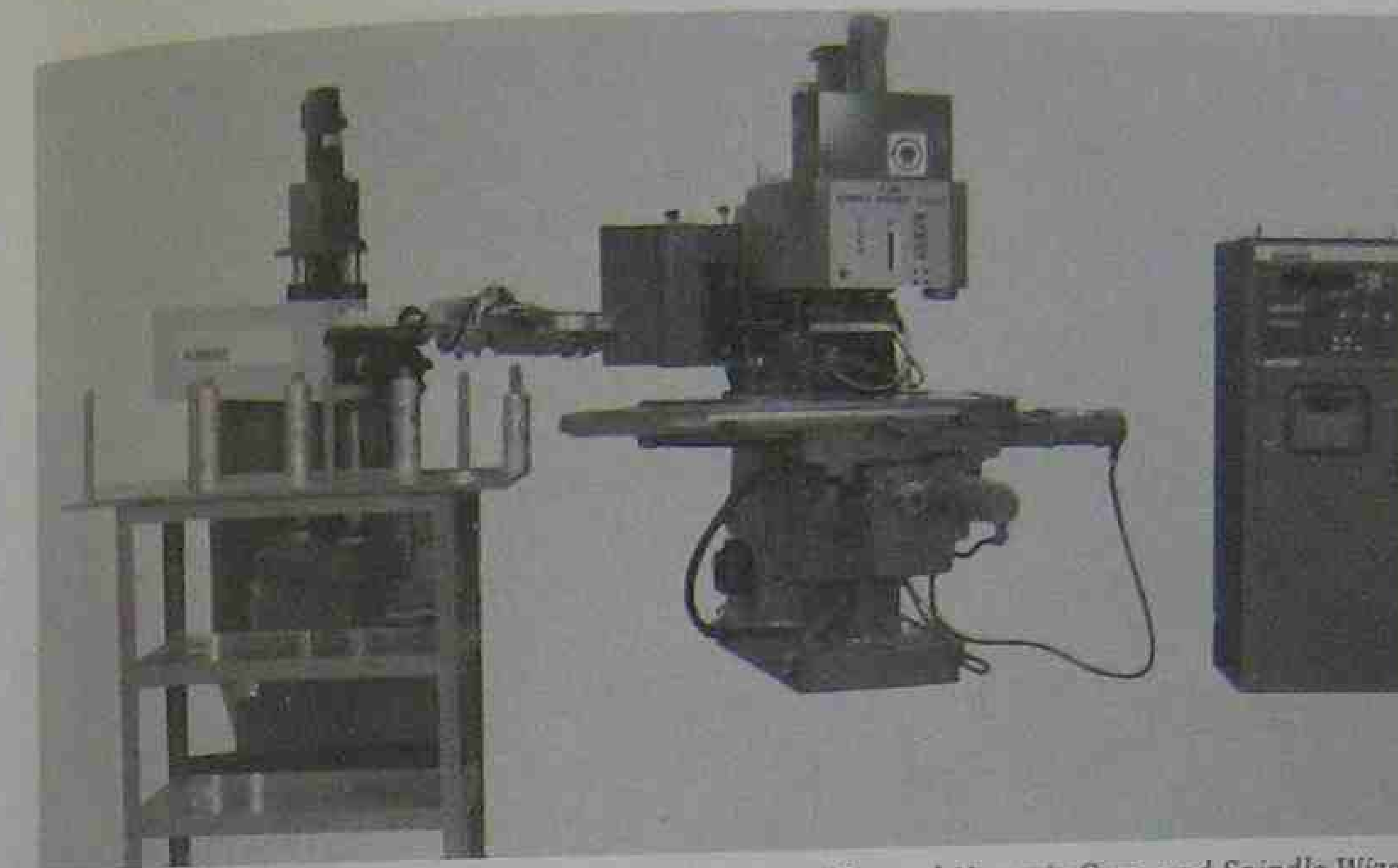
Numerical Control Chronology

1980	Continuing growth of CAD/CAM, especially in the areas of process planning, group technology, and graphic systems
	Still lower cost and more reliable CNC systems
	Much lower cost for stand-alone computer systems
	Closer to the "automatic factory" for medium and large lot size runs of machined parts
1990	Expansion of NC applications

A significant step toward the automatic factory is shown in Fig. 9–14. Parts to be machined are transported from one numerical control machine to the next via carts that are attached to twolines located beneath the floor. Since the

Courtesy of Kearney & Trecker Corporation

Fig. 9–11. Shown is a multiple pallet changer which automatically feeds a variety of parts to a numerical control machining center.



Courtesy of General Numeric Corp. and Spindle Wizard

Fig. 9–12. Raw material is automatically loaded, and finished parts unloaded, onto the table of this relatively small machining center. Instructions for operating the robot are contained on the numerical control tape.

Courtesy of Cincinnati Milacron

Fig. 9–13. The robot shown in the photograph is feeding two numerical control lathes. Raw material enters from the right on a belt conveyor. It is then picked up by the robots "hand" and placed in the chuck of one of the lathes. After the part is machined it is removed by the robot and positioned in the electronic gage shown in the front of the photograph. If the part passes inspection, it is put on a pallet; if it fails, it is put on another, or scrap, pallet. If the parts are becoming out of tolerance, say due to tool wear, tool compensation is automatically entered into the control system.

Courtesy of Kearney and Trecker Corporation

Fig. 9–14. The numerical control machining centers shown form a production line designed for machining a variety of parts simultaneously. Parts are transported from one machine to the next via carts and automatically loaded and unloaded onto the machines.

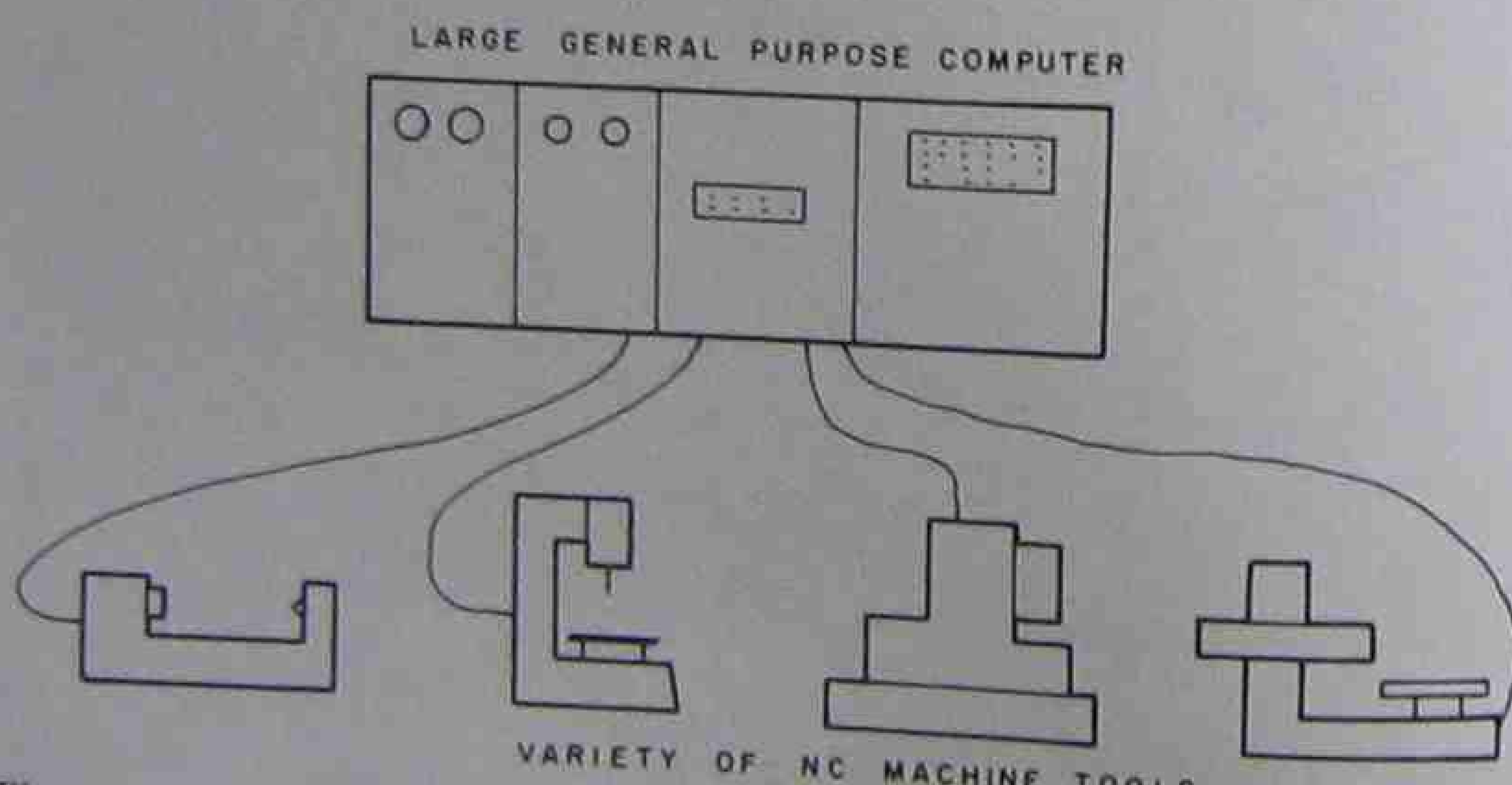


Fig. 9–15. Direct Numerical Control, or DNC, involves the operation of a number of NC machines from a central computer on a time-share basis. Instructions are sent to the machine on a block by block basis as needed. DNC also stands for Distributed Numerical Control. In this case the complete program may be transmitted to the CNC system at the machine, which may be operated independently of the central computer.

entire operation is computer controlled, different parts may be intermixed and the different programs fed to the specified control system as required.

DNC VS DNC

The simultaneous operation of a number of numerical control machines by a single computer is called *Direct Numerical Control*, or DNC. DNC began to appear, commercially, in approximately 1968 and, at the time, was thought to be the wave of the future. However, there were problems, the most significant being the development of the necessary computer software. There was also a matter of having a large manufacturing operation rely on a single, and sometimes not-to-reliable computer. When the computer stopped, either for a software bug or hardware malfunction, everything stopped. This did not sit too well with shop floor superintendents. An illustration of a Direct Numerical Control operation is shown in Fig. 9–15. The control systems are wired directly to the large general purpose computer and can be of the conventional hardwire type in which instructions, in the form of block data, are fed directly behind the tape reader, or they can be specially designed. The computer operates on a time-share basis and can run a number of machines simultaneously.

Another form of DNC began to appear in about 1975, called *Distributed Numerical Control*. In this case the entire program is fed to a CNC system, and, once the program is stored in the memory of the system, the system is disconnected from the main computer. This has the key advantage of not having to rely solely on a single large computer for an entire shop operation. It also reduces the software problem of having to coordinate the time-share operation of a number of machines. A low cost Distributed Numerical Control system is shown in Fig. 9–16.

Whether Direct Numerical Control or Distributed Numerical Control, one

Courtesy of Computer Operations, Inc.

Fig. 9–16. Shown above is a relatively low cost Distributed Numerical Control arrangement, which also has part program editing capabilities and a hard-copy printer. The unit on the right accepts floppy discs, on which part programs are stored and from which the part programs can be directed to any one of eight NC machines.

of the key advantages of this type of arrangement is that shop operations can be transmitted directly to the computer and assimilated. Examples would be an exact count of the number of good, and bad, parts produced or a current record of downtime experience that can be transmitted from the shop floor and reviewed in the shop office on a timely basis.

TECHNOLOGY VS ECONOMICS

Technology, in both the hardware and software areas of numerical control and of all computer-integrated manufacturing, may be expected to advance at an accelerating rate. However, it is not so much the status of technology that is significant but rather the capability to implement the technology. And implementation depends, to a very great extent, on economics. Any capital purchase, whether it be a machine tool, a computer system, or a software program, must be economically justifiable. Technology usually far outweighs the money to pay for implementing it.

QUESTIONS TO CHAPTER 9

1. What are the three main characteristic advantages of CNC over its hardwire predecessors?
2. What is an automatic programming system?
3. What does CAD mean?
4. What does CAM mean?
5. Computer-aided design includes manufacturing processes such as NC part programming and process planning. (True, False.)
6. Group technology refers to groups of people that have gathered to discuss a specific technology. (True, False.)
7. Group technology requires the coding of parts so that the computer can search for like parts and print a listing of these parts as a group. (True, False.)
8. Although being accepted somewhat belatedly, group technology in the United States is far ahead of other countries. (True, False.)
9. What are the three major types of robots?
10. All robots must be run through their motions in order to record the path in the memory of the control system. (True, False.)
11. Highly "intelligent" robots are being designed that can select parts by sensory and visual means. (True, False.)
12. The hardware technologies required for the "automatic factory" are developed. The challenge that lies ahead is, chiefly, to justify the system and assemble the pieces and make them work. (True, False.)
13. DNC may stand for either Direct Numerical Control or Distributed Numerical Control. (True, False.)

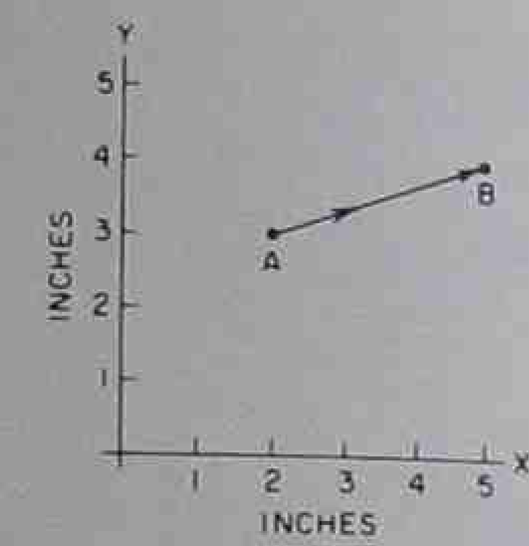
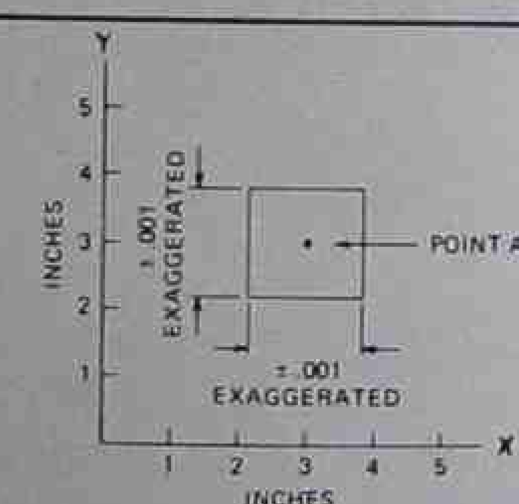
CHAPTER TEN

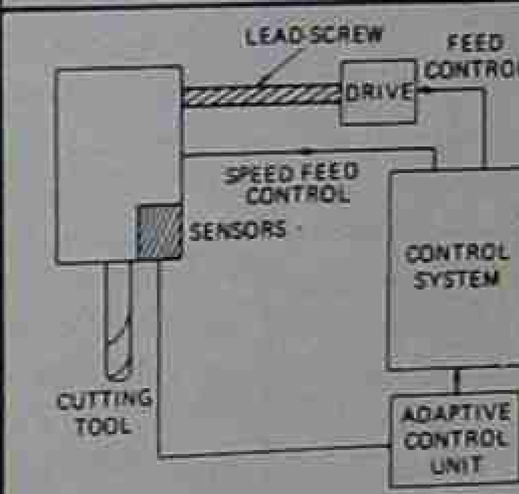
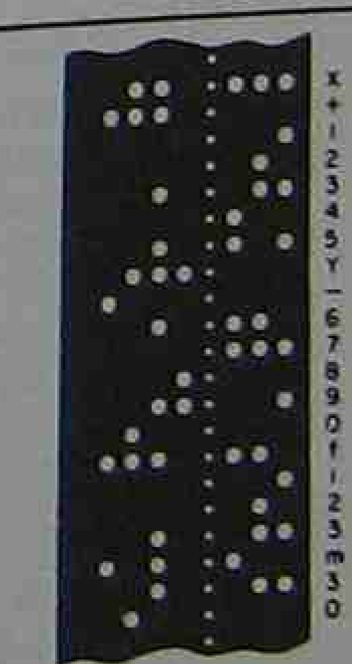
Numerical Control Terms, Definitions, and Standards

As with any new technology, the development of numerical control has resulted in the coining of numerous terms which are unique and most often perplexing to those entering the field. Questions such as the distinction between zero reset and zero offset, or the difference between a block and a bit can understandably create confusion, particularly when it is necessary that close communications be maintained between vendors and customers, and between the various departmental personnel attempting to utilize numerical control within a company.

Many of the terms associated with numerical control may be readily familiar to computer personnel and others to the machine tool people, but the requirement for each group to learn the others' language has created a situation requiring some education. The terms, definitions and examples which follow cover most of the language requirements.

Although the definitions given in this chapter are based in the main upon those standardized by the Electronics Industry Association and others it must be pointed out that for some terms there are no agreed-upon industry-wide definitions. For these the author has had to rely on his own judgment in selecting the ones to be included.

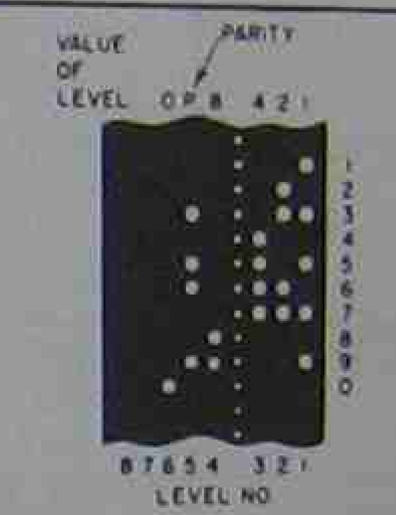
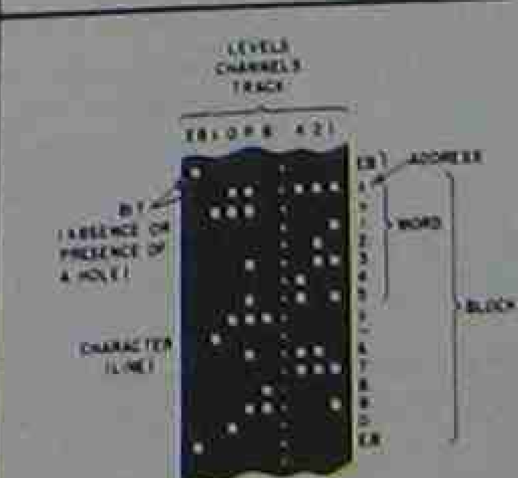
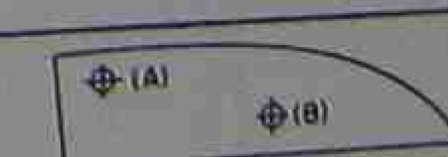
Term	Definition	Example
Absolute Dimensioning	A means of describing movement instructions as a distance from the axes. Word instructions on the tape are noted in absolute form.	In the example shown below, if we want to move from point A to point B, the absolute coordinates of B (distances from the X and Y axes) would be noted on the manuscript. Most PTP systems employ an absolute dimensioning system.  COORDINATES PT X Y A 2.000 3.000 B 5.000 4.000
Accuracy	Usually applies to the machine tool and describes its conformity to a position of indicated value. The accepted understanding applies to the accuracy of an individual axis movement. *Note:* Although accuracy is often measured by a single movement to a designated point, the National Machine Tool Builders Association (NMTBA) has proposed that accuracy "be defined as being the sum of the signed values of the difference between the mean and the target at any point plus the value of the dispersion at that same point which gives the largest absolute sum." For further details it is suggested that the booklet entitled *NMTBA Definition and Evaluation of Accuracy and Repeatability for Numerically Controlled Machine Tools* be pro-	 Assume that the coordinate position of point A, as described on the tape, is X = 3.000 and Y = 3.000. If the machine accuracy is guaranteed = ±.001", then the X-axis movement could fall between 2.999 and 3.001 *and* the Y-axis movement could fall between 2.999 and 3.001, or anywhere within the square shown above.

Term	Definition	Example
	cured directly from the NMTBA at the address noted under the *Numerical Controls Standards* section of this chapter.	
Acoustic Coupler	An electronic device that sends and receives digital data through a standard telephone handset. To transmit data the digital signals are converted to audible tones that are acoustically "coupled" to a telephone handset. To receive data the acoustically coupled audible signals are converted to digital signals. Often used in conjunction with a modem.	
Adaptive Control	A technique for measuring performance of a process and then adjusting the process in order to obtain optimum performance. In the machine tool field, a means of automatically adjusting the feed and speed of a cutting tool, from sensor feedback, in order to realize the optimum cut. Sensors may measure any one or all of the variable factors involved, such as: heat, vibration, torque, or deflection.	
Address	Means of identifying information which is to be stored in a computer. With numerical control, the address usually consists of a letter noted on the tape which requires some action, the extent of which is described by numbers following the letter. In the example shown x, y, f, and m are addresses which describe the x and y motions together with the feedrate required (f), and the auxiliary function (m). (RS-244-B standard is shown.)	
Algorithm	A prescribed set of well-defined rules or processes for the solution of a problem in a finite number of steps. Contrast with *Heuristic* which pertains to exploratory methods of problem-solving in which solutions are discovered by evaluation of the progress made toward the final result.	

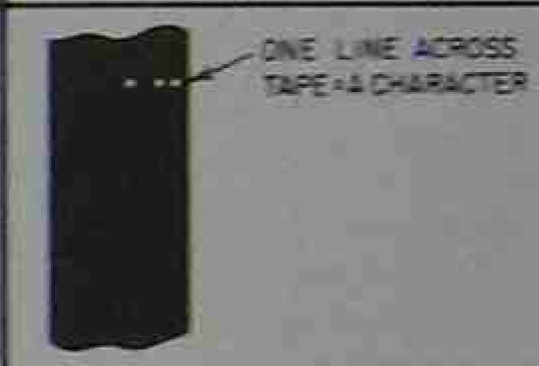
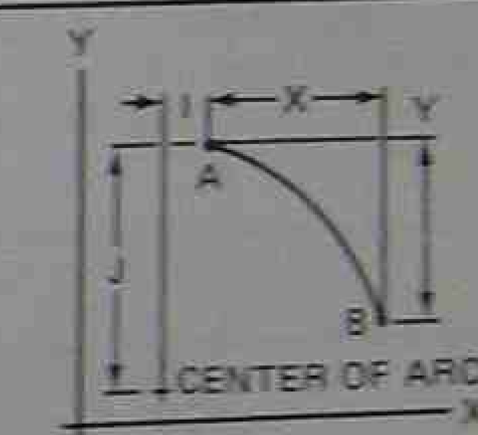
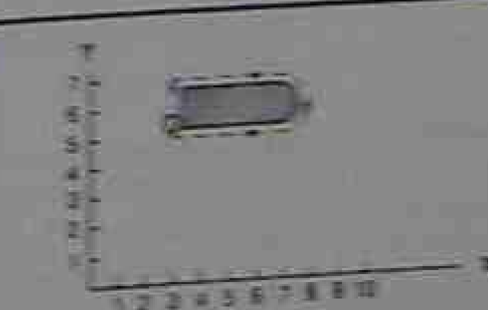
Term	Definition	Example
Alphanumeric	A combination of alphabetic and numeric symbols and associated special characters.	An APT statement would generally be alphanumeric, e.g., P1 = PØINT/INTØF, L12, L14
Analog	Analog data imply continuity of information such as would be developed directly from a rheostat or speedometer. An analog device is generally dependent for operation on a continuous signal which varies in magnitude with the variation in voltage or current as determined by the sensing unit. Although sufficiently precise for certain problems, it may not be accurate enough for others. The degree of accuracy obtainable is dependent upon the sensitivity of the measuring and indicating equipment employed. (Often contrasted with DIGITAL)	
ANSI	Stands for American National Standards Institute. Organization that develops standards and participates in International Standards efforts. Further details are described under the *Numerical Control Standards* section in this chapter.	
APT	Stands for Automatically Programmed Tools. A computer-aided part programming system which consists of: (a) the input language, (b) the APT processor, (c) an APT postprocessor, and (d) a computer of sufficient size to run the APT program. The APT system was initially developed for three-, four-, and five-axis milling machines, but, due to further development, is presently capable of a wide range of applications including point-to-point and turning work.	
ARELEM	Stands for Arithmetic Element and is Section II of the APT processor. It calculates the cutter locations based on the motion commands which were input and canonical forms of the geometry input as converted by Section I of the APT processor.	

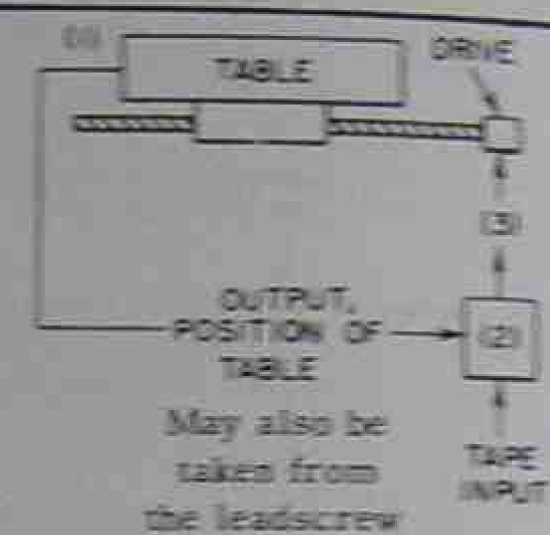
Term	Definition	Example
Assembly Program	A computer program that translates a symbolic language into machine language. The symbol is generally more easily recognizable than a machine language instruction and is, therefore, referred to as a *mnemonic* form.	An assembly language instruction might be: ADD INCR. The assembly computer program translates this into digits, which is the only form the computer understands, i.e., 63 4218. As can be seen, it is far easier for a programmer to recognize and handle "ADD INCR" meaning "add increments" than to recall that "63" means "add," and "4218" means "increments."
Automatic Acceleration and Deceleration	A feature in the control system which enables the machine to accelerate and decelerate smoothly without the necessity for steps. This is accomplished in some systems by denoting a G08 or G09 word; G08 applying to acceleration and G09 to deceleration.	 With automatic acceleration the machine will accelerate somewhat along the smooth curve shown rather than in the step fashion noted. Automatic or smooth acceleration has become the preferred practice.
Automatic Program	A computer program whereby a part programmer, using a relatively simple language, may prepare a manuscript covering the complete part which may then be inserted into a computer and automatically processed through to the final tape by either one or several passes. Contrasts to special programs wherein the computer is used as an assist in generally complicated calculations.	
Auxiliary or Miscellaneous Function	An operation performed by or associated with the machine other than positioning or contouring.	1. Starting or stopping the spindle. 2. Turning the coolant on or off. 3. Positioning a turret. 4. Selecting a tool. 5. Initiating clamping devices.

Term	Definition	Example
Auxiliary Storage	Storage which supplements the main memory storage of a computer, such as disc, tape, or drum. Also refers to the section of a hardwire numerical control system that stores a block of data prior to its movement into active storage and execution by the numerical control machine.	
Axis	A reference line of a coordinate system, e.g., the *X*, *Y*, or *Z* axis of the Cartesian coordinate system. Also, a direction along which a movement of a tool or workpiece occurs.	
Backlash	Movement between interacting mechanical parts resulting from looseness.	 Drive motor, if directly coupled, turns 1 degree with no motion of the table. Backlash would be equal to whatever linear motion should result from a 1-degree turn of the motor.
Band	A unit of signaling speed equal to the number of discrete conditions or signal events per second.	
BASIC	A relatively simple computer programming language; BASIC stands for Beginner's All Purpose Symbolic Instruction Code and was originally developed at Dartmouth College for use on time-sharing computers by people with little or no computer experience.	
Batch Processing	Pertaining to the technique of executing a set of computer programs such that each is completed before the next program of the set is started.	
Binary	A system for describing numbers using only two digits.	Practically all computers operate on a form of binary system wherein a number can be expressed by a combination of "on" and "off" circuits. The concept is

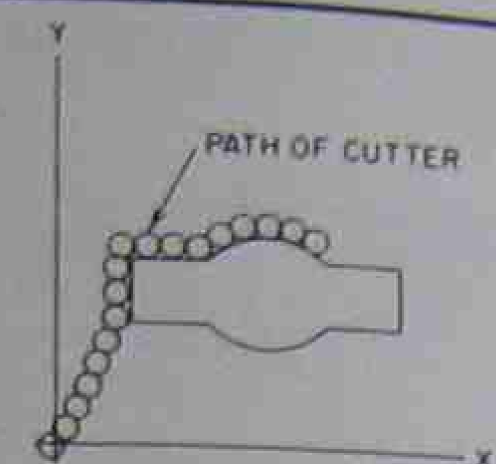
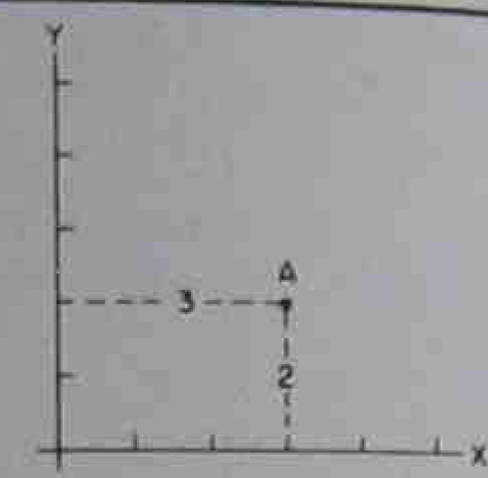
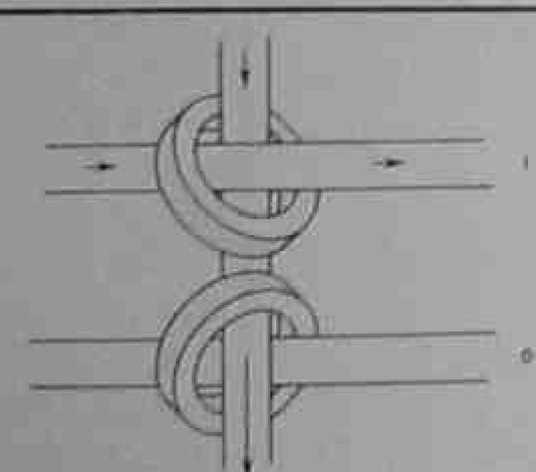
Term	Definition	Example
		particularly well-suited for numerical control since a number may be expressed by either a "hole" or a combination of "holes" and "no holes" on a tape.
Binary Coded Decimal Format (BCD Format)	A system of representing numbers comprised of a combination of four binary bits running across the tape. (See Chapter 3.) Letters are also expressed by a combination of binary bits.	 Considering only the four right-hand rows (levels, tracks) of the tape, numbers are expressed by combining holes across the tape in compliance with the numerical value of the row. Zero is expressed by a hole punched in one of the other four rows.
Bit	One of two possible states (i.e., either on or off). On the numerical control tape it can mean either the absence or presence of a hole.	
Block	A "word" or group of "words" considered as a unit separated from other such units by an "end of block" character (EB). On a punched tape, it consists of one or more characters or rows across the tape that collectively provide sufficient information for a complete cutting operation. (RS-244-B standard is shown.)	
Block Count Readout	Display of the cumulative number of blocks that have been read from the tape. The count is triggered by the "end of block" character (EB). Has generally been replaced by a "sequence number readout."	
Block Delete	This feature provides a means for skipping certain blocks by programming a slash (/) code imme-	

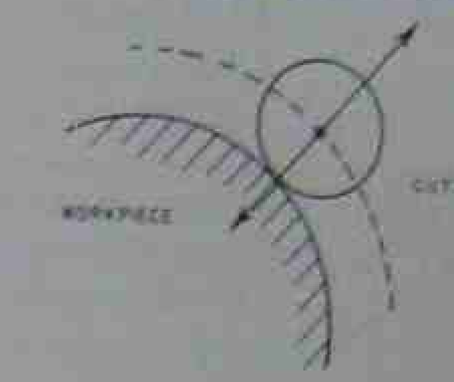
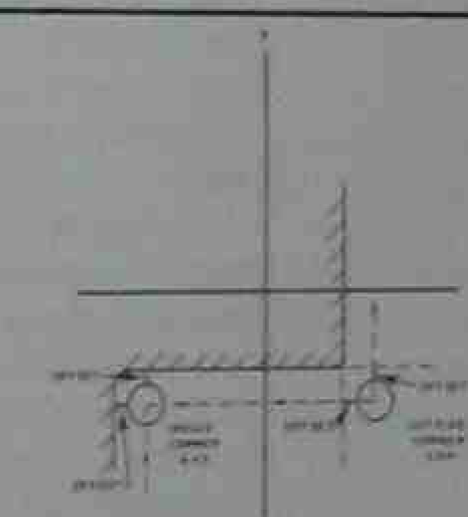
Term	Definition	Example
	diately ahead of the block. The feature is useful when the operator desires to leave off certain cuts on a particular part configuration.	Assume it was desired to cut the perimeter of the above part and then drill holes (A) and (B) on one lot of parts and then on the next lot cut the perimeter and drill only hole (B). The same tape could be used for both lots by pushing the block delete button on the second lot and eliminating the drilling of hole (A) (assuming the (/) character had been incorporated before block (A)).
Buffer Storage	A place for storing information in a control system or computer for anticipated utilization. Information from the buffer storage section of a control system can be transferred almost instantaneously to active storage which is that portion of the control system commanding the operation at the particular time. Buffer storage offers the ability of a control system to act immediately on stored information rather than wait for this information to be read into the machine via the tape reader which is relatively slow.	
Byte	A sequence of adjacent binary digits operated upon as a unit and usually shorter than a word.	A word usually consists of either 16 or 32 bits, or 2 or 4 bytes (i.e. 8 bits per byte).
CAD	Stands for Computer-Aided Design and refers to any system that uses a computer in the creation or modification of a design. Also, usually refers to the use of a computer graphic (CRT) system.	
CAM	Stands for Computer-Aided Manufacturing and refers to the effective utilization of computer technology in the management, control, and operations of the manufacturing facility through either direct or indirect computer interface with the physical and human resources of the company or organization.	
Canned Cycle	A preset sequence of events (hardware or software) initiated by a single command. Normally	G84 would initiate a tapping cycle in which the tap approaches the workpiece at rapid

Term	Definition	Example
	handled by a preparatory, or G, code. Ideal for positioning operations such as hole drilling, tapping, boring, and reaming. Canned cycles decrease programming time and length of tape.	traverse; feeds to a prescribed depth; withdraws from the hole; and then moves back to the starting point at rapid traverse.
Chad	The pieces of material that are removed when punching holes in tape or cards.	
Channel	See Track.	
Character	A number, letter or symbol read on one line across the tape.	
Circular Interpolation	A simplified means of programming circular arcs in one plane which eliminates the necessity for segmenting the arc into calculated straight-line increments. (See Chapter 4.)	 The above circular arc (A-B) may be programmed by denoting essentially the X, Y, I, and J dimensions. (For further details consult Chapter 4.)
CL Information	Stands for Cutter Location and describes the coordinates of the path of the center of the cutter resulting from a basic computer program. This information is common to all machine-tool-system combinations and is the input to the postprocessor. Associated terms are: CL Data—a listing of the coordinates CL File—a listing of the coordinates held in the computer	 In accordance with the above the CL computer print-out would include the following coordinate information:

Term	Definition	Example
	CLPRNT—an APT statement calling for CL data	X Y 3.000 5.000 3.000 7.000 7.000 7.000 7.010 6.995 7.018 6.989 7.024 6.981 ↓ ↓ (7.010–7.024:) Small straight-line increments rounding circular arc.
Closed Loop System	A system in which the output, or some result of the output is measured and fed back for comparison with the input. In a numerical control system, the output would be the position of the table or head; the input would be the tape information which ordinarily differs from the output. This difference would be measured and result in a machine movement to reduce and eliminate this variance.	 (1) Position or motion of the table fed back to (2). (2) Device for comparing tape input information with actual position. (3) Output of (2) actuates drive to move table until actual position agrees with tape input.
CNC	Stands for Computer Numerical Control and is a numerical control system wherein a dedicated, stored-program computer is used to perform some, or all, of the basic numerical control functions in accordance with a control program stored in the computer. Considered to be the fourth generation of numerical control. (See Chapter 2.)	
Code	A system describing the formation of characters on a tape for representing information in a language that can be understood and handled by the control system.	
Command	A signal or group of signals or pulses initiating one step in the execution of a program.	

Term	Definition	Example
Command Readout	See Readout, Command	
COMPACT II	A proprietary part-programming system offered by Manufacturing Data Systems Inc., Ann Arbor, Michigan. (See Chapter 5.)	
Compatibility	A term used to describe the degree of interchangeability of tapes or numerical control language between numerical control machine tools and their respective control systems.	The ideal situation would exist wherein one tape could be utilized with various control systems of different manufacturers as well as different machine tools of the same type. Although a considerable degree of compatibility presently exists with regard to programming language, tape size and, to a great extent, tape format, minor peculiarities between the systems as well as differences in the mechanical features of the machine tools generally prohibit the interchangeability of tapes between various systems and machine tools.
Compiler	Similar in concept to an assembler in that the programmer employs a symbolic language that is more recognizable than a machine language. A compiler goes further than an assembler, however, in that it provides linkages to subroutines, selects the required subroutines from a library of routines, and assembles these parts into an object program. The two most common compilers are FORTRAN, which is applicable to scientific type programming and COBOL which is concerned with business type problems. Numerical control programming, because of its geometric and formular nature, generally utilizes the FORTRAN compiler. The significance of FORTRAN lies in its universal nature, as most computers will accept this language. (See "FORTRAN" in this chapter for further explanation).	

Term	Definition	Example
Contour Control System	A system which continuously controls the path of the machine (e.g., cutting tool, pen or scribe, welding head or torching head) by a coordinated simultaneous motion of two or more axes.	
Coordinates	The positions of points in space with respect to *X*, *Y*, and *Z* axes.	 The coordinates of point A are: x = 3 y = 2 Assuming the point lies on the plane formed by the *X* and *Y* axes; the z coordinate would be zero.
Core	Also referred to as *Magnetic Core*, it is a ring of ferrite material which can be magnetized to represent the binary digits 0 and 1. Core memory of the computer consists of many ferromagnetic cores strung on wire in matrix arrays.	 Each of the cores can be magnetized by passing current through both of the crossed wires. The magnetized core could then represent an "on" condition; a nonmagnetized core would represent an "off" condition. A series of on–off conditions could then "spell out" a binary coded number or notation.
CPU	Stands for Central Processing Unit and is the heart of the computing system. It contains the	

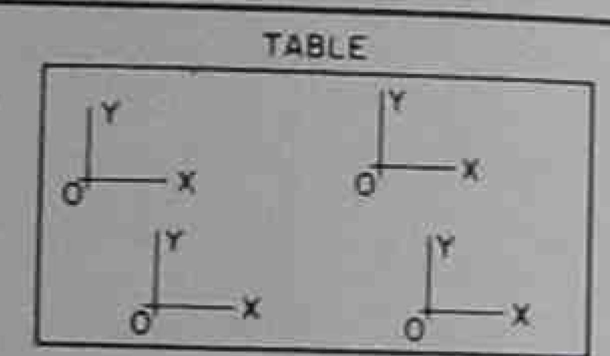
Term	Definition	Example
	memory section arithmetic unit for computation; control circuits to direct operation of the system; and an operator console.	
Cutter Compensation	A means of manually adjusting the cutter center path on a contouring system so as to compensate for the variance in nominal cutter radius and the actual cutter radius. The net effect is to move the path of the center of the cutter closer to or away from the edge of the work-piece. Special considerations must be noted on the tape if it is anticipated that cutter compensation will be used by the operator.	 Cutter compensation is particularly helpful for making adjustments for closer accuracy and for compensating for cutter wear.
Cutter Offset	The distance of the cutter adjustment parallel to an axis.	
Damping	A means of suppressing oscillation in a system which is caused by the system's response to correction signals.	
Data Base	A comprehensive collection of libraries of data for computer processing stored on tape, drum, disc, or other computer-storage media.	
Data Link	Linkage of a Teletypewriter with a remote computer, using public telephone lines. Generally on a time-shared basis wherein a number of Teletype units at different locations share the computer on an effective simultaneous basis.	(See Fig. 6–26, page 167)
Dead Band	The range through which an input can be varied to the servo portion without initiating response at the machine tool. Generally the nar-	

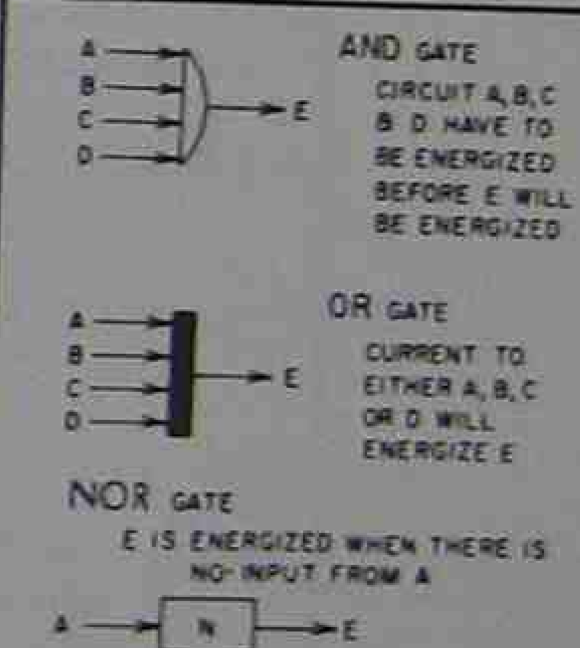
Term	Definition	Example
	lower the dead band the better the response of a machine-tool-system combination. Analogous to mechanical backlash.	
Decode	To translate from coded language into an easily recognizable language. (Opposite of encode.)	
Diagnostic Routine	A specific computer routine designed to locate an error or malfunction in the program. A "print-out" can be made available which will describe the location of the error and prescribe its correction.	
Digital	Refers to discrete states of a signal (i.e., either on or off). Since a combination of on–off signals makes up a specific value, the magnitude of each signal is irrelevant. The theoretical extent of accuracy of a digital system is therefore unlimited and will depend on the amount and cost of electronics involved. Numerical control systems may be of the analog, digital, or combination analog–digital type of construction.	
Digital Computer	A computer that utilizes discrete numbers rather than related variable quantities in processing data.	
Digitize	The process of optical sensing of physical surfaces (lines, points, and curves) to produce Cartesian coordinate approximations of these surfaces.	The inverse of a numerical control drafting/plotting machine. In this case an optical device follows a line and prints out the coordinates, or automatically prepares a tape which can then be run on a numerically controlled milling and/or drilling machine.
DNC	Stands for Direct Numerical Control and refers to the operation of a number of machines directly from a single relatively large computer. Unlike CNC, in which a computer, usually a minicomputer or microcomputer, replaces hardwired electronic components, a DNC system normally	Refer to Chapter 9 for more detail.

Term	Definition	Example
	supplements conventional electronics and automatically regulates the operation of a number of machines. There can be as many as several hundred machines involved in a DNC system. One of the advantages of a DNC system is overall factory control. Another definition for DNC that has evolved is Distributed Numerical Control. In this case tape image data are sent directly from a main computer to a CNC control system. After the complete program has been stored in the CNC system, the link between the main computer and the CNC system is cut and the CNC system operates on its own stored program.	
Double Precision	The utilization of two 16-bit words, in a computer, to obtain a greater range of numerical values.	Most minicomputers operate with 16-bit words. This restricts the numerical range of the computer to ± 32,767, which is normally not suitable for NC calculations. Therefore two 16-bit words may be joined which will then extend the numerical range to ± 1,073,741,823. Computers are coming onto the market with 32-bit word lengths which will satisfy the larger numerical requirements.
Dwell	A time delay that is programmed on the tape.	As an example, a time delay, wherein there is no cutter motion, is useful to allow for the rotation of a turret.
Edit	To modify the form or format of data, e.g., to insert or delete characters.	
Encode	To translate from an easily recognizable language into a coded language.	
Encoder	A feedback device which generates pulses as it rotates. The source of the pulses is often an interrupted light beam. Encoders are becoming popular with CNC systems because of their digital	

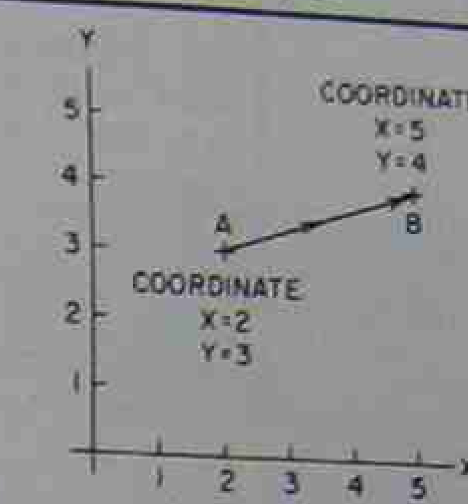
Term	Definition	Example
	quality which can be readily compared with the pulses generated by the CNC system.	
End of Block Character (EB)	A character punched on the tape which denotes the end of a block of data (see definition for Block).	
Executive	Computer software which controls the execution of computer programs and which may provide scheduling, debugging, input/output control, accounting, compilation, storage assignment, data management, and related services.	
External Memory	A storage device such as punched cards, punched tape or magnetic tape which is external to the computer.	
Feedback	That part of a closed-loop system which feeds back information regarding the condition which is being controlled for comparison with the input values. (See definition of Closed-Loop System.)	
Feed Function	The relative travel motion between the cutting tool and the work-piece. The movement is usually expressed in inches per minute (ipm), but may also be expressed as inches per revolution (ipr).	
Feedhold	A method to arrest machine slide travel manually without losing tape data.	
Feedrate Number	A coded number read from the tape which describes the feedrate function. The coding method may differ among systems. The feedrate number is denoted as the "F word" on the tape and usually follows the "dimension words."	One method of denoting the feedrate (now practically obsolete) is to describe a three digit number as follows: $FRN = \frac{\text{Velocity-Inches per minute (ipm)}}{\text{Distance traveled in the block}}$ If, for example, it were required to specify the FRN when the feed velocity is to be 10 ipm over a straight-line distance of 2 inches, then: $FRN = \frac{10}{2} = 500 = \text{F word (three digits)}$

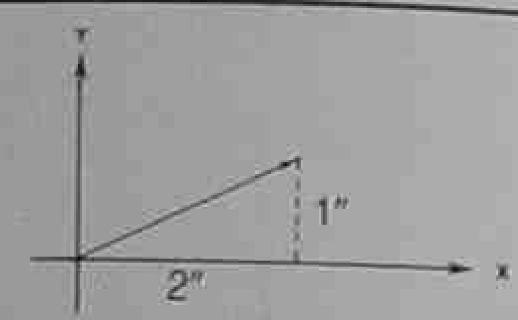
Term	Definition	Example
		The most common way of describing feedrate is directly in inches per minute. A feedrate of, for example, 14.5 inches would be expressed as F14.5.
Feedrate Override	A manual function, usually a rotary dial, which can override the programmed feedrate. The range can be from a few percent to over 100 percent of the programmed feedrate and is usually infinitely variable. (See Manual Feedrate Override.)	If the programmed feedrate as described on the tape is 20 ipm and the operator wanted to lower it to 2 ipm, he would set the dial at 10 percent. There is a justifiable tendency, therefore, for part programmers to specify feedrates "slightly on the high side."
Firmware	Logic circuits in read-only memory that may be altered by software under certain circumstances. Falls between read-only memory in which the program is fixed at the time of manufacture of the circuit and a read/write memory in which a program can be put into memory via software.	
First Generation	In the NC field, the period of technology associated with vacuum tubes, relays, and stepping switches. In the computer field, the technology using vacuum tubes, off-line storage on drum or disc, and programming in machine language.	
Fixed Sequential Format	A means of identifying a word by its location in the block. Words must be presented in a specific order and all possible words preceding the last desired word must be present in the block. An address code is therefore not necessary; however, more characters are generally required per block than with the Word Address format. The Fixed Sequential format system has steadily been losing favor to the preferred Word Address format and may now be considered obsolete.	
Flip–flop	A device or circuit which may have two stable states. The circuit remains in either state until	

Term	Definition	Example
	excited by an input signal which causes it to change to the other state. Generally made up of a combination of NOT and OR gates.	
Floating Point Arithmetic	Arithmetic in which the location of the decimal point for each number in an arithmetic operation is defined as a power of ten and all exponents are equalized prior to the operation.	In adding the floating point decimal numbers $.3 \times 10^5$ and $.27 \times 10^7$, $.3 \times 10^5$ is written as $.003 \times 10^7$ and the two numbers are added which gives the results of $.273 \times 10^7$. Calculations are as follows: $\begin{array}{r} .003 \times 10^7 \\ +\ .270 \times 10^7 \\ \hline .273 \times 10^7 \end{array}$ The number might be expressed in the computer as .273E7. The purpose of floating point arithmetic is to conserve digit spaces in the computer. If floating point arithmetic were not used, the above number would be expressed as 2,730,000.
Floating Zero (also described as Free Floating Zero)	A characteristic of a numerical control system which allows the zero reference point (target point) to be established readily by manual adjustment at any position over the full travel of the machine tool. The control system retains no information on the location of any previously established zero. One advantage of a Floating Zero system is the ability to utilize negative as well as positive coordinate values thus reducing the part programming time required on left and right-hand parts and parts that are symmetrical about center lines.	 The "0" or target point may be established at any position over the entire working surface.
Following Error	The distance lag between the actual position and command position in a contouring machine at any specific time.	
FORTRAN	Stands for FORmula TRANslation and is probably the most popular computer program for scientifically oriented problems. Because of its universal nature, statements in this language will be acceptable to practically all scientific	As will be seen from the example below, the FORTRAN statement is very similar to the conventional arithmetic statement. <u>Algebraic</u> $x = a^2 + bc - 1.5$

Term	Definition	Example
	type computers. The program is machine-independent and the programmer, therefore, need not know the details of how the computer operates. FORTRAN is used in handling the APT system.	<u>FORTRAN</u> x = c**2 + b*c − 1.5 The FORTRAN statement must be *translated* into the specific machine language required by the computer. A special translation routine called a compiler must, therefore, be prepared for each different computer.
Full Duplex	Pertains to a communications circuit that permits two-way simultaneous transmission. Contrasts with half-duplex which permits alternate or one-way at a time communications.	Some part programming time-sharing services operate on half duplex systems while others operate on full-duplex.
Gain	The amount of increase in a signal as it passes through a control system or a control element. The gain in a control system would refer to its sensitivity and its ability to raise the power of a signal to a required output.	
Gate	A circuit having one output and one or a number of inputs. It is designed so that the output is energized when certain input conditions are met. There are three types of circuit gates, viz: AND gates, OR gates, and NOR gates. An AND gate is one whose output is energized only when all of the inputs are energized. An OR gate is one whose output is energized when one or more of the inputs are energized. A NOR gate is one whose output is energized when there is NO input.	
Half Duplex	A communication channel which can receive and transmit, but not simultaneously. (See Full Duplex.)	
Hardware	Physical equipment, e.g., mechanical, electrical, hydraulic, magnetic devices, computer mainframe. Contrasted to software.	One sage definition sometimes heard is that "if you kick it and it hurts your foot, its hardware."

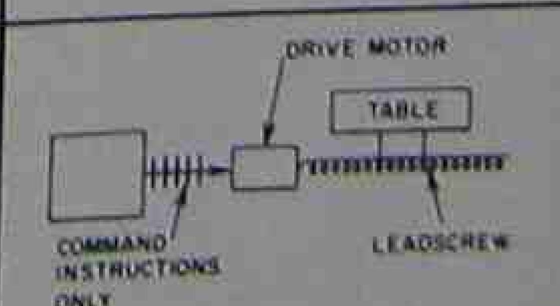
Term	Definition	Example
Hardwired Numerical Control	A numerical control system wherein the response to data input, data handling sequence, and control functions is determined by the fixed and committed circuit interconnections of discrete decision elements and storage devices. Changes in the response, sequence, or functions can be made by changing the interconnections. Contrasted to software numerical control. There are few hardwired numerical control systems still being built.	
Heuristic	Pertaining to exploratory methods of problem solving in which solutions are discovered by evaluation of progress made toward the final result. Contrasted to algorithmic.	
Hexadecimal	Pertaining to a numeration system with a base of sixteen. That is, the hexadecimal system has 16 "digits": 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.	The decimal number 11,983 would be represented in hexadecimal as 2ECF and would be represented in the following way: $16^3 \times 2 + 16^2 \times E + 16^1 \times C + 16^0 \times F =$ $16^3 \times 2 + 16^2 \times 14 + 16^1 \times 12 + 16^0 \times 15$ $8{,}192 + 3{,}584 + 192 + 15 = 11{,}983$ The decimal number 324 can be converted to hexadecimal in the following way: 324 $-256 = 16^2 \times 1$ 68 $-64 = 16^1 \times 4$ 4 $-4 = 16^0 \times 4$ 0 $16^2 \times 1 + 16^1 \times 4 + 16^0 \times 4$ $256 + 64 + 4 = = 324$ Therefore $324_{10} = 144_{16}$
Home Position	A fixed point along an axis referenced with respect to a machine datum.	Typically used for tool change and pallet change.

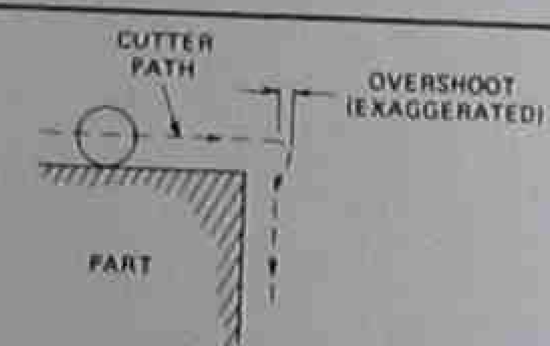
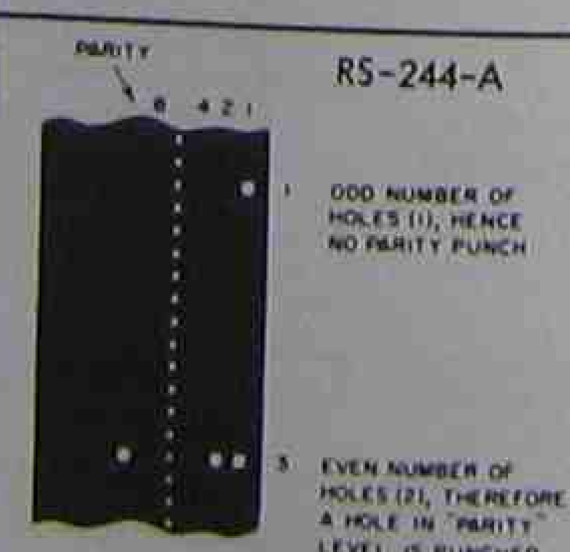
Term	Definition	Example
Incremental Dimension Word	A word defining a dimension or movement with respect to the preceding point in a sequence of points. Most contouring systems utilize incremental dimensioning. (Most Point-to-Point systems utilize Absolute Dimensioning.) Most CNC systems can operate in either an absolute or incremental mode.	 Moving from A to B in the example shown would require programming the incremental X and Y distances between points A and B, e.g., Incremental distance = X movement = 3. Incremental distance = Y movement = 1.
Input	Transfer of external information into the control system.	
Integrated Circuit	A complete functional circuit consisting of transistors, diodes, capacitors, and resistors all constructed within or on the surface of a microlitic chip of silicon. A refinement of its predecessor, solid state, having greater reliability and requiring much less control space due, primarily, to a process resulting in condensed circuit modules.	
Interface	A hardware component or circuit for linking two pieces of electrical equipment having separate functions. A hardware component or circuit for linking the computer to an external IO device.	Tape reader to tape processor, or control system to machine.
IO's	Stands for Input Output devices which are means of entering or extracting information from the CPU.	Magnetic tape units, printers, card readers, cathode ray tube displays, special typewriters.
Leading Zero Suppression	Feature of a control system whereby the zeros preceding the	The number .0033 need be described only as 33. The control

Term	Definition	Example
	first nonzero integer of a number need not be described on the tape. Contrasted to trailing zero suppression.	system will automatically interpret the number as .0033. This assumes that the control system reads to four decimal places.
Lead-Screw Compensation	Automatic compensation for measured errors in each lead screw. Can be accomplished by cam compensation or by an error table stored in the memory of the CNC system. Can improve positioning accuracy.	
Level	See Track.	
Linear Interpolation	The ability to control the motion of one or more axes linearly. The type of interpolation in which straight lines are employed to approximate a required surface, by controlling the motions of one or more machine axes.	 The control system would meter pulses proportionately to the *X* and *Y* axes so that the resultant travel is along a straight line. In the example shown two *X* pulses would be metered for every *Y* pulse.
Loop	A sequence of instructions that are executed repeatedly until a terminal condition prevails.	The "feedback" system of a numerical control machine or a computer instruction that is repeated until some prescribed condition is met.
Machine Language	A computer term referring to exact digital instructions to a computer which it can execute directly, without modification or translation. Most programmers prefer not to program in machine language which requires keeping track of all memory locations as well as being difficult to identify since only numbers are generally used.	A machine language statement might be: 63 4218 where 63 is the code for addition, and 4218 is the address of a memory location.
Macro	A source language instruction from which many machine language instructions can be generated. With numerical control part programming it means that a	An APT Macro could be as follows: M1 = MACRØ GØ/TØ,L1 GØRGT/L1,TØ,L2

Term	Definition	Example
	complete operation can be varied by the substitution of a few computer part programming words. This saves the part programmer the effort of having to write all the part programming statements.	GØRGT/L2,TØ,L3 GØTØ/X,Y,Z TERMAC At any point in the program the Macro could be called and values assigned to the variables *X*, *Y*, and *Z*, e.g., CALL/M1,X=1,Y=2,Z=5 All the operations would be repeated with the *X*, *Y*, and *Z* values assigned.
Manual Data Input	A means of inserting complete format data manually into the control system. These data are identical to information that could be inserted by means of a tape (including all auxiliary functions, feedrate number, etc.).	
Manual Feedrate Override	Enables the operator to reduce or increase the feedrate should the programmed rate noted on the tape be excessive or too slow for the particular piece of material being machined. This feature generally consists of a dial on the operator's console which will enable the operator to adjust or "override" the programmed feedrate. The percentage of override may vary from approximately 1 to approximately 150 percent of the programmed feedrate. The range is usually infinitely variable between these values. (Usually pertains to contouring systems.)	
Manuscript	A form used by the part programmer for listing the detailed instructions which can be transcribed directly by the tape preparation device or fed to a computer for further calculation. (See Chapters 4 and 5.)	
Memory	Synonymous with *Storage* and pertains to a computer device into which data can be entered, held, and/or retrieved. Usually refers to the *internal* capacity of the computer and is handled by a magnetic *core* or semiconductor	A computer that can handle approximately 32,000 words in internal storage is said to have a memory of 32K. The words may be of different bit lengths, two common ones being 16 and 32.

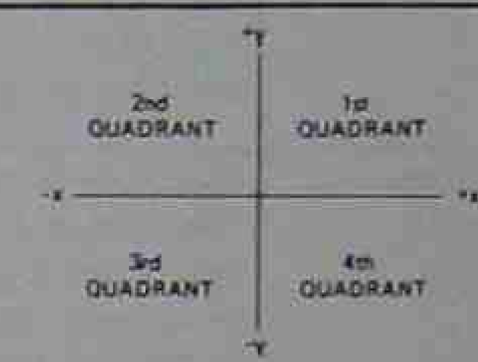
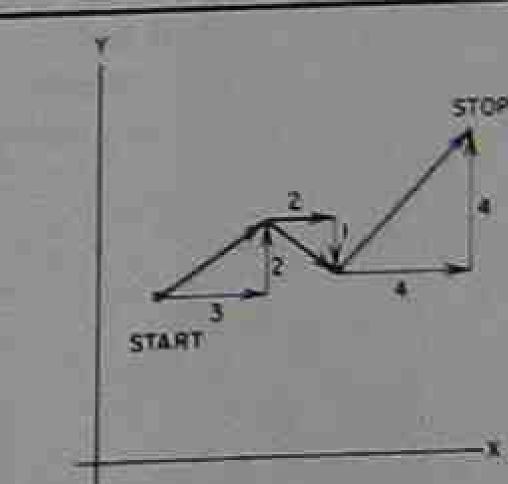
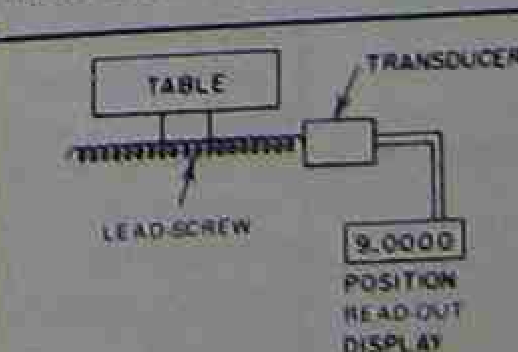
Term	Definition	Example
	arrangement in contrast to *external* storage which refers to such mediums as magnetic tape, discs, or drum.	
Microcomputer	A class of computer having all major central processing functions contained on a single printed circuit board constituting a stand-alone module. Microcomputers are typically implemented by a small number of LSI circuits and are characterized by a word size not exceeding 16 bits, and very low cost, usually under $1,000. A microcomputer is constructed using a microprocessor as the basic element.	
Microprocessor	A basic element of a Central Processing Unit that is a single integrated circuit. A micropressor requires additional circuits to become a suitable central processing unit. It is the basic element of a microcomputer. (See Microcomputer.)	
Mnemonic Code	Instructions for a computer written in a form which is easy for the programmer to remember but which must later be converted into machine language.	
Modem	Stands for modulator–demodulator and is an electronic device that sends and receives digital data using telecommunication lines. To transmit data, the digital signals are used to vary (modulate) an electronic signal that is coupled into the telecommunication lines. To receive data, the electronic signals are converted (demodulated) to digital data.	
Nanosecond	One-billionth of a second.	
Noise	Random electrical impulses similar to radio static, occurring within the control system that can disturb its normal operation.	

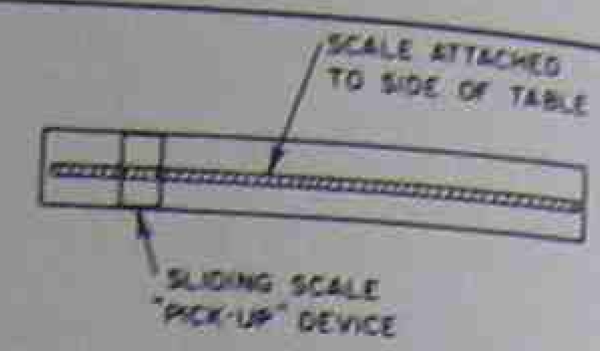
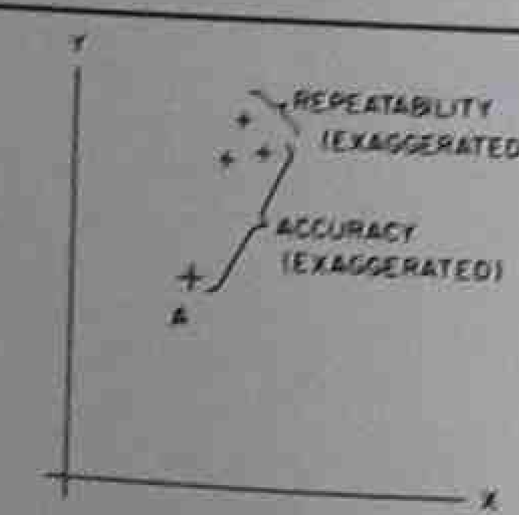
Term	Definition	Example
Numerical Control System	A system in which actions are controlled by the direct insertion of numerical data at some point. The system must automatically interpret at least some portion of these data.	
Numerical Data	Data in which information is expressed by a set of numbers or symbols that can only assume discrete values or configurations.	
Octal Numbers	As with the binary system which operates to a base 2, this system operates to a base 8. Its advantage is that fewer digits are required than for the binary code. It is also more readily convertible to the decimal system than is the binary system and is used as an intermediate step, within the computer, to convert from decimal to binary and vice versa. This is less cumbersome than going directly from decimal to binary and vice versa.	A decimal number such as 25, may be expressed in *Octal* form by dividing the number by 8; noting the number of times it goes into it; and then noting the remainder, e.g., $\frac{25}{8}$ = 3 plus a remainder of 1. The octal number is 31. In binary code, 25 would be expressed as: 11001. It will be seen that converting octal 31 to decimal 25 can be accomplished readily by inspection (i.e., $3 \times 8 + 1 = 25$). It would be somewhat more difficult to attempt converting 11001 and, certainly, far more difficult with larger numbers expressed in binary.
Off-Line Operation	Usually applies to peripheral equipment which operates independently of the central computer. This equipment is generally utilized where it is not necessary to operate the full capabilities of the major portion of the computer, thus saving time and expense.	Key punch device, card sorter, printer, tape preparation devices
On-Line Operation	Applies to computer operations and calculations which are performed by the computer itself or the major portion of the computer.	
Open-Loop System	A control system that has no means for comparing the output with the input for control purposes (i.e., there is no feedback).	

Term	Definition	Example
Operating System	A program or related series of computer programs which govern the operation of a computer and associated peripheral devices.	
Optional Stop	A miscellaneous function command similar to a program stop except that in this instance the control ignores the command unless the operator has previously pushed a button to recognize the command (generally denoted as an M01 code in accordance with standards).	
Overlay	A technique for bringing computer routines into high-speed storage from some other form of storage during processing, so that several routines will occupy the same storage locations at different times. Overlay is used when the total storage requirements for instructions exceed the available main storage.	The APT system which requires a large amount of storage has been developed for small computers by the use of overlay techniques.
Overshoot	The amount of overtravel beyond the command position. The amount of overshoot is related to factors such as: system gain, servo response, mechanical clearances; and inertia factors relating to mass, feedrate, and strain.	
Parabolic Interpolation	The simultaneous and coordinated control of two axes of motion such that the resulting cutter path describes a segment of a parabola. A convenience feature, similar to circular interpolation.	
Parity Check	Considering the numerical control standard RS-244-B a hole punched in one of the track columns (channels) whenever the total number of holes representing a character would be even. The net result would be an odd number of holes in the character. This is utilized as a check when reading the tape since an even number of holes would indicate a source of error in the punching. The control system recognizes	

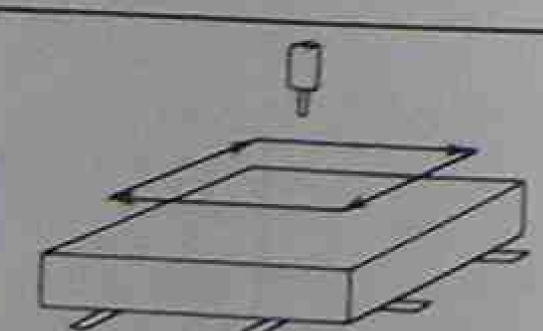
Term	Definition	Example
	only an odd number of holes in the character and will automatically stop if an even number of holes in a character is read. The opposite applies with the RS-358-B standard code in that an even parity is sought and the control system will automatically stop when an odd number of holes is read across the tape.	The number "3" is denoted as a hole in the first channel having a value of 1, and a hole in the second channel having a value of 2. A third hole is required and is punched in the parity column. All three holes are punched simultaneously in the character. The number "7," for example, would not require a parity punch since there are an odd number of holes (values 4, 2, 1) which comprise the character.
Point-to-Point Control System (PTP)	A system in which controlled motion is required only to reach a given end point with no path control during the transition from one end point to the next. Since contouring control systems can also perform point to point operations, and since the cost of contouring systems has been greatly reduced due to CNC, practically all systems now being offered are of the contouring type.	
Position Transducer (Resolver)	A device usually geared to a precision ball–nut lead screw for measuring position and converting this measurement to an electrical signal for transmission back to the control cabinet where a comparison is made with the input instruction. This cylindrical device measures approximately 1½ inches long by 1 inch in diameter.	
Postprocessor	A set of computer instructions which transforms tool center-line data (CL) into machine commands using the proper tape codes and format required by the specific machine/control system.	The APT instruction CØØLNT/ØN would be translated to the auxiliary word M07 by the postprocessor. Also, other codes peculiar to the machine tool/system combination such as the feedrate number (Fxxx), automatic tape rewind (M30), and tool change (Txx) are handled by the postprocessor. In instances where the control operates on incremental dimension words, the postprocessor makes the conversion from the CL coordinate form.

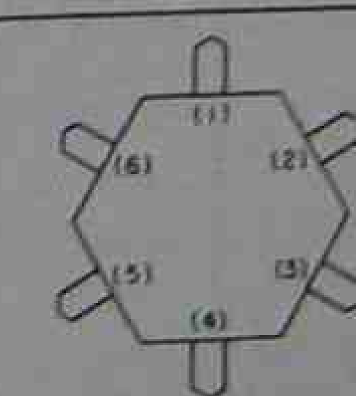
Term	Definition	Example
Precision (as a computer term)	The degree of exactness with which a quantity is stated. In a computer it represents the number of storage positions allowed for describing a number. In single precision the largest number that can be represented is dependent on the size in a single word. If the word is not large enough, then double precision is required. Double precision could therefore carry a greater number of decimal places.	A 16-bit word consists of 2 bits for an address and 14 bits for a numerical value and has the highest value when all of the 14 bits are ones, i.e., $2^{16-2} = 32{,}767$ is the highest value (decimal). If greater accuracy is needed, then *double precision*, or two words, is required.
Precision (as a machine tool term)	See Repeatability	
Preparatory Function	A command on the tape changing the mode of operation of the control which is generally noted at the beginning of a block and consists of the letter character "G," plus a two-digit number. (See Chapter 3.)	Some preparatory functions are: G01 Linear interpolation G02 Circular interpolation G05 Hold G08 Acceleration G09 Deceleration
Programmer (Part Programmer)	A person who prepares the planned sequence of events for the operation of a numerically controlled machine tool. The programmer's principal tool is the manuscript on which the instructions are recorded. There are two types of part programming: manual and computer assisted. With *manual* part programming the part programmer writes the instructions (blocks) that would go directly to the control system. With computer-assisted part programming the part programmer writes instructions that are fed to the computer and then converted to block instructions required for the control system. A person who prepares computer programs which are to be used internally in the computer in order to solve the specific problems presented by the part programmer. A numerical control computer programmer generally requires a formal advanced mathematical background in ad-	Manual part programming instruction: N004G01X00103Y002M30 Computer-assist part programming instruction: GØFWD/LIN3

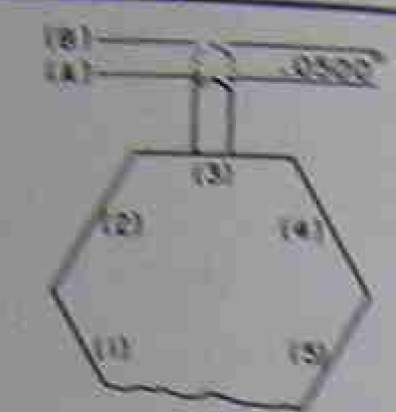
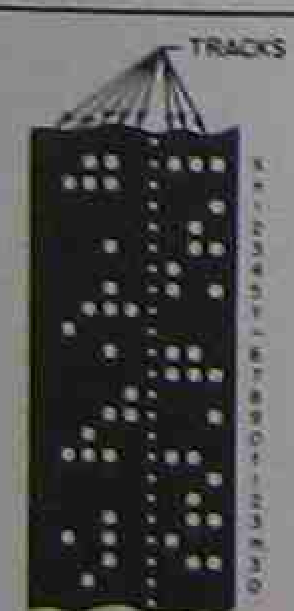
Term	Definition	Example
	dition to a thorough knowledge of computer operations.	
Program Stop (sometimes referred to as Planned Stop)	A miscellaneous function to stop the coolant, spindle, and feed after completion of other commands in the block (denoted as an M00 code). The operator may restart the cutting cycle without loss of accuracy.	
Quadrant	Defines the fourth part of a circle or a fourth part of a graph in reference to the *X* and *Y* axes.	
ROM	Stands for Read Only Memory. A memory unit that is set when manufactured. The memory cannot be altered and can only be "read" as opposed to a read/write memory in which instructions can be programmed into the unit.	Hand calculators employ Read Only Memory (ROM) units.
Read-Out, Command (Display)	Generally applies to contouring systems and to the display of absolute position as derived from the position commands within the control system. In effect, the display is an accumulative read-out of the incremental movements which have been fed into the control system. Assuming that the machine tool is operating in accordance with the control system, it can be reasonably assumed that the read-out represents the actual position of the table.	 At the STOP point the read-out display would appear as follows: X 9.0000 Y 5.0000 Thus, the incremental X and Y movements as noted on the tape have been added, algebraically.
Read-Out, Position (sometimes referred to as Absolute Read-Out)	Display of absolute position as derived from a position feedback device such as a transducer. The transducer is normally attached to the lead-screw. Although theoretically more reliable than the command read-out, it is generally	

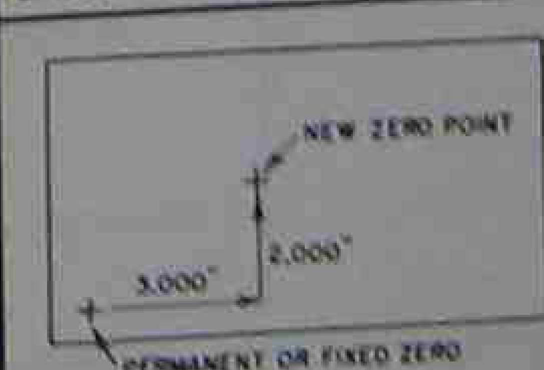
Term	Definition	Example
	more expensive since a separate feedback circuit is required.	
Read-Out, Slide Location	Display of absolute position as derived from a sliding scale measurement device such as an Inductosyn which is attached directly to the sliding moving part, i.e., the table. Theoretically it is the most reliable of the three readout devices (command, position, slide location). Also it is the most expensive of the three.	
Real Time Operation	A computer that operates at a rate compatible with the operation of the physical equipment or processes. The computations are performed while the operation is taking place and may be used to guide the operation.	A part programming system that analyzes each statement and offers a diagnostic note if incorrect.
Remote Job Entry (Remote Access)	Pertains to communication with a data processing facility by one or more stations that are distant from that facility.	A time-share part programming service connected via telephone lines.
Repeatability	Closeness of agreement of repeated position movements to the same indicated location and under the same conditions.	 Assuming A is the point programmed: The accuracy is the ability of the machine-system combination to position to A. The repeatability is the error between different repeated attempts to hit point A. It may also be defined as the consistency of inaccuracy.
Reproducibility	The ability to be precise or repeatable over a relatively long period of time.	
Resolution	The smallest increment of distance that will be developed by the numerical control system in order to actuate a machine tool.	A system having a resolution of .0001" should theoretically move a machine tool .0001" for each .0001" pulse.

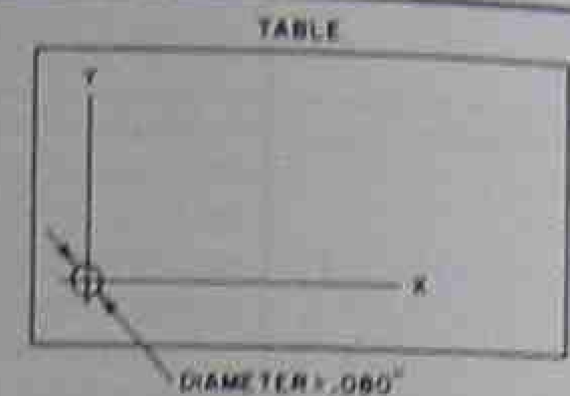
Term	Definition	Example
Retrofit	Stands for RETROactively FITting and applies to the up-dating or significant modification of a piece of equipment that is in the field.	A conventional piece of equipment to which a numerical control system is added. A control system that is replaced by a much more up-to-date CNC system.
Row	A path perpendicular to the edge of the tape along which information may be stored by presence or absence of holes or magnetized areas. A character would be represented by a combination of holes and no holes across the tape which would fall along a row.	
Second Generation	In the NC field, the period of technology associated with transistors (solid state). In the computer field, the period of technology utilizing solid state circuits, off-line storage, and a significant development in software.	
Sequence Number	A number identifying the relative location of blocks, or groups of blocks on a tape. The sequence number is identified by the letter "n" as the address and usually falls at the beginning of a block. A control system incorporating a sequence number display would "read out" the number of the block being read by the tape reader which would correspond to the number following the address letter. This feature enables the operator to identify the tape location with respect to the position of the machine. An additional feature called *block search* allows the operator to manually enter a block, or sequence number, and the control system will automatically search for the number. Sometimes referred to as *tape search*. (RS-244-B standard is shown.)	 234 above, following "n," would identify the block and would be displayed.
Servo Mechanism	A type of closed-loop control system in which mechanical position is the controlled variable.	

Term	Definition	Example
Software	Computer programs used as an assist in part programming. The APT program is an example of an automatic programming software package.	Contrasted to hardware which refers to the machine tool, the control system, and other pieces of equipment.
Softwired Control System	See CNC.	
Solid State System (also designated as a "Second Generation" System)	A control system comprised solely of solid state electronic and electrical components (i.e., having no moving, vacuum tube, or gaseous tube components. Mechanical tape readers or photo tape reading tubes are not considered).	
Source Language	A language used as input to a computer which is translated to the *machine language* that the computer requires.	FORTRAN is an example. Also numerical control part programming languages, such as APT and COMPACT II, are often referred to as source languages.
Speed	Term referring to the rotation of the spindle in revolutions per minute (rpm). Since the speed of an NC machine is programmed in rpm, it is often necessary to convert from the conventional form of describing the speed in feet per minute (fpm).	The formula for the conversion is: $\text{rpm} = \frac{\text{fpm} \times 12}{\pi \times \text{Dia. of Cutter}}$ The "12" converts feet to inches.
Stepping Motor	A type of motor that rotates a fixed amount for every electrical pulse it receives.	Used as a drive motor, usually with open loop systems. Each pulse rotates the rotor shaft 1.8°. Using a .2" pitch lead screw, each pulse therefore, would result in a linear movement of .001" per pulse.
Straight Cut Control System	A system which has feedrate control only along a single axis and, therefore, can control cutting action only along a path parallel to the linear (or circular) machine ways. Cannot coordinate two or more axes to produce true contouring. Also designated as "Picture Frame" milling.	
Sub-Routine	A section of a computer program which is stored in the computer memory and can be used over and over again to accomplish a	

Term	Definition	Example
	certain operation, e.g., square root, cube root, sine, log, etc.	
Swarf Cut	The removal of metal in such a manner as to generate a cutter-path whose axis is constantly changing in relation to the part surface.	Accomplished with machines having four or five axes on a single spindle, or with a combination of a single spindle and movable holding fixture.
Tab Sequential Format	Means of identifying a word by the number of tab characters preceding the word in a block. The first character of each word is a tab character. Words must be presented in a specific order but all characters in a word, except the tab character, may be omitted when the command represented by that word is not desired. (See Chapter 3.)	
Tachometer	A speed-measuring instrument generally used to determine revolutions per minute. Tachometers may be used in conjunction with contouring systems as a supplemental control for governing feedrates.	
Temporary Storage	See Buffer Storage	
Third Generation	In the NC field, the period of technology associated with integrated circuits. In the computer field, the period of technology utilizing integrated circuits, core memory, and advanced programming concepts.	
Tool Function	A tape command identifying a tool and calling for its selection. The address is normally a "T" word and may be used in conjunction with a turret or automatic tool changer.	 T01 would bring (1) position into operation.

Term	Definition	Example
Tool Offset	In order to avoid the time consuming requirement for setting the depth of the tool to an exact dimension in a turret, the tool may be set to an approximate dimension (generally within 0.1 inch of the required setting) and manual switches can be used to make up the difference. Found extremely helpful with turrets on a lathe or adjusting the "Z" or depth motion of a turret drill having three axes of control.	 When setting single-point tools, as in the turret of a lathe, it is often difficult, if not impossible, to determine cutter deflection or the net effect that machining has on the position of the cutting tool. An adjustment is most often necessary after the first cut. This may be readily accomplished with numerical control by setting the exact depth via manual switches. If, as an example, the tool is initially set at position (A) and after cutting it is realized that due to forces and deflection the tool should be at position (B), the tool need not be touched but the difference dialed in. Whenever the T03 turret position is called for by the code on the tape the turret will automatically adjust to position (B). Each tool face may be adjusted independently. Correct compensation is automatically adjusted as the tool face is indexed to the cutting position.
Track	Also known as level or channel. A path parallel to the edge of the tape along which information may be stored by the presence or absence of holes or magnetized areas. The EIA standard 1-inch wide tape has eight tracks.	
Transducer	A device which converts one form of energy into another form of energy. A thermocouple which	

Term	Definition	Example
	converts temperatures into millivolts is a type of transducer. In numerical control design, it usually applies to a device for converting rotary motion into a varying sine wave voltage. A resolver is a type of transducer.	
Transistor	A solid state electronic device which functions in a manner similar to a triode vacuum tube. It consists of a small block of a semiconductor material that has at least three "electrodes." It is light, almost unbreakable, long-lived and highly efficient.	
Undercut	A machine cut where the cutter failed to arrive at a programmed point following a command change in direction (in contrast to Overshoot).	
Variable Block Format	A tape format which allows the number of words in successive blocks to vary.	The word address system is a variable block format.
Word	An ordered set of characters which may be used to cause a specific action of a machine tool.	*X*, *Y*, and *Z* together with the numerical values would be described as dimension words. Other words with their appropriate numerical values would be the G and F words describing preparatory and feedrate functions, respectively.
Word Address Format	Addressing each word in a block by one or more characters which identify the meaning of the word.	*X* would be the address for the word X54693 which would call for a motion in the *X* direction of 5.4693 inches.
Zero Offset (sometimes referred to as Zero Shift)	A means of shifting the coordinate zero point from a fixed known zero point. The zero offset "shift" is normally accomplished by a series of switches or dials.	 If it is required to shift the permanent zero position, say, 3.000 inches in the + *X* direction and 2.000 inches in the + *Y* direction, it would be necessary to dial in

Term	Definition	Example
		these dimensions. A new zero point will then have been established and all part dimensions, when programming, will be taken from the new zero point.
Zero Reset (also known as Zero Synchronization)	The ability to automatically realign each axis slide to the zero or target point by pressing a button when the slides are brought within close proximity of the target point.	 If only the *X* and *Y* axes are considered, it is not necessary to "manually" realign the cutting tool to the exact starting or target point. It is only necessary that the tool be brought within a circle whose radius is approximately .040" around the target point and the machine will automatically realign itself to the starting point by pressing a button.
Zero-suppression	The elimination of non-significant zeros *either* before or after the significant figures in a tape command. The purpose is to reduce the number of characters that are required to be read.	In the tape word X+01000 A control system responding only to trailing zero-suppression would require: X+01 A system responding only to leading zero-suppression would require: X+1000 The preceding and trailing zeros may be shown if desired. There are some CNC systems that allow suppression of both leading and trailing zeros. In those cases, the decimal point must be shown.

NUMERICAL CONTROL STANDARDS

Standardization has played a significant role in the design and manufacture of both consumer and industrial products for some time. Many standardized items which are now taken for granted required considerable study and negotiation by the manufacturers in order to arrive at specifications which would

provide economy in manufacture and convenience and safety in use. An example of a common consumer item that has been standardized is the electric light bulb which required an agreement as to thread and socket dimensions. Another electrical standard which is used daily and taken for granted is the distance between the receptacle holes of an electrical outlet. Imagine the confusion and inconvenience if every manufacturer of light bulbs and electrical outlets had his own proprietary dimensions. Numerous industrial standards covering machined items and fabricated materials have necessarily been in force for some time. A certain amount of standardization is, therefore, essential if a product is to be satisfactorily utilized by the public.

As a rule most standards describe minimum performance requirements and are of a nature that allows manufacturers wide latitude in their development of products to meet these general requirements. A problem arises in developing a standard when a number of manufacturers have individual products on the market which differ somewhat widely. It is then necessary that these manufacturers cooperate on a technical basis through committee action in order to develop a useful standard. When the standard is adopted, it is often necessary that most or all of the manufacturers modify their product to some degree to meet the standard requirements.

During the early periods of numerical control development, it was realized that both performance and product standards would be required if the field was to be successful. The task of developing standards was undertaken by both the Electronic Industries Association (EIA) which represents manufacturers of control systems and machine tools as well as users and the Aerospace Industries Association (AIA) which represents user companies in the aerospace industry. Considerable effort and committee action have been spent on these standards over the past two decades, which has resulted in a considerable amount of compatibility and which, as an example, permits users of numerical control equipment to prepare tapes for different control systems and machine tools on the same piece of equipment, as well as procure one type of raw tape stock. The tape format and language has also been standardized so that information can be readily exchanged between contractors as well as subcontractors and between in-plant personnel.

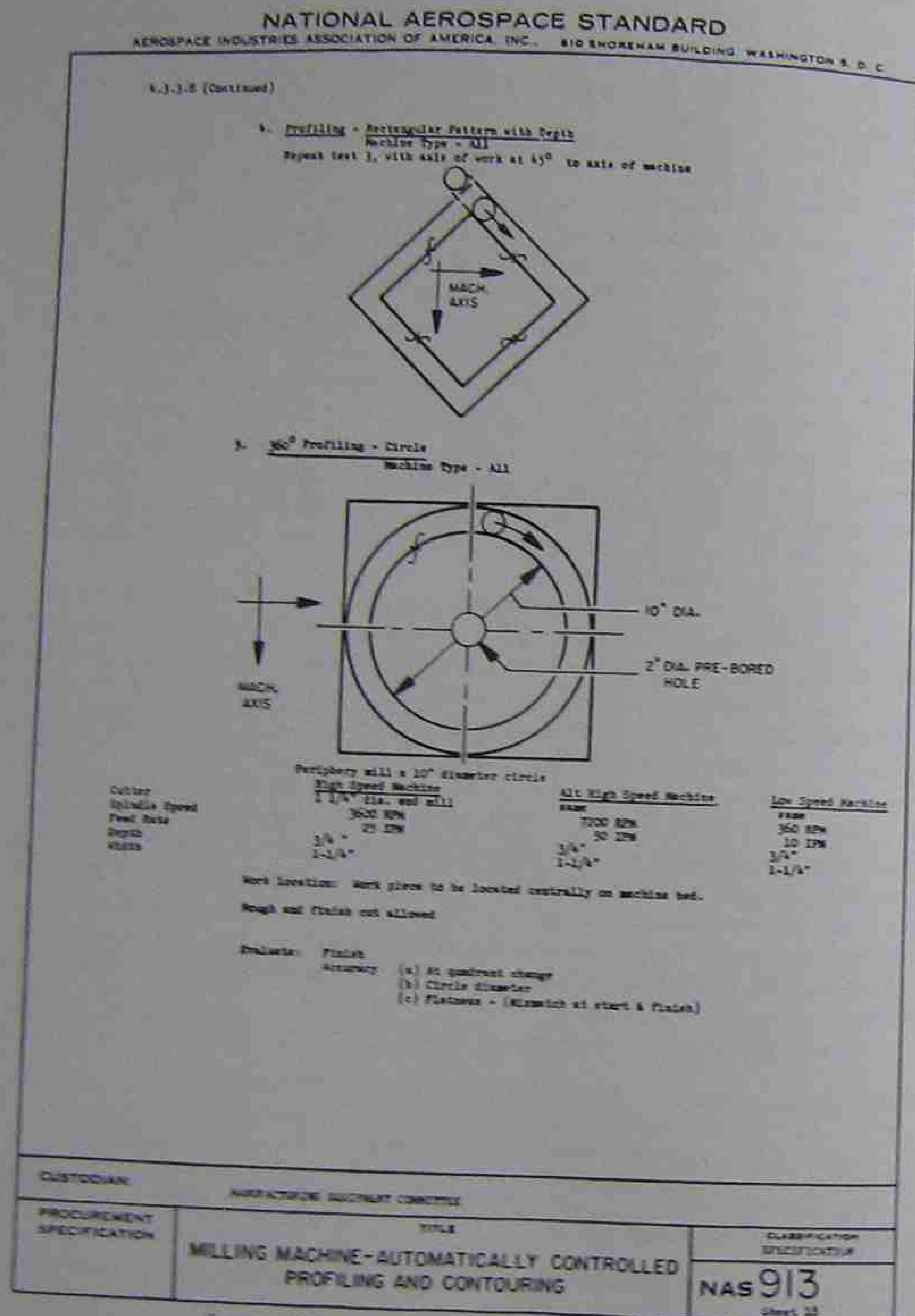
EIA standards may be procured at a nominal charge from:

Electronic Industries Association
2001 Eye Street, N. W.
Washington, D. C. 20006

Reproducible copies of Aerospace Industries standards may be procured also at a nominal charge from:

National Standards Association, Inc.
5161 River Road
Bethesda, MD 20816

It should be noted that the EIA standards which were prepared by manufacturers of numerical control equipment and a broad range of users are more general in nature since the application of numerical control equipment was considered for a wide range of manufacturing requirements; whereas the AIA



Courtesy of Aerospace Industries Association of America, Inc.

Fig. 10–1. Typical page from AIA standard NAS 913. Shown are cutting tests for a numerically controlled milling machine.

standards were developed by a particular industry and are concerned primarily with requirements of aerospace manufacturing. Tolerance requirements and other details listed in the AIA specs may necessarily, therefore, be more stringent than those requirements normally encountered in other manufacturing fields. Equipment manufacturers are not in the least reluctant to meet relatively close tolerance requirements, however, it must be remembered that the cost of equipment increases considerably as the tolerances become tighter. These standards are, therefore, offered as a guide to both equipment manufacturers, when designing for numerical control, and for users, when considering procurement specifications.

The EIA and AIA have been joined by other organizations that are either reviewing, identifying, or proposing numerical control standards. These include the American National Standards Institute (ANSI), Computer Aided Manufacturing–International, Inc. (CAM-I), the Department of Defense (DoD), the National Machine Tool Builders Association (NMTBA), the Numerical Control Society (NCS), and the Society of Manufacturing Engineers (SME).

The addresses of these organizations and their relation to numerical control standards activities are described below.

American National Standards Institute (ANSI) 1430 Broadway New York, NY 10018 (212) 354-3300	Develops American National Standards and participates in international standards efforts. Heavily involved with computer software. Two ANSI committees related to numerical control are X3J7, which is responsible for developing APT programming language standards, and X3J5, which is responsible for developing standards for the COMPACT II/ACTION/SPLIT family of languages. ANSI has also developed a standard (ANSI B5.50) for toolholders for automatic tool changing NC machines entitled *"V" Flange Tool Shanks for Machining Centers with Automatic Tool Changers.*
Computer Aided Manufacturing–International, Inc. (CAM-I) 611 Ryan Plaza Suite 1107 Arlington, TX 76012 (817) 265-5328	Does not develop standards; however, active in identifying and promoting standards which apply to numerical control.
Department of Defense (for specifications) Commanding Officer Navy Publications & Forms Center Code 512 5801 Tabor Ave. Philadelphia, PA 19120	Have developed a wide range of standards (MILSPECS) for numerically controled machine tools. Standards are used primarily for the procurement of numerical control machines by government agencies; however, may be used as a guide by industry.

Department of Defense (for information) Commanding Officer Defense Industrial Plant Equipment Center Airways Blvd. Memphis, TN 38102	
International Organization for Standardization (ISO) 1, rue de Varembe Geneva, Switzerland	Issues standards that have been accepted on an international level. Many of the ANSI standards also become ISO standards. ISO standards may be purchased in the U.S. from the American National Standards Institute (ANSI). See address above.
National Machine Tool Builders Association (NMTBA) 7901 Westpark Drive McLean, VA 22101 (703) 893-2900	Does not develop standards; however, offers a number of publications covering numerical control machine tools and related standards including a *Directory of NC Machine Tools*, an *NC Character Code Cross Reference Chart*, a publication on *Common Words as They Relate to NC Software*, and a publication entitled *A Definition of Accuracy and Repeatability of NC Machine Tools*.
Numerical Control Society (NCS) 519-520 Zenith Drive Glenview, IL 60025 (312) 297-5010	Does not develop standards; however, active in identifying and promoting standards which apply to numerical control.
Society of Manufacturing Engineers (SME) Technical Divisions SME Headquarters 20501 Ford Road, P. O. Box 930 Dearborn, MI 48128 (313) 271-1500	Does not develop standards; however, active in identifying and promoting standards which apply to numerical control.

Since the EIA standards are probably the most recognized and comprehensive of the numerical control standards, they are described below. It should be noted that these are standards that existed at the time of this writing. Others are expected to be added and, most likely, some will be deleted.

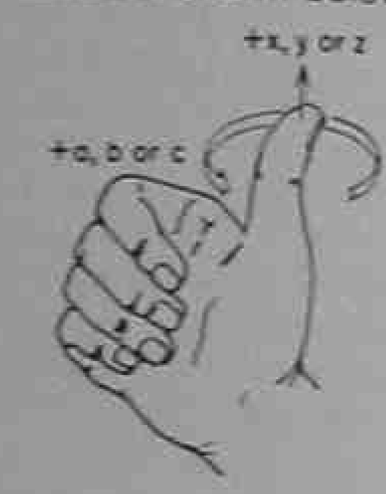
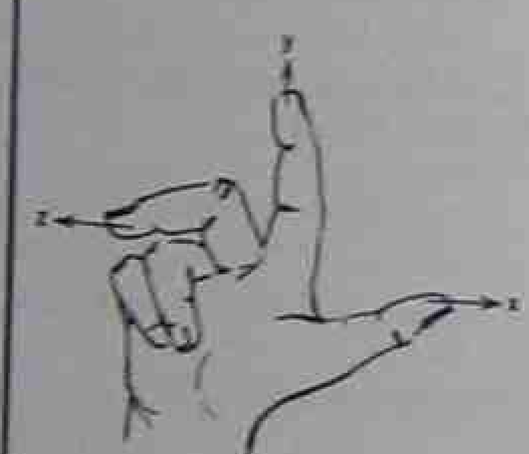
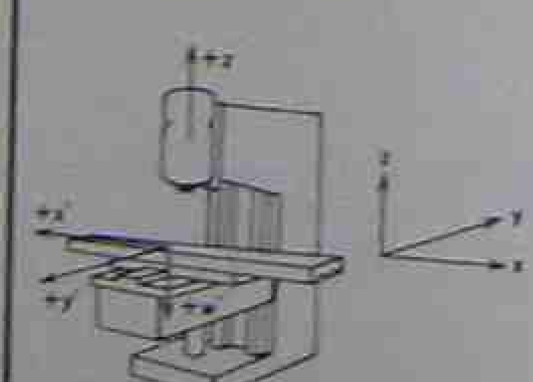
EIA Standards

Standard Number	Title	Explanation
EIA Automation Bulletin No. 3C Issued March 1973	Glossary of Terms for Numerically Controlled Machines	Definitions are offered covering many of the terms used in numerical control.

EIA Standards (*Continued*)

Standard Number	Title	Explanation
RS 227A Revised Oct 1971	One-inch Perforated Tape	Describes the dimensions and tolerances recommended for the tape input medium.
RS 232-C Latest Revision August 1969	Interface Between Data Terminal Equipment and Data Communication Equipment Employing Serial Binary Data Interchange	Describes the interface (connections) between data terminal equipment, such as a teletype typewriter and a communication modem which would be used with a remote time-share part programming system. It is intended that RS 449 gradually replace this standard. A common NC reference to RS 232 is the plug (or plug receptacle) in a CNC system for tieing in an external tape reader, such as a teletype typewriter or other intelligent programming terminal, with the CNC control system.
Bulletin No. 9	Application Notes for EIA Standard RS 232-C	Expands on RS 232-C by describing details of different applications.
RS 244-B Revised October 1976	Character Code for Numerical Machine Control Perforated Tape	Describes the physical characteristics of the tape (1" wide, 8 track) in addition to the code punching for various letter and numerical characters. Although the EIA has not officially dropped this standard in favor of RS 358 (ASCII), it does note that "users and builders of numerically controlled equipment should be aware that the flexibility and expandability of the code (RS 358) . . . make it more applicable than RS 244 for newer, advanced numerical control systems."

EIA Standards (*Continued*)

Standard Number	Title	Explanation
RS 267-A Latest Revision (June 1967) (Reaffirmed May 1973)	Axis and Motion Nomenclature for Numerically Controlled Machines	Identifies the axis motions of numerous types of machine tools. In the early development of numerical control, it was found necessary that specific machine motions be identified in a common language in order that people in the field, particularly programmers, would have a common basis of understanding as to the various motions. The illustration shown below,   which is taken directly from the standard describes the general right-hand rule for both linear and rotary motions. The axis designations on the vertical knee mill, shown below, are also 

EIA Standards (*Continued*)

Standard Number	Title	Explanation
		taken directly from the standard. The positive *x'* and *y'* are table motions which actually result in positive *x* and *y* cutting motion on the workpiece with respect to the spindle.
RS 273-A Latest Revision May 1967	Interchangeable Perforated Tape Variable Block Format for Positioning and Straight Cut Numerically Controlled Machines	In addition to describing the character coding as noted in RS 244-A it was found necessary to also assign word address codes to various operations such as for auxiliary functions, spindle speed and feed codes as well as preparatory and sequence number addresses. *Note: The EIA no longer supports this standard.* (Refer to RS 274-D.)
RS 274-D Latest Revision February 1979	Interchangeable Variable Block Data Format for Positioning, Contouring, and Contouring/Positioning Numerically Controlled Machines	This revised standard includes the coverage that was once offered in RS 273-A. The peculiarities of contouring, such as circular interpolation are covered in addition to decimal point programming and application notes on four-axis lathes. A means of describing the blocks, their limitations, and a format code which could be used when specifying equipment is also part of this specification. (This is described in detail in Chapter 3.)
EIA Automation Bulletin No. 4 Issued March 1969	Recommended Interchangeable Perforated Tape Variable Block Format for Contouring and Positioning Numerically Controlled Stored Program Machines	Developed as an extension to RS 273 and RS 274 in order to cover CNC controls.
RS 281-B Latest Revision June 1979	Electrical and Construction Standards for Numerical Machine Control	This standard is primarily intended as a guide to system manufacturers as an assist in the design of numerical control equipment.

EIA Standards (*Continued*)

Standard Number	Title	Explanation
		It may also be very successfully utilized by those investigating the purchase of numerical control equipment. Some of the items covered include: 1. Those diagrams, layouts and instruction books which should be furnished with the control system. 2. Specifications for components particularly with regard to identification and maintenance. 3. General construction methods and practices covering the recommended types of cabinets, mounting printed circuit boards and electrical accessories.
RS 326-A Revised January 1969	Interchangeable Perforated Tape Fixed Block Format for Positioning and Straight Cut Numerically Controlled Machines	This standard differs from RS 273-A (Interchangeable Perforated Tape *Variable* Block Format for Positioning and Straight Cut Numerically Controlled Machines) in that the number of characters in a block is fixed instead of using a variable number. *Note: The EIA no longer supports this standard.* (Refer to RS 274-D.)
RS 358-B Revised February 1979	Subset of American National Standard Code for Information Interchange for Numerical Machine Control Perforated Tape	This standard has, in effect, replaced the long-standing RS 244 standard. The character coding and the parity are different than that of RS 244. Both standards are described in Chapter 3.
RS 408 Issued March 1973	Interface Between Numerical Control Equipment and Data Terminal Equipment Employing Parallel Binary Data Interchange	Refers to the "behind the tape reader" (BTR) interface between a general purpose computer and the control system in a DNC (direct numerical control) operation.

EIA Standards (*Continued*)

Standard Number	Title	Explanation
RS 431 Issued September 1976	Electrical Interface between Numerical Control and Machine Tools	This standard is intended to serve as a guide in the design of an electrical interface between Numerical Control Systems and electrical equipment associated with machine tools.
RS 441 Issued January 1979	Operator Interface Functions of Numerical Control	Describes the various operator control functions and lists their nomenclature. For example, operator functions such as feedhold, jog, and emergency stop are described. Identifies basic operator devices on an NC panel.
RS 447 Issued October 1977	Operational Command and Data Format for Numerically Controlled Machines	Defines input data, commands, and format for input to a CNC control system for such functions as machine setup, control initialization and editing. CNC systems may receive this type of information and data from the punched tape, keyboard, or other media.
RS 449 Issued November 1977	General Purpose 37-Position and 9-Position Interface for Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange	An improvement over RS 232-C, this standard is expected to eventually replace RS 232-C.
Bulletin No. 12	Application Notes on Interconnection Between Interface Circuits Using RS 449 and RS 232-C	Expands on RS 232-C and RS 449 by describing details of different applications.
RS 474 Issued March 1982	Flexible Disk Format for Numerical Control Equipment Information Interchange	Provides for the use of floppy disks as an alternative to punched paper tape.

Index

A

absolute dimensioning, 90–91, (def.) 252
accuracy, 166–169, (def.) 252–253
acoustic coupler, (def.) 253
adaptive control, (def.) 253
address, (def.) 253
address characters, tables of,
  additional, 70–72
  G word, 63–68
  M word, 69–70
Aerospace Industries Association (AIA), 5, 203, 287
algorithm, (def.) 253
alphanumeric, (def.) 254
American National Standards Institute (ANSI), 289
analog, (def.) 254
APT, 112–127, 233, (def.) 254
  fundamentals of, 116–127
    geometry statements, 119–126
    moving the cutter, 116–119
    point-to-point patterns, 126–127
  writing a program
    for milling, 133–135
    for turning, 137–139
ARELEM, 254
assembler, 115
assembly program, (def.) 255
automatic acceleration and deceleration, (def.) 255
automatic factory, 243–244, 246, 249
automatic part handling, 241–243
automatic program, (def.) 255
auxiliary or miscellaneous function (def.) 255
auxiliary storage, (def.) 256
axis, (def.) 256

B

backlash, (def.) 256
BASIC, (def.) 256
batch processing, (def.) 256
band, (def.) 256
Binary Coded Decimal (BCD) format, (def.) 257
8-level BCD, 58
binary numbers, 114–115, (def.) 256–257
bit, 55, (def.) 257
block, 56–57, (def.) 257
block count readout, (def.) 257
block delete, (def.) 257–258
buffer storage, (def.) 258
byte, (def.) 258

C

canned cycles, 95, 96–101, (def.) 258–259
drills, 96–97
lathes, 99–101
machining centers, 96–97
Cartesian coordinate system, 79–87
cathode ray tube (CRT), 233
chad, (def.) 259
channel (see level)
character, 55, (def.) 259
check surface, APT, 116–117
circuit board, 30–32
circular interpolation, 38, 92–95, (def.) 259
closed loop system, 41–44, (def.) 260
CNC (see computer numerical control)
CNC part programming (see *also* computer numerical control), capabilities of, 95–105
code, (def.) 260
command, (def.) 260
readout, (def.) 279
COMPACT II, 112–116, 127–133
fundamentals of, 127–133
geometry statements, 129–132
moving the cutter, 127–129
point-to-point patterns, 132–133
writing a program
for milling, 135–137
for turning, 139–140
compatibility, (def.) 261
compiler, 115, (def.) 261
FORTRAN, 115
computer-aided design (CAD), 233, (def.) 258
computer-aided manufacturing (CAM), 233, (def.) 258
Computer-Aided Manufacturing–International, Inc. (CAM-I), 112, 289
computer-aided part programming, 110–141
suitability of, 110–111
computer graphics, 233
computer-integrated manufacturing (CIM), 233
computer numerical control (CNC), 28, 34, (def.) 260
and software approach, 29
computers, 113–116
analog, 114
digital, 114
contour control system, 16–24, (def.) 262
contour part programming vs point-to-point programming, 87–90
coordinates, (def.) 262
core, of computer, (def.) 262
CPU (cental processing unit), (def.) 262–263
cutter
compensation, 96, 103–104, (def.) 263
location (CL) information, 259–260
offset, (def.) 263

D

damping, (def.) 263
data base, (def.) 263
data link, (def.) 263
data storage and control, 36–38
dead band, (def.) 263–264
decode, (def.) 264
Department of Defense, 289, 290
diagnostic
programs, maintenance, 218
routine, (def.) 264

differences between NC and conventional machines, 145–146
digital, (def.) 264
digital clock, 39
digital computer, 114, (def.) 264
digital differential analyzer (DDA), 38–39
digitize, (def.) 264
direct numerical control (DNC), 248, 249–250, (def.) 264–265
director for numerically controlled system, 8–9
distributed numerical control (DNC), 248, 249–250, (def.) 264–265
DNC (see direct numerical control; distributed numerical control)
double precision, (def.) 265
drive mechanism, 44–47
drive surface, APT, 116–117
duplex
full, (def.) 269
half, (def.) 269
dwell, (def.) 265

E

economic lot size, 3
edit, (def.) 265
Electronics Industries Association (EIA), 1, 54, 203, 287
format classification sheet, 73, 75–77
standards, 290–295
RS-227-A, 54
RS-244-B, 54, 56, 57, 58, 59
RS-274-D, 54
RS-358-B, 57, 58, 59
encode, (def.) 265
encoder, (def.) 265–266
end of block character (EB), 55, (def.) 266
executive, (def.) 266
external memory, (def.) 266

F

feedback, 10, 12, (def.) 266
device, 10, 27, 40
loop, 40–44
closed, 40
modulator, 42
open, 46
feed function, (def.) 266
feedhold, (def.) 266
feedrate
number, formula for, 62, (def.) 266–267
word, 62
feedrate override, (def.) 267
firmware, (def.) 267
fixed sequential format, (def.) 267
flip–flop, (def.) 267–268
floating point arithmetic, (def.) 268
floating zero, (def.) 268
following error, (def.) 268
FORTRAN, (def.) 268–269
free floating zero, (def.) 268

G

gain, (def.) 269
gate, (def.) 269
generation, NC
first, 28–29, (def.) 267
fourth, 28–29
second, 28–29, (def.) 281
third, 28–29, (def.) 283
G function, 62
table of, 63–68
G92 code, 104–105
group technology, 233–236

H

hardware, (def.) 269
hardwired numerical control, (def.) 270
heuristic, (def.) 270
hexadecimal number, (def.) 270
home position, (def.) 270

I

incremental dimension word, 90–91, (def.) 271

input, (def.) 271
installation, 205–209
integrated circuit, (def.) 271
interface, (def.) 271
International Standards Organization (ISO), 290
interpolation
circular, 38, 92–95, (def.) 259
linear, 90–91, (def.) 272
parabolic, (def.) 276
IO, (def.) 271

L

leading zero suppression, (def.) 271
lead screw, 44
lead-screw compensation, (def.) 272
level, (def.) 57
parity, 57
linear interpolation, 38, 90–91, (def.) 272
loop, (def.) 272

M

machine language, 115, (def.) 272
machining centers, 145–146, 149, 152, 157
macro, (def.) 272–273
magnetics cabinet for numerically controlled system, 10
maintenance, 209–218
manual data input, (def.) 273
manual feedrate override, (def.) 273
manual part programming, 79–106
suitability of, 105–106
manuscript, 79, 85–87, 220, (def.) 273
MAPI method, 184–199
memory, (def.) 273–274
microcomputer, 217, (def.) 274
microprocessor, (def.) 274
miscellaneous function words (see M word)
mnemonic code, (def.) 274
modem, (def.) 274
module (circuit board), 30
M word, 73
table of, 69–70

N

National Machine Tool Builders Association (NMTBA), 288, 290
noise, (def.) 274
numerical control (NC)
advantages of, 3, 145–174
accuracy, 166–167, 169
design versatility, 173
elimination of scrap, 170
improved planning and control, 173
less floor space needed, 170–171
less-skilled operators required, 174
lower capital investment, 170
lower inventory, 173
repeatability, 166–167, 169
savings
in facility requirements, 170–171
in sheet metal fabricating, 150, 152
time savings
in contour milling, 152, 153, 154, 155, 156–157
floor-to-floor machining, 148, 149, 150, 151
floor-to-floor production, 147–148
reduced inspection time, 171–172
in turning, 159–161
tooling cost savings, 162–163
examples, 163–166
continuous-path, 16–17, 20, 24
contouring control, 16–17, 20, 24
definitions, 1, 251–286
developmental chronology, 245–246
and electro-discharge machining (EDM), 20, 23, 24
equipment
considerations for selection and procurement, 202–205
design considerations for, 214, 216–218
establishing in-plant program for, 210, 212–214
installation of, 205–209
general considerations, 226–228
justification of, 176–199
considerations, 177, 179
methods for justifying
payback, 183–184
MAPI, 184–199
suitability of, 179–182
factors affecting wider utilization, 232–233
history of, 4–7
maintenance of, 209–210
preventive, 218, 219
point-to-point, 12–13, 16
standards, 286–295
system, (def.) 275
technology vs economics, 250
numerical control installation
control system, 28–40
data storage and control, 36–38
integrated circuitry, 28
pulse generation, 38–40
and circular interpolation, 38
digital differential analyzer (DDA), 38, 39
and linear interpolation, 38
pulse rate multiplier, 38
solid state components, 28
tape reader, 32–36
elements comprising, 26
Numerical Control Society (NCS), 289, 290
numerical data, (def.) 275
numerically controlled machines
director, 8–9
growth of, 4, 7
magnetics cabinet, 10
operator's console, 10
pendant, 10

O

octal numbers, (def.) 275
off-line operation, (def.) 275
on-line operation, (def.) 275
open-loop system, (def.) 275
operating system, (def.) 276
operator's console for numerically controlled system, 10
optional stop, (def.) 276
overlay, (def.) 276
overshoot, (def.) 276

P

parabolic interpolation, (def.) 276
parity check, (def.) 276–277
part handling, automatic, 241–243
part programming, 218, 220–222
programmer qualifications for, 221
systems, 140–141
training for, 220, 221
part surface, APT, 116–117
payback method, 183–184
pendant for numerically controlled system, 10
point-to-point control system, 12–13, (def.) 277
point-to-point programming
programmer qualifications for, 221
vs contour part programming, 87–90
position transducer, 42, (def.) 277
postprocessor, (def.) 277
precision, (def.) 278
double, (def.) 265
preload of registers (see G92 code)
preparatory function (see *also* G function), (def.) 278
preventive maintenance, 218
programmer, part, (def.) 278–279
program stop, (def.) 279
pulse generation, 38–40
pulse rate multiplier, 39

R

read-out position, (def.) 279–280
read-out, slide location, (def.) 280
real time operation, (def.) 280
remote job entry, (def.) 280
repeatability, 166–169, (def.) 280
reproducibility, (def.) 280
resolution, (def.) 280

resolver (see *also* position transducer), 42
retrofit, 48, 50–51, (def.) 281
  advantages of, 51
  limitations of, 48, 50
"right-hand" rule, 56, 292
robots, 236–241
ROM (read only memory), (def.) 279
row, (def.) 281

S

selection and procurement, 202–205
sequence number, 58, (def.) 281
servo mechanism, 40–44, (def.) 281
size of machine, determination of, 187–188
Society of Manufacturing Engineers (SME), 289, 290
software, (def.) 282
softwire vs hardwire systems, 29
solid state system, (def.) 282
source language, (def.) 282
speed, (def.) 282
speed word (see S word)
standards for numerical control, 290–295
stepping motor, 40, (def.) 282
straight cut control system, (def.) 282
subroutine, 96, 102–103, (def.) 282–283
  parametric, 96, 103
swarf cut, (def.) 283
S word, 62, 73

T

tab sequential format, 73, 74, (def.) 283
tachometer, (def.) 283
tape
  coding, 55–77
  8-level BCD, 58
  format
    tab sequential, 73, 74
    word address, 73
  Mylar, 35
  preparation, 223, 225–226
  reader, 32, 36
tool function, (def.) 283
tool offset, (def.) 284
tool word (see T word)
track (see *also* level), (def.) 284
transducer, (def.) 284–285
transistor, 29, (def.) 285
T word, 73

U

undercut, (def.) 285

V

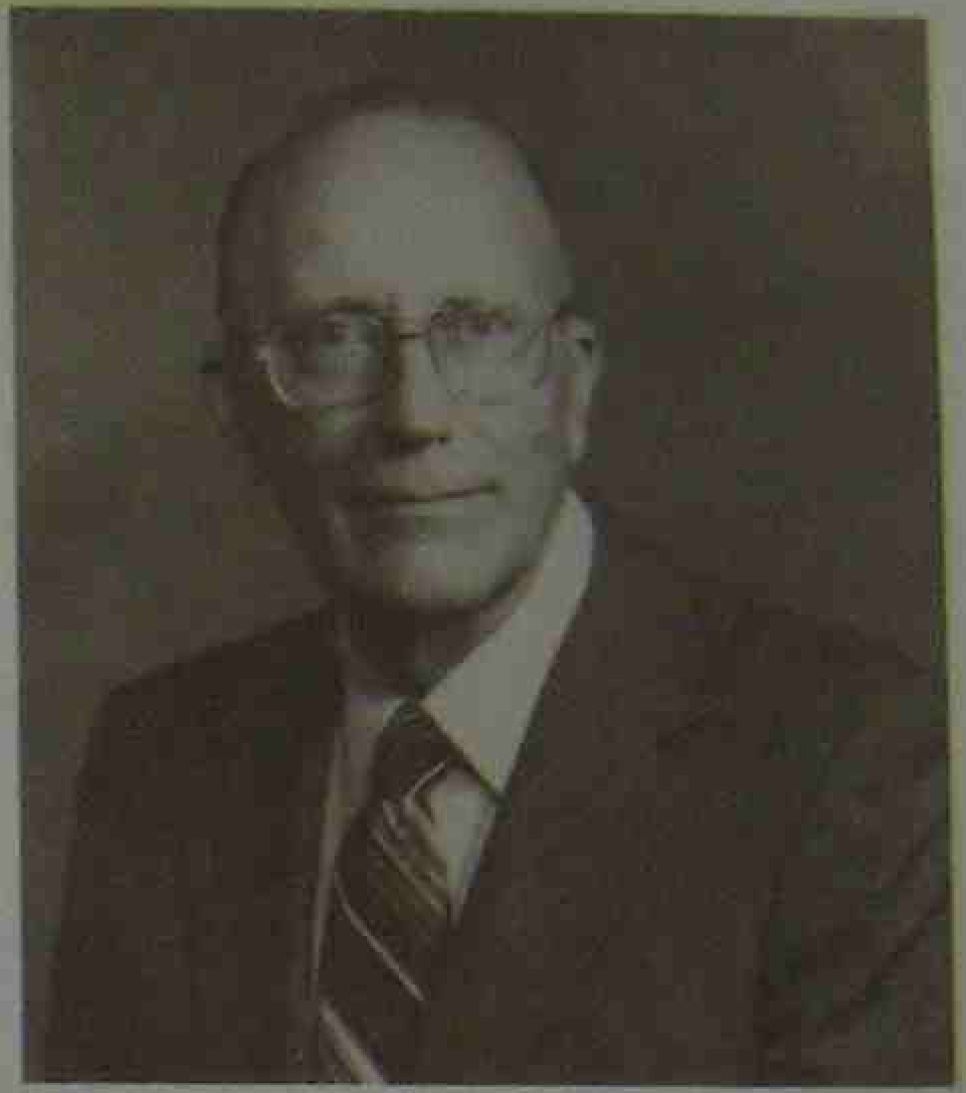
variable block format, (def.) 285

W

word, 55–56, (def.) 285
word address format, (def.) 285

Z

zero offest, (def.) 285–286
zero reset, (def.) 286
zero suppression, (def.) 286



James J. Childs is president of James J. Childs Associates, Inc., a professional consulting organization established in 1964, that is devoted exclusively to numerical control and computer integrated manufacturing. He received his B.S. degree in Industrial Engineering from Columbia University, School of Engineering and has been working exclusively in the numerical control computer integrated manufacturing field since 1955 when he was made head of Republic Aviation Corporation's numerical control machine tool program. He has been chairman of the Numerical Control Sub-Committee of the Aerospace Industries Association and has served on the Electronic Industries Association's Committee for Numerical Control Standards. He is a certified manufacturing engineer of the Society of Manufacturing Engineers and a founding member of the Numerical Control Society, having served on its National Board of Directors and as a chapter chairman. He has taught industrial engineering courses at the State University of New York and numerical control and computer integrated manufacturing courses to numerous government, educational, and industrial organizations and is author of **Numerical Control Part Programming**, also published by Industrial Press Inc.

The Computer Revolution in Manufacturing

COMPUTER APPLICATIONS IN MANUFACTURING

By Thomas G. Gunn

1981, 204pp., Illus. ISBN 0-8311-1087-2

A timely, up-to-date assessment of the computer revolution within manufacturing industries. As a reference guide and practical handbook for executives, students, and engineers, this accessible text presents a step-by-step introduction to a manufacturing system integrated by computer applications with useful tips on the implementation of manufacturing software application packages and other computer integrated manufacturing applications.

How To Produce—Efficiently and Economically

PRODUCTION PROCESSES:

The Productivity Handbook, 5th Edition

By Roger W. B[illegible]

1981, 1089pp., [illegible] ISBN 0-8311-1088-0

Here are 1089 pages of new perspectives on design for production. The most complete coverage of manufacturing processes, products and their components available. Design details, materials, practical tolerances, and working finishes—all prerequisites for successful economical production are examined in detail; an easy-to-read format allows even the novice to understand and evaluate the various processes discussed.

Numerical Control Textbook/Training Manual

NUMERICAL CONTROL PART PROGRAMMING

By James J. Childs

1973, 354 pp., Illus. ISBN 0-8311-1099-6

A highly recommended textbook/training manual which reflects both the needs of the classroom and the practices of industry. Starting at the beginner's level, the subject is developed to include a detailed treatment of how to write both manual point-to-point and manual and computer-assisted (APT) continuous path programs. The author has used his wide experience as a numerical control consultant to develop each topic in a logical and clearly understandable manner.

INDUSTRIAL PRESS INC.
200 MADISON AVENUE
NEW YORK, NEW YORK 10157

ISBN 0-8311-1135-6