

## **Unit -7 :**

### **CNC MACHINING CENTERS**

#### **INTRODUCTION TO COMPUTER NUMERICAL CONTROL**

The variety being demanded in view of the varying tastes of the consumer calls for a very small batch sizes. Small batch sizes will not be able to take advantage of the mass production techniques such as special purpose machines or transfer lines. Hence, the need for flexible automation is felt, where you not only get the benefits of rigid automation but are also able to vary the products manufactured thus bringing in the flexibility. Numerical control fits the bill perfectly and we would see that manufacturing would increasingly be dependent on numerical control in future.

#### **Numerical control**

Numerical control of machine tools may be defined as a method of automation in which various functions of machine tools are controlled by letters, numbers and symbols. Basically a NC machine runs on a program fed to it. The program consists of precise instructions about the methodology of manufacture as well as movements. For example, what tool is to be used, at what speed, at what feed and to move from which point to which point in what path. Since the program is the controlling point for product manufacture, the machine becomes versatile and can be used for any part. All the functions of a NC machine tool are therefore controlled electronically, hydraulically or pneumatically. In NC machine tools, one or more of the following functions may be automatic.

- a. Starting and stopping of machine tool spindle.
- b. Controlling the spindle speed.
- c. Positioning the tool tip at desired locations and guiding it along desired paths by automatic control of motion of slides.
- d. Controlling the rate of movement of tool tip ( feed rate)
- e. Changing of tools in the spindle.

#### **Functions of a machine tool**

The purpose of a machine tool is to cut away surplus material, usually metal from the material supplied to leave a work piece of the required shape and size, produced to an acceptable degree of accuracy and surface finish. The machine tool should possess certain capabilities in order to fulfill these requirements. It must be

- a. Able to hold the work piece and cutting tool securely.
- b. Endowed the sufficient power to enable the tool to cut the work piece material at economical rates.
- c. Capable of displacing the tool and work piece relative to one another to produce the required work piece shape. The displacements must be controlled with a degree of precision which will ensure the desired accuracy of surface finish and size.

### **Concept of numerical control**

Formerly, the machine tool operator guided a cutting tool around a work piece by manipulating hand wheels and dials to get a finished or somewhat finished part. In his procedure many judgments of speeds, feeds, mathematics and sometimes even tool configuration were his responsibility. The number of judgments the machinist had to make usually depended on the type of stock he worked in and the kind of organization that prevailed. If his judgment was an error, it resulted in rejects or at best parts to be reworked or repaired in some fashion.

Decisions concerning the efficient and correct use of the machine tool then depended on the craftsmanship, knowledge and skill of the machinist himself. It is rare that two expert operators produced identical parts using identical procedure and identical judgment of speeds, feeds and tooling. In fact even one craftsman may not proceed in same manner the second time around.

. Process planners and programmers have now the responsibilities for these matters.

It must be understood that NC does not alter the capabilities of the machine tool. The  
With NC the correct and most efficient use of a machine no longer rests with the operator. Actual machine tool with a capable operator can do nothing more than it was capable of doing before a MCU was joined to it. New metal removing principles are not involved. Cutting speeds, feeds and tooling principles must still be adhered to. The advantage is idle time is reduced and the actual utilization rate is much higher (compresses into one or two years that a conventional machine receives in ten years).

### **Historical Development**

1947 was the year in which Numerical control was born. It began because of an urgent need. John C Parsons of the parson's corporation, Michigan, a manufacturer of helicopter rotor blades could not make his templates fast enough. so he invented a way of coupling computer equipment with a jig borer.

The US air force realized in 1949 that parts for its planes and missiles were becoming more complex. Also the designs were constantly being improved; changes in drawings were frequently made. Thus in their search for methods of speeding up production, an air force study contract was awarded to the Parson's Corporation. The servomechanisms lab of MIT was the subcontractor.

In 1951, the MIT took over the complete job and in 1952; a prototype of NC machine was successfully demonstrated. The term "Numerical Control" was coined at MIT. In 1955 seven companies had tape controlled machines. In 1960, there were 100 NC machines at the machine tool shown in Chicago and a majority of them were relatively simple point to point application.

During these years the electronics industry was busy. First miniature electronic tubes were developed, then solid state circuitry and then modular or integrated circuits. Thus the reliability of the controls has been greatly increased and they have become most compact and less expensive.

Today there are several hundred sizes and varieties of machines, many options and many varieties of control system available.

### **Definition:**

The simplest definition is as the name implies, "a process a controlled by numbers ". Numerical Control is a system in which the direct insertions of programmed numerical value, stored on some form of input medium are automatically read and decoded to cause a corresponding function on the machine tool which it is controlling.

### **Advantages of NC machine tools:**

#### **1. Reduced lead time:**

Lead time includes the time needed for planning, design and manufacture of jigs, etc. This time may amount to several months. Since the need for special jigs and fixtures is often entirely eliminated, the whole time needed for their design and manufacture is saved.

#### **2. Elimination of operator errors:**

The machine is controlled by instructions registered on the tape provided the tape is correct and machine and tool operate correctly, no errors will occur in the job. Fatigue, boredom, or inattention by operator will not affect the quality or duration of the machining. Responsibility is transferred from the operator to the tape, machine settings are achieved without the operator reading the dial.

### **3. Operator activity:**

The operator is relieved of tasks performed by the machine and is free to attend to matters for which his skills and ability are essential. Presetting of tools, setting of components and preparation and planning of future jobs fall into this category. It is possible for two work stations to be prepared on a single machine table, even with small batches. Two setting positions are used, and the operator can be setting one station while machining takes place at the other.

### **4. Lower labor cost**

More time is actually spent on cutting the metal. Machine manipulation time ex.: Gear changing and often setting time are less with NC machines and help reduce the labor cost per job considerably.

### **5. Smaller batches**

By the use of preset tooling and presetting techniques downtime between batches is kept at a minimum. Large storage facilities for work in progress are not required. Machining centers eliminate some of the setups needed for a succession of operation on one job; time spent in waiting until each of a succession of machine is free is also cut. The components circulate round the machine shop in a shorter period, inter department costs are saved and 'program chasing' is reduced.

### **6. Longer tool life**

Tools can be used at optimum speeds and feeds because these functions are controlled by the program.

### **7. Elimination of special jigs and fixtures**

Because standard locating fixtures are often sufficient of work on machines. the cost of special jigs and fixture is frequently eliminated. The capital cost of storage facilities is greatly reduced. The storage of a tape in a simple matter, it may be kept for many years and manufacturing of spare parts, repeat orders or replacements is made much more convenient.

### **8. Flexibility in changes of component design**

The modification of component design can be readily accommodated by reprogramming and altering the tape. Savings are affected in time and cost.

### **9. Reduced inspection.**

The time spent on inspection and in waiting for inspection to begin is greatly reduced. Normally it is necessary to inspect the first component only once the tape is proved; the repetitive accuracy of the machine maintains a consistent product.

## **10. Reduced scrap**

Operator error is eliminated and a proven tape results in accurate component.

## **11. Accurate costing and scheduling**

The time taken in machining is predictable, consistent and results in a greater accuracy in estimating and more consistency in costing.

### **Evolution of CNC:**

With the availability of microprocessors in mid 70's the controller technology has made a tremendous progress. The new control systems are termed as computer numerical control (CNC) which are characterized by the availability of a dedicated computer and enhanced memory in the controller. These may also be termed "soft wired numerical control".

There are many advantages which are derived from the use of CNC as compared to NC.

- Part program storage memory.
- Part program editing.
- Part program downloading and uploading.
- Part program simulation using tool path.
- Tool offset data and tool life management.
- Additional part programming facilities.
- Macros and subroutines.
- Background tape preparation, etc.

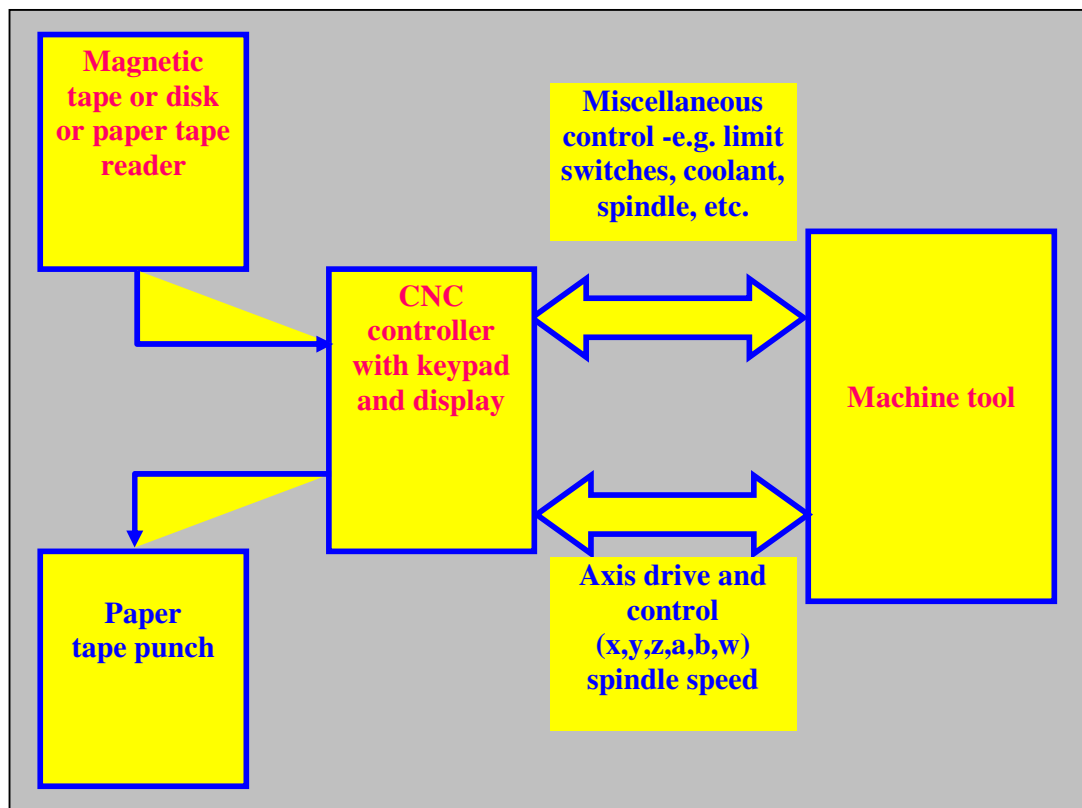
The controls with the machine tools these days are all CNC and the old NC control do not exist any more.



## DEFINITION AND FEATURES OF CNC

### Computer Numerical Control (CNC)

CNC refers to a computer that is joined to the NC machine to make the machine versatile. Information can be stored in a memory bank. The programme is read from a storage medium such as the punched tape and retrieved to the memory of the CNC computer. Some CNC machines have a magnetic medium (tape or disk) for storing programs. This gives more flexibility for editing or saving CNC programs. Figure 1 illustrates the general configuration of CNC.



**Figure 1 The general configuration of CNC.**

### Advantages of CNC

1. Increased productivity.
2. High accuracy and repeatability.
3. Reduced production costs.
4. Reduced indirect operating costs.
5. Facilitation of complex machining operations.

6. Greater flexibility.
7. Improved production planning and control.
8. Lower operator skill requirement.
9. Facilitation of flexible automation.

### **Limitations of CNC:**

1. High initial investment.
2. High maintenance requirement.
3. Not cost-effective for low production cost.

### **Features of CNC**

Computer NC systems include additional features beyond what is feasible with conventional hard-wired NC. These features, many of which are standard on most CNC Machine Control units (MCU), include the following:

- **Storage of more than one part program :** With improvements in computer storage technology, newer CNC controllers have sufficient capacity to store multiple programs. Controller manufacturers generally offer one or more memory expansions as options to the MCU
- **Various forms of program input :** Whereas conventional (hard-wired) MCUs are limited to punched tape as the input medium for entering part programs, CNC controllers generally possess multiple data entry capabilities, such as punched tape, magnetic tape, floppy diskettes, RS-232 communications with external computers, and manual data input (operator entry of program).
- **Program editing at the machine tool :** CNC permits a part program to be edited while it resides in the MCU computer memory. Hence, a part program can be tested and corrected entirely at the machine site, rather than being returned to the programming office for corrections. In addition to part program corrections, editing also permits cutting conditions in the machining cycle to be optimized. After the program has been corrected and optimized, the revised version can be stored on punched tape or other media for future use.
- **Fixed cycles and programming subroutines :** The increased memory capacity and the ability to program the control computer provide the opportunity to store frequently used machining cycles as macros, that can be called by the part program. Instead of writing the full instructions for the particular cycle into every program, a programmer includes a call statement in the part program to indicate that the macro cycle should be executed. These cycles often require that certain parameters be defined, for



example, a bolt hole circle, in which the diameter of the bolt circle, the spacing of the bolt holes, and other parameters must be specified.

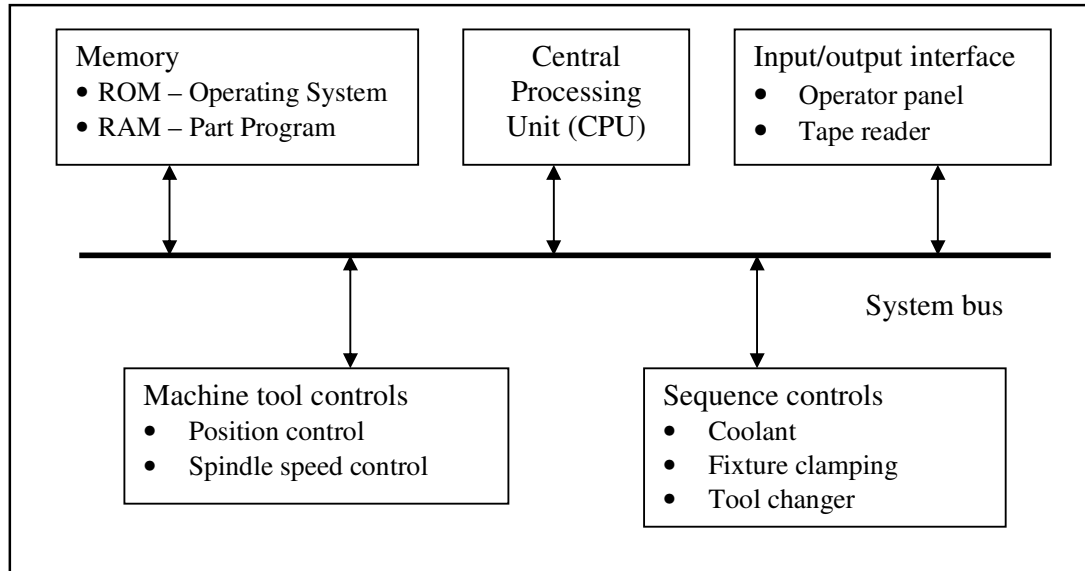
- **Interpolation** : Some of the interpolation schemes are normally executed only on a CNC system because of computational requirements. Linear and circular interpolation are sometimes hard-wired into the control unit, but helical, parabolic, and cubic interpolations are usually executed by a stored program algorithm.
- **Positioning features for setup** : Setting up the machine tool for a given workpart involves installing and aligning a fixture on the machine tool table. This must be accomplished so that the machine axes are established with respect to the workpart. The alignment task can be facilitated using certain features made possible by software options in the CNC system. Position set is one of the features. With position set, the operator is not required to locate the fixture on the machine table with extreme accuracy. Instead, the machine tool axes are referenced to the location of the fixture using a target point or set of target points on the work or fixture.
- **Cutter length and size compensation** : In older style controls, cutter dimensions had to be set precisely to agree with the tool path defined in the part program. Alternative methods for ensuring accurate tool path definition have been incorporated into the CNC controls. One method involves manually entering the actual tool dimensions into the MCU. These actual dimensions may differ from those originally programmed. Compensations are then automatically made in the computed tool path. Another method involves use of a tool length sensor built into the machine. In this technique, the cutter is mounted in the spindle and the sensor measures its length. This measured value is then used to correct the programmed tool path.
- **Acceleration and deceleration calculations** : This feature is applicable when the cutter moves at high feed rates. It is designed to avoid tool marks on the work surface that would be generated due to machine tool dynamics when the cutter path changes abruptly. Instead, the feed rate is smoothly decelerated in anticipation of a tool path change and then accelerated back up to the programmed feed rate after the direction change.
- **Communications interface** : With the trend toward interfacing and networking in plants today, most modern CNC controllers are equipped with a standard RS-232 or other communications interface to link the machine to other computers and computer-driven devices. This is useful for various applications, such as (1) downloading part programs from a central data file; (2) collecting operational data such as workpiece counts, cycle times, and machine utilization; and (3) interfacing with peripheral equipment, such as robots that unload and load parts.

- **Diagnostics** : Many modern CNC systems possess a diagnostics capability that monitors certain aspects of the machine tool to detect malfunctions or signs of impending malfunctions or to diagnose system breakdowns.

### **The Machine Control Unit (MCU) for CNC**

The MCU is the hardware that distinguishes CNC from conventional NC. The general configuration of the MCU in a CNC system is illustrated in Figure 2. The MCU consists of the following components and subsystems: (1) Central Processing Unit, (2) Memory, (3) Input/Output Interface, (4) Controls for Machine Tool Axes and Spindle Speed, and (5) Sequence Controls for Other Machine Tool Functions. These subsystems are interconnected by means of a system bus, which communicates data and signals among the components of a network.

- **Central Processing Unit** : The central processing unit (CPU) is the brain of the MCU. It manages the other components in the MCU based on software contained in main memory. The CPU can be divided into three sections: (1) control section, (2) arithmetic-logic unit, and (3) immediate access memory. The control section retrieves commands and data from memory and generates signals to activate other components in the MCU. In short, it sequences, coordinates, and regulates all the activities of the MCU computer. The arithmetic-logic unit (ALU) consists of the circuitry to perform various calculations (addition, subtraction, multiplication), counting, and logical functions required by software residing in memory. The immediate access memory provides a temporary storage of data being processed by the CPU. It is connected to main memory of the system data bus.
- **Memory** : The immediate access memory in the CPU is not intended for storing CNC software. A much greater storage capacity is required for the various programs and data needed to operate the CNC system. As with most other computer systems, CNC memory can be divided into two categories: (1) primary memory, and (2) secondary memory. Main memory (also known as primary storage) consists of ROM (read-only memory) and RAM (random access memory) devices. Operating system software and machine interface programs are generally stored in ROM. These programs are usually installed by the manufacturer of the MCU. Numerical control part programs are stored in RAM devices. Current programs in RAM can be erased and replaced by new programs as jobs are changed.



**Figure 2** Configuration of CNC machine control unit

High-capacity secondary memory (also called auxiliary storage or secondary storage) devices are used to store large programs and data files, which are transferred to main memory as needed. Common among the secondary memory devices are hard disks and portable devices that have replaced most of the punched paper tape traditionally used to store part programs. Hard disks are high-capacity storage devices that are permanently installed in the CNC machine control unit. CNC secondary memory is used to store part programs, macros, and other software.

- **Input/Output Interface** : The I/O interface provides communication software between the various components of the CNC system, other computer systems, and the machine operator. As its name suggests, The I/O interface transmits and receives data and signals to and from external devices, several of which are illustrated in Figure 2. The operator control panel is the basic interface by which the machine operator communicates to the CNC system. This is used to enter commands related to part program editing, MCU operating mode (e.g., program control vs. manual control), speeds and feeds, cutting fluid pump on/off, and similar functions. Either an alphanumeric keypad or keyboard is usually included in the operator control panel. The I/O interface also includes a display (CRT or LED) for communication of data and information from the MCU to the machine operator. The display is used to indicate current status of the program as it is being executed and to warn the operator of any malfunctions in the CNC system.

Also included in the I/O interface are one or more means of entering the part program into storage. As indicated previously, NC part programs are stored in a variety of ways. Programs can also be entered manually by the machine operator or stored at a central computer site and transmitted via local area network (LAN) to the CNC system. Whichever means is employed by the plant, a suitable device must be included in the I/O interface to allow input of the program into MCU memory.

- **Controls for Machine Tool Axes and Spindle Speed :** These are hardware components that control the position and velocity (feed rate) of each machine axis as well as the rotational speed of the machine tool spindle. The control signals generated by MCU must be converted to a form and power level suited to the particular position control systems used to drive the machine axes. Positioning systems can be classified as open loop or closed loop, and different hardware components are required in each case.

Depending on the type of machine tool, the spindle is used to drive either (1) workpiece or (2) a rotating cutter. Turning exemplifies the first case, whereas milling and drilling exemplify the second. Spindle speed is a programmed parameter for most CNC machine tools. Spindle speed components in the MCU usually consist of a drive control circuit and a feedback sensor interface. The particular hardware components depend on the type of spindle drive.

- **Sequence Controls for Other Machine Tool Functions :**

In addition to control of table position, feed rate, and spindle speed, several additional functions are accomplished under part program control. These auxiliary functions are generally on/off (binary) actuations, interlocks, and discrete numerical data. To avoid overloading the CPU, a programmable logic controller is sometimes used to manage the I/O interface for these auxiliary functions.

## **Classification Of CNC Machine Tools**

### **(1) Based on the motion type 'Point-to-point & Contouring systems'**

There are two main types of machine tools and the control systems required for use with them differ because of the basic differences in the functions of the machines to be controlled. They are known as point-to-point and contouring controls.

#### **(1.1) Point-to-point systems**

Some machine tools for example drilling, boring and tapping machines etc, require the cutter and the work piece to be placed at a certain fixed relative positions at which they must remain while the cutter does its work. These machines are known as point-to-point machines as shown in figure 3 (a) and the control equipment for use with them are known as point-to-point control equipment. Feed rates need not to be programmed. In these machine tools, each axis is driven separately. In a point-to-point control system, the dimensional information that must be given to the machine tool will be a series of required position of the two slides. Servo systems can be used to move the slides and no attempt is made to move the slide until the cutter has been retracted back.

## (1.2) Contouring systems (Continuous path systems)

Other type of machine tools involves motion of work piece with respect to the cutter while cutting operation is taking place. These machine tools include milling, routing machines etc. and are known as contouring machines as shown in figure 3 (b), 3 (c) and the controls required for their control are known as contouring control. Contouring machines can also be used as point-to-point machines, but it will be uneconomical to use them unless the work piece also requires having a contouring operation to be performed on it. These machines require simultaneous control of axes. In contouring machines, relative positions of the work piece and the tool should be continuously controlled. The control system must be able to accept information regarding velocities and positions of the machines slides. Feed rates should be programmed.

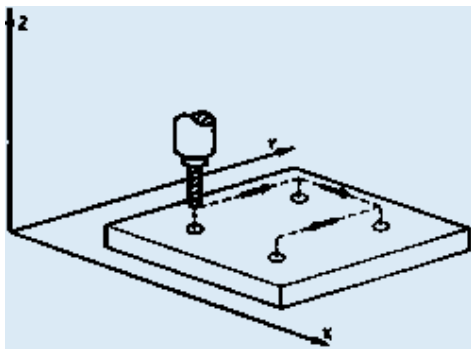


Figure 3 (a) Point-to-point system

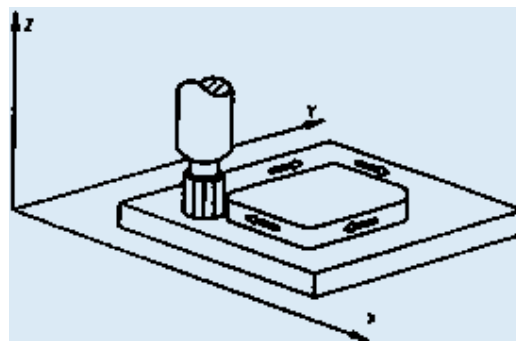


Figure 3 (b) Contouring system

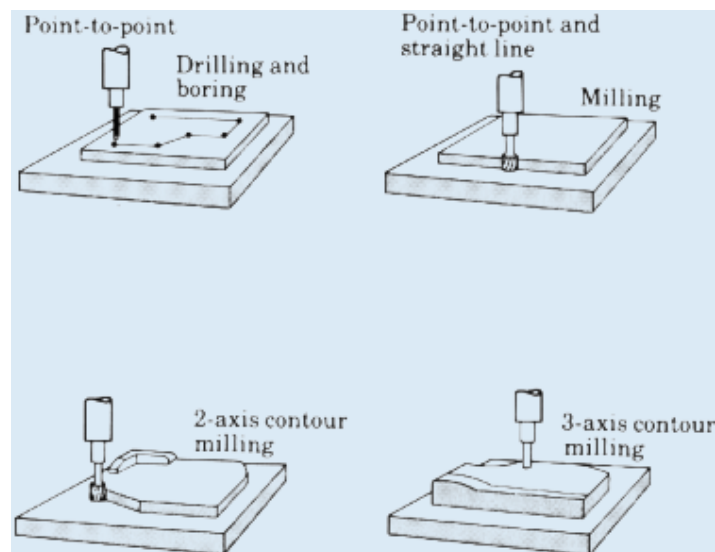


Figure 3 (c) Contouring systems

## (2) Based on the control loops ‘Open loop & Closed loop systems’

### (2.1) Open loop systems (Fig 4(a)):

Programmed instructions are fed into the controller through an input device. These instructions are then converted to electrical pulses (signals) by the controller and sent to the servo amplifier to energize the servo motors. The primary drawback of the open-loop system is that there is no feedback system to check whether the program position and velocity has been achieved. If the system performance is affected by load, temperature, humidity, or lubrication then the actual output could deviate from the desired output. For these reasons the open-loop system is generally used in point-to-point systems where the accuracy requirements are not critical. Very few continuous-path systems utilize open-loop control.

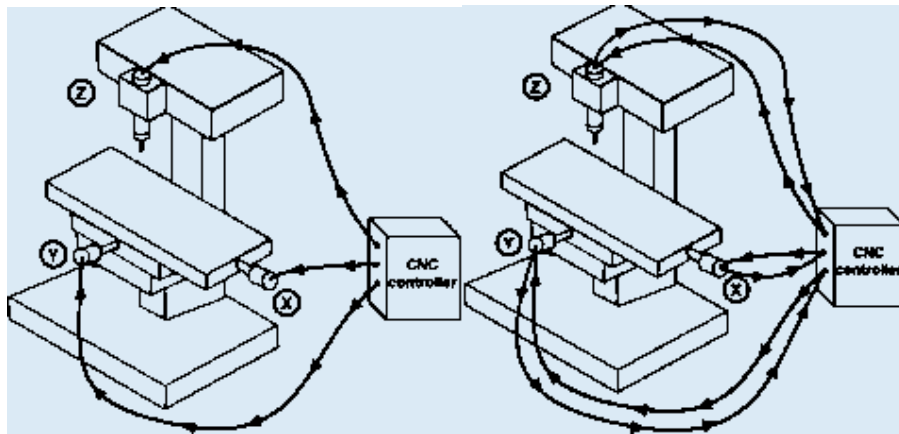


Figure 4(a) Open loop control system Figure 4(b) Closed loop control system

### (2.2) Closed loop systems (Fig 4(b)):

The closed-loop system has a feedback subsystem to monitor the actual output and correct any discrepancy from the programmed input. These systems use position and velocity feedback. The feedback system could be either analog or digital. The analog systems measure the variation of physical variables such as position and velocity in terms of voltage levels. Digital systems monitor output variations by means of electrical pulses. To control the dynamic behavior and the final position of the machine slides, a variety of position transducers are employed. Majority of CNC systems operate on servo mechanism, a closed loop principle. If a discrepancy is revealed between where the machine element should be and where it actually is, the sensing device signals the driving unit to make an adjustment, bringing the movable component to the required location. Closed-loop systems are very powerful and accurate because they are capable of monitoring operating conditions through feedback subsystems and automatically compensating for any variations in real-time.

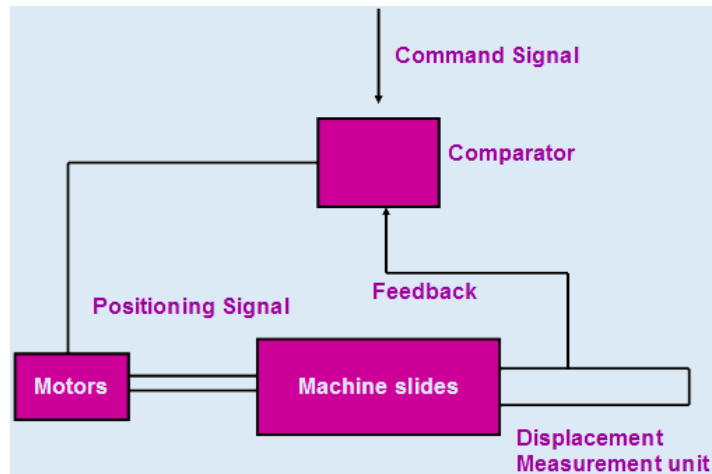


Figure 4 (c) Closed loop system

### (3) Based on the number of axes ‘2, 3, 4 & 5 axes CNC machines’

#### (3.1) 2& 3 axes CNC machines:

CNC lathes will be coming under 2 axes machines. There will be two axes along which motion takes place. The saddle will be moving longitudinally on the bed (Z-axis) and the cross slide moves transversely on the saddle (along X-axis). In 3-axes machines, there will be one more axis, perpendicular to the above two axes. By the simultaneous control of all the 3 axes, complex surfaces can be machined.

#### (3.2) 4 & 5 axes CNC machines (Fig. 5):

4 and 5 axes CNC machines provide multi-axis machining capabilities beyond the standard 3-axis CNC tool path movements. A 5-axis milling centre includes the three X, Y, Z axes, the A axis which is rotary tilting of the spindle and the B-axis, which can be a rotary index table.

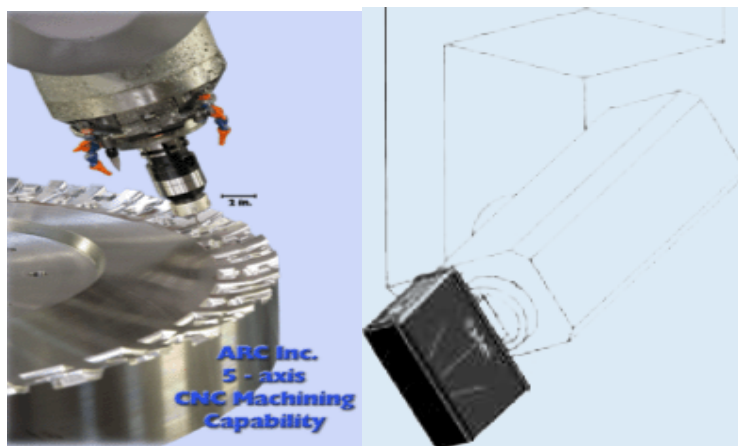


Figure 5: Five axes CNC machine

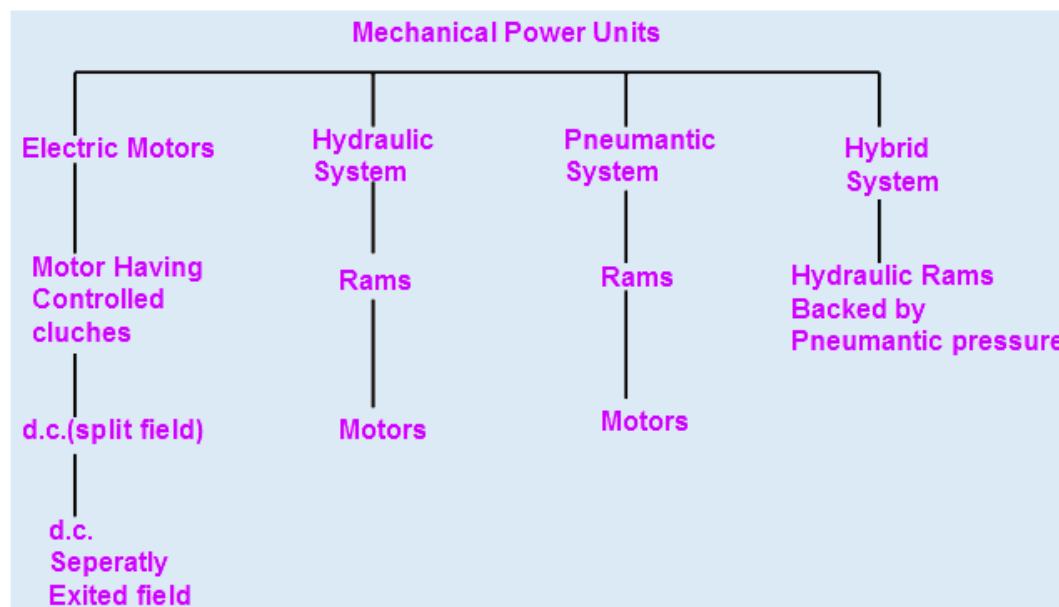
### Importance of higher axes machining:

Reduced cycle time by machining complex components using a single setup. In addition to time savings, improved accuracy can also be achieved as positioning errors between setups are eliminated.

- Improved surface finish and tool life by tilting the tool to maintain optimum tool to part contact all the times.
- Improved access to under cuts and deep pockets. By tilting the tool, the tool can be made normal to the work surface and the errors may be reduced as the major component of cutting force will be along the tool axis.
- Higher axes machining has been widely used for machining sculptures surfaces in aerospace and automobile industry.

### (4) Based on the power supply 'Electric, Hydraulic & Pneumatic systems'

Mechanical power unit refers to a device which transforms some form of energy to mechanical power which may be used for driving slides, saddles or gantries forming a part of machine tool. The input power may be of electrical, hydraulic or pneumatic.



#### (4.1) Electric systems :

Electric motors may be used for controlling both positioning and contouring machines. They may be either a.c. or d.c. motor and the torque and direction of rotation need to be controlled. The speed of a d.c. motor can be controlled by varying either the field or the armature supply. The clutch-controlled motor can either be an a.c. or d.c. motor. They are generally used for small machine tools because of heat losses in the clutches. Split field motors are the simplest form of motors and can be controlled in a manner according to the machine tool. These are small and generally run at high maximum speeds and so require reduction gears of high ratio.



Separately excited motors are used with control systems for driving the slides of large machine tools.

#### **(4.2) Hydraulic systems:**

These hydraulic systems may be used with positioning and contouring machine tools of all sizes. These systems may be either in the form of rams or motors. Hydraulic motors are smaller than electric motors of equivalent power. There are several types of hydraulic motors. The advantage of using hydraulic motors is that they can be very small and have considerable torque. This means that they may be incorporated in servosystems which require having a rapid response.

## **CNC MACHINING CENTERS**

The machining centre, developed in the late 50's is a machine tool capable of multiple machining operations on a work part in one setup under NC program control.

### **Classification**

Machining centres are classified as vertical, horizontal, or universal. The designation refers to the orientation of the machine spindle.

1. A vertical machining centre has its spindle on a vertical axis relative to the work table. A vertical machining centre (VMC) is typically used for flat work that requires tool access from top. E.g. mould and die cavities, Large components of aircraft
2. A horizontal machining centre (HMC) is used for cube shaped parts where tool access can be best achieved on the sides of the cube.
3. A universal machining centre (UMC) has a work head that swivels its spindle axis to any angle between horizontal and vertical making this a very flexible machine tool. E.g.: Aerofoil shapes, Curvilinear geometries.

The term "Multi tasking machine" is used to include all of these machine tools that accomplish multiple and often quite different types of operations. The processes that might be available on a single multi tasking machine include milling, drilling, tapping, grinding and welding. Advantage of this new class of highly versatile machine compared to more conventional CNC machine tools include:

- Fewer steps,
- Reduced part handling,
- Increased accuracy and repeatability because the parts utilize the same fixture through out their processing
- Faster delivery of parts in small lot sizes.

### **Features of CNC machining centers:**

CNC machining centers are usually designed with features to reduce non productive time. The features are:

- **Automatic tool changer :**

The tools are contained in a storage unit that is integrated with the machine tool. When a cutter needs to be changed, the tool drum rotates to the proper position and an automatic tool changer (ATC) operating under program control, exchanges the tool in the spindle for the tool in the tool storage unit. Capacities of tool storage unit commonly range from 16 to 80 cutting tools.

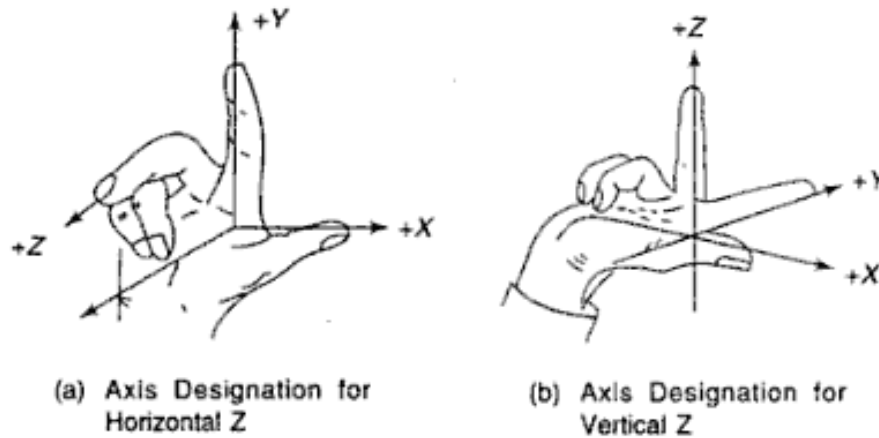
- **Automatic work part positioner:**

Many horizontal and vertical machining centers have the capability to orient the work part relative to the spindle. This is accomplished by means of a rotary table on which work part is fixtured. The table can be oriented at any angle about a vertical axis to permit the cutting tool to access almost the entire surface of the part in a single setup.

- **Automatic pallet changer:**

Machining centers are often equipped with two (or more) separate pallets that can be presented to the cutting tool using an automatic pallet changer. While machining is performed with one pallet in position at the machine, the other pallet is in a safe location away from the spindle. In this location, the operator can unload the finished part and then fixture the raw work part for next cycle.

**Axes Designation in horizontal and vertical machining centres (Fig 1) :**



**Fig 1: Axes Designation in horizontal and vertical machining centres**

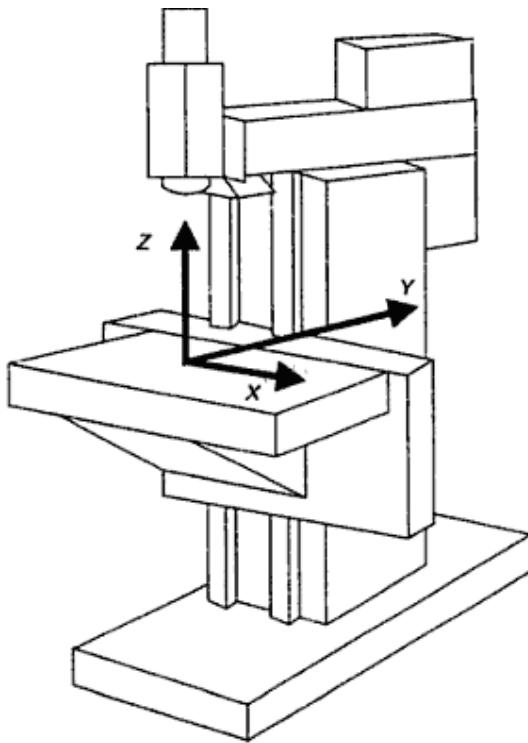


Fig 2 : Vertical Machining Centre

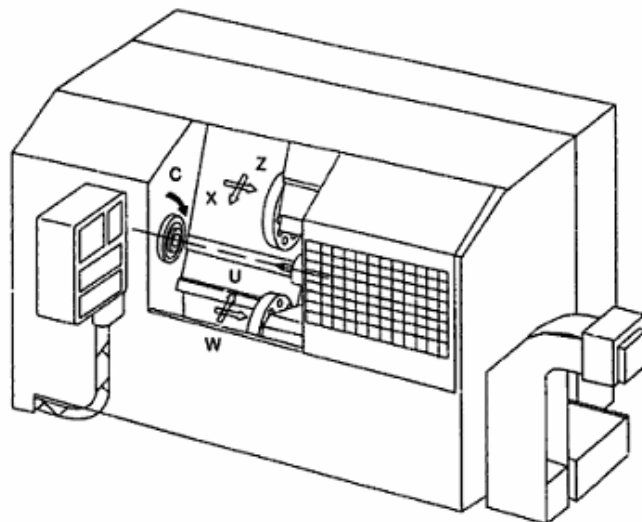


Fig 3 : Horizontal Machining Centre

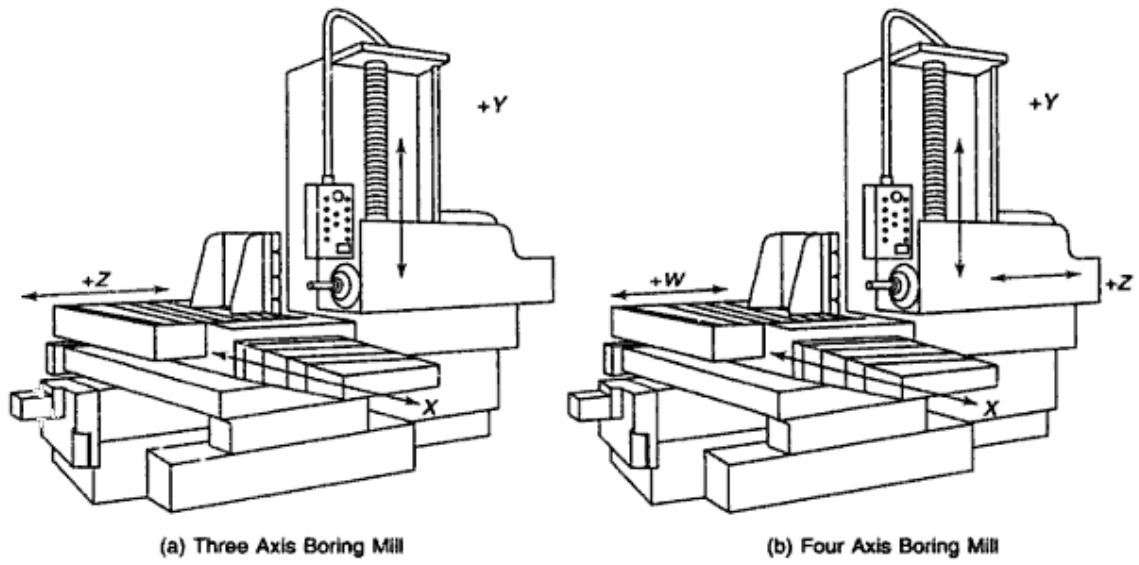


Fig 4 : CNC Horizontal Boring Mills in 3 and 4 axes

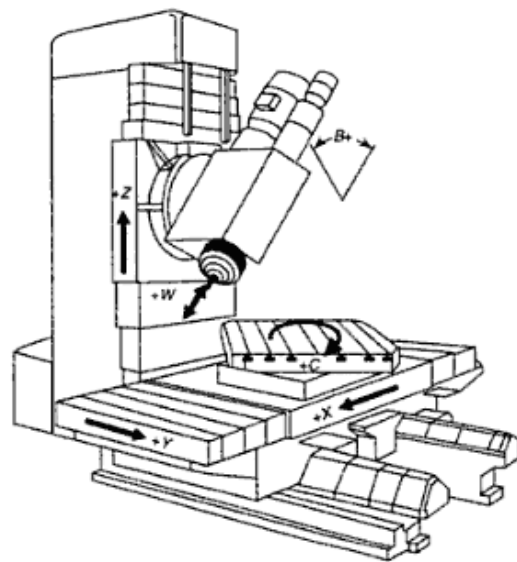
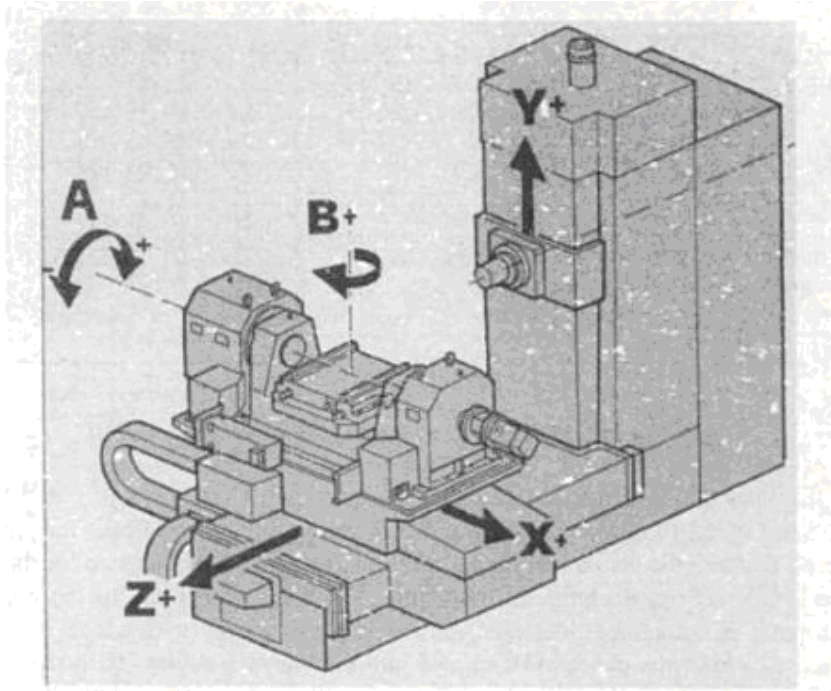
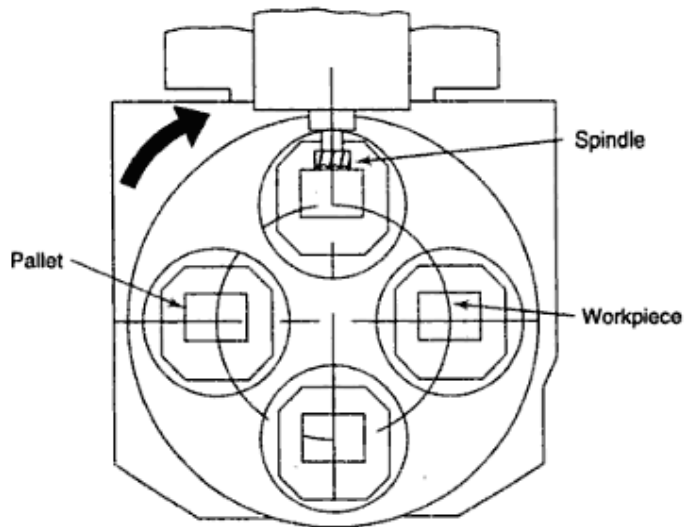


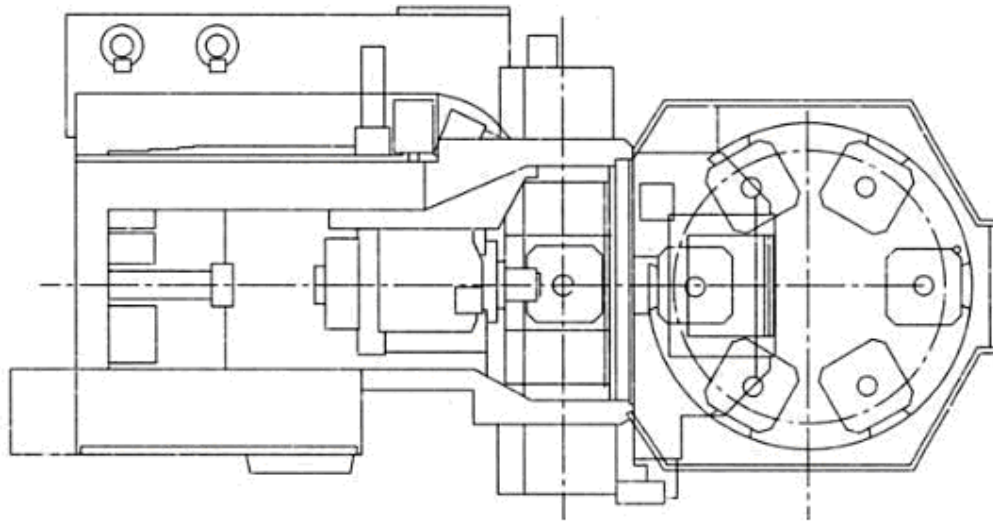
Fig 5 : Five Axes CNC Vertical Axis Machining Centre



**Fig 6 : Five Axes CNC Horizontal Axis Machining Centre**



**Fig 7 : Typical Rotary Type Pallet Changer**



**Fig 8 : Typical Shuttle Pallet Changer with Six Pallet Carousel**

# **CNC PART PROGRAMMING**

## **(1) Programming fundamentals**

Machining involves an important aspect of relative movement between cutting tool and workpiece. In machine tools this is accomplished by either moving the tool with respect to workpiece or vice versa. In order to define relative motion of two objects, reference directions are required to be defined. These reference directions depend on type of machine tool and are defined by considering an imaginary coordinate system on the machine tool. A program defining motion of tool / workpiece in this coordinate system is known as a part program. Lathe and Milling machines are taken for case study but other machine tools like CNC grinding, CNC hobbing, CNC filament winding machine, etc. can also be dealt with in the same manner.

### **(1.1) Reference Point**

Part programming requires establishment of some reference points. Three reference points are either set by manufacturer or user.

#### **a) Machine Origin**

The machine origin is a fixed point set by the machine tool builder. Usually it cannot be changed. Any tool movement is measured from this point. The controller always remembers tool distance from the machine origin.

#### **b) Program Origin**

It is also called home position of the tool. Program origin is point from where the tool starts for its motion while executing a program and returns back at the end of the cycle. This can be any point within the workspace of the tool which is sufficiently away from the part. In case of CNC lathe it is a point where tool change is carried out.

#### **c) Part Origin**

The part origin can be set at any point inside the machine's electronic grid system. Establishing the part origin is also known as zero shift, work shift, floating zero or datum. Usually part origin needs to be defined for each new setup. Zero shifting allows the relocation of the part. Sometimes the part accuracy is affected by the location of the part origin. Figure 1 and 2 shows the reference points on a lathe and milling machine.



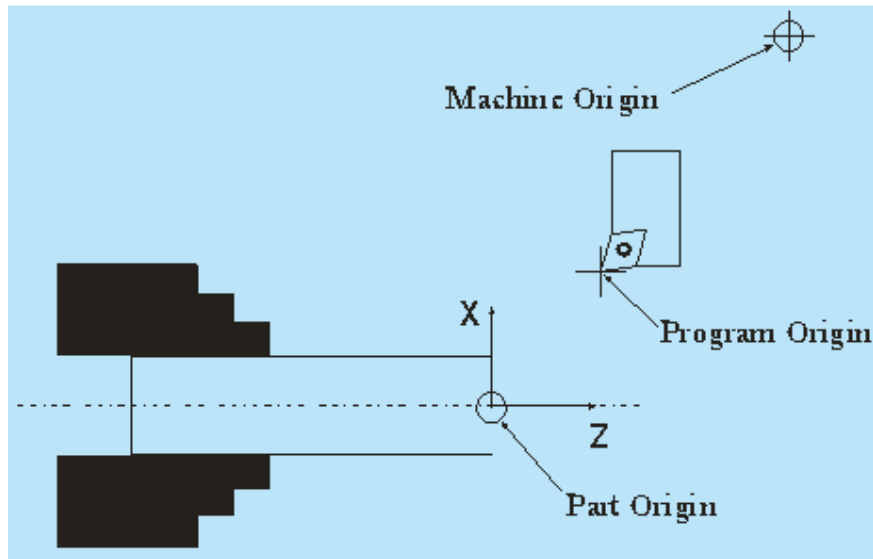


Figure 1 - Reference points and axis on a lathe

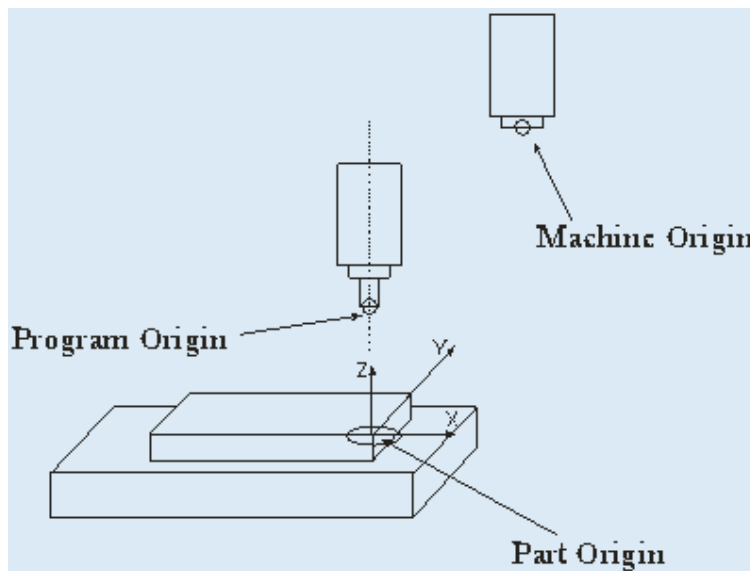


Figure 2 - Reference points and axis on a Milling Machine

## 1.2 )Axis Designation

An object in space can have six degrees of freedom with respect to an imaginary Cartesian coordinate system. Three of them are linear movements and other three are rotary. Machining of simple part does not require all degrees of freedom. With the increase in degrees of freedom, complexity of hardware and programming increases. Number of degree of freedom defines axis of machine.

Axes interpolation means simultaneous movement of two or more different axes generate required contour.

For typical lathe machine degree of freedom is 2 and so it called 2 axis machines. For typical milling machine degree of freedom is  $2^{1/2}$ , which means that two axes can be interpolated at

a time and third remains independent. Typical direction for the lathe and milling machine is as shown in figure 1 and figure 2.

### 1.3 ) Setting up of Origin

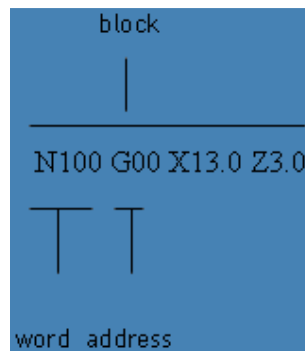
In case of CNC machine tool rotation of the reference axis is not possible. Origin can set by selecting three reference planes X, Y and Z. Planes can be set by touching tool on the surfaces of the workpiece and setting that surfaces as X=x, Y=y and Z=z.

### (1.4 ) Coding Systems

The programmer and the operator must use a coding system to represent information, which the controller can interpret and execute. A frequently used coding system is the Binary-Coded Decimal or BCD system. This system is also known as the EIA Code set because it was developed by Electronics Industries Association. The newer coding system is ASCII and it has become the ISO code set because of its wide acceptance.

### (2) CNC Code Syntax

The CNC machine uses a set of rules to enter, edit, receive and output data. These rules are known as CNC Syntax, Programming format, or tape format. The format specifies the order and arrangement of information entered. This is an area where controls differ widely. There are rules for the maximum and minimum numerical values and word lengths and can be entered, and the arrangement of the characters and word is important. The most common CNC format is the word address format and the other two formats are fixed sequential block address format and tab sequential format, which are obsolete. The instruction block consists of one or more words. A word consists of an address followed by numerals. For the address, one of the letters from A to Z is used. The address defines the meaning of the number that follows. In other words, the address determines what the number stands for. For example it may be an instruction to move the tool along the X axis, or to select a particular tool.



Most controllers allow suppressing the leading zeros when entering data. This is known as leading zero suppression. When this method is used, the machine control reads the numbers from right to left, allowing the zeros to the left of the significant digit to be omitted. Some controls allow entering data without using the trailing zeros. Consequently it is called trailing zero suppression. The machine control reads from left to right, and zeros to the right of the significant digit may be omitted.

### 3) Types of CNC codes

#### (3.1) Preparatory codes

The term "preparatory" in NC means that it "prepares" the control system to be ready for implementing the information that follows in the next block of instructions. A **preparatory function** is designated in a program by the word address G followed by two digits. Preparatory functions are also called **G-codes** and they specify the control mode of the operation.

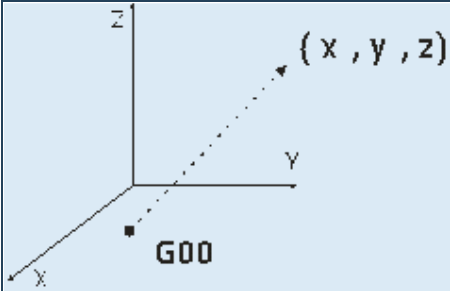
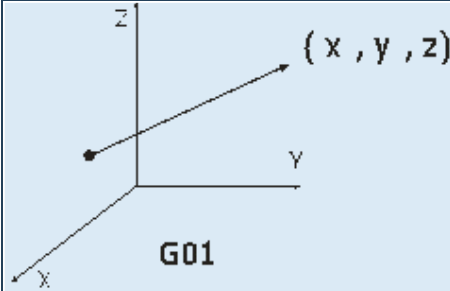
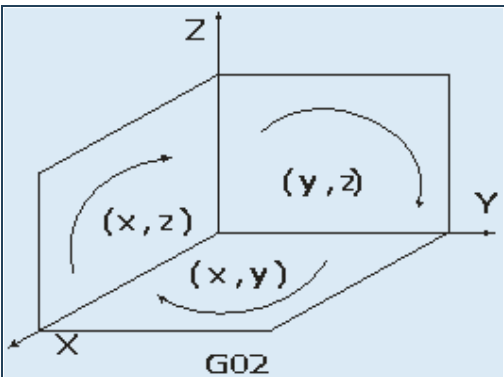
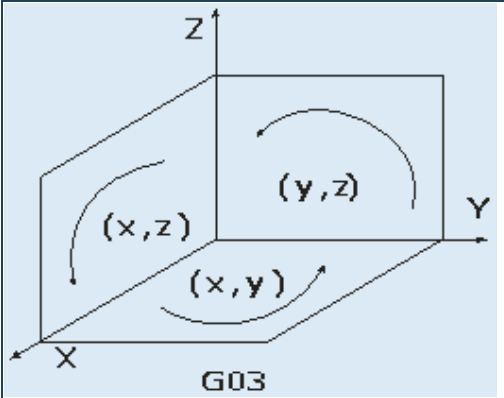
#### (3.2) Miscellaneous codes

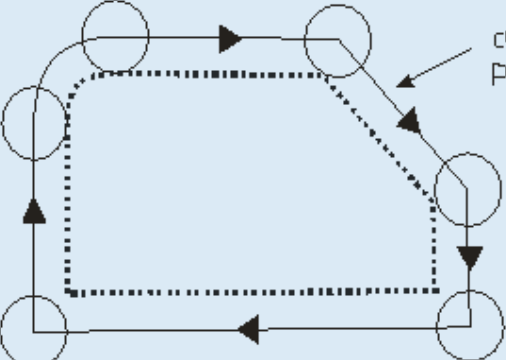
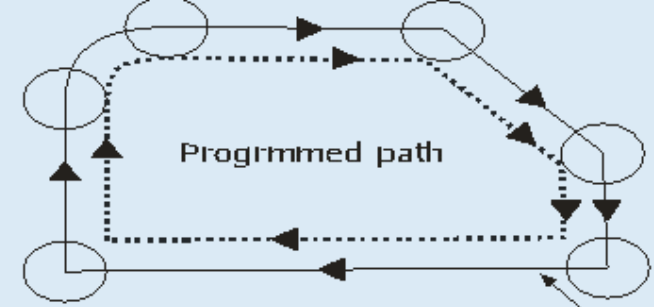
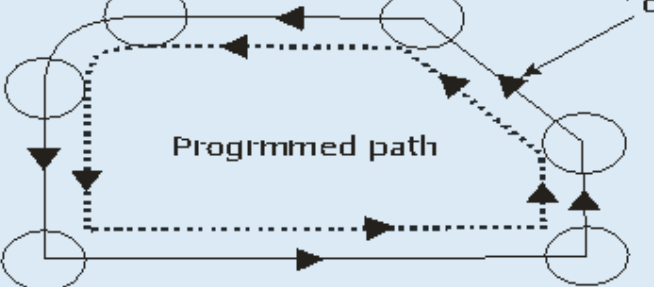
Miscellaneous functions use the address letter M followed by two digits. They perform a group of instructions such as coolant on/off, spindle on/off, tool change, program stop, or program end. They are often referred to as machine functions or **M-functions**. Some of the M codes are given below.

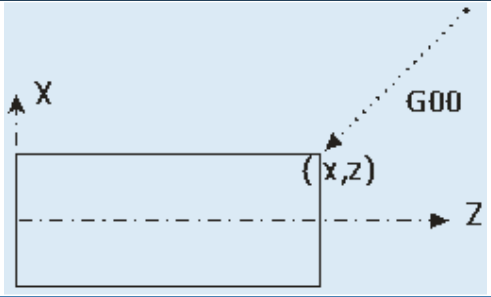
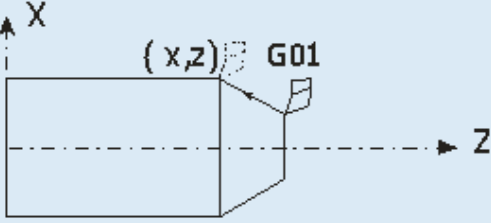
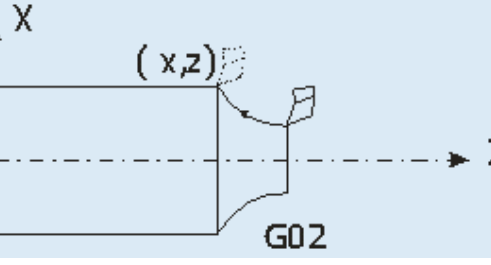
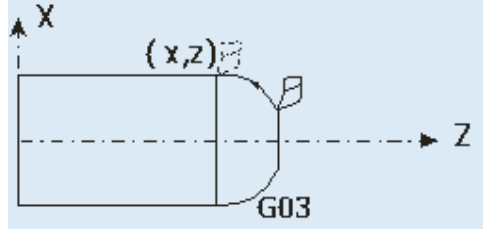
M00 Unconditional stop  
M02 End of program  
M03 Spindle clockwise  
M04 Spindle counterclockwise  
M05 Spindle stop  
M06 Tool change (see Note below)  
M30 End of program

In principle, all codes are either modal or non-modal. **Modal code** stays in effect until cancelled by another code in the same group. The control remembers modal codes. This gives the programmer an opportunity to save programming time. **Non-modal code** stays in effect only for the block in which it is programmed. Afterwards, its function is turned off automatically. For instance G04 is a non-modal code to program a dwell. After one second, which is say, the programmed dwell time in one particular case, this function is cancelled. To perform dwell in the next blocks, this code has to be reprogrammed. The control does not memorize the non-modal code, so it is called as one shot codes. One-shot commands are **non-modal**. Commands known as "canned cycles" (a controller's internal set of preprogrammed subroutines for generating commonly machined features such as internal pockets and drilled holes) are non-modal and only function during the call.

On some older controllers, cutter positioning (axis) commands (e.g., G00, G01, G02, G03, & G04) are non-modal requiring a new positioning command to be entered each time the cutter (or axis) is moved to another location.

Command group	G-code	Function and Command Statement	Illustration
<b>Tool motion</b>	G00	Rapid traverse G00 Xx Yy Zz	
	G01	Linear interpolation G01 Xx Yy Zz Ff	
	G02	Circular Interpolation in clock-wise direction G02 Xx Yy Ii Jj G02 Xx Zz Ii Kk G02 Yy Zz Jj Kk	
	G03	Circular interpolation in counter-clockwise direction G03 Xx Yy Ii Jj G03 Xx Zz Ii Kk G03 Yy Zz Jj Kk	

Command group	G-code	Function and Command Statement	Illustration
<b>Offset and compensation</b>	G40	Cutter diameter compensation cancel	 <p>cutter center path and Programmed path are same</p>
	G41	Cutter diameter compensation left	 <p>Programmed path</p> <p>cutter center</p>
	G42	Cutter diameter compensation right	 <p>Programmed path</p> <p>cutter center</p>

Command group	G-code	Function and Command Statement	Illustration
<b>Tool motion</b>	G00	Rapid traverse G00 Xx Zz	 A 2D coordinate system with X and Z axes. A rectangular block is shown. A dashed line indicates a rapid traverse path from the origin to a point (x,z) on the top-right corner of the block. The label 'G00' is placed near the end of this dashed line.
	G01	Linear interpolation G01 Xx Zz	 A 3D perspective view of a rectangular block. A tool tip is shown moving along a straight line from a point (x,z) on the top surface to a point on the side surface. The label 'G01' is placed near the tool tip.
	G02	Circular Interpolation in clock-wise direction G02 Xx Zz Ii Kk (or) G02 Xx Zz Rr	 A 2D coordinate system with X and Z axes. A rectangular block is shown. A curved arrow indicates a clockwise circular interpolation path from a point (x,z) on the top surface to a point on the side surface. The label 'G02' is placed near the end of the path.
	G03	Circular interpolation in counter-clockwise direction G03 Xx Zz Ii Kk (or) G03 Yy Zz Rr	 A 2D coordinate system with X and Z axes. A rectangular block is shown. A curved arrow indicates a counter-clockwise circular interpolation path from a point (x,z) on the top surface to a point on the side surface. The label 'G03' is placed near the end of the path.

### Illustrative Example Program

A contour illustrated in figure 3 is to be machined using a CNC milling machine. The details of the codes and programs used are given below.

#### Example:

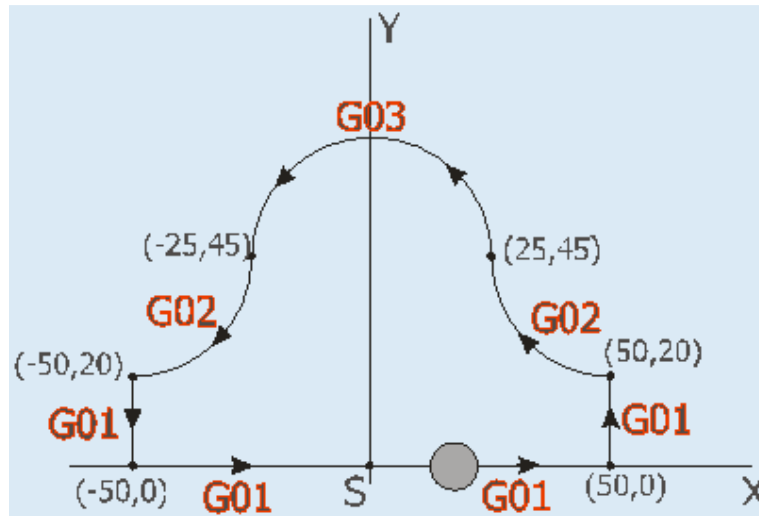


Figure 3 : An illustrative example

<b>O5678</b>	Program number
<b>N02 G21</b>	Metric programming
<b>N03 M03 S1000</b>	Spindle start clockwise with 1000rpm
<b>N04 G00 X0 Y0</b>	Rapid motion towards (0,0)
<b>N05 G00 Z-10.0</b>	Rapid motion towards Z=-10 plane
<b>N06 G01 X50.0</b>	Linear interpolation
<b>N07 G01 Y20.0</b>	Linear interpolation
<b>N08 G02 X25.0 Y45.0 R25.0</b>	Circular interpolation clockwise(cw)
<b>N09 G03 X-25.0 Y45.0 R25.0</b>	Circular interpolation counter clockwise(ccw)
<b>N10 G02 X-50.0 Y20.0 R25.0</b>	Circular interpolation clockwise(cw)
<b>N11 G01 Y0.0</b>	Linear interpolation
<b>N12 G01 X0.0</b>	Linear interpolation
<b>N13 G00 Z10.0</b>	Rapid motion towards Z=10 plane
<b>N14 M05 M09</b>	Spindle stop and program end

## 4. CNC Part Programming II

In the previous section, fundamentals of programming as well basic motion commands for milling and turning have been discussed. This section gives an overview of G codes used for changing the programming mode, applying transformations etc.

### 4.1 Programming modes

Programming mode should be specified when it needs to be changed from absolute to incremental and vice versa. There are two programming modes, absolute and incremental and is discussed below.

#### 4.1.1 Absolute programming (G90)

In absolute programming, all measurements are made from the part origin established by the programmer and set up by the operator. Any programmed coordinate has the absolute value in respect to the absolute coordinate system zero point. The machine control uses the part origin as the reference point in order to position the tool during program execution (Figure 4).

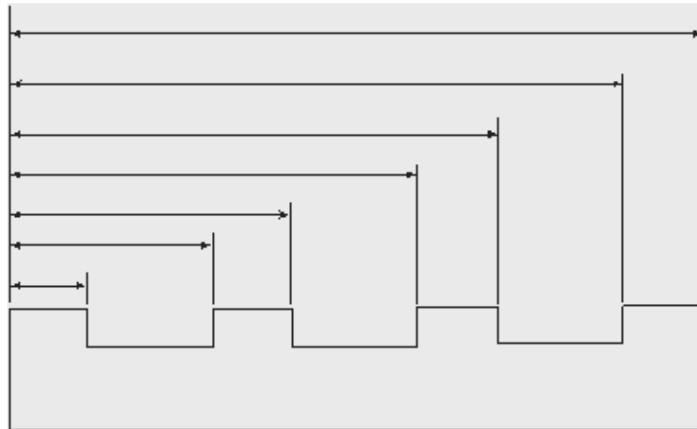


Figure 4 Absolute distances measured from reference zero

#### 4.1.2 Relative programming (G91)

In incremental programming, the tool movement is measured from the last tool position. The programmed movement is based on the change in position between two successive points. The coordinate value is always incremented according to the preceding tool location. The programmer enters the relative distance between current location and the next point ( Figure 5).

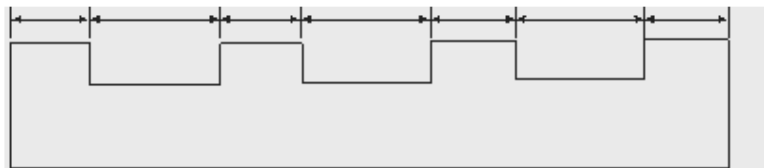


Figure 5 Incremental distances measured from previous locations



## **4.2 Spindle control**

The spindle speed is programmed by the letter 'S' followed by four digit number, such as S1000. There are two ways to define speed :

1. Revolutions per minute (RPM)
2. Constant surface speed

The spindle speed in revolutions per minute is also known as constant rpm or direct rpm. The change in tool position does not affect the rpm commanded. It means that the spindle RPM will remain constant until another RPM is programmed. Constant surface speed is almost exclusively used on lathes. The RPM changes according to diameter being cut. The smaller the diameter, the more RPM is achieved; the bigger the diameter, the less RPM is commanded. This is changed automatically by the machine speed control unit while the tool is changing positions. This is the reason that, this spindle speed mode is known as diameter speed.

## **4.3 Tool selection**

Tool selection is accomplished using 'T' function followed by a four digit number where, first two digits are used to call the particular tool and last two digits are used to represent tool offset in the program. The tool offset is used to correct the values entered in the coordinate system preset block. This can be done quickly on the machine without actually changing the values in the program.

Using the tool offsets, it is easy to set up the tools and to make adjustments

## **4.4 Feed rate control**

Cutting operations may be programmed using two basic feed rate modes:

1. Feed rate per spindle revolution
2. Feed rate per time

The feed rate per spindle revolution depends on the RPM programmed.

## **5.0 Tool radius Compensation**

The programmed point on the part is the command point. It is the destination point of the tool. The point on the tool that is used for programming is the tool reference point. These points may or may not coincide, depending on the type of tool used and machining operation being performed. When drilling, tapping, reaming, countersinking or boring on the machining center, the tool is programmed to the position of the hole or bore center - this is the command point.

When milling a contour, the tool radius center is used as the reference point on the tool while writing the program, but the part is actually cut by the point on the cutter periphery. This point is at 'r' distance from the tool center. This means that the programmer should shift the tool center away from the part in order to perform the cutting by the tool cutting edge. The

shift amount depends upon the part geometry and tool radius. This technique is known as tool radius compensation or cutter radius compensation.

In case of machining with a single point cutting tool, the nose radius of the tool tip is required to be accounted for, as programs are being written assuming zero nose radius. The tool nose radius center is not only the reference point that can be used for programming contours. On the tool there is a point known as imaginary tool tip, which is at the intersection of the lines tangent to the tool nose radius.

Cutter compensation allows programming the geometry and not the toolpath. It also allows adjusting the size of the part, based on the tool radius used to cut part. This is useful when cutter of the proper diameter is not found. This is best explained in the Figure 11.

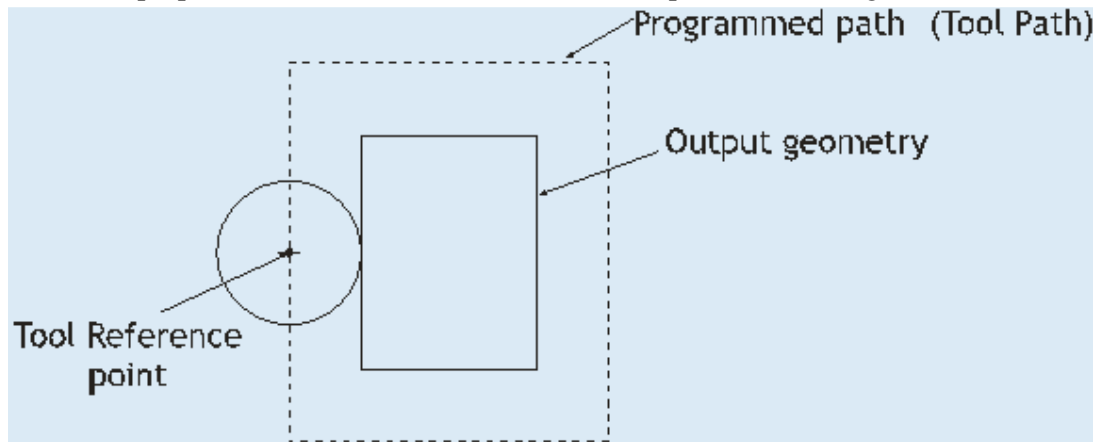


Figure 11. Cutter diameter compensation

The information on the diameter of the tool, which the control system uses to calculate the required compensation, must be input into the control unit's memory before the operation. Tool diameter compensation is activated by the relevant preparatory functions (G codes) as shown in Figure 12.

Compensation for tool radius can be of either right or left side compensation. This can be determined by direction of tool motion. If you are on the tool path facing direction of tool path and if tool is on your left and workpiece is on your right side then use G41 (left side compensation). For, reverse use other code G42 (Right side compensation). Both the codes are modal in nature and remain active in the program until it is cancelled by using another code, G40.

## 5.1 Subroutines

Any frequently programmed order of instruction or unchanging sequences can benefit by becoming a subprogram. Typical applications for subprogram applications in CNC programming are :

- Repetitive machining motions
- Functions relating to tool change
- Hole patterns
- Grooves and threads
- Machine warm-up routines
- Pallet changing
- Special functions and others

Structurally, subprograms are similar to standard programs. They use the same syntax rules. The benefits of subroutines involve the reduction in length of program, and reduction in program errors. There is a definition statement and subroutine call function.

Standard sub-routine

```
N10  
N20  
N30  
....  
N70 G22 N5  
N80  
N90  
....  
N100 G24  
....  
N160 G20 N5
```

In the above example G22 statement defines the start block of the sub-routine and G24 marks the end of the sub-routine statement. The subroutine is called by another code G20 identified by the label N5.

Parametric subroutine

```
..  
..  
G23 N18  
G01 X P0 Y P1  
..  
..  
G21 N18 P0=k10 P1=k20
```

In the above example G23 starts the subprogram label and starts the definition, and the parameters P0, P1 are defined for values of x and y. The G21 statement is used to call the subroutine and to assign the values to the parameters.

## 5.2 Canned Cycles

A canned cycle is a preprogrammed sequence of events / motions of tool / spindle stored in memory of controller. Every canned cycle has a format. Canned cycle is modal in nature and remains activated until cancelled. Canned cycles are a great resource to make manual programming easier. Often underutilized, canned cycles save time and effort.

### 5.2.1 Machining a Rectangular pocket

This cycle assumes the cutter is initially placed over the center of the pocket and at some clearance distance (typically 0.100 inch) above the top of the pocket. Then the cycle will take over from that point, plunging the cutter down to the "peck depth" and feeding the cutter around the pocket in ever increasing increments until the final size is attained. The process is repeated until the desired total depth is attained. Then the cutter is returned to the center of the pocket at the clearance height as shown in figure 14

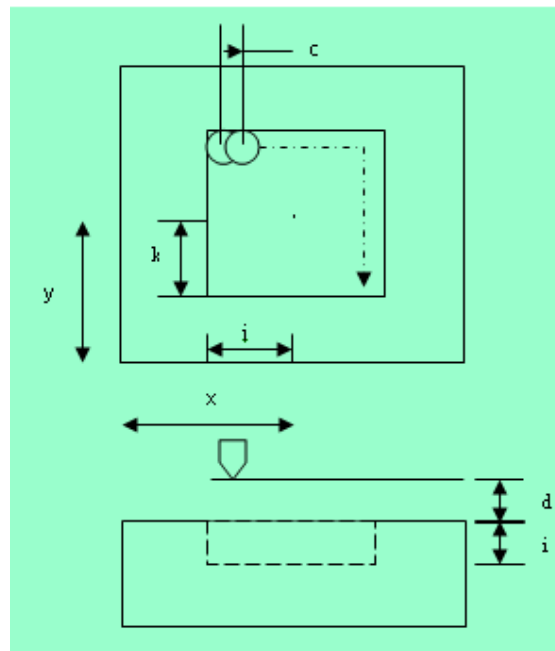


Figure 14. Pocket machining

The overall length and width of the pocket, rather than the distance of cutter motion, are programmed into this cycle.

The syntax is : G87 Xx Yy Zz Ii Jj Kk Bb Cc Dd Hh Ll Ss (This g code is entirely controller specific and the syntax may vary between controller to controller).

Description:

x,y - Center of the part

z - Distance of the reference plane from top of part

i - Pocket depth

j,k - Half dimensions of the target geometry (pocket)

b - Step depth

c - Step over

d - Distance of the reference plane from top of part

h - Feed for finish pass

l - Finishing allowance

s - Speed

For machining a circular pocket, the same syntax with code G88 is used

### Common G-Words:

**TABLE A72** Common G-words (Preparatory Word)

<i>G-word</i>	<i>Function</i>
G00	Point-to-point movement (rapid traverse) between previous point and endpoint defined in current block. Block must include x-y-z coordinates of end position.
G01	Linear interpolation movement. Block must include x-y-z coordinates of end position. Feed rate must also be specified.
G02	Circular interpolation, clockwise. Block must include either arc radius or arc center; coordinates of end position must also be specified.
G03	Circular interpolation, counterclockwise. Block must include either arc radius or arc center; coordinates of end position must also be specified.
G04	Dwell for a specified time.
G10	Input of cutter offset data, followed by a P-code and an R-code.
G17	Selection of x-y plane in milling.
G18	Selection of x-z plane in milling.
G19	Selection of y-z plane in milling.
G20	Input values specified in inches.
G21	Input values specified in millimeters.
G28	Return to reference point.
G32	Thread cutting in turning.

G40	Cancel offset compensation for cutter radius (nose radius in turning).
G41	Cutter offset compensation, left of part surface. Cutter radius (nose radius in turning) must be specified in block.
G42	Cutter offset compensation, right of part surface. Cutter radius (nose radius in turning) must be specified in block.
G50	Specify location of coordinate axis system origin relative to starting location of cutting tool. Used in some lathes. Milling and drilling machines use G92.
G90	Programming in absolute coordinates.
G91	Programming in incremental coordinates.
G92	Specify location of coordinate axis system origin relative to starting location of cutting tool. Used in milling and drilling machines and some lathes. Other lathes use G50.
G94	Specify feed per minute in milling and drilling.
G95	Specify feed per revolution in milling and drilling.
G98	Specify feed per minute in turning.
G99	Specify feed per revolution in turning.

Note: Some G-words apply to milling and/or drilling only, whereas others apply to turning only.

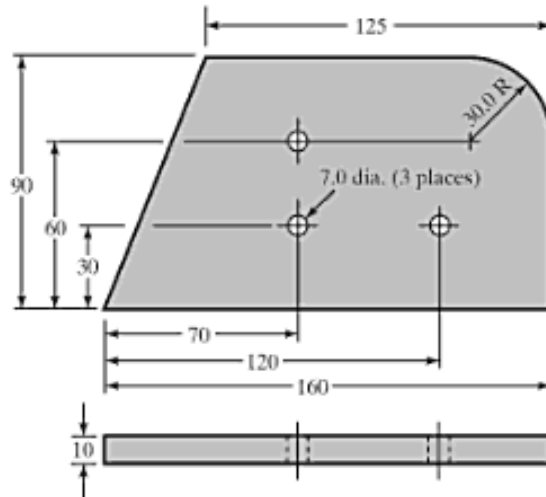
## Common M-Words

**TABLE A7.3** Common M-words Used in Word Address Format

<i>M-Word</i>	<i>Function</i>
M00	Program stop; used in middle of program. Operator must restart machine.
M01	Optional program stop; active only when optional stop button on control panel has been depressed.
M02	End of program. Machine stop.
M03	Start spindle in clockwise direction for milling machine (forward for turning machine).
M04	Start spindle in counterclockwise direction for milling machine (reverse for turning machine).
M05	Spindle stop.
M06	Execute tool change, either manually or automatically. If manually, operator must restart machine. Does not include selection of tool, which is done by T-word if automatic, by operator if manual.
M07	Turn cutting fluid on flood.
M08	Turn cutting fluid on mist.
M09	Turn cutting fluid off.
M10	Automatic clamping of fixture, machine slides, etc.
M11	Automatic unclamping.
M13	Start spindle in clockwise direction for milling machine (forward for turning machine) and turn on cutting fluid.
M14	Start spindle in counterclockwise direction for milling machine (reverse for turning machine) and turn on cutting fluid.
M17	Spindle and cutting fluid off.
M19	Turn spindle off at oriented position.
M30	End of program. Machine stop. Rewind tape (on tape-controlled machines).

## Part programming for vertical machining centres

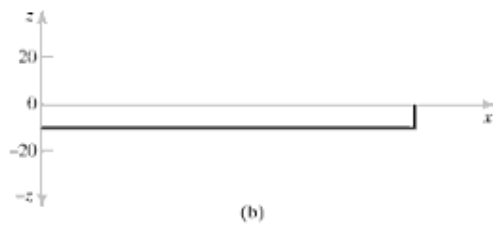
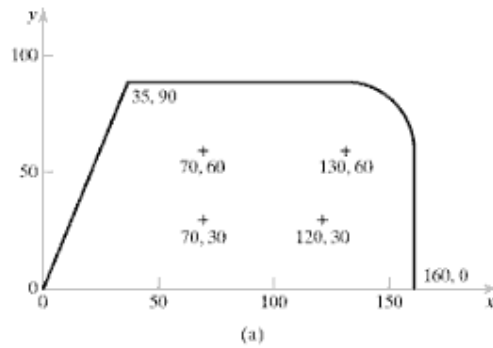
### Part programming example 1:



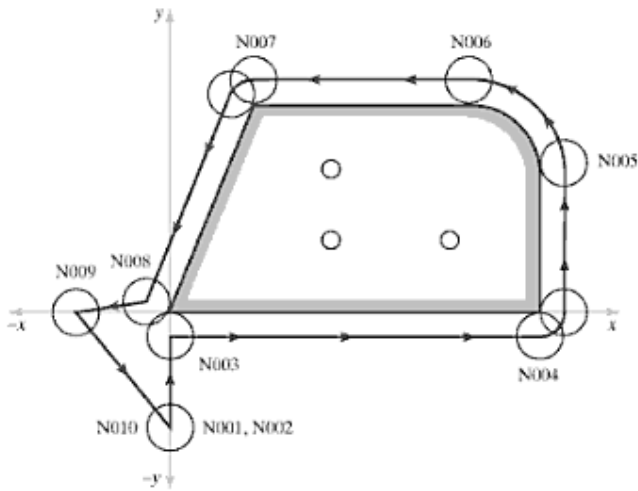
**Figure** Sample part to illustrate NC part programming. Dimensions are in millimeters. General tolerance =  $\pm 0.1$  mm. Work material is a machinable grade of aluminum.

NC Part Program Code	Comments
N001 G21 G90 G92 X0 Y-050.0 Z010.0;	Define origin of axes.
N002 G00 X070.0 Y030.0;	Rapid move to first hole location.
N003 G01 G95 Z-15.0 F0.05 S1000 M03;	Drill first hole.
N004 G01 Z010.0;	Retract drill from hole.
N005 G00 Y060.0;	Rapid move to second hole location.
N006 G01 G95 Z-15.0 F0.05;	Drill second hole.
N007 G01 Z010.0;	Retract drill from hole.
N008 G00 X120.0 Y030.0;	Rapid move to third hole location.
N009 G01 G95 Z-15.0 F0.05;	Drill third hole.
N010 G01 Z010.0;	Retract drill from hole.
N011 G00 X0 Y-050.0 M05;	Rapid move to target point.
N012 M30;	End of program, stop machine.

## Part programming example 2:



Sample part aligned relative to (a) x- and y-axes, and (b) z-axis. Coordinates are given for significant part features in (a).

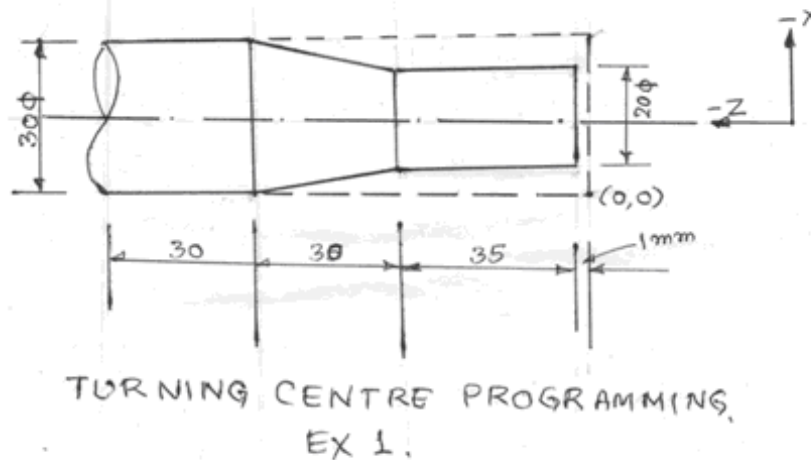




<b>NC Part Program Code</b>	<b>Comments</b>
N001 G21 G90 G92 X0 Y-050.0 Z010.0;	Define origin of axes.
N002 G00 Z-025.0 S1000 M03;	Rapid move to cutter depth, turn spindle on.
N003 G01 G94 G42 Y0 D05 F40;	Engage part, start cutter offset.
N004 G01 X160.0;	Mill lower part edge.
N005 G01 Y060.0;	Mill right straight edge.
N006 G17 G03 X130.0 Y090.0 R030.0;	Circular interpolation around arc.
N007 G01 X035.0;	Mill upper part edge.
N008 G01 X0 Y0;	Mill left part edge.
N009 G40 G00 X-040.0 M05;	Rapid exit from part, cancel offset.
N010 G00 X0 Y-050.0;	Rapid move to target point.
N011 M30;	End of program, stop machine.

## Programming on turning centre

### Example 1:

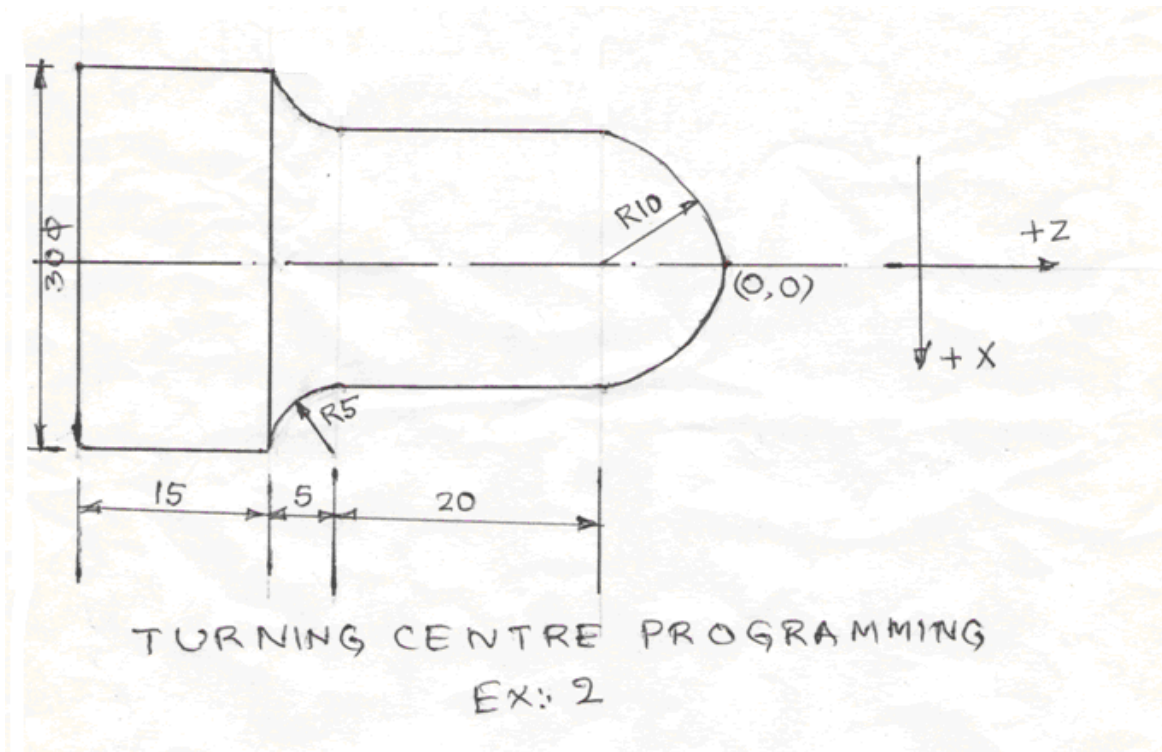


N01 G91 G71 M03 S800	(incremental mode, metric, spindle start with a speed of 800 rpm)
N02 G00 X1.0	(tool away from work piece 1mm, rapid)
N03 G00 Z-1.0	(tool to left 1mm for facing)
N04 G01 X-16.0 F200	(facing cut at a feed rate 200mm/min)
N05 G00 Z1.0	(move tool to right 1mm from that position)
N06 G00 X10.0	(move 10 mm away from the centre in x direction)
N07 G01 Z-36.0	(plane turning over a length of 35 mm)
N08 G01 X5.0 Z-30.0	(simultaneous movement in X and Z directions for taper turning)
N09 G00 X1.0 Z66.0	(move to the starting position)
N10 M02	(end of program)

Note:

- I is the X offset is defined as the distance from the beginning of the arc to the centre of the arc in the X-direction
- K is the Z offset is defined as the distance from the beginning of the arc to the centre of the arc the Z direction

**Example 2:**



N01 G90 G71 M03 S800

(absolute, metric, start spindle at a speed 800 rpm)

N02 G00 X0.0 Y0.0

(move tool to the beginning of cut)

N03 G02 X10.0 Z-10.0 I 0.0 K-10.0 F150

(clockwise circular interpolation, I&K are offsets, feed rate 150mm/min)

N04 G00 Z-30.0

(move from point B to point C)

N05 G03 X15.0 Z-35.0 I5.0 K0.0

(counterclockwise interpolation, I&K are offsets, move to point D)

N06 M02

(end of program)

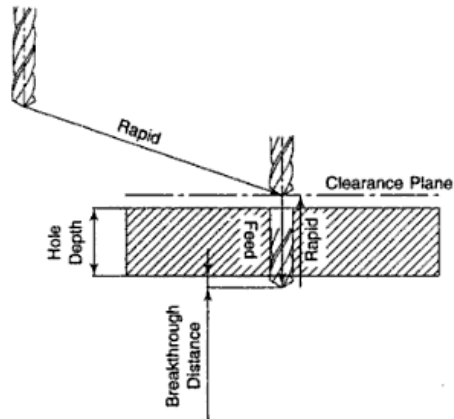
Note:

- I is the X offset is defined as the distance from the beginning of the arc to the centre of the arc in the X-direction
- K is the Z offset is defined as the distance from the beginning of the arc to the centre of the arc the Z direction

**Program without the use of canned cycles:**

```

N01 G00 X25.0 Y35.0 Z2.0 *
N02 G01 Z-18.0 F125 *
N03 G00 Z2.0 *
N04 X55.0 Y50.0 *
N05 G01 Z-18.0 F125 *
N06 G00 Z2.0 *
N07 X75.0 Y70.0 *
N08 G00 Z2.0 *
N09 X0 Y0 Z50.0 *
    
```

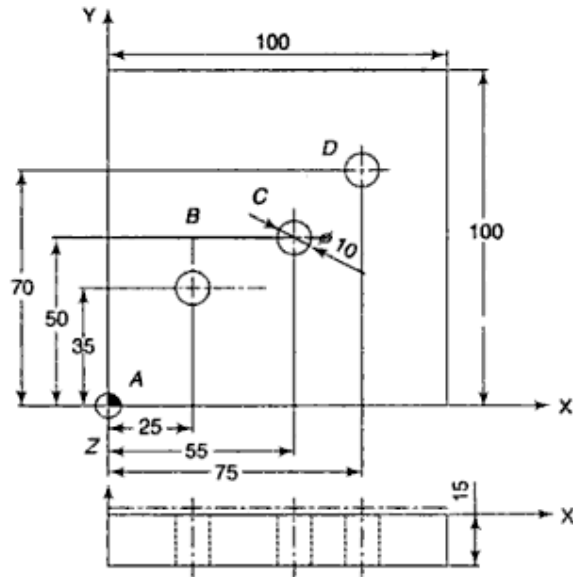


**Fig.** Typical Motions Embedded in G81 Canned Cycle

**Program using canned cycles:**

```

N01 G81 X25.0 Y35.0 Z-18.0
R2.0 F125 *
N02 X55.0 Y50.0 *
N03 X75.0 Y70.0 *
N04 G80 X0 Y0 Z50.0 *
    
```



**Fig.** Example for Canned Cycles

## **PART PROGRAMMING WITH APT**

APT is an acronym that stands for Automatically Programmed Tooling. It is a three dimensional NC part programming system that was developed in the late 1950s and early 1960s. Today it remains an important language in the United States and around the world, and most of the CAD/CAM approaches to part programming are based on APT. APT is also important because many of the concepts incorporated into it formed the basis for other subsequently developed systems in interactive graphics. APT was originally intended as a contouring language, but modern versions can be used for both point-to-point and contouring operations in upto five axes. Our discussion will be limited to the three linear axes, x, y and z. APT can be used for a variety of machining operations. Our coverage will concentrate on drilling (point-to-point) and milling (contouring) operations. There are more than 500 words in the APT vocabulary. Only a small (but important) fraction of the total lexicon will be covered here.

APT is not a language; it is also the computer program that processes the APT statements to calculate the corresponding cutter positions and generate the machine tool control commands. To program in APT, the programmer must first define the part geometry. Then the tool is directed to various point locations and along surfaces of the workpart to accomplish the required machining operations. The viewpoint of the programmer is that the workpiece remains stationary, and the tool is instructed to move relative to the part. To complete the program, speeds and feeds must be specific, tools must be called, tolerances must be given for circular interpolation, and so forth. Thus, there are four basic types of statements in the APT language.

1. **Geometry statements** are used to define the geometry elements that comprise the part.
2. **Motion commands** are used to specify the tool path
3. **Postprocessor statements** control the machine tool operation, for example, to specify speeds and feeds, set tolerance values for circular interpolation, and actuate other capabilities of the machine tool.
4. **Auxiliary statements** are a group of miscellaneous statements used to name the part program, insert comments in the program, and accomplish similar functions.

These statements are constructed of APT vocabulary words, symbols, and numbers, all arranged using appropriate punctuation. APT vocabulary words consist of six or fewer characters. Such a restriction seems archaic today, but it must be remembered that APT was developed in the 1950s, when computer memory technology was extremely limited. Most APT statements include a slash (/) as part of the punctuation. APT vocabulary words that immediately precede the slash are called **major words**, whereas those that follow the slash are called **minor words**.

## APT Geometry Statements

The geometry of the part must be defined to identify the surfaces and features that are to be machined. Accordingly, the points, lines, and surfaces must be defined in the program prior to specifying the motion statements. The general form of an APT geometry statements is the following:

$$\text{SYMBOL} = \text{GEOMETRY TYPE}/\text{descriptive data}$$

An example of such a statement is

$$P1 = \text{POINT}/20.0, 40.0, 60.0$$

An APT geometry statement consists of three sections. The first is the symbol used to identify the geometry element. A symbol can be any combination of six or fewer alphabetical and numerical characters, at least one of which must be alphabetical. Also, the symbol cannot be an APT vocabulary word. The second section of the APT geometry statement is an APT major word that identifies the type of geometry element. Examples are POINT, LINE, CIRCLE and PLANE. The third section of the APT geometry statement provides the descriptive data that define the element precisely, completely, and uniquely. These data may include numerical values to specify dimensional and position data, previously defined geometry elements, and APT minor words.

Punctuation in an APT geometry statement is indicated in the preceding geometry statements. The geometry definition is written as an equation, the symbol being equated to the element type, followed by a slash with descriptive data to the right of the slash. Commas are used to separate the words and numerical values in the descriptive data. There are a variety of ways to specify geometry elements. In the following discussion, examples of APT statements will be presented for points, lines, planes, and circles.

### **Points :**

Specification of a point is most easily accomplished by designating its x-,y-, and z-coordinates.

$$P1 = \text{POINT}/20.0, 40.0, 60.0$$

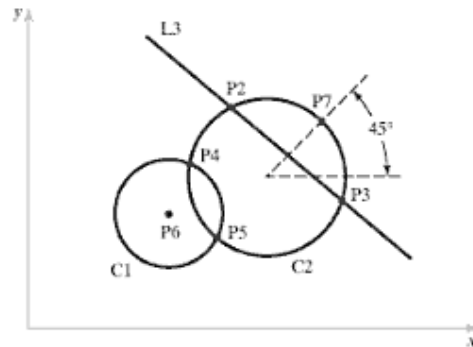
where the descriptive data following the slash indicate x-,y-, and z-coordinates. The specification can be done in either inches or millimeters (metric). We use metric values in our examples. As an alternative, a point can be defined as the intersection of two intersecting lines, as in the following:

$$P1 = \text{POINT}/\text{INTOF}, L1, L2$$

where the APT word INTOF in the descriptive data stands for “intersection of”.

Other methods of defining points are also available. Several are illustrated in Figure 1. The associated points are identified in the following APT statements:

P2= POINT/YLARGE, INTOF, L3, C2  
 P2= POINT/XSMALL, INTOF, L3, C2  
 P3= POINT/XLARGE, INTOF, L3, C2  
 P3= POINT/YSMALL, INTOF, L3, C2  
 P4= POINT/YLARGE, INTOF, C1, C2  
 P5= POINT/YSMALL, INTOF, C1, C2  
 P6= POINT/CENTER, C1  
 P7= POINT/C2, ATANGL, 45

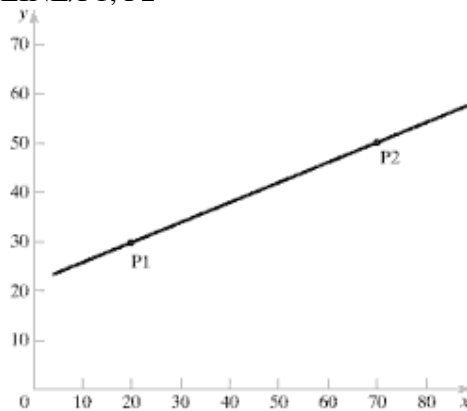


**Figure 1** Defining a point using intersections of previously defined lines and circles.

### Lines :

A line defined in APT is considered to be of infinite length in both directions. Also, APT treats a line as a vertical plane that is perpendicular to the x-y plane. The easiest way to specify a line is by two points through which it passes, as in Figure 2:

L1= LINE/P1, P2



**Figure 2** Defining a line using two previously defined points.

The same line can be defined by indicating the coordinate positions of the two points by giving their x-,y-, and z-coordinates in sequence; for example,

L1= LINE/20, 30, 0, 70, 50, 0

In some situations, the part programmer may find it more convenient to define a new line as being parallel to or perpendicular to one of the axes or another line that has been previously defined; for example, with reference to Figure 3,

L5= LINE/P2, PARLEL, L3  
L6= LINE/P2, PERPTO, L3  
L7= LINE/P2, PERPTO, XAXIS

where PARLEL and PERPTO are APT's way of spelling "parallel to" and "perpendicular to", respectively.

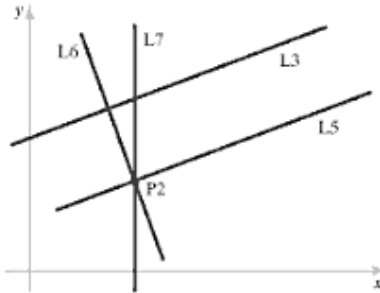


Figure 3 : Defining a line using a point and parallelism or perpendicularity to another line

Lines can also be defined in relation to a point and a circle, as in Figure 4, as in the geometry statements

L1= LINE/P1, LEFT, TANTO, C1  
L2= LINE/P1, RIGHT, TANTO, C1

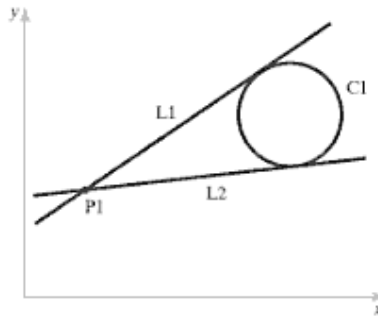
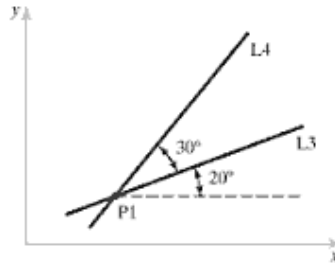


Figure 4 Defining a line using a point and a circle.

where the words LEFT and RIGHT are used by looking in the direction of the circle from the point P1, and TANTO means "tangent to".





**Figure 5** Defining a line using a point and the x-axis or another

Finally, lines can be defined using a point and the angle of the line relative to the x-axis or some other line, as in Figure B7.5. The following statements illustrate the definitions:

```
L3= LINE/P1, ATANGL, 20, XAXIS
L4= LINE/P1, ATANGL, 30, L3
```

### Planes :

A plane can be defined by specifying three points through which the plane passes, as in the following:

```
PL1= PLANE/P1, P2, P3
```

Of course, the three points must be non-collinear. A plane can also be defined as being parallel to another plane that has been previously defined; for instance,

```
PL2= PLANE/P2, PARLEL, PL1
```

which states that plane PL2 passes through point P2 and is parallel to plane PL1. In APT, a plane extends indefinitely.

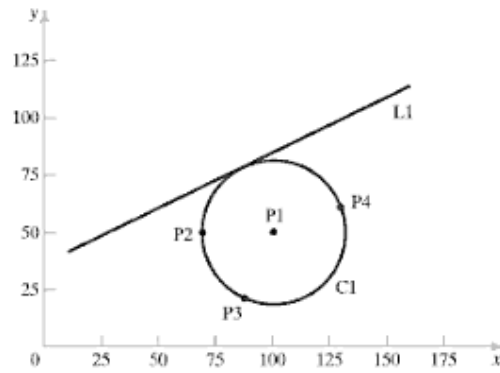
### Circles :

In APT, a circle is considered to be a cylindrical surface that is perpendicular to the x-y plane and extends to infinity in the z-direction. The easiest way to define a circle is by its center and radius, as in the following two statements, illustrated in Figure 6.

```
C1= CIRCLE/CENTER, P1, RADIUS, 32
C1= CIRCLE/CENTER, 100, 50, 0, RADIUS, 32
```

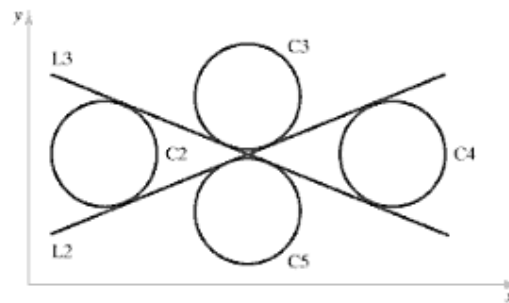
Two additional ways of defining a circle utilize previously defined points P2, P3, and P4, or line L1 in the same figure:

```
C1= CIRCLE/CENTER, P2, P3, P4 (P2, P3 and P4 must not be collinear)
C1= CIRCLE/CENTER, P1, TANTO, L1
```



**Figure 6** Defining a circle.

Other ways to define circles make use of existing lines L2 and L3 in Figure 7. The statements for the four circles in the figure are the following:



**Figure 7** Defining a circle using two intersecting lines.

C2= CIRCLE/XSMALL, L2, YSMALL, L3, RADIUS, 25  
 C3= CIRCLE/YLARGE, L2, YLARGE, L3, RADIUS, 25  
 C4= CIRCLE/XLARGE, L2, YLARGE, L3, RADIUS, 25  
 C5= CIRCLE/YSMALL, L2, YSMALL, L3, RADIUS, 25

**Ground Rules :**

Certain ground rules must be obeyed when formulating APT geometry statement. Following are four important rules in APT:

1. Coordinate data must be specified in the order x, then y, then z, because the statement

$$P1=POINT/20.5, 40.0, 60.0$$

is interpreted to mean x = 20.5 mm, y = 40.0 mm, and z = 60.0 mm

2. Any symbols used as descriptive data must have been previously defined: for example, in the statement

$$P1=POINT/INTOF, L1, L2$$

the two lines L1 and L2 must have been previously defined. In setting up the list of geometry statements, the APT programmer must be sure to define symbols before using them in subsequent statements.

3. A symbol can be used to define only one geometry element. The same symbol cannot be used to define two different elements. For example, the following statements would be incorrect if they were included in the same program:

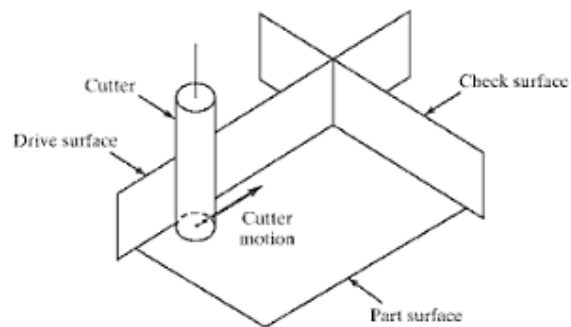
```
P1=POINT/20, 40, 60
P1=POINT/30, 50, 70
```

4. Only one symbol can be used to define any given element. For example, the following two statements in the same part program would be incorrect:

```
P1=POINT/20, 40, 60
P2=POINT/20, 40, 60
```

### Contouring motions :

Contouring commands are more complicated than PTP commands because the tool's position must be continuously controlled throughout the move. To exercise this control, the tool is directed along two intersecting surfaces until it reaches a third surface, as shown in Figure 8.

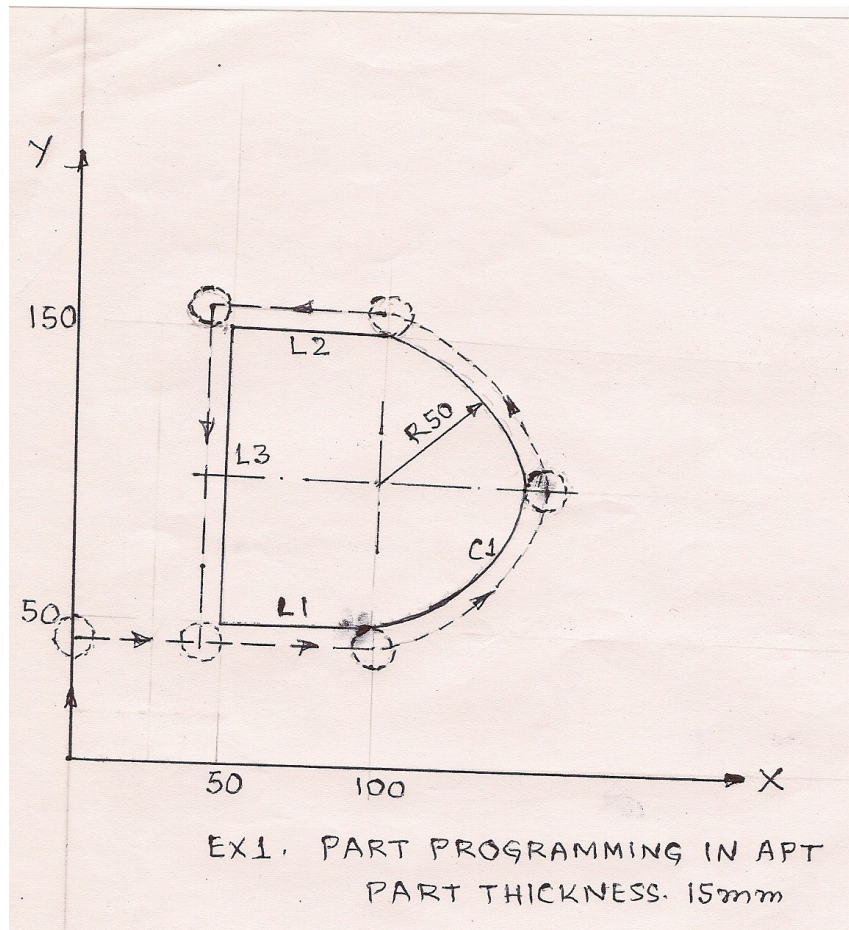


**Figure 8** Three surfaces in APT contouring motions that guide the cutting tool.

These three surfaces have specific names in APT:

1. **Drive surface** : This is the surface that guides the side of the cutter. It is pictured as a plane in our figure.
2. **Part surface** : This is the surface, again pictured as a plane, on which the bottom or nose of the tool is guided.
3. **Check surface** : This is the surface that stops the forward motion of the tool in the execution of the current command. One might say that the surface “checks” the advance of the tool.

### Example 1 : Apt programming



MACHIN/CNC1

CLPRNT

STPT=POINT/0,0

L1=LINE/50,50,100,50

L2=LINE/50,50,100,150

L3=LINE/50,50,50,150

C1=CIRCLE/100,100,RADIUS,50

P1=POINT/0,0,-20

P2=POINT/50,0,-20

P3=POINT/50,50,-20

PLN=PLANE/P1,P2,P3

CUTTER/10

SPINDL/350

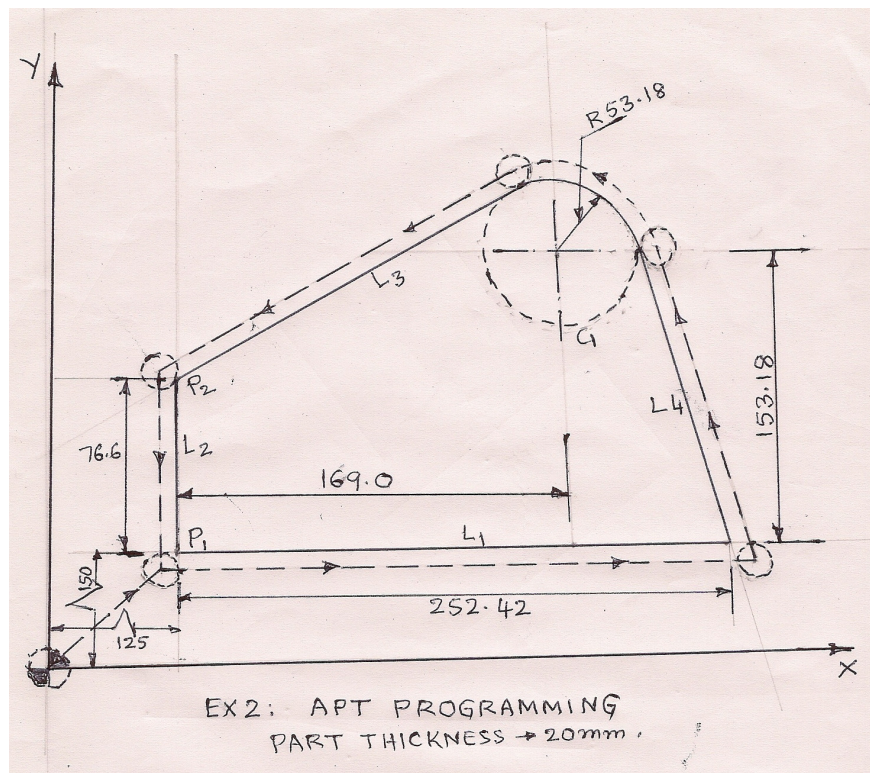
FEDRAT/30  
COOLNT/ON

FROM/STPT  
GO/TOL1,TO,L3,TO,PLN  
TLRGT,GORGT/L1,TANTO,C1  
GOFWD/C1,TANTO,L2  
GOFWD/L2,PAST,L3  
GOLFT/L3,PAST,L1  
GOTO/STPT  
COOLNT/OFF  
FINI

Note: GO/TO is used to initiate a sequence of contouring motions. Example: motion start up command in contouring to position cutter against the drive surface, part surface and check surface.

GOTO moves the cutter to only one point. E.g. GOTO/STPT

### Example 2 : Apt programming

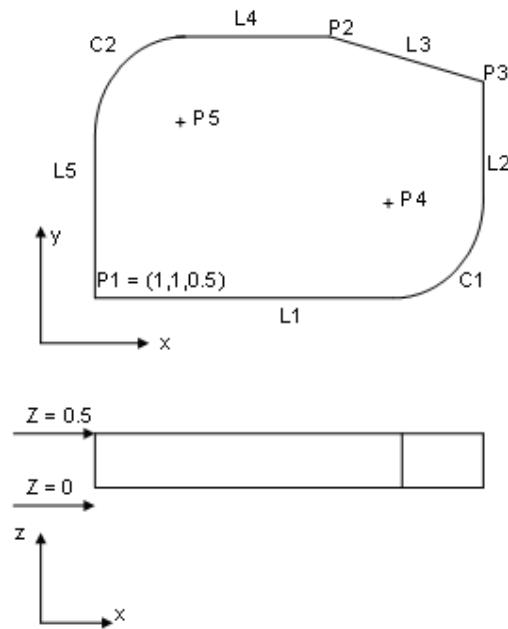


MACHIN/TMATIC  
CLPRNT  
NOPOST  
STPT=POINT/0,0,0  
P1=POINT/125,150  
P2=POINT/125,226.6  
P3=POINT/377.42,150  
L1=LINE/P1,P3  
L2=LINE/P2,PERPTO,L1  
C1=CIRCLE/294,303.18,53.18  
L3=LINE/P2,LEFT,TANTO,C1  
L4=LINE/P3,RIGHT,TANTO,C1  
P4=POINT/0,0,-25  
P5=POINT/50,0,-25  
P6=POINT/50,25,-25  
PL1=PLANE/P4,P5,P6  
CUTTER/12  
FEDRAT/300  
OUTTOL/0.025  
SPINDL/800  
FROM/STPT  
INDIRV/1,1,0  
GO/TO,L1,TO,L2,TO,PL1  
TLRGT,GORGT/L1,PAST,L4  
GOLFT/L4,TANTO,C1  
GOFWD/C1,TANTO,L3  
GOFWD/L3,PAST,L2  
GOLFT/L2,PAST,L1  
SPINDL/OFF  
GOTO/STPT  
FINI

### Example 3 : Apt contouring example

```

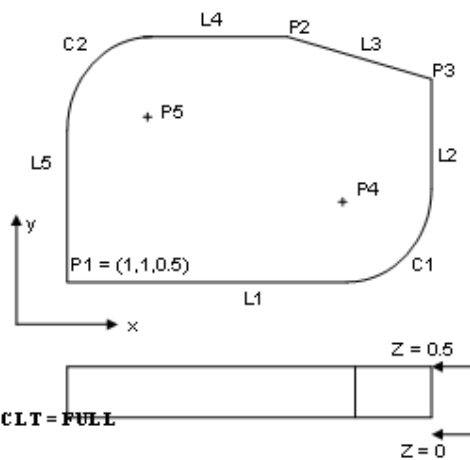
PARTNO P1534
      MACHIN/ MILL, 4
      CLPRINT
      OUTT OL/ 0.0015
P0   = POINT / 0, 0, 1.1
P1   = POINT / 1, 1, 0.5
P2   = POINT / 4, 3.5, 0.5
P3   = POINT / 5.85, 2.85, 0.5
PL1  = PLANE / P1, P2, P3
PL2  = PLANE / PARLEL, PL1, ZSMALL, 0.5
P4   = POINT / 5, 1.85, 0.5
P5   = POINT / 2, 2.5, 0.5
C1   = CIRCLE / CENTER, P4, RADIUS, 0.85
C2   = CIRCLE / CENTER, P5, RADIUS, 1.0
L1   = LINE / P1, RIGHT, TANTO, C1
L2   = LINE / P3, LEFT, TANTO, C1
L3   = LINE / P2, P3
L4   = LINE / P2, RIGHT, TANTO, C2
L5   = LINE / P1, LEFT, TANTO, C2
MILLS = MACRO / CUT, SSP, FRT, CLT
      CUTTER / CUT
  
```



### Example 4 : Apt contouring example

```

FEDRA T/FRT
SPINDL/ SSP
COOLNT/ CLT
FROM/ P0
GO/ TO, L1, TO, PL2, ON, L5
GORG/ L1, TANTO, C1
GOFWD/ C1, TANTO, L2
GOFWD/ L2, PAST, L3
GOLFT/ L3, PAST, L4
GOFWD/ L4, TANTO, C2
GOFWD/ C2, TANTO, L5
GOFWD/ L5, PAST, L1
COOLNT/ OFF
GO TO/ P0
TERMAC
TURRET 4
CALL/ MILLS, CUT = 0.52, SSP = 600, FRT = 3.0, CLT = FULL
TURRET 6
CALL/ MILLS, CUT = 0.5, SSP = 900, FRT = 2.0, CLT = FULL
SPINDL/ 0
END
FINI
  
```



### Example 5 : Apt programming

```

P0 = POINT/ 0, -2, 0
P1 = POINT/ 0.312, 0.312, 0
P2 = POINT/ 4, 1, 0
C1 = CIRCLE/ CENTER, P1, RADIUS, 0.312
C2 = CIRCLE/ CENTER, P2, RADIUS, 1
L2 = LINE/ RIGHT, TANTO, C2, RIGHT, TANTO,
    C1
L1 = LINE/ LEFT, TANTO, C2, LEFT, TANTO, C1
PL1 = PLANE/ P0, P1, P2
MILL = MACRO/ DIA
FROM/ P0
GO/TO, L1, TO, PL1, TO, C2
GOLFT/ L1, PAST, C1
GOFWD/ C1, PAST, L2
GOFWD/ L2, PAST, C2
GOFWD/ C2, PAST, L1
GOTO/ P0
TERMAC
CALL MILL / DIA = 0.70
END
FINI

```

