COMPUTER NUMERICAL CONTROL OF MACHINE TOOLS

Laboratory for Manufacturing Systems and Automation
Department of Mechanical Engineering and Aeronautics
University of Patras, Greece

LMS

Dr. Dimitris Mourtzis
Associate professor

Patras, 2019

Table of Contents

An Introduction to Numerical Control Machinery

Why Automate?

Benefits of Automation

- Save **time**
- Reduce **human error**
- Make processes more **efficient**
- Eliminate **redundancies** and **wasted effort**
- Implement **process improvements** faster
- Use **skilled employees** for more valuable tasks
- **Avoid disruptions** when new employees are hired or longtime employees retire

Ann Mazakas, ''The Art of Automation'', courses materials of ESPRIT World Conference on CNC

Numerical Control (NC)

- One of the most important developments in manufacturing automation is **Numerical Control (NC):**

> **"A Numerical Control machine is a machine positioned automatically along a pre-programmed path by means of coded instructions"**
>
> *(Computer Numerical Control: Concepts & Programming, W. Seames, 2001)*

- **Numerical Control (NC)** helps solve the problem of making Manufacturing Systems (MFG) more flexible

Numerical Control (NC)

NC is a general term used for **N**umerical **C**ontrol

CNC refers specifically to **COMPUTER NUMERICAL CONTROL**

CNC machines are all NC machines

BUT not all NC machines are CNC machines

NC Definition, it’s Concepts And Advantages

Advantages

- **Flexibility** that speeds changes in design
- Better **accuracy** of parts
- **Reduction in parts handling**
- Better **uniformity** of parts
- Better **quality** control
- **Improvement in manufacturing control**

Disadvantages

- Increase in **electrical maintenance**
- High **initial investment**
- Higher per-hour **operating cost** than traditional machine tools
- **Retraining** of existing personnel

Numerical Control Systems' Components

Components of traditional NC systems

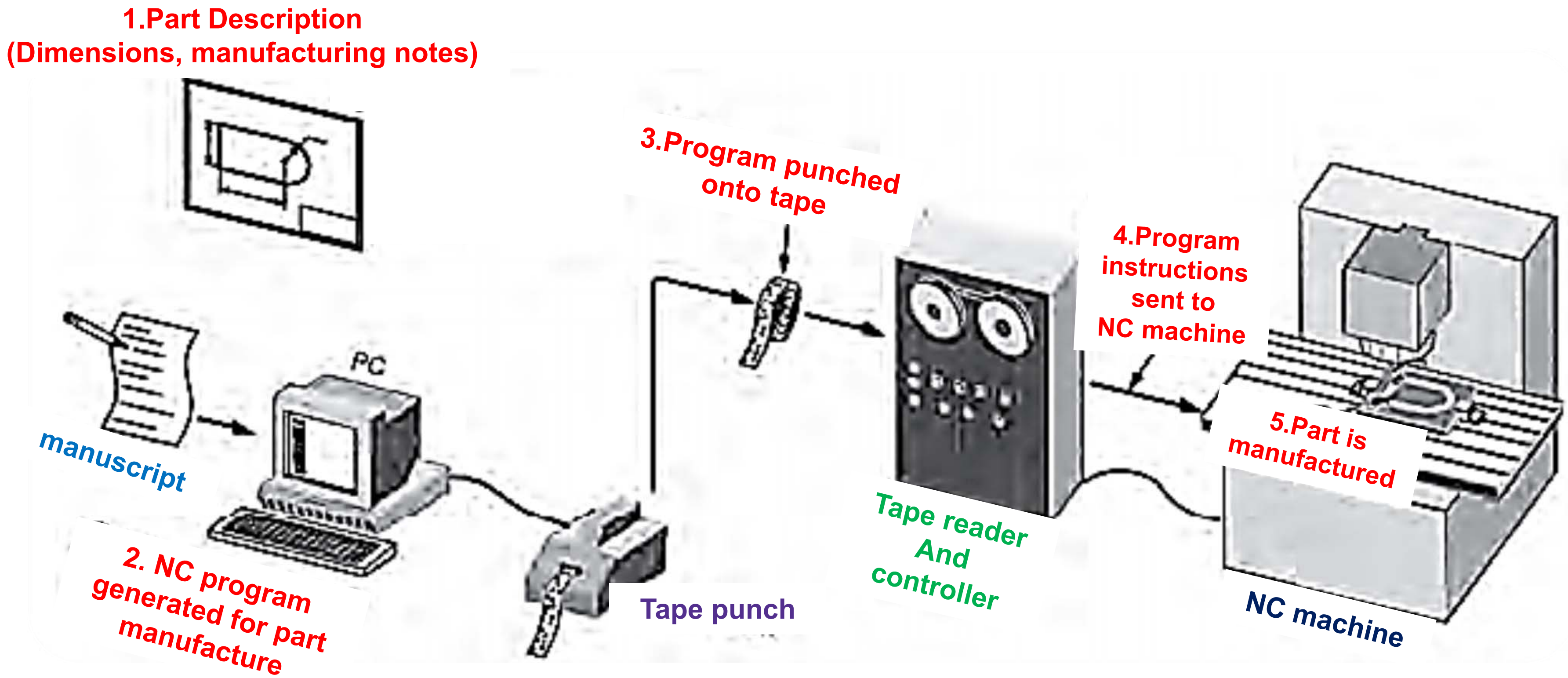


Figure 1-1 : Components of traditional NC systems

(Seams W., "Computer Numerical Control, Concepts & Programming,4th edition)

Components of Modern NC systems

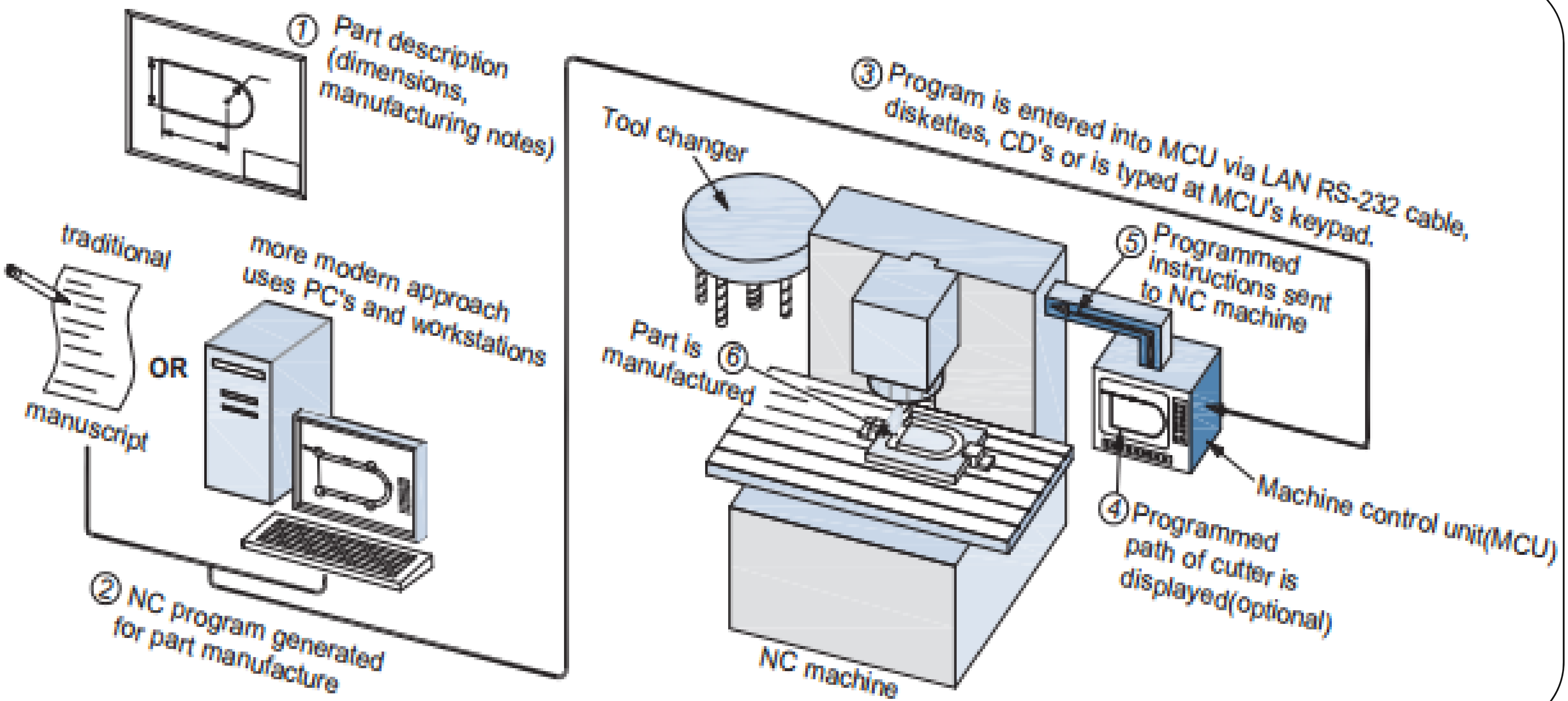


Figure 1-4(a) :Components of Modern CNC Systems
(Introduction to Computer Numerical Control, 4th Edition, J.V. Valentino, E.V. Goldenberg, 2007)

Direct Numerical Control

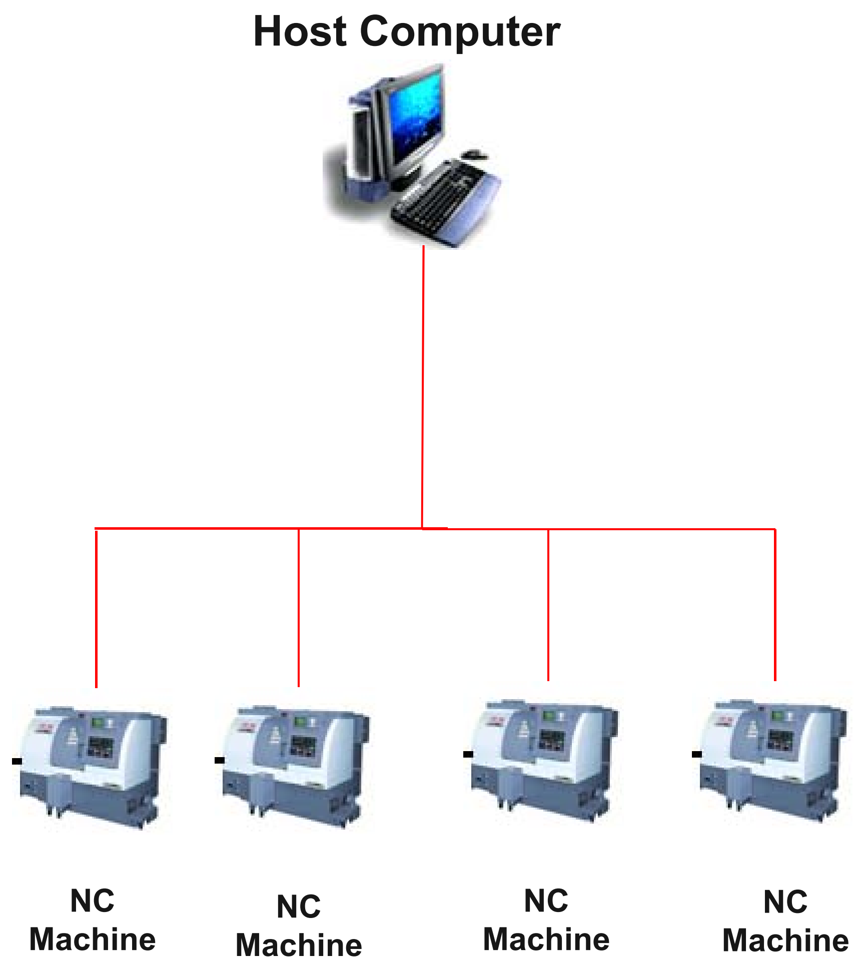


Figure 1-9: Direct numerical control
(adapted from Seams W., "Computer Numerical Control, Concepts & Programming,4th edition)

Direct Numerical Control:

A computer is used as a partial or complete controller of one or more NC machines

Expensive mainframe or mini-computers were required in the past

Due to cost the use of DNC was limited to large companies

Powerful PCs given rise to affordable PC-based DNC systems

Most of PC-based DNC systems running on MS Windows OS

Distributed Numerical Control

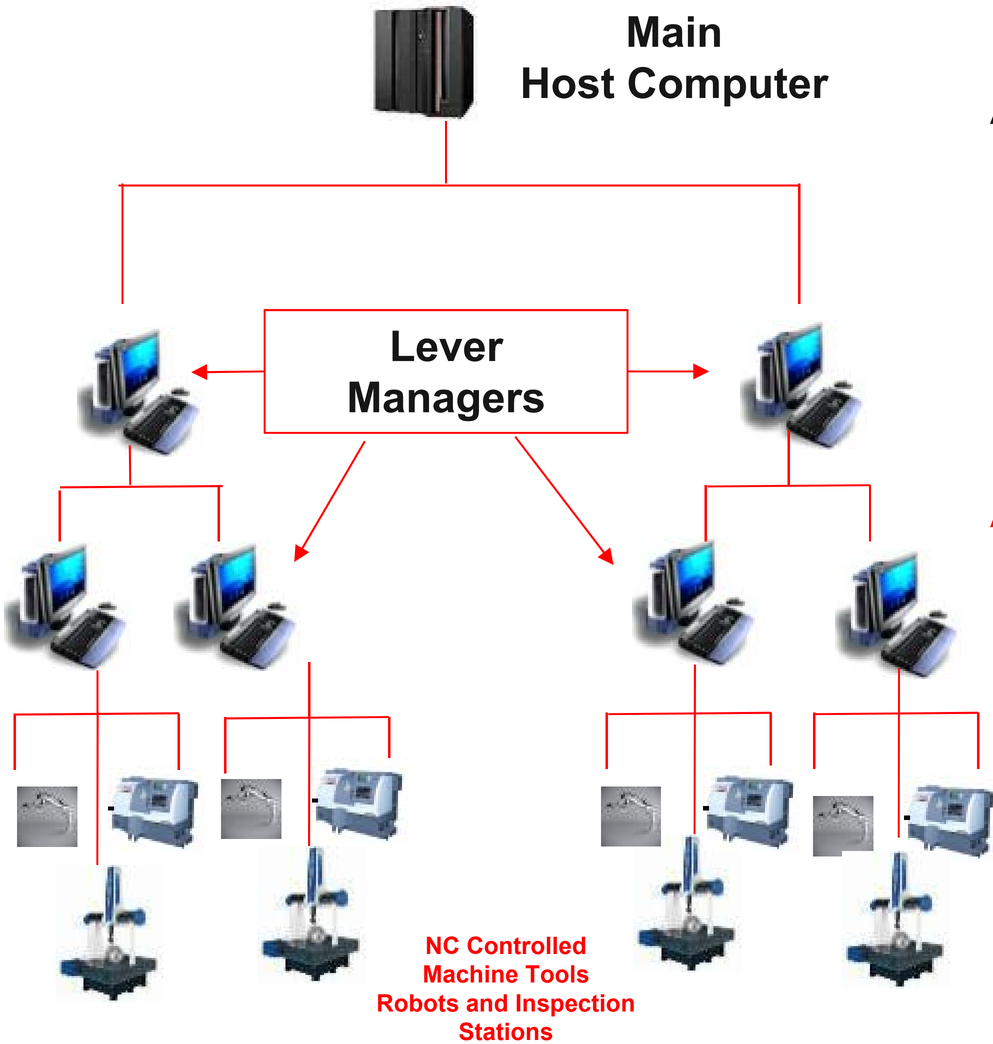


Figure 1-10: Distributed numerical control

(adapted from Seams W., "Computer Numerical Control, Concepts & Programming,4th edition)

Distributed Numerical Control: A network of computers is used to coordinate the operation of a number of CNC machines

Ultimately an **entire factory** can be coordinated in this manner

Alternative System 1: NC program is transferred in its entirety from a host computer directly to machines controller

Alternative System 2: NC program is transferred from a mainframe or a host computer to a PC on the Shop Floor, stored and used when needed → transferred to machine controller

Financial Rewards of CNC Investment

Financial Rewards of CNC Investment

Using Payback Period to Estimate Investment Efficiency

- The **Payback Period calculation** **estimates the number of years required to recover the net cost of the CNC machine tool :**

$$\text{Payback Period} = \frac{\text{Net cost of CNC} - \text{Net cost of CNC} \times \text{Tax Credit}}{\text{Savings} - \text{Savings} \times \text{Tax Rate} + \text{Yearly Depreciation of CNC} \times \text{Tax Rate}}$$

Financial Rewards of CNC Investment

Using ROI to Estimate Investment Efficiency

- The **ROI calculation** predicts what **percent of the net cost of the CNC will be recovered each year**:
- The **ROI calculation** accounts for the useful life of the CNC machine tool

$$\text{ROI} = \frac{\text{Average Yearly Savings} - \text{Net cost of CNC/ Years of life}}{\text{Net cost of CNC}}$$

Example

- Given the investment figures in Table 1-1 for implementing a new CNC machine tool, determine the payback period and the annual return on investment.

  (The CNC is conservatively estimated to have a useful life of 12 years)

Table 1-1

Initial Investment ($)	One-time savings in tooling ($)	Net Cost of CNC ($)	Average yearly savings ($)	Tax Credit (10%)	Tax Rate (46%)	Yearly depreciation of CNC ($)
130,250	35,000	95,250	63,100	0,1	0,46	10,900

Example

Payback Period = $\dfrac{95{,}250 - 95{,}250 \times 0.1}{63{,}100 - 63{,}100 \times 0.46 + 10{,}900 \times 0.46}$

Payback Period = 2.19 years

- This calculation estimates that the net cost of the CNC will be recovered in 2.19 years

ROI= $\dfrac{63{,}100 - 95{,}250 / 12}{95{,}250}$ ➡ **ROI=** 0.57

- This calculation estimates that the investor can expect **57%** of the net cost of the CNC or **(.57 x $95,250) = $54,293** to be recovered each year if the CNC machine's useful life is 12 years

Numerical Control Systems

CNC Components

- A CNC machine consists of **two major components:**

CNC components

The Machine-Tool

Controller – Machine Control Unit (MCU)

NOTE

MCU is an **on-board** computer

MCU and **Machine Tool** may be manufactured by the same company

(W. S. Seames Computer Numerical Control: Concepts and Programming)

Controllers

- Each **MCU** is manufactured with a standard set of **build in codes**
- Other codes are added by the machine tool builders

Program codes vary somewhat from machine to machine

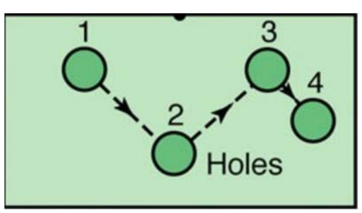
- Every CNC machine is a **collection of systems coordinated** by the **controller**

Types of Control Systems

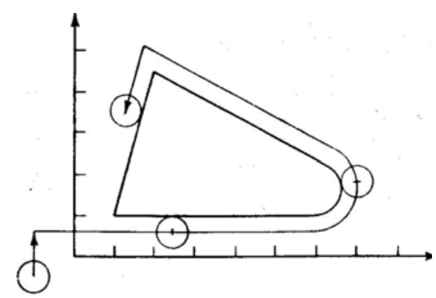
- There are **two types of control systems** used on CNC machines:

Control Systems

Point – to – Point Systems



Continuous – Path Systems



Types of Control Systems

Point – to – Point machines:

I. Move in straight lines

II. They are **limited in practical sense to hole operations**:

- Drilling
- Reaming
- Boring etc

III. Straight milling cuts parallel to a machine axis

IV. When making an axis move all affected drive motors run at the same speed

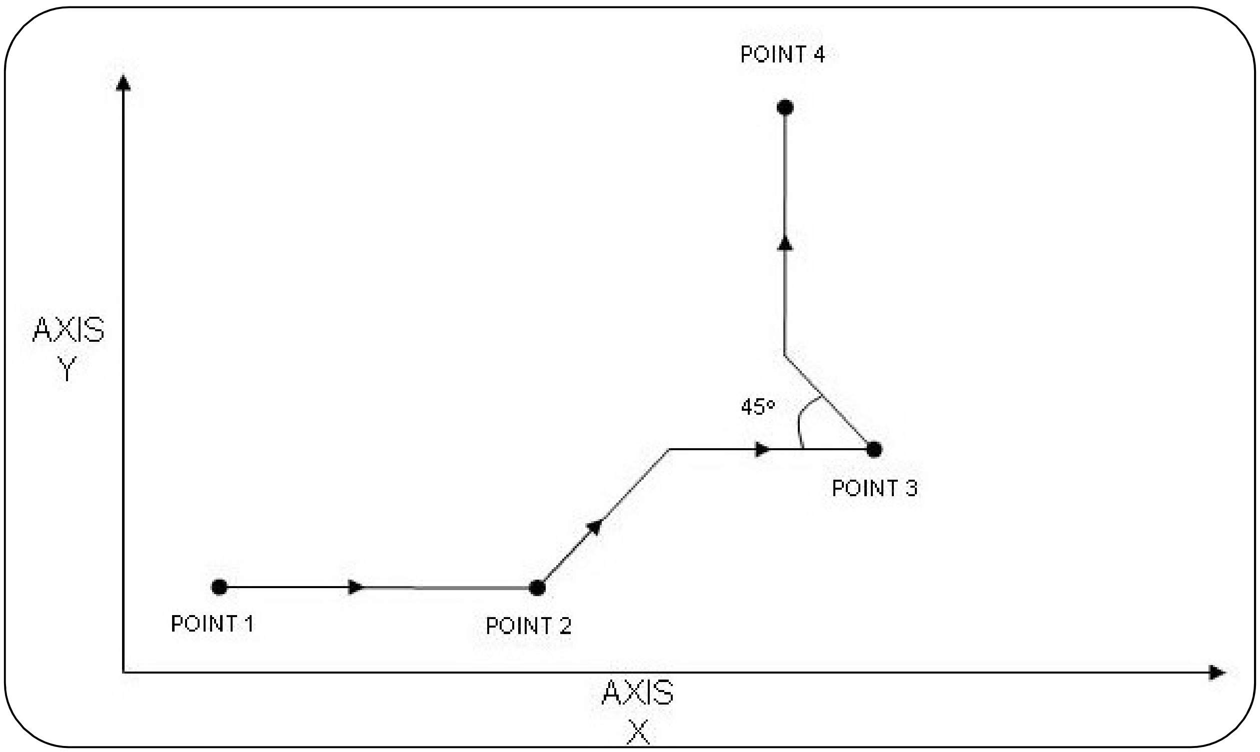
Cutting of 45^o angles is possible
BUT
not angles or arcs other than 45^o angles

(W. S. Seames Computer Numerical Control: Concepts and Programming)

Types of Control Systems

Point – to – Point machines example

- Move to (X1, Y1)
- Move to (X2, Y1)
- Move to (X3’, Y3) where X3’ < X3
- Move to (X3, Y3)
- Move to (X4, Y4’) and move to (X4, Y4)



(W. S. Seames Computer Numerical Control: Concepts and Programming)

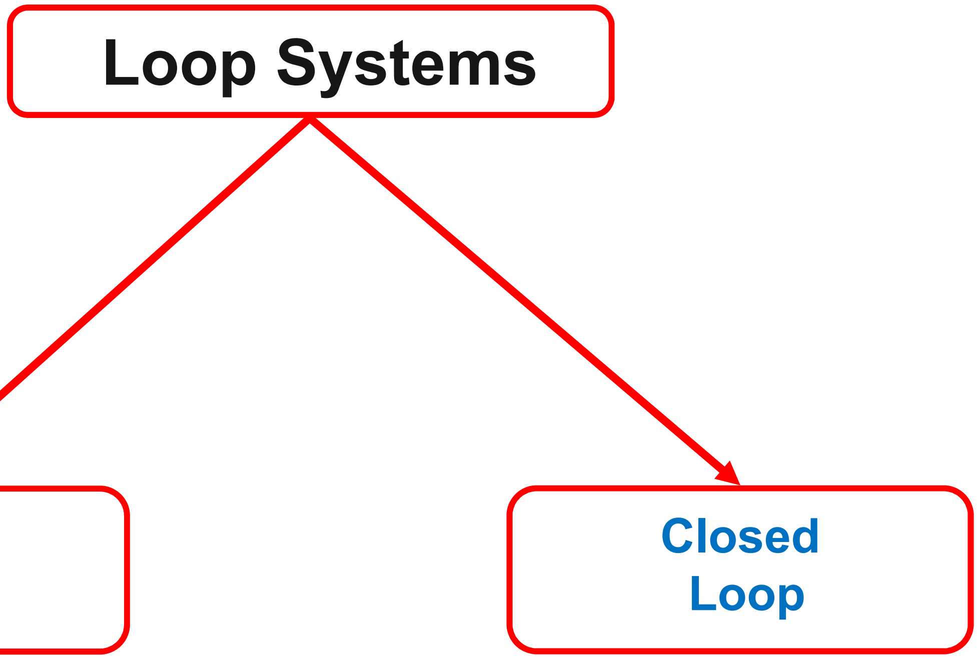
Types of Control Systems

- **Point – to – Point Machines where common**
- Their **electronics** where **less expensive** to produce
- The **machine tools** where **less expensive** to acquire
- Technological advancements have **narrowed the cost difference** between **point – to point** and **continuous – path** machines

Most CNC machines now manufactured are of continuous – path type

Loop Systems

- **Loop systems** are **electronic feedback systems** that send and receive electronic information from the drive motors



- The type of system used **affects the overall accuracy** of the machine
- **Open Loop** use **Stepper Motors**
- **Closed Loop** usually use **Hydraulic, AC and DC Servos**

Loop Systems For Controlling Tool Movement

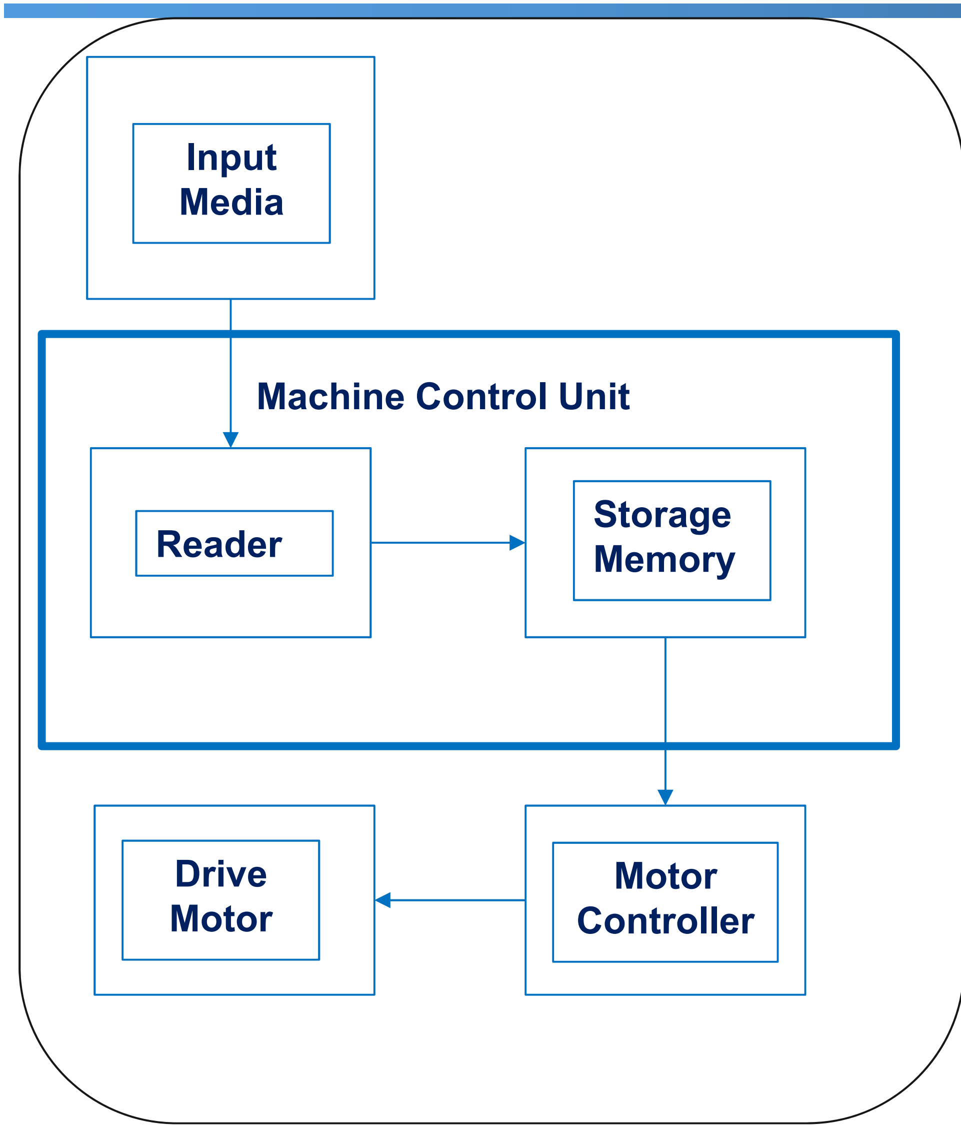


Figure 2-7: An Open – Loop system

Open – Loop System:

- The machine receives its information from the **reader** and stores it in the **storage device**
- When the information is needed it is sent to the **drive motor (s)**
- After the motor has completed its move a signal is sent back to the storage device telling it **that the move has been completed** and the next instruction may be received
- There is ***no process to correct for error*** induced by the drive system

Loop Systems For Controlling Tool Movement

Open – Loop System

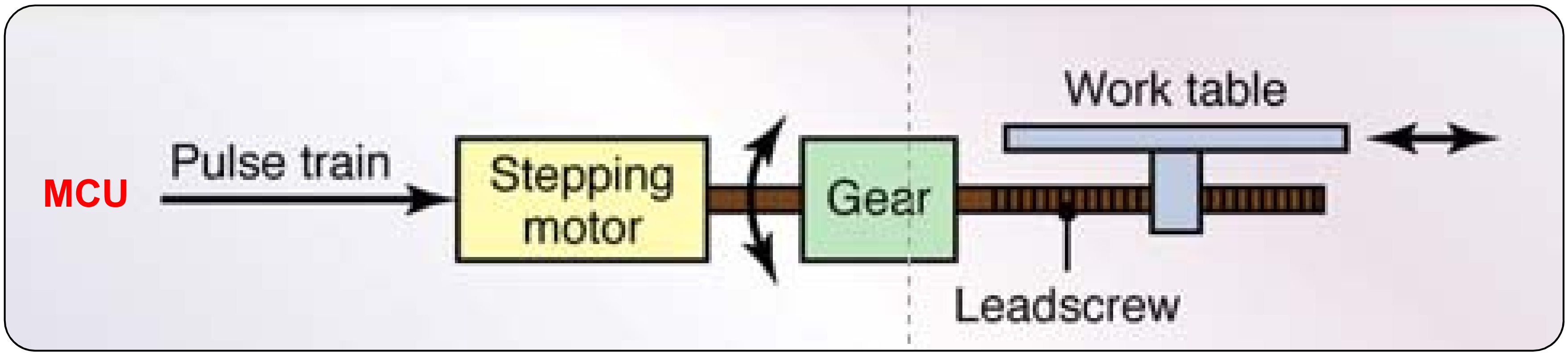


Figure 2-6: An open-loop control system for a numerical-control machine
(Source: Manufacturing, Engineering & Technology, Fifth Edition,S. Kalpakjian and S. R. Schmid)

Loop Systems For Controlling Tool Movement

Open – Loop System

- An open loop system **utilizes stepping motors to create machine movements**. These motors rotate a ***fixed amount, usually 1.8°, for each pulse received***.

- **Stepping motors** are driven by **electrical signals coming from the MCU**. The motors are connected to the machine table ball-nut lead screw and spindle

- Upon receiving a signal, they move the table and/or spindle a fixed amount. The motor controller sends **signals** back indicating the motors have **completed the motion**

The feedback, however, is not used to check how close the actual machine movement comes to the exact movement programmed

Loop Systems For Controlling Tool Movement

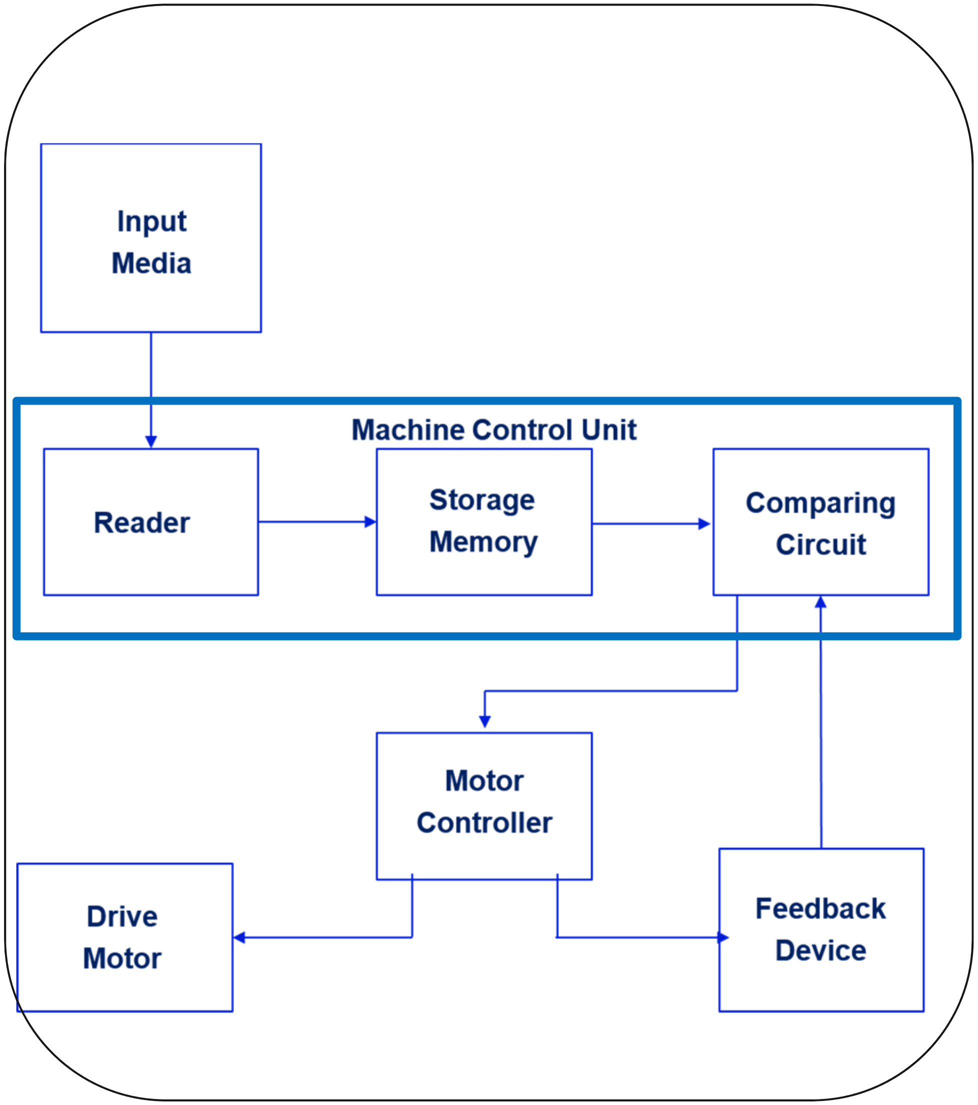


Figure 2-9: A Closed – Loop system

Closed – Loop System:

- The machine receives its information from the reader and stores it in the storage device
- When the information is sent to drive motor the motor's position is **monitored** by the system and compared to what was sent
- If an error is detected the necessary **correction** is sent to the drive system
- If the **error is large** the machine may **stop** executing the program for correcting the inaccuracy
- Most errors produced by the drive motors are eliminated
- **Advanced Stepper Motors make possible extremely accurate Open – Loop Systems and less HW**

Loop Systems For Controlling Tool Movement

Closed – Loop System

DAC : "digital-to-analog converter"

MCU

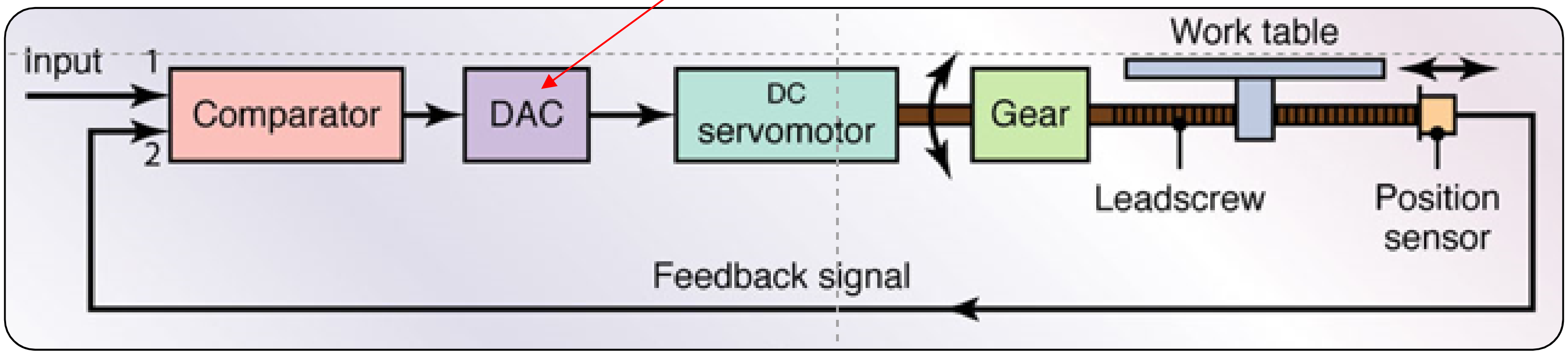


Figure 2-8: A closed-loop control system for a numerical-control machine

(Source: Manufacturing, Engineering & Technology, Fifth Edition,S. Kalpakjian and S. R. Schmid)

Loop Systems For Controlling Tool Movement

Closed – Loop System

- Special motors called servos are used for **executing machine movements** in closed loop systems

- Motor types include **AC servos**, **DC servos**, and **hydraulic servos**. **Hydraulic servos**, being the most powerful, are used on large CNC machines. **AC servos** are next in strength and are found on many machining centers

- A servo does not operate like a pulse counting stepping motor. The speed of an AC or DC servo is variable and depends upon the amount of current passing through it

- The **speed** of a hydraulic servo depends upon the **amount of fluid passing through it**. The strength of current coming from the MCU determines the speed at which a servo rotates

(W. S. Seames Computer Numerical Control: Concepts and Programming)

The Cartesian Coordinate System

The Cartesian Coordinate System in machines

- The **basis for all machine movement** is the Cartesian Coordinate system
- On a machine tool an **axis is a direction of movement**
- In a Two – Axis Milling Machine:
  - **X** is the direction of the Table travel
  - **Y** is the direction of the Cross travel

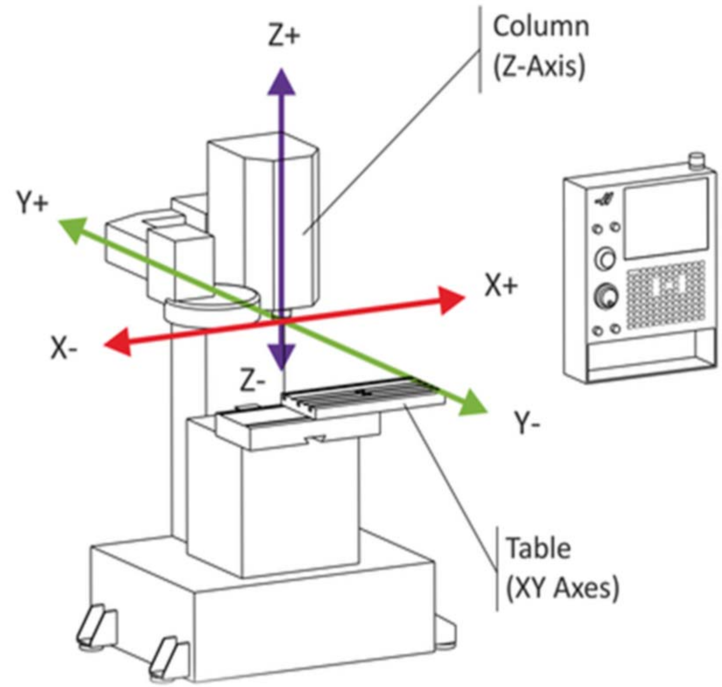


Figure 2-19: Directions of movement on a machine

(http://www.hsmworks.com/)

(W. S. Seames Computer Numerical Control: Concepts and Programming)

The Cartesian Coordinate System in machines

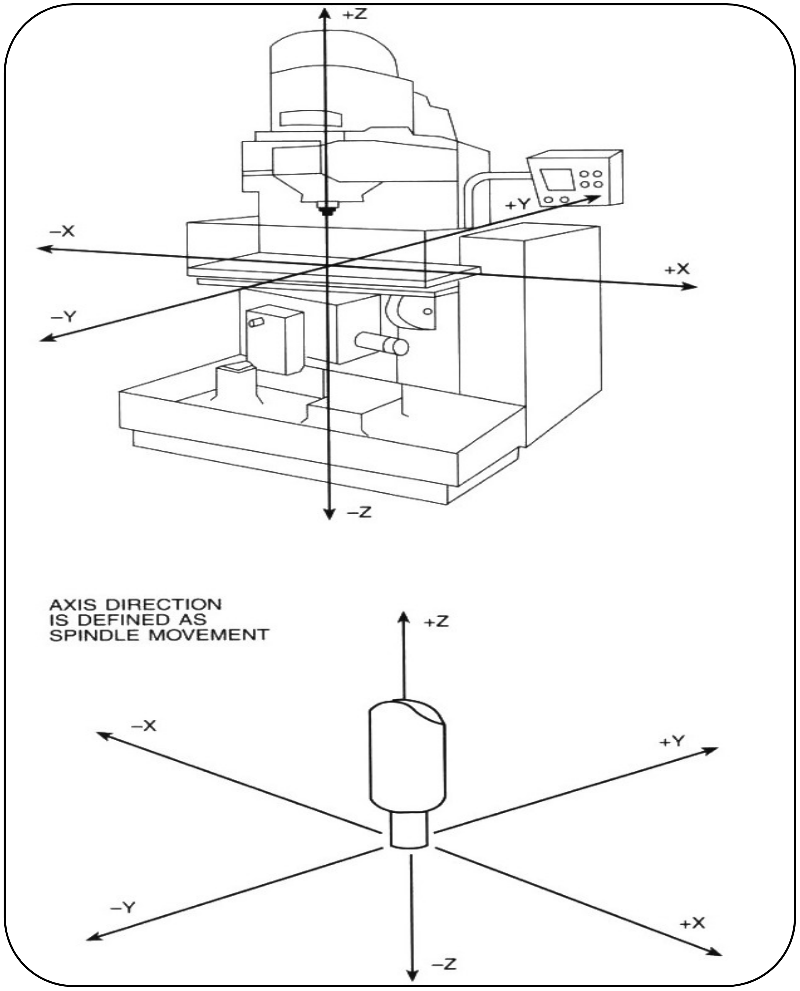


Figure 2-20: Three – Axis vertical mill

Three – Axis Milling machine:

- In a Three – Axis Vertical Milling Machine:
  - **X** is the direction of the Table travel
  - **Y** is the direction of the Cross travel
  - **Z** the Spindle travel up – down

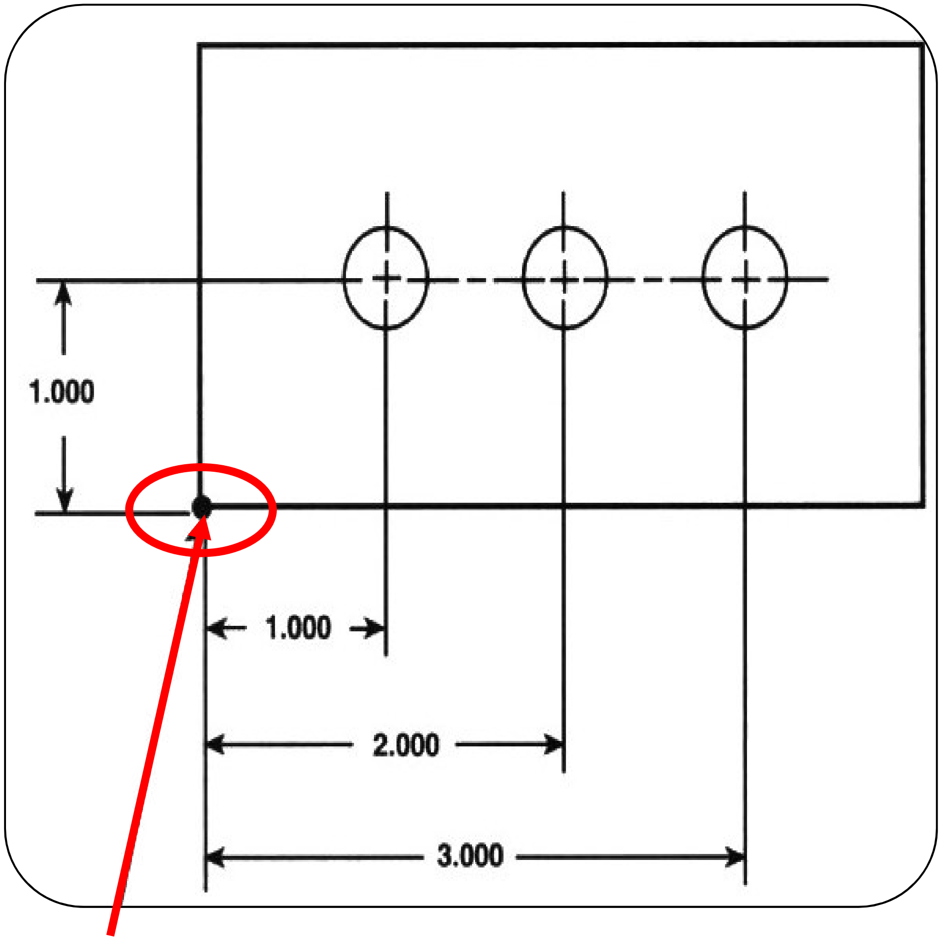
Positive and Negative Movement

- Machine axis direction is defined in terms of **spindle movement**
- On some axes the machine slides actually move; on other axes the spindle travels
- **For standardization the positive and negative direction for each axis is always defined as if the spindle did the travelling**
- The arrows saw the positive and negative direction of spindle movement along axes

Example

- To make a move in the +X direction (spindle right) the table would move to the left
- To make a move in the +Y direction (spindle toward the column) the saddle would move away the column
- The Z-axis movement is always positive (+Z) when the spindle moves towards the machine head and negative (–Z) when it moves toward the workpiece

Positioning Systems



ZERO REFERENCE POINT
FOR A MOVE TO ANY LOCATION

Figure 2-24: Absolute positioning

Absolute Positioning:

- All machine locations are taken from **one fixed zero point**
- All positions on the part are taken from the (X0, Y0) point at the **lower left corner** of the part
- **The 1st hole** will have coordinates of **(X1.000, Y1.000)**
- **The 2nd hole** will have coordinates of **(X2.000, Y1.000)**
- **The 3rd hole** will have coordinates of **(X3.000, Y1.000)**
- Every time the machine moves the controller references **the lower left corner of the part**

(W. S. Seames Computer Numerical Control: Concepts and Programming)

Positioning Systems

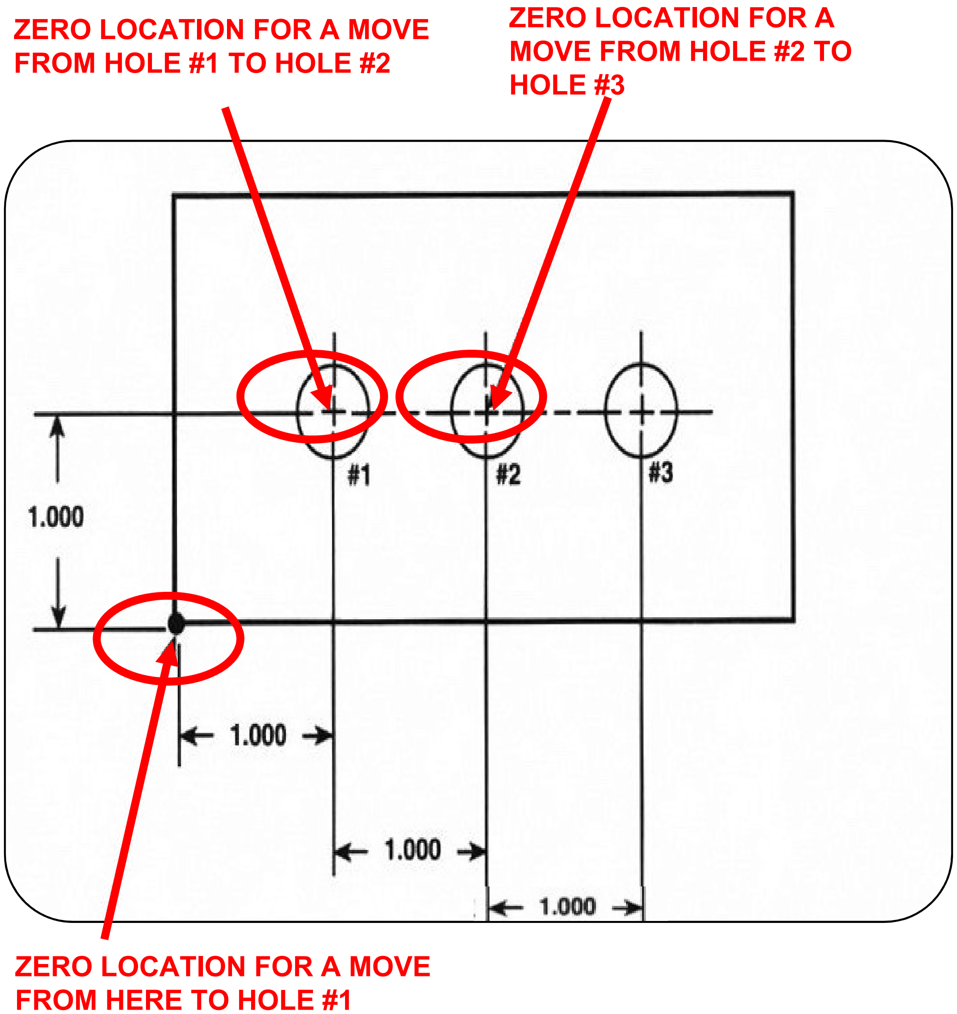


Figure 2-25: Incremental positioning

Incremental Positioning:

- The (X0, Y0) point moves with the machine spindle
- Each position is specified in relation to the previous one
- **The 1st hole** coordinates are: **(X1.000, Y1.000)**
- **The 2nd hole** coordinates are **(X1.000, Y0)**
- **The 3rd hole** coordinates are **(X1.000, Y0)**
- After each machine move **the current location is reset to (X0, Y0) for the next move**
- The **coordinate system moves with the location** and the machine controller **does not reference** any common zero point

Positioning Systems

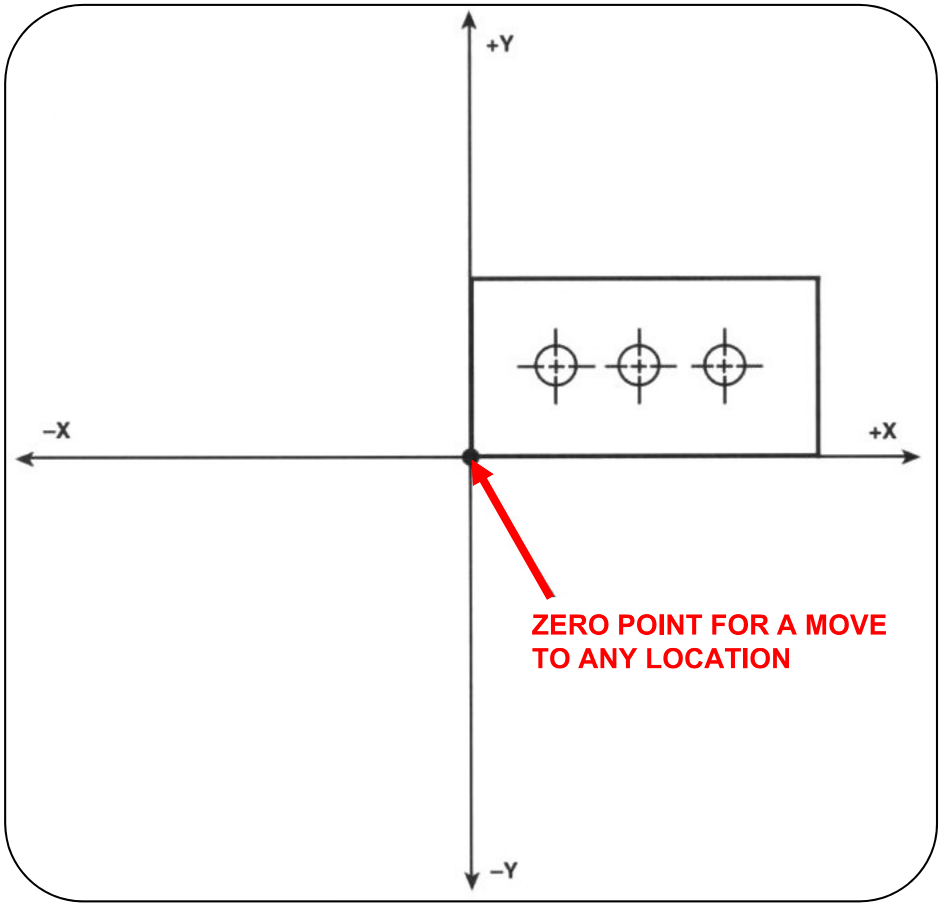


Figure 2-26(a):Relationship of the Cartesian Coordinate system to the part when using absolute positioning

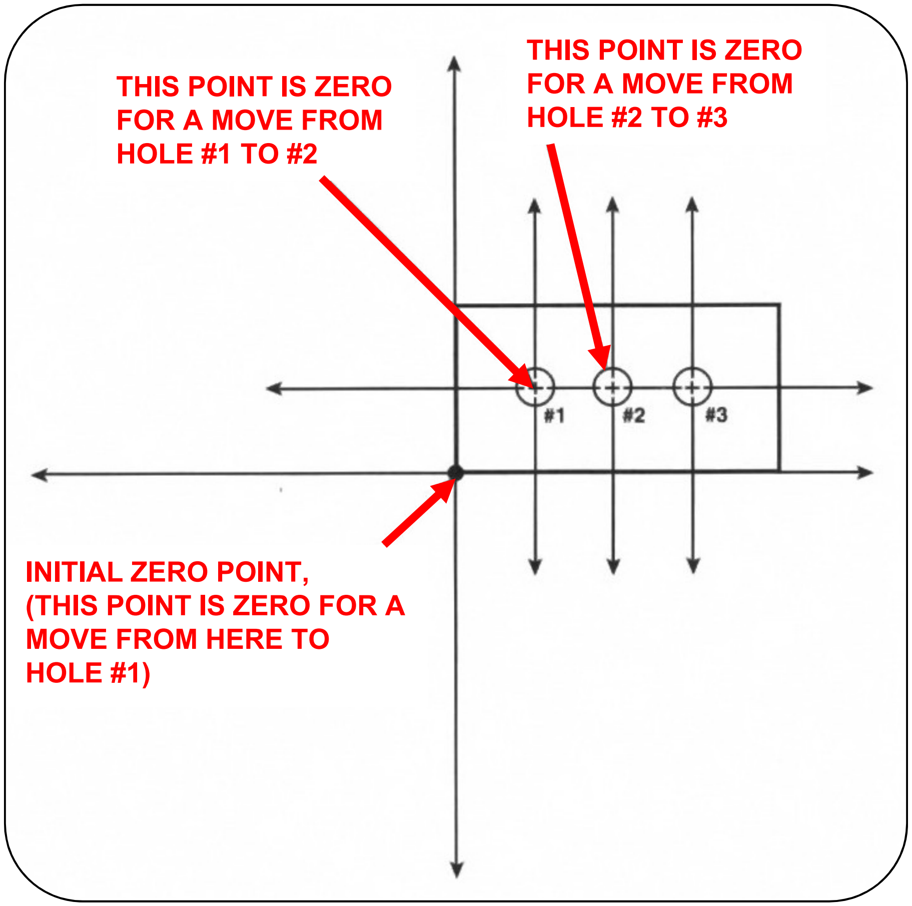
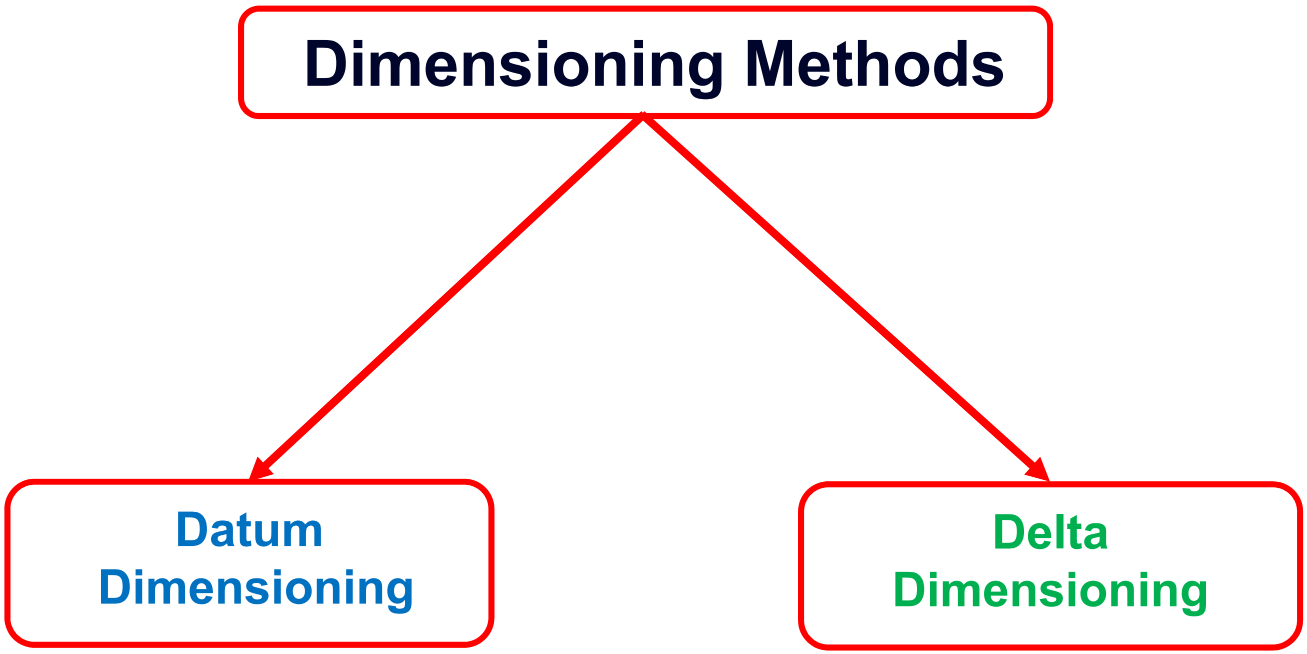


Figure 2-26(b):Relationship of the Cartesian coordinate system to the part when using incremental positioning

Dimensioning Methods

- In conjunction with NC machinery there are **two types of dimensioning practices used on blueprints :**



- These two dimensioning methods are **related to absolute and incremental positioning**

Dimensioning Methods

Datum Dimensioning

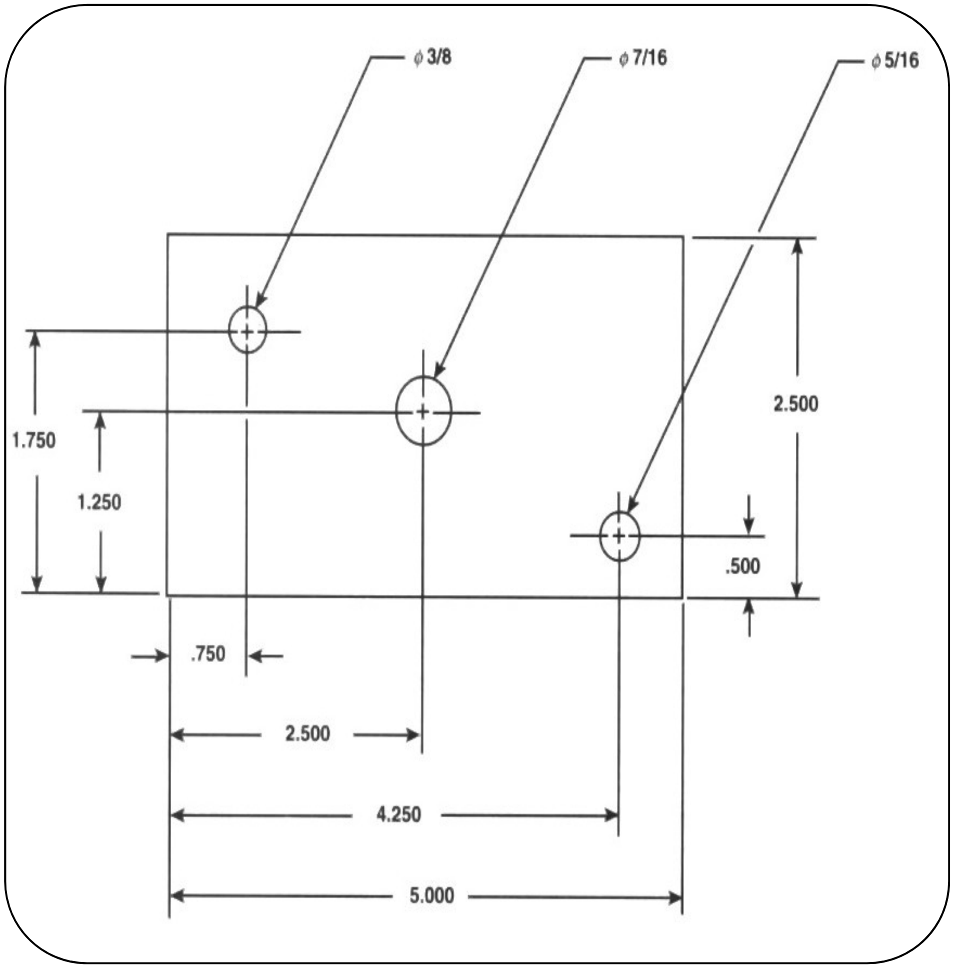


Figure 2-28: A datum dimensioned drawing

- **All dimensions** on a drawing are placed **in reference to one fixed zero point**
- Is ideally suited to **absolute positioning** equipment
- **All dimensions** are taken from the **corner of the part**

Dimensioning Methods

Delta Dimensioning

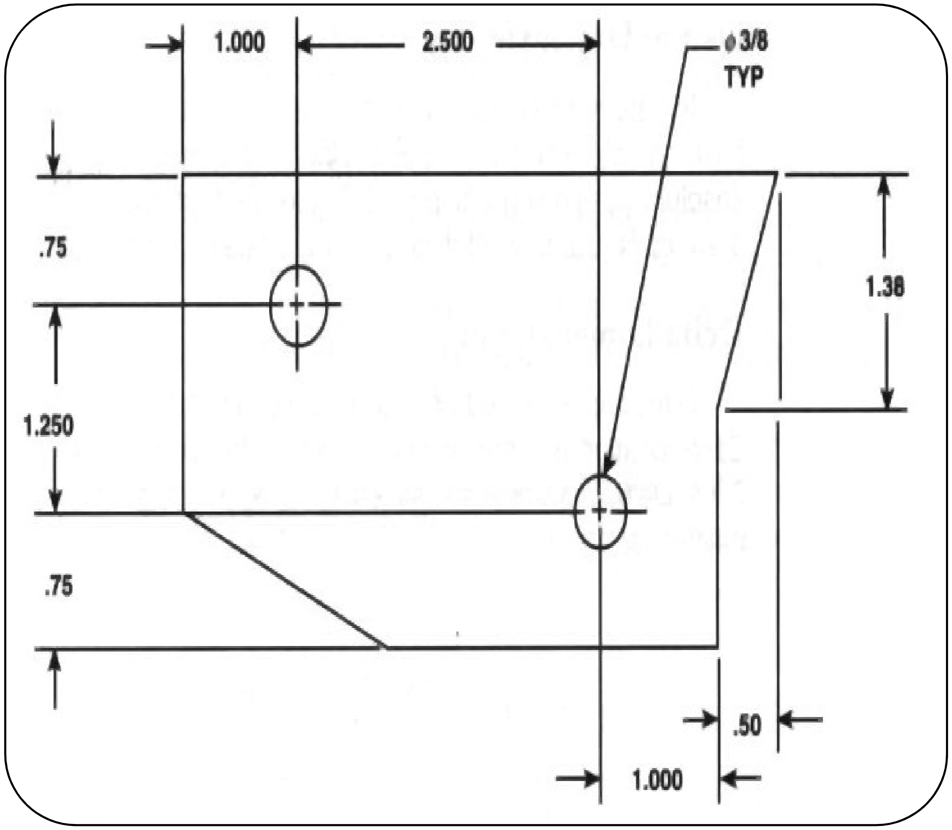


Figure 2-29: A delta dimensioned drawing

- **Dimensions** placed on a Delta Dimensioned drawing are **"chain-linked"**
- Each location is **dimensioned from the previous one**
- **Delta drawings** are suited for programming **incremental positioning** machines
- It is not uncommon to find the **two methods mixed** on one drawing

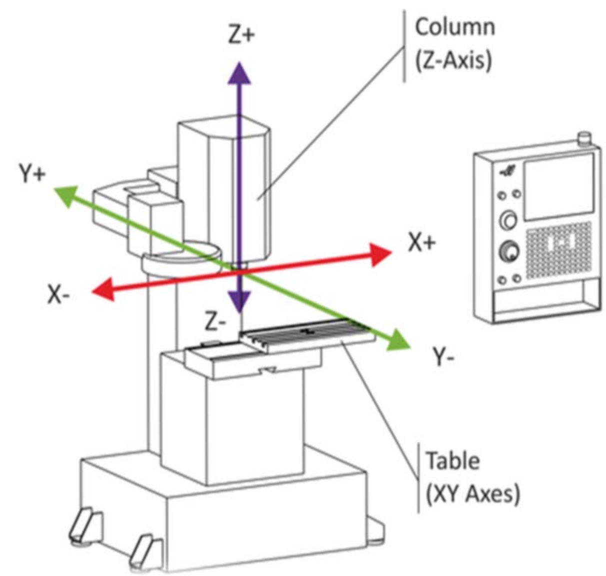
Setting the Machine Origin

Machine Coordinate System

- Most CNC machinery have a default coordinate system assumed during power-up the **Machine Coordinate System**

- The origin of this system is called the **Machine Origin** or **Home Zero Location**

- **Home Zero** is usually located at the **Tool Change** position of a Machining Center

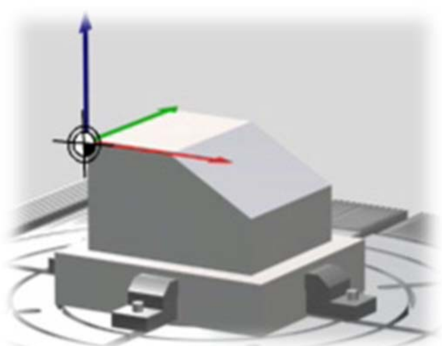


(http://www.hsmworks.com/)

Setting the Machine Origin

Programmer Coordinate System

- A part is programmed **independently** of the **Machine Coordinate System**
- The **programmer can pick a location** on the part or fixture becoming the origin of the coordinate system for that part
- The programmer's coordinate system is called the **Local** or **Part Coordinate System**
- The **Machine** and **Part Coordinate System** will almost **never coincide**
- **Prior running the part program** the coordinate system must be transferred from the machine system to part system, known as setting **ZERO POINT**



(http://content.heidenhain.de)

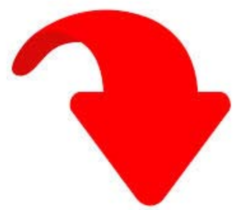
Process Planning and Tool Selection

Process Planning

Process planning can be defined as the **function**, which establishes the **sequence of the manufacturing processes** to be used in order to **convert a part from an initial to a final form**, where the process sequence incorporates **process description**, the **parameters** for the process and possibly **equipment** and/or machine tool selection

(Chryssolouris G., «Manufacturing Systems: Theory and Practice», 2nd Edition, 2006)

Decisions which must be made by the NC programmer to **successfully program a part**:



- **Machine Selection**: Which NC machine should be used?
- **Fixturing**: How will the part be held in the machine?
- **Strategy**: What machining operations & strategy will be used?
- **Tool Selection**: What cutting tools will be used?

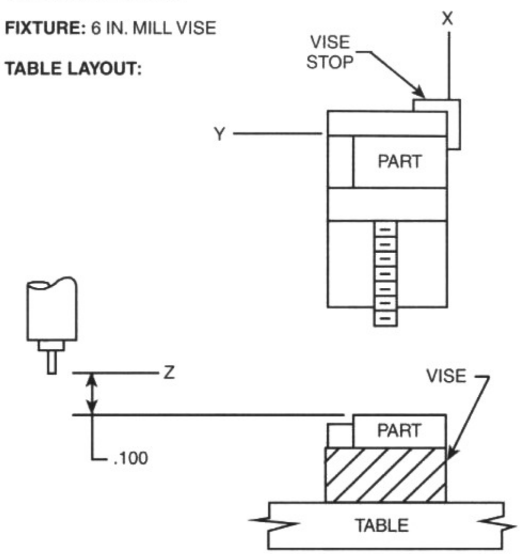
NC Setup Sheet

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

TAPE NUMBER: 1053

FIXTURE: 6 IN. MILL VISE

TABLE LAYOUT:



NOTES: DRILL POINT ANGLES TO BE 118° INCL.

DRWN: WSS	**NC SETUP SHEET FOR:**
PROG: WSS	**MACHINE:** VERTICAL MACHINING CENTER
DATE: 1-10-89	
B/P REV: C	

- The programmer must communicate to the **setup personnel** in the shop what **tools and fixtures** are to be used in the NC program
- The information is placed on **Setup Sheets**
- The **Setup Sheet** should contain **all necessary information to prepare for the job**

Figure 3-2: NC Setup Sheet for a CNC machining center

NC Setup Sheet

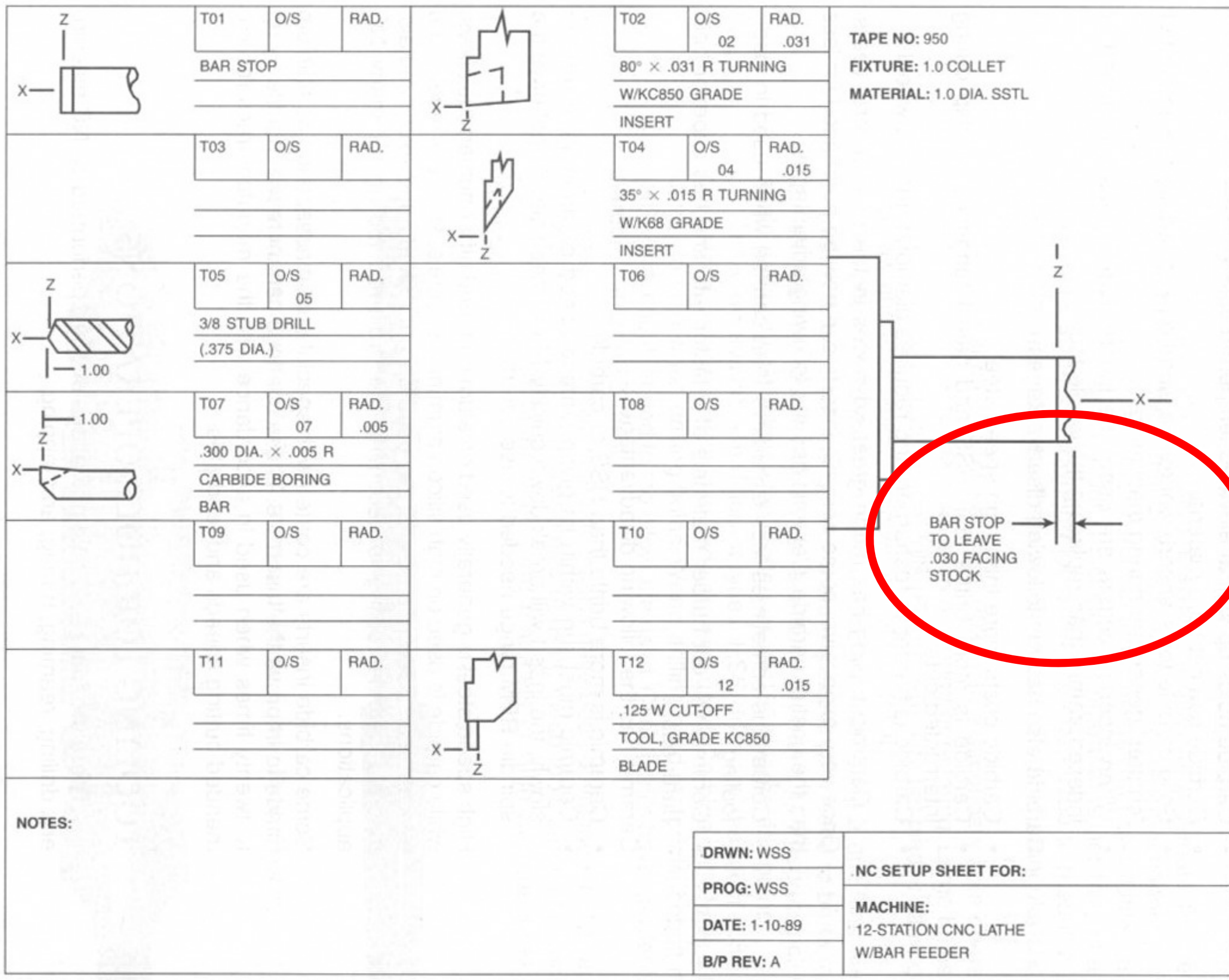


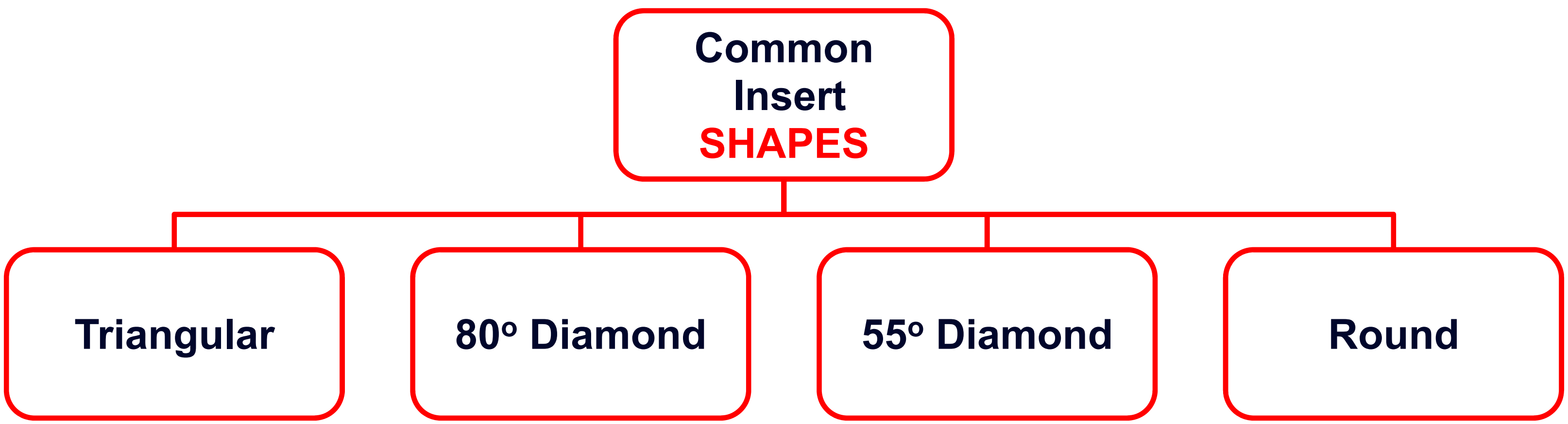
Figure 3-3 NC Setup Sheet for a CNC lathe

- **Special instructions** to the setup personnel or machine operators should be included
- **Special notes** regarding tooling should also be included

Inserts

Carbide Inserts and their Selection

- ***Carbide Inserts*** **are manufactured in a variety of *TYPES* and *GRADES***
- **The *TYPE* of the insert describes the *SHAPE* of the insert**



Speed and Feeds

The efficiency and the life of a cutting tool depend on the cutting feed and the feedrate at which it is run

Cutting Speed

- The ***cutting speed*** is the ***edge*** or ***circumferential*** speed of a tool
- In a machining center or ***milling*** machine the ***cutting speed*** refers to the edge speed of the rotating cutter
- In a turning center or ***lathe*** application the ***cutting speed*** refers to the edge speed of the rotating workpiece
- ***Cutting Speed (CS)*** is expressed in ***surface feet per minute (sfm)***
- ***CS*** is the number of feet a given point on a rotating part moves in one minute
- Proper ***CS*** varies from material to material – ***the softer the material the higher the cutting speed***

Speed and Feeds

Cutting Speed

- The spindle necessary ***rpm*** to achieve a ***given Cutting Speed*** can be calculated by the formula:

$$\mathbf{rpm} = \frac{\mathbf{CS\,x\,12}}{\mathbf{D} \times \boldsymbol{\pi}}$$

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

Cutting Speed

- The cutting speed of a ***particular tool*** can be determined from the rpm using the formula

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

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

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

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

Speed and Feed Example

- An aluminium workpiece is to be milled using a carbide inserted mill cutter
- The cutter is 1,750 inch diameter x 4 flute

What should be the appropriate Spindle rpm and Milling Feedrate?

- **Step 1:** Calculate Spindle Speed (rpm) with the following formula:

$$rpm = \frac{CS\ x12}{D\ x\ \pi}$$

- **Step 2:** Select CS = 1000 sfm (surface feet per minute) for Aluminum

$$rpm = \frac{1000 \times 3,82}{1,75} = 2183$$

(3.82 is derived from 12 divided (π))
The number 12 is used to convert the inch value of the part diameter into feet
Remember, we measure our parts in inches but use feet in cutting speed calculations.

Speed and Feed Example

- **Step 3:** Calculate Feedrate with the following formula:

$$F = R \times T \times rpm$$

- **Step 4:** Select R = 0.004 (chip load per tooth) – values are 0.002 to 0.006

$$F = 2183 \times 4 \times 0,004$$

$$F = 34,91\ inches/min$$

- **Step 5:** Calculate the chip thickness to insure that the inserts will not break down prematurely: It is assumed Width of the Cut = 1.000 inch wide

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

$$CT = \sqrt{\frac{1.000}{1,750}} \times 0.004$$

$$CT = 0,00302$$

- **Step 6:** CT is less than the recommended min of 0.004 and the feed per tooth must be calculated

Speed and Feed Example

- **Step 7:** Calculate Feed per tooth with the following formula and CT = 0,008

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

$$f = \sqrt{\frac{1,75}{1.000}} \times 0,008$$

$$f = 0,010$$

- **Step 8:** The new value for the chip load per tooth is substituted in the feedrate formula and recalculate Feedrate:

$$F = 2183 \times 4 \times 0.010$$

$$F = 87.32\, inches/min$$

Conclusion:

- The 2813 rpm spindle speed and 87.32 inches per min feedrate are "book value" rates
- They will have to be adjusted up or down depending on the machine, fixture and workpiece

Tool Changing and Tool Registers

Tool Changing and Tool Registers

Tool Changes:

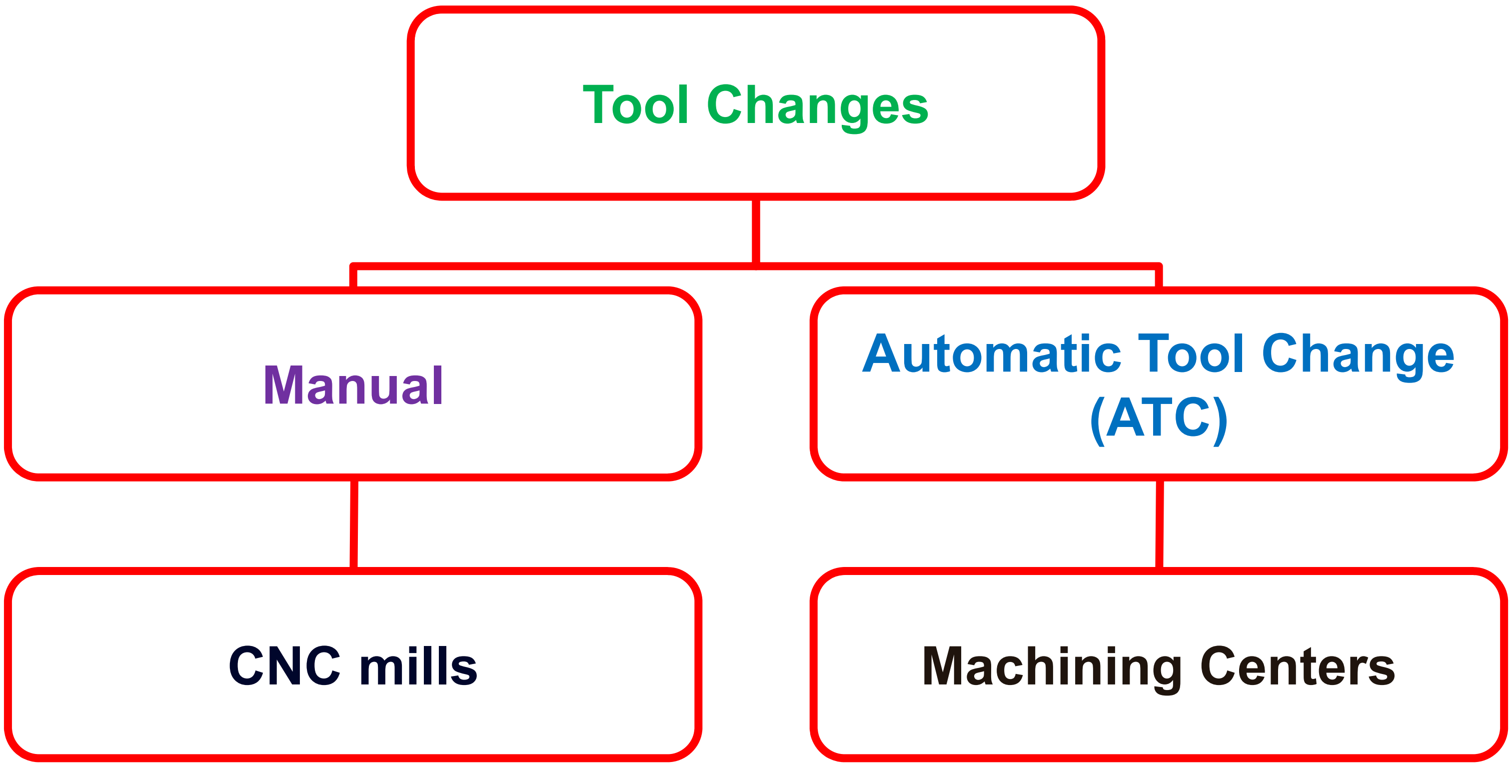
It is the tool changing capability that separates the CNC Machining Center from the CNC Milling machines

- **Machining Centers** like **milling machines** have the capability to do **numerous machining operations** (drilling, tapping, milling etc)
- This is opposed to a machine capable of a **single function only** such as an **NC drilling machine**

Tool Changing and Tool Registers

Tool Changes

There are two types of tool changes:



Tool Changing and Tool Registers

Tooling for Manual Tool Change:

What is to be gained by the speed with which a CNC machine can position itself for hole drilling if the tool changes are so lengthy as to cancel the time and accuracy gained by using NC?

Tool changing greatly influences the efficiency of NC so tool changes should take place as quickly and safety as possible

- The **tool must be accurate located in the spindle** to assure proper machining of the workpiece
- **The tool must be located as accurately** as possible in the same **same location**
- **The tool must be located in the same relationship to the workpiece** each time is inserted to the spindle

Note

This is known as the **repeatability of a tool** – the ability to locate or repeat its position in the spindle each time it is used

Seams W., "Computer Numerical Control, Concepts & Programming"

Tool Changing and Tool Registers

Tooling for Automatic Tool Change

- When automatic tool change is used the **requirements for speed and repeatability are even more critical**
- The **machine's tool changer can not** think for itself or correct misalignments or tool setup errors like a human being
- The tool changer will **carry out its tool-changing cycle and nothing else** since that is all it was programmed to do
- Tooling used with a tool changer therefore **MUST**:
  - ✓ Be easy **to center** in the spindle
  - ✓ Be easy for the tool changer **to grab**
  - ✓ Have some means of providing **safe disengagement** of the tool changer from the tool once it is secured in the spindle

Tool Length and Tool Length Offset

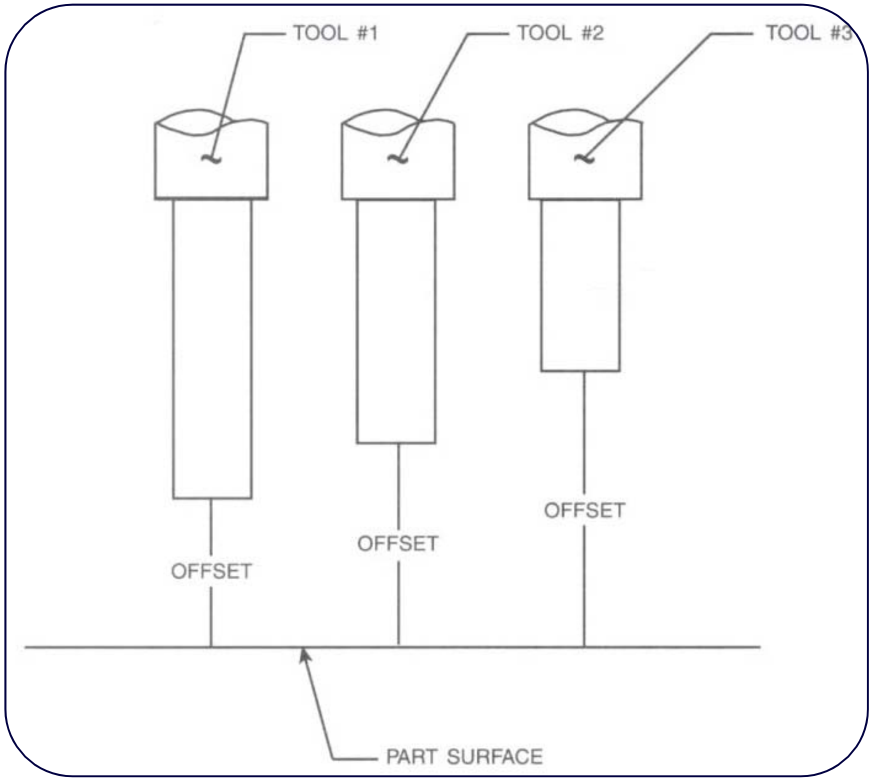


FIGURE 4-21: Tool length offset, difference of gage tool trim method

Seams W., "Computer Numerical Control, Concepts & Programming"

Tool Length Offset

- CNC machinery has revolutionized tool setting by the Programmable Tool Register

Tool Register:

- Is a ***memory spot*** in the computer where the length of the tool may be stored
- When a tool is called up the computer checks the Tool Register to see ***how much offset*** has been programmed for that tool
- Check the ***comments*** for tool offset
- The ***MCU sifts the Z-axis*** by the amount stored in the offset register

Tool Length and Tool Length Offset

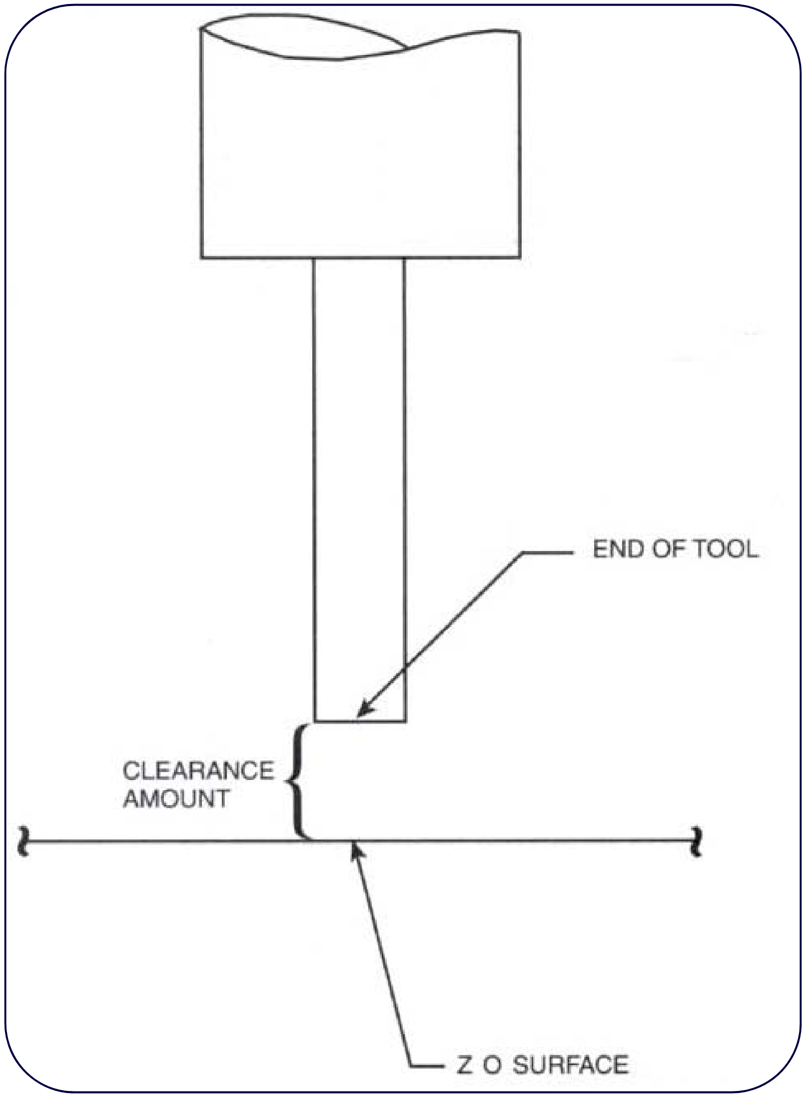


FIGURE 4-22: Tool clearance

Methods for Tool Trimming or Offsetting

- Difference of gage tool trim
- Plus direction trim
- Minus direction trim

Difference of Gage Tool Trim

- It is a variation of the Preset Tool method

Tool Length and Tool Length Offset

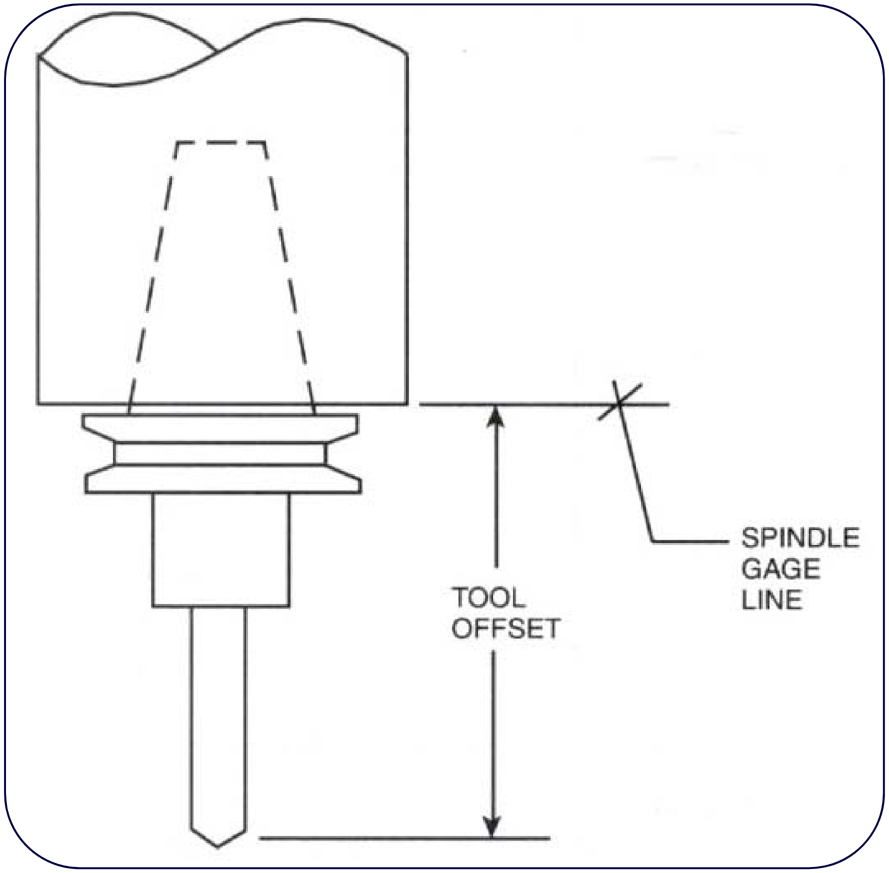


FIGURE 4-23 Tool length offset, plus direction trimming

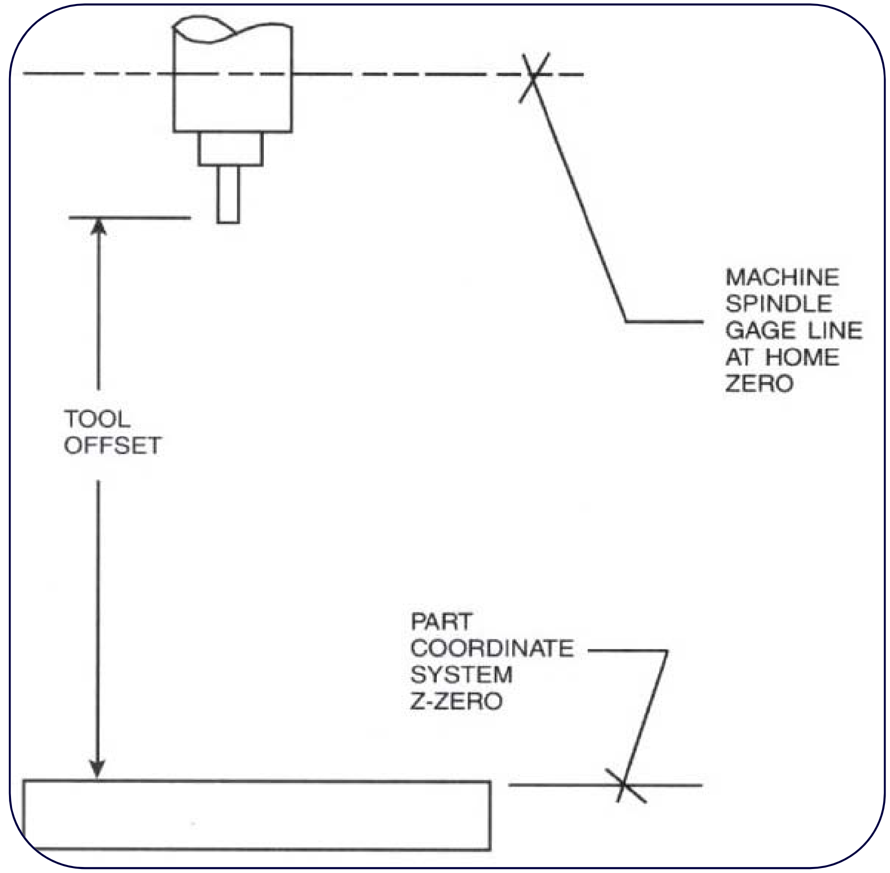


FIGURE 4-24 Tool length offset, minus direction trimming

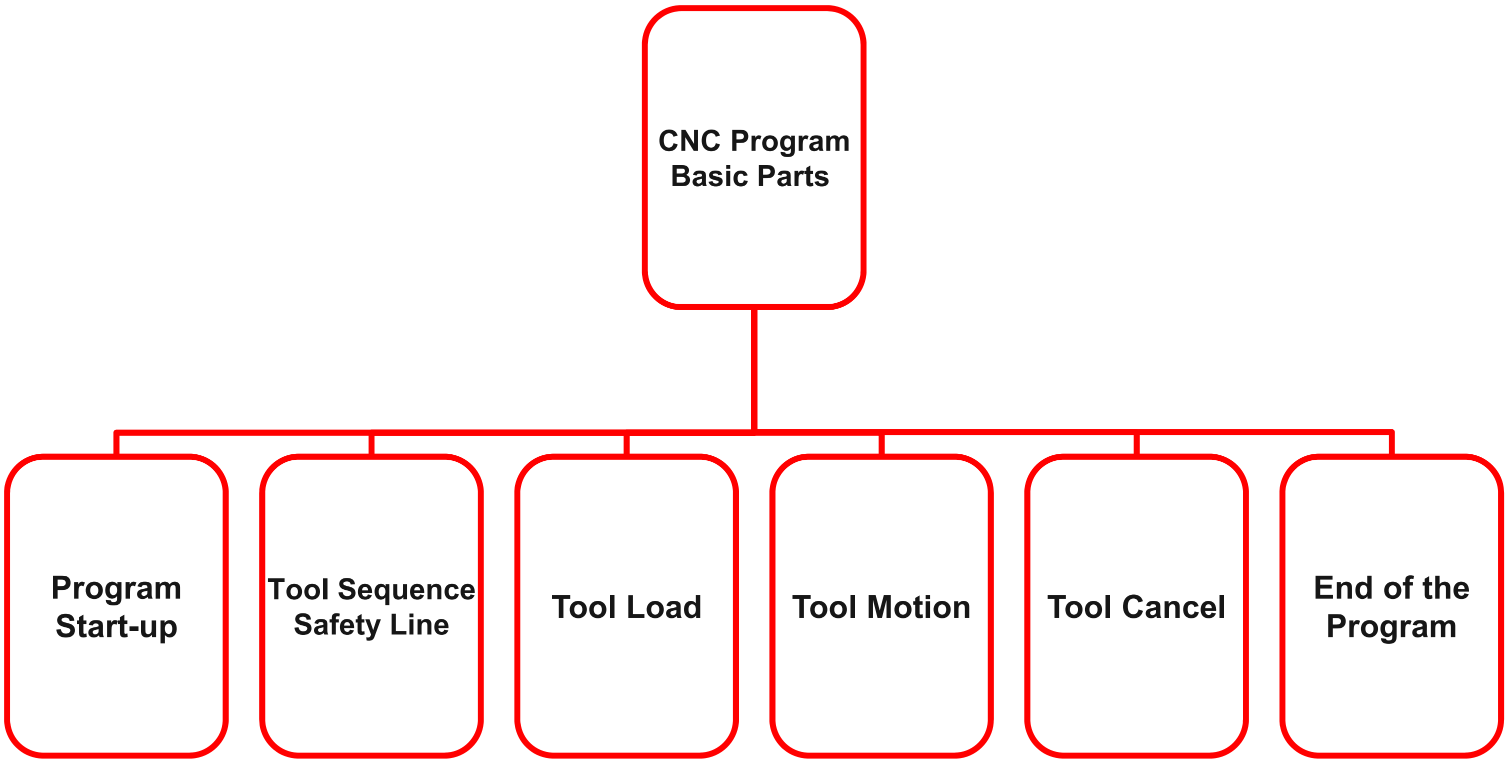
Seams W., "Computer Numerical Control, Concepts & Programming"

Two – Axis Programming

Parts of a CNC Program

Parts of CNC Program

- Regardless the **MCU** being programmed all CNC programs consist of the same basic parts



Word Address Format

- **Programming** is done in a **format** called **Word Address** which is the most common machine code format used today

Addresses

The block format for word address is as follows:

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

Only the information needed on a line need be given

Each of **the letters is called an address** (or word)

Word Address Format

N - The block sequence number

- An **N** number is used to **number the lines of NC code** for operator and/or programmer reference
- **N** numbers are **ignored by the controller** during program execution
- Most NC controls **allow a block to be searched** for by the sequence number for editing or viewing purposes

G - Initiates a preparatory function

- Preparatory functions **change the control mode** of the machine
- Examples of preparatory functions are rapid / feedrate mode, drill mode, tapping mode, boring mode, and circular interpolation
- Preparatory functions are called prep functions or more commonly G Codes

Word Address Format

X: Designates an X-axis coordinate.
X also is used to enter a time interval on FANUC and FANUC style controllers
Y: Designates a Y-axis coordinate
Z: Designates a Z-axis coordinate
I: Identifies the X-axis arc vector (the X-axis center point of an arc)
J: Identifies the Y-axis arc vector (the Y-axis center point of an arc)
K: Identifies the Z-axis arc vector (the Z-axis center point of an arc)
S: Sets the spindle rpm
H: Specifies the tool length compensation register
F: Assigns a feedrate
T: Specifies the standby tool (to be used in the next tool change)
M: Initiates miscellaneous functions (M functions)

- **M functions** control **auxiliary** functions such as :
  - the turning on and off of the spindle and coolant,
  - initiating tool changes, and
  - signaling the end of a program

Preparatory Functions (G Codes) Used in Milling

Following is a list of preparatory functions used in CNC milling examples in this text. Other codes commonly used on General Numeric controllers are also listed.

G00-Rapid traverse positioning.

G01-Linear interpolation (feed rate movement).

G02-Circular interpolation clockwise.

G03-Circular interpolation counterclockwise.

G04-Dwell.

G10-Toollength offset value.

G17-Specifies X/Y plane.

G18-Specifies X/Z plane.

G19-Specifies Y/Z plane.

G20-Inch data input (on some systems).

G21-Metric data input (on some systems).

G22-Safety zone programming.

G23-Cross through safety zone.

G27-Reference point return check.

G28-Return to reference point.

G29-Return from reference point.

G30-Return to second reference point.

G40-Cutter diameter compensation cancel.

G41-Cutter diameter compensation left.

G42-Cutter diameter compensation right.

G43-Toollength compensation positive direction.

G44-Toollength compensation negative direction.

G45-Tool offset increase.

G46-Tool offset decrease.

Preparatory Functions (G Codes) Used in Milling

G47-Tool offset double increase.

G48-Tool offset double decrease.

G49-Tool length compensation cancel.

G50-Scaling off.

G51-Scaling on.

G73-Peck drilling cycle.

G74-Counter tapping cycle.

G76-Fine boring cycle.

G80-Canned cycle cancel.

G81-Drilling cycle.

G82-Counter boring cycle.

G83-Peck drilling cycle.

G84-Tapping cycle.

G85-Boring cycle (feed return to reference level).

G86-Boring cycle (rapid return to reference level).

G87-Back boring cycle.

G88-Boring cycle (manual return).

G89-Boring cycle (dwell before feed return).

G90-Specifies absolute positioning.

G91-Specifies incremental positioning.

G92-Program absolute zero point.

G98-Return to initial level.

G99-Return to reference (R) level.

Miscellaneous (M) Functions Used in Milling And Turning

Following is a list of miscellaneous functions used in the milling and turning examples in this text. Other M functions common to General Numeric and FANUC controllers are also listed.

M00-Program stop.

M01-0ptional stop.

M02-End of program (rewind tape).

M03-Spindle start clockwise.

M04-Spindle start counterclockwise.

M05-Spindle stop.

M06-Tool change.

M08-Coolant on.

M09-Coolant off.

M13-Spindle on clockwise, coolant on (on some systems).

M14-Spindle on counterclockwise, coolant on.

M17-Spindle and coolant off (on some systems).

M19-Spindle orient and stop.

M21-Mirror image X axis.

M22-Mirror image Y axis.

M23-Mirror image off.

M30-End of program, memory reset.

M41-Low range.

M42-High range.

M48-0verride cancel off.

M49-0verride cancel on.

M98-Jump to subroutine.

M99-Return from subroutine.

Three - Axis Programming

Parts of a CNC Program

Three-axis Programming

- Three-axis programming is used for a program sequence in which **all three machine axes are used at the same time**

Two-and-half axis programming

- Use all three axes **BUT** Primarily position a location using X and Y axis
- Use **Z axis to perform a drilling or milling operation**
- Is the most common CNC milling programming
- **90% of the CNC** machining center programming
- It is the **practical limit** for manual programming
- Mathematical calculations for 3-axis are **very time consuming**

Parts of a CNC Program

- 3-axis, 4-axis and 5-axis programming are performed **using CAD / CAM systems**

- **Tool length offset** is used

- Operator enters the **tool lengths into the appropriate tool length offset registers** in the CNC controller

- Tool length compensation **adjust Z-axis zero point** to account for the differences in the lengths of the various cutting tools used in the program

A Programming Task Using Three Axes

- **Several new word address commands used :**
- **G28 - Return to reference point command**
- **G28** is used in conjunction with other commands to **cause the spindle to position at the machine's coordinate system origin**
- This point is referred to as **home zero** in most CNC shops
- **If coordinates are specified on the G28 line, the spindle will first move to the coordinates, then to home zero**
- In this manner the spindle may be moved to a **known safe position** before moving to **home zero**

Seams W., "Computer Numerical Control, Concepts & Programming"

A Programming Task Using Three Axes

- **G44 - Calls up a tool length offset register**
- A **G44** accomplishes a **Z-zero** shift toward the workpiece
- **H - Used to assign a tool register**
- **H01** would assign the information stored in **tool length register #1**
- **H02** would assign the information stored in **tool length register #2**
- **G49** - This is the **tool length offset cancel code**

A Programming Task Using Three Axes

- **Several new word address commands used in this program**
- **G81 - This is the canned drill cycle**
- When a **G81** is issued:
  - The spindle rapids to the (X,Y) coordinates specified on the drill cycle line
  - The Z axis then rapids to the specified feed engagement point
  - Feeds to the final drill depth
  - Then rapids out of the hole to either the rapid or initial level
- **G80 - This is the canned cycle cancel code**
- When a **G80** is issued, the active canned cycle code is turned off

A Programming Task Using Three Axes

- **R** - This address stands for the **canned cycle reference level**
- The **reference level** is the spot where the programmer desires the **canned cycle to start feeding into the workpiece**
- The **reference level** is also called the **rapid or gage level**
- **G92 - Absolute zero set command**
- This command tells the control to **reset the part coordinate system origin** - **Coordinates must be specified on the G92 block -** The coordinates tell the machine where **to set the origin, relative to the current spindle position**

A Programming Task Using Three Axes

- **G99/G98**
- **G98** is the **return to initial level** command
- **G99** is the **return to rapid (reference) level** command
- When a **canned cycle is active**, the spindle may be directed to **return to the rapid level** when it exits a hole with a **G99**
- If the programmer desires the **spindle to return to the original starting point Z height**, the **G98** command is issued
- **G99** results in the **faster cycle**
- **G98** is particularly useful for **jumping over clamps** and other obstructions while in a cycle

A Programming Task Using Three Axes

- **M01** - **Program optional stop** code

- **M01** functions as an **M00** with one exception: **it is only effective if the optional stop switch on the machine control is turned on**

- When this switch, -called an **opstop switch**-, is off, the **M01** is ignored by the control

- **M03** - is the code for **turning the spindle on in the clockwise direction**

- **M05** - Turns the **spindle off**

A Programming Task Using Three Axes

- **M06** - **Tool change** code
- When **M06** is issued, the machine's **automatic tool changer sequence will be initiated**
- **M08** - Turns the **flood coolant on**
- **M09** - Turns the **coolant off**
- **T** - **Selects the tool** to be put in the spindle by the tool changer
- **F** - **Assigns feedrates**, as in two-axis programming
- **S** - **Designates the spindle speed**

Modal / Non-Modal Commands

Modal Commands

- Codes that are **active for more than one line** in which they are issued
- **Rapid transverse, Feedrate moves and canned cycle** codes are examples of **modal commands**

Non-Modal command

- Is the one that is **active only in the program block in which it is issued**
- **M00**: **Program Stop** is an example of a **Non-Modal command**

Canned Cycles

- Are **routines** (e.g. **G81**) built into the control to **perform standard operations**
- Drilling, boring and taping are common operations
- The programmer can **call a canned cycle instead of repetitive programming**

Modal Commands

- Most G codes put the machine in a "permanent" status, which remains in effect until it is changed or canceled by another G command
- Those are the **modal commands**

G00	Rapid Transverse	G43	Tool length compensation (plus)
G01	Linear Interpolation	G44	Tool length compensation (minus)
G02	Circular Interpolation, CW	G49	Tool length compensation cancel
G03	Circular Interpolation, CCW	G80	Cancel canned cycles
G17	XY Plane	G81	Drilling cycle
G18	XZ Plane	G82	Counter boring cycle
G19	YZ Plane	G83	Deep hole drilling cycle
G20/G70	Inch units	G90	Absolute positioning
G21/G71	Metric Units	G91	Incremental positioning
G40	Cutter compensation cancel		
G41	Cutter compensation left		
G42	Cutter compensation right		
G43	Tool length compensation (plus)		

Figure 4: Example showing G00 and G01 modal commands

Definitions and Codes

Cutter Diameter Compensation

Programs presented in previous chapters required an **allowance** for the cutter radius in the programmed coordinates

- Most CNC machines have a built-in feature called **cutter diameter compensation (cutter comp)** that allows the part line to be programmed.
- (Confusion may be caused by use of the terms "**offset**" and "**compensation**'') In this text, **"compensation"** refers to cutter diameter offset
- The term "**offset**" refers to **tool length offset** and the change in axis coordinates when programming arcs and angles.)
- **Cutter comp** is also called **Cutter Radius Offset (CRO)** by some controller manufacturers

Seams W., "Computer Numerical Control, Concepts & Programming"

Definitions and Codes

Cutter comp is accomplished through the use of G codes : G40, G41, G42

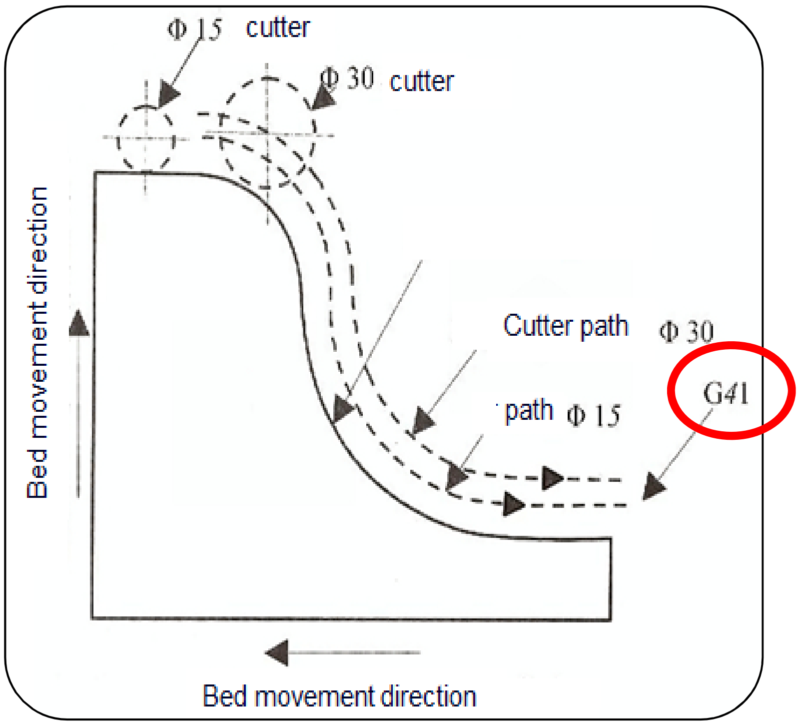
G40 – **Cutter diameter compensation cancel**. Upon receiving a **G40**, cutter diameter compensation **is turned off**. The tool will change from a compensated position to an uncompensated position on the next X, Y, or Z axis move

G41 – **Cutter diameter compensation left**. Upon receiving a **G41**, the tool will **compensate to the left** of the programmed surface. The tool will move to a compensated position on the next X, Y, or Z axis move after the **G41** is received

G42 – **Cutter diameter compensation right**. **Compensates to the right** of the programmed surface

Cutter Compensation

- **NOTE that there might be changes in cutter's diameter** due to:
  - ➢ Deterioration
  - ➢ Change cutter
  - ➢ Rounding of the edge radius of the cutting tool



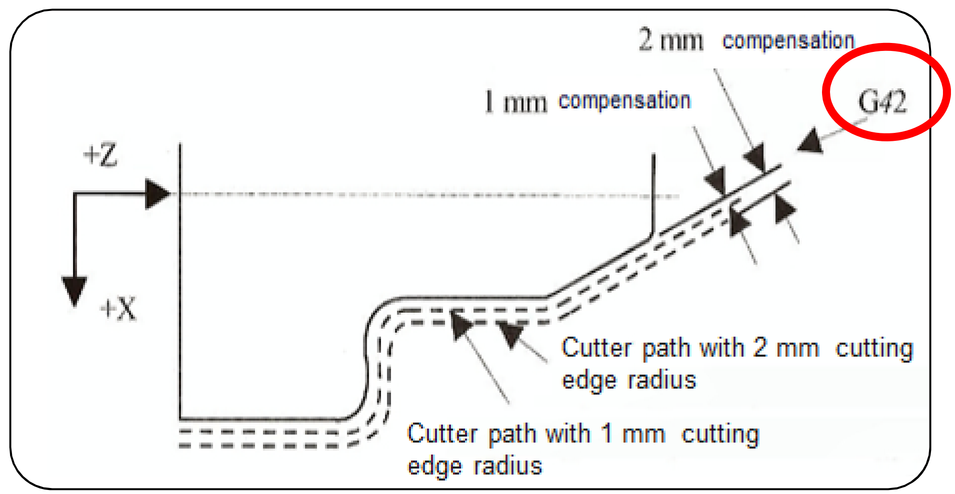


Figure 2:Cutter compensation G41,G42,
(source: Σύγχρονες μέθοδοι κατεργασίας υλικών και προγραμματισμός με Ηλεκτρονικό Υπολογιστή (Η/Υ) ,Δ. Μούρτζης ,κ.α.)

Codes G41, G42

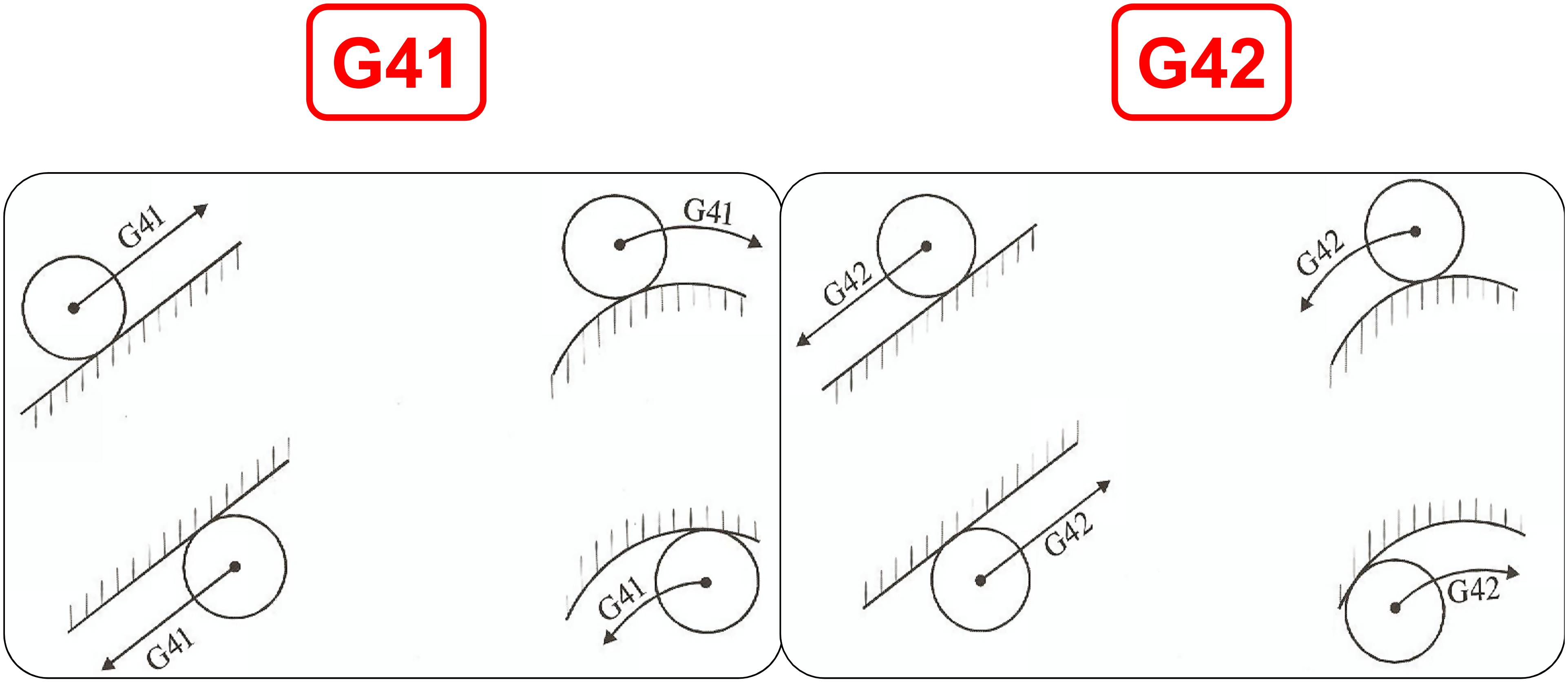


Figure 3:Example of G41,G42 codes

(source: Σύγχρονες μέθοδοι κατεργασίας υλικών και προγραμματισμός με Ηλεκτρονικό Υπολογιστή (Η/Υ) ,Δ. Μούρτζης ,κ.α.)

Codes G41, G42

G41,G42

- Command format

N.. G01 G41 X.. Y.. D..
N.. G02 G41 X.. Y.. I.. J.. D..

Where:

- **D** is the **memory address of machine's MCU** where the compensation value is registered

G40

- **Compensation cancel** (**G41** and **G42**) of cutter radius
- Activated **automatically** by machine at the beginning of each program
- «**Modal**» command

(source: Σύγχρονες μέθοδοι κατεργασίας υλικών και προγραμματισμός με Ηλεκτρονικό Υπολογιστή (Η/Υ) ,Δ. Μούρτζης ,κ.α.)

Linear Interpolation

Linear interpolation:

- Means **cutting a straight line between two points**

- Sometimes this is referred to as a **feedrate move** since modern CNC controls **automatically** perform **linear interpolation** on any move made while in feed-rate mode

- Prior to modern CNC controls special codes were necessary to turn on the built-in linear interpolation system

- Some CNC controls also will **interpolate rapid moves** - while others simply move the axes drive motors at maximum speed in rapid traverse mode

Linear Interpolation

- The axis the spindle moves with **basic orthogonal movements** from the beginning to the end of the path

- The programmed rectilinear path is **divided into a large number of short length straight lines**

- **The more lines, the better approximation** is made of the actual path

- **The more lines, the more computational power required** - No longer used for non straight segments

Linear Interpolation

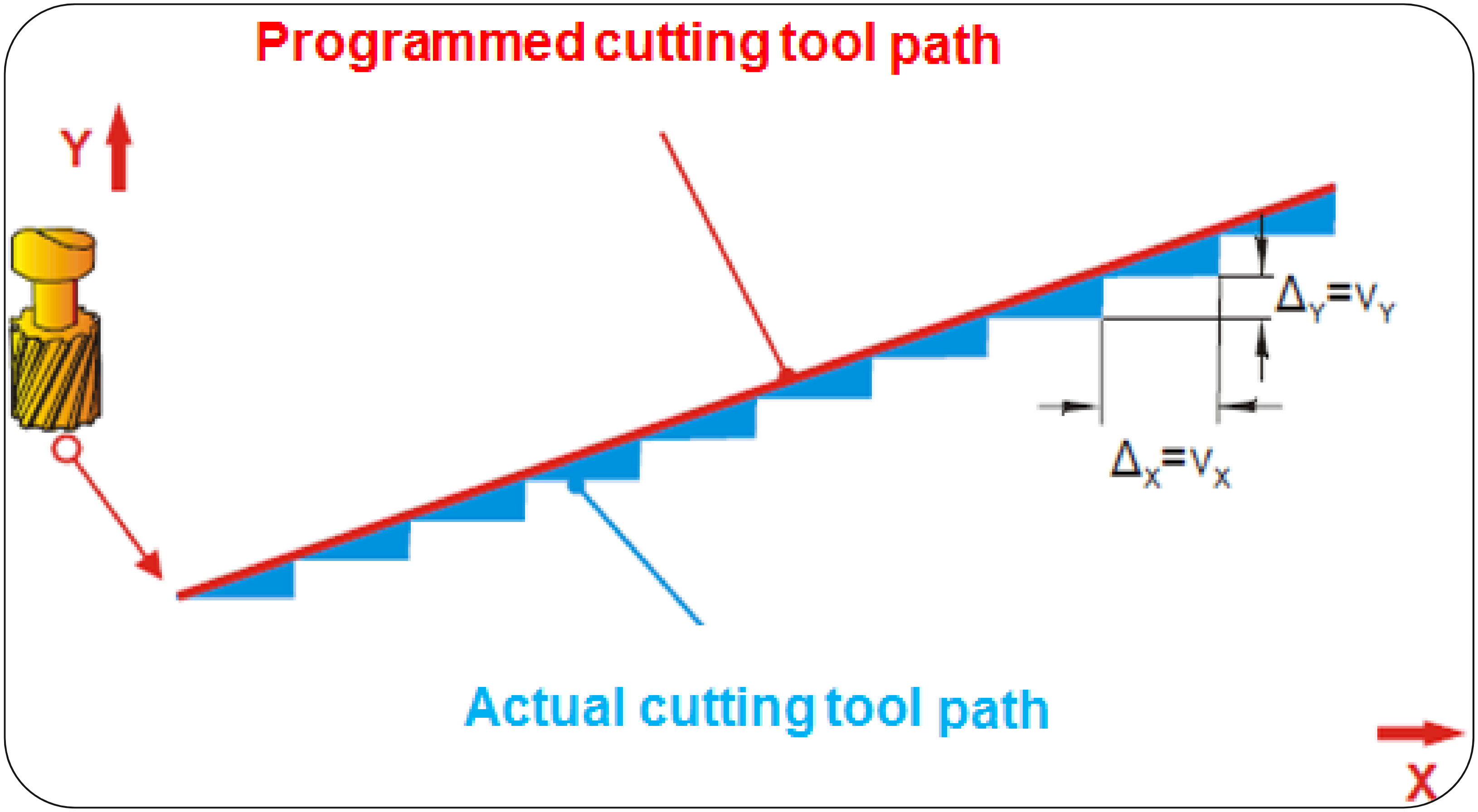


Figure 1:Linear Interpolation
(Source:Σύγχρονες μέθοδοι κατεργασίας υλικών και προγραμματισμός με Ηλεκτρονικό Υπολογιστή (Η/Υ) ,Δ. Μούρτζης κ.α)

Linear Interpolation

Linear interpolation:

- Machines capable of linear interpolation have a **continuous-path control system -** meaning that the drive motors on the various axes can operate at varying rates of speed

- Virtually all modern CNC controls utilize continuous path controls

- When cutting an angle the MCU calculates the angle based on the programmed coordinates

- Since the MCU knows the current spindle location, it can calculate the **difference in the X coordinate** between the current position and the programmed location

The change in the Y coordinate divided by the change in the X coordinate yields the slope of the cutter centerline path

Linear Interpolation

Calculating Cutter Offsets:

- The cutter has already been positioned at location #1
- A .500-inch diameter end mill is being used
- Before the angle can be cut it is necessary to first position the spindle at location #2
- The Y-axis coordinate for location #2 - as dimensioned on the part - is not the same point as the edge of the angle
- To determine this Y-axis cutter offset it will be necessary to determine the amount that must be added to the dimension on the part prior to place the spindle at location #2
- It will be necessary to calculate an amount to be subtracted from the point on the part designated as "P" to arrive at the X-axis coordinate for location #3

Linear Interpolation

Interpolating in Word Address Format (G01)

To illustrate linear interpolation, the following program lines would move the cutter from location #1 to locations #2, #3, and #4

```
N...G01 Y1.144        (move from #1 to #2)
N...X1.665 Y2.25      (move from #2 to #3)
N...X4.25             (move from #3 to #4)
```

- A **G01** is given to turn on linear interpolation (feedrate mode)
- The coordinate **Y1.144** moves the cutter to **location #2**
- The **X1.665**, **Y2.25** move the cutter from **location #2** to **#3**
- The **X4.25** coordinates then move the **cutter from #3** to **#4**
- Note: **G01** is a modal code: The machine remains in **linear interpolation mode (feedrate mode)** for all the coordinates specified. The **G01** is active until cancelled by another motion mode G code (**G00**, **G02**, **G03**, or **G04**)

Notes

Computer Aided Manufacturing (CAM) programming systems automatically calculate **cutter offsets** with **speed and accuracy no programmer can match**

- For this reason **CAM** systems have become the preferred programming system in many shops

- A good programmer or CNC operator must still know how to calculate cutter offsets in order **to edit programs in the machine control during the first piece setup**

Linear Interpolation

- Other cutter situations will present themselves in CNC part programming such as **arcs tangent to an angle** or **arcs tangent to other arcs**

- **CAM** programming systems **automatically calculate cutter offsets** with speed and accuracy no programmer can match

- For this reason **CAM** systems have become the preferred programming system in many shops

- A good programmer or CNC operator must still know ***how to calculate cutter offsets in order to edit programs*** in the machine control during the first piece setup

Circular Interpolation

- In **cutting arcs**, the **M**achine **C**ontrol **U**nit **(MCU)** uses its ability to **generate angles to approximate an arc**

- Since the machine axes do not revolve around a centerpoint in a typical three-axis arrangement, **the cutting of a true arc is not possible**

- **Circular interpolation** is the term used to describe generating a **move consisting of a series of straight-line chord segments** by the **MCU** in two axes to simulate circular motion, as illustrated in Figure 7

- These **chord segments** are very small and practically indistinguishable from a true arc

- Figure 8 shows a part with a radius to be machined

Circular Interpolation

- **Limited to the main plane** of the machined surface

- **Unable participation of the rotary machining axis** of the machine-tool

- **Not used for interpolation in the space** due to requirement of the combined movement of three or more machining axes

- **Ideal** for moving the axes when **the path of the cutting tool in a plane contains circles, half circles or arcs**. In this case only the coordinates of the ends of the arc, the radius and center are required

Circular Interpolation

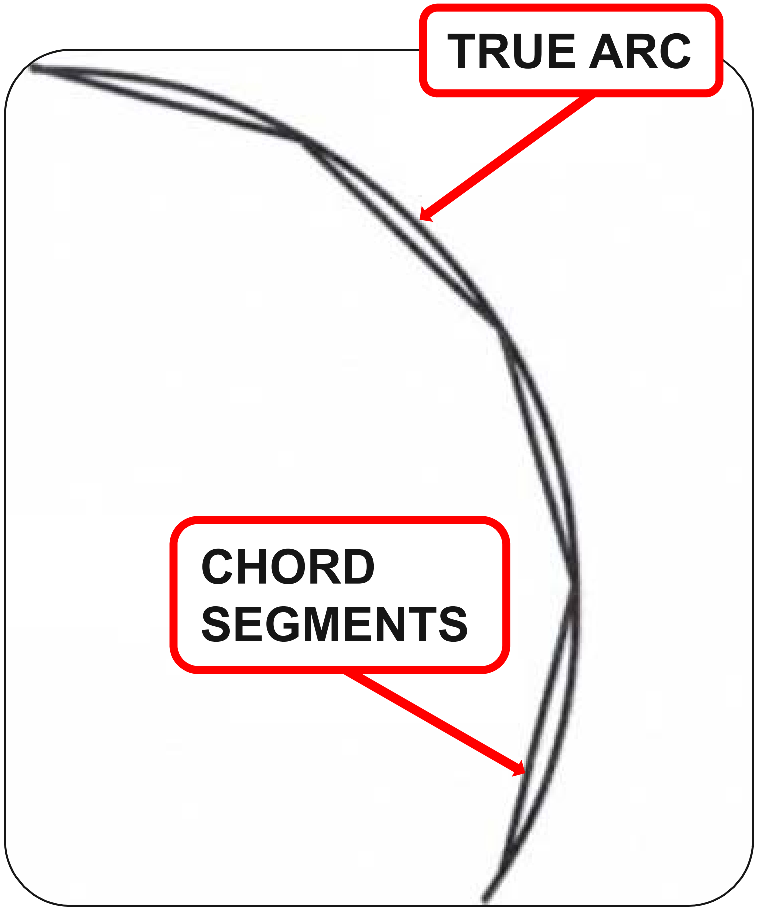


Figure 7: Circular interpolation chord segments

Seams W., "Computer Numerical Control, Concepts & Programming"

- A **move consisting of a series of straight-line chord segments** by the MCU in two axes to simulate circular motion

Circular Interpolation

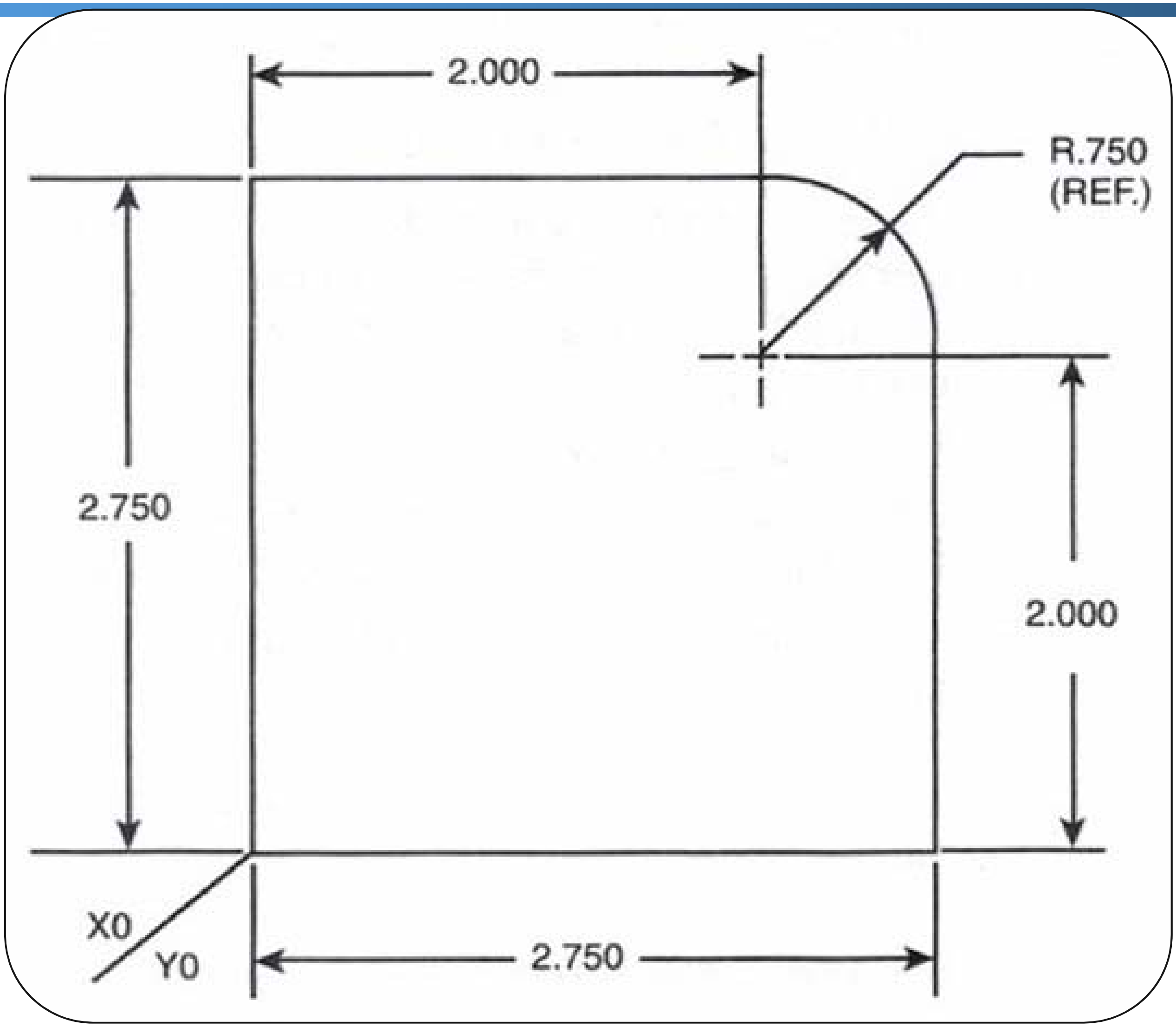
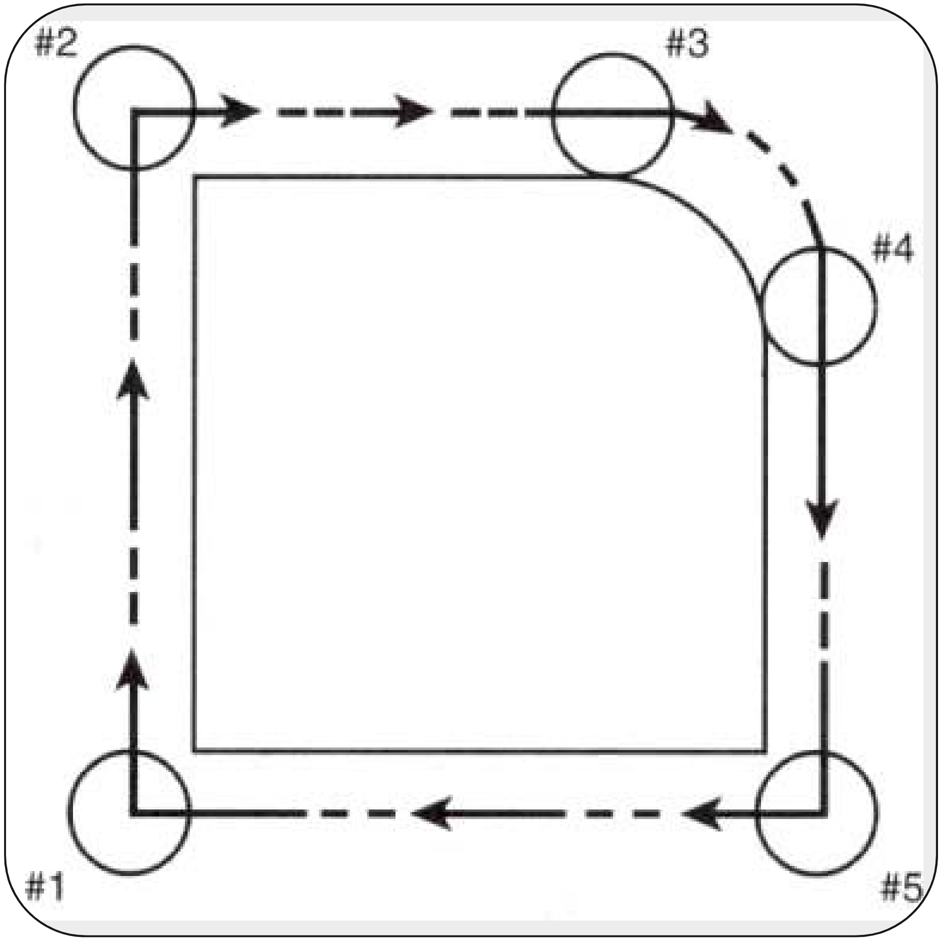


Figure 8: Part with radius to be machined

Seams W., "Computer Numerical Control, Concepts & Programming"

Circular Interpolation



To generate an arc, the MCU needs to know the following information

1. The **axes** to be used in generating the arc
2. The **direction** of interpolation, **clockwise** or **counterclockwise**
3. The **starting X/Y/Z coordinate of the arc**
4. The **ending X/Y/Z coordinate of the arc**
5. The **X/Y/Z coordinates of the arc centerpoint**

Figure 9: Cutter path for part shown in Figure 8. In order to generate the radius, circular interpolation will be used to send the cutter from location #3 to location #4,a .500-inch diameter end mill will be used *(Seams W., "Computer Numerical Control, Concepts & Programming")*

Circular Interpolation

Specifying Axis for Interpolation

- Circular interpolation by definition involves only two axes.
- On FANUC-style controls, a plane designation code is used to select which pair of axis will be used to generate the arc motion.
- There are three G codes used to specify these planes:

G17 – Selects the X/Y plane (X and Y axis)

G18 – Selects the Y/Z plane (Y and Z axis)

G19 – Selects the Z/X plane (Z and X axis)

- These G codes are modal. A **G17**, for example, is cancelled only by a **G18** or **G19**
- The X/Y plane (using the X and Y axis) is the most common orientation for circular interpolation, therefore, **G17** will be used throughout the examples in this text

Circular Interpolation

Specifying Arc Direction

Circular interpolation can be accomplished in one of two directions: **clockwise, or counterclockwise**. There are two G codes used to specify direction:

G02 – Circular interpolation clockwise (CLW)

G03 – Circular interpolation counterclockwise (CCLW)

- **G02**/**G03** codes are modal
- They will cancel an active **G00** (rapid traverse) or **G01** (linear interpolation) codes
- **G02**/**G03** are feedrate mode codes, just as **G01** is.
- The difference lies in the type of interpolation used.
- **G01** generates straight-line interpolation motion. **G02**/**G03** generates arc simulation interpolation motion

Circular Interpolation

Specifying Beginning and Ending Arc Coordinates

- The **MCU** requires the spindle be positioned at the start of the arc when the **G02**/**G03** command is given
- The current spindle position is the beginning arc coordinates. The axis coordinates given on the **G02**/**G03** line are the spindle ending points of the arc motion

Specifying Arc Center points

- There are two methods used to specify arc centerpoints: **arc vector method** and **radius method** (*see* Figure 10)

- The arc vector method involves specifying the coordinates of the arc centerpoint as X/Y values

- In the radius method, the arc centerpoint is calculated internally by the **MCU**. The programmer simply specifies the radius value required

Circular Interpolation

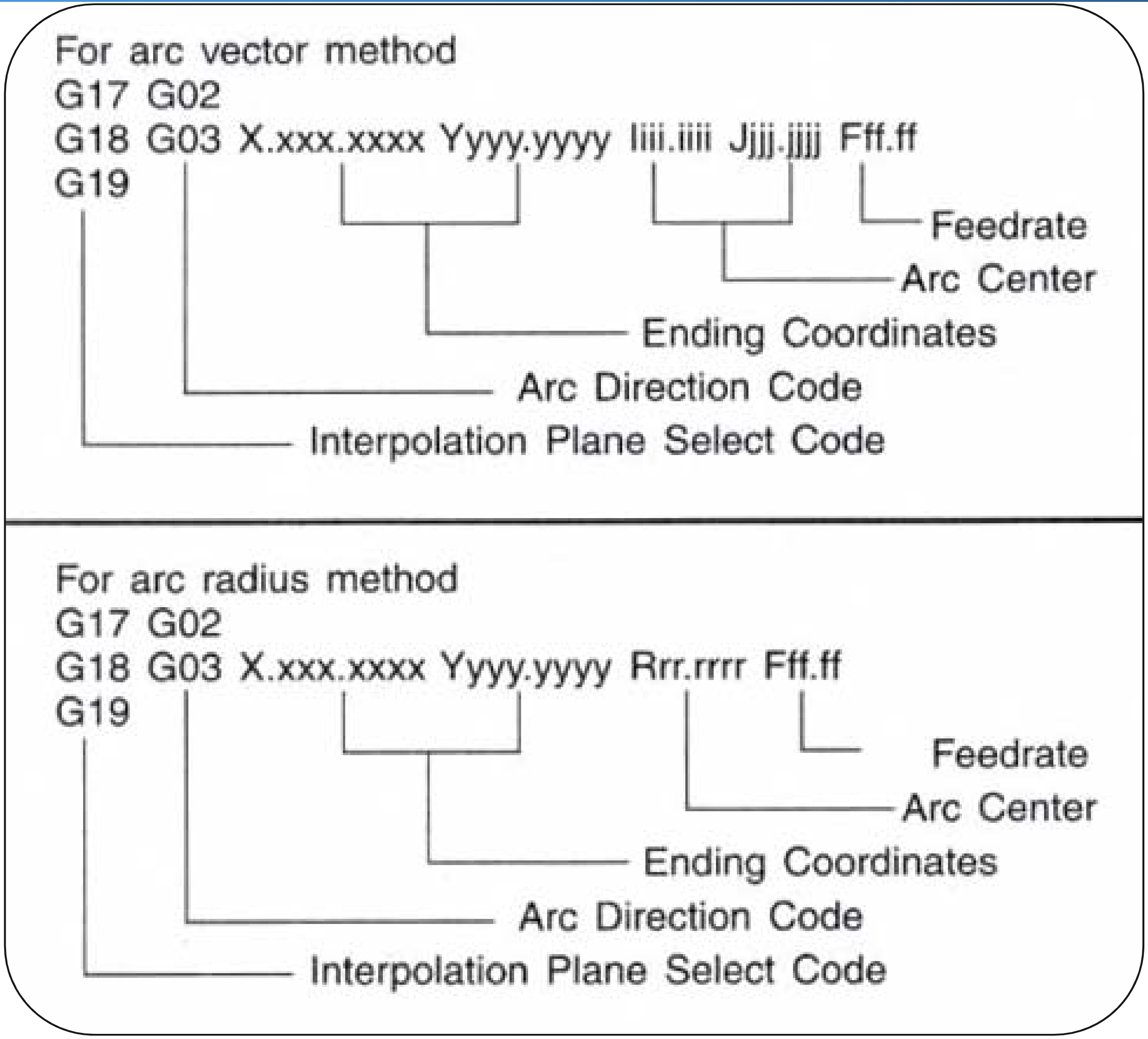


Figure 10:Arc vector method and radius method *(Seams W., "Computer Numerical Control, Concepts & Programming")*

Circular Interpolation

Arc Vector Method

- Since ***X, Y, and Z addresses are used to specify the end point of an arc***, **secondary addresses** are required to specify the **centerpoint of an arc**. The following addresses are used to designate arc center points

- **I—X-axis coordinate of an arc**. **J—Y-axis coordinate of an arc**. **K—Z-axis coordinate of an arc**

- Since **circular interpolation** occurs only in two axes, only two of these three codes will be required to generate an arc. When using the X/Y plane for milling arcs, as this text does, the **I** and **J** addresses are used

- The different ways controllers required the arc centerpoints to be specified complicate this matter: ***absolute coordinates***, ***to circle center***, or ***from circle center***. FANUC-style controls usually utilize the *to circle center* method

Circular Interpolation

Absolute Coordinates

- Some controls require the arc centerpoints specified by **I**, **J**, and/or **K** be the position of the **arc center relative to the coordinate system origin**

- In other words, the center of the arc is specified just as if it were a cutter coordinate using absolute positioning

- In Figure 8, the arc centerpoints are at X2.000, Y2.000. They would be specified as 12.0000 J2.0000 as in the following circular interpolation block:

N120 G17 G02 X3.Y2. 12. J2. F7

Circular Interpolation

To Circle Center

- Some controls require the **arc center points** be specified as an **incremental coordinate**, looking **from the center of the cutter to the center of the circle**
- In Figure 8, the radius of the arc is .750. The radius of the .500-diameter end mill is .25 inch
- To specify the centerpoint of the arc when the cutter is positioned at location #3, Figure 9 the incremental value of 0.0000 inch in X and -1.000 inch in Y would be specified as 10.0000 J-1.0000 as given in the following block of CNC code:

N120 G17 G02 X3.Y2. 10. J-1. F7.2

- The 1.000 incremental J value is calculated by adding the .250-inch cutter radius to the .750 part radius.
- A minus value is required since the direction from the cutter centerline to the arc centerline is in a minus direction
- The spindle is really generating a 1.000-inch arc when the cutter center is taken into account

Circular Interpolation

From Circle Center

- The **from circle center** method is the same as the **to circle center** **except the incremental coordinate is specified looking from the center of the arc to the center of the cutter**
- The signs associated with the **I**, **J**, and **K** addresses will be the **reverse** of the **to circle center** method
- The following line of code specifies the arc coordinates when the cutter is positioned at location #3, Figure 9

N120 G17 G02 X3.Y2. 10. J1. F7.2

Notice that the only difference between this block of code and the one given previously is the sign of the J address.

Seams W., "Computer Numerical Control, Concepts & Programming"

Circular Interpolation

Radius Method

- When using **the radius method**, the programmer only needs to **specify the radius to be cut** when programming the cut

- Instead of using the **I**, **J**, and/or **K** addresses **the R address** is used to specify the **arc radius**

- The following block of CNC code moves the cutter from location #3, Figure 9 to location #4 using the radius method

```
N120 G17 G02 X3.Y2. R1. F7.2
```

Notice that the radius to be cut is still 1.000 inch

- The controller is commanding motion of the spindle centerline. It does not know that there is a .500-inch diameter cutter in the spindle. The true cutter path is still a 1.000-inch arc

Circular Interpolation

- Although the radius method is easier to use than the arc vector method, the latter method is still common

- This is most likely because the radius method became available only with the advent of modern CNC controllers

- Many of today's programming practices have ties to the tape-controlled MCU of days gone by. This use of the arc vector method is one of these

Milling the Arc

- Putting together all these pieces, the following sections of CNC code will mill the part surface in Figures 8 and 9

Computer Aided Manufacturing

(CAM)

Definition

Computer Aided Manufacturing (CAM) can be defined as the use of computer systems to **plan**, **manage** and **control** the **operations of a manufacturing plant** through either direct or indirect computer interface with the plant's production resources

- In other words, the **use of computer system in non-design activities** but in manufacturing process is called CAM (Elanchezhian et al. 2007)

Strategic Role of CAM

- The application of CAM in the production offers advantages to a company to **develop capabilities** by combining traditional economies of scale with economies of scope resulting in the **desired flexibility and efficiency**

Definition

Strategic Role of CAM

- Amongst other **benefits** provided by CAM, Post identifies the following (Post 2003):
  - **Greater supervision of the production**
  - **Fast response to changes in market demand**
  - **Greater flexibility**
  - **Product variety**
  - **Small lot-sizes**
  - **Distributed processing capability**
  - **Reduced waste**

Application of CAM in the Production

- The utilization of **CAM** enables the **automation** and **computer support** of all the production activities on the **shop floor**, in order to manufacture parts designed with **computer-aided design (CAD)** and analysed with **computer-aided engineering (*CAE*)**

- The equipment on the shop-floor, such as **robots**, **controllers**, **machine tools** and **machining centres** are controlled and operated using **CAM** systems (Post 2003)

- **CAM** technologies comprise **NC machines**, **expert systems**, **machine vision**, **robots**, **lasers** and **FMS** technologies used alongside with **computer hardware**, **databases** and **communication technologies**

- **CAM** systems are tightly connected with ***CAD* systems**

Application of CAM in the Production

- The **CAD** databases must reflect the manufacturing requirements such as tolerances and features
- The *part drawings must be designed having in mind* **CAM** *r*equirements. Moreover, the manufacturing systems nowadays require **high coordination** due to their ***networking*** **characteristics**
- Synchronization among robots, vision systems, manufacturing cells, material handling systems and other shop floor tasks are challenging tasks that **CAM** addresses
- The role of **CAD**/**CAM** systems in the production can be as the intersection of five sets:
    - design tools
    - manufacturing tools
    - geometric modelling
    - computer graphics concepts and
    - networking concepts (Zeid 1991)

Application of CAM in the Production

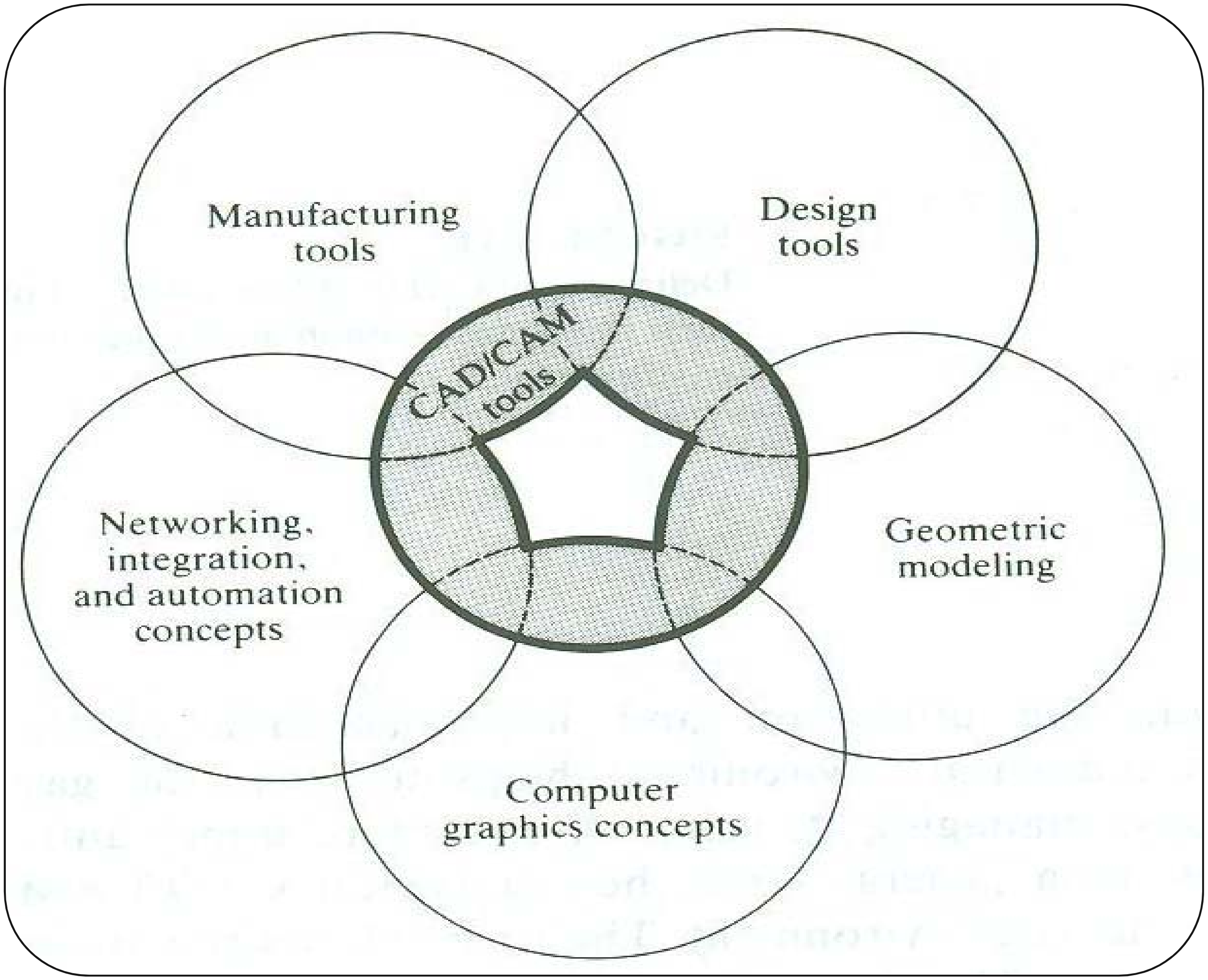


Figure 3: CAD/CAM and their constituents

Application of CAM in the Production

- Apart from the fact that the **CAM** technology has brought a revolution in manufacturing systems by enabling **mass production** and **greater flexibility** (Yeung2003)

- It has also enabled the **direct link between the three-dimensional (3D)** ***CAD*** **model and its production**

- The **data exchange** between **CAM**, **CAD** and **CAPP** is a **dynamic procedure** and takes place through various production stages

Application of CAM in the Production

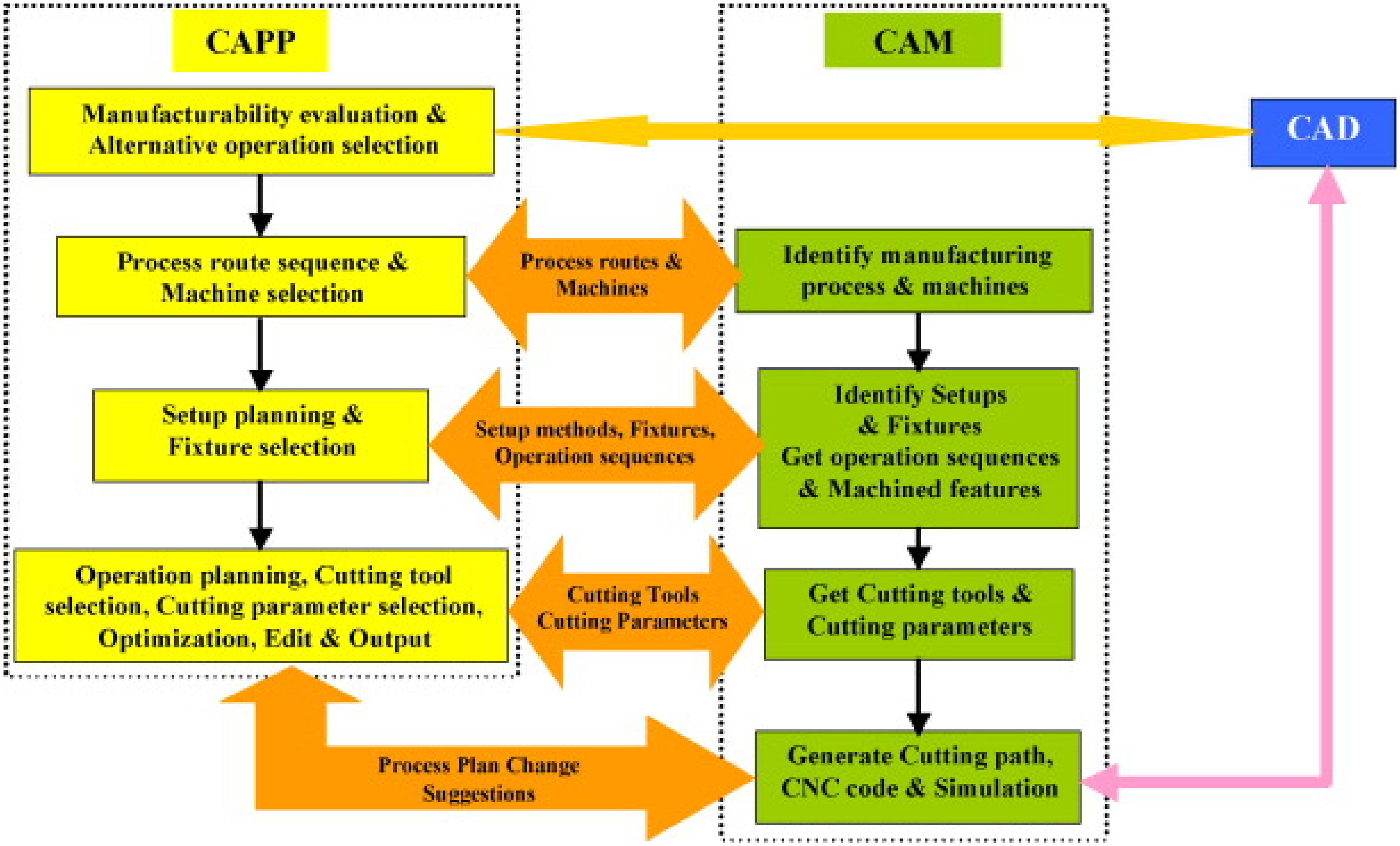


Figure 5:The collaboration between CAM, CAPP and CAD systems (Ming et al.2008)

CNC in Manufacturing

Computer Integrated Manufacturing (CIM)

- Includes all of the **engineering functions of CAD/CAM**
- Also includes the firm's business functions that are related to manufacturing
- **Ideal CIM system applies computer and communications technology to all of the operational functions and information processing functions in manufacturing**

From order receipt → **Through design and production** → **To product shipment**

Automation, Production Systems, and Computer-Integrated Manufacturing, Third Edition, by Mikell P. Groover.

Flexible Manufacturing Systems

(FMS)

Flexible Manufacturing System

- A **Flexible Manufacturing System (FMS)** is a system of CNC machines, robots, and part transfer vehicles that can take a part from raw stock or casting and perform all necessary machining, part handling, and inspection operations to make a finished part or assembly

An **FMS** is an entire **unmanned, software-based, manufacturing/ assembly** line

- An **FMS** consists of **four major components**:

1. CNC machining centres
2. Coordinate measuring machines
3. Part handling and assembly robots
4. Part / tool transfer vehicles