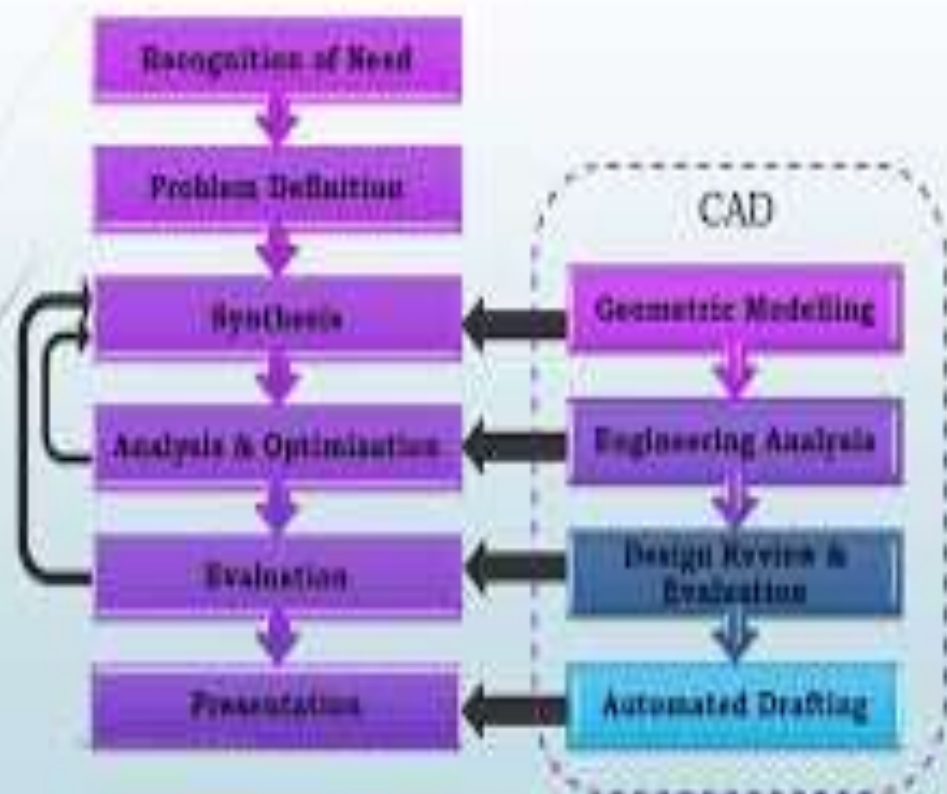


CAD/CAM

- Introduction:
- It involves any type of design activity which make use of computer to develop activity which make use of the computer to develop analysis or modify an engineering design known as CAD which make use of interactive computer graphics
- Definition:
- It is a software used to create electronic files for print, machining and other manufacturing operations , cad arrival increased the productivity of designers and improved the quality of design amongst other benefits.

CAD DESIGN PROCESS



- THE DESIGN PROCESS
- 1 Recognition of need
- 2. Definition of problem
- 3. Synthesis
- 4. Analysis and optimization
- 5. Evaluation
- 6. Presentation

- THE APPLICATION OF COMPUTERS FOR DESIGN:

The various design-related tasks which are performed by a modern computer-aided design-system can be grouped into four functional areas:

1. Geometric modeling

2. Engineering analysis

3. Design review and evaluation

4. Automated drafting

- Geometric modeling
- In computer-aided design, geometric modeling is concerned with the computer-compatible mathematical description of the geometry of an object. The mathematical description allows the image of the object to be displayed and manipulated on a graphics terminal through signals from the CPU of the CAD system. The software that provides geometric modeling capabilities must be designed for efficient use both by the computer and the human designer.
- There are several different methods of representing the object in geometric modeling. The basic form uses wire frames to represent the object. In this form, the object is displayed by interconnecting lines as shown in Figure. Wire frame geometric modeling is classified into three types depending on the capabilities of the ICG system. The three types are:
- I. 2D. Two-dimensional representation is used for a flat object.

- Engineering analysis:
- In the formulation of nearly any engineering design project, some type of analysis is required. The analysis may involve stress-strain calculations, heat-transfer computations, or the use of differential equations to describe the dynamic behavior of the system being designed. The computer can be used to aid in this analysis work. It is often necessary that specific programs be developed internally by the engineering analysis group to solve a particular design problem. In other situations, commercially available general-purpose programs can be used to perform the engineering analysis.

- Design review and evaluation
- Checking the accuracy of the design can be accomplished conveniently on the graphics terminal. Semiautomatic dimensioning and tolerance routines which assign size specifications to surfaces indicated by the user help to reduce the possibility of dimensioning errors. The designer can zoom in on part design details and magnify the image on the graphics screen for close scrutiny.

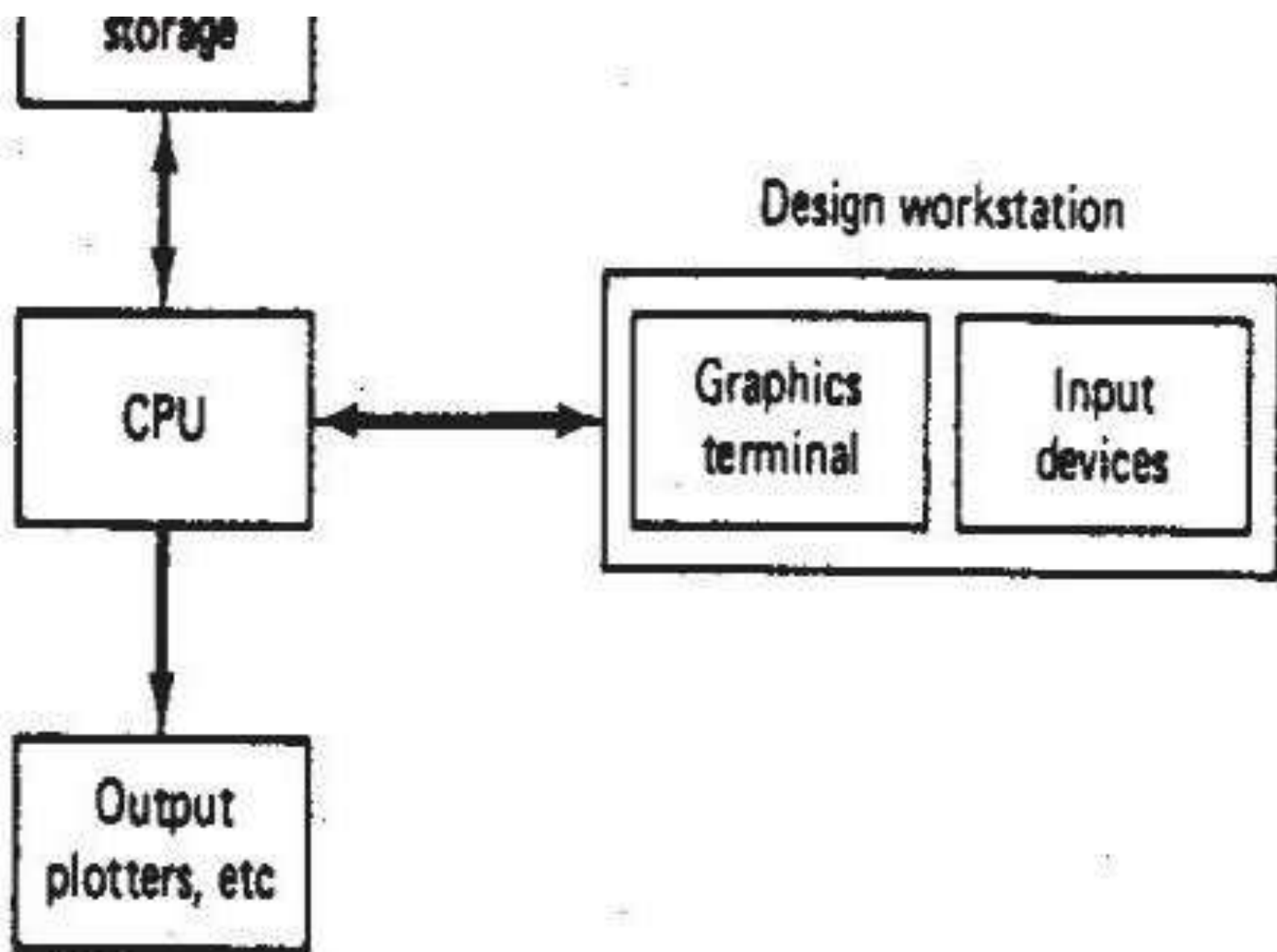
- Automated drafting
- Automated drafting involves the creation of hard-copy engineering drawings directly from the CAD data base. In some early computer-aided design departments, automation of the drafting process represented the principal justification for investing in the CAD system. Indeed, CAD systems can increase productivity in the drafting function by roughly five times over manual drafting.

- BENEFITS OF CAD:

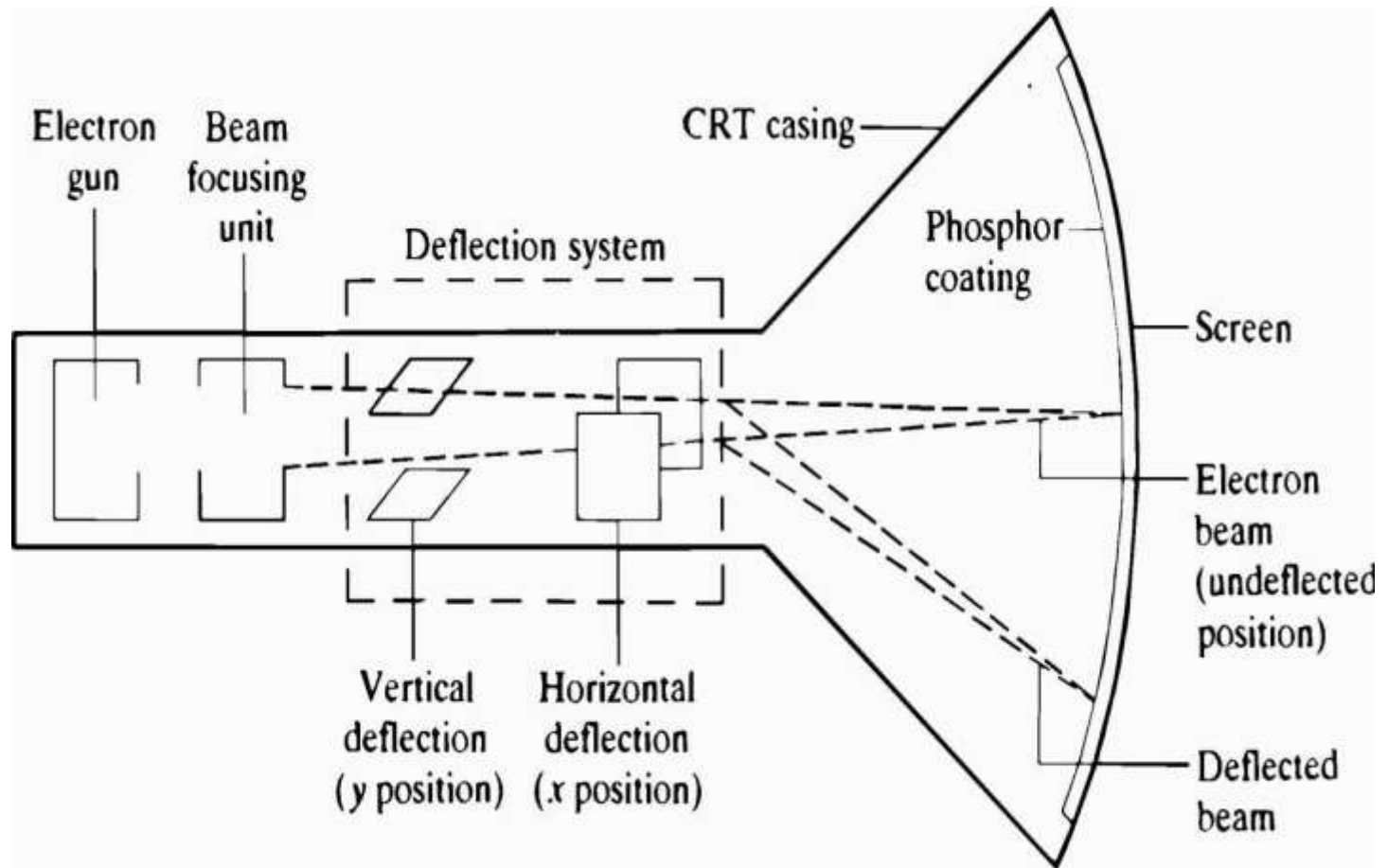
1. Improved engineering productivity
2. Shorter lead times
3. Reduced engineering personnel requirements
4. Customer modifications are easier to make
5. Faster response to requests for quotations
6. Avoidance of subcontracting to meet schedules
7. Minimized transcription errors
8. Improved accuracy of design
9. In analysis, easier recognition of component interactions
10. Provides better functional analysis to reduce prototype testing II. Assistance in preparation of documentation

- 12. Designs have more standardization
- 13. Better designs provided
- 14. Improved productivity in tool design
- 15. Better knowledge of costs provided
- 16. Reduced training time for routine drafting tasks and NC part programming
- 17. Fewer errors in NC part programming
- 18. Provides the potential for using more existing parts and tooling
- 19. Helps ensure designs are appropriate to existing manufacturing techniques
- 20. Saves materials and machining time by optimization algorithms
- 21. Provides operational results on the status of work in progress
- 22. Makes the management of design personnel on projects more effective
- 23. Assistance in inspection of complicated parts
- 24. Better communication interfaces and greater understanding among engineers, designers, drafters, management, and different project groups.

- **HARDWARE IN COMPUTER-AIDED DESIGN**
- Hardware components for computer-aided design are available in a variety of sizes, configurations, and capabilities. Hence it is possible to select a CAD system that meets the particular computational and graphics requirements of the user firm. Engineering firms that are not involved in production would choose a system exclusively for drafting and design-related functions. Manufacturing firms would choose a system to be part of a company-wide CAD/CAM system.



GRAPHICAL TERMINAL:



- Nearly all computer graphics terminals available today use the cathode ray tube (CRT) as the display device. Television sets use a form of the same device as the picture tube. 'The operation of the CRT is illustrated in Figure. A heated cathode emits a high-speed electron beam onto a phosphor-coated glass screen. 'The electrons energize the phosphor coating, causing it to glow at the points where the beam makes contact. By focusing the electron beam, changing its intensity, and controlling its point of contact against the phosphor coating through the use of a deflector system, the beam can be made to generate a picture on the CRT screen.
- There are two basic techniques used in current computer graphics terminals for generating the image on the CRT screen. They are:

1. Stroke writing

2. Raster scan

The stroke-writing system uses an electron beam which operates like a pencil to create a line image on the CRT screen. The image is constructed out of a sequence of straight-line segments. Each line segment is drawn on the screen by directing the beam to move from one point on the screen to the next, where each point is defined by its x and y coordinates. The process is portrayed in Figure . Although the procedure results in images composed of only straight lines, smooth curves can be approximated by making the connecting line segments short enough.

2D Transformation:

Transformations play an important role in computer graphics to reposition the graphics on the screen and change their size or orientation.

-To perform a sequence of transformation such as translation followed by rotation and scaling, we need to follow a sequential process

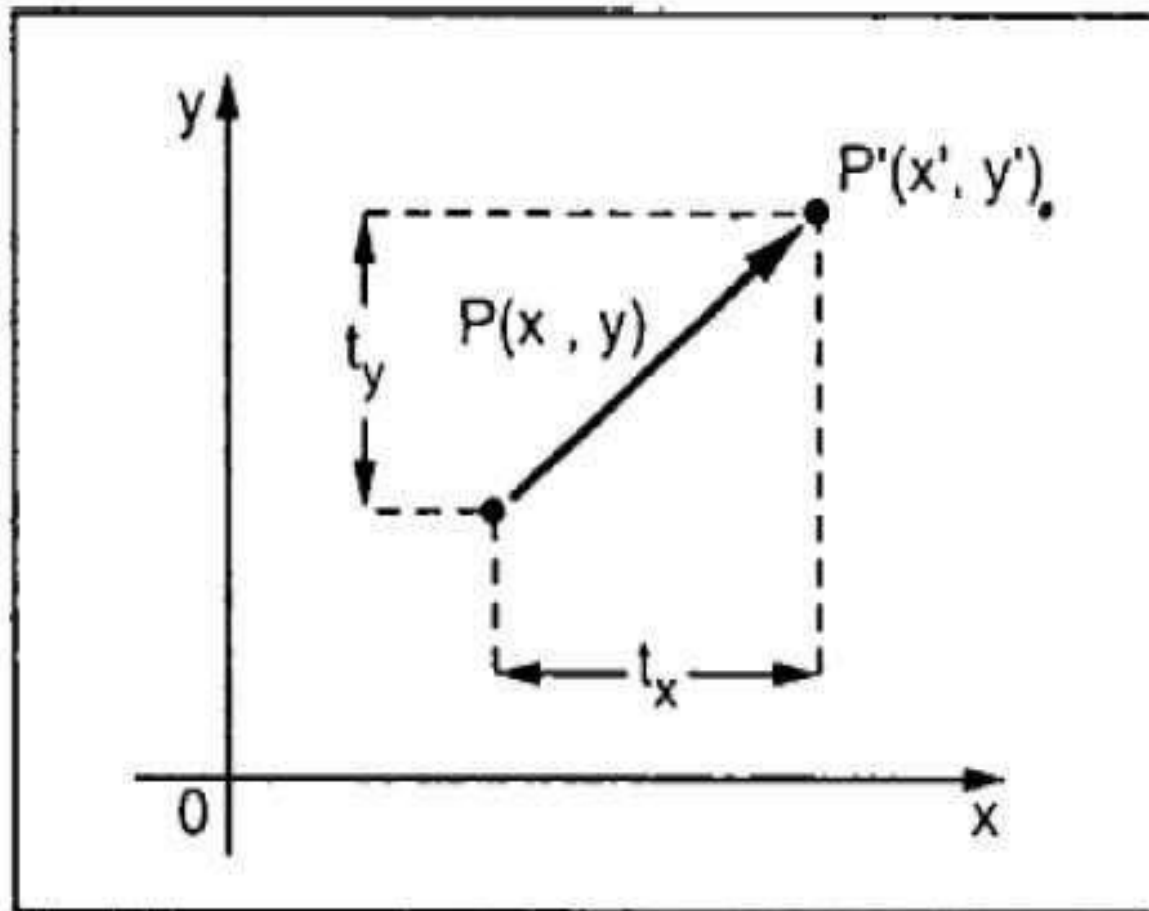
- Translate the coordinates,
- Rotate the translated coordinates, and then
- Scale the rotated coordinates to complete the composite transformation.

-To shorten this process, we have to use 3×3 transformation matrix instead of 2×2 transformation matrix. To convert a 2×2 matrix to 3×3 matrix, we have to add an extra dummy coordinate W .

- represent the point by 3 numbers instead of 2 numbers, which is called **Homogenous Coordinate** system. In this system, we can represent all the transformation equations in matrix multiplication. Any Cartesian point $P(X, Y)$ can be converted to homogenous coordinates by $P' (X_h, Y_h, h)$.

Translation

A translation moves an object to a different position on the screen. You can translate a point in 2D by adding translation coordinate (t_x, t_y) to the original coordinate (X, Y) to get the new coordinate (X', Y') .



From the above figure, you can write that –

$$\mathbf{X}' = \mathbf{X} + \mathbf{t}_x$$

$$\mathbf{Y}' = \mathbf{Y} + \mathbf{t}_y$$

The pair (t_x, t_y) is called the translation vector or shift vector. The above equations can also be represented using the column vectors.

$$\mathbf{P} = \begin{bmatrix} X \\ Y \end{bmatrix} \quad \mathbf{p}' = \begin{bmatrix} X' \\ Y' \end{bmatrix} \quad \mathbf{T} = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

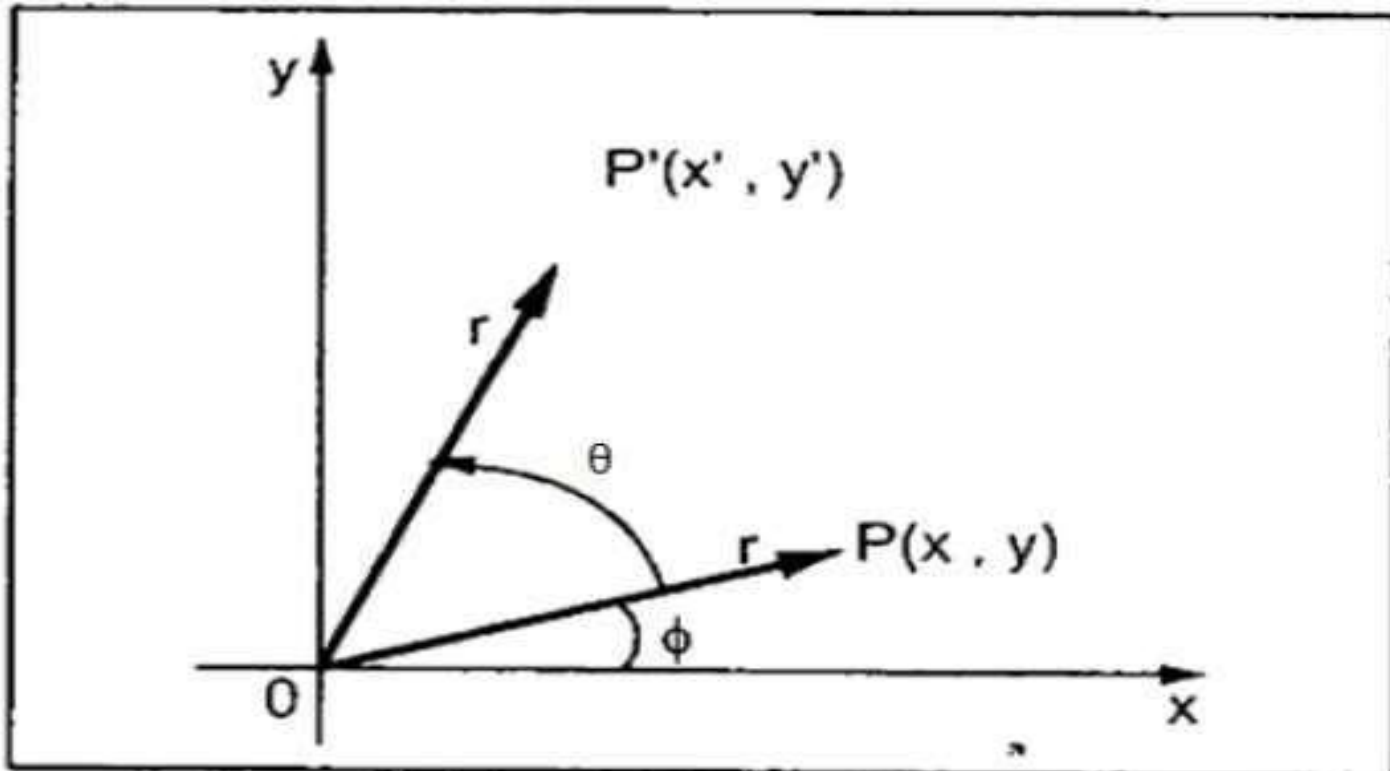
We can write it as –

$$\mathbf{P}' = \mathbf{P} + \mathbf{T}$$

Rotation

In rotation, we rotate the object at particular angle θ (theta) from its origin. From the following figure, we can see that the point $P(X, Y)$ is located at angle ϕ from the horizontal X coordinate with distance r from the origin.

Let us suppose you want to rotate it at the angle θ . After rotating it to a new location, you will get a new point $P'(X', Y')$.



$$X = r \cos \phi \dots \dots (1)$$

$$Y = r \sin \phi \dots \dots (2)$$

Same way we can represent the point P' (X', Y') as –

$$x' = r \cos (\phi + \theta) = r \cos \phi \cos \theta - r \sin \phi \sin \theta \dots \dots (3)$$

$$y' = r \sin (\phi + \theta) = r \cos \phi \sin \theta + r \sin \phi \cos \theta \dots \dots (4)$$

Substituting equation (1) & (2) in (3) & (4) respectively, we will get

$$x' = x \cos \theta - y \sin \theta$$

$$y' = x \sin \theta + y \cos \theta$$

Representing the above equation in matrix form,

$$[X'Y'] = [XY] \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \text{OR}$$

$$P' = P \cdot R$$

Where R is the rotation matrix

$$R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

The rotation angle can be positive and negative.

For positive rotation angle, we can use the above rotation matrix. However, for negative angle rotation, the matrix will change as shown below –

$$\begin{aligned} R &= \begin{bmatrix} \cos(-\theta) & \sin(-\theta) \\ -\sin(-\theta) & \cos(-\theta) \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \quad (\because \cos(-\theta) = \cos\theta \text{ and } \sin(-\theta) = -\sin\theta) \end{aligned}$$

- **Scaling**
- To change the size of an object, scaling transformation is used. In the scaling process, you either expand or compress the dimensions of the object. Scaling can be achieved by multiplying the original coordinates of the object with the scaling factor to get the desired result.
- Let us assume that the original coordinates are (X, Y) , the scaling factors are (S_x, S_y) , and the produced coordinates are (X', Y') . This can be mathematically represented as shown below –

$$\mathbf{X' = X \cdot S_x \text{ and } Y' = Y \cdot S_y}$$

The scaling factor S_x, S_y scales the object in X and Y direction respectively.

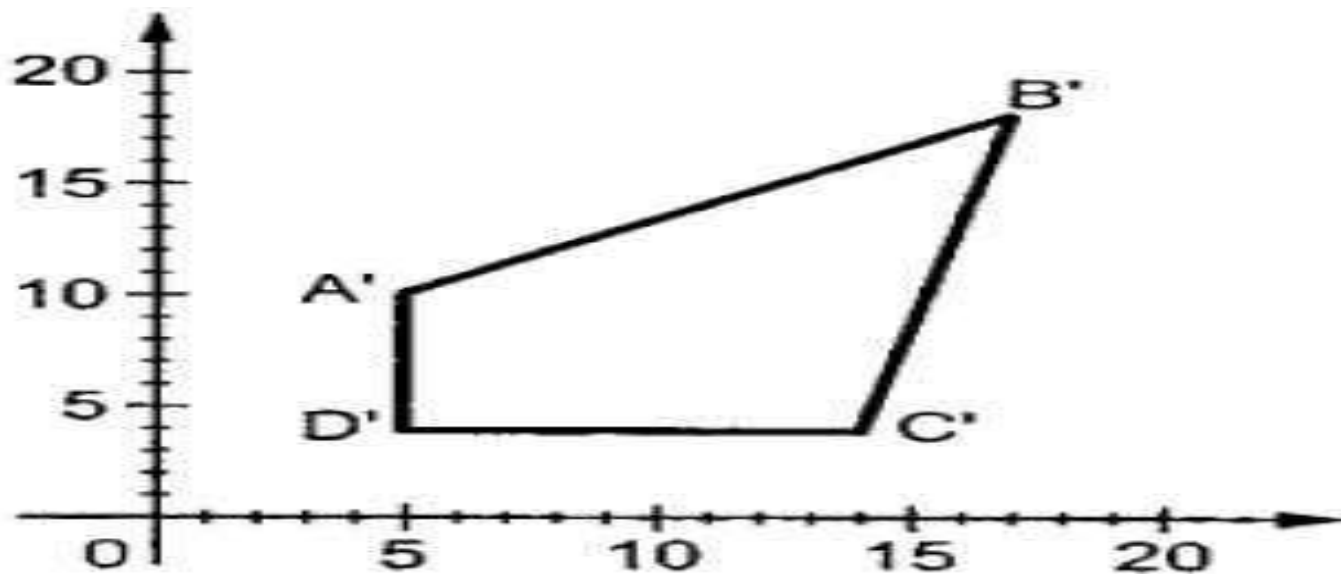
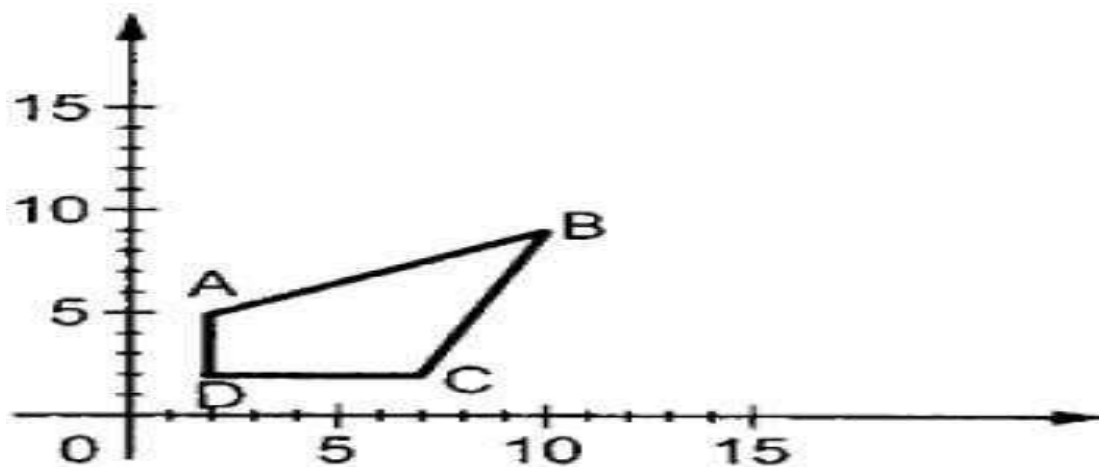
The above equations can also be represented in matrix form as below –

$$(X'Y')=(XY)[S_x00S_y](X'Y')=(XY)[S_x00S_y]$$

OR

$$\mathbf{P' = P \cdot S}$$

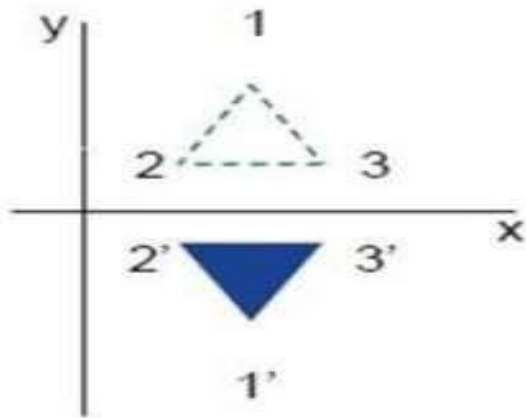
Where S is the scaling matrix. The scaling process is shown in the following figure.



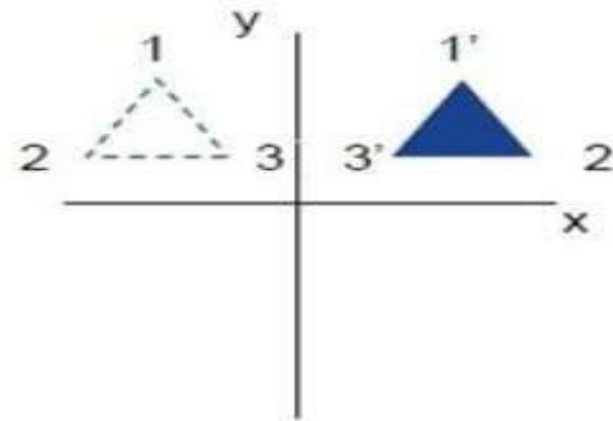
Reflection

Reflection is the mirror image of original object. In other words, we can say that it is a rotation operation with 180° . In reflection transformation, the size of the object does not change.

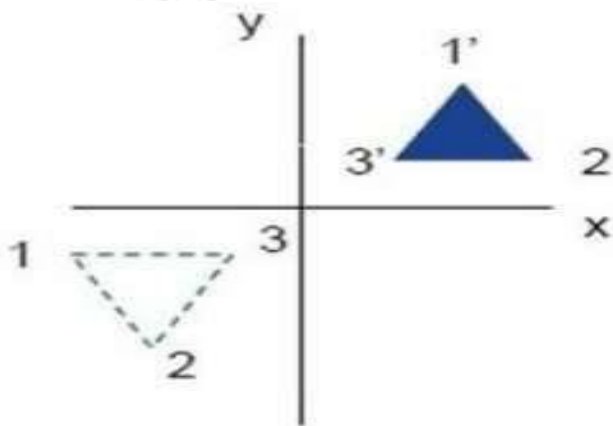
The following figures show reflections with respect to X and Y axes, and about the origin respectively.



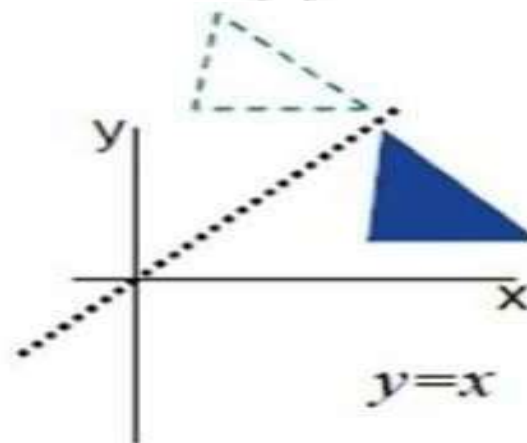
(a)



(b)



(c)



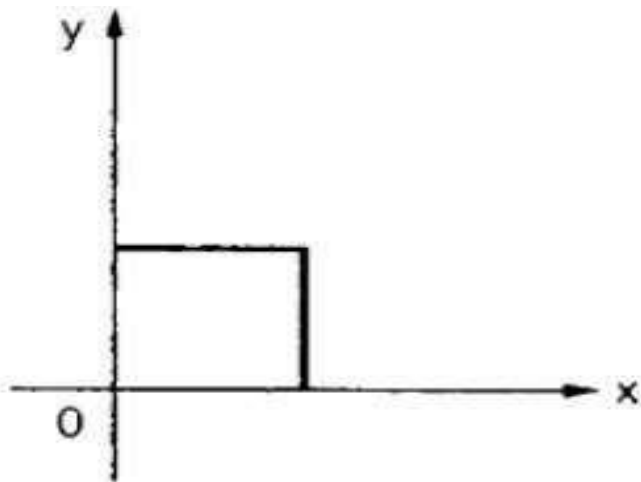
(d)

Shear

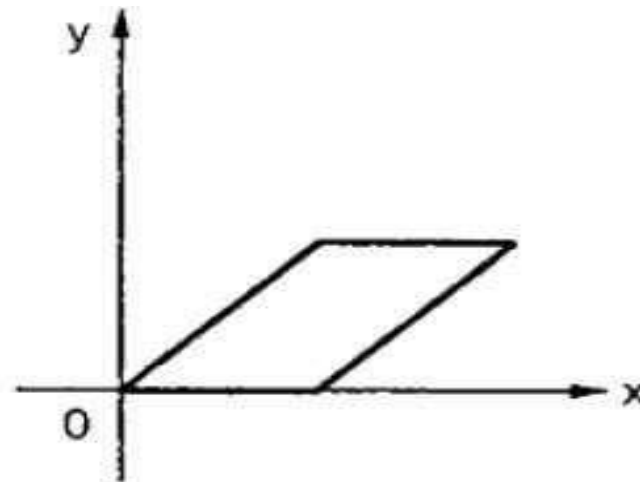
A transformation that slants the shape of an object is called the shear transformation. There are two shear transformations **X-Shear** and **Y-Shear**. One shifts X coordinates values and other shifts Y coordinate values. However; in both the cases only one coordinate changes its coordinates and other preserves its values. Shearing is also termed as **Skewing**.

X-Shear

The X-Shear preserves the Y coordinate and changes are made to X coordinates, which causes the vertical lines to tilt right or left as shown in below figure.



(a) Original object



(b) Object after x shear

The transformation matrix for X-Shear can be represented as –

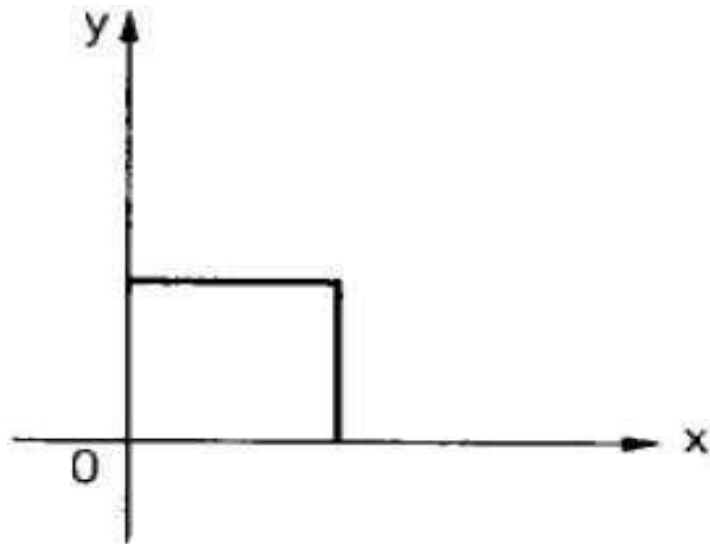
$$X_{sh} = \begin{bmatrix} 1 & shx & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$Y' = Y + Sh_y \cdot X$$

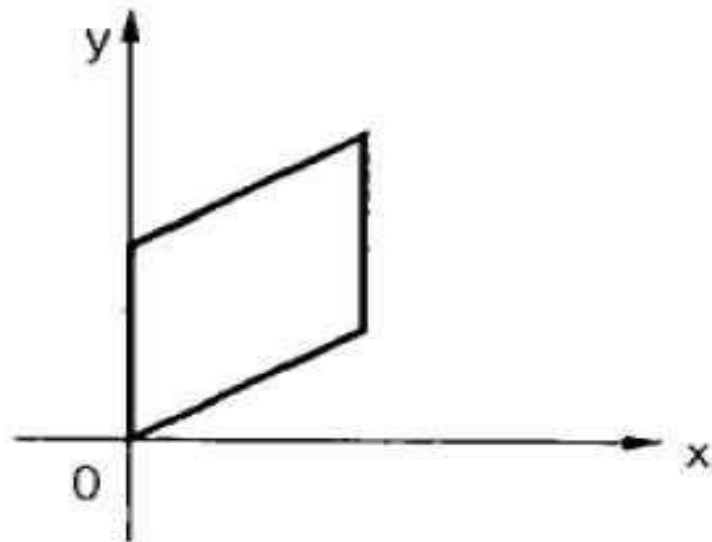
$$X' = X$$

Y-Shear

The Y-Shear preserves the X coordinates and changes the Y coordinates which causes the horizontal lines to transform into lines which slopes up or down as shown in the following figure.



(a) Original object



(b) Object after y shear

The Y-Shear can be represented in matrix form as –

$$Y_{sh} \begin{bmatrix} 1 & 0 & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$X' = X + Sh_x \cdot Y$$

$$Y' = Y$$

- **Composite Transformation**
- If a transformation of the plane T_1 is followed by a second plane transformation T_2 , then the result itself may be represented by a single transformation T which is the composition of T_1 and T_2 taken in that order. This is written as $T = T_1 \cdot T_2$.
- Composite transformation can be achieved by concatenation of transformation matrices to obtain a combined transformation matrix.
- A combined matrix – Translation, scaling, Shearing, rotation and reflection

- to rotate an object about an arbitrary point (X_p, Y_p) , we have
- to carry out three steps –
 - Translate point (X_p, Y_p) to the origin.
 - Rotate it about the origin.
 - Finally, translate the center of rotation back where it belonged.

Viewing transformation:

the picture is stored in the computer memory using any convenient Cartesian co-ordinate system, referred to as World Co-Ordinate System (WCS). However, when picture is displayed on the display device it is measured in Physical Device Co-Ordinate System (PDCS) corresponding to the display device. Therefore, displaying an image of a picture involves mapping the co-ordinates of the Points and lines that form the picture into the appropriate physical device co-ordinate where the image is to be displayed. This mapping of co-ordinates is achieved with the use of co-ordinate transformation known as viewing transformation.

- The viewing transformation which maps picture co-ordinates in the WCS to display co-ordinates in PDCS is performed by the following transformations.
- Converting world co-ordinates to viewing co-ordinates.
- Normalizing viewing co-ordinates.
- Converting normalized viewing co-ordinates to device co-ordinates.

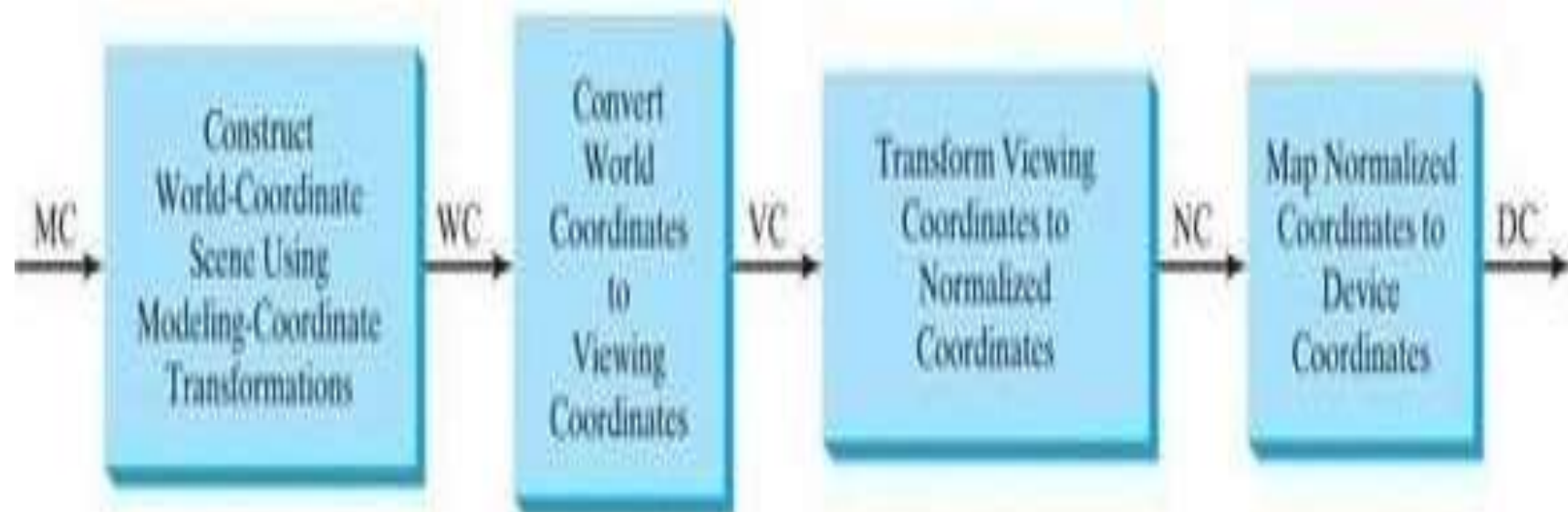


Fig. (c) Two-dimensional viewing transformation pipeline

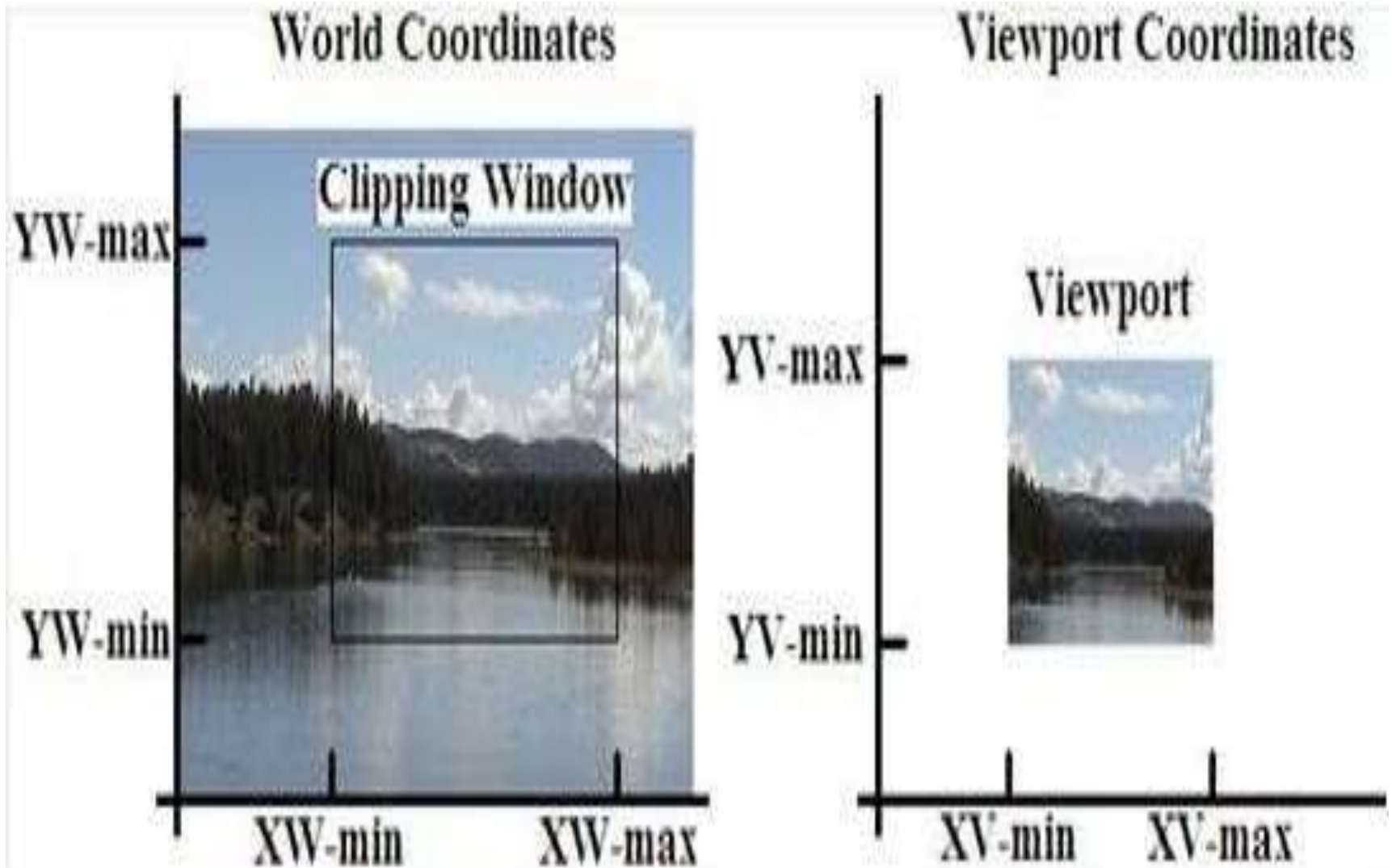
window and viewport:

- A world-coordinate area selected for display is called a window. In computer graphics, a window is a graphical control element
- An area on a display device to which a window is mapped is called a viewport. An area on a display device to which a window is mapped is called a viewport.

Window to viewport transformation:

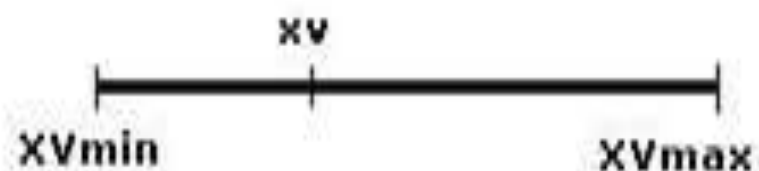
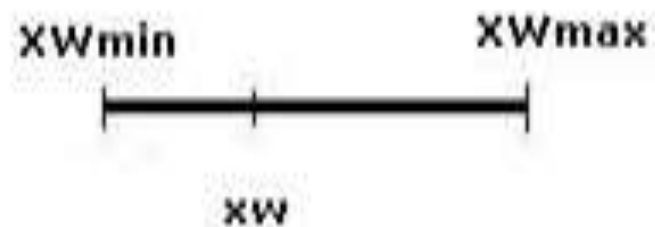
- Window-to-Viewport transformation is the process of transforming a two-dimensional, world-coordinate scene to device coordinates.
- the clipping window is used to select the part of the scene that is to be displayed. The viewport then positions the scene on the output device.

Example:

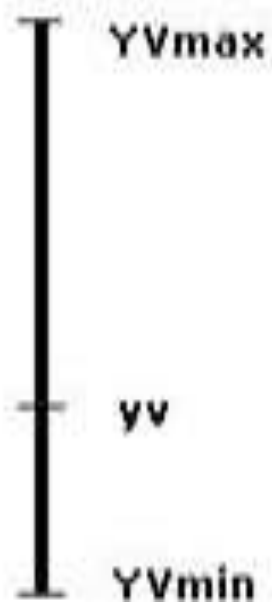


- This transformation involves developing formulas that start with a point in the world window, say (x_w, y_w) .
- The formula is used to produce a corresponding point in viewport coordinates, say (x_v, y_v) . We would like for this mapping to be "proportional" in the sense that if x_w is 30% of the way from the left edge of the world window, then x_v is 30% of the way from the left edge of the viewport.
- Similarly, if y_w is 30% of the way from the bottom edge of the world window, then y_v is 30% of the way from the bottom edge of the viewport. The picture below shows this proportionality.

For proportionality in x:



For proportionality in y:



1. Using this proportionality, the following ratios must be equal.

$$\frac{xv - xv_{min}}{xv_{max} - xv_{min}} = \frac{xw - xw_{min}}{xw_{max} - xw_{min}}$$

$$\frac{yv - yv_{min}}{yv_{max} - yv_{min}} = \frac{yw - yw_{min}}{yw_{max} - yw_{min}}$$

1. By solving these equations for the unknown viewport position (xv, yv) , the following becomes true:

$$xv = S_x xw + t_x$$

$$yv = S_y yw + t_y$$

1. And the translation factors (T_x, T_y) would be:

$$t_x = \frac{xW_{max}xV_{min} - xW_{min}xV_{max}}{xW_{max} - xW_{min}}$$

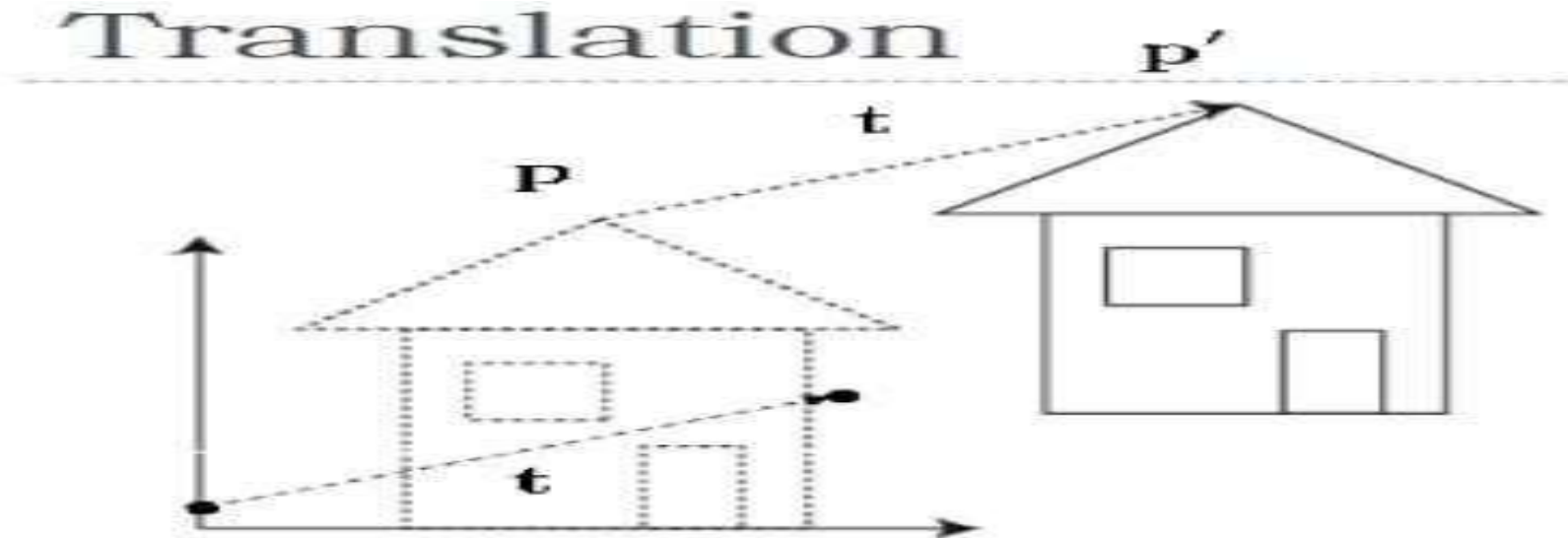
$$t_y = \frac{yW_{max}yV_{min} - yW_{min}yV_{max}}{yW_{max} - yW_{min}}$$

3D Transformation:

Translation

In 3D translation, we transfer the Z coordinate along with the X and Y coordinates. The process for translation in 3D is similar to 2D translation. A translation moves an object into a different position on the screen.

The following figure shows the effect of translation –



A point can be translated in 3D by adding translation coordinate (t_x, t_y, t_z) to the original coordinate (X, Y, Z) to get the new coordinate (X', Y', Z') .

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_x & t_y & t_z & 1 \end{bmatrix}$$

$$P' = P \cdot T$$

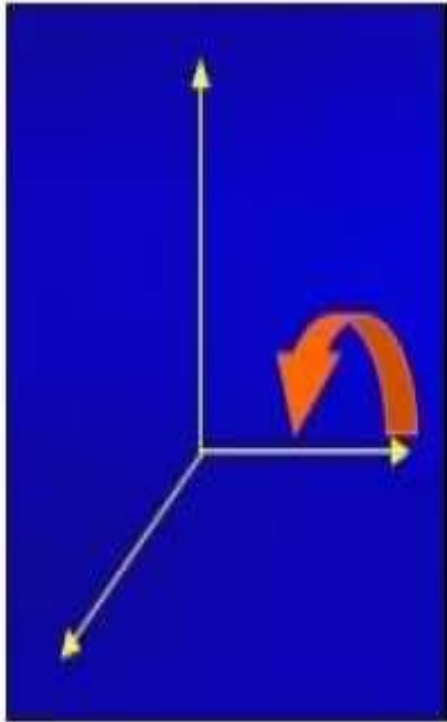
$$\begin{aligned} [X' \ Y' \ Z' \ 1] &= [X \ Y \ Z \ 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_x & t_y & t_z & 1 \end{bmatrix} \\ &= [X + t_x \ Y + t_y \ Z + t_z \ 1] \end{aligned}$$

- **Rotation**
3D Transformation

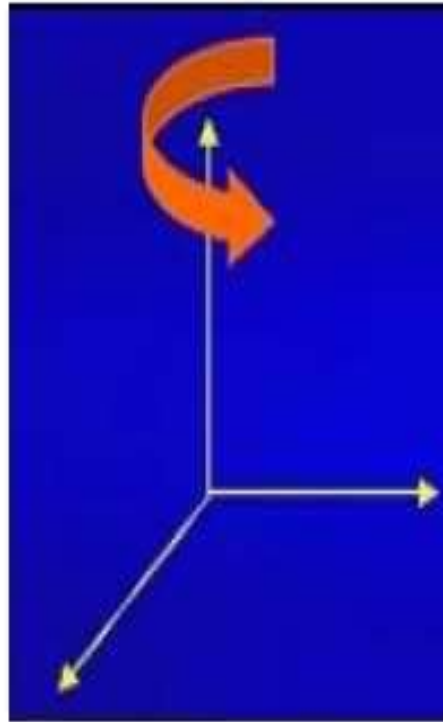
3D rotation is not same as 2D rotation. In 3D rotation, we have to specify the angle of rotation along with the axis of rotation. We can perform 3D rotation about X, Y, and Z axes. They are represented in the matrix form as below –

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} R_z(\theta)$$
$$= \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

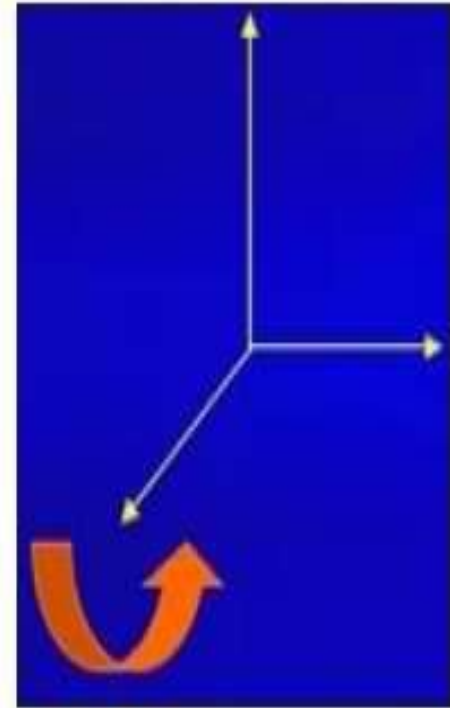
The following figure explains the rotation about various axes –



Rotation about x-axis



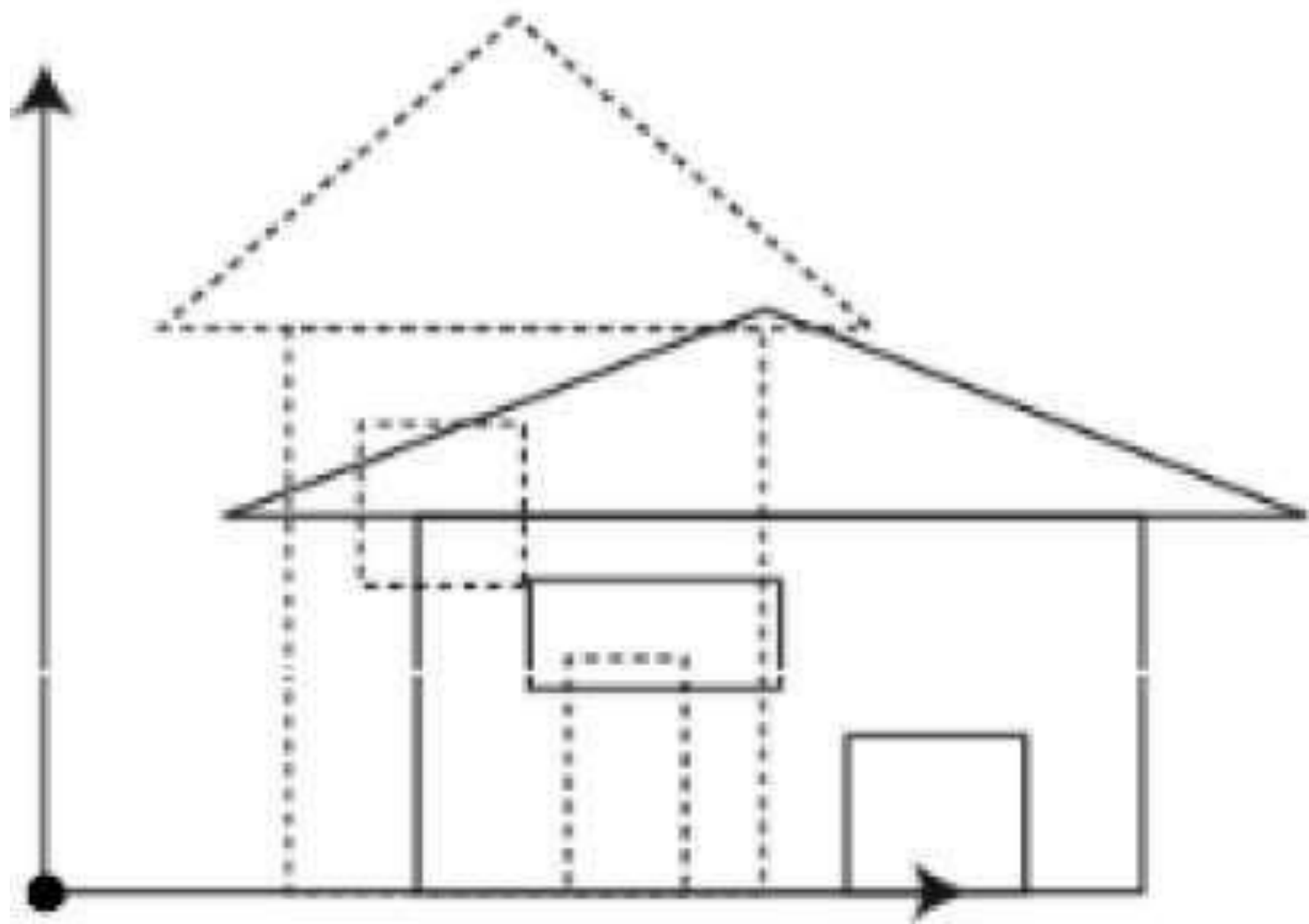
Rotation about y-axis



Rotation about z-axis

Scaling

-You can change the size of an object using scaling transformation. In the scaling process, you either expand or compress the dimensions of the object. Scaling can be achieved by multiplying the original coordinates of the object with the scaling factor to get the desired result. The following figure shows the effect of 3D scaling –



In 3D scaling operation, three coordinates are used. Let us assume that the original coordinates are (X, Y, Z), scaling factors are (S_x, S_y, S_z) respectively, and the produced coordinates are (X', Y', Z'). This can be mathematically represented as shown below –

$$S = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

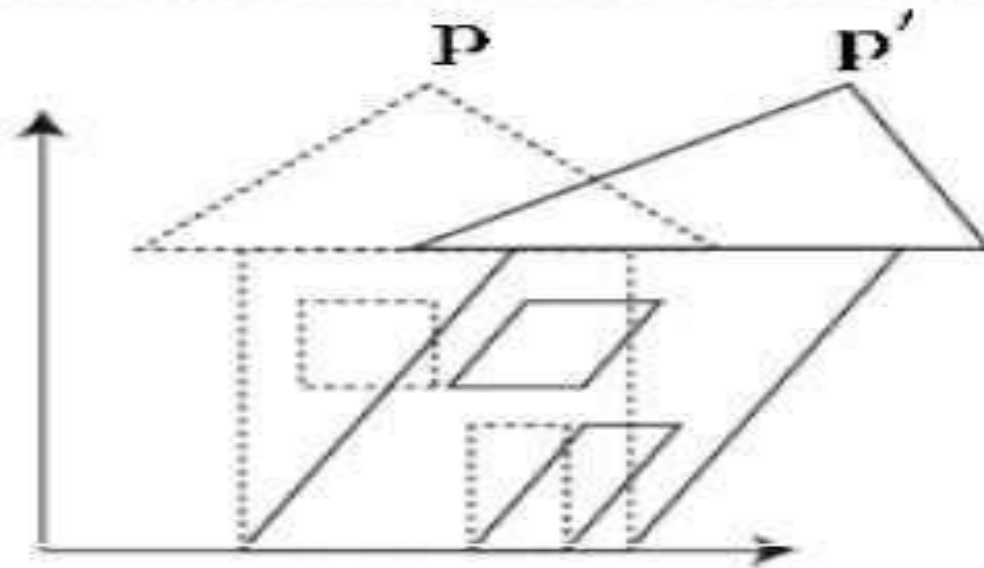
$$P' = P \cdot S$$

$$\begin{aligned} [X' \ Y' \ Z' \ 1] &= [X \ Y \ Z \ 1] \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= [X \cdot S_x \ Y \cdot S_y \ Z \cdot S_z \ 1] \end{aligned}$$

- **Shear**

A transformation that slants the shape of an object is called the **shear transformation**. Like in 2D shear, we can shear an object along the X- axis, Y-axis, or Z-axis in 3D.

Shear



As shown in the above figure, there is a coordinate P. You can shear it to get a new coordinate P', which can be represented in 3D matrix form as below

$$Sh = \begin{bmatrix} 1 & sh_x^y & sh_x^z & 0 \\ sh_y^x & 1 & sh_y^z & 0 \\ sh_z^x & sh_z^y & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P' = P \cdot Sh$$

$$X' = X + Sh_x^y Y + Sh_x^z Z$$

$$Y' = Sh_y^x X + Y + sh_y^z Z$$

$$Z' = Sh_z^x X + Sh_z^y Y + Z$$

UNIT 2: Synthetic surfaces and representations:

The curve that goes through the data points are called is called synthetic curve.

The components of synthetic curves are

- Hermit cubic spline curve
- Bezier curve
- B-spline curve

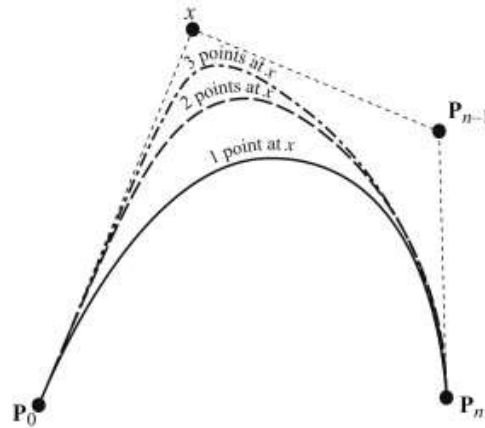
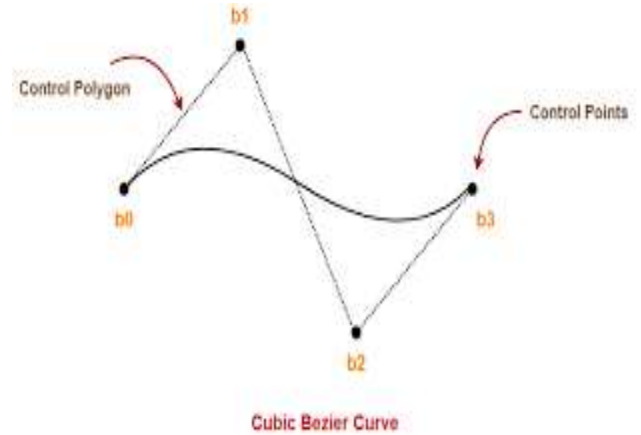
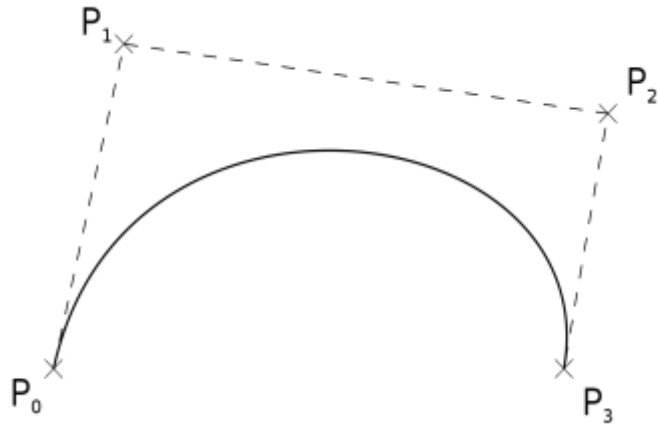
Hermit cubic spline curve:

Hermit cubic curve is determined by defining “ 2 positions and 2 tangent vectors at the data points The hermit cubic curve is also known as parametric cubic curve and cubic spline



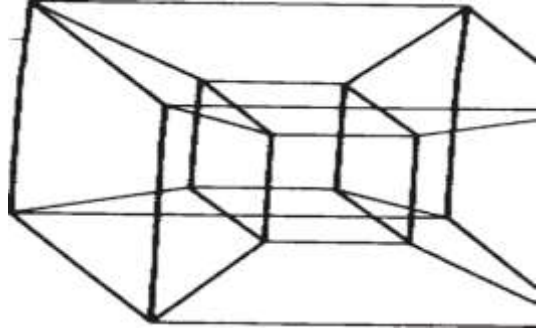
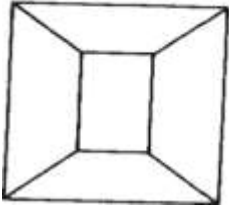
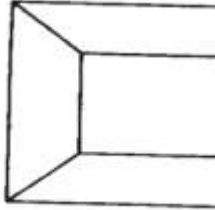
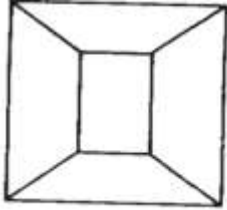
Hermite Specification

Bezier curve:



It is a parametric curve used in computer graphics and related fields. A set of discrete "control points" defines a smooth, continuous curve by means of a formula. Usually the curve is intended to approximate a real-world shape that otherwise has no mathematical representation or whose representation is unknown or too complicated. The Bézier curve is named after French engineer Pierre Bézier (1910–1999), who used it in the 1960s for designing curves for the bodywork of Renault cars. Other uses include the design of computer fonts and animation. Bézier curves can be combined to form a Bézier spline, or generalized to higher dimensions to form Bézier surfaces. The Bézier triangle is a special case of the latter.

- WIRE FRAME MODELING
- Most current day graphics systems use a form of modeling called wire-frame modeling. In the construction of the wire-frame model the edges of the objects are shown as lines. For objects in which there are curved surfaces, contour lines can be added; as shown in Figure, to indicate the contour. The image assumes the appearance of a frame constructed out of wire-hence the name “wireframe” model.



- Solid model:
- An improvement over wire-frame models, both in terms of realism to the user and definition to the computer, is the solid modeling approach. In this approach, the models are displayed as solid objects to the viewer, with very little risk of misinterpretation. When color is added to the image, the resulting picture becomes strikingly realistic. It is anticipated that graphics systems with this capability will find a wide range of applications outside computer-aided design and manufacturing. These applications will include color illustrations in magazines and technical publications, animation in movie films, and training simulators (e.g., aircraft pilot training).

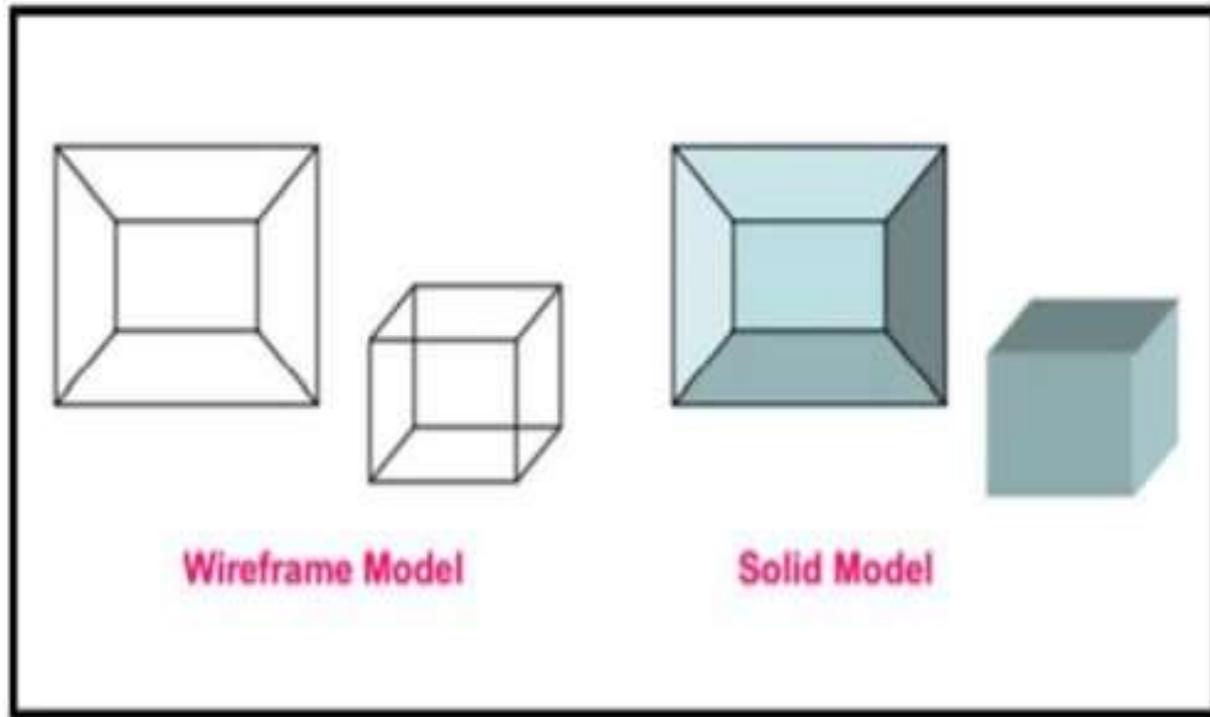
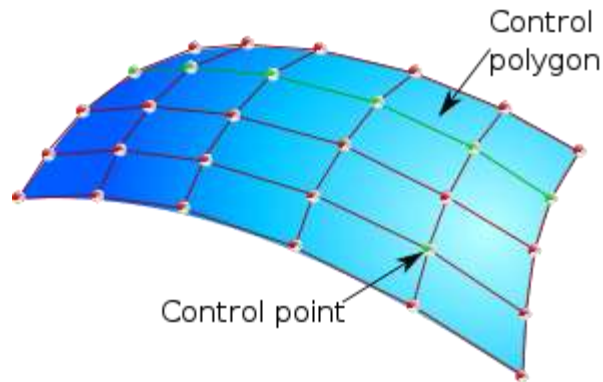


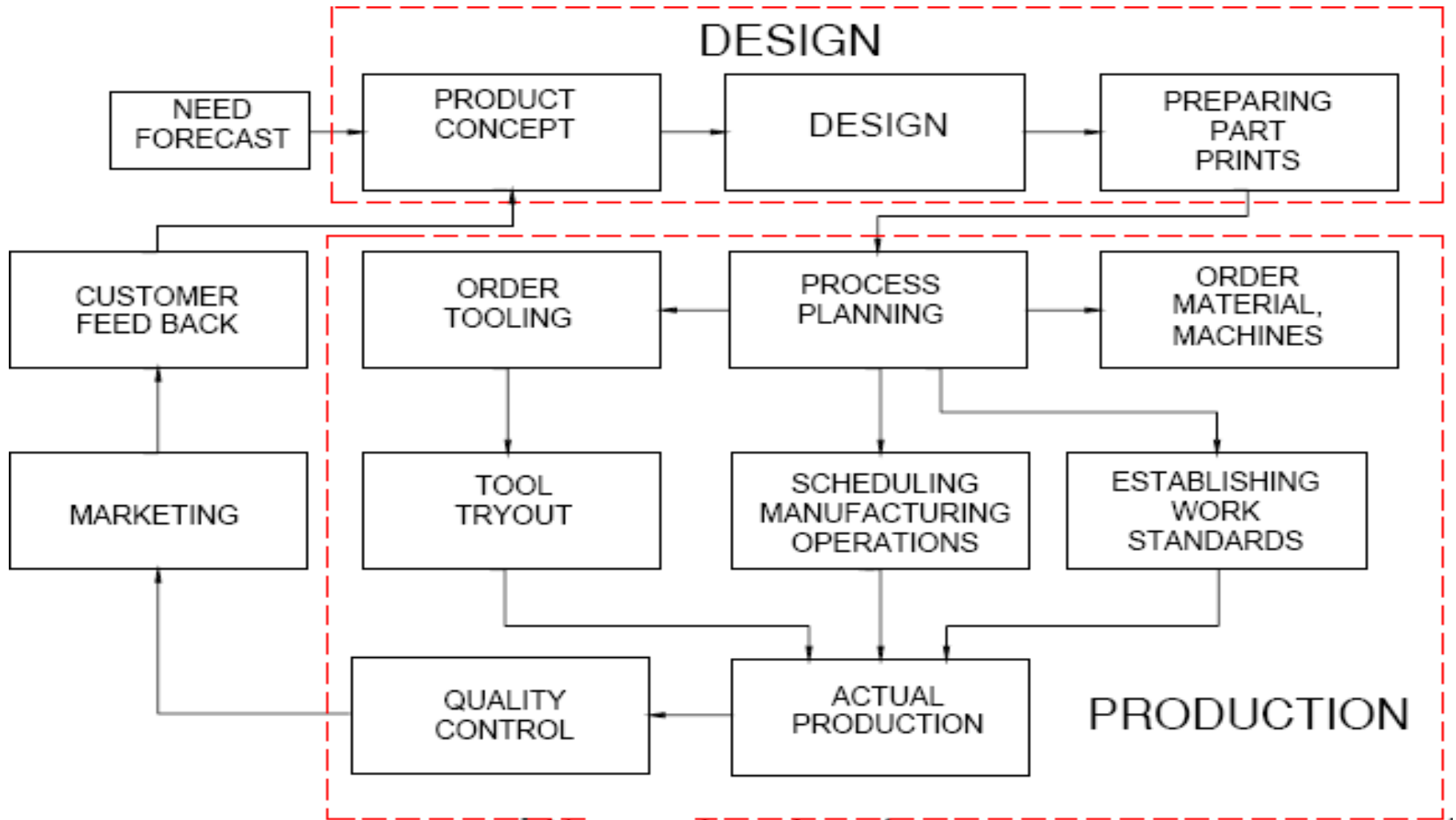
Fig: 4.1 Solid Model

- Surface model:
- Surface Modeling is **the method of showing or presenting solid objects**. The process requires you to convert between different 3D modeling types, such as converting the 3D object to show procedural surfaces, validate imperfections, and apply smoothness.

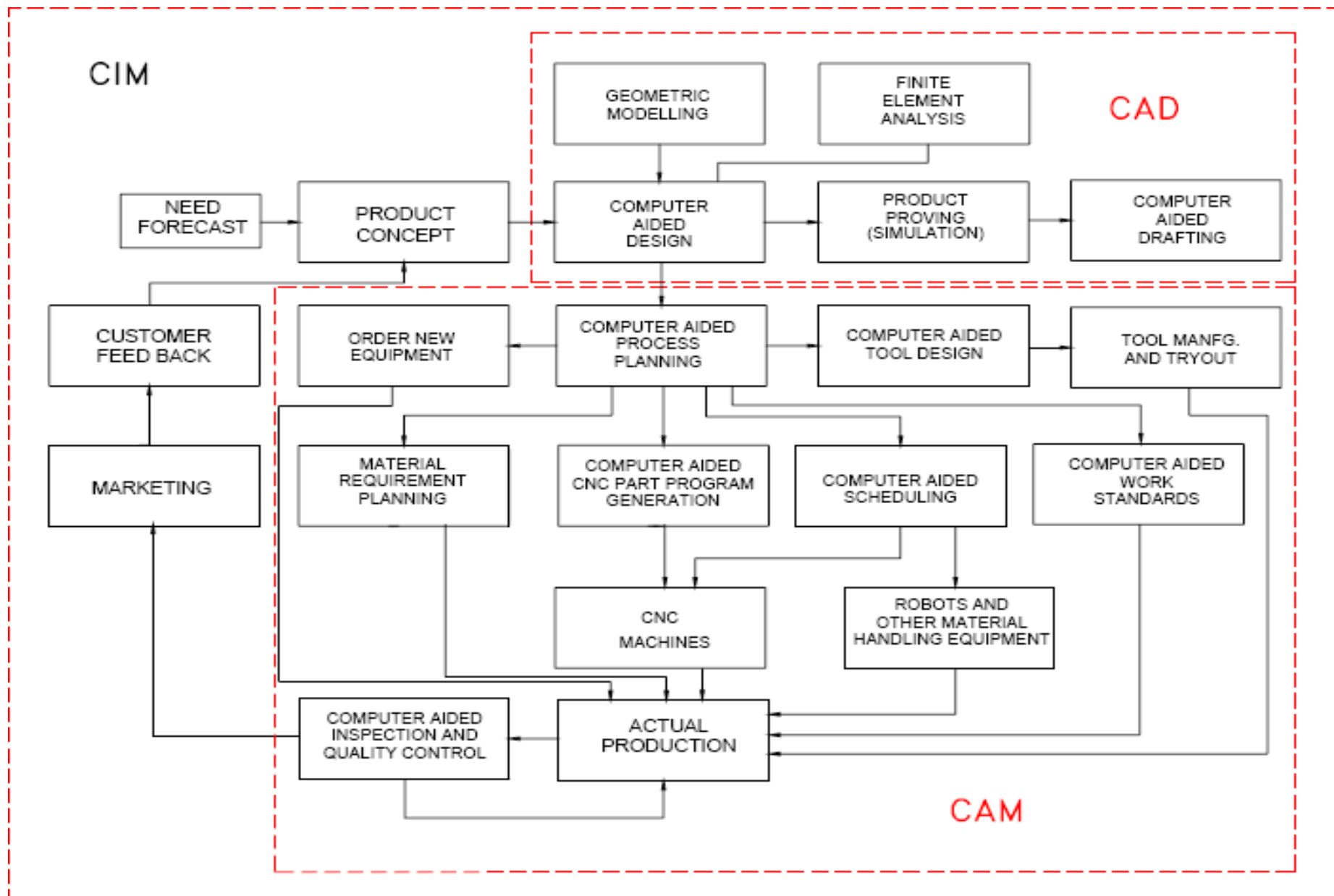


CAM

Product cycle of a traditional Manufacturing system



Product cycle in a Computer Aided Manufacturing Environment



Benefits of CAM:

- Flexibility in design
- Increased Productivity
- High operating Flexibility
- Minimized Maintenance
- Less lead time
- Better Management control
- Less Scrap
- No Rework

NC SYSTEM:

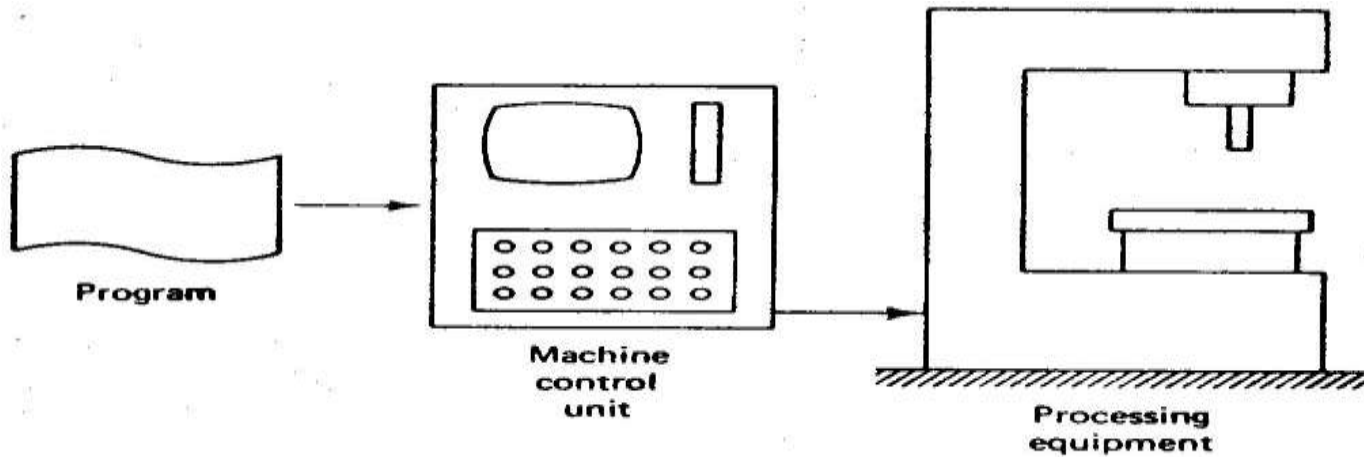


FIGURE 8.1 Basic components of an NC system.

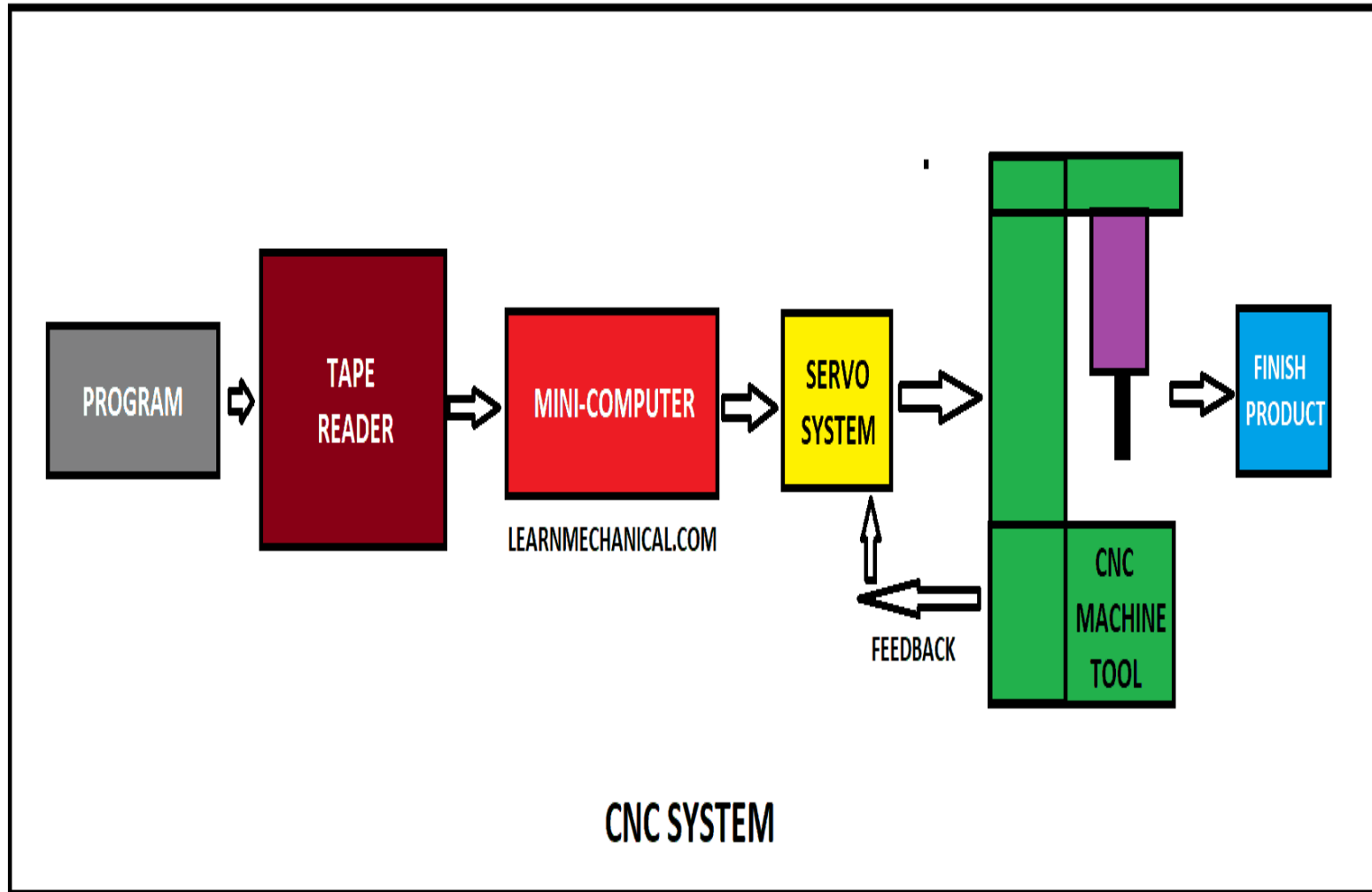
Basic components:

Program

Machine control unit

Machine tool

CNC system:



- Components of cnc system
- Tape reader
- 2) Mini or micro computer
- 3) Computer hardware and Servo system
- Machine tool
- Tape reader:
- The main purpose of tape reader is to read the part program instructions from tape reader once and loading the part program in to mini or micro computer

- Mini or micro computer:
- It is the important unit of CNC SYSTEM.
- It replaces MCU and its functions is to store part program from punched tape and store in memory
- Its converts part program instructions in to action of machine tool
- Computer hardware and servo system:
- Its function is to convert part program instructions in to mechanical motions

- Its function is to perform useful work on the work piece and finally made finished product

Machine tool:

Advantages:

Part program and tape reader used once.

memory storage to store data

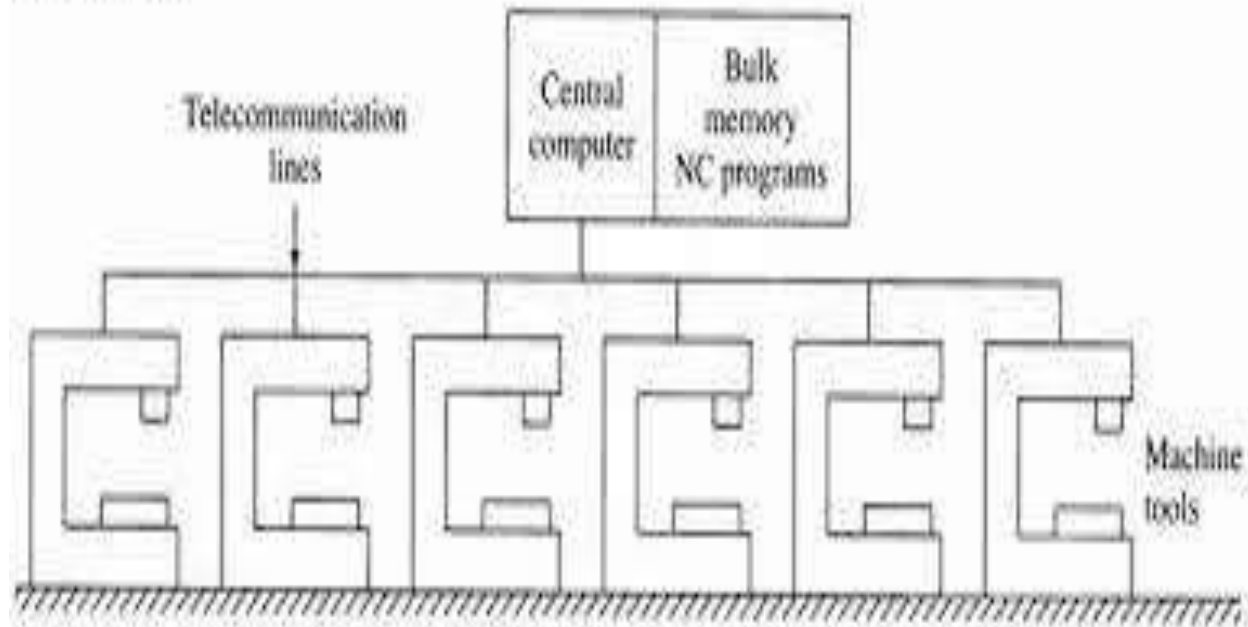
program can be used repeatedly because stored in memory

modifications and design changes can be done easily

more flexible

DNC System:

DNC machine



Components of DNC system:

Central computer

Bulk memory program

Telecommunication lines

Machine tools

UNIT 4: NC Part Programing:

Definition:

- The part program is a sequence of instructions, which describe the work, which has to be done on a part, in the form required by a computer under the control of a numerical control computer program.
- It is the task of preparing a program sheet from a drawing sheet.
- All data is fed into the numerical control system using a standardized format.

- The part programming contains the list of coordinate values along the *X*, *Y* and *Z* directions of the entire tool path to finish the component.
- The program should also contain information, such as feed and speed. Each of the necessary instructions for a particular operation given in the part program is known as an **NC word**.
- A group of such NC words constitutes a complete NC instruction, known as **block**. The commonly used words are *N*, *G*, *F*, *S*, *T*, and *M*.

- The machining data is as follows :
 - Machining sequence classification of process, tool start up point, cutting depth, tool path, etc.
 - Cutting conditions, spindle speed, feed rate, coolant, etc.
 - Selection of cutting tools.

- Steps to be follow for preparing a part program:
 - Determine the startup procedure, which includes the extraction of dimensional data from part drawings and data regarding surface quality requirements on the machined component.
 - Select the tool and determine the tool offset.
 - Set up the zero position for the workpiece.
 - Select the speed and rotation of the spindle.
 - Set up the tool motions according to the profile required.
 - Return the cutting tool to the reference point after completion of work.
 - End the program by stopping the spindle and coolant.

-Methods of part programming can be of two types depending upon the two techniques.

- Manual part programming.
- Computer aided part programming.

Manual Part Programming:

- The programmer first prepares the program manuscript in a standard format.

- Manuscripts are typed with a device known as flexo writer, which is also used to type the program instructions.

- After the program is typed, the punched tape is prepared on the flexo writer.

- Complex shaped components require difficult calculations. This type of programming is carried out for simple machining parts produced on point-to-point machine tool.

-To create a part program manually, need the following information :

(a) Knowledge about various manufacturing processes and machines.

(b) Sequence of operations to be performed for a given component.

(c) Knowledge of the selection of cutting parameters.

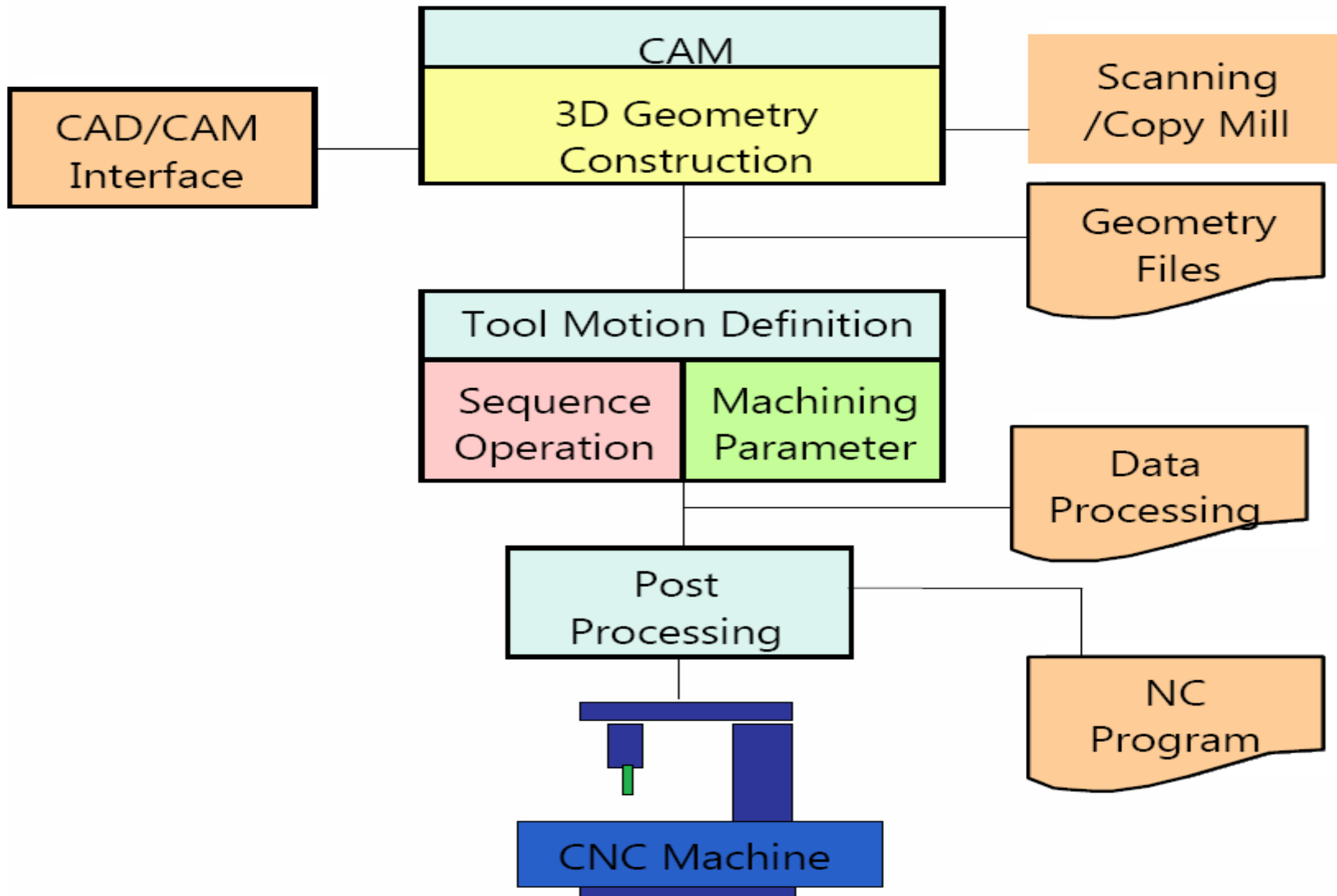
(d) Editing the part program according to the design changes.

(e) Knowledge about the codes and functions used in part programs.

- The programmer communicates with this system through the system language, which is based on words.
- There are various programming languages developed in the recent past, such as APT (Automatically Programmed Tools), ADAPT, AUTOSPOT, COMPAT-II, 2CL, ROMANCE, SPLIT is used for writing a computer programme, which has English like statements.
- A translator known as compiler program is used to translate it in a form acceptable to MCU.

- The programmer has to do only following things :
- Define the work part geometry.
- Defining the repetition work.
- Specifying the operation sequence.

Interactive Graphic System in Computer Aided Part Programming:



Fundamental Elements for Developing Manual Part Programme:

1) Type of Dimensioning System:

Axis Designation:

-The programmer also determines how many axes are available on machine tool. Whether machine tool has a continuous path and point-to-point control system.

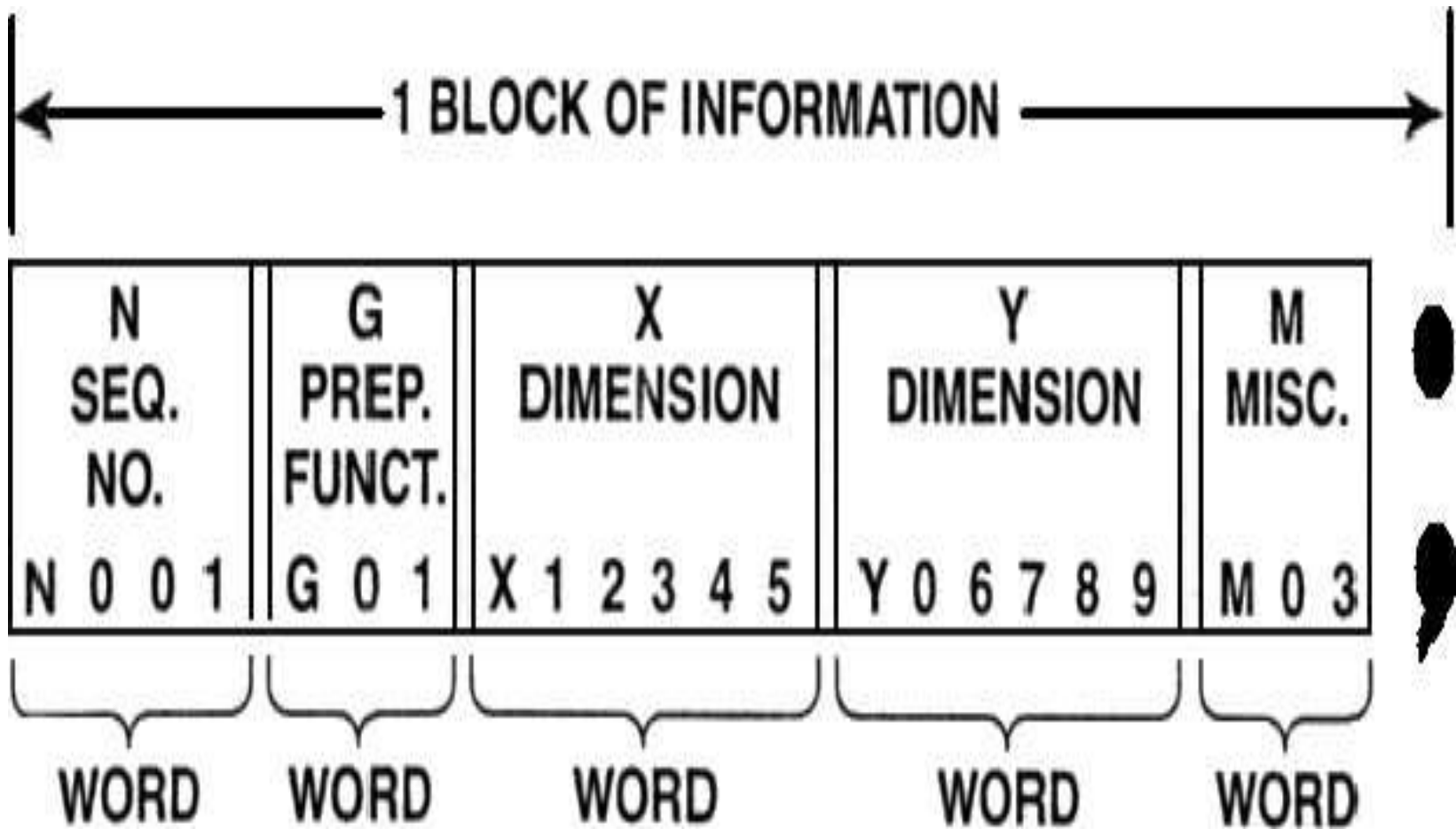
NC Words:

The NC word is a unit of information, such as a dimension or feed rate and so on.

A block is a collection of complete group of NC words representing a single NC instruction. An end of block symbol is used to separate the blocks. NC word is where all the machining data are compiled and where the data are translated in to a language, which can be understood, by the control system of the machine tool.

- NC information is generally programmed in blocks of words.
- Each word conforms to the EIA standards and they are written on a horizontal line.
- If five complete words are not included in each block, the machine control unit (MCU) will not recognize the information; therefore the control unit will not be activated.
- It consists of a character N followed by a three digit number raising from 0 to 999.

Block of Information:



- N001 – represents the sequence number of the operation.
- G01 – represents linear interpolation.
- X12345 – will move the table in a positive direction along the *X*-axis. Y06789 – will move the table along the *Y*-axis.
- M03 – Spindle on CW and
- ; – End of block.

Standard G and M Codes:

The most common codes used when programming NC machines tools are G-codes (preparatory functions), and M codes (miscellaneous functions).

Other codes such as *F*, *S*, *D*, and *T* are used for machine functions such as feed, speed, cutter diameter offset, tool number, etc.

G-codes are sometimes called cycle codes because they refer to some action occurring on the *X*, *Y*, and/or *Z*-axis of a machine tool.

The G-codes are grouped into categories such as Group 01, containing codes G00, G01, G02, G03, which cause some movement of the machine table or head.

Group 03- includes either absolute or incremental programming.

G00 code rapidly positions the cutting tool while it is above the workpiece from one point to another point on a job

During the rapid traverse movement, either the *X* or *Y*-axis can be moved individually or both axes can be moved at the same time. The rate of rapid travel varies from machine to machine.

The total number of G codes are 100

G-Codes (Preparatory Functions)

Code Function

G00 Rapid positioning

G01 Linear interpolation

G02 Circular interpolation clockwise (CW)

G03 Circular interpolation counterclockwise (CCW)

G20 Inch input (in.)

G21 Metric input (mm)

G24 Radius programming

G28 Return to reference point

G29 Return from reference point

G32 Thread cutting

- G40 Cutter compensation cancel
- G41 Cutter compensation left
- G42 Cutter compensation right
- G43 Tool length compensation positive (+) direction
- G44 Tool length compensation minus (-) direction
- G49 Tool length compensation cancels
- G 53 Zero offset or M/c reference
- G54 Settable zero offset
- G84 canned turn cycle
- G90 Absolute programming
- G91 Incremental programming

M-Codes (Miscellaneous Functions):

M or miscellaneous codes are used to either turn ON or OFF different functions, which control certain machine tool operations.

M-codes are not grouped into categories, although several codes may control the same type of operations such as M03, M04, and M05, which control the machine tool spindle.

Some of important codes are given as under with their functions:

Code Function

M00 Program stop

M02 End of program

M03 Spindle start (forward CW)

M04 Spindle start (reverse CCW)

M05 Spindle stop

M06 Tool change

M08 Coolant on

M09 Coolant off

M10 chuck clamping

M11 Chuck - unclamping

M12 Tailstock spindle out

M13 Tailstock spindle in

M17 Tool post rotation normal

M18 Tool post rotation reverse

M30 End of tape and rewind or main program

end M98 Transfer to subprogram

M99 End of subprogram

Tape Programming Format:

Both EIA and ISO use three types of formats for compiling of NC data into suitable blocks of information with slight difference.

Three types of format:

- Word Address Format
- Tab Sequential Format
- Fixed Block Format

1) Word Address Format:

- This type of tape format uses alphabets called address, identifying the function of numerical data followed.
- This format is used by most of the NC machines, also called variable block format.

Ex :

N20 G00 X1.200 Y.100 F325 S1000 T03 M09 <EOB>

or

N20 G00 X1.200 Y.100 F325 S1000 T03 M09;

The MCU uses this alphabet for addressing a memory location in it

2) Tab Sequential Format:

Here the alphabets are replaced by a Tab code, which is inserted between two words. The MCU reads the first Tab and stores the data in the first location then the second word is recognized by reading the record Tab.

A typical Tab sequential instruction block will be as below :

>20 >00 >1.200 >.100 >325 >1000 >03 >09

3) Fixed Block Format:

- In fixed block format no letter address or Tab code are used and none of words can be omitted.
- The main advantage of this format is that the whole instruction block can be read at the same instant, instead of reading character by character
- This format can only be used for positioning work only
- A typical fixed block instruction block will be as below:

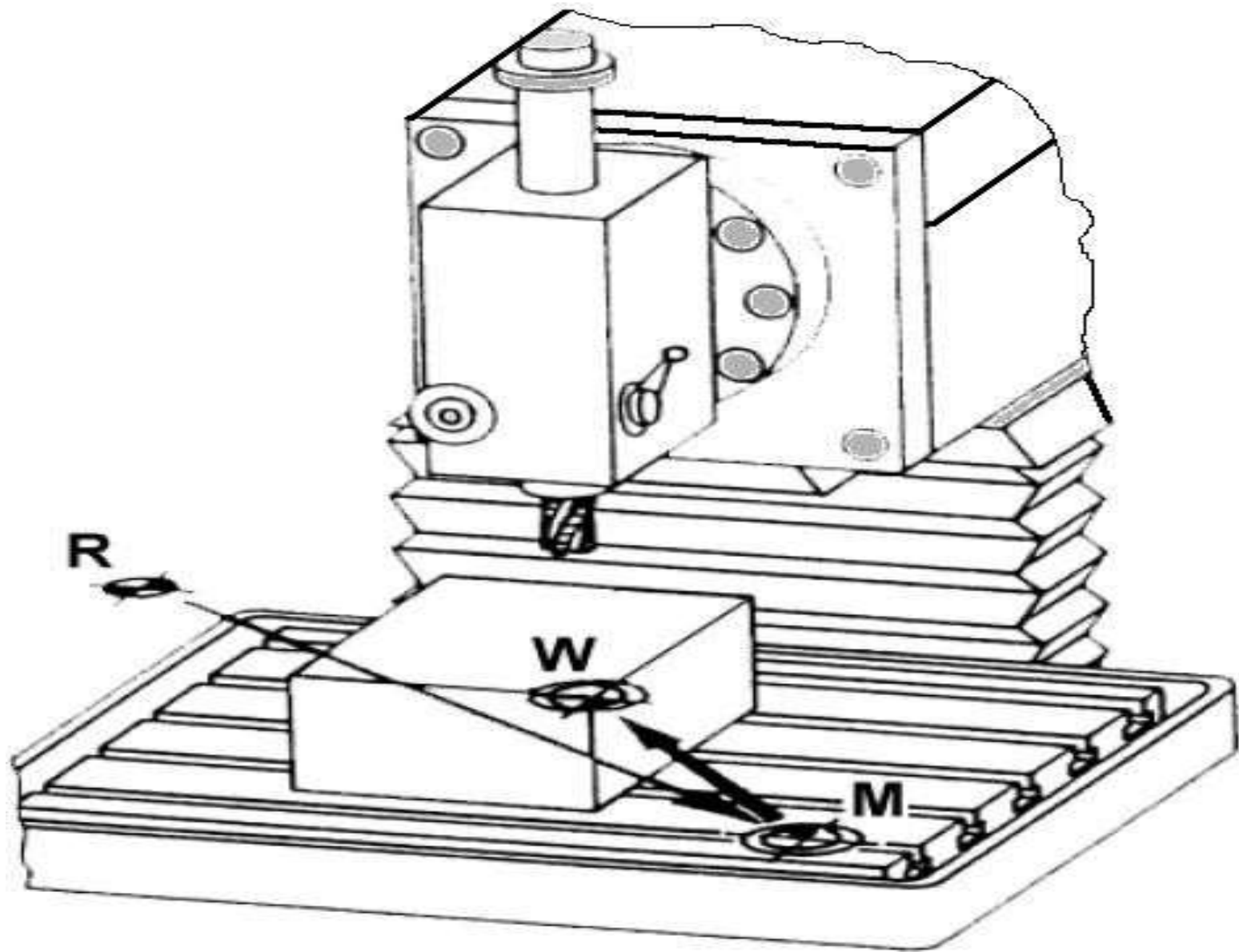
20 00 1.200 .100 325 1000 03 09 <EOB>

Machine Tool Zero Point Setting:

- The machine zero point can be set by two methods by the operator, manually by a programmed absolute zero shift, or by work coordinates, to suit the holding fixture or the part to be machined.

1) Manual Setting

The operator can use the MCU controls to locate the spindle over the desired part zero and then set the X and Y coordinate registers on the console to zero.



Absolute zero shift:

- The absolute zero shift can change the position of the coordinate system by a command in the CNC program.
- The programmer first sends the machine spindle to home zero position by a command in the program.
- Then another command tells the MCU how far from the home zero location, the coordinate system origin is to be positioned.

R = Reference point (maximum travel of machine)

W = Part zero point workpiece coordinate sys

M = Machine zero point (X0, Y0, Z0) of machine coordinate system

The sample commands may be as follows :

N1 G28 X0 Y0 Z0 (sends spindle to home zero position or Return to reference point).

N2 G92 X3.000 Y4.000 Z5.000 (the position the machine will reference as part zero or Programmed zero shift).

Coordinate Word:

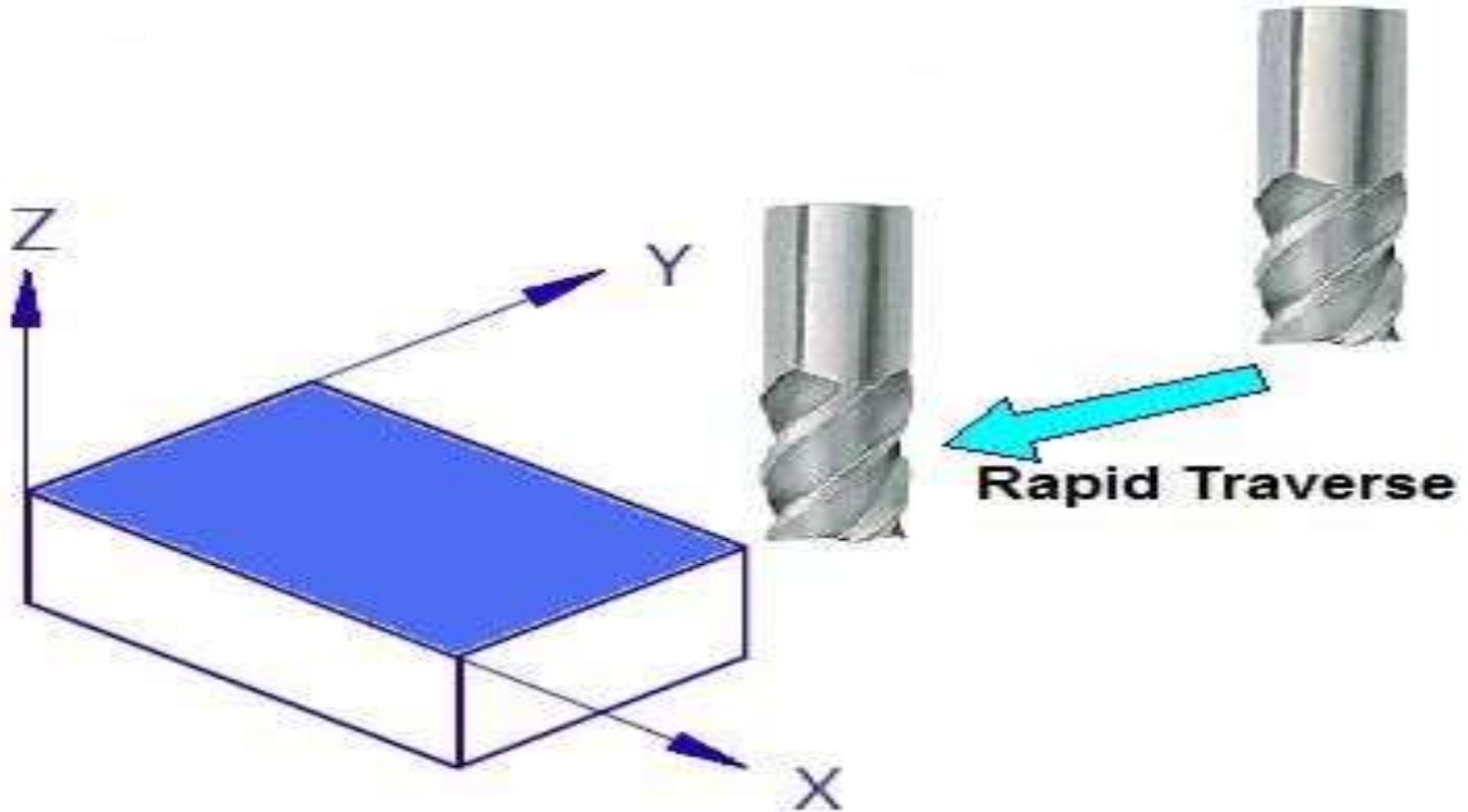
- A coordinate word specifies the target point of the tool movement or the distance to be moved. The word is composed of the address of the axis to be moved and the value and direction of the movement

Example :

- X150 Y-250 represents the movement to (150, - 250). Whether the dimensions are absolute or incremental will have to be defined previously using G-Codes.

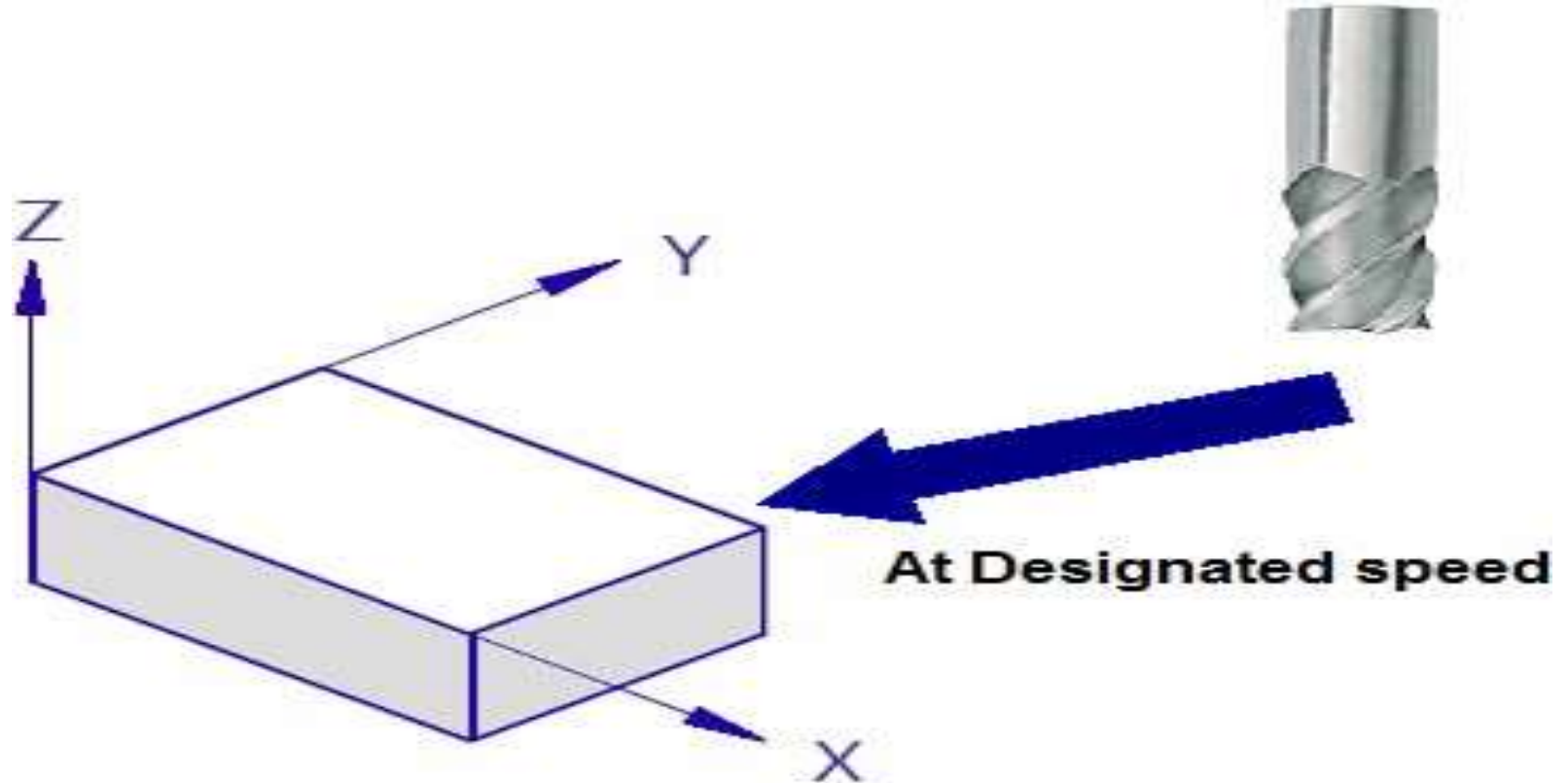
Rapid positioning:

This is to command the cutter to move from the existing point to the target point at the fastest speed of the machine.



Linear Interpolation:

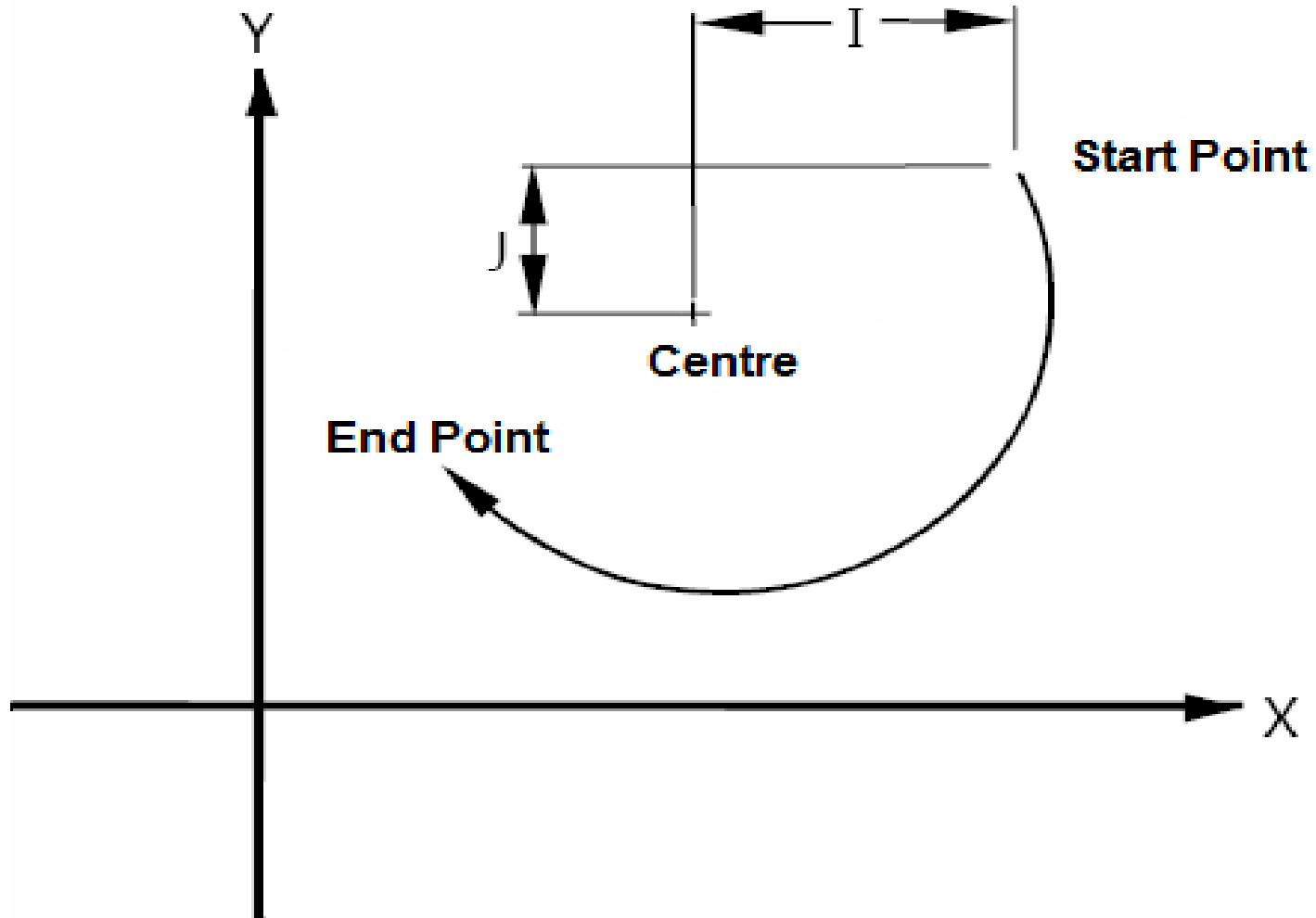
This is to command the cutter to move from the existing point to the target point along a straight line at the speed designated by the F address.



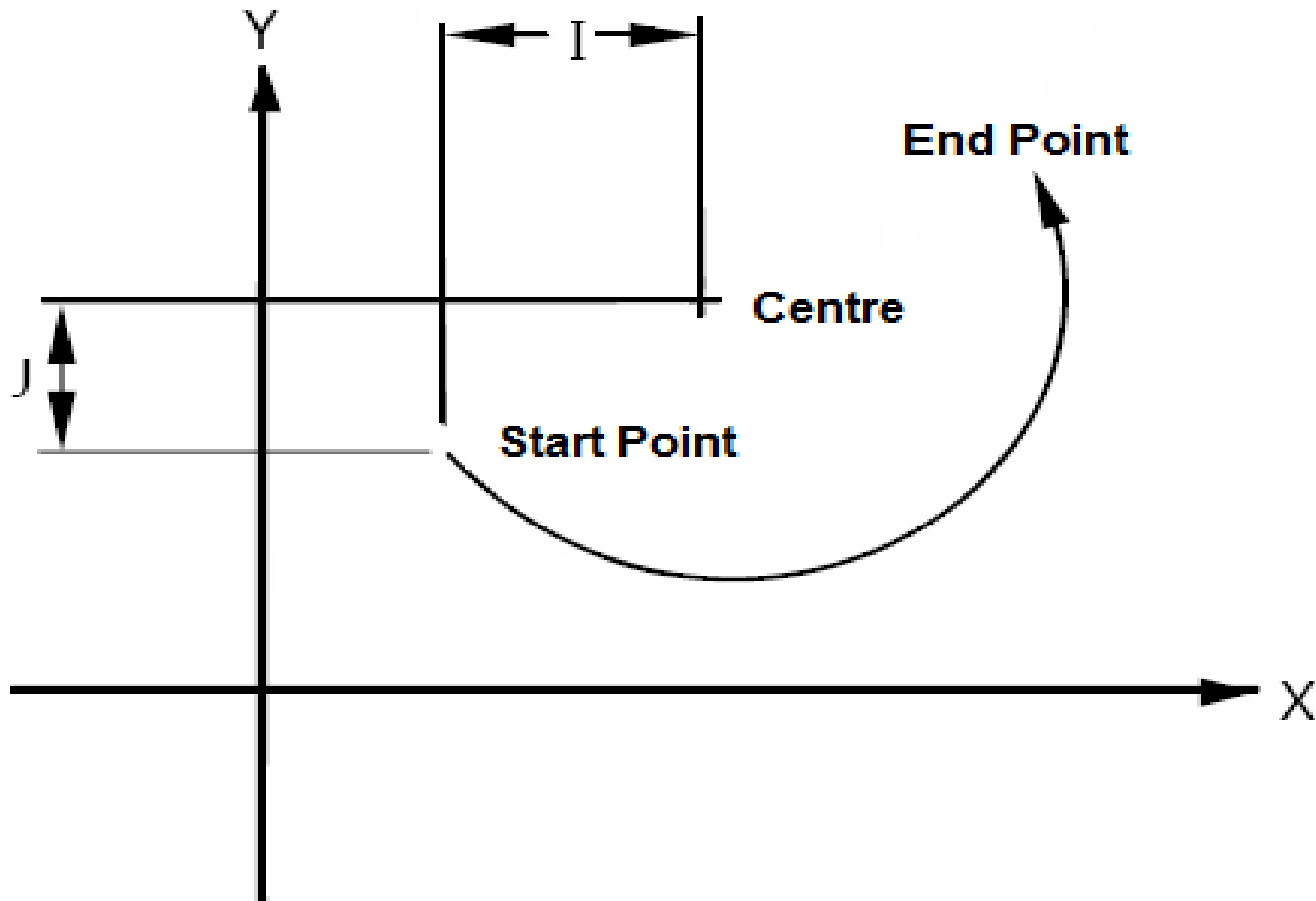
Circular Interpolation:

- This is to command the cutter to move from the existing point to the target point along a circular arc in clockwise direction or counter clockwise direction.
- The parameters of the center of the circular arc is designated by I, J and K addresses.
- I is the distance along the X-axis, J along the Y, and K along the Z.
- This parameter is defined as the vector from the starting point to the center of the arc.

Circular interpolation clock wise:



Counter Clockwise Circular Interpolation:



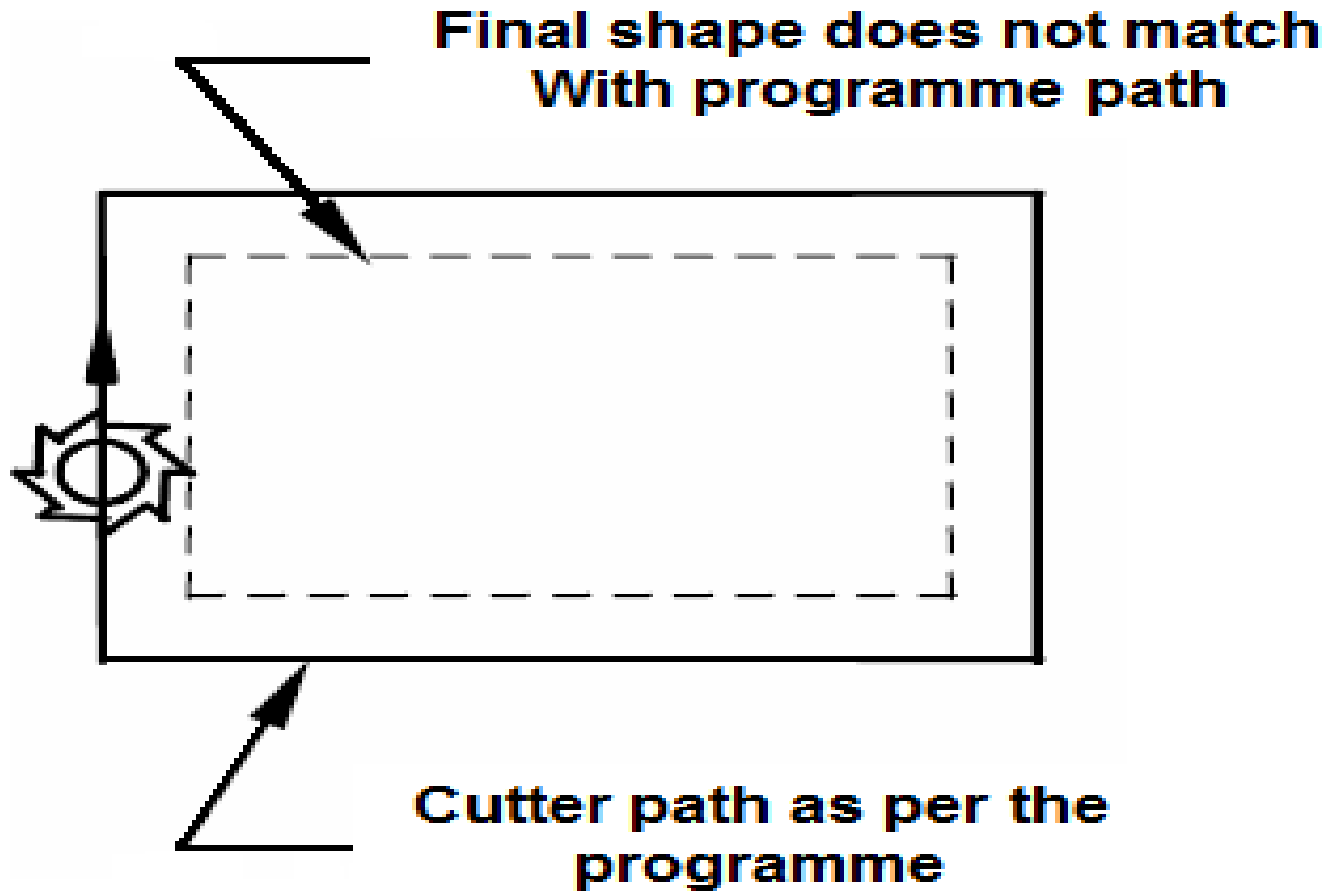
Tool Path with Cutter Compensation:

In NC machining, if the cutter axis is moving along the programmed path, the dimension of the workpiece obtained will be incorrect since the diameter of the cutter has not been taken into account.

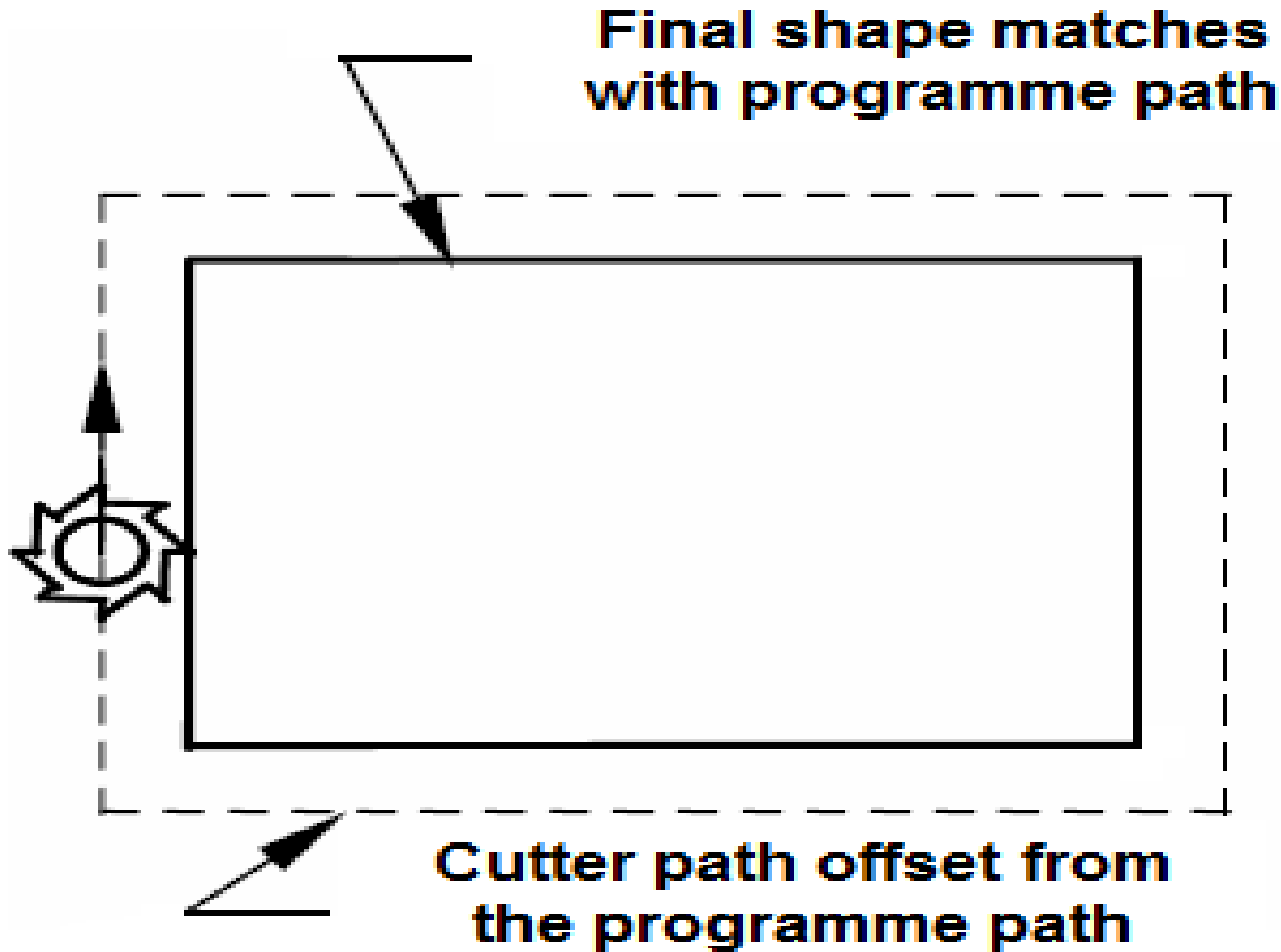
What the system requires are the programmed path, the cutter diameter and the position of the cutter with reference to the contour.

The cutter diameter is not included in the programme. It has to be input to the NC system in the tool setting process.

Tool Path without Cutter Compensation



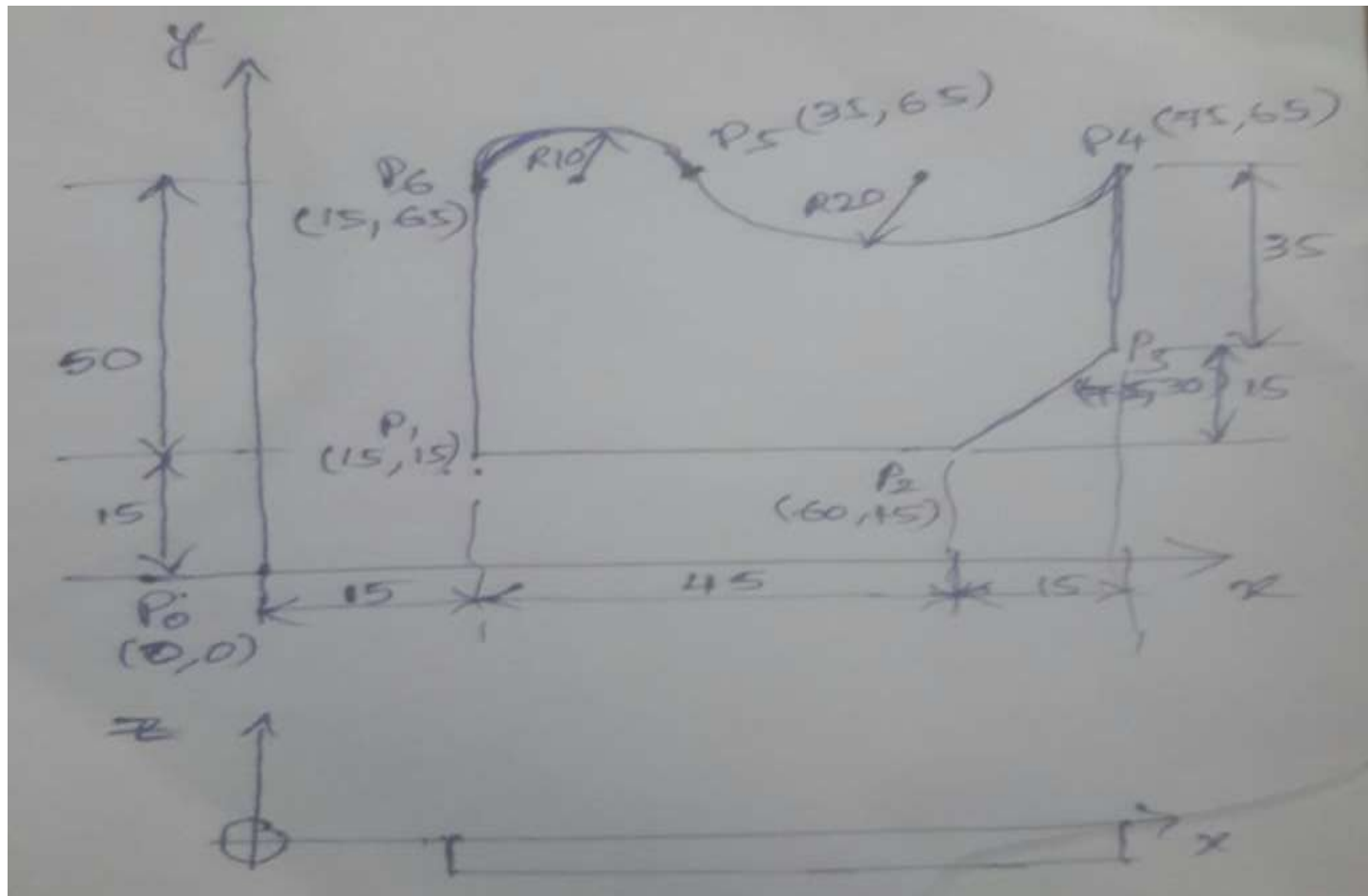
Tool Path with Cutter Compensation



Symbols used:

- % – Main Programme (1 to 9999)
- L – Sub program (1 to 999)/Home position
- N – Sequence of block number.
- Lf – Block end (EOB) means “; or *”
- T – Tool number or Tool station number
- D- Tool offset
- S – Spindle speed F – Feed
- M – Switching function
- G – Transverse commands
- R – Parameters
- I, J, K – Circle parameters
- B/U/R – Radius
- X/Y/Z – Axis coordinates
- P – Passes.

Example:



- N0 G17 (G17-Principle x, y plane)
- N1 G71 (G71-Metro mode program)
- N2 G40 (G40-Cancel cutter radius compensation)
- N3 G80 (G80-Cancel canned cycle)
- N4 G28 U0 V0 W0
- N5 G91 X0 Y0 Z0 (G91-incremental programming)
- N6 G92 (G92-reset to original point)
- N7 M06 T01 (T01-tool change)
- N8 G90 (G90-absolute programming mode)
- N9 G00 X0 Y0 Z0 (G00-rapid move)
- N10 G43 M08 M04 (G43-tool length compensation, M08-coolant ON, M04-spindle ON)
- N11 G00 X12 Y12 M03 S1300
- N12 G01 Z-5 F 500 (G01-rapid tool positioning)
- N13 G42 X15 D11 (G42- cutter radius compensation)
- N14 G01 X60 (G01-rapid move)

- N15 G01 X75 Y30
- N16 G01 Y65
- N17 G02 X35 Y65 I-20 J00 (G02-clock wise circular interpolation)
- N18 G03 X15 Y65 I-10 J00 (G03-counter clock wise
circular interpolation)
- N19 G01 Y15
- N20 G00 Z0
- N21 G91 X0 Y0 Z0 (G91-incremental programming)
- N22 G40 (G40-cancel cutter radius
compensation)
- N23 G80 (G80- cancel canned cycle)
- N24 G49 (G49- tool length compensation
cancel)
- N25 M09 (M09- coolant off)
- N26 M02 (M02- program end)
- N27 M30 (M30- end of the tape)

UNIT 5: ROBOTICS

- The anatomy of industrial robots deals with the assembling of outer components of a robot such as wrist, arm and body.

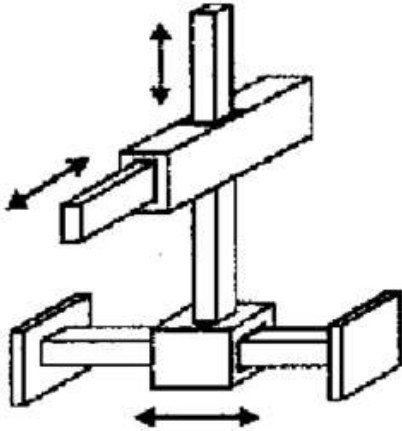
Before jumping into robot configurations, here are some of the key facts about robot anatomy.

- (a) Joints and Links
- (b) Common Robot Configurations

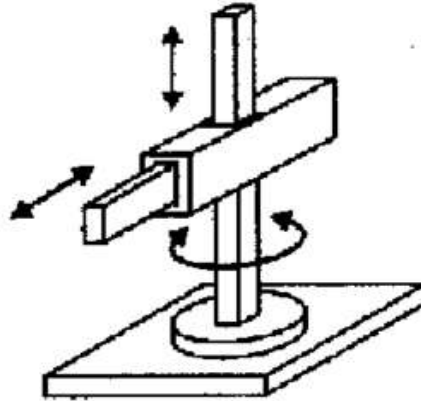
JOINTS AND LINKS:

- A Joint provides controlled relative movement between the input link and
- output link. A robotic link is the rigid component of the robot manipulator.
- • Most of the robots are mounted upon a stationary base, such as the floor.
- From this base, a joint-link numbering scheme may be recognized

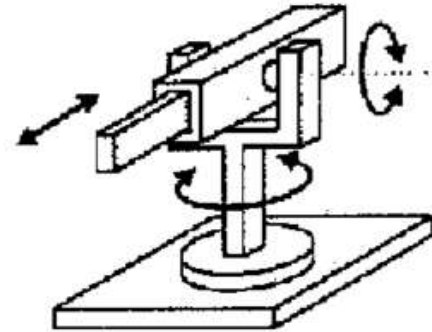
CONFIGURATION OF ROBOT



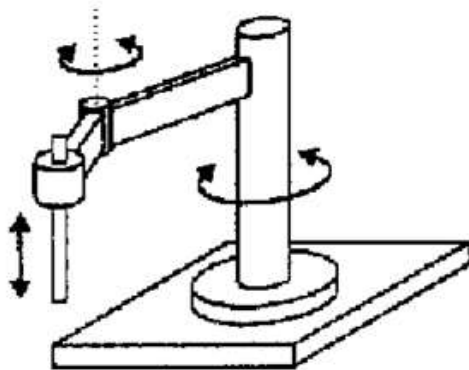
Cartesian Robot



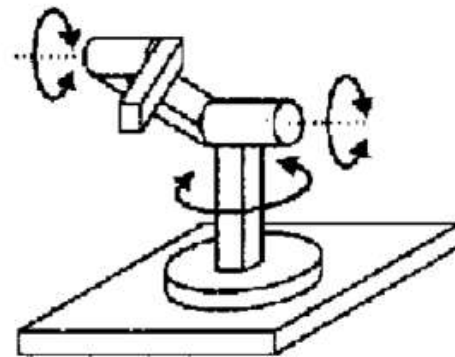
Cylindric Robot



Polar or spheric Robot



SCARA Robot



Angular or anthropomorphic Robot

Polar configuration:

It consists of a sliding arm L-joint, actuated relative to the body, which rotates around both a vertical axis (T-joint) and horizontal axis (R-joint).

Cylindrical configuration:

It consists of a vertical column. An arm assembly is moved up or down relative to the

vertical column. Arm can be moved in and out relative to the axis of the column. Common configuration is to use a T-joint to rotate the column about its axis.

An L-joint is used to move the arm assembly vertically along the column, while an O-joint is used to achieve radial movement of the arm.

Cartesian co-ordinate robot:

It is also known as rectilinear robot and x-y-z robot. It consists of three sliding

joints, two of which are orthogonal O-joints.

- Jointed-arm robot:

It is similar to the configuration of a human arm. It consists of a vertical column that swivels about the base using a T-joint. Shoulder joint (R-joint) is located at the top of the column. The output link is an elbow joint (another R joint).

- SCARA: Its full form is 'Selective Compliance Assembly Robot Arm'.

It is similar in construction to the jointer-arm robot, except the shoulder and

GRIPPER:

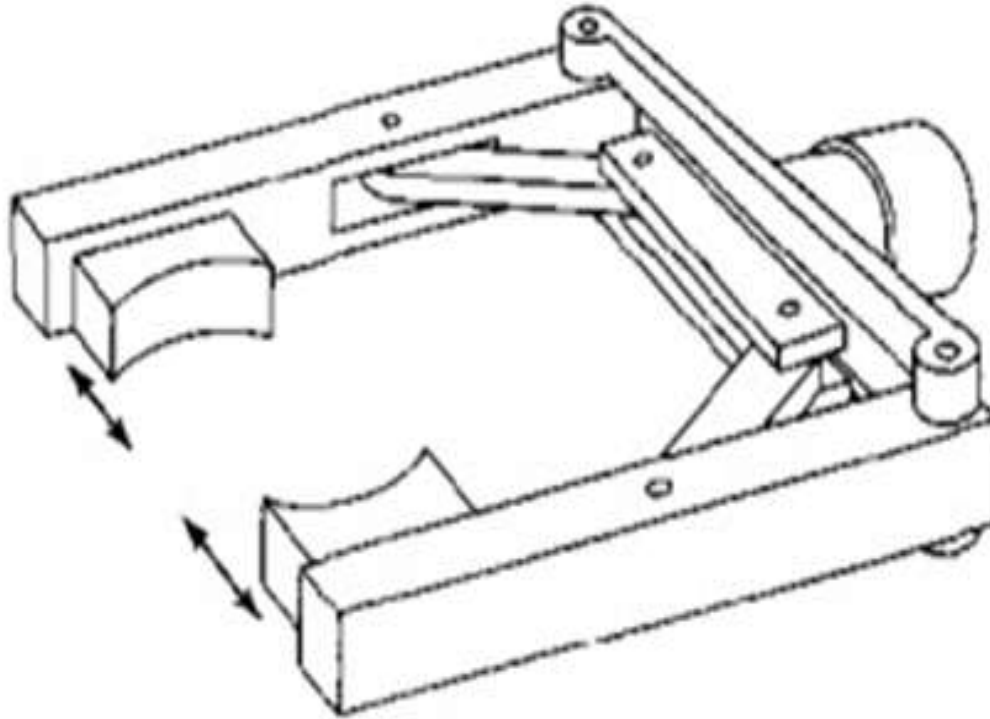


Figure 7.10 Robot mechanical gripper.

Grippers grasp and manipulate objects during the work cycle.

Typically the objects grasped are work parts that need to be loaded or unloaded from one station to another.

It may be custom-designed to suit the physical specifications of the work parts they have to grasp.

Group Technology (GT)

- GT is a theory of management based on the principle that similar things should be done similarly
- GT is the realization that many problems are similar, and that by grouping similar problems, a single solution can be found to a set of problems thus saving time and effort
- GT is a manufacturing philosophy in which similar parts are identified and grouped together to take advantage of their similarities in design and production

Types of Layout

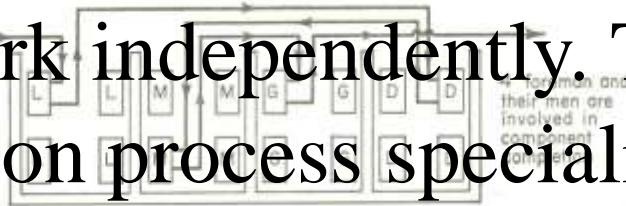
In most of today's factories it is possible to divide all the made components into families and all the machines into groups, in such a way that all the parts in each family can be completely processed in one group only.

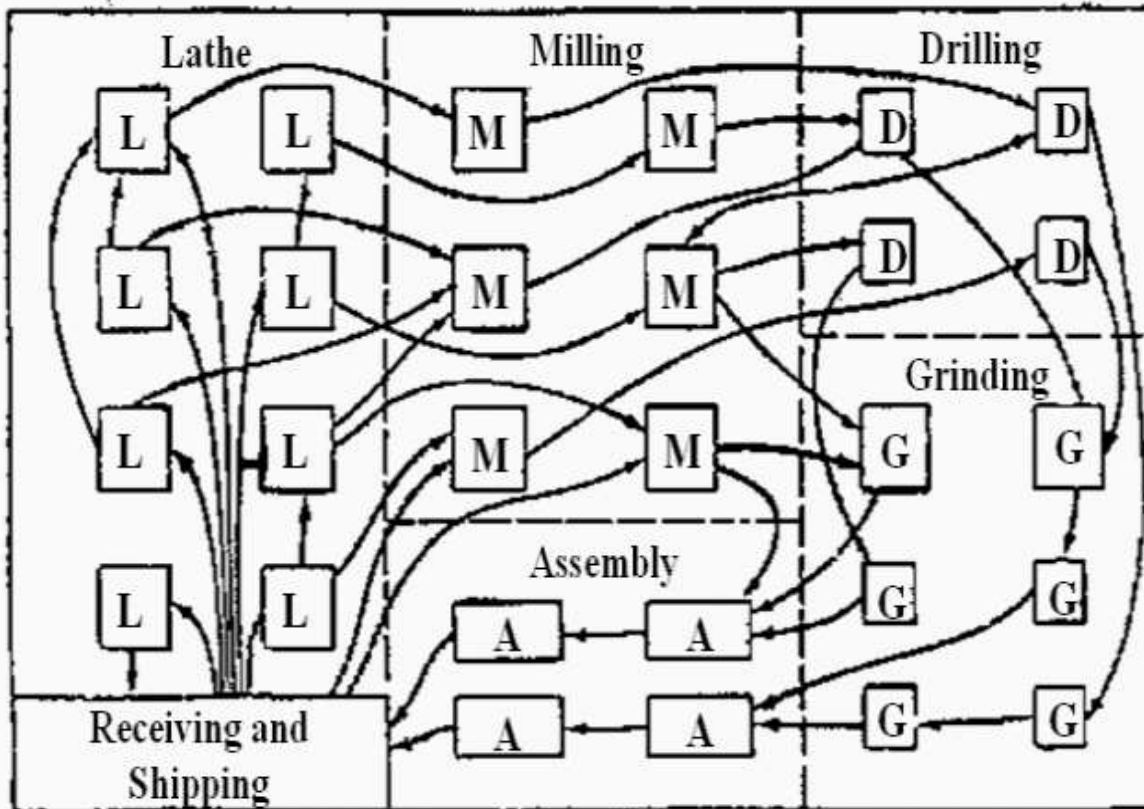
The two main types of layout are:

- Functional Layout
- Group Layout

Functional Layout(or) process layout

- In Functional Layout, all machines of the same type are laid out together in the same section under the same foreman. Each foreman and his team of workers specialize in one process and work independently. This type of layout is based on process specialization.





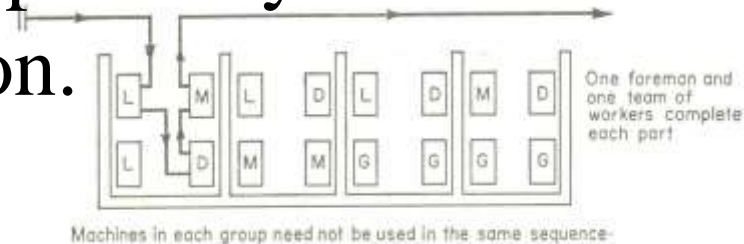
PROCESS-TYPE LAYOUT

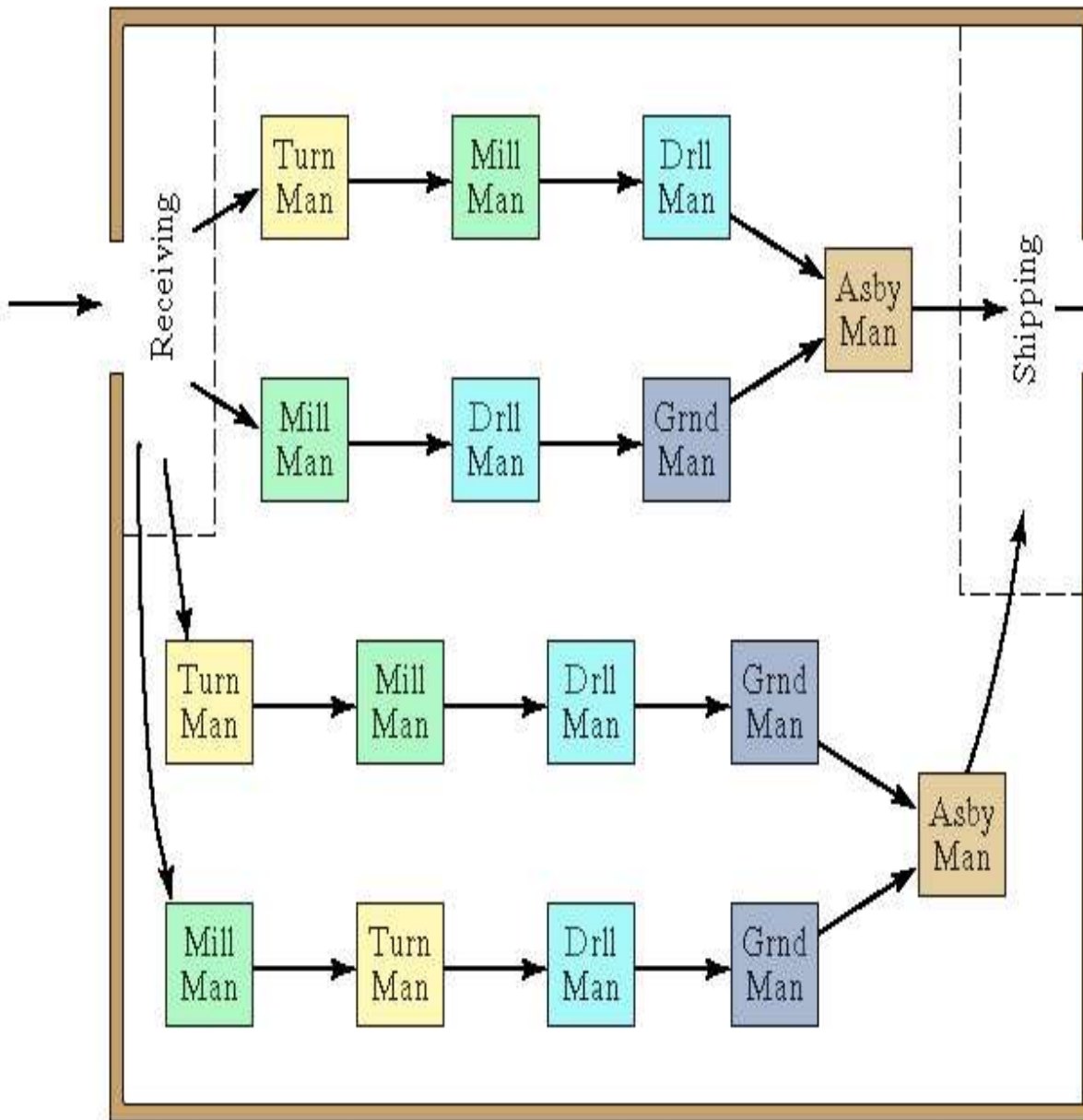
In process lay out all the machine tools of same process are grouped in a single department and placed together.

1. This results in a significant amount of material handling.
2. A large in process inventory.
3. Usually more setups than necessary.
4. Long lead times.

Group Layout

- In Group Layout, each foreman and his team specialize in the production of one list of parts and co-operate in the completion of common task. This type of layouts based on component specialization.



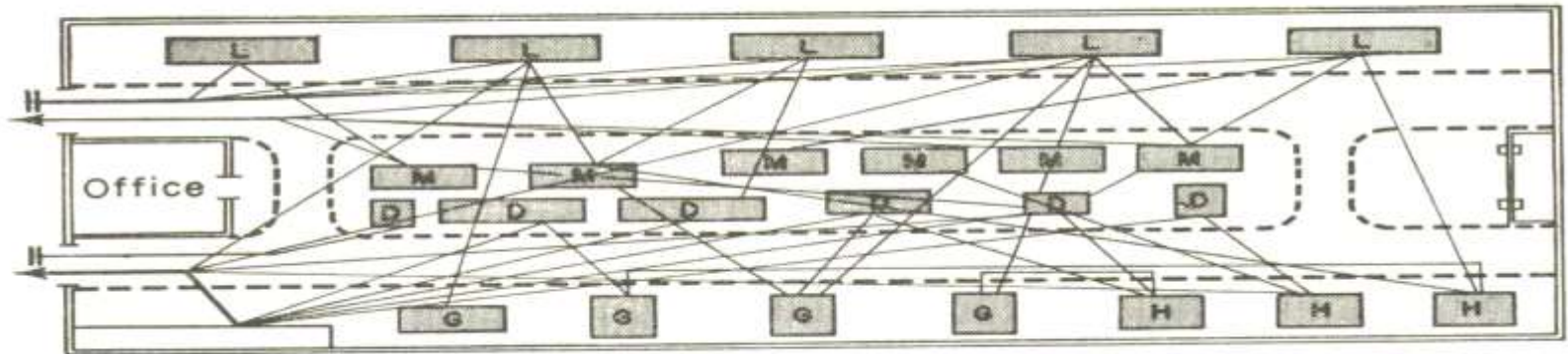


Advantages are gained in the form of reduced

1. Work piece handling
2. Lower setup times
3. Less in process inventory
4. Less floor space and shorter lead times

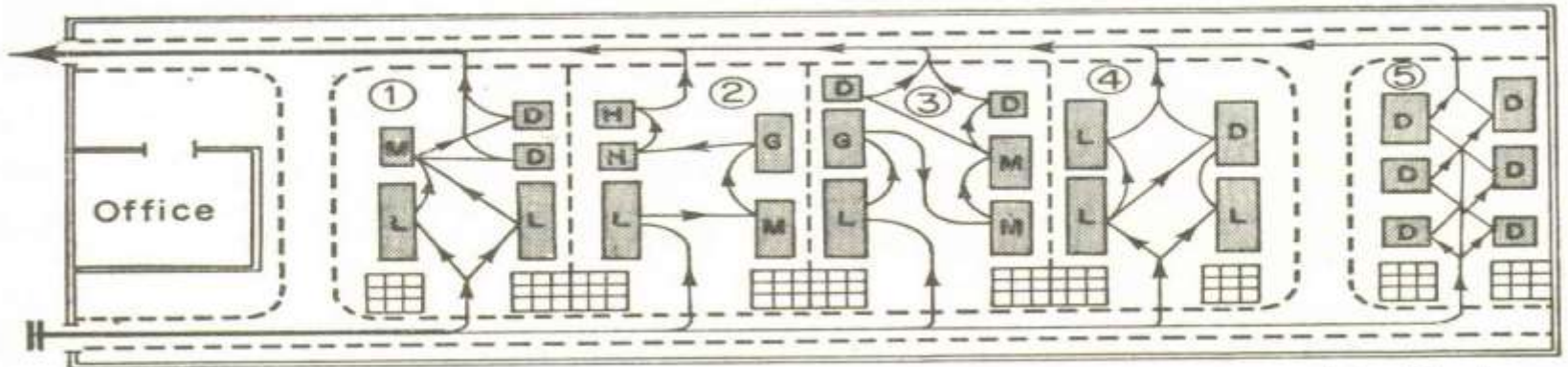
The Difference between group and functional layout:

(i) Complicated



N.B. 'Functional layout' - machines grouped by process type

(ii) Simplified
(By laying out in family machine 'groups')



N.B. 'Group layout' - machines grouped for component 'Families'