# COMPUTER NUMERICAL CONTROL OF MACHINE TOOLS

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


**Dr. Dimitris Mourtzis**

**Associate professor**

Patras, 2017

# Chapter 8:
# Two – Axis Programming

# Table of Contents

# Objectives

- Identify the **basic parts of a CNC program**
- Describe the **word address code** format
- Write simple **two-axis programs** in word address format to perform **hole** operations
- Write simple **two-axis milling** programs using the word address format
- Write simple **two-axis programs** that combine **milling and hole operations**

# Introduction

- This chapter is concerned with **manual** programming of CNC machinery
- For purposes of continuity the **same machine** will be used for the next several chapters
- No two CNC machines program exactly alike
- However, learning to program the machine used in the examples, only **minimal effort** will be required to program other CNC machines
- Programming is done in a format called **Word Address** which is the most common machine code format used today
- **The machine programmed in this chapter is a vertical machining center**

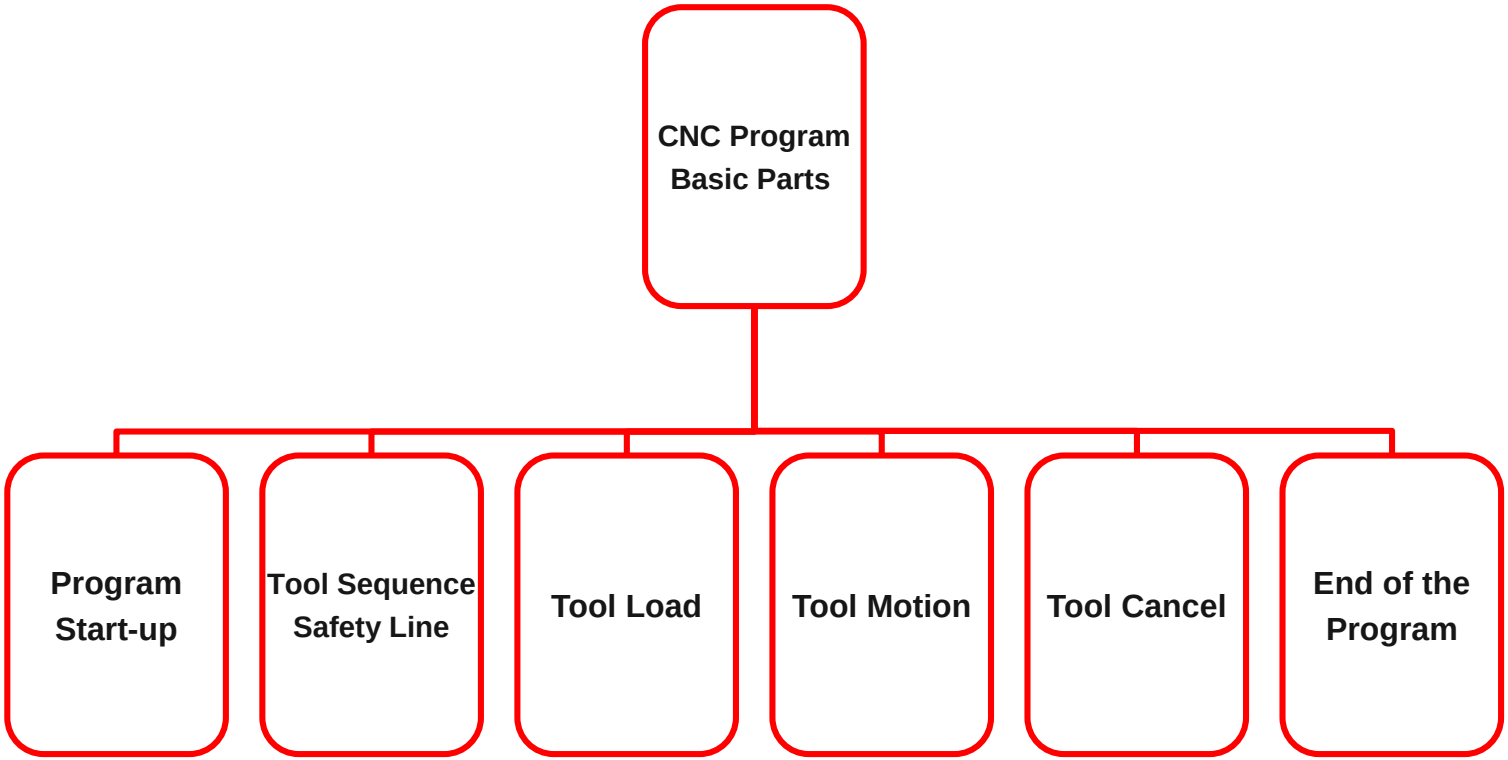
# Introduction

- The machining center is equipped with a **FANUC Machine Control Unit (MCU)**
- It is a **Continuous Path Type Machine**
- The program codes used on FANUC MCU are similar to those used on other MCUs such as **General Numeric and General Electric**
- **Two-Axis mill** programming is not so common in real world application but **in educational level is a prerequisite for easier understanding of Three-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



# Parts of a CNC Program

## Program Startup

- The **program startup** section serves to issue any commands required at **the start of the tape only**
- For instance, setting the program to **inch mode** would only be required at the **beginning of the program**

## Tool Safety Block

- The **tool sequence safety block(s)** serves to issue commands to **cancel for any machine modes** that could have been left active if the machine operator interrupted the tool cycle
- By issuing a safety block, the programmer and operator **know the state of the machine at the beginning of the tool cycle**

# Parts of a CNC Program

## Tool Load Blocks

- The tool load section are those blocks **of a tool sequence where the tool is placed in the spindle**, either manually or by the machine's automatic tool changing mechanism
- The **tool length compensation** is turned on

## Tool Motion Blocks

- The tool motion section contains the code for the ***actual cutting tool motion***
- It is where all the machining work is actually done

# Parts of a CNC Program

## Tool Cancel Blocks

- The tool cancel section turns off the tool length compensation and ***returns the tool to the tool change position***
- All active cycle commands should be turned off in this section and the control left in a state ready to load the next tool

## End of Tape Blocks

- The end of program blocks ***issue any commands necessary after all tool motion is complete***, but before the program terminates
- Often this section consists simply of the end of program code

# Word Address Format

## Characteristics

- Word Address was developed as a **tape programming format**
- Word Address is also named **Variable Block Format** because the program **lines (blocks)** may vary in length according to the information contained in them
- Earlier tape formats required an entry for all possible machine registers
- In these earlier formats a **zero was programmed as a null input** if the register values were to be unaffected
- In Word Address **the blocks need only contain necessary information**
- Although Word Address was developed as a Tape Format is used as the format for **M**anual **D**ata **I**nput (**MDI**) on many CNC machines

# Word Address Format

## 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.

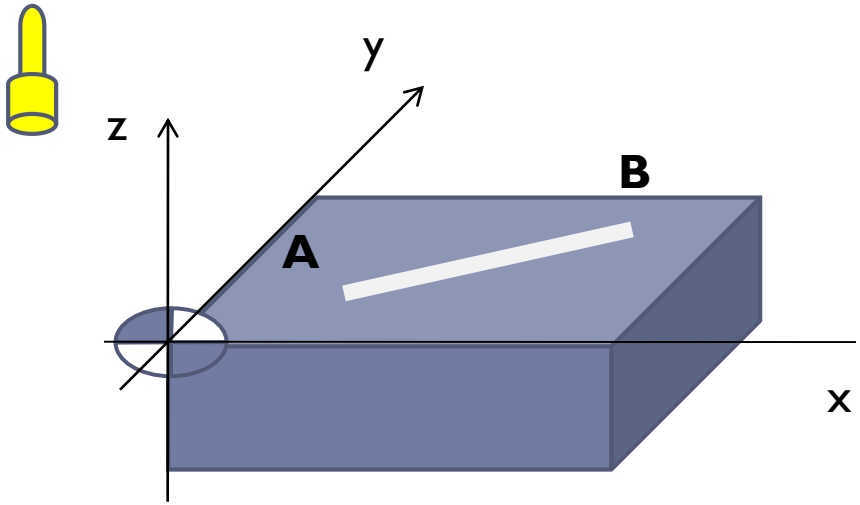
**M49**-0verride cancel on.

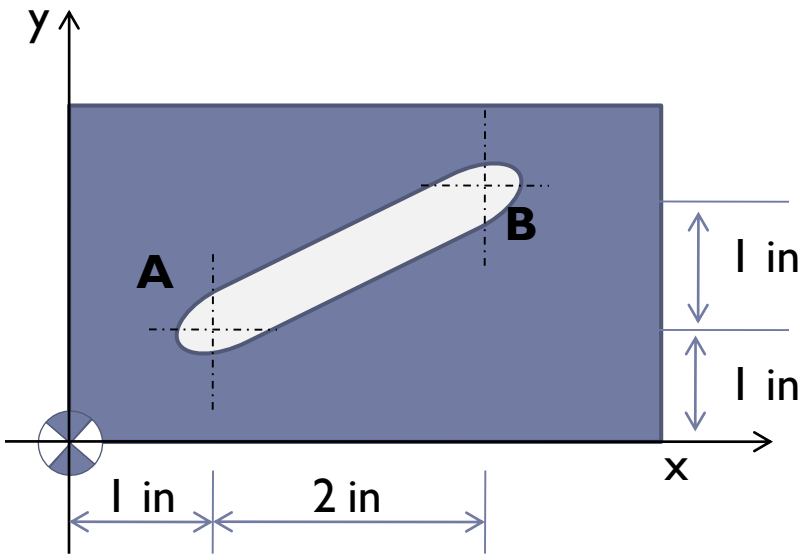
**M98**-Jump to subroutine.

**M99**-Return from subroutine.

# Example

- Construct a linear hole of 0.125 inches depth, from point A to point B




NOTE

1. Specify machining strategy
2. Design analysis based on zero point
3. Write CNC code

| Point | X | Y |
|---|---|---|
| A | 1 | 1 |
| B | 3 | 2 |

(Modern methods of processing materials and programming with PC, D. Mourtzis et al)

# Program phases

| | | |
|---|---|---|
| %<br>:1045<br>N5 G90 G70<br>N10 M06 T2<br>N15 M03 S1200 | Start program<br>Program number<br>Absolute positioning/inch mode<br>Halts program for tool change, tool #2<br>Spindle rotation, CW, 1200 rpm | Phase 1 |
| N20 G00 X1 Y1<br>N25 Z0.125<br>N30 G01 Z-0.125 F5<br>N35 X3 Y2 F10<br>N40 G00 Z1<br>N45 X0 Y0 | Rapid movement at point (X1,Y1)<br>Rapid movement at Z axis to Z0.125<br>Feedrate 5ipm, at depth Z-0.125<br>Feedrate 10ipm, στο σημείο (X3,Y2)<br>Rapid movement at height Z1<br>Rapid movement at point (X0,Y0) | Phase 2 |
| N50 M05<br>N55 M30 | Spindle stop<br>End of program | Phase 3 |

(Modern methods of processing materials and programming with PC, D. Mourtzis et al)

# Drilling Example – Absolute Positioning

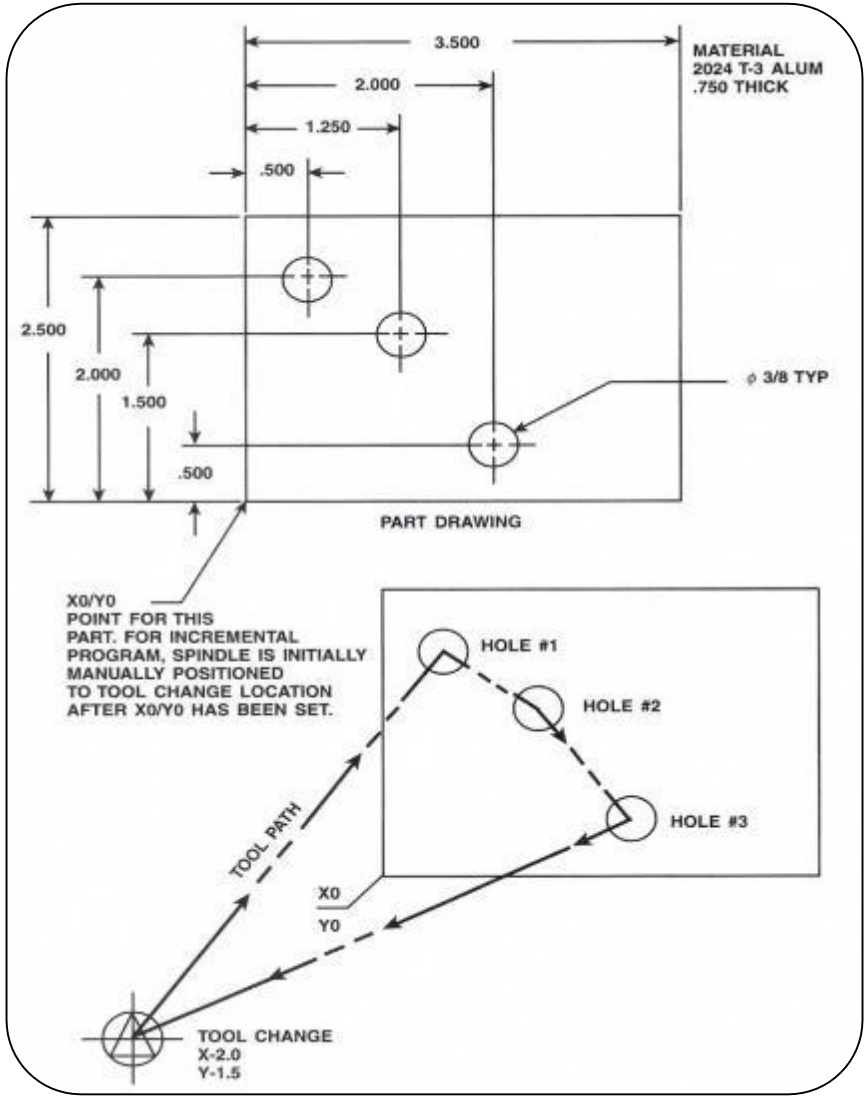



**Figure 1: Hole operations part drawing, *inch***

```
%
O6003
(************************************)
(* PROGRAM X0/Y0 = TOOL CHANGE)
(* SET TOOL CHANGE X-2.0 Y-1.5 PRIOR TO)
(* STARTING FIRST CYCLE)
(* ************************************)
(TOOL 1 - 3/8 DRILL)
(SPINDLE SPEED 2500 RPM)
N010 G00 G70 G90 M06   (Program and tools startup)
N020 X.5 Y2.
N030 M00
N040 X1.25 Y1.5        (Tool motion blocks)
N050 M00
N060 X2. Y.5
N070 M00
N080 X-2. Y-1.5        (Tool cancellation block)
N090 M30               (End of program block)
%
```

**Figure2:Drilling program, *inch absolute positioning* for the part in Figure 1**

# Drilling Example – Absolute Positioning

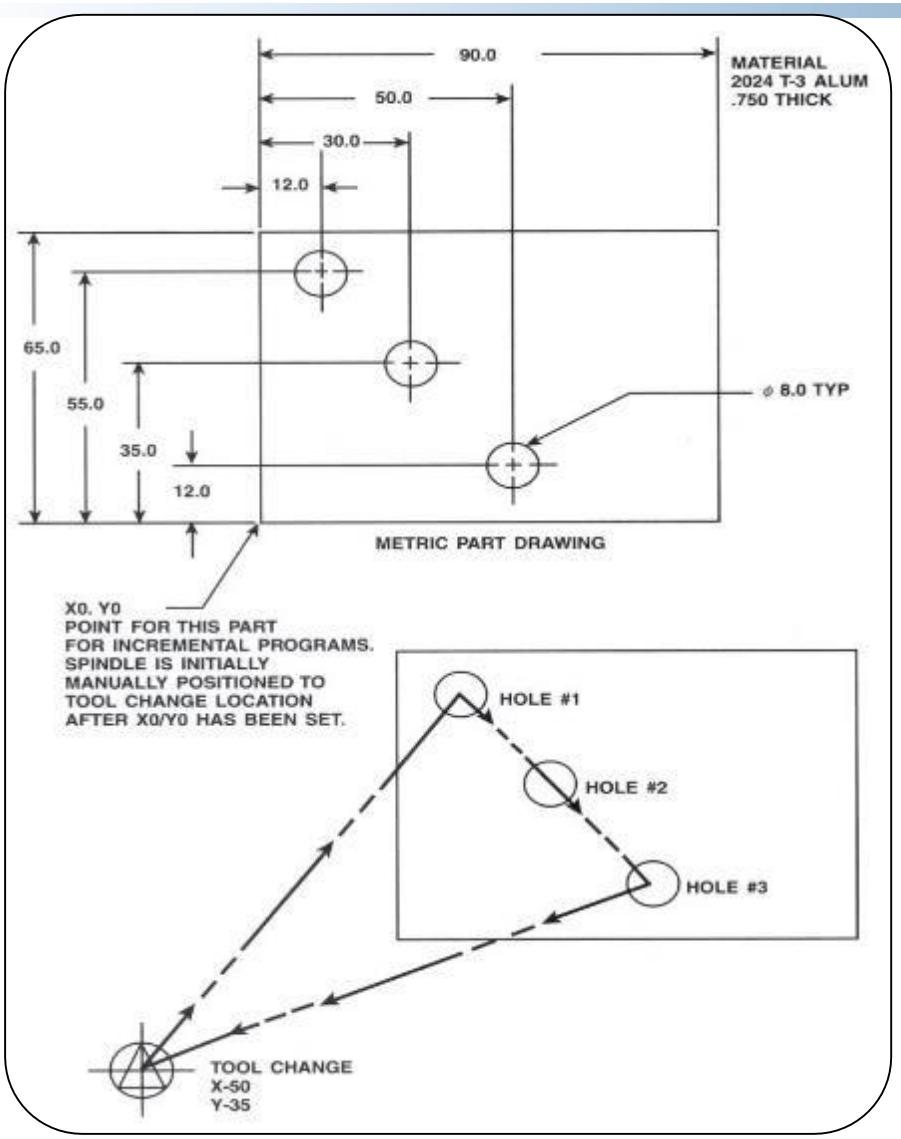



**Figure 3: Hole operations part drawing, *metric***

```
%
O6004
(* ************************************)
(* PROGRAM X0/Y0 = TOOL CHANGE)
(* SET TOOL CHANGE X-50. Y-35. PRIOR TO)
(* STARTING FIRST CYCLE.)
(* ************************************)
(TOOL 1 - 3/8 DRILL)
(SPINDLE SPEED 2500 RPM)
N010 G00 G71 G90 M06
N020 X12. Y55.
N030 M00
N040 X30. Y35.
N050 M00
N060 X50. Y12.
N070 M00
N080 X-50. Y-35.
N090 M30
%
```

**Figure 4:Drilling program, metric absolute positioning, for the part in Figure 3**

# Drilling Example – Absolute Positioning

**Program Explanation**

**O6003**

This is the program number. All FANUC-style controllers (and many other controls) use the letter "O" to designate the start of a program in controller memory

**N010**

N010 - This is the sequence number

- Each block (line) of the program begins with a unique number, prefaced with the N address
- This program follows the practice of numbering the blocks by tens
- This particular control does not care if the numbers are not in order
- Numbering the program by tens allows any blocks inserted by way of in-shop editing to be numbered consecutively

**G00** - Puts the machine in rapid traverse mode

- All machine moves made while **G00** is active will be made at maximum speed

**G70** - Puts the machine in inch input mode

- All program coordinates will be read in decimal inches

**G90** - Puts the machine in absolute positioning mode

**M06** - issues the tool change command. In this two-axis milling program it causes the control to stop and wait for the operator to install a tool-holder in the spindle

# Drilling Example – Absolute Positioning

**N020**

N020 - The block sequence number

- X/Y coordinates
- Cause the machine to position the tool from the tool change position to hole #1

**N030**

N030 - Block sequence number

- **M00** - Program stop command
- This halts the program execution allowing the operator to drill the holes

**N040**

N040 - Block sequence number

**X/Y** coordinates - to move from hole #1 to hole #2.

**N050**

N050 - Block sequence number

**M00** - Program stop command - This halts the program so that hole #2 can be drilled

**N060**

N060 - Block sequence number

**X/Y** coordinates - To move from hole #2 to hole #3

# Drilling Example – Absolute Positioning

**N070**

N070 - Block sequence number

**M00** - Program stop command - Hole #3 is drilled.

**N080**

N080 - Block sequence number

**X/Y** coordinates - to move from hole #3 to tool change

**N090**

N090 - Block sequence number

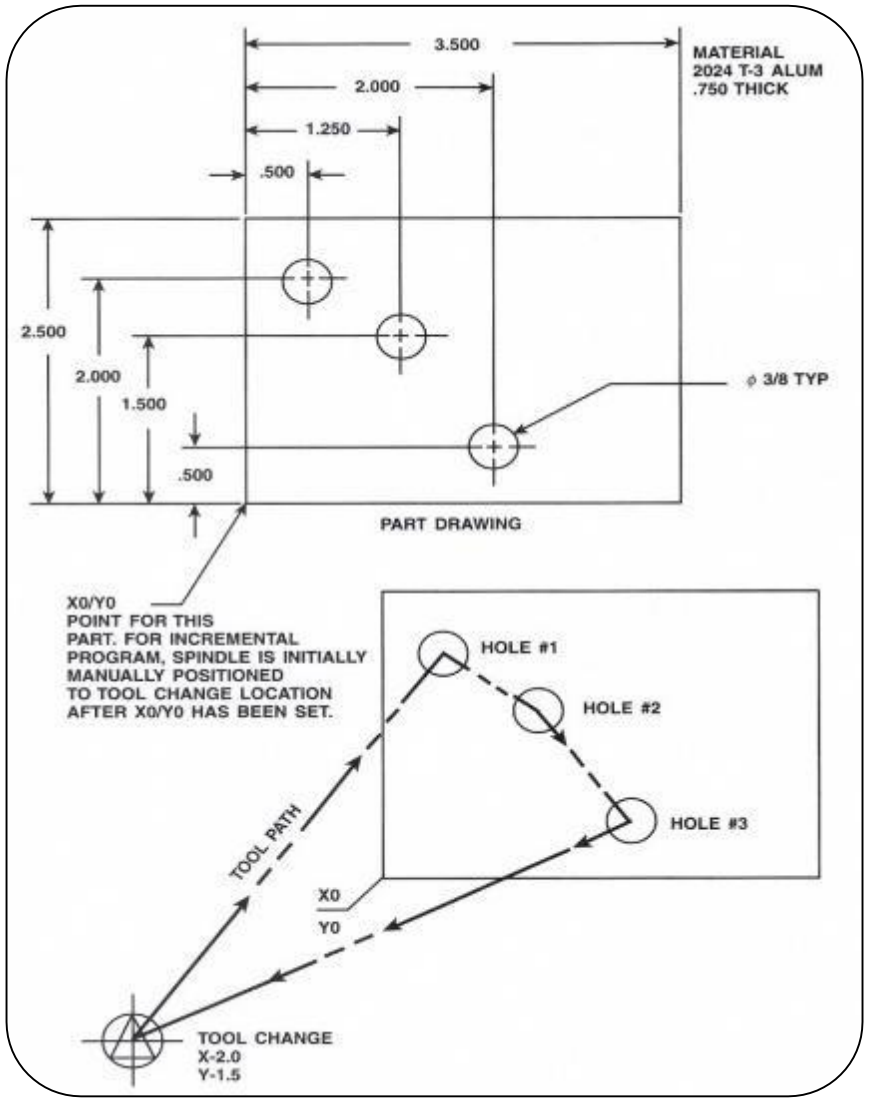
**M30** - Signals that the program has ended and resets the computer's memory to the start of the program

**%**

(%) is used on FANUC-style controllers as an end of program marker

This second percent sign signals the program read port on the MCU to stop reading characters from the input device

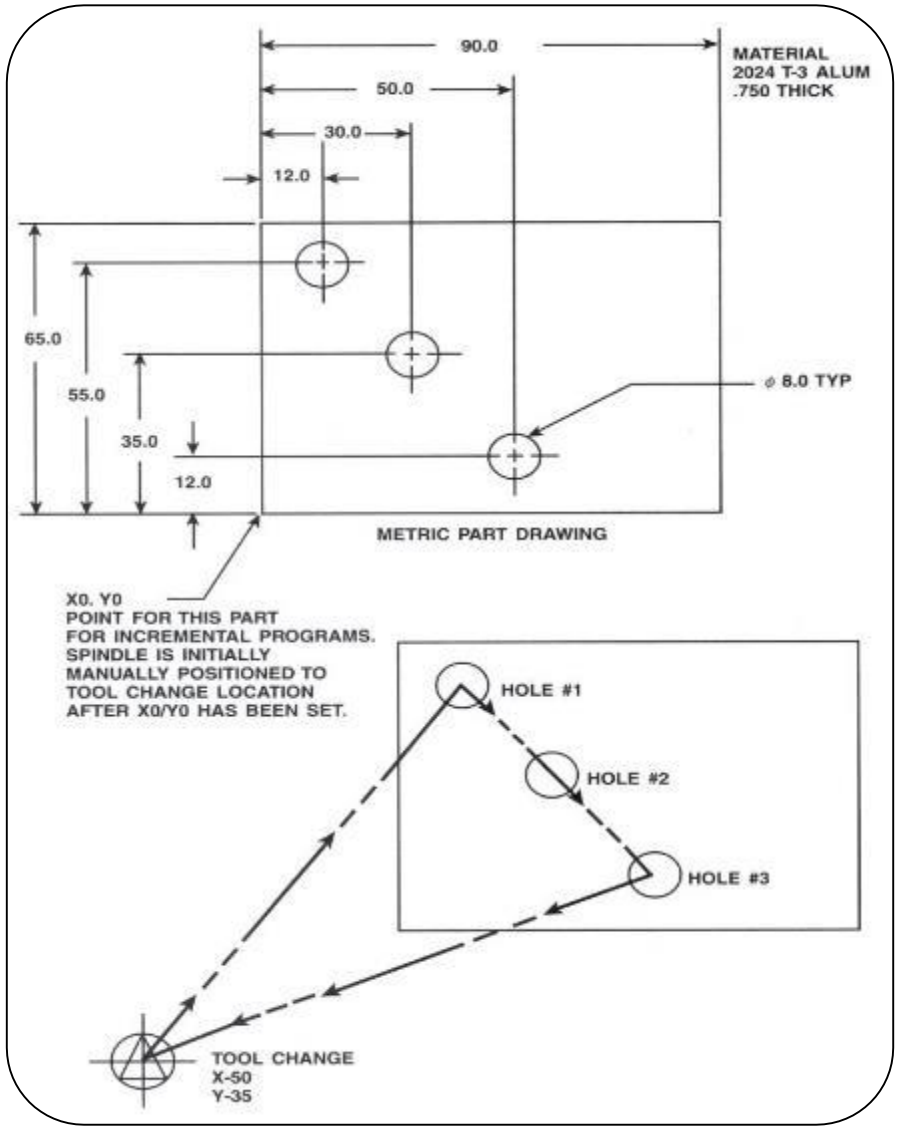
# Drilling Example – Incremental Positioning




```
%
O6005
(* ************************************)
(* PROGRAM X0/Y0 = TOOL CHANGE)
(* SET TOOL CHANGE X-2.0 Y-1.5 PRIOR TO)
(* STARTING FIRST CYCLE.)
(* ************************************)
(TOOL 1 - 3/8 DRILL)
(SPINDLE SPEED 2500 RPM)
N010 G00 G70 G91 M06
N020 X2.5 Y3.5
N030 M00
N040 X.75 Y-.5
N050 M00
N060 X.75 Y-1.
N070 M00
N080 X-4. Y-2.
N090 M30
%
```

**Figure 5:Drilling program, *inch incremental* positioning, for the part in Figure 1**

# Drilling Example – Incremental Positioning




```
%
O6006
(* ************************************)
(* PROGRAM X0/Y0 = TOOL CHANGE)
(* SET TOOL CHANGE X-50. Y-35. PRIOR TO)
(* STARTING FIRST CYCLE.)
(* ************************************)
(TOOL 1 - 3/8 DRILL)
(SPINDLE SPEED 2500 RPM)
N010 G00 G71 G91 M06
N020 X62. Y90.
N030 M00
N040 X18. Y-20.
N050 M00
N060 X20. Y-23.
N070 M00
N080 X-100. Y-47.
N090 M30
%
```

**Figure 6:Drilling program, metric incremental positioning, for the part in Figure 2**

# Drilling Example – Incremental Positioning

**Program Explanation**

**%**

Percent sign (%) - program start code.

**O6005**

This is the program number {the "0" number).

**N010**

N010 - The block sequence number.

**G00** - Puts the machine in rapid traverse mode. All machine moves made while **G00** is active will be made at maximum speed.

**G70** - Puts the machine in inch input mode. All program coordinates will be read in decimal inches.

**G91** - Puts the machine in incremental positioning mode. In the absolute positioning example, the **G90** code was used.

**M06** - Issues the tool change command. In this two-axis milling program, it causes the control to stop and wait for the operator to install a tool-holder in the spindle.

**N020**

N020 - The block sequence number.

**X/Y** coordinates - Causes the machine to position the tool from the tool change position to hole #1.

**N030**

N030 - Block sequence number.

**M00** - Program stop command. This halls the program execution, allowing the operator to drill the holes.

**N040**

N040 - Block sequence number.

**X/Y** coordinates-to move from hole #1 to hole #2.

# Drilling Example – Incremental Positioning

**N050**

N050 - Block sequence number.

**M00** - Program stop command. This halts the program so that hole #2 can be drilled.

**N060**

N060 - Block sequence number.

**X/Y** coordinates - To move from hole #2 to hole #3.

**N070**

N070 - Block sequence number.

**M00** - Program stop command. Hole #3 is drilled.

**N080**

N080 - Block sequence number.

**X/Y** coordinates - to move from hole #3 to tool change.

**N090**

N090 - Block sequence number.

**M30** - Signals that the program has ended and resets the computer's memory to the start of the program.

**Percent sign (%)** - Program stop code.

# Milling

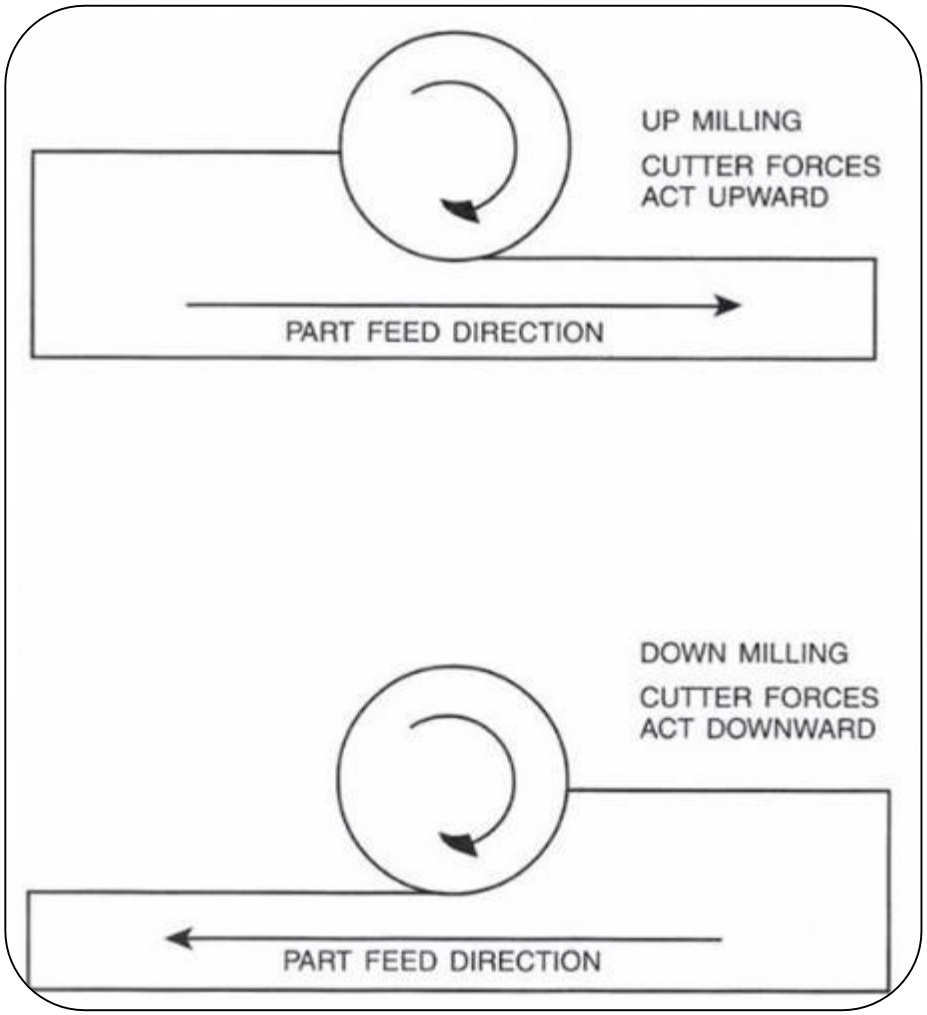



**Figure 7a: Up milling and down milling**
(Seams W., "Computer Numerical Control, Concepts & Programming")

## Up Milling (Conventional)

- Cutting forces acting on the part try to lift part up of the table
- Cutting of ferrous materials, brass and bronze

## Down Milling (Climb Cutting)

- Cutting forces try to push the part on the table
- Finishing cuts on Aluminium and Aluminium alloys
- Not suitable for hardened materials
- Less power required
- More stress on the machine slides and ball screws

# Milling

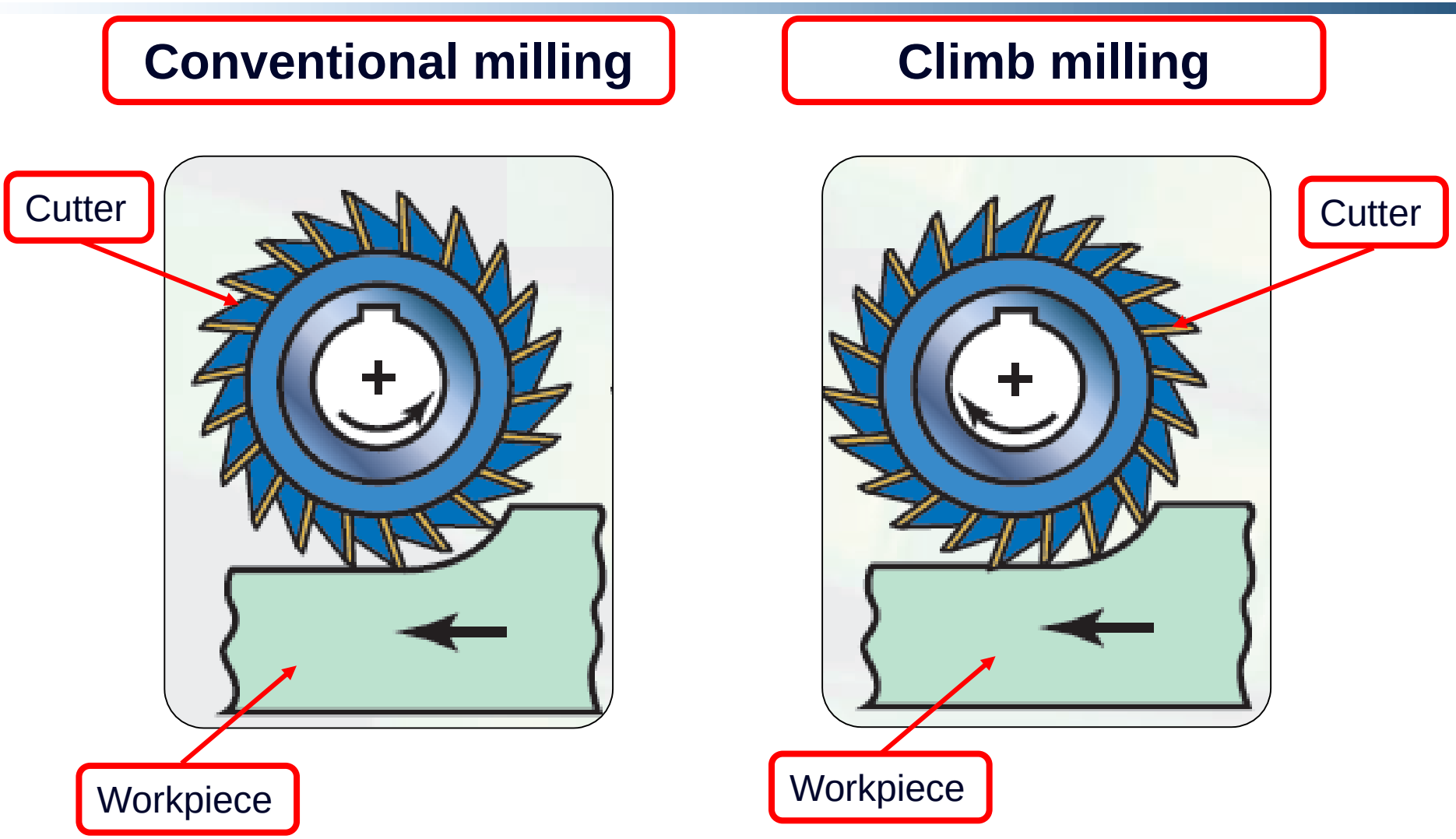
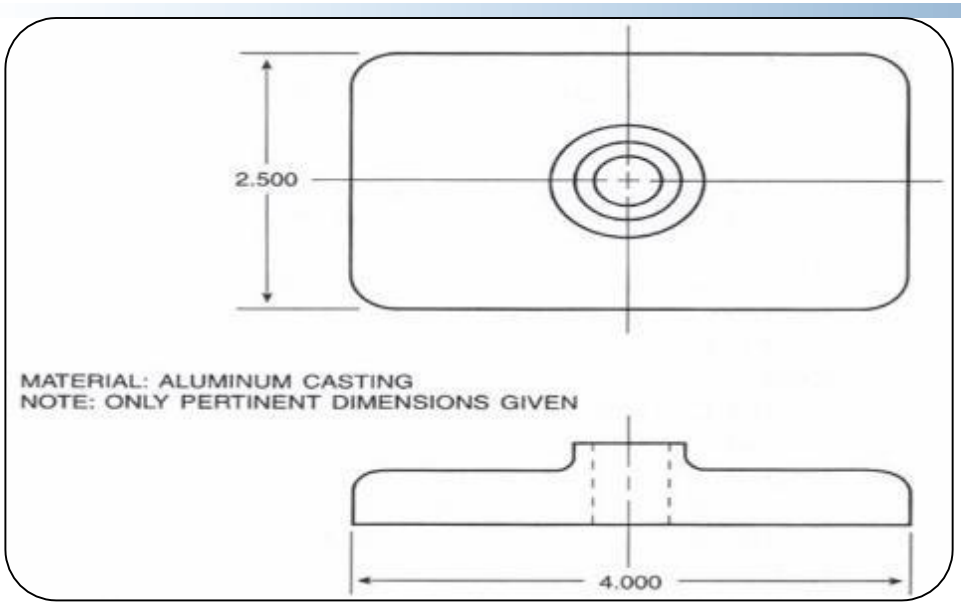
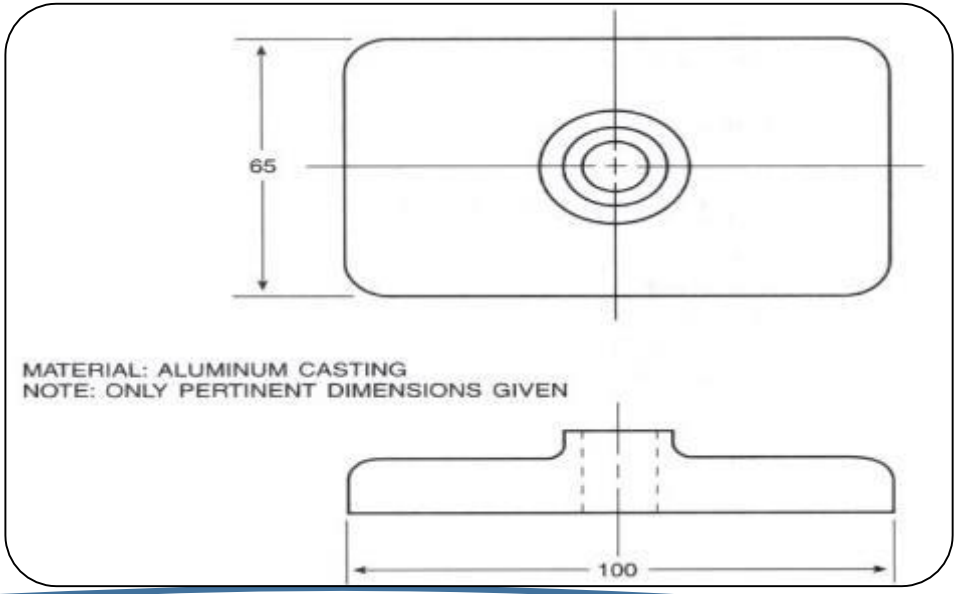



**Figure 7b: Difference between conventional milling and climb milling**

(Manufacturing Processes for Engineering Materials, 5th ed. Kalpakjian , Schmid)

# Milling Example 1




- The part of Fig. 8 to be **milled**
- The part is an aluminium cast
- The **length** and **width** to be milled only
- **Clamping** will be done through the center hole
- Two passes around the part will be made:
  1. A roughing pass and a finishing pass
  2. O.10 inch will be left for the finishing pass

**Figures 8, 9:Milling part drawing, inch and metric**

# Milling Example 1

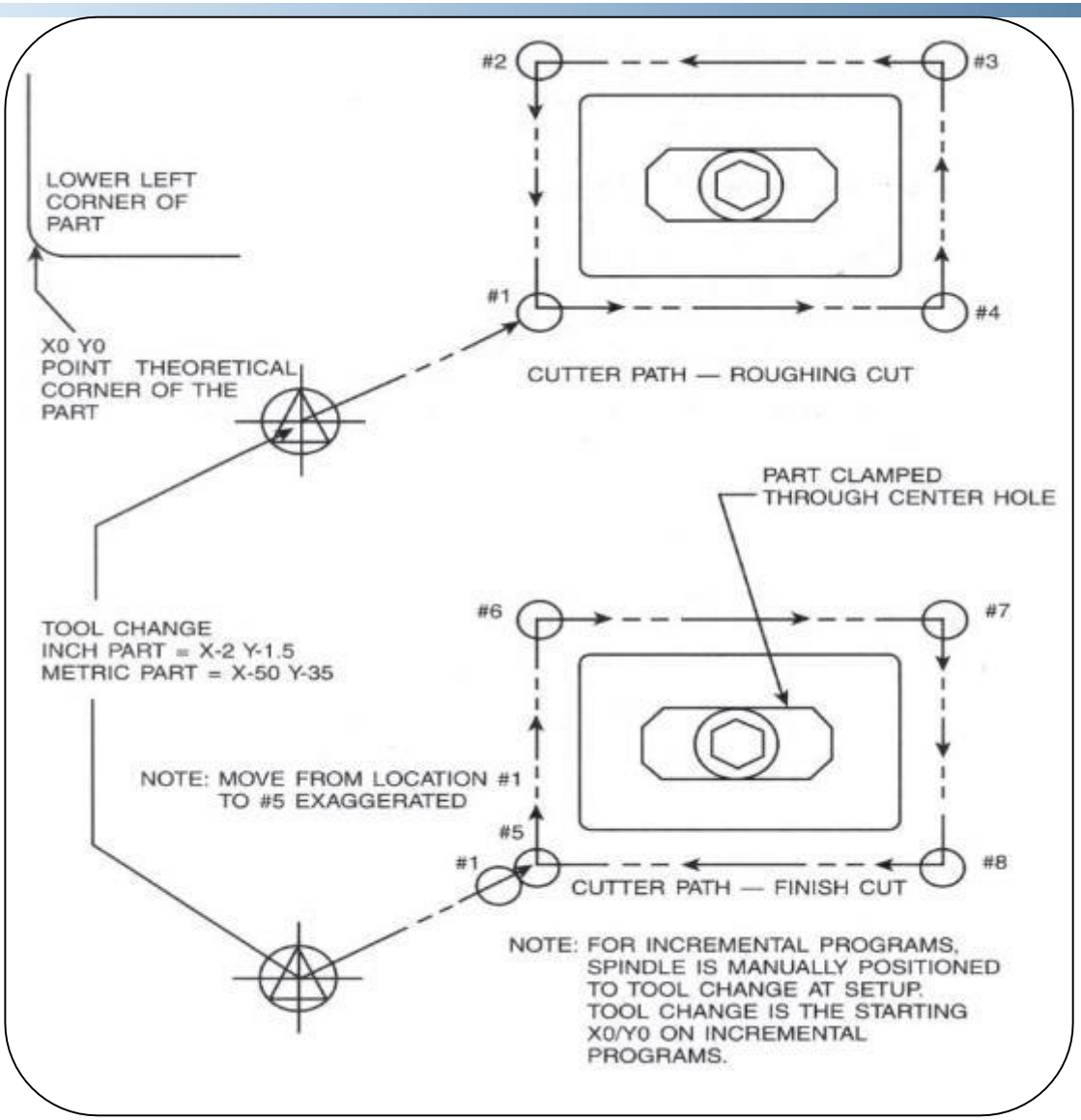



**Figure 10: Setup drawing for the part in Figures 8 and 9**

# Milling Example 2

```
%
O6011
(*************************************************)
(* THIS PROGRAM USES ABSOLUTE POSITIONING)
(* X/Y ORIGIN IS LOWER LEFT CORNER OF PART)
(* PLACE 1/2 END MILL IN SPINDLE PRIOR TO CYCLE
      START)
(*************************************************)
(SET PARAMETERS TO RAPID - INCH INPUT -ABS.
      POS.)
N010 G00 G70 G90
(AT PROG. STOP - LOWER SPINDLE AND CLAMP)
N020 X-.26 Y-.26
N030 M00
(BEGIN ROUGH MILL CUT AT FEEDRATE)
N040 G01 X4.26 F20
N050 Y2.76
N060 X-.26
N070 Y-.26
```

```
( “Program Continue”)

(BEGIN FINISH MILL CUT)
N080 X-.25 Y-.25
N090 Y2.75
N100 X4.25
N110 Y-.25
N120 X-.25
(AT PROG. STOP UNCLAMP AND RAISE SPINDLE)
N130 M00
(RETURN TO TOOL CHANGE LOCATION AND END CYCLE)
N140 G00 X-2. Y-1.5
N150 M30
%
```

**Figure 11:Milling program: *inch absolute positioning* for the part in Figure 8**

# Milling Example 2

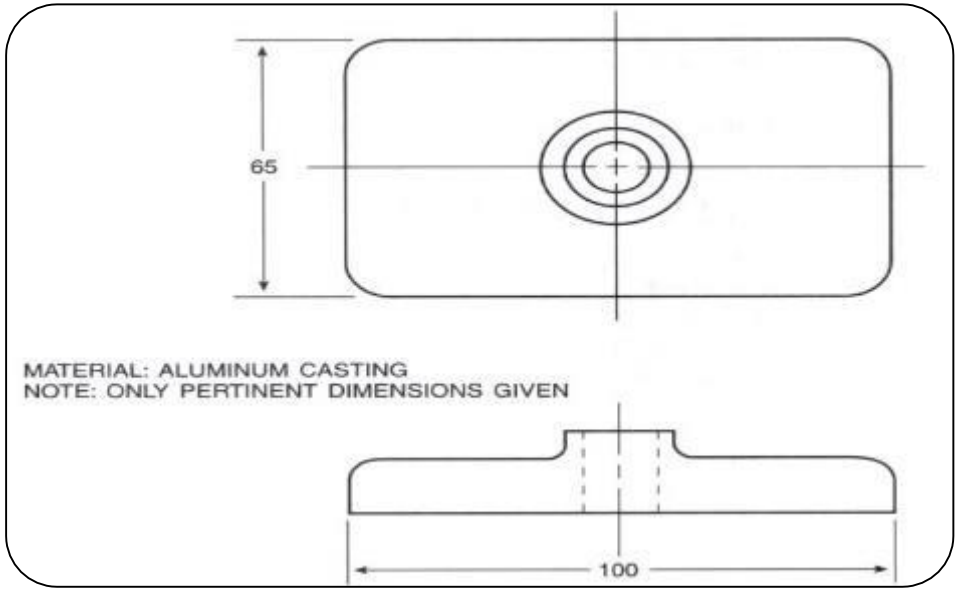



**Figure 12:Milling program, metric absolute positioning for the part in Figure 9**

```
%
O6012
(*************************************************)
(* THIS PROGRAM USES ABSOLUTE POSITIONING)
(* X/Y ORIGIN IS LOWER LEFT CORNER OF PART)
(* PLACE 1/2 END MILL IN SPINDLE PRIOR TO CYCLE
      START)
(*************************************************)
(SET PARAMETERS TO RAPID - INCH INPUT -ABS. POS.)
N010 G00 G71 G90
(AT PROG. STOP - LOWER SPINDLE AND CLAMP)
N020 X-2.75 Y-2.75
N030 M00
(BEGIN ROUGH MILL CUT AT FEEDRATE)
N040 G01 X102.75 F500.
N050 Y67.75
N060 X-2.75
N070 Y-2.75
(BEGIN FINISH MILL CUT)
N080 X-2.5 Y-2.5
N090 Y67.5
N100 X102.5
N110 Y-2.5
N120 X-2.5
(AT PROG. STOP UNCLAMP AND RAISE SPINDLE)
N130 M00
(RETURN TO TOOL CHANGE LOCATION AND END CYCLE)
N140 G00 X-50. Y-35.
N150 M30
%
```

# Milling Example 2

**Program Explanation**

**06011**

The program number

**N010**

N010-The block sequence number.

**G00**-Puts the machine in rapid traverse mode.

**G70**/**G71**-Puts the machine in inch or metric mode.

**G90**-Selects absolute positioning

**N020**

N020-Block sequence number

**X/Y** coordinates-To move to location #1

**N030**

N030-Sequence number

**M00**-Program stop. This command halts the program execution to allow the machine operator to lower and clamp the spindle

**N040**

N040-Sequence number

**G01**-Puts the machine in feedrate mode (also called linear interpolation mode)

**X** coordinate-Moves spindle from position #1 to #4

**F20**.-Specifies a feedrate of 20 inches per minute is to be used during feedrate moves

**N050**

N050-Sequence number

**Y** coordinate-Moves spindle from position #4 to #3

# Milling Example 2

**N060**

N060-Sequence number

**X** coordinate-Moves spindle from position #3 to #2

**N070**

N070-Sequence number

**Y** coordinate-Moves spindle from position #2 to #1

**N080**

N080-Sequence number

**X/Y** coordinates-Move spindle from position #1 to #5

**N090**

N090-Sequence number

**Y** coordinate-Moves spindle from position #5 to #6

**N100**

N100 - Sequence number

**X** coordinate - Moves spindle from position #6 to #7

**N110**

N110 - Sequence number

**Y** coordinate - Moves spindle from position #7 to #8

**N120**

N120 - Sequence number

**X** coordinate - Moves spindle from position #8 to #5

# Milling Example 2

**N130**

N130 - Sequence number

**M00** - Program stop code - This command halts the program execution, allowing the machine operator to unclamp and raise the spindle.

**N140**

N140 - Sequence number.

**G00** - Places the machine in rapid traverse mode. X/Y coordinates  Move the spindle from position #5 to the tool change location

**N150**

N150 - Sequence number

**M30** - Signals end of program

**%**

Percent sign - Program

# Milling and Drilling Example

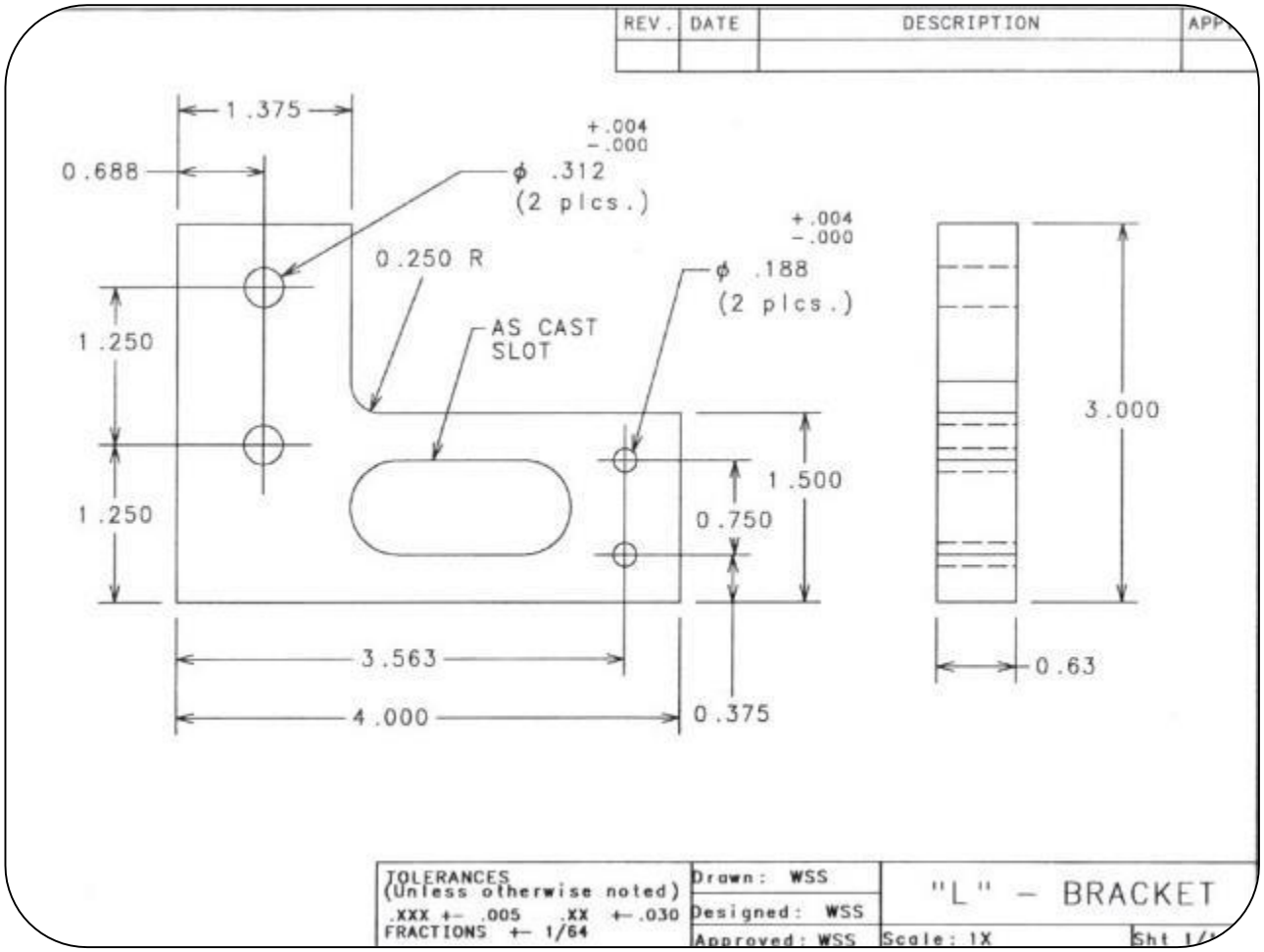



**Figure 13:Milling and drilling part drawing**

# Milling and Drilling Example

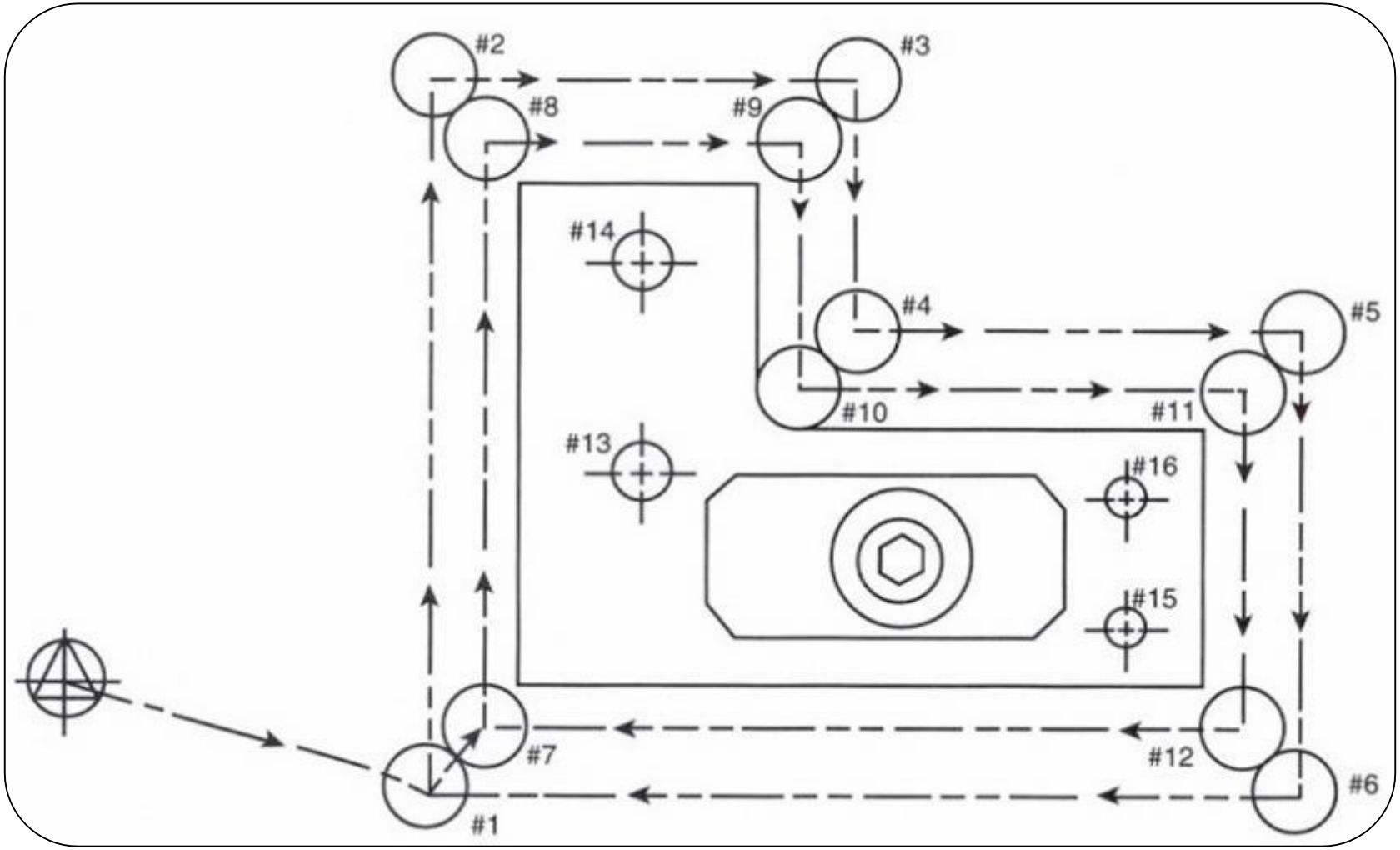



**Figure 14:Cutter path for milling the part in Figure 13**

# Milling and Drilling Example

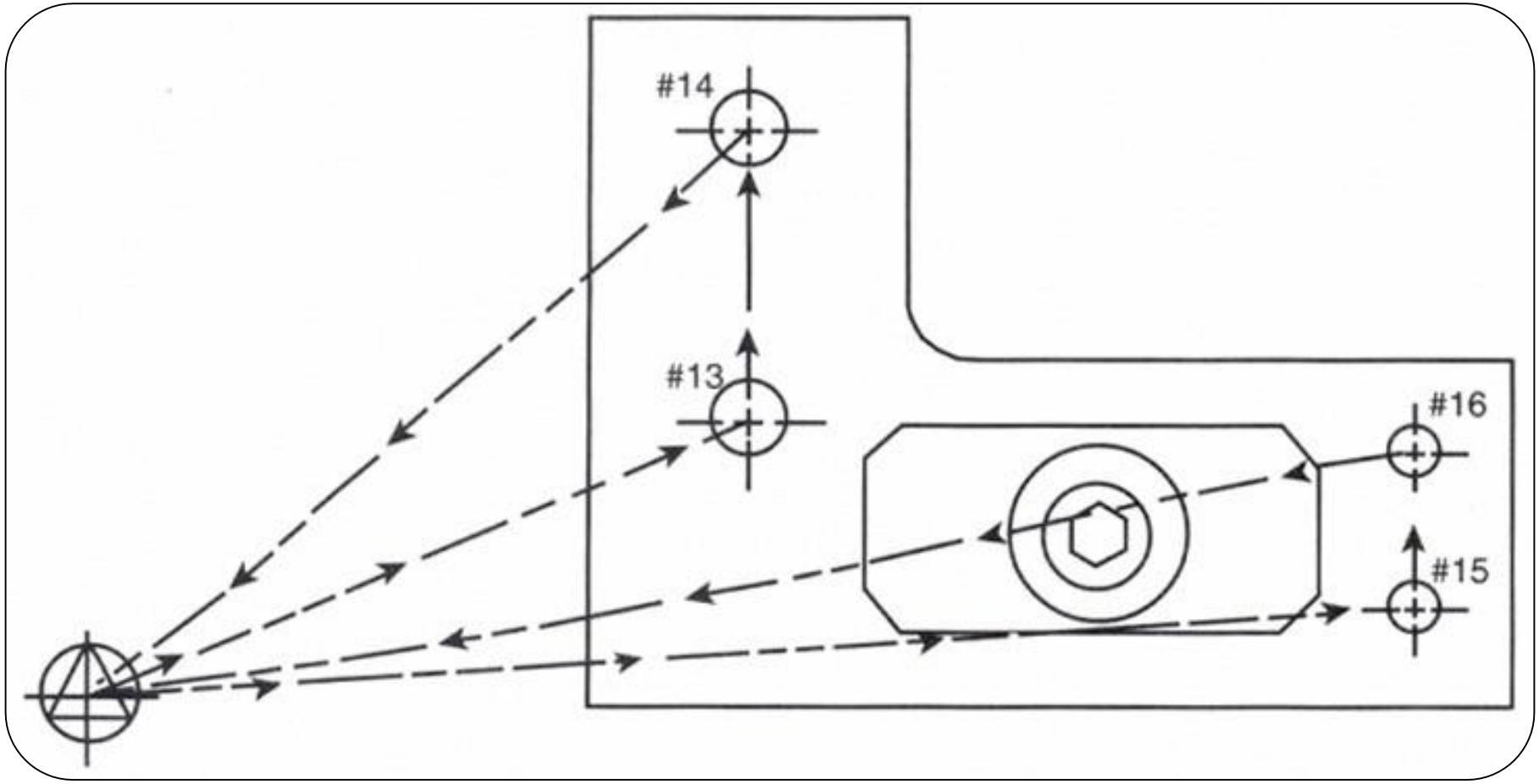



**Figure 15 :Cutter path for drilling the part in Figure 13**

# LAB Milling and Drilling Example

```
%
06016
(* ******************)
(* L-BRACKET 06/09/93)
(* ******************)
N001
(* ************)
(* TOOL NO. 1)
(* .500 DIA. END MILL)
(* ROUGH/FINISH PART PERIPHERY)
(* ************)
N100 G00 G90 G70
N101 X.26 Y.26 S1200 M03
N102 M00
(LOWER AND CLAMP SPINDLE)
(BEGIN ROUGH MILL PASS - LEAVE .01 STK/SIDE)
N103 G01 Y3.26 F7.2
N104 X1.635
N105 Y1.76
N106 X4.26
N107 Y-.26
N108 X-.26
(BEGIN FINISH MILL PASS)
N109 X-.25 Y-.25
N110 Y3.25
N111 X1.625
N112 Y1.75
N113 X4.25
N114 Y-.25
N115 X-.25
N116 M00
(UNCLAMP AND RAISE SPINDLE)
(AT CYCLE START RETURNS TO TOOL CHANGE)
N117 G00 X-10. Y0.
N118 M00
```

```
N002
(* ************)
(* TOOL NO. 2)
(* 5/16 STUB DRILL)
(* DRILL THE .312 DIA. HOLES)
(* ************)
N200 G00 G90 G70
N201 X.688 Y1.25 S1500 M03
N202 M00
(DRILL HOLE)
N203 Y2.5
N204 M00
(DRILL HOLE)
N205 X-10. Y.0
N206 M00

N003
(* ************)
(* TOOL NO. 3)
(* 3/16 STUB DRILL
(* DRILL .188 DIA. HOLES)
(* ************)
N300 G00 G90 G70
N301G00 X3.563 Y.375 S2000 M03
N302 M00
(DRILL HOLE)
N303 Y1.125
N304 M00
(DRILL HOLE)
N305 G00 X-10. Y.0
N306 M30
%
```

**Figure 16 :Program to mill and drill the part in Figure 13**

# LAB Milling and Drilling Example

**Program Explanation**

**%** Percent sign – Program start code

**O6018**

Program ‘’O’’ number

**N001**

N001 – Block sequence number, used as a separator line

**N100**

N100 – Sequence number

**G00** – Selects rapid traverse mode

**G90** – Selects absolute positioning mode

**G70** – Selects inch input mode

**N101**

N101-Sequence number.

**X/Y** coordinates-Positions spindle at location #1.

**S1200**-Selects a spindle speed of 1200 to be used for milling the part. M03- Turns spindle on clockwise.

**N102**

N 102-Sequence number.

**M00**-Program stop command. Halts program execution so operator can lower and clamp the spindle.

**N103**

N103-Sequence number. **G01**-Selects feedrate mode.

**Y** coordinate-Moves spindle from position #1 to #2. **F7.2**-Sets the feed rate to 7.2 inches per minute.

# Milling and Drilling Example

**N104**

N104-Sequence number.

**X** coordinate-Moves spindle from position #2 to #3.

**N105**

N105-Sequence number.

**Y** coordinate-Moves spindle from position #3 to #4.

**N106**

N 106-Sequence number.

**X** coordinate-Moves spindle from position #4 to #5.

**N107**

N107-Sequence number.

**Y** coordinate-Moves spindle from position #5 to #6.

**N108**

N108-Sequence number.

**X** coordinate-Moves spindle from position #6 back to #1.

**N109**

N109-Sequence number.

**XIY** coordinates-Moves spindle from position #1 to #7.

**N110**

N110-Sequence number.

**Y** coordinate-Moves spindle from position #7 to #8.

# LAB Milling and Drilling Example

**N111**

N111-Sequence number.

**X** coordinate-Moves spindle from position #8 to #9.

**N112**

N112-Sequence number.

**Y** coordinate-Moves spindle from position #9 to #10.

**N113**

N113-Sequence number.

**X** coordinate-Moves spindle from position #10 to #11.

**N114**

N114-Sequence number.

**Y** coordinate-Moves spindle from position #11 to #12.

**N115**

N115-Sequence number.

**X** coordinate-Moves spindle from position #12 back to #7.

**N116**

N116-Sequence number.

**M00** program stop code-Halts program execution so operator can unclamp and raise the spindle.

**N117**

N117-Sequence number.

***XIY*** coordinates-Move the spindle from position #7 back to the tool change location.

# Milling and Drilling Example

**N118**

N118-Sequence number.

**M00**-Halts program for a tool change.

**N002**

N002-Block sequence number used as a separator line for tool 2.

**N200**

N200-Sequence number.

**G00**-Selects rapid traverse mode. **G90**-Selects absolute positioning mode. **G70**-Selects inch input mode.

**N201**

N201-Sequence number.

***XIY*** coordinates-Position spindle at location #13.

**S1500**-Selects a spindle speed of 1500 rpm to be used for drilling the holes.

**N202**

N202-Sequence number.

**M00**-Program stop command. Halts program execution so operator can drill the hole.

**N203**

N203-Sequence number.

**Y** coordinate-Moves spindle from hole #13 to #14.

**N204**

N204-Sequence number.

**M00**-Program stop command. Halts program execution so operator can drill the hole

# Milling and Drilling Example

**N205**

N205-Sequence number.

**XJY** coordinates-Move spindle back to the tool change position.

**N206**

N206-Sequence number.

**M00**-Halts program for a tool change.

**N003**

N003-Block sequence number used as a separator line for tool 3.

**N300**

N300-Sequence number.

**G00**-Selects rapid traverse mode.

**G90**-Selects absolute positioning mode.

**G70**-Selects inch input mode.

**N301**

N301-Sequence number.

**XJY** coordinates-Position spindle at location #15.

**S2000**-Selects a spindle speed of 2000 rpm to be used for drilling the holes.

**M03**- Turns spindle on clockwise.

**N302**

N302-Sequence number.

**M00**-Program stop command. Halts program execution so operator can drill the hole.

# Milling and Drilling Example

**N303**

N303-Sequence number.

**Y** coordinate-Moves spindle from hole #15 to #16.

**N304**

N304-Sequence number.

**M00**-Program stop command. Halts program execution so operator can drill the hole.

**N305**

N305-Sequence number.

**XJY** coordinates-Move spindle back to the tool change position.

**N306**

N306-Sequence number.

**M30**-Signals end of program.

**%**

Percent sign-Program stop code.

# Summary 1/3

The important concepts presented in this chapter are:

- An **NC or CNC program** consists of **six basic parts**
  I. Program startup section
  II. Tool sequence safety line
  III. Tool load (or tool change) section
  IV. Tool motion sequence
  V. Tool cancel section
  VI. End of program section

- In word address format, *each CNC command is called a word.* Each word begins with an alpha address which identifies the command's function

- The address is followed by a *numeric value*. Some values are used to set machine modes. Others are used to specify positioning coordinates

# Summary 2/3

- The spindle must be positioned safely out of the way at the end of the program, to allow *safe loading and unloading of the workpiece*
- This is accomplished in both the milling and drilling examples by *sending the spindle back to its tool change location at the end of the program*
- Incremental programs differ from absolute programs only in the *coordinates* used
- Programs in absolute and incremental positioning use the same programming logic
- In *incremental positioning*, it is imperative that the machine start and stop in the same location. Failure to program for this will result in *incorrect positioning* for the second cycle

# Summary 3/3

- To perform *hole operations*, it is necessary to position the spindle over the centerline of the hole

- A *program stop command* is used at hole locations to halt the program and enable the operator to drill the hole

- When programming coordinates for milling, an *allowance* must be made for the size of the cutter

# Vocabulary Introduced in this chapter

- Addresses
- End of tape blocks
- Leading zero
- Program start-up blocks
- Tool cancel blocks
- Tool load blocks
- Tool motion blocks
- Tool safety blocks
- Trailing zero
- Two-axis programming

# References


1. Chryssolouris G., «Manufacturing Systems: Theory and Practice», 2nd Edition, 2006, Springer-Verlag
2. http://www.dptechnology.com/
3. Kalpakjian S., «Manufacturing Engineering and Technology», 2nd Edition, 1992, Addison-Wesley Publishing company
4. Manufacturing Processes for Engineering Materials, 5th ed.,2008,S.Kalpakjian - R.Schmid
5. Mattson M., “CNC Programming, Principles and Applications”, Delmar, 2002
6. Seams W., “Computer Numerical Control, Concepts & Programming”, 4th Edition, Delmar, 2002
7. Γ. Χρυσολούρης, «Συστήματα Παραγωγής Θεωρία και Πράξη» Μέρος Ι και ΙΙ, Εκπαιδευτικές Σημειώσεις, Πανεπιστήμιο Πατρών, 2001
8. Σύγχρονες μέθοδοι κατεργασίας υλικών και προγραμματισμός με Ηλεκτρονικό Υπολογιστή (Η/Υ) ,Δ. Μούρτζης ,Κ. Σαλωνίτης