



## PROGRAMMING WORKBOOK



HAAS AUTOMATION, INC.  
2800 Sturgis Rd.  
Oxnard, CA 93030

JUNE 1, 2000



# CONTENTS

INTRODUCTION .....	1
THE COORDINATE SYSTEM .....	2
MACHINE HOME .....	5
ABSOLUTE AND INCREMENTAL POSITIONING .....	5
POSITIONING EXERCISE.....	7
PROGRAMMING WITH CODES .....	8
PROGRAM STRUCTURE .....	9
PROGRAM FORMAT.....	13
MISCELLANEOUS FUNCTIONS (M CODES).....	16
PREPARATORY FUNCTIONS (G CODES) .....	18
MACHINE DEFAULTS .....	22
LIST OF CANNED CYCLES .....	23
ALPHABETICAL ADDRESS CODES .....	24
RAPID POSITION COMMANDS (G00).....	28
INTERPOLATION COMMANDS (G01).....	29
CIRCULAR INTERPOLATION COMMANDS (G02,G03).....	30
INTERPOLATION EXERCISE.....	39
PROGRAM START-UP LINES .....	40
PROGRAM ENDING LINES .....	41
DWELL (G04) .....	42
CIRCULAR POCKET MILLING (G12, G13).....	43
CIRCULAR POCKET MILLING EXERCISE .....	46
CIRCULAR PLANE SELECTION (G17, G18, G19) .....	47
INCH / METRIC SELECTION (G20, G21) .....	51
REFERENCE POINT DEFINITION AND RETURN (G28) .....	52
CUTTER COMPENSATION (G40, G41, G42) .....	53
CUTTER COMPENSATION EXERCISE #1 .....	59
CUTTER COMPENSATION EXERCISE #2 .....	62
HELICAL MOTION .....	64
TOOL LENGTH COMPENSATION (G43) .....	66
WORK COORDINATE SELECTION (G54-59, G110-129, G52, G53, G92)...	67
NON-MODAL COORDINATE SELECTION (G53).....	67
ANOTHER WAY TO SEND MACHINE HOME USING G53 SELECTION .....	68
CANNED CYCLES DESCRIPTION .....	69
CANCEL - CANNED CYCLE (G80) .....	70
DRILL - CANNED CYCLE (G81) .....	71
SPOT DRILL - CANNED CYCLE (G82) .....	72



# CONTENTS

DEEP HOLE PECK DRILLING CANNED CYCLE (G83).....	73
CANNED CYCLE EXERCISE #1 .....	76
TAPPING - CANNED CYCLE (G84) .....	78
BORE IN - BORE OUT - CANNED CYCLE (G85) .....	79
BORE IN - STOP - RAPID OUT - CANNED CYCLE (G86) .....	80
BORE IN - MANUAL RETRACT - CANNED CYCLE (G87) .....	81
BORE IN - DWELL - MANUAL RETRACT - CANNED CYCLE (G88) .....	82
BORE IN - DWELL - BORE OUT - CANNED CYCLE (G89) .....	83
CANNED CYCLE EXERCISE #2 .....	84
HIGH SPEED PECK DRILL - CANNED CYCLE (G73) .....	86
REVERSE TAPPING - CANNED CYCLE (G74) .....	90
FINE BORING - CANNED CYCLE (G76) .....	91
BACK BORE - CANNED CANNED CYCLE (G77).....	92
CANNED CYCLE RETURN PLANES (G98, G99) .....	93
CANNED CYCLES BOLT HOLE PATTERNS (G70, G71, G72) .....	94
CANNED CYCLES BOLT HOLE PATTERN (G70) .....	95
CANNED CYCLES BOLT HOLE PATTERN (G71) .....	96
CANNED CYCLES BOLT HOLE PATTERN (G72) .....	97
CANNED CYCLE EXERCISE #3 .....	98
SUBROUTINES (M97, M98, M99) .....	100
GENERAL PURPOSE POCKET MILLING (G150) .....	103
FINAL EXERCISE .....	108



## INTRODUCTION

This manual provides basic programming principles necessary to begin programming the HAAS C.N.C. Milling Machine.

In a “CNC” (Computerized Numerical Control) machine, the tool is controlled by a computer and is programmed with a machine code system that enables it to be operated with minimal supervision and with a great deal of repeatability.

The same principles used in operating a manual machine are used in programming a CNC machine. The main difference is that instead of cranking handles to position a slide to a certain point, the dimension is stored in the memory of the machine control once. The control will then move the machine to these positions each time the program is run.

In order to operate and program a CNC controlled machine, a basic understanding of machining practices and a working knowledge of math is necessary. It is also important to become familiar with the control console and the placement of the keys, switches, displays, etc., that are pertinent to the operation of the machine.

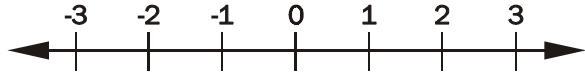
This workbook can be used for both operator's and programmer's. It is intended to give a basic understanding of CNC programming and it's applications. It is not intended as an in-depth study of all ranges of machine use, but as an overview of common and potential situations facing CNC programmers. Much more training and information is necessary before attempting to program on the machine.

This programming manual is meant as a supplementary teaching aid to users of the HAAS Mill. The information in this workbook may apply in whole or in part to the operation of other CNC machines. Its use is intended only as an aid in the operation of the HAAS Milling Machine. For a complete explanation and an in-depth description, refer to the Programming and Operation Manual that is supplied with your HAAS Lathe.



## THE COORDINATE SYSTEM

The first diagram we are concerned with is called a NUMBER LINE. This number line has a zero reference point that is called an ABSOLUTE ZERO and may be placed at any point along the number line.



Horizontal number line

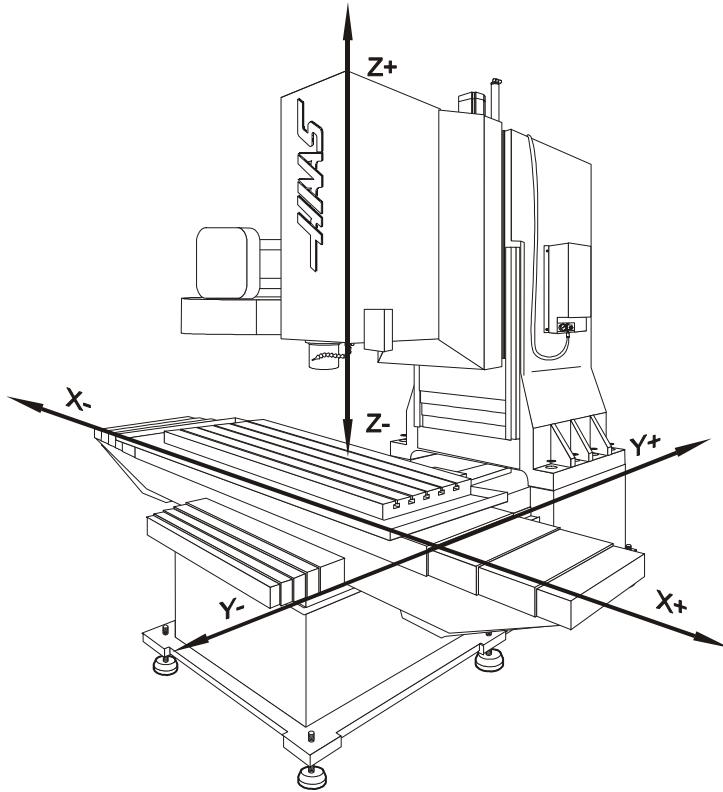
The number line also has numbered increments on either side of absolute zero. Moving away from zero to the right are positive increments. Moving away from zero to the left are negative increments. The "+", or positive increments, are understood, therefore no sign is needed. We use positive and negative signs along with increment value's to indicate its relationship to zero on the line. If we choose to move to the third increment on the *minus* (-) side of zero, we would call for -3. If we choose the second increment in the *plus* range, we would call for 2. Our concern is the distance and the direction from zero.

Remember that zero may be placed at any point along the line, and that once placed, one side of zero has negative increments and the other side has positive increments.

Vertical number line

The machine illustration shows three directions of travel available on a vertical machine center. To carry the number line idea a little further, imagine such a line placed along each axis of the machine. It shows the three directions to position the coordinates around a part origin, which is where these number lines intersect on a vertical machining center with the X, Y, and Z axis lines.

The first number line is easy to conceive as belonging to the left-to-right, or "X", axis of the machine. If we place a similar number line along the front-to-back, or "Y" axis, the increments (not the table) toward the operator, from Y zero, are the negative increments. The increments on the other side of zero away from the operator are positive increments.





The third axis of travel on our machine is the up-and-down, or "Z" axis. When we place a number line on the Z travel, the positive increments are up above zero, and the negative values are down below zero. The increments of each number line on HAAS machining centers equals .0001 inches. Also, while a line theoretically travels infinitely in either direction once established, the three lines placed along the X, Y, and Z axes of the machine do not have unlimited accessibility. That is to say, we are limited by the range of travel on the model of machining center.

MACHINE	X-axis travel	Y-axis travel	Z-axis travel
<b>VF-E / VF-0 / VF1</b>	20"	16"	20"
<b>VF-EXT / VF-OE</b>	30"	16"	20"
<b>VF2</b>	30"	16"	20"
<b>VF3</b>	40"	20"	25"
<b>VF4</b>	50"	20"	25"
<b>VF5</b>	50"	25"	25"
<b>VF6</b>	64"	32"	30"
<b>VF7</b>	84"	32"	30"
<b>VF8</b>	64"	40"	30"
<b>VF9</b>	84"	40"	30"
<b>VF10</b>	120"	32"	30"
<b>VF11</b>	120"	40"	30"
<b>VR11</b>	120"	40"	30"
<b>HS-1/1R/1RP</b>	24"	20"	22"
<b>HS-2RP</b>	38"	35"	30"

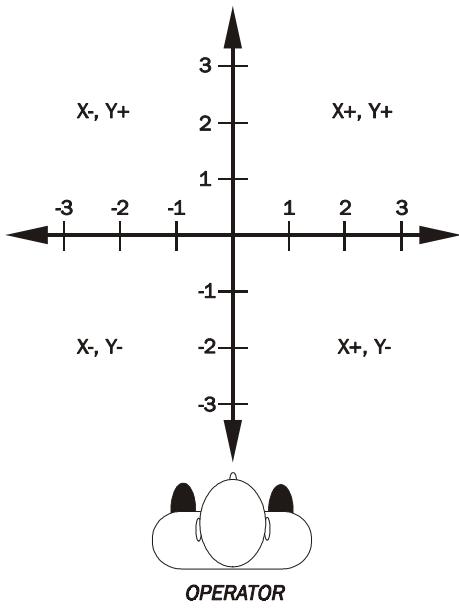
Remember, when we are moving the machine, we are concerned with positioning the center of the spindle in relation to X,Y and Z zero. Although the machine table is the moving part, we have to keep in mind our coordinates are based off our theoretical spindle movement.

Keep in mind that the part zero position may be defined at any point along each of the three axes, and will usually be different for each setup of the machine.

It is noteworthy to mention here that the Z-axis is set with the machine zero position in the upward position, or the tool change position. This will place most all Z moves in a negative range of travel.



Fig. 1-4 view shows the X, Y work zero grid from above. The work part zero for the Z-axis is usually set at the top of the part surface, and this will be entered in the tool length offset as a negative value for each tool. The range of Z-axis travel on the HAAS VF-1, for example, is 20 inches total; four of these inches are above tool change position and is listed as a positive tool length offset, and 16 inches are below tool change position and listed as a negative. The diagram shows a top view of the grid as it would appear on the machine tool. This view shows the X and Y axes as the operator faces a vertical machine table. Note that at the intersection of the two lines, a common zero point is established. The four areas on each side and above and below the lines are called "QUADRANTS" and make up the basis for what is known as rectangular coordinate programming.



two lines, a common zero point is established. The four areas on each side and above and below the lines are called "QUADRANTS" and make up the basis for what is known as rectangular coordinate programming.

**QUADRANT 1 IS ON THE TOP RIGHT = X+ Y+**  
**QUADRANT 2 IS ON THE TOP LEFT = X- Y+**  
**QUADRANT 3 IS ON THE BOTTOM LEFT = X- Y-**  
**QUADRANT 4 IS ON THE BOTTOM RIGHT = X+ Y-**

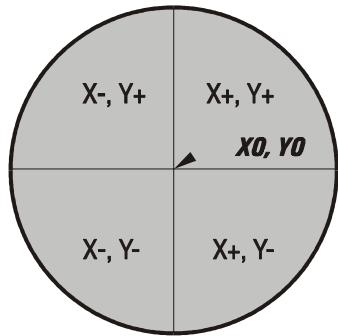
Whenever we set a zero point somewhere on the X-axis and, a zero point somewhere on the Y-axis, we have automatically set a work zero point and an intersection of the two number lines. This intersection where the two zeros come together will automatically have the four quadrants to its sides, above, and below it. How much of a quadrant we will be able to access is determined by where we place the zero point within the travel of the machine axes. For example, for a VF-1, if we set zero exactly in the middle of the travel of X and Y (table center), we have created four quadrants that are 10 inches by 8 inches in size.



## MACHINE HOME

When a zero return (ZERO RET) is performed at machine start up, all three axes are moved to extreme positive locations until limit switches are reached. When this condition is satisfied, the only way to move any of the three axes is in the negative direction (except for a positive four inches in Z-axis). This is because this position is defined as your MACHINE HOME for each of the three axes automatically when the machine was sent home with the POWER-UP/RESTART key.

In effect, now the positive quadrants cannot be reached from machine home position in X and Y axes, and all the moves will be found to be in the X-, Y- quadrant. It is only by setting a new part zero somewhere within the travel of each axes that other quadrants are able to be reached. Sometimes it is useful in the machining of a part to utilize more than one of these X,Y quadrants. An example of this is a round part that has it's datum lines running through the center. The setup of such a part may need machining to be performed in all four quadrants of a part. This is why you would want to make use of all four quadrants of the X and Y axes on a milling machine. As you gain more experience in machine tool programming and of setup techniques, you'll have a better understanding of how to position your machine tool and how to define a part zero origin and how to position a tool around that origin .



All four quadrants will have to be accessed to position around this part XY zero point.

would want to make use of all four quadrants of the X and Y axes on a milling machine. As you gain more experience in machine tool programming and of setup techniques, you'll have a better understanding of how to position your machine tool and how to define a part zero origin and how to position a tool around that origin .

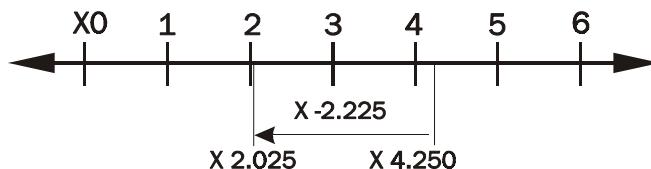
## ABSOLUTE AND INCREMENTAL POSITIONING

Up to this point, we have dealt with a system of positioning the tool that is known as absolute programming. In absolute, all coordinate positions are given with regard to their relationship to a fixed zero, origin point, that is referred to as part zero. This is the most common type of positioning.

Another type of positioning is called incremental positioning. Incremental positioning concerns itself with distance and direction from the last position. A new coordinate is entered in terms of its relationship to the previous position, and not from a fixed zero or origin. In other words, after a block of information has been executed, the position that the tool is now at is the new zero point for the next move to be made.



An example of the use of the incremental system is below. Note that to move from X 4.25 to X 2.025 on the scale, an incremental move of X -2.225 is made, even though the move still places the tool on the plus side of the scale. Therefore the move was determined from the last point, with no regard for the zero position. The + and - signs are used in terms of direction, and not in regard to the position of zero.



An example of an incremental move.

Keep in mind that when positioning in **absolute**, we are concerned with distance and direction from a fixed zero reference point, and when positioning in **incremental** we are concerned with distance and direction from the last position.

## ABSOLUTE / INCREMENTAL SELECTION

### G90 ABSOLUTE POSITION COMMAND

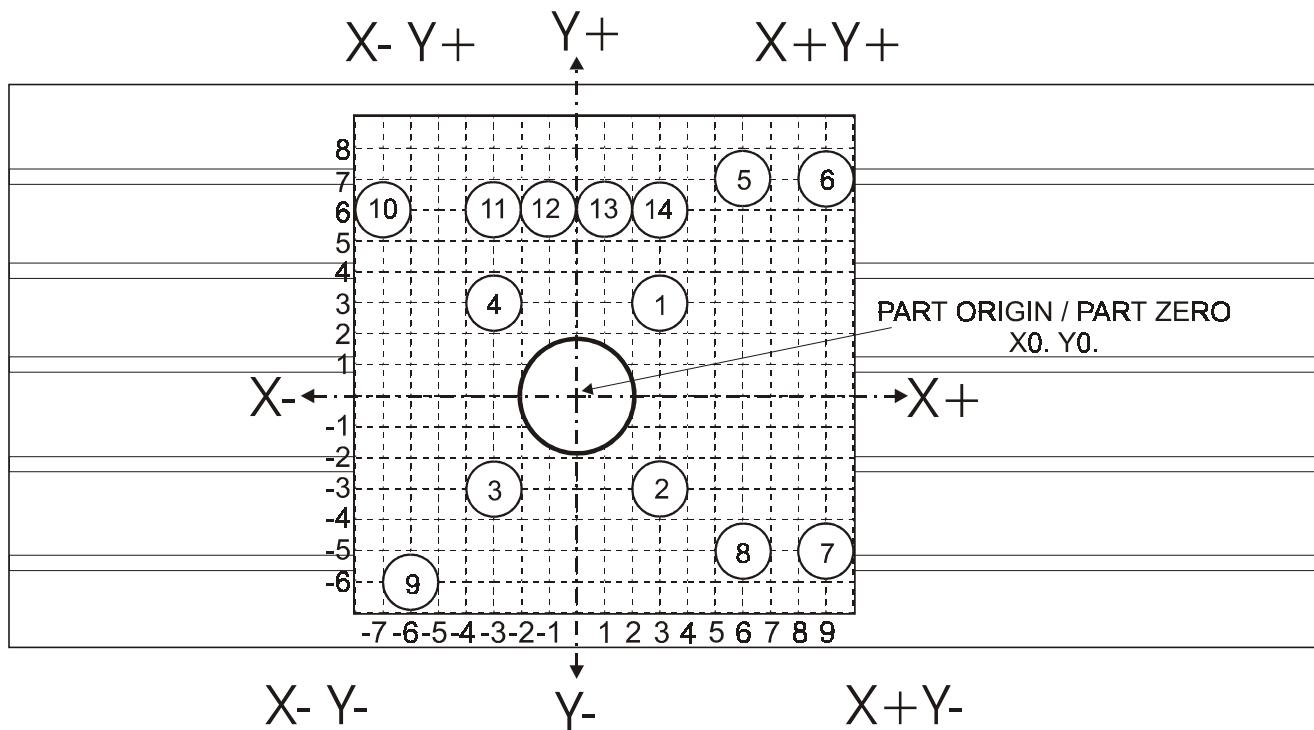
This code is modal and changes the way axis motion commands are interpreted. **G90** makes all subsequent commands absolute positions within the selected user coordinate system. Each axis that is moved will be placed at the position coded in the command block.

### G91 INCREMENTAL POSITION COMMAND

This code is modal and changes the way axis motion commands are interpreted. **G91** makes all subsequent commands incremental. Each axis that is moved will be moved by the amount coded in the command block.



## POSITIONING EXERCISE



What is the value in X and Y for each hole in absolute G90 positioning when each move is defined from a single fixed part zero point of an X0 Y0 origin point.

$$\begin{array}{ll}
 \text{PT1} = & \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{PT2} = & \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{PT3} = & \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{PT4} = & \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{PT5} = & \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{PT6} = & \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{PT7} = & \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{PT8} = & \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}}
 \end{array}$$

What is the value for each hole in INCREMENTAL G91 positioning when each move is defined from the previous position and the zero point shifts with the new position.

$$\begin{array}{llll}
 \text{From} & \text{PT8} & \text{to} & \text{PT9} = \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{From} & \text{PT9} & \text{to} & \text{PT10} = \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{From} & \text{PT10} & \text{to} & \text{PT11} = \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{From} & \text{PT11} & \text{to} & \text{PT12} = \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{From} & \text{PT12} & \text{to} & \text{PT13} = \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}} \\
 \text{From} & \text{PT13} & \text{to} & \text{PT14} = \text{X} \underline{\hspace{1cm}} \quad \text{Y} \underline{\hspace{1cm}}
 \end{array}$$



## PROGRAMMING WITH CODES

The definition of a part program for any CNC consists of movements of the tool, and speed changes to the spindle RPM. It also contains auxiliary command functions such as tool changes, coolant on or off commands, or external M code commands.

Tool movements consist of rapid positioning commands, straight line moves or movement along an arc of the tool at a controlled feedrate.

This milling machine has three (3) linear axes defined as X axis, Y axis, and Z axis. The X and Y axis will move the machine table below and around the spindles centerline, while the Z axis moves the tool spindle down toward or up and away from the machine table. The machine zero position is where the spindle is pointing down at the upper right corner, with the machine table all the way to the left in X axis and all the way toward you in the Y axis and Z axis is up at the tool change position. Motion in the X axis will move the machine table to the left with negative values and to the right with positive values. Motion in the Z axis will move the tool toward the machine table with negative values and away from the machine table with positive values.

A program is written as a set of instructions given in the order they are to be performed. The instructions, if given in English, might look like this:

**LINE #1 = SELECT CUTTING TOOL**  
**LINE #2 = RAPID TO STARTING POSITION OF PART, TURN COOLANT ON**  
**LINE #3 = TURN SPINDLE ON AND SELECT THE RPM**  
**LINE #4 = CHOOSE THE PROPER FEED RATE AND MAKE THE CUT(S)**  
**LINE #5 = TURN OFF THE SPINDLE AND THE COOLANT**  
**LINE #6 = RETURN TOOL TO HOLDING POSITION AND SELECT NEXT TOOL**

and so on. But our machine control understands only these messages when given in machine code, or G and M code programming.



## PROGRAM STRUCTURE

A CNC part program consists of one or more blocks of commands. When viewing the program, a block is the same as a line of text. Blocks shown on the CRT are always terminated by the “ ; ” symbol which is called an EOB. Blocks are made up of alphabetical address codes. Address codes are always an alphabetical character followed by a numeric value. For instance, the specification of the position to move the X-axis would be a number preceded by the X symbol.

Programs must begin and end with a % sign. The % signs are automatically entered in for you, if you enter the program in on a HAAS control.

A program may also contain a “ / ” symbol. The “ / ” symbol, sometimes called a slash, is used to define an optional block. If a block contains this symbol, any information that follows the slash in a program block, will be ignored when the BLOCK DELETE button is selected when running a program.

There is no positional requirement for the address codes. They may be placed in any order within the block. The following page is a sample program as it would appear on the control screen. The words following the “ : ” are not part of the actual program but are put there as further explanation.

This program will drill four holes and mill a two-inch hole in a four-inch square plate with X and Y zero at the center. The program with comment statements would appear like this.

%  
**001010****(MILL PART PROGRAM EXAMPLE)****N1 (DRILL 4 PLACES)****T1 M06 (½ IN. DIA. STUB DRILL)****G90 G54 G00 X-1.5 Y1.5 S1400 M03****G43 H01 Z1. M08****G73 G99 Z-0.625 Q0.2 R0.1 F5.****Y-1.5****/ X1.5****/ Y1.5****G80 G00 Z1. M09****G28 G91 Z0 M05****N2 (COUNTERSINK 4 PLACES)****T2 M06 (5/8 DIA. 90 DEG. C'SINK)****G90 G54 G00 X-1.5 Y1.5 S900 M03****G43 H02 Z1. M08****G82 G99 Z-0.27 P0.5 R0.1 F12.****Y-1.5****/ X1.5****/ Y1.5****G80 G00 Z1. M09****G28 G91 Z0 M05****N3 (SET D3 DIA. OFFSET VALUE TO .500)****T3 M06 (½ DIA. END MILL)****G90 G54 G00 X-2.3 Y2.3 S1100 M03****G43 H03 Z0.1 M08****G01 Z-0.625 F30.****G41 Y2. D03 F11.****X2.0****Y-2.0****X-2.0****Y2.25****G40 X-2.3 Y2.3****G00 Z1. M09****G28 G91 Y0 Z0 M05****T1 M06****M30****%**

:PROGRAM MUST BEGIN AND END WITH %  
:PROGRAM NUMBER, LETTER "O" WITH FOUR DIGIT NUMBER  
:(COMMENTS IN PARENTHESIS ARE IGNORED BY CONTROL)  
:(COMMENTS IN PARENTHESIS ARE IGNORED BY CONTROL)  
:TOOL CHANGE TO TOOL #1, (NOTES TO OPERATOR)  
:ABS POSIT, WORK OFFSET#, RAPID X Y, SPINDLE ON CW  
:TOOL LENGTH COMP #1, Z POSITION, COOLANT ON  
:HIGH SPEED PECK DRILLING, DRILL Z-.625 DEEP, .2 PECK  
:DRILL ANOTHER HOLE RAPID PLANE IS AT R.1  
:DRILL THIRD HOLE WITH AN *OPTIONAL* BLOCK DELETE  
:DRILL FOURTH HOLE WITH AN *OPTIONAL* BLOCK DELETE  
:CANCEL CANNED CYCLE, RAPID Z1. COOLANT OFF  
:RETURN Z TO MACHINE ZERO, SPINDLE OFF

:(COMMENTS IN PARENTHESIS ARE IGNORED BY CONTROL)  
:TOOL CHANGE TO TOOL #2, (NOTES TO OPERATOR)  
:ABS POSIT, WORK OFFSET#, RAPID TO POSIT, SPINDLE ON CW  
:TOOL LENGTH COMP #2, Z POSITION, COOLANT ON  
:SPOT DRILL CYCLE TO Z-.27 DEEP, DWELL .5 SECOND  
:SECOND HOLE , RAPID PLANE IS AT R.1  
:THIRD HOLE WITH AN *OPTIONAL* BLOCK DELETE  
:FOURTH HOLE WITH AN *OPTIONAL* BLOCK DELETE  
:CANCEL CANNED CYCLE, RAPID Z1., COOLANT OFF  
:RETURN Z TO MACHINE ZERO, SPINDLE OFF

:(COMMENTS IN PARENTHESIS ARE IGNORED BY CONTROL)  
:TOOL CHANGE TO TOOL #3 (NOTES TO OPERATOR)  
:ABS POSIT, WORK OFFSET#, RAPID TO POSIT, SPINDLE ON CW  
:TOOL LENGTH COMP #3, Z POSITION, COOLANT ON  
:FEED TO DEPTH  
:COMPENSATE CUTTER LEFT OF LINE  
:CUT A 2.0 IN. SQUARE CW WITH TOOL DIA. COMP **D03**

:G40 CANCELS COMP WHILE POSITIONING AWAY FROM PART  
:RAPID Z1., COOLANT OFF  
:RETURN Y AND Z TO MACHINE ZERO  
:TOOL CHANGE BACK TO TOOL #1  
:PROGRAM STOP AND REWIND TO BEGINNING  
:PROGRAM MUST BEGIN AND END WITH %

To change tools, all that is needed is an M06 even without a G28 in the previous line. A G28 can be specified to send all axes to machine home, or it can be defined to send a specific axis home with G28 G91 Z0 and/or Y0 and/or X0 to send just these axis specified to home position.



## OFTEN USED PREPARATORY (G) CODES:

- G00** Rapid traverse motion; Used for positioning of non-cutting moves.  
NOTE: Machine rapids at 710 inches per minute (IPM).
- G01** Linear interpolation motion; Used for actual machining and metal removal. Program with a feedrate in inches (or mm) per minute. Maximum feedrate of machine is 300 inches per minute.
- G02** Circular Interpolation - *Clockwise*
- G03** Circular Interpolation - *Counterclockwise*
- G28** Machine Home (*Rapid traverse*)
- G40** Cutter Compensation m CANCEL
- G41** Cutter Compensation LEFT
- G42** Cutter Compensation RIGHT
- G43** Tool LENGTH Compensation+
- G54** Work Coordinate #1 (Part zero)
- G80** Canned *Cycle* Cancel
- G81** Drill Canned *Cycle*
- G82** Spot Drill Canned *Cycle*
- G83** Peck Drill Canned *Cycle*
- G84** Tapping Canned *Cycle*
- G90** Absolute Programming Positioning
- G91** Incremental ProgrammingPositioning
- G98** *Initial Point Return*
- G99** *Reference Plane Return*



## MISCELLANEOUS M CODES:

- M00** The M00 code is used to stop a program. It also stops the spindle, turns off the coolant and stops interpretation lookahead processing. The CYCLE START button will continue program operation from the next block.
- M01** The M01 command is identical to M00 except that it only stops if optional stop, OPT STOP key, is turned on from the control panel. A cycle start will continue program operation from the next block.
- M03** Starts the spindle *Clockwise*. Must have a spindle speed defined.
- M04** Starts the spindle *Counterclockwise*. Must have a spindle speed defined.
- M05** Stops the spindle.
- M06** Tool change command along with a tool number will execute a tool change for that tool. This command will automatically stop the spindle, Z-axis will automatically move up to the machine zero position and the selected tool will be put in spindle. The coolant pump will turn off during a tool change.
- M08** Coolant ON command.
- M09** Coolant OFF command.
- M30** Program end and rewind to beginning of program.
- M97** Local subroutine call.
- M98** Subprogram call.
- M99** Subprogram return, or loop.

**NOTE:** Only one "M" code can be used per line. The "M" code will be the last command code to be performed in a line, regardless of where it's located in that line.



## PROGRAM FORMAT

The format defines the "language of the machine tool." A description of the particular words that a machine may accept, the order in which they must appear, the number of numeric digits associated with each word, the location of the decimal point, and the presence or absence of signs.

Program format, or program style is an important part of CNC machining. Each individual can format their programs many different ways. There are some program command formats that can be moved around, and some commands need to be a certain way, and there are some standard program rules that are just good to follow. The point is that a programmer needs to have an organized program format that's consistent and efficient so that any CNC machinist in your shop can understand it.

Some standard program rules to consider are:

Program X, Y and Z in alphabetical order on any block. The machine will read Y, Z or X in any order, but we want to be consistent. Write X first, Y second Z third.

You can put G and M codes anywhere on a line of code. But, in the beginning when N/C programming was being developed G codes had to be at the beginning, of a program line, and M codes had to be at the end. And this rule, a lot of people still follow and is a good standard to continue.

Some CNC machines allow you to write more than one M code per line of code and some won't. On the HAAS, only one M code may be programmed per block and all M codes are activated or cause an action to occur after everything else on the line has been executed. The list of M codes is on page 16.

The format defines a description of the line of code or the sequence of the particular commands in a program that a machine may accept and execute. Along with the order in which it must appear, the number of numeric digits associated with each word, the location of the decimal point, and the presence or absence of signs. The program format defines the "language of the machine tool."



## DEFINITIONS WITHIN THE FORMAT

- 1. CHARACTER :** A single alphanumeric character value or the "+" and "-" sign.
- 2. WORD :** A series of characters defining a single function such as, "X" *displacement* or an "F" *feedrate*. A letter is the first character of a word for each of the different functions. The balance of the word is made up of a plus (+) or minus (-) sign. A plus (+) is recognized if no sign is given in a word.
- 3. BLOCK :** Series of words defining a single instruction. An instruction may consist of a single linear motion, a circular motion or a canned cycle, plus additional information such as a feedrate or stop command.
- 4. POSITIVE SIGNS :** If the value following an address letter command such as A, B, C, I, J, K, R, U, V, W, X, Y, Z, is positive, *the plus sign need not be programmed in. If it has a minus (-) value it must be programmed in with a minus (-).*
- 5. LEADING ZERO'S :** If the digits proceeding a significant number are zero, they need not be programmed. *The HAAS control will automatically enter in the leading zero's.*  
**EXAMPLE:** G0 for G00 and M1 for M01,  
Trailing zeros must be programmed: M30 *not* M3, G70 *not* G7.
- 6. MODAL COMMANDS :** Codes that are active for more than the line in which they are issued are called **MODAL** commands. Rapid traverse, feedrate moves, and canned cycles are all examples of modal commands. A **NONMODAL** command which, once called, are effective only in the calling block, and are then immediately forgotten by the control.
- 7. PREPARATORY FUNCTIONS :** "G" codes use the information contained on the line to make the machine tool do specific operations, such as :
  - 1.) Move the tool at rapid traverse.
  - 2.) Move the tool at a feedrate along a straight line.
  - 3.) Move the tool along an arc at a feedrate in a clockwise direction.
  - 4.) Move the tool along an arc at a feedrate in a counterclockwise direction.
  - 5.) Move the tool thru a series of repetitive operations controlled by "fixed cycles" such as, spot drilling, drilling, boring, and tapping.
- 8. MISCELLANEOUS FUNCTIONS :** "M" codes are effective or cause an action to occur at the end of the block and only one M code is allowed in each block.
- 9. SEQUENCE NUMBERS :** Are code N1 thru N9999 that are only used to locate and identify a block or line and its relative position within a CNC program. A program can be input with or without SEQUENCE NUMBERS. The only function of SEQUENCE NUMBERS is to locate a certain block or line within a CNC program.

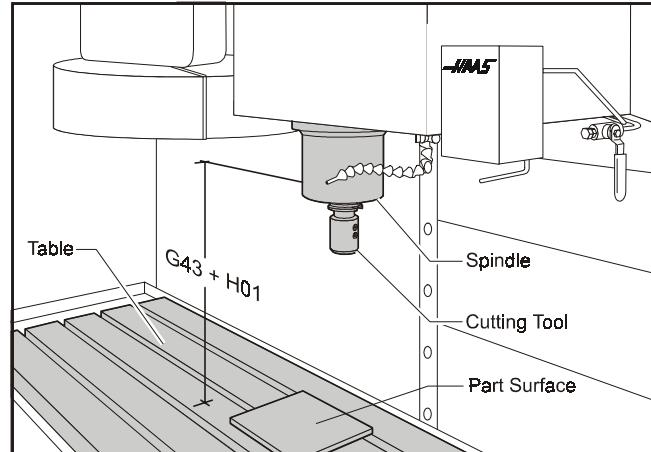


An example of a program's first three lines might look like this :

**T1 M06 ;  
G90 G54 G00 X0. Y0. S2500 M03;  
G43 H01 Z1. M08;**

or another format you might choose is:

**T1 M06 ;  
G00 G90 G54 X0. Y0.  
S2500 M03;  
G43 Z1. H01 M08;**



The tool startup lines with the necessary codes for each tool are listed above. These formats are a good example for startup lines that are entered in for each tool.

The FIRST line or block in a program should be a tool number (T1) and tool change (M06) command.

The SECOND line or block should contain an absolute (G90) or an incremental (G91 rarely used in a startup line) command, along with a work offset part zero command (G54 is the default) location for X and Y axis, a rapid (G00) command, a positioning X Y coordinate location, a spindle speed command (*Snnnn*), and a spindle ON clockwise command (M03), or you could have the spindle speed and clockwise command defined on a separate line.

The NEXT line or block contains a "Read tool length compensation" command (G43), a tool length offset register number (H01), a Z-axis positioning move (Z1.0), and an *optional* coolant ON command (M08).

Note: A tool length offset number should usually always remain numerically matched with the tool number. Setting 15 (the H & T code agreement) will ensure the tool number and the tool length offset number will match. (Example: T1 in line #1 should have H01 in line #3 or an alarm will occur if Setting 15 is ON.)



## MISCELLANEOUS FUNCTION (M CODES)

All **M** codes are activated or cause an action to occur at the end of a block, and only one **M** code is allowed per block in a program.

<b>M00</b>	Stop Program
<b>M01</b>	Optional Program Stop
<b>M02</b>	Program End
<b>M03</b>	Spindle Forward
<b>M04</b>	Spindle Reverse
<b>M05</b>	Spindle Stop
<b>M06</b>	Tool Change
<b>M08</b>	Coolant On
<b>M09</b>	Coolant Off
<b>M10</b>	Engage 4th Axis Brake
<b>M11</b>	Release 4th Axis Brake
<b>M12</b>	Engage 5th Axis Brake
<b>M13</b>	Release 5th Axis Brake
<b>M16</b>	Tool Change (same as M06)
<b>M19</b>	Orient Spindle
<b>M21-M24</b>	Optional Pulsed User M Function with Fin
<b>M30</b>	Program End and Rewind
<b>M31</b>	Chip Conveyor Forward
<b>M32</b>	Chip Conveyor Reverse
<b>M33</b>	Chip Conveyor Stop
<b>M34</b>	Increment Coolant Spigot Position
<b>M35</b>	Decrement Coolant Spigot Position
<b>M36</b>	Pallet Rotate
<b>M39</b>	Rotate Tool Turret
<b>M41</b>	Low Gear Override
<b>M42</b>	High Gear Override
<b>M51-M58</b>	Optional User M Activate
<b>M61-M68</b>	Optional User M Deactivate
<b>M75</b>	Set Measure Point
<b>M76</b>	Disable Displays
<b>M77</b>	Enable Displays
<b>M78</b>	Alarm if Skip Signal Found
<b>M79</b>	Alarm if Skip Signal Not Found



- M82** Tool Unclamp
- M86** Tool Clamp
- M88** Through the Spindle Coolant ON
- M89** Through the Spindle Coolant OFF
- M95** Sleep Mode
- M96** Jump if No Input
- M97** Local Sub-Program Call
- M98** Sub Program Call
- M99** Sub Program Return or Loop



## PREPARATORY FUNCTIONS (G CODES)

- 1) G Codes come in groups. Each group will have a specific group number.
- 2) A G code from the same group can be replaced by another code in the same group. By doing this the programmer establishes modes of operation. The universal rule here is, codes from the same group cannot be used more than once on the same line.
- 3) There are modal G codes which, once established, remain effective until replaced with another code from the same group.
- 4) There are non-modal G codes (Group 00) which, once called, are effective only in that block they are in, and then, do not affect any other blocks.

The rules above govern the use of all codes for programming the Haas (and other) controls. The concept of grouping codes and rules that apply will have to be remembered if we are to effectively program the machine tool.

The following is a summary of the G codes. A “\*” indicates the default within each group, if there is one:

## PREPARATORY FUNCTIONS (G CODES)

Code: Group:	Function:
<b>G00 *</b>	01 Rapid Motion
<b>G01</b>	01 Linear Interpolation Motion
<b>G02</b>	01 CW Interpolation Motion
<b>G03</b>	01 CCW Interpolation Motion
<b>G04</b>	00 Dwell
<b>G09</b>	00 Exact Stop
<b>G10</b>	00 Set Offsets
<b>G12</b>	00 CW Circular Pocket Milling (Yasnac)
<b>G13</b>	00 CCW Circular Pocket Milling (Yasnac)
<b>G17 *</b>	02 XY Circular Plane Selection
<b>G18</b>	02 ZX Circular Plane Selection
<b>G19</b>	02 YZ Circular Plane Selection
<b>G20</b>	06 Verify Inch (Setting 9 will need to be on INCH)
<b>G21</b>	06 Verify Metric (Setting 9 will need to be on METRIC)
<b>G28</b>	00 Return to Reference Point
<b>G29</b>	00 Set ReturnThrough Reference Point

(M codes continued next page)



<b>Code:</b>	<b>Group:</b>	<b>Function:</b>
<b>G31</b>	00	Feed Until Skip (optional)
<b>G35</b>	00	Automatic Tool Diameter Measurement (optional)
<b>G36</b>	00	Automatic Work Offset Measurement (optional)
<b>G37</b>	00	Automatic Tool Offset Measurement (optional)
<b>G40 *</b>	07	Cutter Comp Cancel
<b>G41</b>	07	2D Cutter Compensation Left
<b>G42</b>	07	2D Cutter Compensation Right
<b>G43</b>	08	Tool Length Compensation +
<b>G44</b>	08	Tool Length Compensation -
<b>G47</b>	00	Text Engraving
<b>G49 *</b>	08	G43/G44/G143 Cancel
<b>G50</b>	11	G51 Cancel
<b>G51</b>	11	Scaling (optional)
<b>G52</b>	12	Select Work Coordinate System (Yasnac)
<b>G52</b>	00	Set Local Shift Coordinate System (Fanuc)
<b>G52</b>	00	Set Local Shift Coordinate System (HAAS)
<b>G53</b>	00	Non-Modal Machine Coordinate Selection
<b>G54 *</b>	12	Select Work Coordinate System 1
<b>G55</b>	12	Select Work Coordinate System 2
<b>G56</b>	12	Select Work Coordinate System 3
<b>G57</b>	12	Select Work Coordinate System 4
<b>G58</b>	12	Select Work Coordinate System 5
<b>G59</b>	12	Select Work Coordinate System 6
<b>G60</b>	00	Unidirectional Positioning
<b>G61</b>	13	Exact Stop Modal
<b>G64 *</b>	13	G61 Cancel
<b>G65</b>	00	Macro Subroutine Call (optional)
<b>G68</b>	16	Rotation (optional)
<b>G69</b>	16	G68 Cancel (optional)
<b>G70</b>	00	Bolt Hole Circle (Yasnac)
<b>G71</b>	00	Bolt Hole Arc (Yasnac)
<b>G72</b>	00	Bolt Holes Along an Angle (Yasnac)
<b>G73</b>	09	High Speed Peck Drill Canned Cycle
<b>G74</b>	09	Reverse Tap Canned Cycle
<b>G76</b>	09	Fine Boring Canned Cycle
<b>G77</b>	09	Back Bore Canned Cycle

(M codes continued next page)



<b>Code:</b>	<b>Group:</b>	<b>Function:</b>
<b>G80</b> *	09	Canned Cycle Cancel
<b>G81</b>	09	Drill Canned Cycle
<b>G82</b>	09	Spot Drill Canned Cycle
<b>G83</b>	09	Regular Peck Drill Canned Cycle
<b>G84</b>	09	Tapping Canned Cycle
<b>G85</b>	09	Bore in/Bore out Canned Cycle
<b>G86</b>	09	Bore in/Stop/Rapid out Canned Cycle
<b>G87</b>	09	Bore in/Manual Retract Canned Cycle
<b>G88</b>	09	Bore/Dwell/Manual Retract Canned Cycle
<b>G89</b>	09	Bore/Dwell/Bore out Canned Cycle
<b>G90</b> *	03	Absolute
<b>G91</b>	03	Incremental
<b>G92</b>	00	Set Work Coordinates
<b>G93</b>	05	Inverse Time Feed Mode
<b>G94</b> *	05	Feed Per Minute Mode
<b>G98</b> *	10	Initial Point Return
<b>G99</b>	10	R Plane Return
<b>G100</b>	00	Cancel Mirror Image
<b>G101</b>	00	Enable Mirror Image
<b>G102</b>	00	Programmable Output to RS-232
<b>G103</b>	00	Limit Block Buffering (Block Lookahead)
<b>G107</b>	00	Cylindrical Mapping
<b>G110</b>	12	Select Work Coordinate System 7
<b>G111</b>	12	Select Work Coordinate System 8
<b>G112</b>	12	Select Work Coordinate System 9
<b>G113</b>	12	Select Work Coordinate System 10
<b>G114</b>	12	Select Work Coordinate System 11
<b>G115</b>	12	Select Work Coordinate System 12
<b>G116</b>	12	Select Work Coordinate System 13
<b>G117</b>	12	Select Work Coordinate System 14
<b>G118</b>	12	Select Work Coordinate System 15
<b>G119</b>	12	Select Work Coordinate System 16
<b>G120</b>	12	Select Work Coordinate System 17
<b>G121</b>	12	Select Work Coordinate System 18
<b>G122</b>	12	Select Work Coordinate System 19
<b>G123</b>	12	Select Work Coordinate System 20

(M codes continued next page)



<b><u>Code:</u></b>	<b><u>Group:</u></b>	<b><u>Function:</u></b>
<b>G124</b>	12	Select Work Coordinate System 21
<b>G125</b>	12	Select Work Coordinate System 22
<b>G126</b>	12	Select Work Coordinate System 23
<b>G127</b>	12	Select Work Coordinate System 24
<b>G128</b>	12	Select Work Coordinate System 25
<b>G129</b>	12	Select Work Coordinate System 26
<b>G136</b>	00	Automatic Work Offset Center Measurement
<b>G141</b>	07	3D+ Cutter Compensation
<b>G143</b>	08	5 Axis Tool Length Compensation (optional)
<b>G150</b>	00	General Purpose Pocket Milling
<b>G174</b>	00	Special Purpose Rigid Tapping CCW
<b>G184</b>	00	Special Purpose Rigid Tapping CW
<b>G187</b>	00	Accuracy Control for High Speed Machining

Each G code defined in this control is part of a group of G codes. The Group 0 codes are non-modal; that is, they specify a function applicable to that block only and do not affect other blocks. The other groups are modal and the specification of one code in the group cancels the previous code applicable from that group. A modal G code applies to all subsequent blocks so those blocks do not need to re-specify the same G code.

There is also one case where the Group 01 G codes will cancel the Group 9 (canned cycles) codes. If a canned cycle is active (G73 through G89), the use of G00 or G01 will cancel the canned cycle.



## MACHINE DEFAULTS

A default is an automatic function of the machine tool control. After powering up the machine, the control will recognize the default "G" code values. The machine will go to the part zero that was entered in for G54 if no other work coordinate code was specified in the actual program, because the machine automatically recognizes the G54 column upon start-up. That is a default. The defaults for the Haas mill are indicated by an asterisk ( \* ) in the "Preparatory Functions (G Codes)" list of this workbook.

The control automatically recognizes these G codes when your HAAS mill is powered up:

- G00** Rapid Traverse
- G17** X,Y Circular Plane Selection
- G20** Verify Inch (Setting 9 will need to be on INCH)
- G40** Cutter Compensation Cancel
- G49** Tool length Compensation Cancel
- G50** G51 Cancel
- G54** Work Coordinate Zero #1 (1 of 26 available)
- G64** Exact Stop Cancel
- G69** G68 Cancel (optional)
- G80** Canned Cycle Cancel
- G90** Absolute Programming
- G94** Inverse Time Feed Deactivate
- G98** Initial Point Return

There is no default feedrate (F code), but once an F code is programmed, it will apply until another feedrate is entered or the machine is turned off.



## CANNED CYCLES

A canned cycle is used to simplify programming of a part. Canned cycles are defined for most common Z-axis repetitive operations such as drilling, tapping, and boring. Once selected, a canned cycle is active until canceled with the G80 code. There are six operations involved in every canned cycle:

- 1) Positioning of X and Y axes (optional A, rotary axis).
- 2) Rapid traverse to the reference plane.
- 3) Drilling, boring, or tapping action.
- 4) Operation at the bottom of the hole.
- 5) Retraction to the reference plane.
- 6) Rapid traverse to the initial starting point.

Canned cycle will only be executed in the Z-axis when positioning to a new X and/or Y axis position after a canned cycle is selected. G98 and G99 are modal commands which change the way a canned cycle operates. G98 (default) will cause Z-axis to return to the initial starting point, and G99 will return Z-axis to the R (reference) plane after a cycle has been executed and positions to a new location to execute another cycle.

Listed below are the canned cycles that are available. To see the start of the detailed descriptions listed for each of these canned cycles, see page 69.

- G73** High Speed Peck Drill Canned Cycle  
**G74** Reverse Tap Canned Cycle  
**G76** Fine Boring Canned Cycle  
**G77** Back Bore Canned Cycle  
**G80\*** Canned Cycle Cancel  
**G81** Drill Canned Cycle  
**G82** Spot Drill Canned Cycle  
**G83** Peck Drill Canned Cycle  
**G84** Tapping Canned Cycle  
**G85** Boring Canned Cycle  
**G86** Bore/Stop Canned Cycle  
**G87** Bore/Manual Retract Canned Cycle  
**G88** Bore/Dwell/Manual Retract Canned Cycle  
**G89** Bore/Dwell/Canned Cycle



## ALPHABETICAL ADDRESS CODES

The following is a list of the Address Codes used in programming the Mill.

### A FOURTH AXIS ROTARY MOTION

The A address character is used to specify motion for the optional fourth, A, axis. It specifies an angle in degrees for the rotary axis. It is always followed by a signed number and up to three fractional decimal positions. If no decimal point is entered, the last digit is assumed to be 1/1000 degrees.

### B FIFTH AXIS ROTARY MOTION

The B address character is used to specify motion for the optional fifth, B, axis. It specifies an angle in degrees or the rotary axis. It is always followed by a signed number and up to three fractional decimal positions. If no decimal point is entered, the last digit is assumed to be 1/1000 degrees.

### C AUXILIARY EXTERNAL ROTARY AXIS

The C address character is used to specify motion for the optional external sixth, C, axis. It specifies an angle in degrees for the rotary axis. It is always followed by a signed number and up to three fractional decimal positions. If no decimal point is entered, the last digit is assumed to be 1/1000 degrees.

### D TOOL DIAMETER OFFSET SELECTION

The D address character is used to select the tool diameter or radius used for cutter compensation. The number following must be between 0 and 100. The Dn selects that number offset register, that is in the offset display, which contains the tool diameter/radius offset amount when using cutter compensation (G41 G42). D00 will cancel cutter compensation so that the tool size is zero and it will cancel any previously defined Dn.

### E CONTOURING ACCURACY

The E address character is used, with G187, to select the accuracy required when cutting a corner during high speed machining operations. The range of values possible is 0.0001 to 0.25 for the E code. Refer to the "Contouring Accuracy" section of your machine manual for more information.

### F FEED RATE

The F address character is used to select the feed rate applied to any interpolation functions, including pocket milling and canned cycles. It is either in inches per minute with four fractional positions or mm per minute with three fractional positions.

### G PREPARATORY FUNTIONS (G codes)

The G address character is used to specify the type of operation to occur in the block containing the G code. The G is followed by a two or three digit number between 0



and 187. Each G code defined in this control is part of a group of G codes. The Group 0 codes are non-modal; that is, they specify a function applicable to this block only and do not effect other blocks. The other groups are modal and the specification of one code in the group cancels the previous code applicable from that group. A modal G code applies to all subsequent blocks so those blocks do not need to re-specify the same G code. More than one G code can be placed in a block in order to specify all of the setup conditions for an operation.

## **H TOOL LENGTH OFFSET SELECTION**

The H address character is used to select the tool length offset entry from the offsets memory. The H is followed by a two digit number between 0 and 100. H0 will clear any tool length offset and Hn will use the tool length entered in on n from the Offset display. You must select either G43 or G44 to activate a tool length (H) offsets. The G49 command is the default condition and this command will clear any tool length offsets. Reset and M30 will also cancel tool length offsets.

## **I CIRCULAR INTERPOLATION OR CANNED CYCLE DATA**

The I address character is used to specify data for either canned cycles or circular motions. It is defined in inches with four fractional positions or mm with three fractional positions.

## **J CIRCULAR INTERPOLATION OR CANNED CYCLE DATA**

The J address character is used to specify data for either canned cycles or circular motions. It is defined in inches with four fractional positions or mm with three fractional positions.

## **K CIRCULAR INTERPOLATION OR CANNED CYCLE DATA**

The K address character is used to specify data for either canned cycles or circular motions. It is defined in inches with four fractional positions or mm with three fractional positions.

## **L LOOP COUNT TO REPEAT COMMAND CYCLES**

The L address character is used to specify a repeat count for some canned cycles and auxiliary functions. It is followed by a number between 0 and 32767.

## **M M CODE MISCELLANEOUS FUNCTIONS**

The M address character is used to specify an M code. These codes are used to control miscellaneous machine functions. Note that only one M code is allowed per block in a CNC program and all M codes are performed secondary in a block.

## **N NUMBER OF BLOCK**

The N address character is entirely optional. It can be used to identify or number each block of a program. It is followed by a number between 0 and 99999. The M97 function needs to reference an N line number.



## O PROGRAM NUMBER

The O address character is used to identify a program. It is followed by a number between 0 and 99999. A program saved in memory always has a Onnnnn identification in the first block. Altering the Onnnnn in the first block causes the program to be renumbered. If you put a (program name) between parenthesis in the first three lines in your program, that program name will also be seen in your program list. You can have up to 200 program numbers in your List of Programs.

## P DELAY OF TIME / PROGRAM OR SEQUENCE NUMBER

The P address character is used to enter either a time in seconds or a program number for a subroutine call. If it is used as a time (for a dwell time with G04 or in some canned cycles) or as a program sequence number reference (for a M97), or a program reference number (for a M98) the value is a positive number without decimal point up to 9999 when used with an M98. If it is used as a dwell time, it may be a positive decimal with fraction between 0.001 and 1000.0.

## Q CANNED CYCLE OPTIONAL DATA

The Q address character is used in canned cycles and is always a positive number in inches between 0.001 and 100.0.

## R CIRCULAR INTERPOLATION OR CANNED CYCLE DATA

The R address character is used in canned cycles or circular interpolation. It's either in inches with four fractional positions or mm with three fractional positions. It is followed by number in inches or metric. It's usually used to define the reference plane for canned cycles.

## S SPINDLE SPEED COMMAND

The S address character is used to specify the spindle speed in conjunction with M41 and M42. The S is followed by an unsigned number between 1 - 99999. The S command does not turn the spindle on or off; it only sets the desired speed. If a gear change is required in order to set the commanded speed, this command will cause a gear change to occur even if the spindle is stopped. If spindle is running, a gear change operation will occur and the spindle will start running at the new speed.

## T TOOL SELECTION CODE

The T address character is used to select the tool for the next tool change. The number following must be a positive number between 1 and (20) the number in Parameter 65. It does not cause the tool change operation to occur. The Tn may be placed in the same block that starts tool change (M6 or M16) or in any previous block.



## **U AUXILIARY EXTERNAL LINEAR AXIS**

The U address character is used to specify motion for the optional external linear, U-axis. It specifies a position of motion in inches. It is always followed by a signed number and up to four fractional decimal positions. If no decimal point is entered, the last digit is assumed to be 1/10000 inches. The smallest magnitude is 0.0001 inches, the most negative value is -8380.0000 inches, and the largest number is 8380.0000 inches.

## **V AUXILIARY EXTERNAL LINEAR AXIS**

The V address character is used to specify motion for the optional external linear, V-axis. It specifies a position of motion in inches. It is always followed by a signed number and up to four fractional decimal positions. If no decimal point is entered, the last digit is assumed to be 1/10000 inches.

## **W AUXILIARY EXTERNAL LINEAR AXIS**

The W address character is used to specify motion for the optional external linear, W-axis. It specifies a position of motion in inches. It is always followed by a signed number and up to four fractional decimal positions. If no decimal point is entered, the last digit is assumed to be 1/10000 inches.

## **X LINEAR X-AXIS MOTION**

The X address character is used to specify motion for the X-axis. It specifies a position or distance along the X-axis. It is either in inches with four fractional positions or mm with three fractional positions. It is followed by a signed number in inches or metric. If no decimal point is entered, the last digit is assumed to be 1/10000 inches or 1/1000 mm.

## **Y LINEAR Y-AXIS MOTION**

The Y address character is used to specify motion for the Y-axis. It specifies a position or distance along the Y-axis. It is either in inches with four fractional positions or mm with three fractional positions. It is followed by a signed number in inches or metric. If no decimal point is entered, the last digit is assumed to be 1/10000 inches or 1/1000 mm.

## **Z LINEAR Z-AXIS MOTION**

The Z address character is used to specify motion for the Z-axis. It specifies a position or distance along the Z-axis. It is either in inches with four fractional positions or mm with three fractional positions. It is followed by a signed number in inches or metric. If no decimal point is entered, the last digit is assumed to be 1/10000 inches or 1/1000 mm.

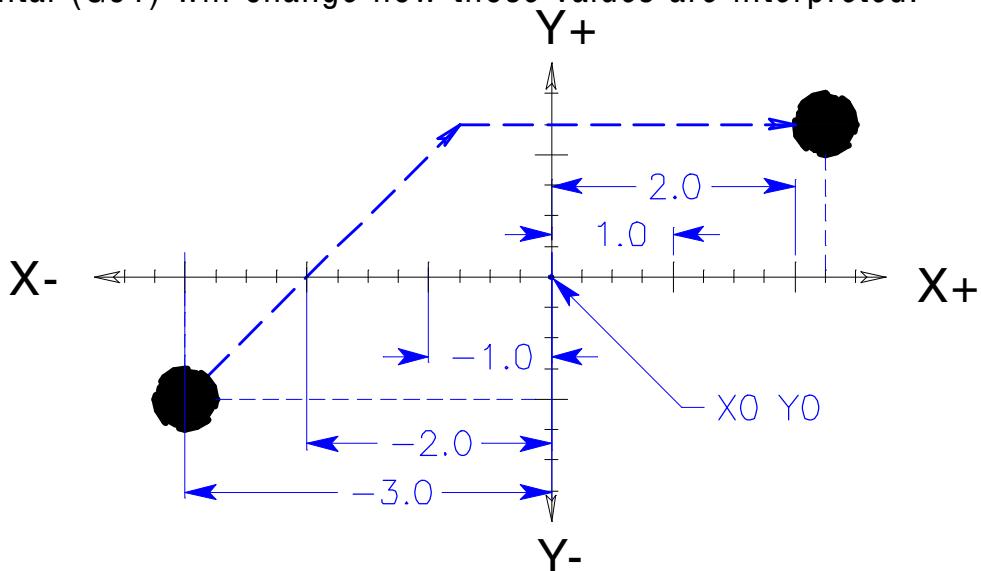


## RAPID POSITION COMMANDS

### G00 Rapid Positioning Motion

- X Optional X-axis motion command
- Y Optional Y-axis motion command
- Z Optional Z-axis motion command
- A Optional A axis motion command

This G code is for rapid traverse of the three or four axes of the machine. This G00 code is modal and causes all the following blocks to be in rapid (710 in./min.) motion until another Group 01 code is specified. Generally, rapid motions "will not" be in a straight line. All the axes specified are moved at the maximum speed and will not necessarily complete each axis move at the same time. So you need to be careful of any obstructions to avoid with this type of rapid move. The tool will first move from the current position in a straight line along a 45 degree angle to an intermediate location when one of these axes has completed its move. Then the machine will position parallel to the X or Y axis to complete the move to the final location. If the Z axis is also in the program move, it will operate in the same manner along with the X and Y axes. Only the axes specified are moved and the commands for absolute (G90) or incremental (G91) will change how those values are interpreted.



**ABSOLUTE POSITIONING** - G90 G00 X2.25 Y1.25

OR

**INCREMENTAL POSITIONING** - G91 G00 X5.25 Y2.25

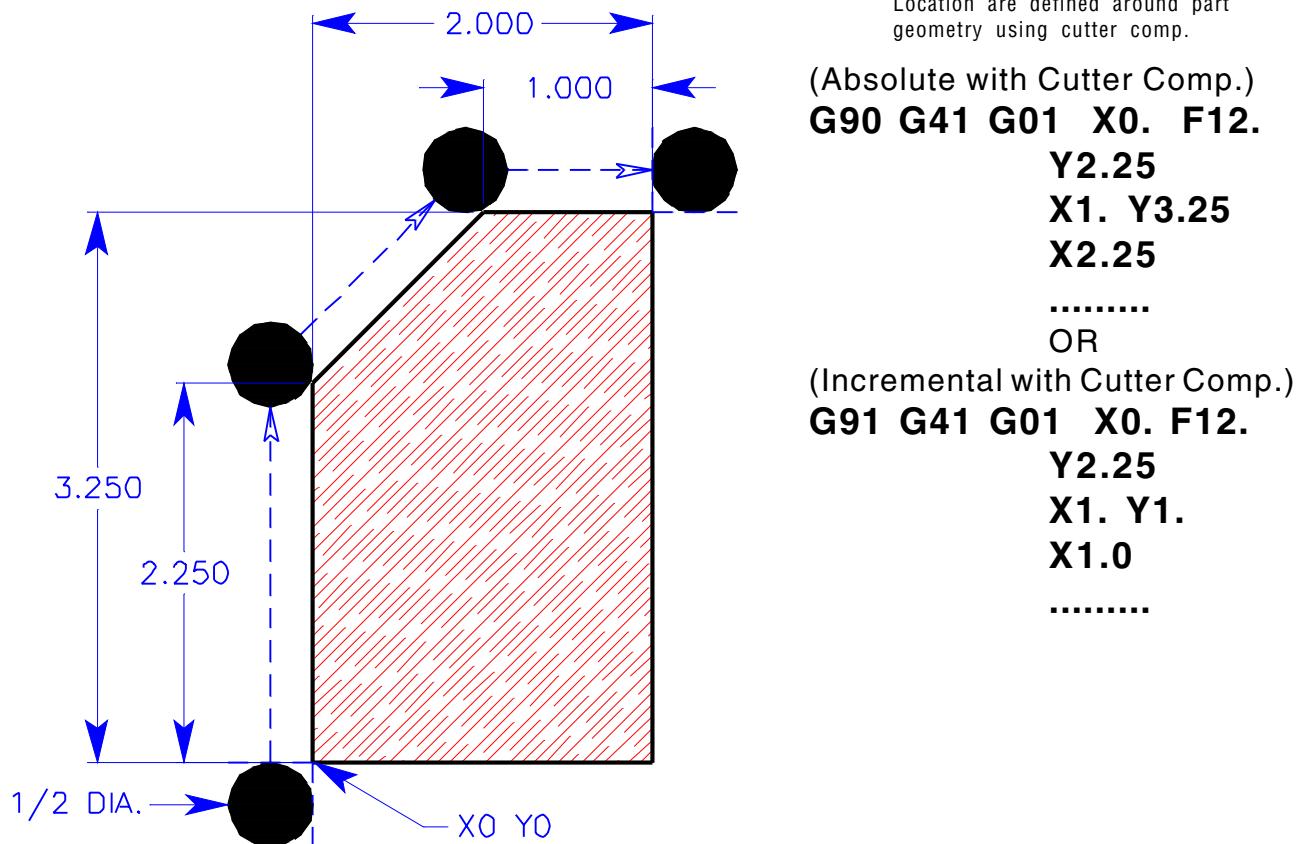


## INTERPOLATION COMMANDS

### G01 Linear Interpolation Motion

- X Optional X-axis motion command
- Y Optional Y-axis motion command
- Z Optional Z-axis motion command
- A Optional A axis motion command
- F Feed rate in inches (mm) per minute

This G code provides for straight line (linear) motion from point to point. Motion can occur in 1, 2 or 3 axes. All axes start and finish motion at the same time. The rotary axis may also provide motion around an axis or center point. The speed of all axes are controlled by a feedrate specified along with axis moves. Rotary axis feedrate is dependent on rotary axis diameter setting (Setting 34) and will provide a controlled motion. The F command is modal and may be specified in a previous block. Only the axes specified are moved in either absolute (G90) or incremental (G91) modal commands which change how values are interpreted.





## CIRCULAR INTERPOLATION COMMANDS

### G02 CW CIRCULAR INTERPOLATION MOTION

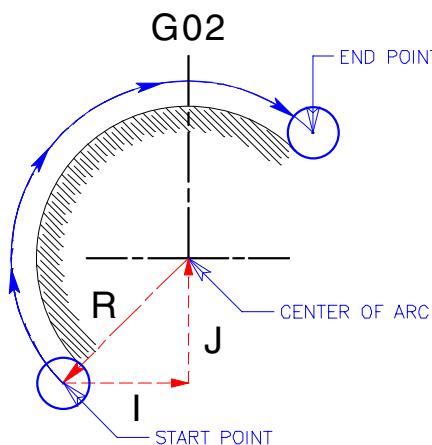
- X End point X-axis Motion command
- Y End point Y-axis Motion command
- Z End point Z-axis Motion command
- A End point A-axis Motion command
- I Distance from start point to arc center in X-axis (optional)
- J Distance from start point to arc center in Y-axis (optional)
- K Distance from start point to arc center in Z-axis (optional)
- R Radius of the arc to be machined (If I, J, K are not used)
- F Feed rate in inches (or mm) per minute

### G03 CCW CIRCULAR INTERPOLATION MOTION

G03 will generate a counterclockwise circular motion, but is otherwise defined the same way as G02.

These G codes are used to specify a clockwise or counterclockwise motion of two, of the linear axes. Circular motion is possible in two of the three axes in either the X, Y, and/or Z axes as selected by G17, G18, and G19. The X, Y, and Z in a circular command (G02 or G03) is used to define the end point of that motion in either absolute (G90) or incremental (G91) motion. If any of the axes, X, Y, or Z for the selected plane is not specified, the endpoint location of the arc will then be recognized the same as the starting point of the arc, for that axis. There are two basic command formats for defining circular interpolation, depending on whether the IJK method or the R method is used to define the arc center.

Circular interpolation commands are used to move a tool along a circular arc to the commanded end position. Five pieces of information are required for executing a circular *interpolation command*:



1. Plane selection
2. Arc start position coordinates
3. Rotation direction
4. Arc end position coordinates
5. Arc center coordinates or arc radius.



## The Five pieces of information for executing a circular interpolation command.

Item	Command	Remark
① Plane selection command Plane selection command Plane selection command	G17 G18 G19	Arc parallel to XY-plane Arc parallel to ZX-plane Arc parallel to YZ-plane
② Arc start position coordinates	X,Y,Z	Coordinates of the start position
③ Rotation direction	G02 G03	Clockwise direction Counterclockwise direction
④ Arc end position (G90) Absolute <i>or</i> Arc end position (G91) Incremental	X,Y,Z	Coordinates of the end position on the work coordinate system Distance from start position to end position in X, Y, and Z axes, respectively
⑤ IJK method (arc center coordinate) <i>or</i> R method (arc radius)	I,J,K R	Distance from start position to arc center in X, Y, and Z axes, respectively Arc radius value

There are two ways to specify the center of the circular arc that you're rotating around: the first uses I, J, or K to specify the distance from the starting point to the center of the arc; the second uses R to specify the radius of the arc. These two ways will be further described below:

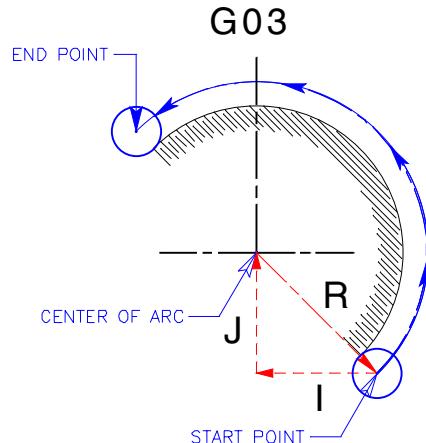
The "I", "J" and "K" values are "INCREMENTAL" distances from the tool's starting point (START POINT) on the arc to the ARC CENTER.

"I" = incremental distance from Start Point to arc center in the "X axis".

"J" = incremental distance from Start Point to arc center in the "Y axis".

"K" = incremental distance from Start Point to arc center in the "Z axis".

When R is used, a complete 360 degree arc is not possible. X, Y, or Z is required to specify an endpoint different from the starting point. R is the distance from the starting point to the center of the circle. With a positive R, the control will generate a circular path of 180 degrees or less, but to generate a circular path of over 180 degrees, specify a *negative* R.





## G02 CIRCULAR INTERPOLATION CLOCKWISE

Contouring a part profile using a circular interpolation command, uses the axis information contained in a block, to move the tool in a CLOCKWISE circular move, up to 360 degrees.

The velocity at which the tool is moving is controlled by the feedrate ( F ) command.

All circles are defined and machined by programming in three pieces of information to the control, they are :

START POINT of the arc

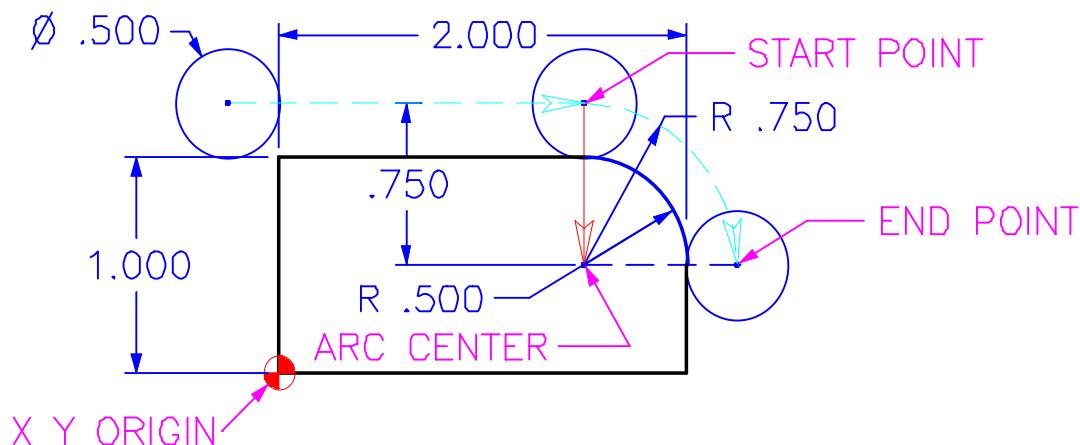
END POINT of the arc

ARC CENTER incremental distance from the start point to the arc center.

The START POINT is the point defined prior to the G02 line, usually by a G01 linear positioning move.

The END POINT is defined by the "X" and "Y" coordinates contained in the G02 line, when in the *G17- XY PLANE*.

The ARC CENTER is defined in the G02 line with the "I" for the Xaxis and "J" for the Y axis incremental distance and direction to the arc center, when in the *G17- XY PLANE*. Or by the "R" Radius value.



N5 G01 Y1.250 F12.

N6 X1.500 (to start point)

N7 G02 X2.250 Y.500 (I0. J-.750 or R.750)

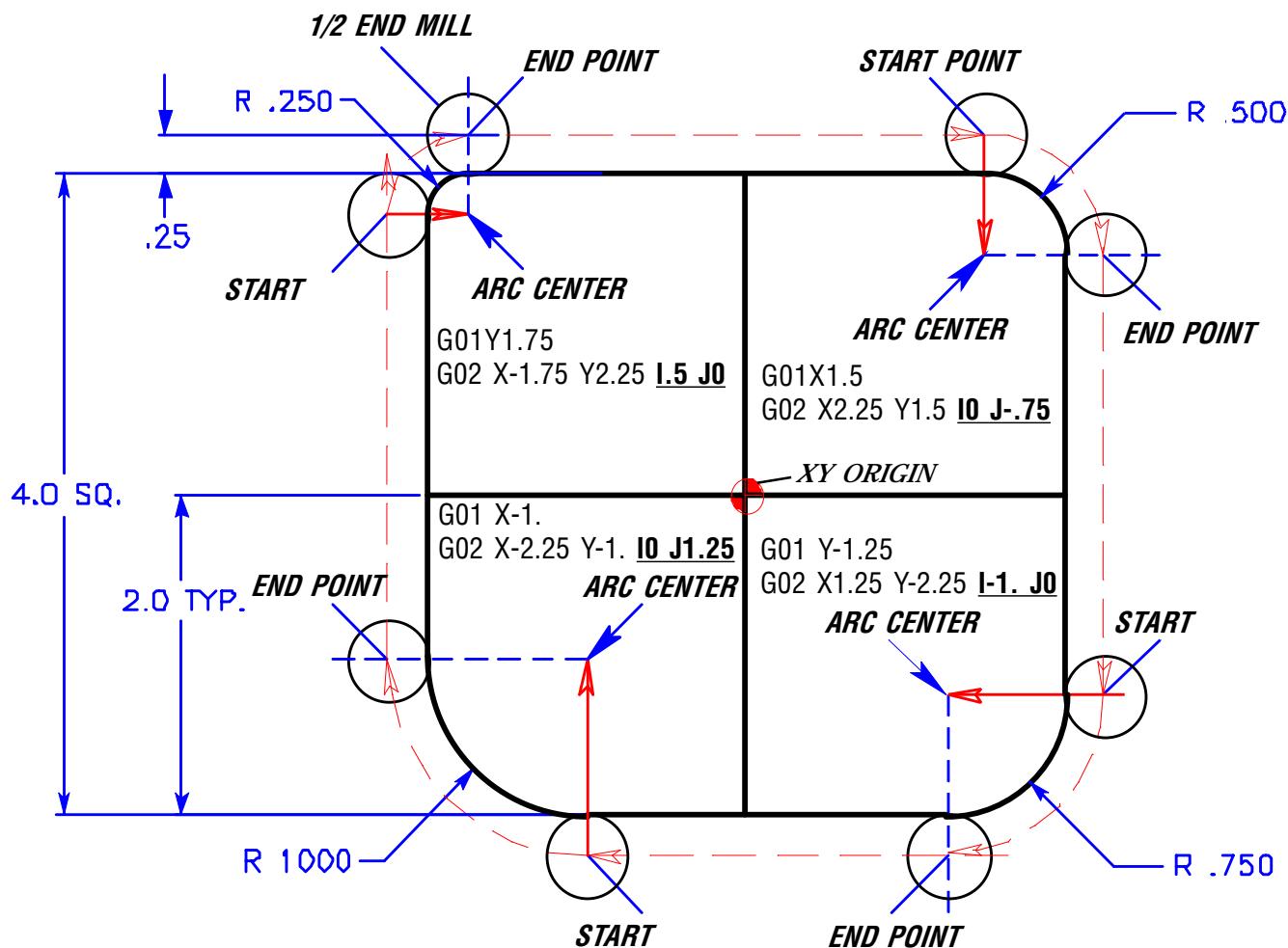


## **G02 CIRCULAR INTERPOLATION CLOCKWISE USING "I", "J" AND "K"**

The "I", "J" and "K" values are INCREMENTAL distances from where the tool starts cutting the arc (START POINT) to the ARC CENTER.

When I, J, or K are used to specify the center of the arc, R is not used. Only the I, J, or K specific to the selected plane (IJ for G17, IK for G18, JK for G19) are allowed. If only one of the I, J, K is specified, the others are assumed to be zero. The I, J, or K is the distance from the starting point to the center of the circle. Use of I, J, or K is the only way to cut a complete 360 degree arc; in this case, the starting point is the same as the ending point and no X, Y, or Z is needed. To cut a complete circle of 360 degrees ( $360^\circ$ ), you do not need to specify an ending point X, Y, or Z; just program I, J, or K to define the center of the circle.

**NOTE:** Example of circular moves are not using cutter compensation, so the circular moves are defined from the center of the cutter around arc. For 90 degree corners or fillets the "I" and "J" values can be defined like you see below.





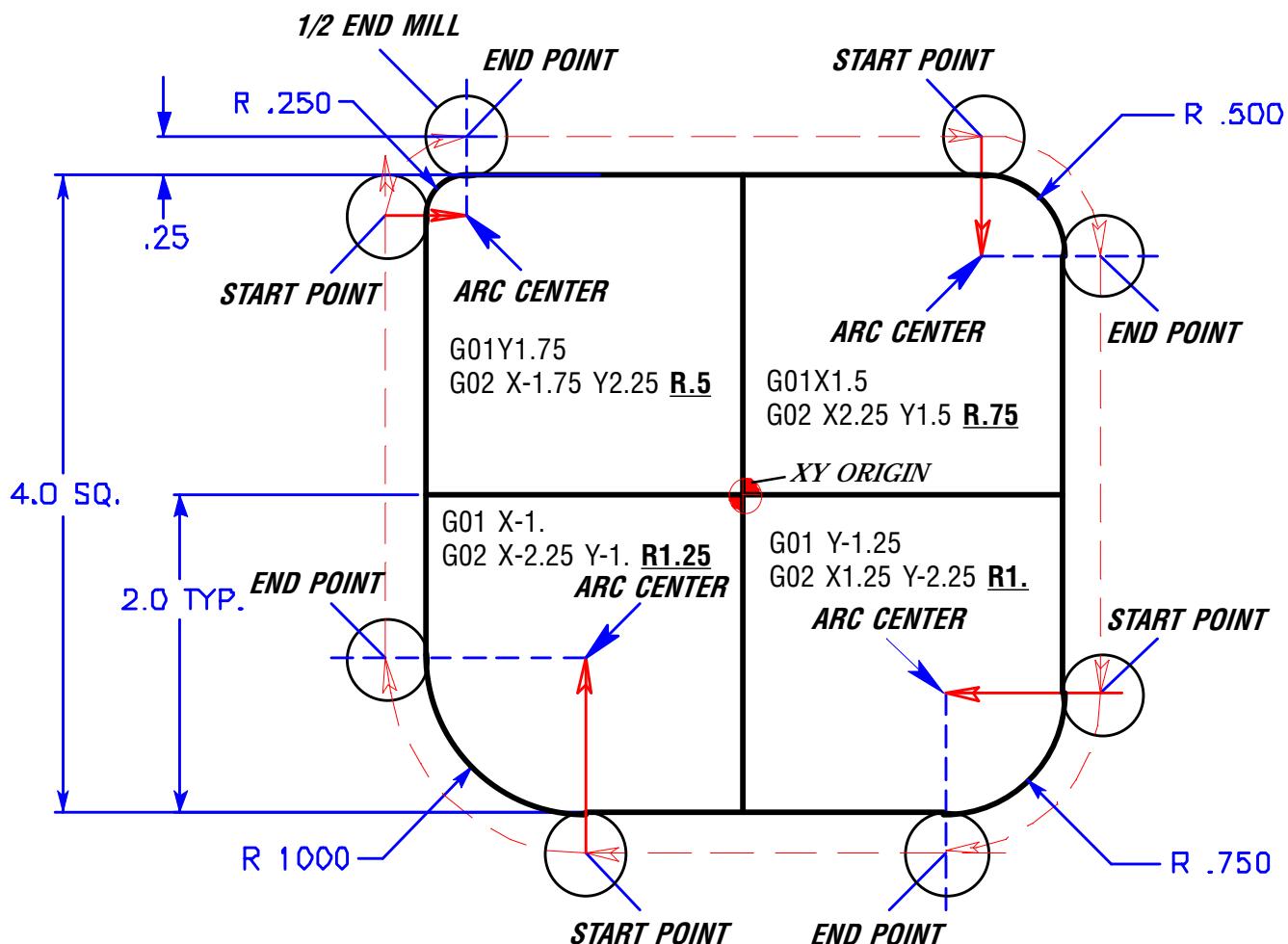
## G02 CW CIRCULAR MOTION USING "R"

The letter address "R" can be substituted for the letter addresses "I", "J" and/or "K" for various corners or fillets and the "R" value is more easily defined.

Just as "I" "J" and "K", are used to locate the ARC CENTER in relation to the START POINT, "R" does the same function with less input.

When R is used to specify the center of the circle, a complete 360 degree arc is not possible. When defining an arc less than 360 degrees using the R command, the X, Y, or Z needs to specify an endpoint different from the starting point. And R defines the distance from the starting point to the center of the circle. With a positive R, the control will generate a radius of 180 degrees or less; to generate a radius of over 180 degrees, you need to specify an R command with a negative value.

NOTE: Example of circular moves are not using cutter compensation, so the circular moves are defined from the center of the cutter around arc. For 90 degree corners or fillets the "R" values can be defined like you see below.





## G03 CIRCULAR INTERPOLATION COUNTER-CLOCKWISE

Contouring a part profile using a circular interpolation command, uses the axis information contained in a block, to move the tool in a COUNTER-CLOCKWISE circular move, up to 360 degrees.

The velocity at which the tool is moving is controlled by the feedrate ( F ) command.

All circles are defined and machined by programming in three pieces of information to the control, they are :

START POINT of the arc

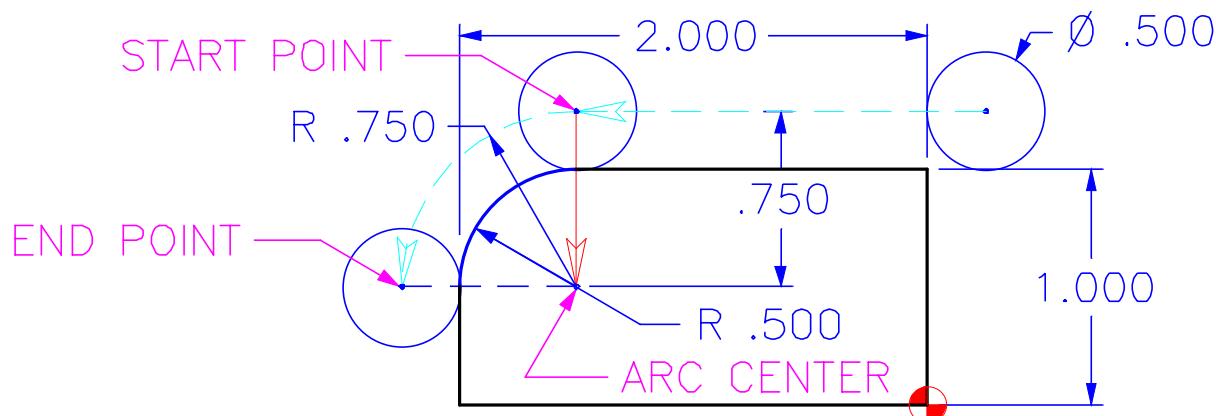
END POINT of the arc

ARC CENTER incremental distance from the start point to the arc center.

The START POINT is the point defined prior to the G02 line, usually by a G01 linear positioning move.

The END POINT is defined by the "X" and "Y" coordinates contained in the G02 line, when in the *G17- XY PLANE*.

The ARC CENTER is defined in the G02 line with the "I" for the Xaxis and "J" for the Y axis incremental distance and direction to the arc center, when in the *G17- XY PLANE*. Or by the "R" Radius value.



N5 G01 Y1.250 F12.

N6 X-1.500 (to start point)

N7 G03 X-2.250 Y.500 (I0. J-.750 or R.750)

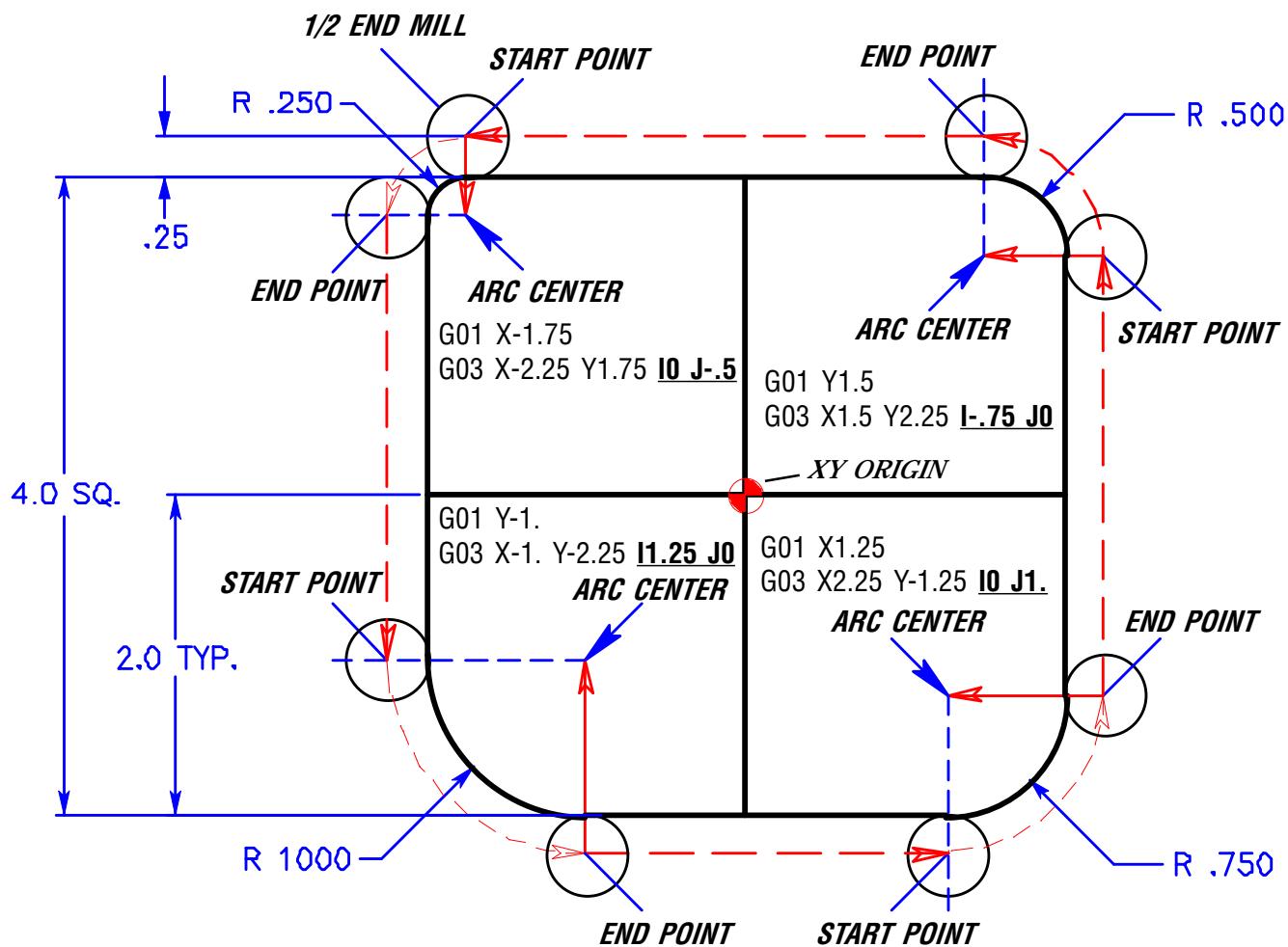


## G03 CIRCULAR INTERPOLATION COUNTER-CLOCKWISE USING "I", "J" AND "K"

The "I", "J" and "K" values are INCREMENTAL distances from where the tool starts cutting the arc (START POINT) to the ARC CENTER.

When I, J, or K are used to specify the center of the arc, R is not used. Only the I, J, or K specific to the selected plane (IJ for G17, IK for G18, JK for G19) are allowed. If only one of the I, J, K is specified, the others are assumed to be zero. The I, J, or K is the distance from the starting point to the center of the circle. Use of I, J, or K is the only way to cut a complete 360 degree arc; in this case, the starting point is the same as the ending point and no X, Y, or Z is needed. To cut a complete circle of 360 degrees ( $360^\circ$ ), you do not need to specify an ending point X, Y, or Z; just program I, J, or K to define the center of the circle.

NOTE: Example of circular moves are not using cutter compensation, so the circular moves are defined from the center of the cutter around arc. For 90 degree corners or fillets the "I" and "J" values can be defined like you see below.





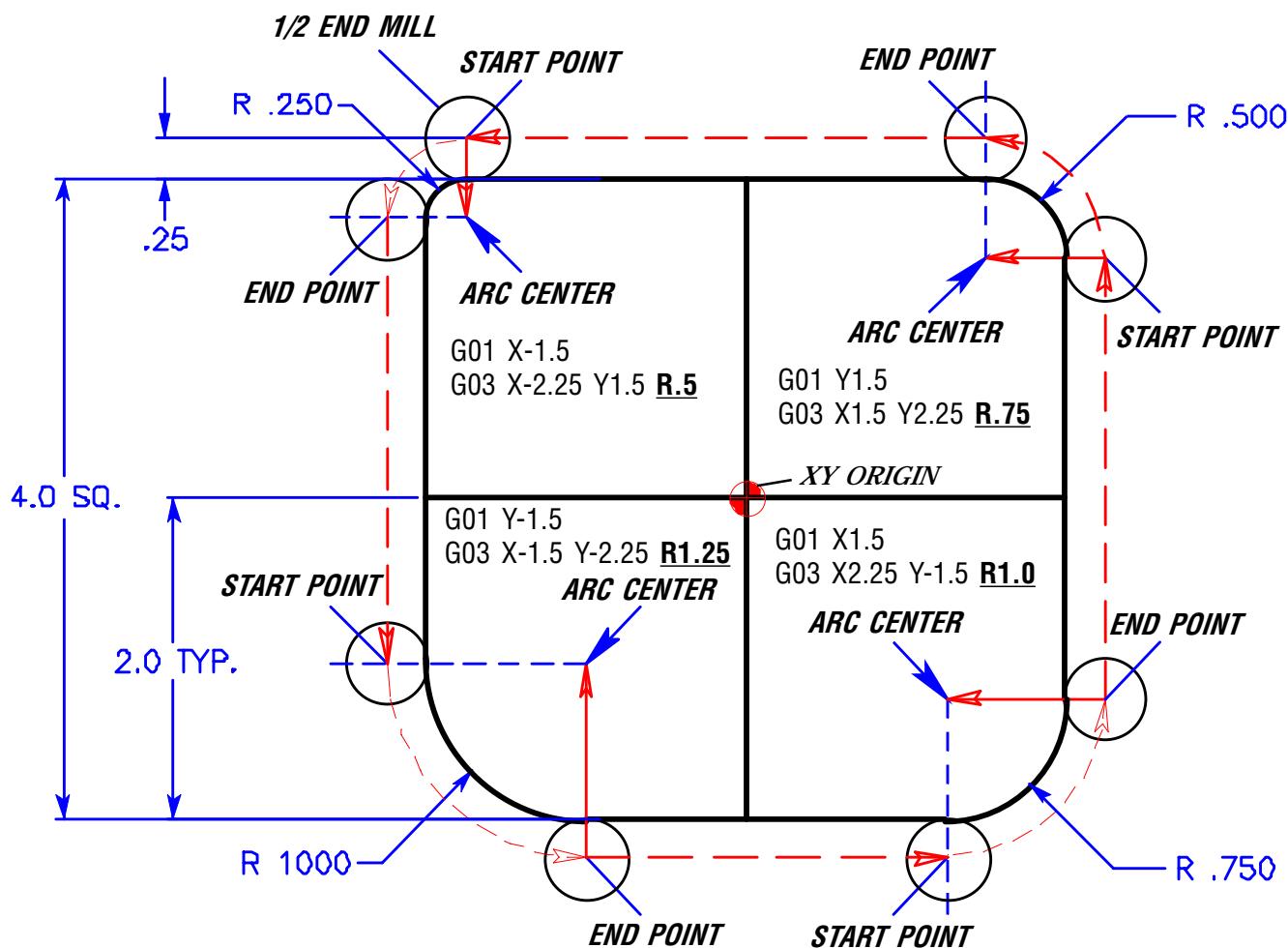
## G03 CCW CIRCULAR MOTION USING "R"

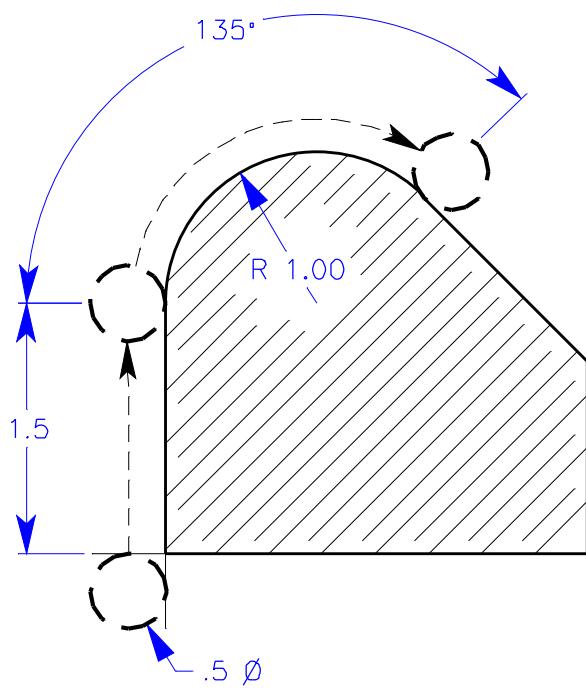
The letter address "R" can be substituted for the letter addresses "I", "J" and/or "K" for various corners or fillets and the "R" value is more easily defined.

Just as "I" "J" and "K", are used to locate the ARC CENTER in relation to the START POINT, "R" does the same function with less input.

When R is used to specify the center of the circle, a complete 360 degree arc is not possible. When defining an arc less than 360 degrees using the R command, the X, Y, or Z needs to specify an endpoint different from the starting point. And R defines the distance from the starting point to the center of the circle. With a positive R, the control will generate a radius of 180 degrees or less; to generate a radius of over 180 degrees, you need to specify an R command with a negative value.

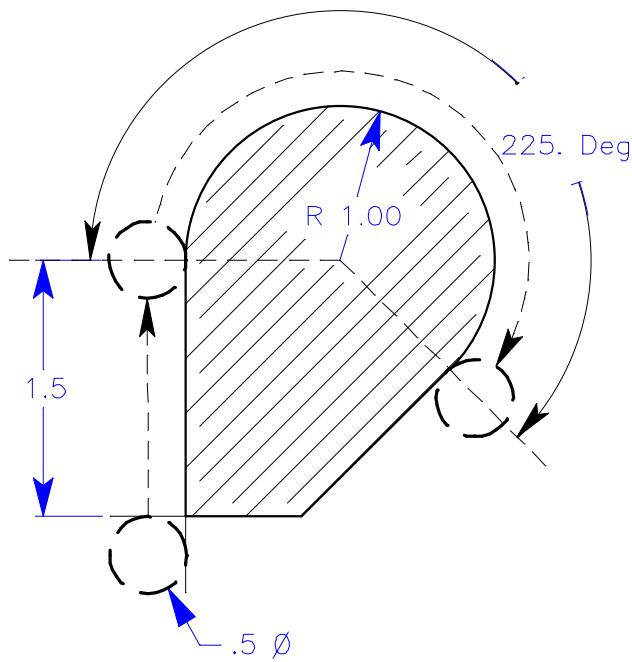
NOTE: Example of circular moves are not using cutter compensation, so the circular moves are defined from the center of the cutter around arc. For 90 degree corners or fillets the "R" values can be defined like you see below.





The following line will cut an arc less than 180 degrees ( $180^\circ$ ) using a positive R value:

**G90 G54 G00 X-0.25 Y-0.25  
G01 Y1.5 F12.  
G02 X1.884 Y2.384 R1.25**



To generate an arc of over 180 degrees you need to specify a negative R value( $180^\circ$ ) using an R minus:

**G90 G54 G00 X-0.25 Y-0.25  
G01 Y1.5 F12.  
G02 X1.884 Y0.616 R-1.25**

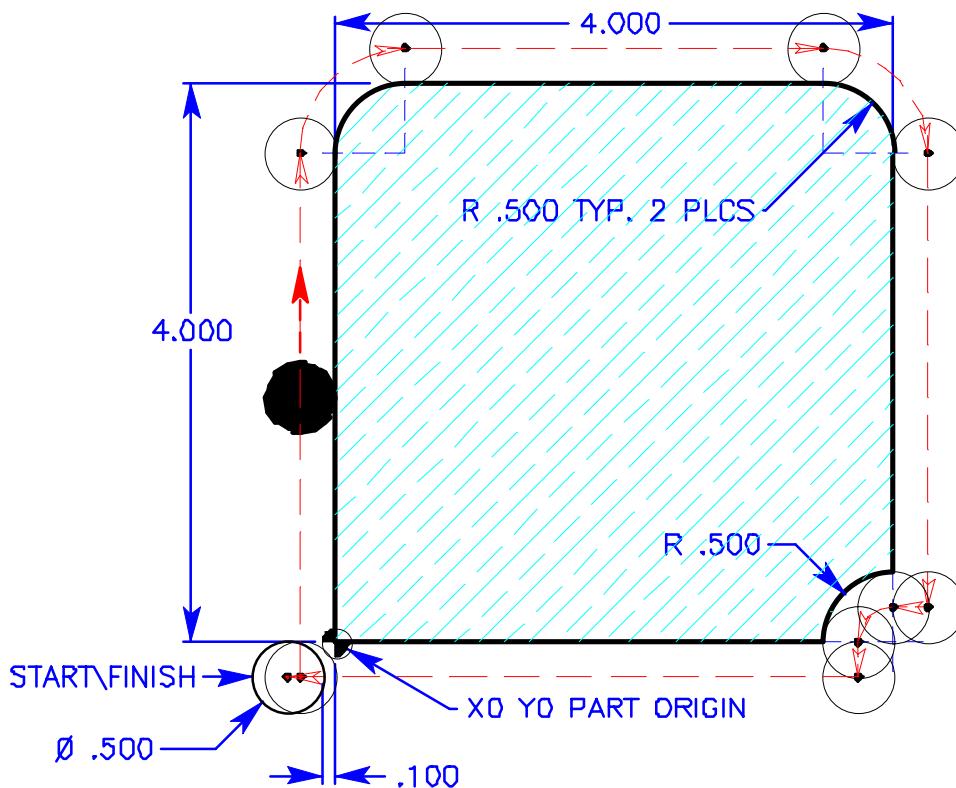
A complete 360 degree arc using an R command is not possible. To do a 360 degree arc in a G02 or G03 you need to use I and/or J to define the center of a circle for a complete 360 degree arc.

### **G03 CCW Circular Interpolation Motion**

G03 will generate counterclockwise circular motion but is otherwise the same as G02.



## INTERPOLATION EXERCISE



Cutter will be a .500 dia. end mill. Start contour from the lower left corner of part and mill around outside of part .625 deep. When defining a circular move (G02 or G03) you can use either an IJK or an R command but not both. *Mill around outside of part with tool by defining the very center of the cutter to position around part.*

### O00010 (INTERPOLATION EXERCISE)

T1 M06 (1/2 DIA. 2 FLT. END MILL)

G90 G54 G00 X\_\_\_\_\_ Y\_\_\_\_\_

S1400 M03

G43 H01 Z0.1 M08

G01 Z\_\_\_\_\_ F50. (feed fast to depth non-cutting move)

X\_\_\_\_\_ F10. (feed onto left side of part programming to the center of tool)

Y\_\_\_\_\_ (feed up to the upper left radius and continue around part to the end)

G0\_\_ X\_\_\_\_\_ Y\_\_\_\_\_ (either I\_\_\_\_\_ J\_\_\_\_\_ or R\_\_\_\_\_ not both)

G0\_\_ X\_\_\_\_\_

G0\_\_ X\_\_\_\_\_ Y\_\_\_\_\_ (either I\_\_\_\_\_ J\_\_\_\_\_ or R\_\_\_\_\_ not both)

G0\_\_ Y\_\_\_\_\_

X\_\_\_\_\_

G0\_\_ X\_\_\_\_\_ Y\_\_\_\_\_ (either I\_\_\_\_\_ J\_\_\_\_\_ or R\_\_\_\_\_ not both)

G0\_\_ Y\_\_\_\_\_

X\_\_\_\_\_

G00 Z1. M09

G28 G91 Z0. M05

M30



## PROGRAM START-UP LINES

**T1 M06**

**T1** - Selects tool number one to be loaded into the spindle.  
**M06** -Activates the tool change sequence.

**G90 G54 G00 X\_\_\_\_ y\_\_\_\_**

**G90** - Activates control to be in ABSOLUTE.  
**G54** - Selects work coordinate system No. 1  
**G00** - Preparatory function for RAPID.  
**X\_\_\_\_**-Axis move to initial "X" position.  
**Y\_\_\_\_**-Axis move to initial "Y" position.

**S1200 M03**

**S1200** - Informs the control that 1200 has been selected as the RPM for this tool.  
**M03**- Turns the spindle "ON" in a clockwise direction at a speed of 1200 RPM.

**G43 H01 Z0.1 M08**

**G43** - Recognizes the tool length offset value stored in the "Hnn" code offset display register in the offset LENGTH Geometry display.  
**H01** - Defines to the control which offset register the tool offset value is stored in.  
**Z0.1** - Informs the control to move from full spindle retract to this "Z" value and apply the tool length offset.  
**M08** - Turns the coolant "ON".



## PROGRAM ENDING LINES

**G00 Z1. M09**

**G00** - Preparatory function for RAPID.  
**Z1.** - Retracts tool to 1.0 above "PART ZERO" in preparation for full spindle retract.  
**M09** - Turns the coolant "off".

**G28 G91 Y0. Z0. M05**

**G28** - This preparatory function causes a return to machine zero of all axes. If an X, Y and Z are specified, only those axis specified will be moved.  
**G91** - Preparatory function for INCREMENTAL mode, which is required for rapid move to the G28 reference point. G28 also cancels tool length compensation.

---

----- USING G53 TO SEND MACHINE HOME INSTEAD OF G28 G91 -----

**G53 G49 Y0. Z0. M05**

**G53** - This preparatory function temporarily cancels work offset and positions the machine axes from machine home. It is non-modal; so the next block will revert back to the previously selected work offset.  
**G49** - This preparatory function cancels tool length compensation.

---

**Y0.** - This will select the Y axis to position the table toward you to its machine zero.  
**Z0.** - This will insure a full retract, of Z axis in the "UP" direction to machine zero  
**M05** - This command will turn off spindle.

**M30**

**M30** - Signals END OF PROGRAM AND REWIND, to the control. The program will reset to the first block of the program and stop. It also stops the spindle, turns off the coolant and cancels tool length offsets. It also makes the control ready for the next cycle.



## MISCELLANEOUS G CODES

### G04 Dwell

P The dwell time in seconds or milliseconds

G04 is used to cause a delay or dwell in the program. The block containing G04 will delay for the time specified in the P code. When programmed on a line following some motion such as G00, G01, G02 and G03 all motion will be stopped for the amount of time specified in the P command in seconds. If the P has no fraction part, the delay is in milliseconds (0.001 seconds); otherwise the delay is in seconds. The slide motion is stopped, but the spindle will continue to rotate at the requested RPM, and the coolant stays on.

**G04 P\_\_\_\_\_**

Minimum value - P.001 of a second

Maximum value - P1000.0 seconds



## CIRCULAR POCKET MILLING

There are two G codes G12 and G13 that will provide for pocket milling of a circular shape. They are different only in which direction of rotation is used. G12 and G13 are non-modal.

### G12 Circular Pocket Milling Clockwise

- X Position in X axis to center of circular pocket
  - Y Position in Y axis to center of circular pocket
  - Z Z depth of cut, *or it's the increment depth of cuts when used with G91*
  - I Radius Of First Circle (Or it's the finish radius if K is not used)
  - K Radius Of Finished Circle (If specified)
  - Q Radius cut increment step-over of the spiral out (Q is used with K only)
  - L Loop count for repeating incremental depth of cuts (L is used with G91)
  - D\* Cutter Comp. Offset Number (Enter cutter size in offset display register)
  - F Feed Rate in inches (mm) per minute
- This G12 Code implies the use of G42

### G13 Circular Pocket Milling Counterclockwise

This G13 Code implies the use of G41 and will also be machining in a counterclockwise direction, but is otherwise the same as G12.

*The tool must be positioned at the center of the circular pocket either in a previous block or in this command using an X and Y position. The cuts are performed entirely with circular motions of varying radiiuses.*

\*In order to get the exact programmed circle diameter, the control uses the selected D code tool size. If this compensation is not desired, program D0.

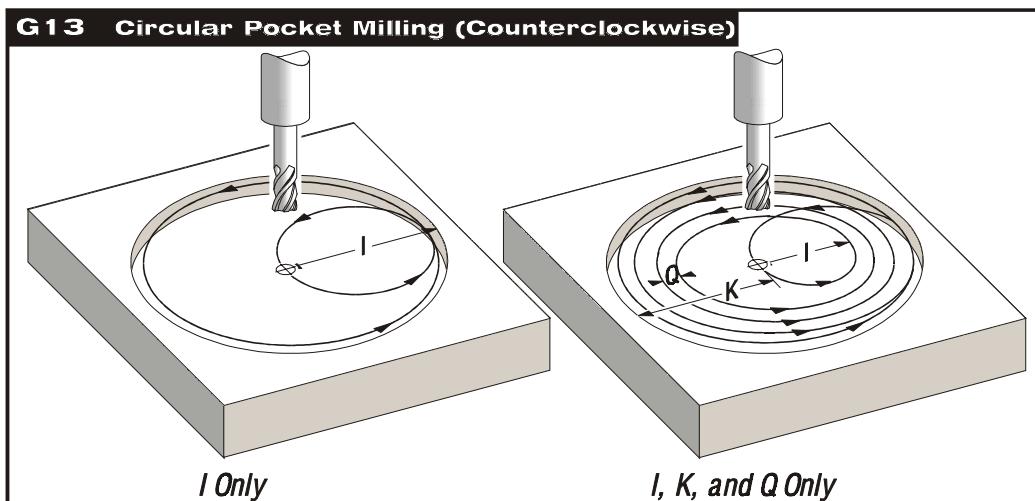
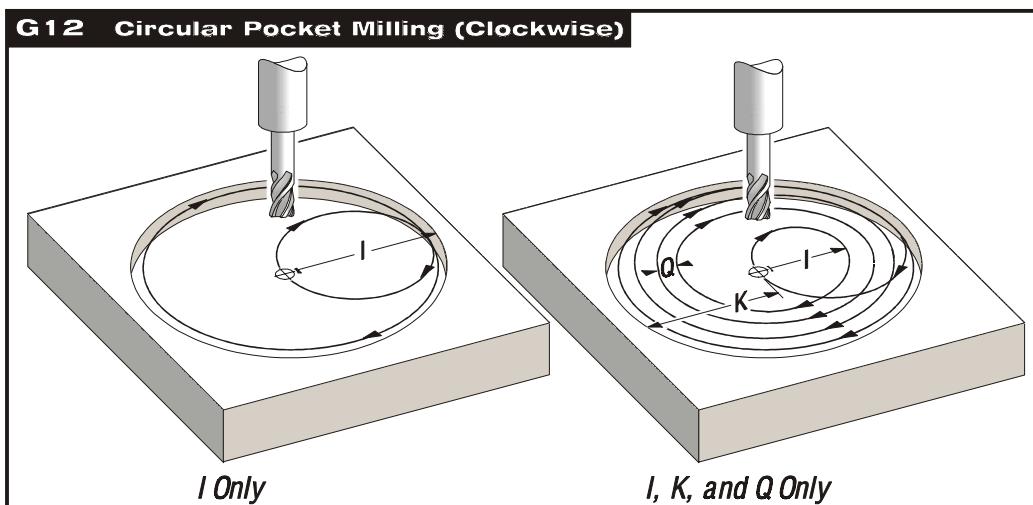
The G12 Code implies the use of G42 cutter compensation right.

The G13 Code implies the use of G41 cutter compensation left.

To remove all the material within the circle use an I and Q value less than the tool diameter and a K value equal to the circle radius. G12 and G13 belongs to Group zero and thus is non-modal.

If no K is specified, the center roughing passes of this command are removed completely and only one finish pass of the circular pocket is performed.

If G91 (incremental) is specified and an L count is included, the Z increment is repeated L times at the F feed rate command.

**EXAMPLE: G13 ONE PASS "I" ONLY****O01041****N1 (D01 DIA. OFFSET IS .500)****N2 T1 M06 (1/2 DIA. 2 FLT END MILL)****N3 G90 G54 G00 X2.5 Y2.5** (position to X Ycenter of circular pocket)**N4 S3600 M03****N5 G43 H01 Z0.1 M08****N6 G13 Z-0.5 I0.5 D01 F15.** (1.0 Dia. x .5 deep circular pocket 1 pass)**N7 G00 Z1. M09****N8 G28 G91 Y0 Z0****N9 M30**

**EXAMPLE: G13 MULTIPLE PASSES I, K & Q****001042****N1 (D01 DIA. OFFSET IS .500)****N2 T2 M06 (1/2 DIA. 2 FLT END MILL)****N3 G90 G54 G00 X2.5 Y2.5** (X Y position to center location of circular pocket)**N4 S3600 M03****N5 G43 H02 Z0.1 M08****N6 G13 Z-0.5 I0.3 K1.5 Q0.3 D02 F15.** (3.0 Dia. x .5 deep circular pocket)**N7 G00 Z1. M09****N8 G28 G91 Y0 Z0****N9 M30****EXAMPLE: G13 MULTIPLE PASSES "I" "K" & "Q" AND INCREMENTAL G91****001042****N1 (D01 DIA. OFFSET IS .500)****N2 T3 M06 (1/2 DIA. 2 FLT END MILL)****N3 G90 G54 G00 X2.5 Y2.5** (X Y position to center location of circular pocket)**N4 S3600 M03****N5 G43 H03 Z0.1 M08****N6 G01 Z0. F30.** (Move down to the start of part surface to begin the increment down)**N7 G13 G91 Z-0.375 I.25 K2. Q .3 D03 L4 F15.** (4.0 Dia. x 1.5 deep circular pocket)**N8 G00 G90 Z1. M09****N9 G28 G91 Y0 Z0****N10 M30**

The above program uses G91 and a "L" count of four. This cycle is multiplied by the "L" command and will do it a total of four times at the Z depth increment of .375 to a total depth of 1.5 in. The G91 and "L" count can also be used for **G12** and **G13 "I"** only line.

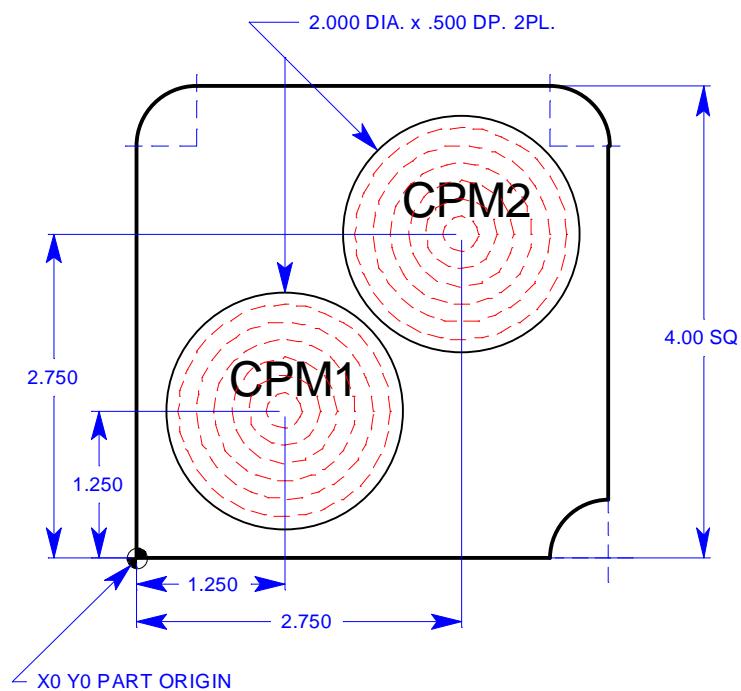


## CIRCULAR POCKET MILLING EXERCISE

**G12** Circular Pocket Milling CW  
or

**G13** Circular Pocket Milling CCW

- X** - Position to center of pocket
- Y** - Position to center of pocket
- Z** - Depth of cut or increment down
- I** - Radius of First Circle (Or the finish radius If K is not used).
- K** - Radius of Finished Circle (If specified).
- Q** - Radius step over Increment ( Must be used with K).
- D** - Cutter Comp. Number (Enter cutter size into offset display register number).
- L** - Loop Count for repeating deeper cuts.
- F** - Feedrate in inch (mm) per min.



**Tool 2 is a 5/8 Dia. 2 flute end mill with a spindle speed of 1400 RPM at 8.7 feed.**

**Circular Pocket Mill CPM 1** to a depth of .500 spiraling out to a rough size using I.25, K.990 and Q.2 for roughing out pocket. Then .Pocket mill another command to run on CPM 1 using I1.0 only as a circular pocket finish pass.

**Circular Pocket Mill CPM 2** which is a 2.0 Dia. x .500 dp. pocket incrementally stepping down .25 depth using an L2 count and an incremental G91 command to do 2 passes on a circular pocket using I.3, K1.0 and Q.35 for machining out pocket.

### O00020 (CIRCULAR POCKET MILLING EXERCISE)

T\_\_\_\_ M\_\_\_\_ (T2 IS A 5/8 DIA. 2FLT. CENTER CUTTING END MILL)

G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_ (Position X Ycenter of CPM1)

S\_\_\_\_\_ M\_\_\_\_\_

G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_ M\_\_\_\_ (Rapid to .1 above part)

G\_\_\_\_ Z\_\_\_\_ I\_\_\_\_ K\_\_\_\_ Q\_\_\_\_ D\_\_\_\_ F\_\_\_\_ (CPM1 with I,K& Q)

G\_\_\_\_ Z\_\_\_\_ I\_\_\_\_ D\_\_\_\_ F\_\_\_\_ (I only for a finish pass of CPM1)

G\_\_\_\_ Z\_\_\_\_ (Z rapid to clearance position above part)

X\_\_\_\_\_ Y\_\_\_\_\_ (X Y position to center of CPM2)

**G01 Z0. F20.** (Feed down to the start point to start incremental depth down)

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ I\_\_\_\_ K\_\_\_\_ Q\_\_\_\_ D\_\_\_\_ L\_\_\_\_ F\_\_\_\_ ( I, K, Q & G91)

G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_ (Rapid 1.0 above part, turn coolant off)

**G28 G91 Z0 M05**

**M30**

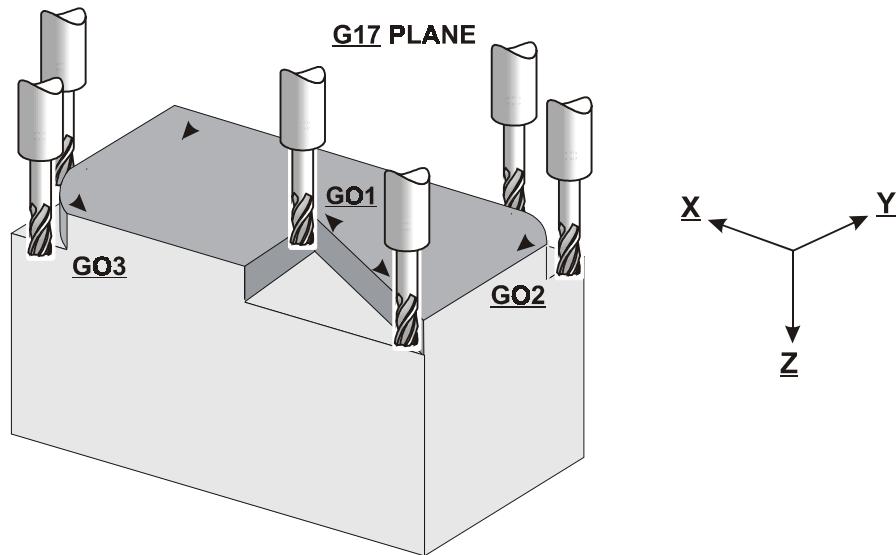


## CIRCULAR PLANE SELECTION

The plane used for circular motions must be comprised of two of the axes, X, Y, or Z. The plane selection is modal and stays in effect for all subsequent circular interpolation moves until you command another plane selection code. There are three G codes used to select the circular plane; G17 for the *XY plane*, G18 for the *XZ plane*, and G19 for the *YZ plane*.

### G17 XY CIRCULAR PLANE SELECTION

The G17 code is used to select the XY plane for circular motion. In this plane, circular motion is defined as clockwise for the operator looking down onto the X-Y table from above



### O01044 (G17 XY ARC PLANE EXAMPLE)

N1 T2 M06 (1/2 DIA. 2 FLT. END MILL)

N2 G90 G54 G00 X4. Y3.25 S2600 M03 (X Y start point of arc)

N3 G43 H02 Z0.1 M08

N4 G01 Z-.5 F50.

N5 G17 G02 X5.25 Y2. R1.25 F10. (G17 circular motion X Y plane)

N6 G00 Z1.

N7 X-.25 Y1. (G17 is default when you power up machine)

N8 G01 Z-.5 F50.

N9 G17 G03 X1. Y-.25 R1.25 F10. (G17 circular motion X Y plane)

N10 G00 Z1. M09

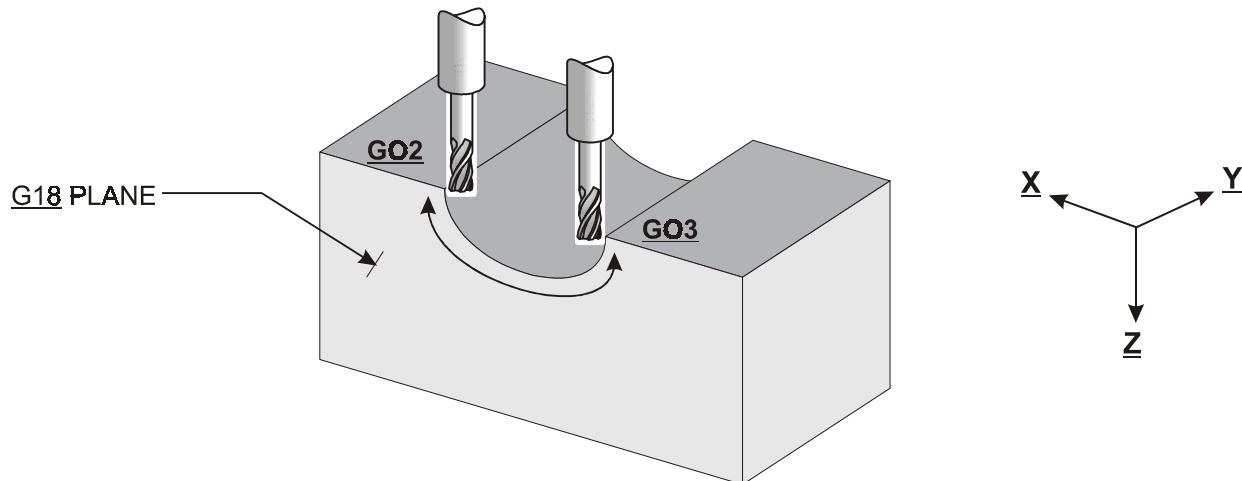
N11 G53 G49 Y0 Z0 M05

N12 M30



## G18 ZX CIRCULAR PLANE SELECTION

The G18 code is used to select the ZX plane for circular motion. In the X-Z plane (G18), circular motion is defined as clockwise for the operator looking from the rear of the machine out toward the control panel.



### 001045 (MAIN PROGRAM G18 ZX ARC PLANE EXAMPLE)

N1 T3 M06 (1/2 DIA. 2 FLT. BALL END MILL)

N2 G90 G54 G00 X1.5 Y0. S2600 M03 (X Y start point of arc)

N3 G43 H03 Z0.1 M08

N4 G01 Z0. F20.

N5 M97 P100 L80 (Local sub-program call done 80 times with L80)

N6 G00 Z1. M09

N7 G53 G49 Y0. Z0. M05

N8 M30

(Local sub-program N100 called up by an M97 on line N5)

N100 G91 G01 Y-0.01

N101 G90

N102 G18 G03 X3. Z0. R.75 F12. (G18 circular motion Z X plane)

N103 G91 G01 Y-0.01

N104 G90

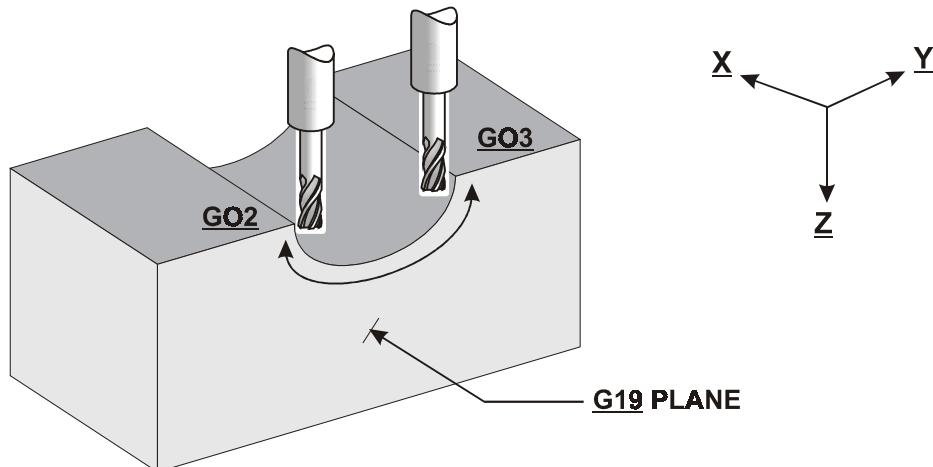
N105 G18 G02 X1.5 Z0. R.75 F12. (G18 circular motion Z X plane)

N106 M99 (An M99 will cause the program to jump back to the next line after the M97 sub program call in the main program.)



## G19 YZ CIRCULAR PLANE SELECTION

The G19 code is used to select the YZ plane for circular motion. In the Y-Z plane (G19), circular motion is defined as clockwise for the operator looking across the table from the side of the machine the control panel is mounted.



### O01045 (MAIN PROGRAM G19 YZ ARC PLANE EXAMPLE)

**N1 T4 M06 (1/2 DIA. 2FLT BALL END MILL)**

**N2 G90 G54 G00 X.0 Y3. S2600 M03** (X Y start point of arc)

**N3 43 H04 Z0.1 M08**

**N4 G01 Z0. F20.**

**N5 M97 P100 L80** (Local sub-program call done 80 times with L80)

**N6 G00 Z1. M09**

**N7 G53 G49 Y0. Z0. M05**

**N8 M30**

(Local sub-program N100 called up by an M97 on line N5)

**N100 G91 G01 X-0.01**

**N101 G90**

**N102 G19 G03 Y1.5 Z0. R.75 F12.** (G19 circular motion Y Z plane)

**N103 G91 G01 X-0.01**

**N104 G90**

**N105 G19 G02 Y3. Z0. R.75 F12.** (G19 circular motion Y Z plane)

**N106 M99** (An M99 will cause the program to jump back to the next line after the M97 sub program call in the main program.)



The default plane selection when the machine is powered on is G17 for the X-Y plane. This means that circular motion in the plane of the X-Y table axis may be programmed without first selecting G17.

A helical motion is possible with G02 or G03 by programming the linear axis which is not in the selected plane. This third axis will be interpolated along the specified axis in a linear manner while the other two axes will be moved in the circular motion. The speed of each axis will be controlled so that the helical rate matches the programmed feed rate.

If cutter radius compensation is selected (G41 or G42), you can only use it for X-Y plane circular motions (G17). Cutter radius compensation is not available for circular motion in the G18 XZ or G19 YZ circular motion planes.



## INCH / METRIC SELECTION (G20, G21)

### G20 Inch programming selection

### G21 metric programming selection

Selection between inch and metric programming can only be done from the Setting 9. Inch programming allows displacements up to 838 inches and a resolution of 0.0001 inches. Metric programming uses millimeter units with a maximum displacement of 8380 mm and a resolution of 0.001 mm.

When in metric, the feed rate also defined as millimeters per minute with a range of 1000. to 0.001 mm/min.

When jogging in metric, the speeds and units on the keypad are interpreted as mm/min but the value used is ten times larger than shown on the keypad.

The optional fourth and fifth axis programming is not effected by the selection of metric. It is always programmed in degrees. The auxiliary C axis is also always in degrees.

Changing the setting from inches to metric or back again will change the content of any programs already stored in memory. You must reload your programs with metric values after changing this setting.

The standard G codes G20 and G21 are sometimes used to select between inch and metric BUT, in this control, the G20 (inch) and G21 (mm) codes can only be used to ensure that the inch/metric setting is set correctly for that program.

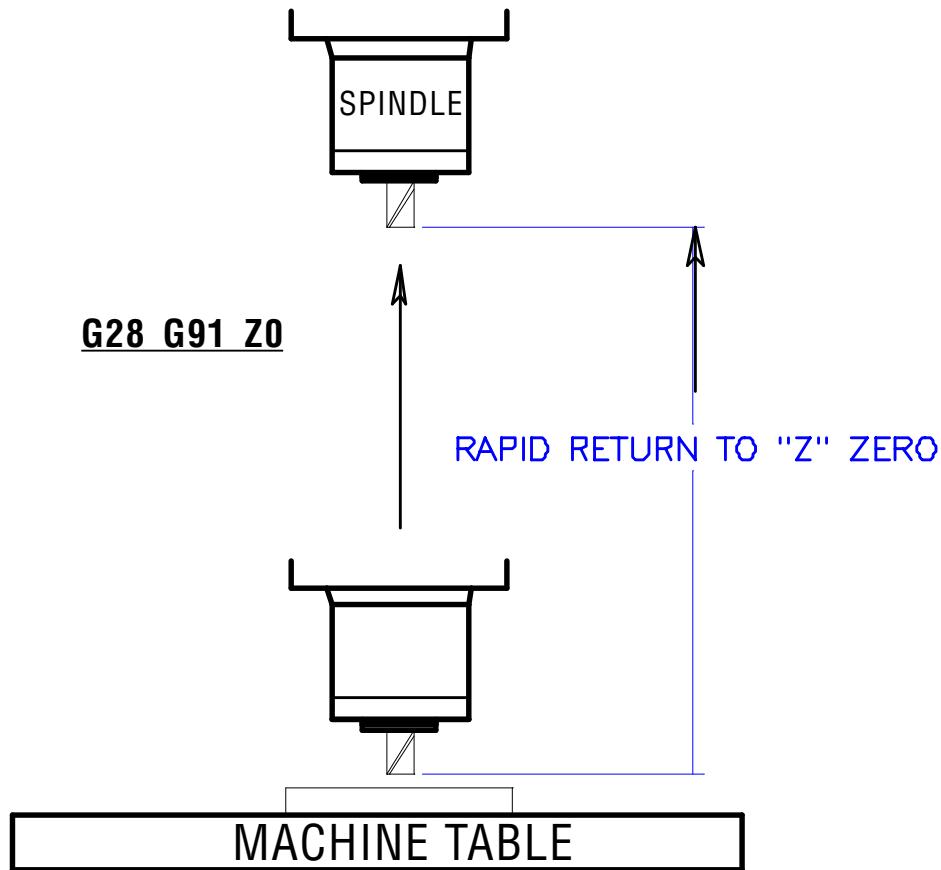


## RETURN TO REFERENCE POINT (G28)

G28 Return To Reference Point, set optional intermediate point

The G28 code is used to return to the machine zero position on all axes. If an X, Y, Z, or A axis is on the same block and specifies a location, only those axes will move and return to the machines' zero reference point and the movement to the machines' zero reference point will be through that specified location. This point is called the intermediate point and is saved, if needed, for use in G29. If you do not want to position through an intermediate point while specifying a specific axis to position to machine zero, than add a incremental (G91) command to this line along with a Z0, Y0, and/or X0 for the specific axis you want to send to machine zero. This will command those axes specified to position incrementally to a zero distance as an intermediate point, and then those axes specified will then position to machine zero. Just be sure to program in an absolute (G90) command in the start-up lines for the next tool which is usually needed for the beginning of each tool.

If no X, Y, Z, or A is specified, all axes will be moved directly to machine zero. Any auxiliary axes (B, C,...) are returned to there machine home after the X, Y, Z, and A axes. G28 also cancels tool length offsets.



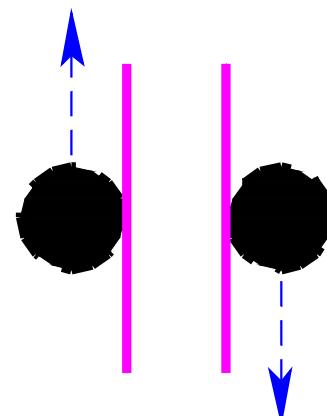


## CUTTER COMPENSATION

Cutter compensation is used to offset the center of the cutter, and shift it the distance of the radius, to the specified side of the programmed path. Complex part geometries having angled lines, lines tangent to arcs, and lines intersecting arcs involve substantial trigonometric computations to determine the center of the cutter. Cutter compensation involves programming the part geometry directly instead of the tool center. The cutter compensation commands are Cutter Comp Off (G40), Cutter Comp Left (G41), and Cutter Comp Right (G42).

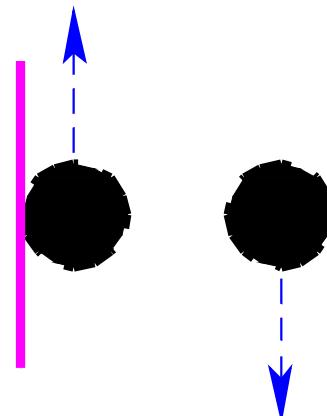
### G40 Cutter Comp Cancel

G40 will cancel the G41 or G42 cutter compensation commands. The tool will change from a compensated position to an uncompensated position. Programming D00 will also cancel cutter compensation.



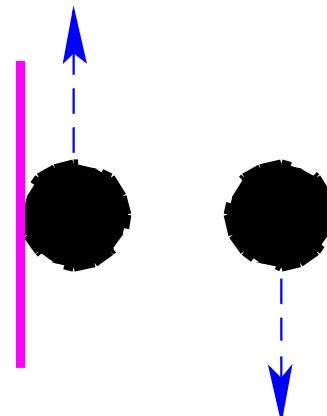
### G41 Cutter Compensation Left

G41 will select cutter compensation left; that is the tool is moved to the left of the programmed path to compensate for the radius of the tool. A Dnn must also be programmed to select the correct tool size from the DIAMETER/RADIUS offset display register.



### G42 Cutter Compensation Right

G42 will select cutter compensation right; that is the tool is moved to the right of the programmed path to compensate for the size of the tool. A Dnn must also be programmed to select the correct tool size from the DIAMETER/RADIUS offset display register.



**Dnn Cutter Comp value** - The actual offset amount must be input in the specified tool offset display number. On the HAAS you have 100 tool DIAMETER/RADIUS offsets to use. Usually, you will have only one cutter offset for a tool, and it's best to use the same cutter offset number as is your tool number. On the HAAS offset tool DISPLAY page, the two columns on the right side of this display are for your cutter diameter offsets. The cutter GEOMETRY column, in the offset display, is to set your initial cutter offset value and you can designate it as either, a DIAMETER value or a RADIUS value, by selecting the one you would like to use with SETTING 40. And the selection you choose will be listed at the top of the offset geometry column. The WEAR column on the right of the tool GEOMETRY column is for any adjustments you need to make from the initial tool GEOMETRY offset. And these values are added together by the control and used as one value.

**G40 Cutter Comp Cancel G41 and G42****G41 Cutter Compensation Left****G42 Cutter Compensation Right**

Understanding cutter compensation can be simplified if one has a basic understanding of manual machining. There are two common types of cutting conditions associated with milling machines. They are CLIMB and CONVENTIONAL cutting.

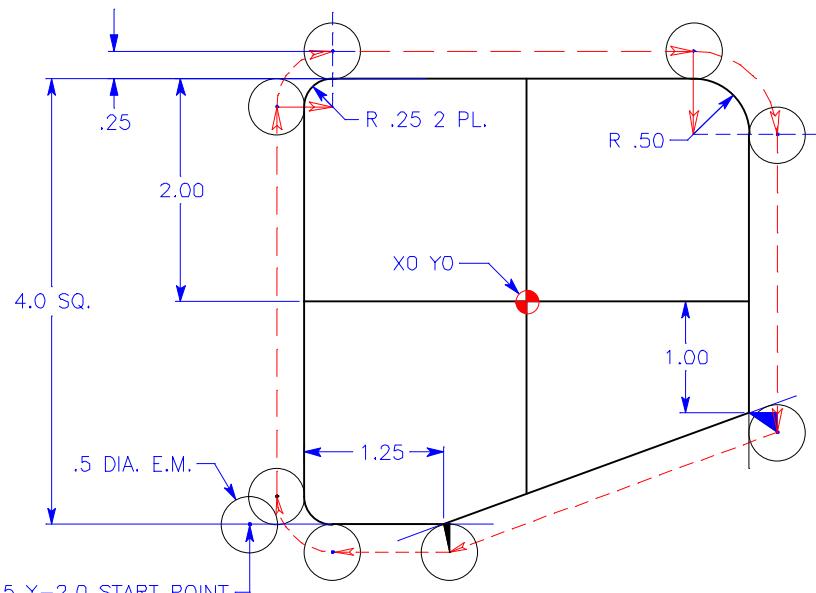
*Two common rules for these types of cuts are:*

If the programmed cutter path needs to mill CLIMB cutting, which is what's usually desired on CNC machines, since the machines are rigid enough to handle this type of cut, and the conditions from this type of cut are usually preferred. And if it's a standard right handed tool, it will then be programmed with G41 cutter LEFT of the programmed path.

If the programmed cutter path needs to mill with CONVENTIONAL cutting, and it's a standard right handed tool, it will then be programmed with G42 cutter RIGHT of the programmed path.

Program without cutter compensation.

```
N101 G00 X-2.5 Y-2.0
N102 G01 Z-.45 F30.
N103 X-2.25 F12.
N104 Y1.75
N105 G02 X-1.75 Y2.25 R.5
N106 G01 X1.5
N107 G02 X2.25 Y1.5 R.75
N108 G01 Y???? (Calculate point)
N109 X???? Y-2.25 (Calculate point)
N110 G01 X-1.75
N111 G02 X-2.25 Y-1.75 R.75
N112 G01 X-2.35
```



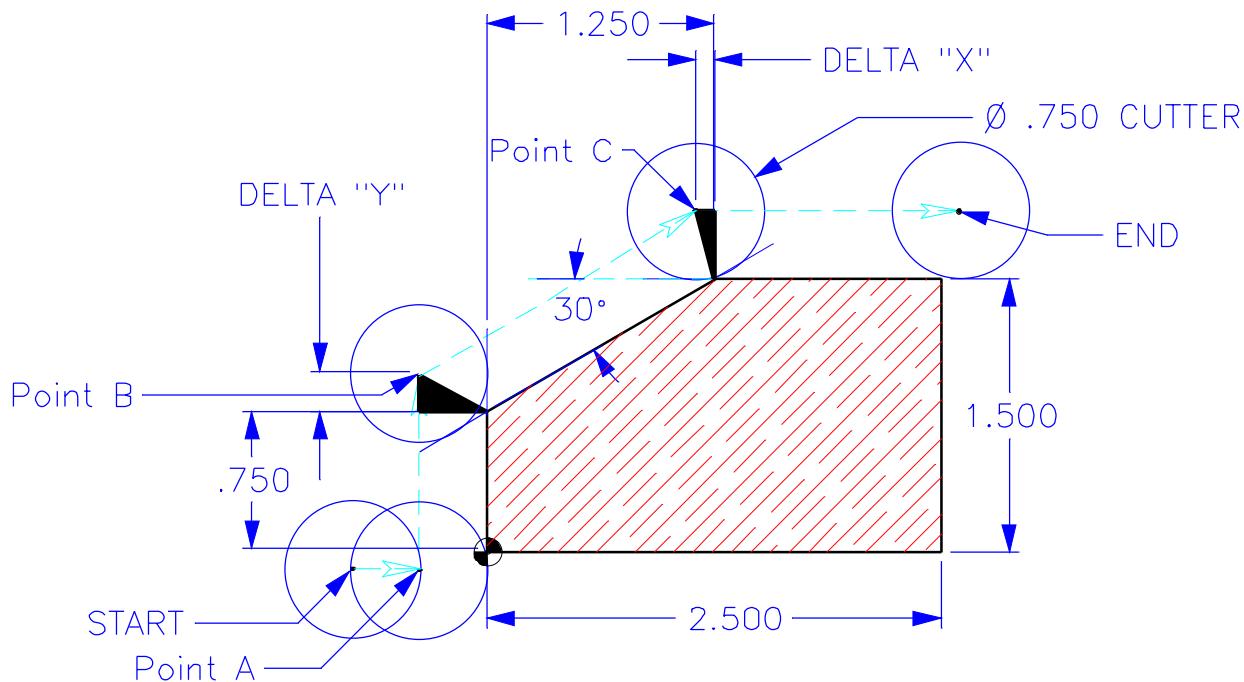
Program with cutter compensation.

Dia. value for D01 would be .500  
in DIAMETER offset register #1.

```
N201 G00 X-2.5 Y-2.0
N202 G01 Z-.45 F0.
N203 G41 X-2. D01 F12. (turn on C.C. with an X and/or Y move)
N204 Y1.75
N205 G02 X-1.75 Y2. R.25
N206 G01 X1.5
N207 G02 X2. Y1.5 R.5
N208 G01 Y-1.
N209 X-.75 Y-2.
N210 X-1.
N211 G02 X-2. Y-1.75 R.25
N212 G40 G01 X-2.35 (turn off C.C. with an X and /or Y move)
```



When programming without cutter compensation, to the center of the cutter, a problem occurs cutting angle geometry. The cutter center must be offset to the part geometry to maintain the cutter tangency. For example, the Y axis move from Point A to Point B must have the Delta "Y" calculated dimension added to the .75 dimension. The X axis move from Point B to Point C must have the Delta "X" calculated dimension subtracted from the 1.25 dimension.



Program is to the center of the cutter manually calculating the cutter compensation of tool.

**001052**

**N1 T1 M06 (3/4 DIA. END MILL)**

**N2 G90 G54 G00 X-0.75 Y-0.15** (X Y positioning tool center to center)

**N3 S2200 M03**

**N4 G43 H01 X0.1 M08**

**N5 G01 Z-0.5 F50.**

**N6 X-0.375**

**N7 Y0.????**

**N8 X?.???? Y1.875**

**N9 X2.6**

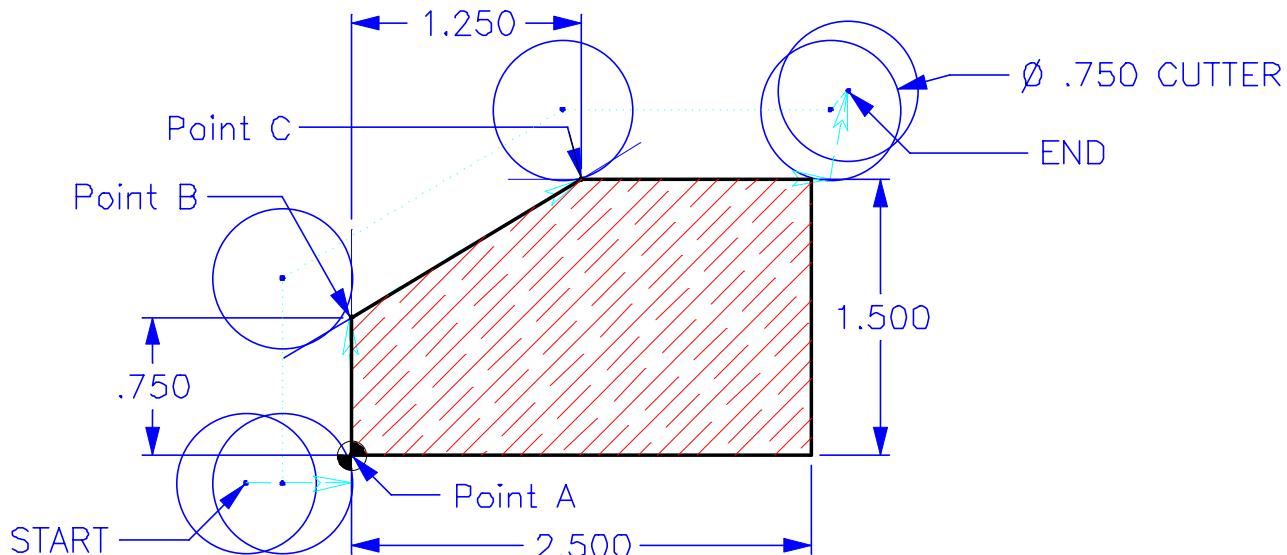
**N10 G00 Z1. M09**

**N11 G28 G91 Y0. Z0. M05**

**N12 M30**



If however, the programming is being done with cutter compensation in effect, the moves from Point A , to Point B, to Point C will reflect the actual geometry or part print X and Y coordinate values. The control will offset the cutter center around Point B and Point C to maintain the required cutter tangency.



### 001053 (Programming part using cutter compensation)

**N1 T1 M06 (3/4 DIA. END MILL)**

**N2 G90 G54 G00 X-0.575 Y-0.2** (X Y position to the center of tool)

**N3 S2200 M03**

**N4 G43 H01 Z0.1 M08**

**N5 G01 Z-0.5 F50.**

**N6 G41 X0. D01 F12.** (Turning on Cutter Compensation)

**N7 Y0.75**

**N8 X1.25 Y1.5**

**N9 X2.6**

**N10 G40 X2.7 Y1.95** (When canceling C.C. position to the X Y center of tool)

**N11 G00 Z1. M09**

**N12 G28 G91 Y0. Z0. M05**

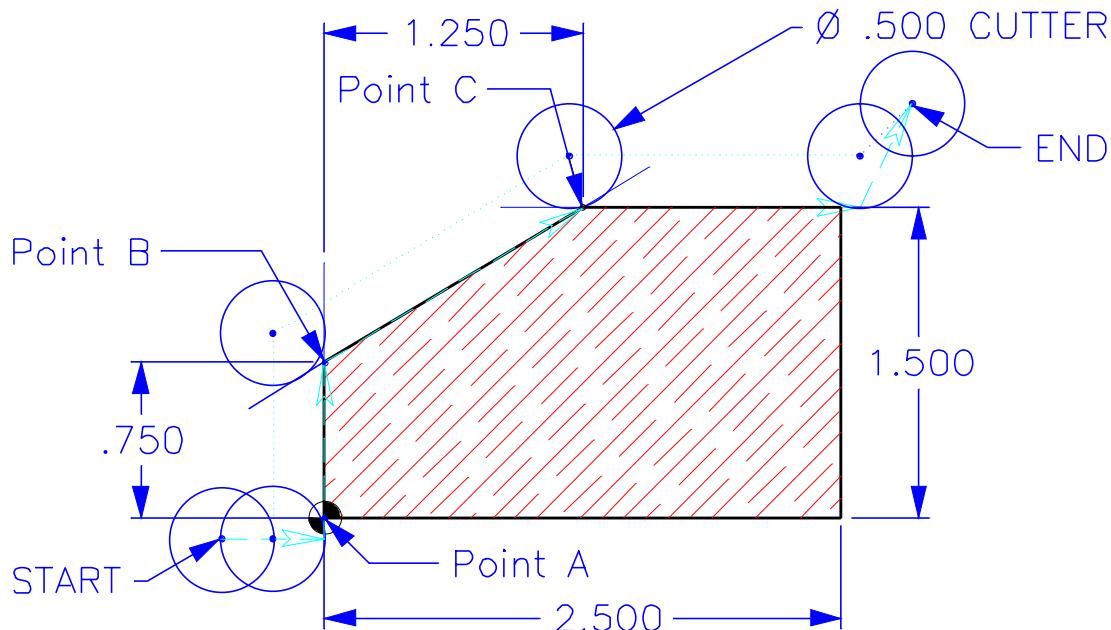
**N13 M30**

When the part has been programmed using cutter compensation we are, in effect, programming with a zero diameter cutter, to the center of the tool. The tool diameter value is entered in on the OFFSET display page under the RADIUS/DIAMETER geometry offset column. *The control will position the tool the offset amount, off of the programmed part line that is programmed so that the edge of the tool is positioning around the part geometry*



If, at the time of setup, the cutting tool requested by the planning is for a diameter not available currently in your shop, then input the available size on the offset page. DO NOT change the program. Cutter compensation takes the stored value for the diameter and calculates the cutter path offset from that value.

If a larger tool is going to be used, you will need to make sure that you change the starting and ending positions so that the distance of the cutter is positioned half the diameter off the part for clearance when you lead onto and off of the part.



Machining part with C.C. using an undersize tool.

**001053**

**N1 T1 M06 (1/2 DIA. END MILL)**

**N2 G90 G54 G00 X-0.5 Y-0.15** (Positioning X Y Center location of tool)

**N3 S2200 M03**

**N4 G43 H01 Z0.1 M08**

**N5 G01 Z-0.5 F50.**

**N6 G41 X0. D01 F12.** (Turning on Cutter Compensation)

**N7 Y0.75**

**N8 X1.25 Y1.5**

**N9 X2.6**

**N10 G40 X2.95 Y2.**

**N11 G00 Z1. M09**

**N12 G28 G91 Y0. Z0. M05**

**N13 M30**



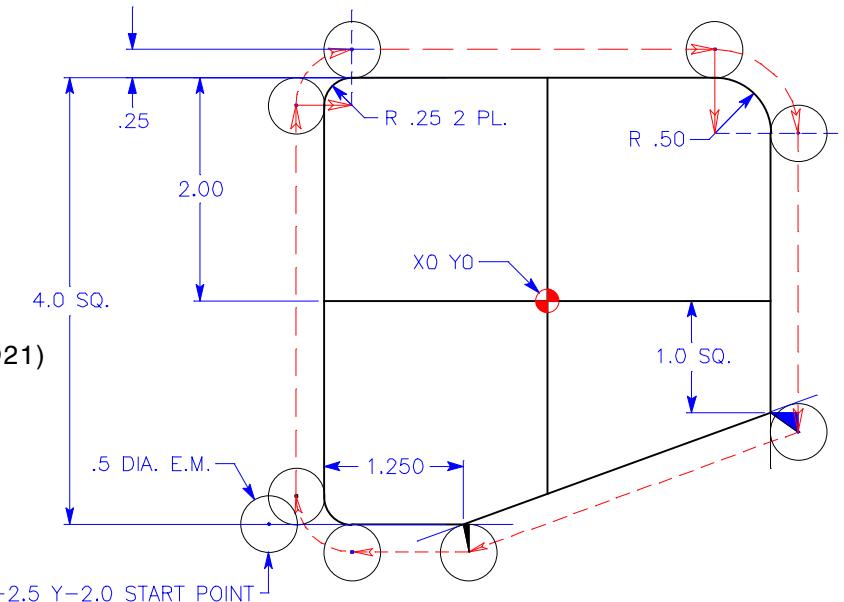
NOTE: Advantage of cutter compensation can be taken when a rough and finish pass must be done with one tool. As we have seen, if we program the finish geometry and use cutter compensation, we will get a finish pass.

If however, prior to the rough pass, we input a diameter offset value that is oversize to the physical size of the end mill, and then program the finish geometry, the control will offset the end mill more, leaving stock for a finish pass. Then program the same end mill using another offset with the correct offset value and the same geometry.

EXAMPLE: If the cutter is .500 diameter, and you want to leave .010 stock for a finish pass. Then enter a diameter offset number (Dn) command for tool DIAMETER Geometry of .520 for machining a roughing pass around part. And then define another pass around part using the same program coordinates but a different diameter offset number (Dn), and enter in the correct diameter offset value of .500 to machine part to the finish size.

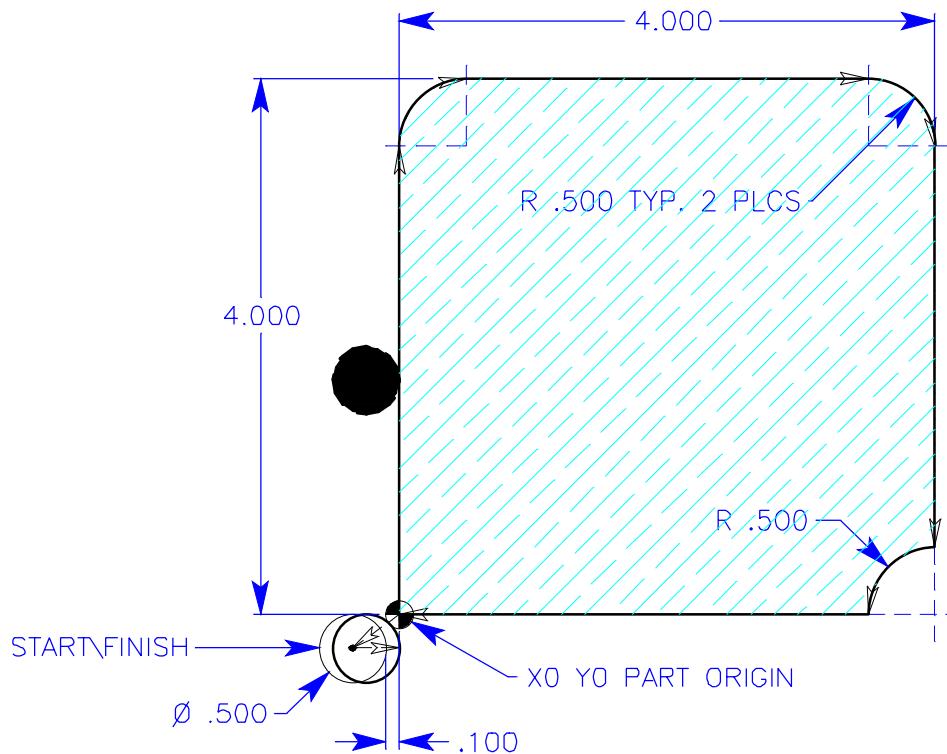
**TOOL 1 HAS .520 IN D21  
AND .500 IN D01**

```
O01052
N1 T1 M06
N2 G90 G54 G00 X-2.5 Y-2. S1600 M03
N3 G43 H01 Z0.1 M08
N4 G01 Z-0.45 F30.
N5 G41 X-2. D21 F12. (C.C. on using D21)
N6 Y1.5
N7 G02 X-1.75 Y2. R0.25
N8 G01 X1.5
N9 G02 X2. Y1.5 R0.5
N10 G01 Y-1.
N11 X-.75 Y-2.
N12 X-1.75
N13 G02 X-2. Y-1.75 R0.25
N14 G40 G01 X-2.35 Y-2. (Turn off C.C. with an X and/or Y move off of part and then do a finish
pass around part using the same part geometry but a different C.C. offset.)
N15 G41 X-2. D01 F8. (Turn on C.C. using D01 for a finish pass using the same program geometry.)
N16 Y1.5
N17 G02 X-1.75 Y2. R0.25
N18 G01 X1.5
N19 G02 X2. Y1.5 R0.5
N20 G01 Y-1.
N21 X-.75 Y-2.
N22 X-1.75
N23 G02 X-2. Y-1.75 R0.25
N24 G40 G01 X-2.35 Y-2. (Turn off C.C. with an X and/or Y move off of part.)
N25 G00 Z1. M09
N26 G28 G91 Z0 Y0 M05
N27 M30
```





## CUTTER COMPENSATION EXERCISE #1



Cutter will be a .500 dia. end mill. Start contour from the lower left corner of part and mill around outside of part .625 deep. Mill around outside of part using cutter compensation for cutter offset and then define the actual geometry of the part to position tool around part.

Cutter is a 1/2 dia. End Mill. Mill around outside of part .625 deep using cutter comp.

### O00030 (CUTTER COMPENSATION EXERCISE #1)

T1 M06 (1/2 DIA. 4FLT. END MILL)

G90 G54 G00 X\_\_\_\_\_ Y\_\_\_\_\_ (Position with the center of tool)

S1400 M03

G43 H01 Z0.1 M08

G01 Z\_\_\_\_\_ F50. (Fast feed to depth non-cutting move)

G\_\_\_\_ X\_\_\_\_\_ D\_\_\_\_ F10. (Turn on C.C. moving onto part)

Y\_\_\_\_\_

G0\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_ R\_\_\_\_\_

G0\_\_\_\_ X\_\_\_\_\_

G0\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_ R\_\_\_\_\_

G0\_\_\_\_ Y\_\_\_\_\_

G0\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_ R\_\_\_\_\_

G0\_\_\_\_ X\_\_\_\_\_ (Position off part at least half the cutter diameter)

G\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_ (Turn off C.C. moving off the part)

G00 Z1. M09

G91 G28 Z0 M05

M30



## ADVANTAGES OF CUTTER COMPENSATION

1. The mathematical computations for determining a tool path are greatly simplified.
2. Because the geometry and not the tool center is programmed, the same program can be used for a variety of different cutter diameters.
3. When using cutter compensation you are then able to control and adjust for part dimensions using your cutter diameter/radius offsets register.
4. The same program path can be used for the roughing passes as well as finishing cuts by using different cutter offset numbers.

## SOME RESTRICTIONS WITH CUTTER COMPENSATION

1. A cutter compensation command (G41, G42 or G40) must be on the same block with an X and/or Y linear command when *moving onto or off of the part* using cutter comp. You cannot turn on or off cutter compensation with a Z axis move.
2. You can use cutter comp. in the G18 (X,Z) or G19 (Y,Z) planes using G141.
3. You cannot turn ON or OFF cutter compensation in a G02 or G03 circular move, it must be in a linear G00 or G01 straight line move.

## WHEN ACTIVATING CUTTER COMPENSATION, CARE MUST BE TAKEN TO:

1. Select a clearance distance without cutter compensation to a start point in "X" and "Y" axis, at least half the cutter diameter, off the part before you start initiating cutter compensation.
2. Bring the "Z" axis down without cutter compensation in effect.
3. Make an X and/or Y axis move with a G41 or G42 call-out on the same line, with a diameter offset Dnn command, which has the cutter diameter value in the offset display register being used.

## WHEN DEACTIVATING CUTTER COMPENSATION, CARE MUST BE TAKEN TO:

1. Select a clearance point in "X" and/or "Y" axis off the part.
2. "DO NOT" cancel cutter compensation on any line that is still cutting the part.
3. Cancel of cutter compensation may be a one or two axis move.
4. Cancel of cutter compensation (G40) may need values entered for both "X" and "Y" axis. This may need to be done to ensure that both axes will position to the location you want, or remain fixed and not move during the cancel (G40) process. This is a programming technique that may be a programmer's preference. The code can be written with only a one axis move, but the programmer should be aware of the results. If only one of the axes, X or Y, is on the cancel (G40) command line, the control will move both axes during the cancel process. And the machine will position the axis that was not defined back to center position of that tool to the last known axis coordinate values.

JUNE 2000

**VF/H5 Series**  
WORKBOOK

PROGRAMMING





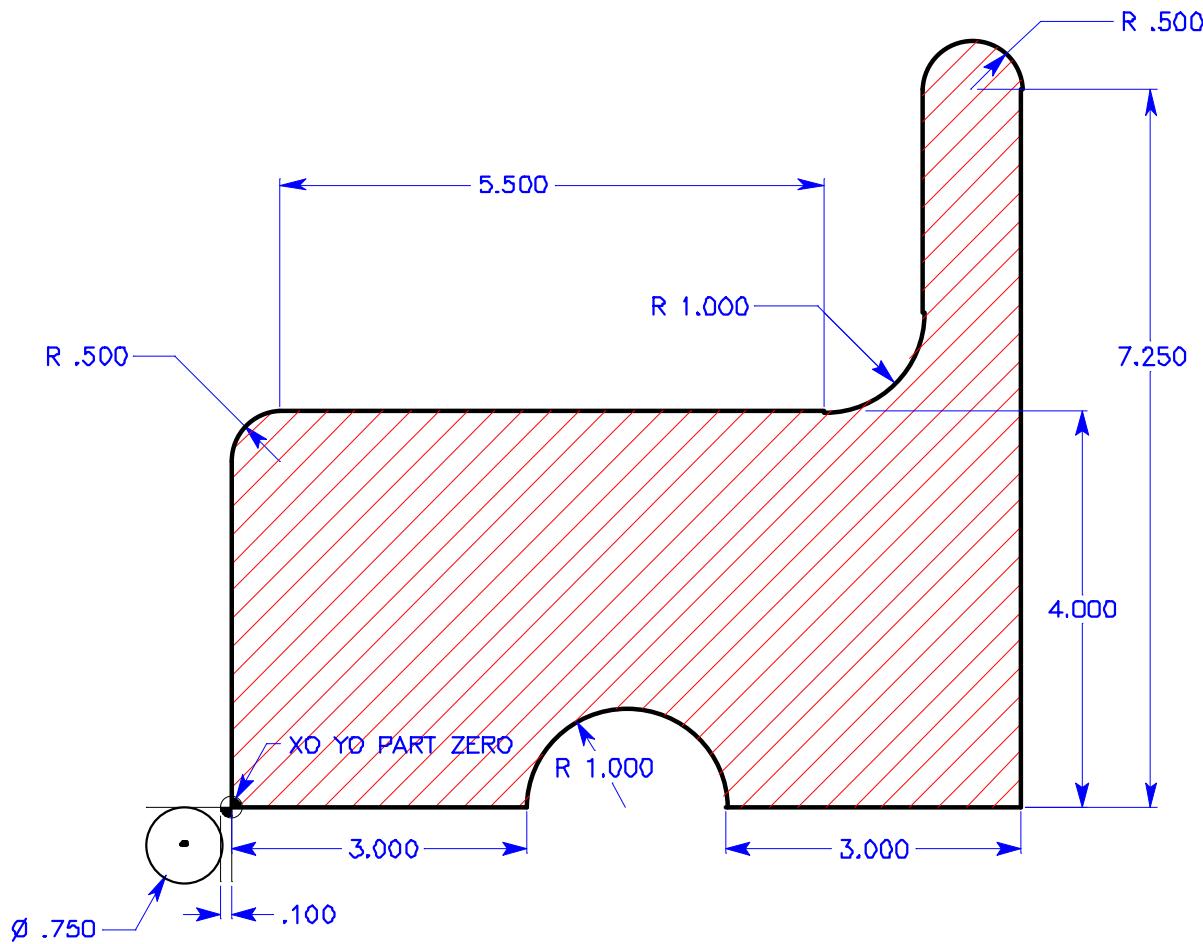
## CUTTER COMPENSATION EXERCISE #2

To program this part profile, use the .750 diameter end mill, starting at the lower left hand corner. Use CUTTER RADIUS COMPENSATION, and Climb cut part profile, with only one finish pass .450 deep.

(Use the HAAS calculator to get the spindle speed and feed)

TOOL 1 - 3/4 DIA. 4 FLT. END MILL

Surface Speed (FPM) - 190.  
Chip Per Flute - .0025





# **CUTTER COMPENSATION EXERCISE #2**

## **O00040 (CUTTER COMPENSATION EXERCISE #2)**

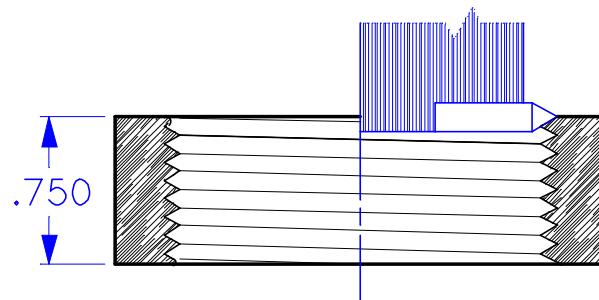
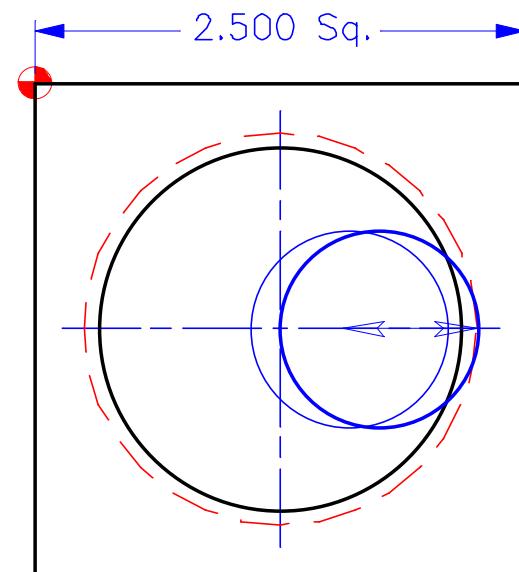


## THREAD MILLING WITH HELICAL MOTION

Thread milling uses a cutter formed with the pitch of the thread to be cut. The cutters are solid carbide, fragile and expensive.

Thread milling is accomplished with Helical milling. Use a standard G02 or G03 move to create the circular move in the X-Y axis and then insert a Z move on the same block corresponding to the thread pitch. This will generate one turn of the thread. If you are using a multiple tooth cutter, it will generate the rest of the thread depth. The speed of each axis will be controlled so that the helical rate matches the programmed feed rate.

Thread milling can be used to machine larger internal or external threads. You can also thread mill, port threads, blind hole threads, metric threads and most all the special type of threads that are good to thread mill. You can also adjust for the thread size using diameter compensation.



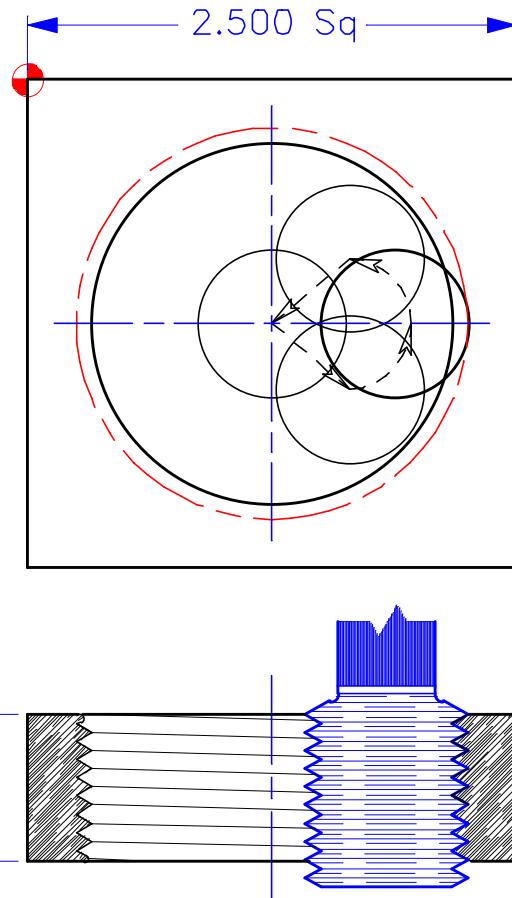
2-12 UN 3B  
Minor Dia. - 1.9100/1.9198

To do a complete 360 degree thread pass using a G02 or G03 you need to use an I and/or J to define the center of a circle for a complete thread pass. Because you cannot do a 360 degree arc using an R command.

```
N1 T1 M06 (1.0 dia. single point thread tool to do a 2.0-12 thread)
N2 G90 G54 G00 X1.6 Y-1.25
N3 S1500 M03
N4 G43 H01 Z.1 M08
N5 G01 Z-.8 F50.
N6 G41 X2.25 D01 F10. (Move over to thread and turn on C.C.)
N7 G91 G03 X0 Y0. I-1. J0. Z.0833 F3. L10 (Repeat 10x to mill thread)
N8 G90 G40 G01 X1.6 Y-1.25 (Move off of thread and turn off C.C.)
N9 G00 Z.1 M09
N10 G91 G28 Z0.
N11 M30
```



## HELICAL MOTION



2-12 UN 3B  
Minor Dia. - 1.9100/1.9198

- N1 T1 M06** (12 pitch x 3/4 inch dia. 4 flt. thread hob for a 2.0-12 thread)
- N2 G90 G54 G00 X1.25 Y-1.25**
- N3 S1500 M03**
- N4 G43 H01 Z.1 M08**
- N5 G01 Z-1.0 F50.**
- N6 G41 X1.75 Y-1.75 D01** (position over to turn on cutter comp)
- N7 G03 X2.25 Y-1.25 R.5 F10.** (arc into thread)
- N8 G03 X2.25 Y-1.25 I-1. J0. Z-.9167 F12.** (to mill complete thread)
- N9 G03 X1.75 Y-.75 R.5** (arc off of thread)
- N10 G40 G01 X1.25 Y-1.25.**(position over to turn off cutter comp)
- N11 G00 Z.1 M09**
- N12 G91 G28 Z0.**
- N13 M30**



## TOOL LENGTH COMPENSATION

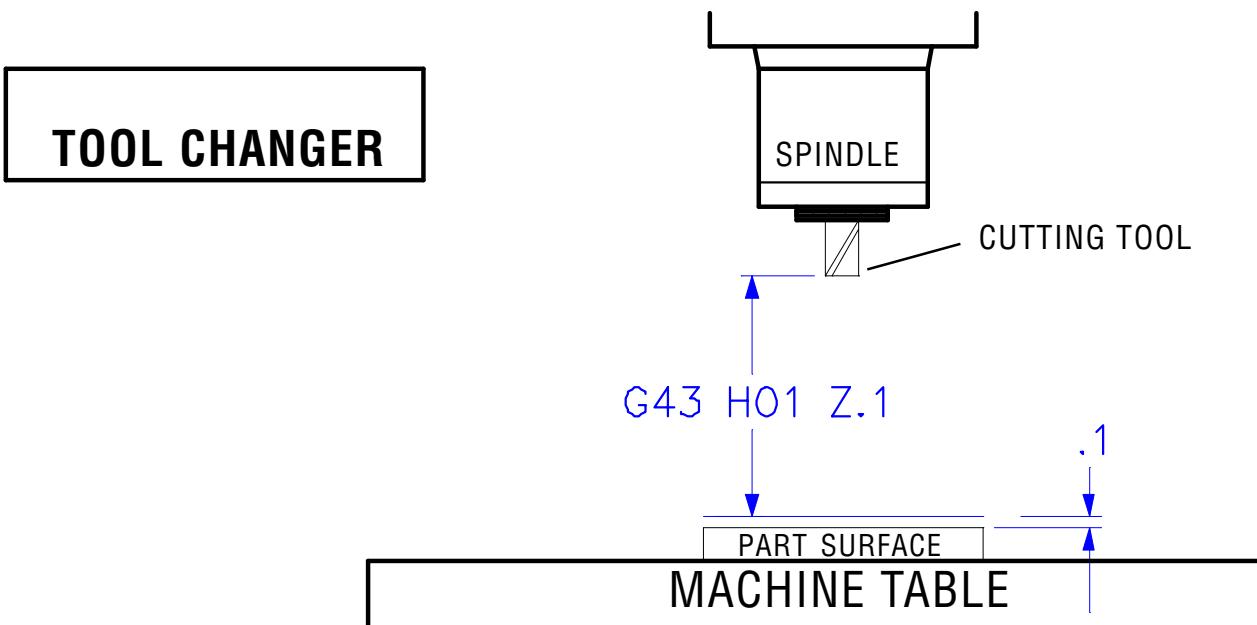
### G43 TOOL LENGTH COMPENSATION + (plus)

This code selects tool length compensation in a positive direction. That is; the tool length offsets are added to the commanded axis positions. An Hnn must be programmed to select the correct offset register from the offset display for that tool being used.

During the setup process, each tool point was touched-off to the part zero surface. From this position a TOOL LENGTH DISTANCE offset was recorded for that tool with the TOOL OFFSET MEASUR key. This TOOL LENGTH is referred to as the "Z" axis origin move to the part zero surface.

The programmed code to position tool:

**G43 H01 Z.1**



### G44 TOOL LENGTH COMPENSATION - (minus)

This code selects tool length compensation in a negative direction. That is; the tool length offsets are subtracted from the commanded axis positions. A Hnn must be programmed to select the correct entry from offsets memory.

### G49 CANCELS G43/G44

This G code cancels tool length compensation. Putting in a H00 will also cancel tool length compensation. G28, M30, and RESET will also cancel tool length comp.



## WORK COORDINATE SELECTION

### G54-59 WORK COORDINATE OFFSET SYSTEM #1 - #6

These codes select one of the six user coordinate offsets stored within the offsets memory. All subsequent references to axes' positions will be interpreted in the new coordinate system. Work coordinate system offsets are entered from the Offsets display page.

### G110-G129 WORK COORDINATE OFFSET SYSTEM #7-26

These codes are the same as work offsets G54 to G59.

## MORE WORK COORDINATE SELECTIONS

**Note:** The G52 command works differently depending on the value of Setting 33. This setting selects the FANUC, HAAS, or YASNAC style of coordinates, which are listed below:

### G52 WORK COORDINATE OFFSET SYSTEM YASNAC

This code selects the G52 work coordinate system. The G52 works the same as G54 etc., except that the G52 system will be set by a G92 code as well as from the offsets display.

### G52 SET LOCAL COORDINATE WORK OFFSET SHIFT VALUE FANUC

This code sets the origin of the local (child) coordinate system to the command location, relative to the current work system origin. G52 is a non-modal, no motion code. The G52 coordinate system will stay in effect for all work systems until it is canceled. The G52 is canceled when RESET is pressed and at the end of a program. It is also canceled during a program by M30, G52, X0 Y0 Z0, or by a G92 command.

### G52 SET LOCAL COORDINATE WORK OFFSET SHIFT VALUE HAAS

This code acts the same as in the Fanuc control except that G52 is not cleared at power-up, RESET, or when an M30 is performed. It is canceled with a G52, X0 and/or Y0 and/or Z0, or by a G92 command.

### G53 NON-MODAL MACHINE POSITION COORDINATE SYSTEM

This code temporarily cancels work coordinates offset and uses the machine coordinate system (machine zero). It is non-modal; so the next block will revert to whatever work offset command that was previously selected.



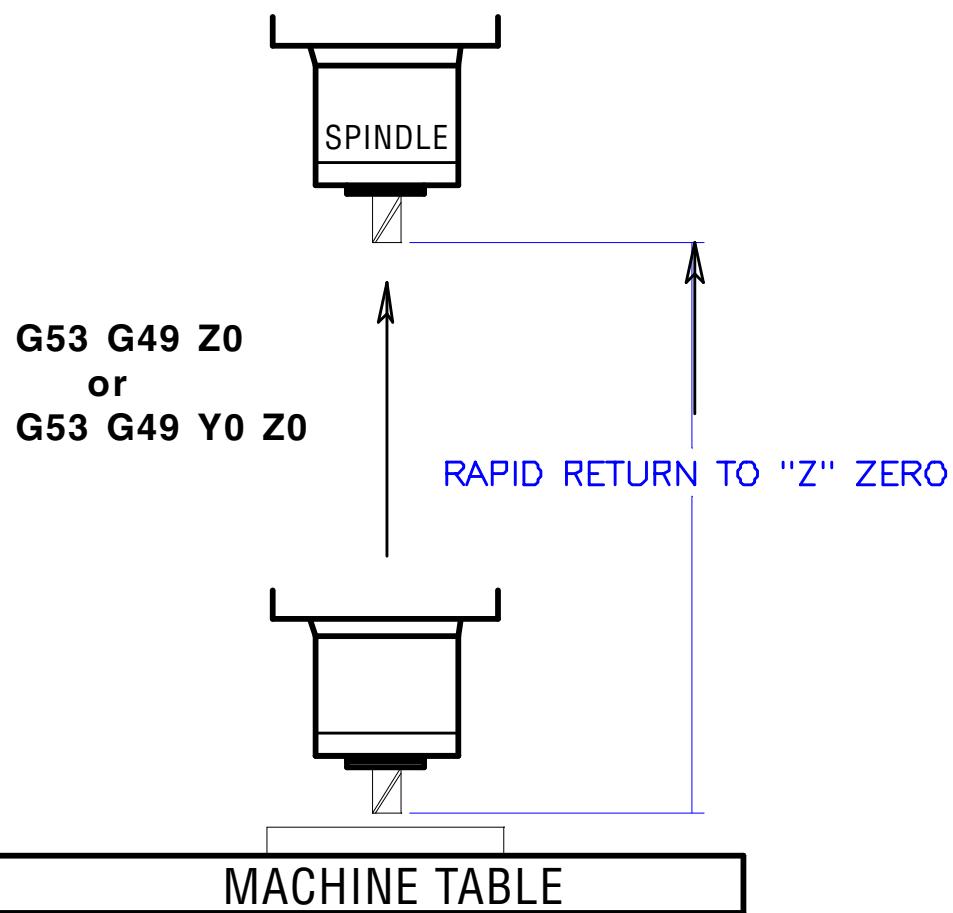
## ANOTHER WAY TO RETURN TO MACHINE ZERO, USING (G53)

### G53 Non-Modal Machine Coordinate Selection

This code temporarily ignores work coordinates offset and uses the machine coordinate system (machine zero). This gives you a command to move to a specific location defined from the machine zero reference point. It is non-modal; so the next command block will revert back to the work coordinate offset command, that was previously active.

This G53 command may be used to send the machine home for a tool change, or to send the machine home in Y, Z axes at the end of a program, rather than using a G28. A G28 works good for sending the machine home, and most people have learned to send it home this way. But to send it straight home with a G28, and a specific axis, you need to also define a G91 so that it positions through the intermediate point incrementally to go straight home (see p.52). And then, you will usually need to be sure to switch back to a G90 absolute command, for your next move. And instead of dealing with switching in and out of absolute and incremental

you could command a G53 along with a Z0, Y0, and/or X0 for the specific axis you want to send to machine zero.



Just be sure to cancel your tool length offset with a (G49) command when you define a G53 G49 Z0 to send Z axis to machine home for a tool change.



## CANNED CYCLES

A canned cycle is used to simplify programming of a part. Canned cycles are defined for the most common Z-axis repetitive operation such as drilling, tapping, and boring. There are 13 canned cycles to choose from and once selected a canned cycle is active until canceled with G80. When active, the canned cycle is executed every time an X and/or Y-axis move is programmed. Those X Y moves are executed as RAPID commands and the Z motions are executed as FEED commands, and the canned cycle is performed again for any X-Y coordinates that follow. The five operations in a canned cycle are:

- 1) Positioning of the X and Y axes (and optional A axis).
- 2) Rapid traverse to the R plane.
- 3) The drilling, tapping, and boring for the canned cycle being used.
- 4) Operation at bottom of hole.
- 5) Retraction to the R plane.

The positioning of a canned cycle in the X and/or Y axes can be done in either in absolute (G90) or incremental (G91).

Incremental (G91) motion in a canned cycle is often useful with a loop (Lnn) count which will repeat the canned cycle operation that many times with each incremental X or Y move for the canned cycle.

If an L0 is in the canned cycle line, the cycle will not execute until the control reads the next X and/or Y position location.

G98 and G99 are modal commands which change the way the canned cycles operate. When G98 (machine default value) is active, the Z-axis will be returned to the *initial start point* at the completion of the canned cycle. When G99 is active, the Z-axis will be returned to the *reference (R) plane* when the canned cycle is completed. Changes to the G98/G99 selection can also be made after the canned cycle is active. If changed, the new G98/G99 value will change all subsequent canned cycle.

If a canned cycle is defined in a block without an X or Y location, there are two common actions taken by other controls; some will execute the canned cycle at that time and some will not. With the HAAS Mill, these two options are selectable with Setting 28.

Once a canned cycle is defined, that operation is performed at every X-Y position subsequently listed in a block. Some of the canned cycle numerical values can also be changed after the canned cycle is defined. The most important of these are the R plane value and the Z depth value, and all subsequent cycles are performed with the new R or Z value.



## CANNED CYCLES

G CODE	Z Drilling Operation	Operation at the End of Hole	Z axis Retraction	Application
<b>G73</b>	intermittent feed in	dwell (optional)	rapid out	high speed peck drilling
<b>G74</b>	spindle CCW feed in	spindle reverse CW	feed out	left hand tapping
<b>G76</b>	feed in	dwell, spindle stop, orient, shift	rapid out	fine boring
<b>G77</b>	spindle stop, shift, rapid in, shift, feed up	orient spindle, shift	rapid out	back boring
<b>G81</b>	feed in	none	rapid out	drilling
<b>G82</b>	feed in	dwell	rapid out	spot drilling
<b>G83</b>	intermittent feed in	dwell (optional)	rapid out	peck drilling cycle
<b>G84</b>	spindle CW feed in	spindle reverse CCW	feed out	tapping cycle
<b>G85</b>	feed in	none	feed out	boring cycle
<b>G86</b>	feed in	spindle stop,orient	rapid out	boring cycle
<b>G87</b>	feed in	spindle stop, orient	manual jog out	boring cycle
<b>G88</b>	feed in	dwell, spindle stop, orient	manual jog out	boring cycle
<b>G89</b>	feed in	dwell	rapid out	boring cycle

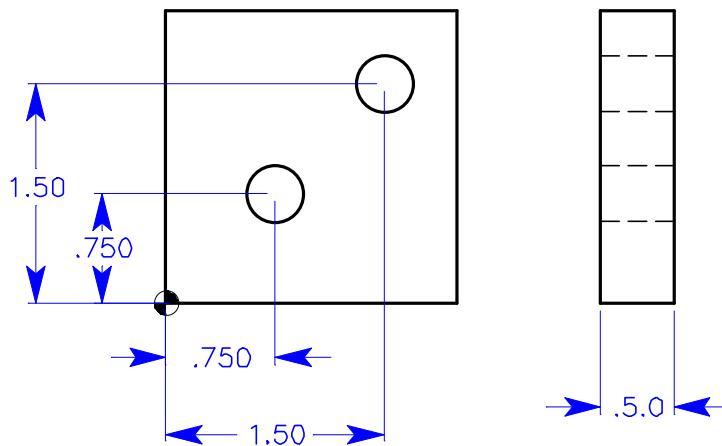
## G80 CANNED CYCLE CANCEL

The G80 code is used to cancel a canned cycle. This G code is modal in that it deactivates all canned cycles G73, G74, G76, G77, or G81-G89 until a new one is selected. In addition to this, a G00 or G01 code will also cancel any active canned cycle.

**G81 DRILL CANNED CYCLE**

- X** Optional rapid X-axis command
- Y** Optional rapid Y-axis command
- Z** Bottom of hole (*Feed point of Z depth starting from R plane*)
- R** Rapid position of the R plane (*Where you rapid, to start feed*)
- F** Feed Rate in inches (mm) per minute

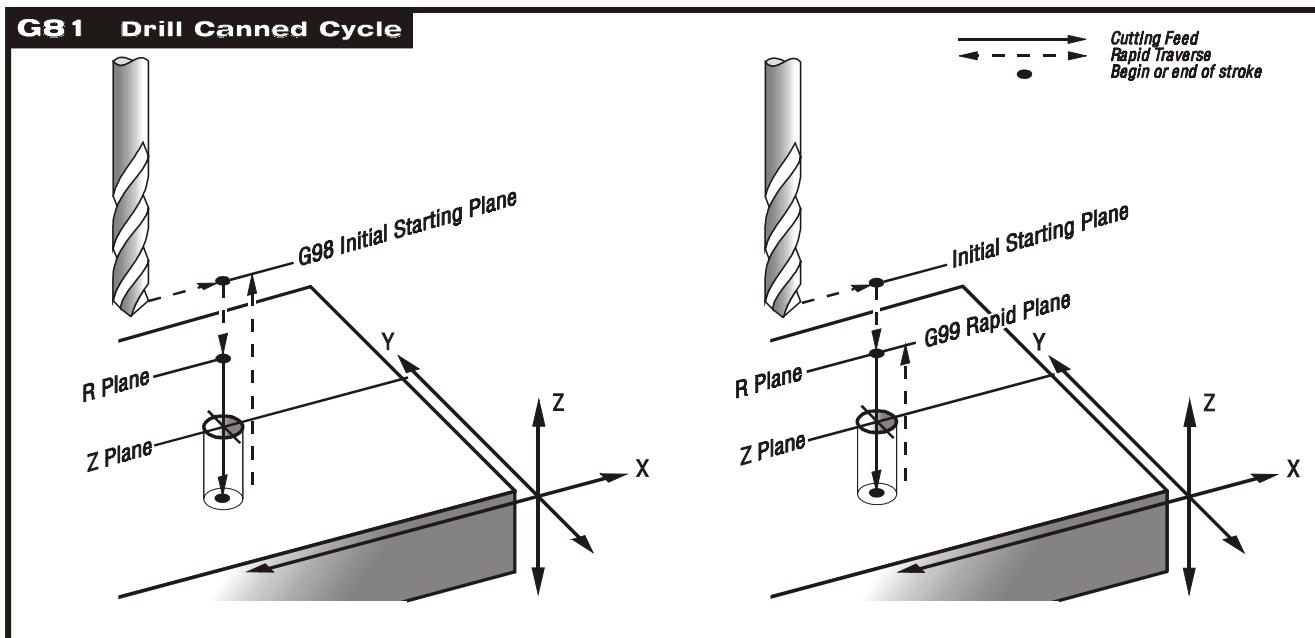
This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. Use G98 and G99 for the Z position clearance location for positioning between holes.



```

N1 T1 M06
N2 G90 G54 G00 X.75 Y.75
N3 S1200 M03
N4 G43 H01 Z1. M08
N5 G81 G99 Z-.625 R.1 F10.
N6 X1.5 Y1.5
N7 G80 G00 Z1. M09
N8 G28 G91 Z0. M05
N9 M30

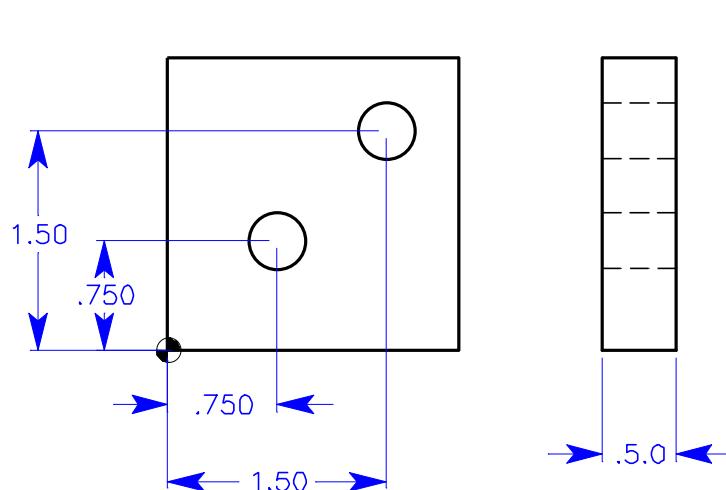
```



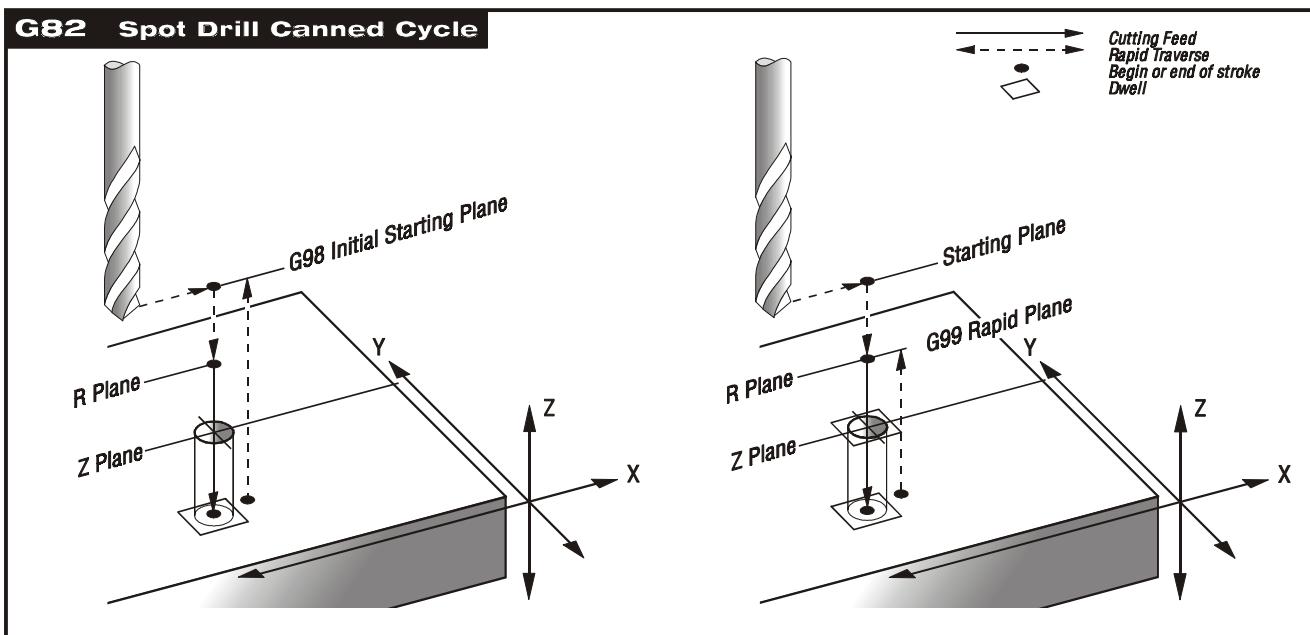
**G82 SPOT DRILL/COUNTERBORE CANNED CYCLE**

- X Optional rapid X-axis command
- Y Optional rapid Y-axis command
- Z Bottom of hole (*Feed point of Z depth starting from R plane*)
- P The dwell time at the bottom of the hole
- R Rapid position of the R plane (*Where you rapid, to start feed*)
- F Feed Rate in inches (mm) per minute

This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. A dwell in seconds/milliseconds is caused at the bottom of each Z-depth in this cycle which is defined with P. Use G98 and G99 for the Z position clearance location for positioning between holes.



**N1 T2 M06**  
**N2 G90 G54 G00 X.75 Y.75**  
**N3 S1200 M03**  
**N4 G43 H02 Z1. M08**  
**N5 G82 G99 Z-.625 P1.5 R.1 F8.**  
**N6 X1.5 Y1.5**  
**N7 G80 G00 Z1. M09**  
**N8 G28 G91 Z0. M05**  
**N9 M30**



**G83 DEEP HOLE PECK DRILL CANNED CYCLE**

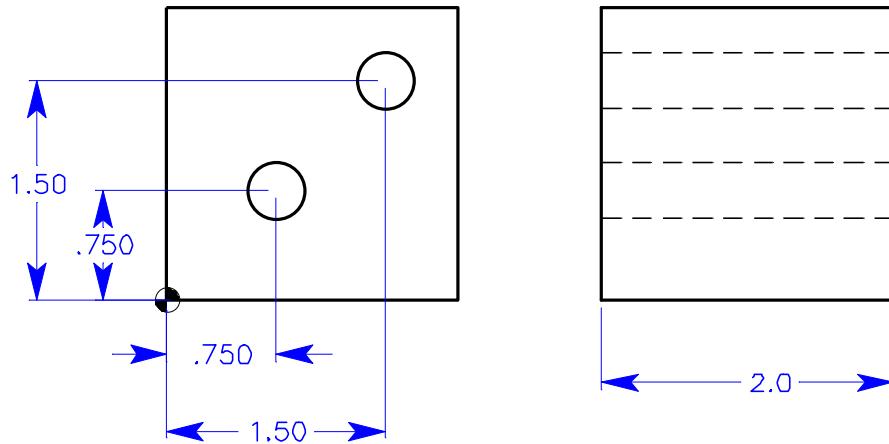
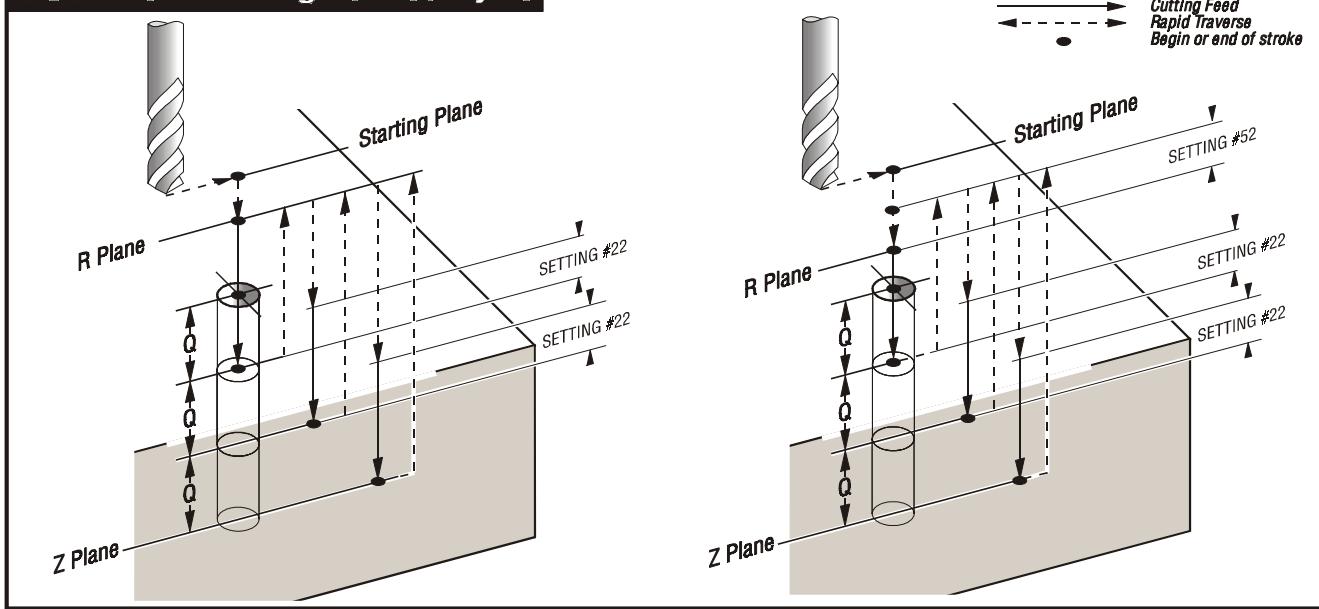
- X** Optional rapid X-axis command
- Y** Optional rapid Y-axis command
- Z** Bottom of hole (*Feed point of Z depth starting from R plane*)
- Q** The peck depth amount value, always incremental
- I** Optional size of first cutting depth (If Q is not used)
- J** Optional amount to reduce cutting depth each pass (If Q is not used)
- K** Optional minimum depth of cut (If Q is not used)
- P** The dwell time at the bottom of the hole
- R** Rapid position of the R plane (*Where you rapid, to start feed*)
- F** Feed Rate in inches (mm) per minute

This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. And the depth for each peck in this cycle will be the amount defined with Q. Then the tool will rapid up to the R plane after each peck and then back in for the next peck until Z depth is reached. Use G98 and G99 for the Z position clearance location for positioning between holes.

If I, J, and K are specified, a different operating mode is selected. The first pass will cut in by I, each succeeding cut will be reduced by amount J, and the minimum cutting depth is K.

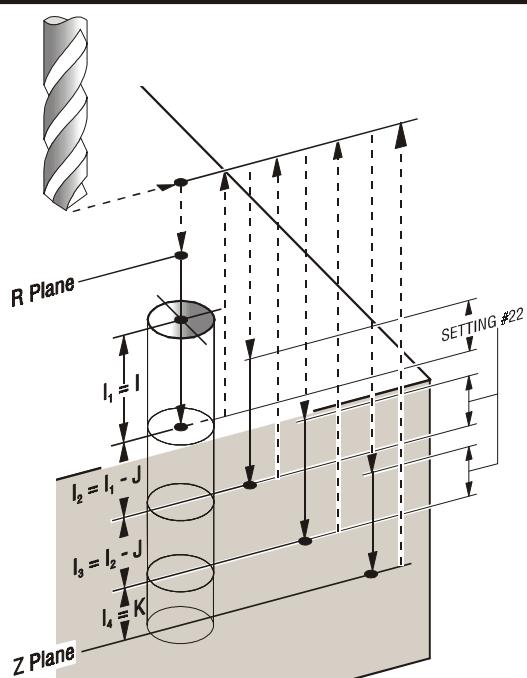
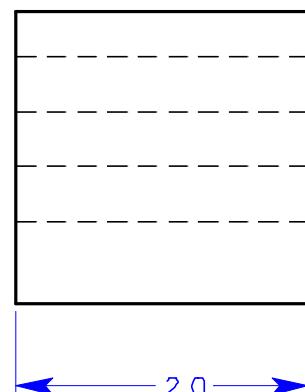
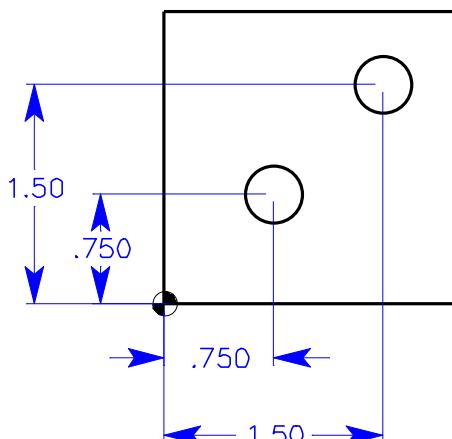
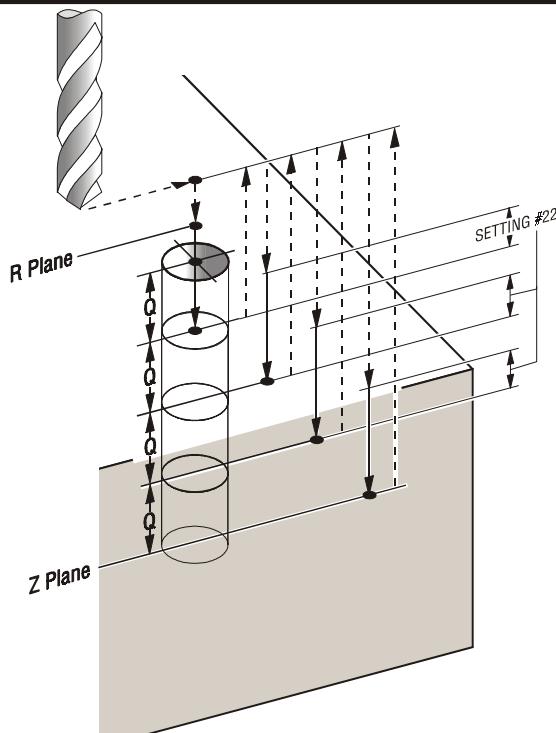
**Setting 22** specifies the distance the tool returns back to above from where it had left off at from the previous peck in a G83 canned cycle.

**Setting 52** also changes the way G83 works when it returns to the R plane. Most programmers set the R plane well above the cut to insure that the chip clear motion actually allows the chips to get out of the hole but this causes a wasted motion when first drilling through this "empty" space. If Setting 52 is set to the distance required to clear chips, the R plane can be put much closer to the part being drilled. When the clear move to R occurs, the Z will be moved above R by this setting.

**G83 Peck Drilling Canned Cycle**

(G83 using Q)

**N1 T3 M06****N2 G90 G54 G00 X.75 Y.75****N3 S1200 M03****N4 G43 H03 Z1. M08****N5 G83 G99 Z-2.125 Q.5 R.1 F8.****N6 X1.5 Y1.5****N7 G80 G00 Z1. M09****N8 G28 G91 Z0. M05****N9 M30**

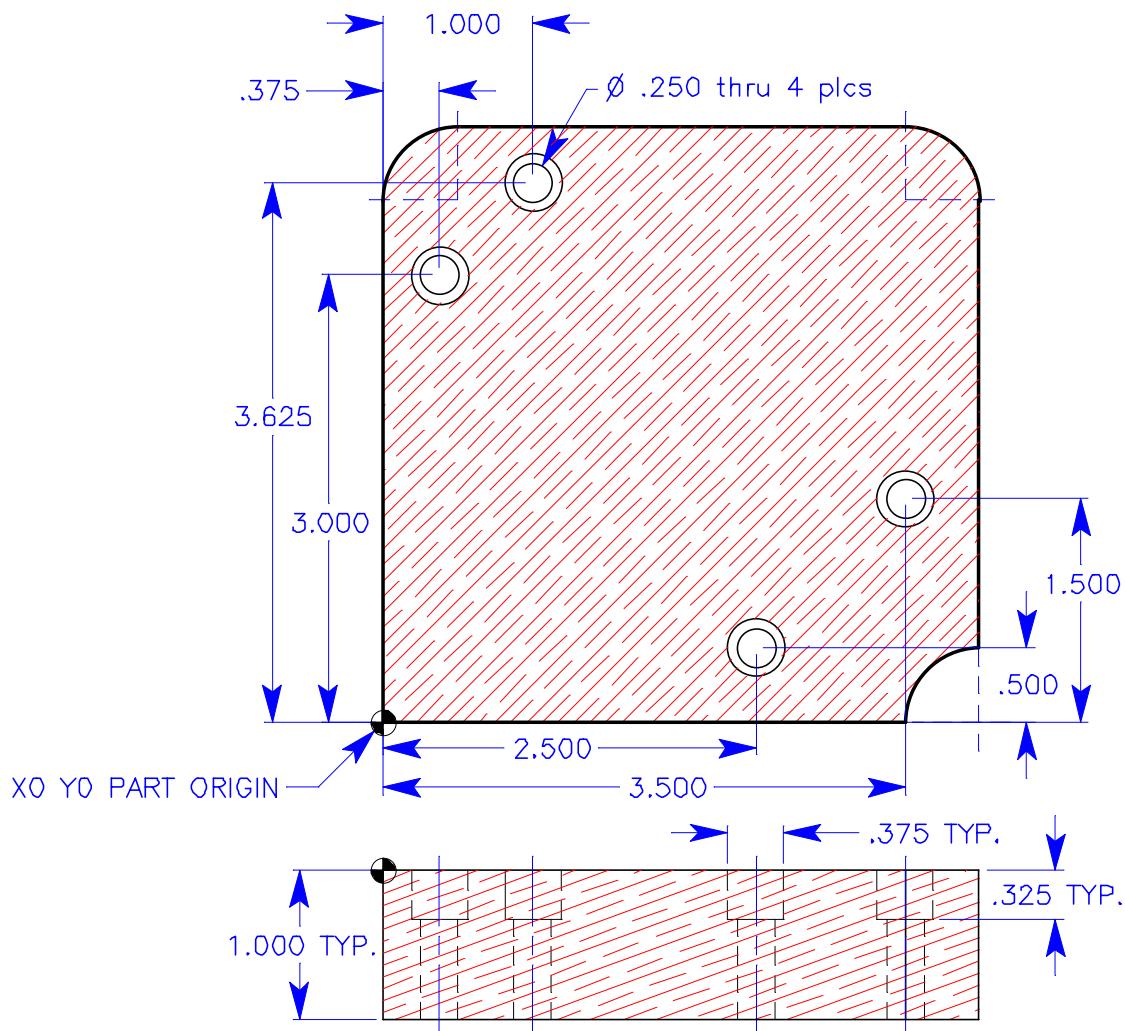
**G83 Peck Drilling with I, J & K options****G83 Normal Peck Drilling**

(G83 using I, J &amp; K)

**N1 T3 M06****N2 G90 G54 G00 X.75 Y.75****N3 S1200 M03****N4 G43 H03 Z1. M08****N5 G83 G99 Z-2.125 I.5 J.1 K.2 R.1 F8.****N6 X1.5 Y1.5****N7 G80 G00 Z1. M09****N8 G28 G91 Z0. M05****N9 M30**



## CANNED CYCLE EXERCISE #1



**TOOL 3 - is a 1/2 DIA. SPOT DRILL** - Drill To a depth of .200 to leave a .400 dia. spot for a counterbore chamfer. (use a G81 canned cycle)

Spindle Speed - S2000 (Surface Speed is 209.)

Feed Per Minute - F8.0 (Chip Load is .002)

**TOOL 4 is a 1/4 DIA. DRILL** - Drill thru to a 1.2 depth (use a G83 canned cycle with a .250 peck amount)

Spindle Speed - S2900 (Surface Speed is 190.)

Feed Per Minute - F14.5 (Chip Load is .0025)

**TOOL 5 is a 3/8 DIA. 4 FLT. END MILL** - Drill to a .325 depth (use a G82 canned cycle with a dwell for a 1/2 second)

Spindle Speed - S1750 (Surface Speed is 172.)

Feed Per Minute - F10.5 (Chip Load is .0015)



## CANNED CYCLE EXERCISE #1

**O00050 (CANNED CYCLE EXERCISE #1)**

T\_\_\_\_ M\_\_\_\_ (T3 - 1/2 DIA. SPOT DRILL)

G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_

S\_\_\_\_ M\_\_\_\_

G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ R\_\_\_\_ F\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

M\_\_\_\_ (Optional stop)

T\_\_\_\_ M\_\_\_\_ (T4 - 1/4 DIA. DRILL)

G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_

S\_\_\_\_ M\_\_\_\_

G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ Q\_\_\_\_ R\_\_\_\_ F\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

M\_\_\_\_ (Optional stop)

T\_\_\_\_ M\_\_\_\_ (T5 - 3/8 DIA. 4 FLT. END MILL)

G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_

S\_\_\_\_ M\_\_\_\_

G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ P\_\_\_\_ R\_\_\_\_ F\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

M\_\_\_\_ (Program stop)

**G84 TAPPING CANNED CYCLE**

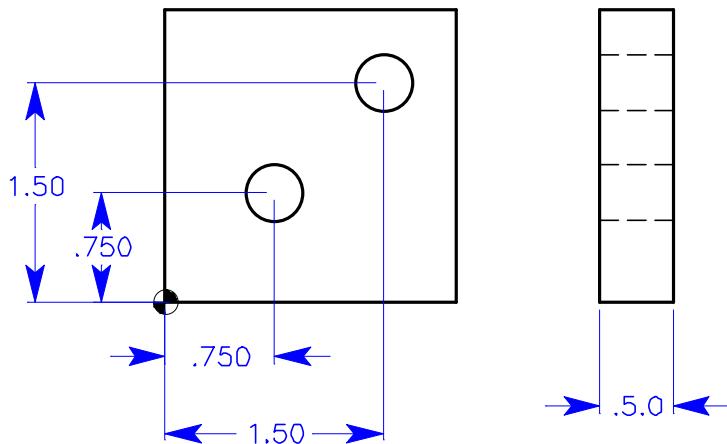
- X** Optional rapid X-axis command
- Y** Optional rapid Y-axis command
- Z** Position of bottom of hole
- R** Rapid position of the R plane
- F** Feed Rate in inches (mm) per minute

You don't need to start the spindle with a M03 for a tap that's using G84 because this cycle will turn on the spindle for you automatically and it will do it quicker.

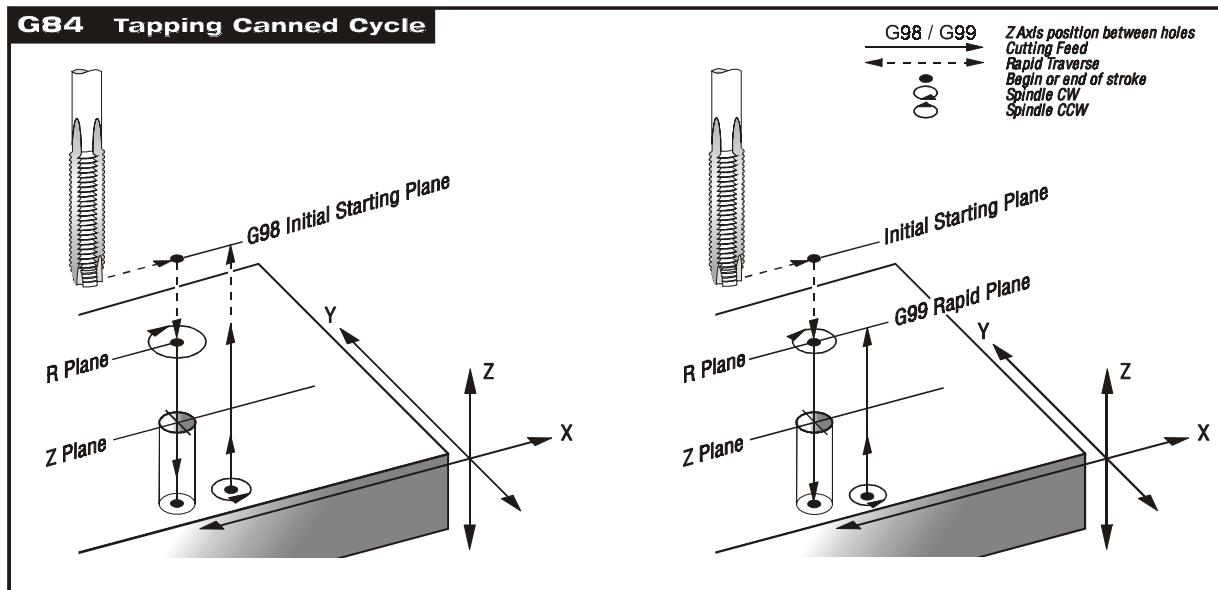
This G code is modal. Use G98 and G99 for the Z position clearance location.

If your using a spindle speed that's in low gear you may want to command M42 to force it into high gear, because most tapping (a smaller size tap) operations don't need the torque of low gear. And in high gear the tapping operation will perform quicker.

With Rigid Tapping, the ratio between feedrate and the spindle speed must be calculated for the thread pitch being cut. The calculation is 1/Threads Per Inch x rpm = tapping feedrate. Use the Haas calculator for the speed and feed numbers.



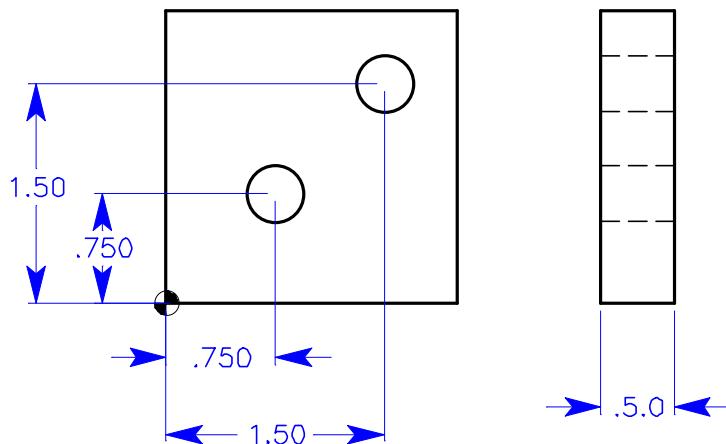
**N1 T4 M06 (1/2-20 tap)  
N2 G90 G54 G00 X.75 Y.75  
N3 S800 M42 (You don't need an M03. The G84 turns on spindle)  
N4 G43 H04 Z1. M08  
N5 G84 G99 Z-.65 R.1 F40.  
N6 X1.5 Y1.5  
N7 G80 G00 Z1. M09  
N8 G28 G91 Z0. M05  
N9 M30**



**G85 BORE IN AND BORE OUT CANNED CYCLE**

- X** Optional rapid X-axis command
- Y** Optional rapid Y-axis command
- Z** Bottom of hole (*Feed point of Z depth starting from R plane*)
- R** Rapid position of the R plane (*Where you rapid, to start feed*)
- F** Feed Rate in inches (mm) per minute

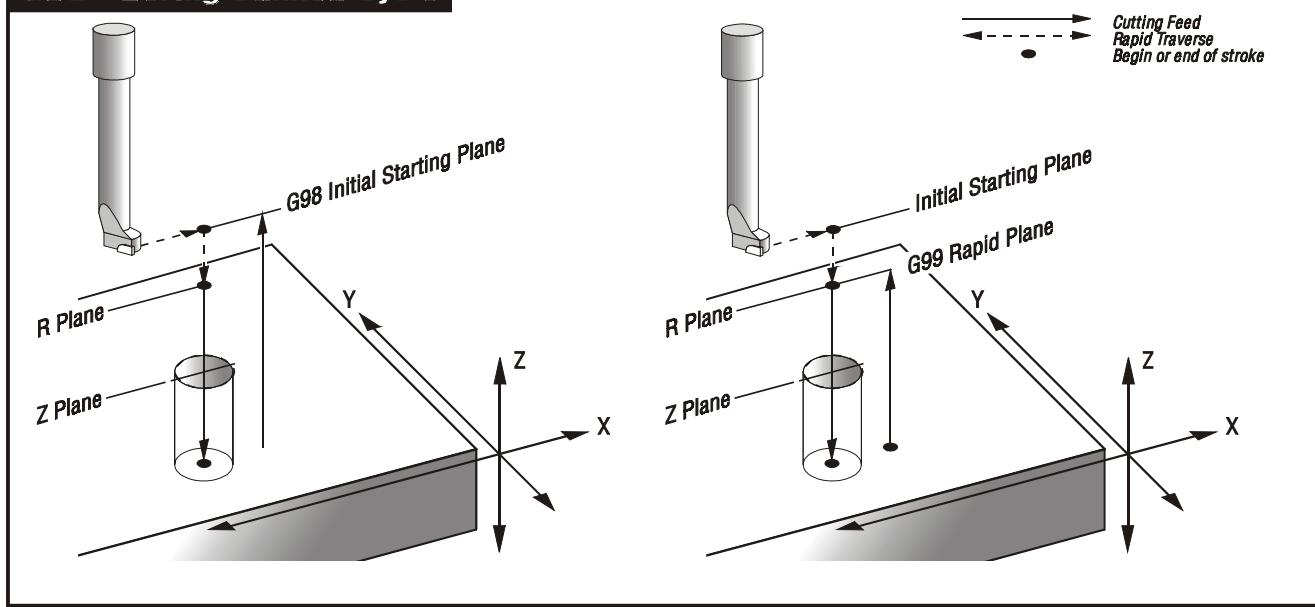
This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. Use G98 and G99 for the Z position clearance location for positioning between holes.



```

N1 T5 M06
N2 G90 G54 G00 X.75 Y.75
N3 S1200 M03
N4 G43 H05 Z1. M08
N5 G85 G99 Z-.55 R.1 F10.
N6 X1.5 Y1.5
N7 G80 G00 Z1. M09
N8 G28 G91 Z0. M05
N9 M30

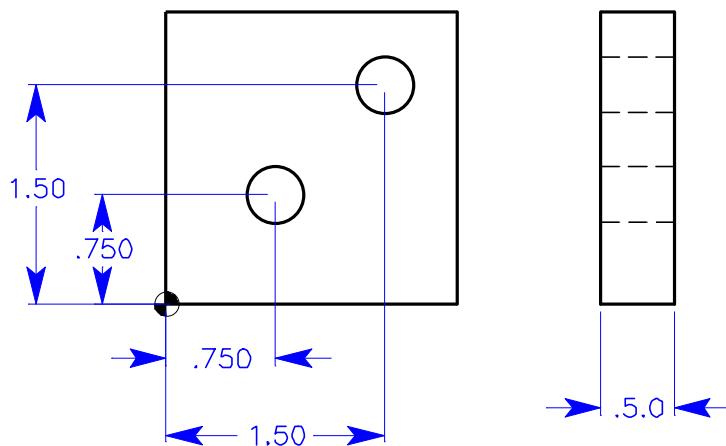
```

**G85 Boring Canned Cycle**

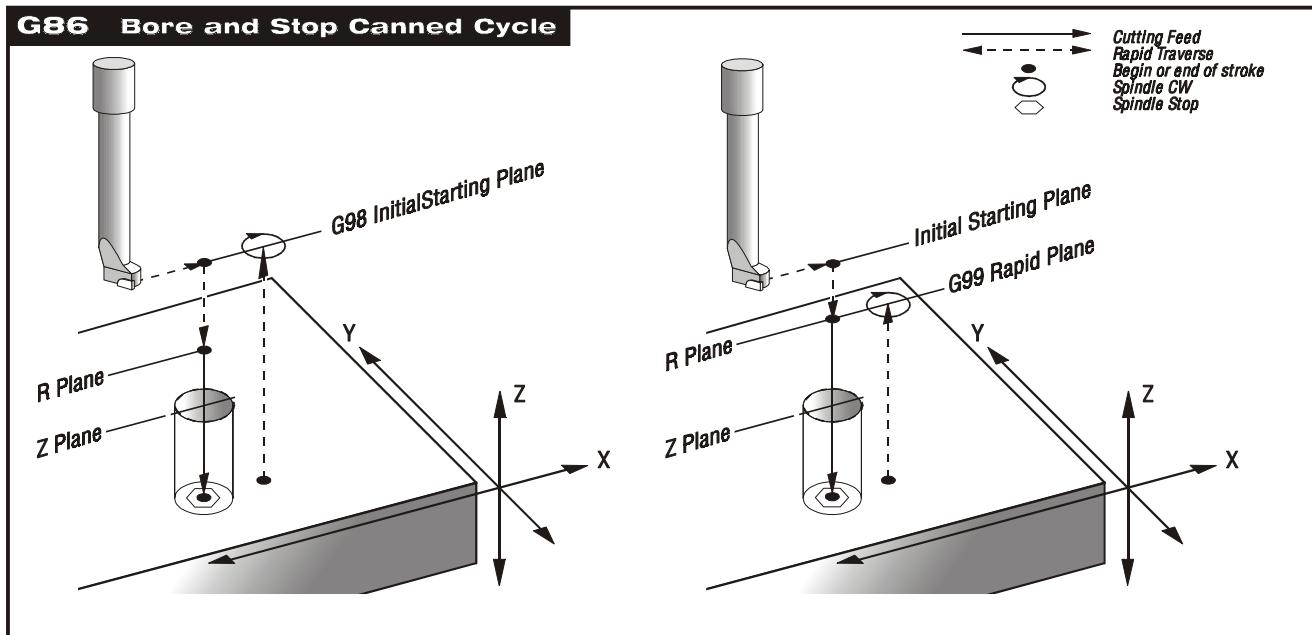
**G86 BORE IN, STOP, and RAPID OUT CANNED CYCLE**

- X Optional rapid X-axis command
- Y Optional rapid Y-axis command
- Z Bottom of hole (*Feed point of Z depth starting from R plane*)
- R Rapid position of the R plane (*Where you rapid, to start feed*)
- F Feed Rate in inches (mm) per minute

This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. Use G98 and G99 for the Z position clearance location for positioning between holes.



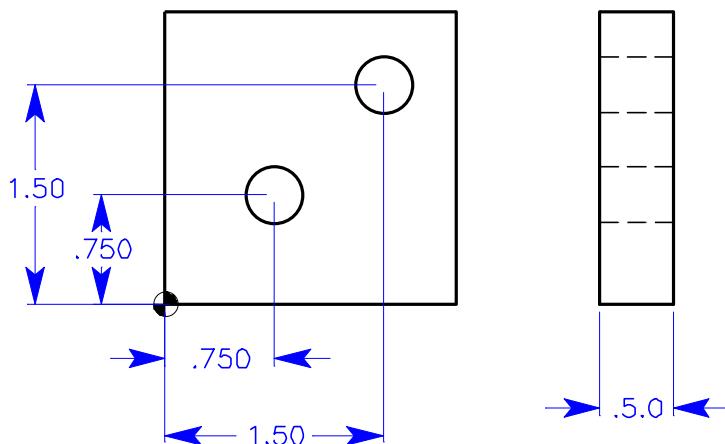
**N1 T6 M06**  
**N2 G90 G54 G00 X.75 Y.75**  
**N3 S1200 M03**  
**N4 G43 H06 Z1. M08**  
**N5 G86 G99 Z-.55 R.1 F10.**  
**N6 X1.5 Y1.5**  
**N7 G80 G00 Z1. M09**  
**N8 G28 G91 Z0. M05**  
**N9 M30**



**G87 BORE IN, and MANUAL RETRACT CANNED CYCLE**

- X Optional rapid X-axis command
- Y Optional rapid Y-axis command
- Z Bottom of hole (*Feed point of Z depth starting from R plane*)
- R Rapid position of the R plane (*Where you rapid, to start feed*)
- F Feed Rate in inches (mm) per minute

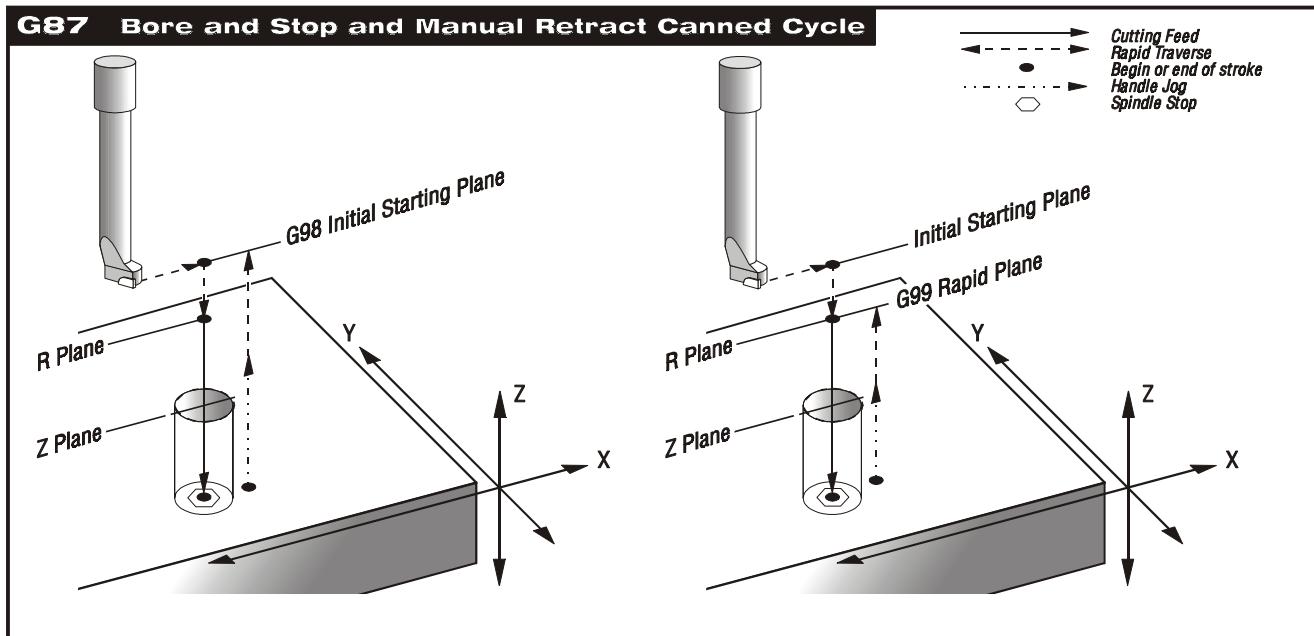
This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. Use G98 and G99 for the Z position clearance location for positioning between holes.



```

N1 T7 M06
N2 G90 G54 G00 X.75 Y.75
N3 S1200 M03
N4 G43 H07 Z1. M08
N5 G87 G99 Z-.55 R.1 F10.
N6 X1.5 Y1.5
N7 G80 G00 Z1. M09
N8 G28 G91 Z0. M05
N9 M30

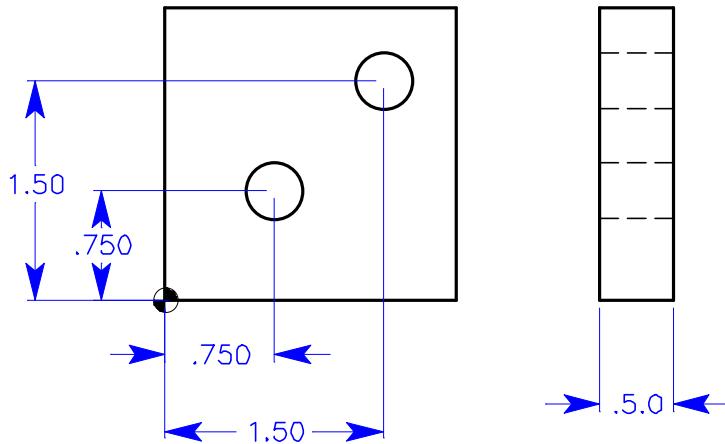
```



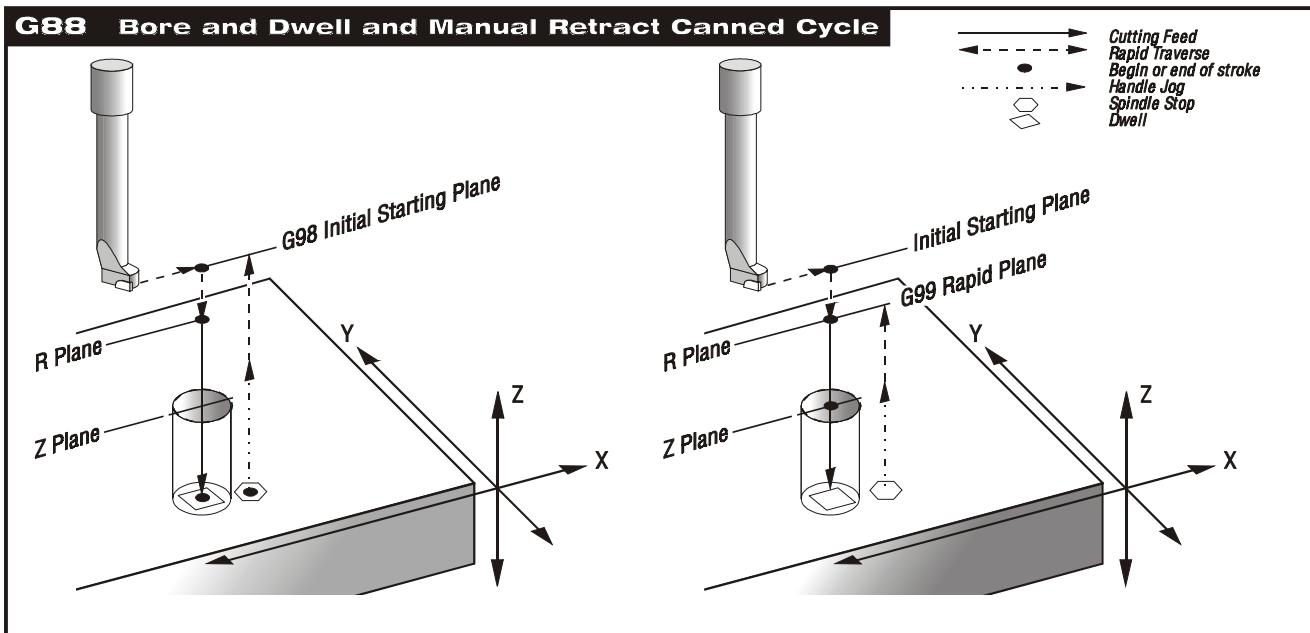
**G88 BORE IN, DWELL and MANUAL RETRACT CANNED CYCLE**

- X Optional rapid X-axis command
- Y Optional rapid Y-axis command
- Z Bottom of hole (*Feed point of Z depth starting from R plane*)
- P The dwell time at the bottom of the hole
- R Rapid position of the R plane (*Where you rapid, to start feed*)
- F Feed Rate in inches (mm) per minute

This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. Use G98 and G99 for the Z position clearance location for positioning between holes.



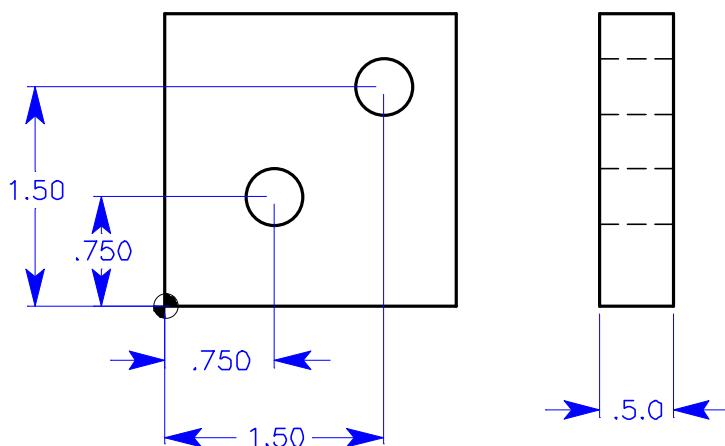
**N1 T8 M06**  
**N2 G90 G54 G00 X.75 Y.75**  
**N3 S1200 M03**  
**N4 G43 H08 Z1. M08**  
**N5 G88 G99 Z-.55 P1.5 R.1 F10.**  
**N6 X1.5 Y1.5**  
**N7 G80 G00 Z1. M09**  
**N8 G28 G91 Z0. M05**  
**N9 M30**



**G89 BORE IN, DWELL, BORE OUT CANNED CYCLE**

- X Optional rapid X-axis command
- Y Optional rapid Y-axis command
- Z Bottom of hole (*Feed point of Z depth starting from R plane*)
- P The dwell time at the bottom of the hole
- R Rapid position of the R plane (*Where you rapid, to start feed*)
- F Feed Rate in inches (mm) per minute

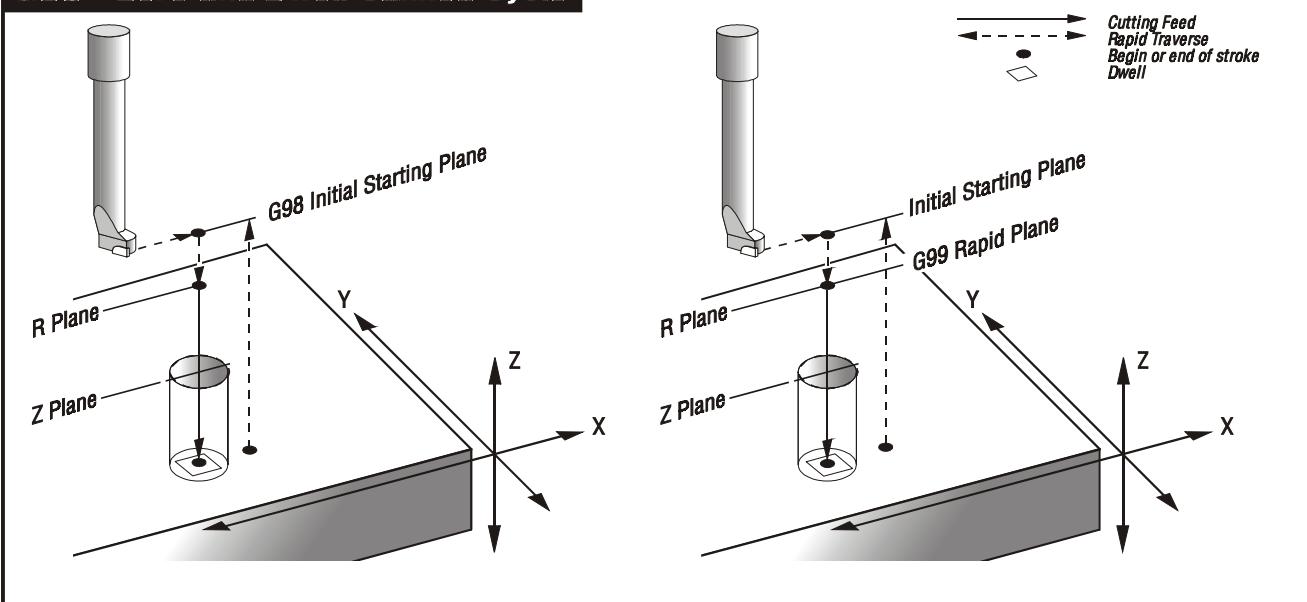
This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. A dwell in seconds.milliseconds is caused at the end of the Z-depth in this cycle which is defined with P. Use G98 and G99 for the Z position clearance location for positioning between holes.



```

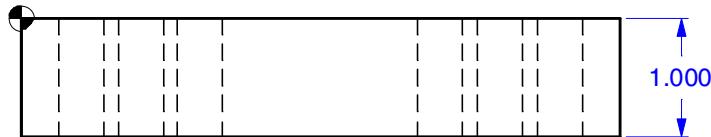
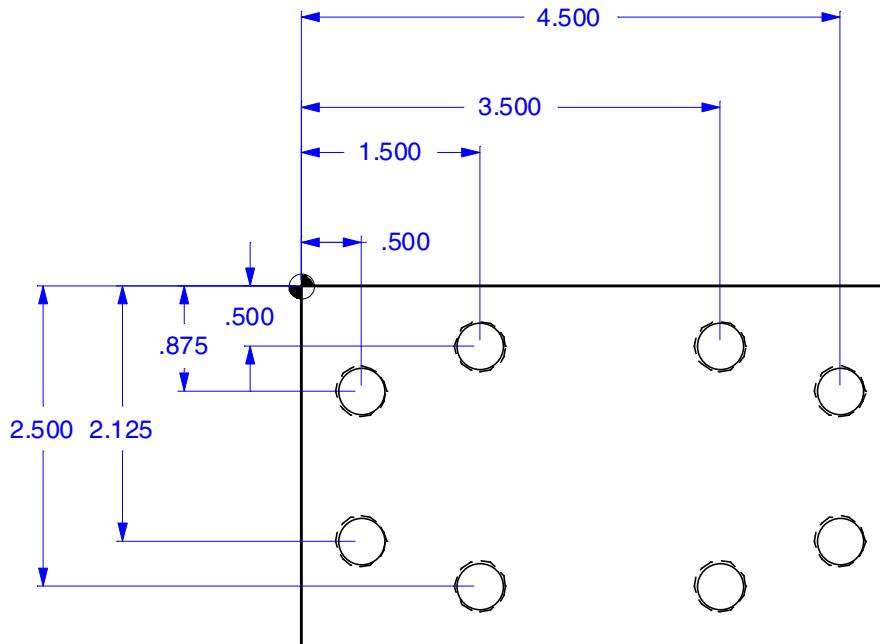
N1 T9 M06
N2 G90 G54 G00 X.75 Y.75
N3 S1200 M03
N4 G43 H09 Z1. M08
N5 G89 G99 Z-.5 P1.0 R.1 F10.
N6 X1.5 Y1.5
N7 G80 G00 Z1. M09
N8 G28 G91 Z0. M05
N9 M30

```

**G89 Bore and Dwell Canned Cycle**



## CANNED CYCLE EXERCISE #2



Use the HAAS CALC Display for Spindle speed and Feedrate  
**TOOL #1 - 90 DEG. 5/8 (.625) DIA. SPOT DRILL TO A .25 DEPTH**

Use G81 CANNED CYCLE

S \_\_\_\_\_

Surface Speed - 230. SFM Chip Load - .0025

F \_\_\_\_\_

**TOOL #2 - 27/64 (.4219) DIA. DRILL THRU TO 1.2 DEPTH**

Use G83 CANNED CYCLE with a Q.25 peck amount

S \_\_\_\_\_

Surface Speed - 230. SFM Chip Load - .003

F \_\_\_\_\_

**TOOL #3 - 1/2-13 TAP WITH RIGID TAP THRU TO 1.15 DEPTH**

Use G84 CANNED CYCLE

Use Tapping Calculator for FEEDRATE at 600 RPM

F \_\_\_\_\_



# **CANNED CYCLE EXERCISE #2**

# O00060 (CANNED CYCLE EXERCISE #2)

**G73 HIGH SPEED PECK DRILL CANNED CYCLE**

- X** Optional rapid X-axis command
- Y** Optional rapid Y-axis command
- Z** Bottom of hole (*Feed point of Z depth starting from R plane*)
- Q** The peck depth amount value, always incremental
- I** Optional size of first cutting depth (If Q is not used)
- J** Optional amount to reduce cutting depth each pass (If Q is not used)
- K** Optional minimum depth of cut (If Q is not used)
- P** The dwell time at the bottom of the hole
- R** Rapid position of the R plane (*Where you rapid, to start feed*)
- F** Feed Rate in inches (mm) per minute

This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. The depth of each peck in this cycle will be the amount defined with Q or using IJ & K or K & Q. The tool will pull back after each peck and then back in for the next peck until Z depth is reached. This cycle is a high speed peck cycle where the retract distance it pulls back after each peck is set by Setting 22. Use G98 and G99 for the Z position clearance location for positioning between holes.

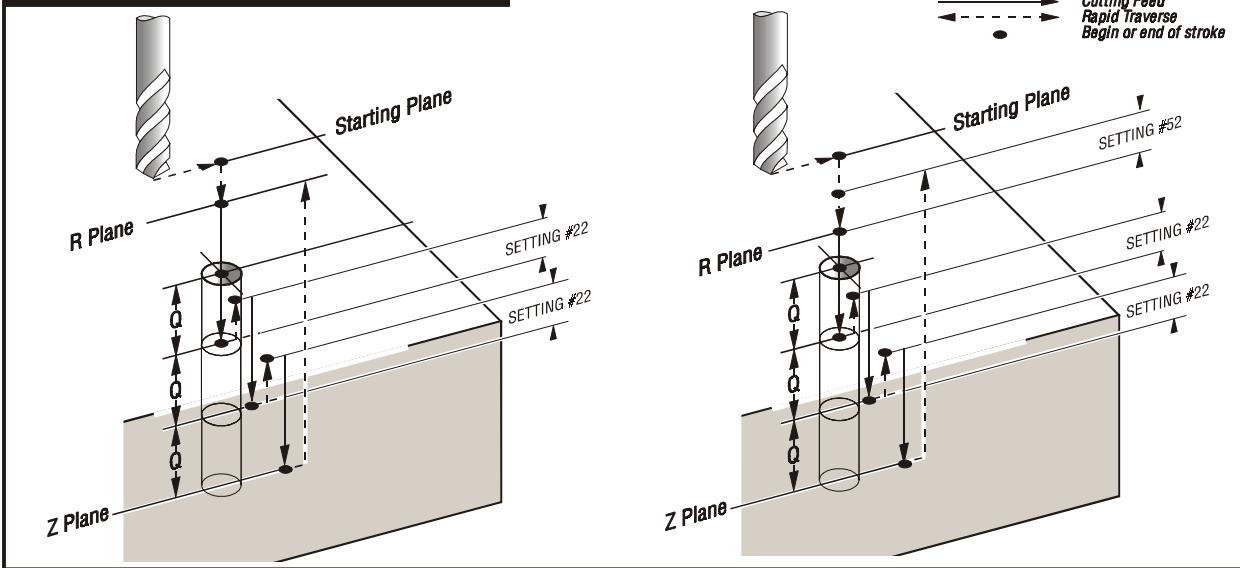
**SETTING 22** specifies the distance that the tool pulls back after each peck in a G73 canned cycle to break the chip.

**SETTING 52** also changes the way G73 works when it returns to the R plane when using K and Q. When the clear move to R occurs, the Z will be moved above R by this setting distance.

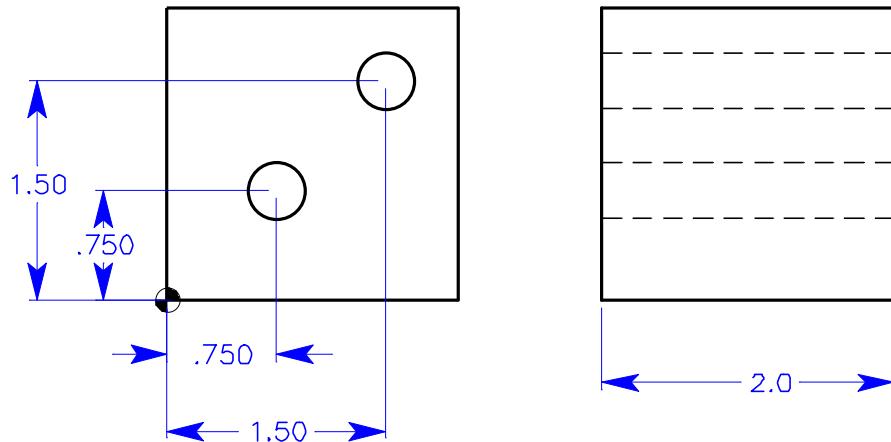


## G73 HIGH SPEED PECK DRILL CANNED CYCLE USING Q

### G73 Peck Drilling Canned Cycle



If Q is used, it specifies the amount each peck will be in a G73 canned cycle.



(G73 using Q)

**N1 T3 M06**

**N2 G90 G54 G00 X.75 Y.75**

**N3 S1200 M03**

**N4 G43 H03 Z1. M08**

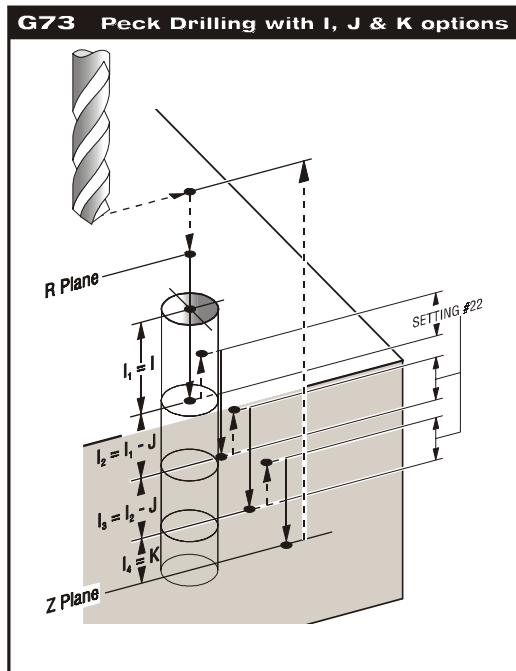
**N5 G73 G99 Z-2.15 Q.1 R.1 F8.**

**N6 X1.5 Y1.5**

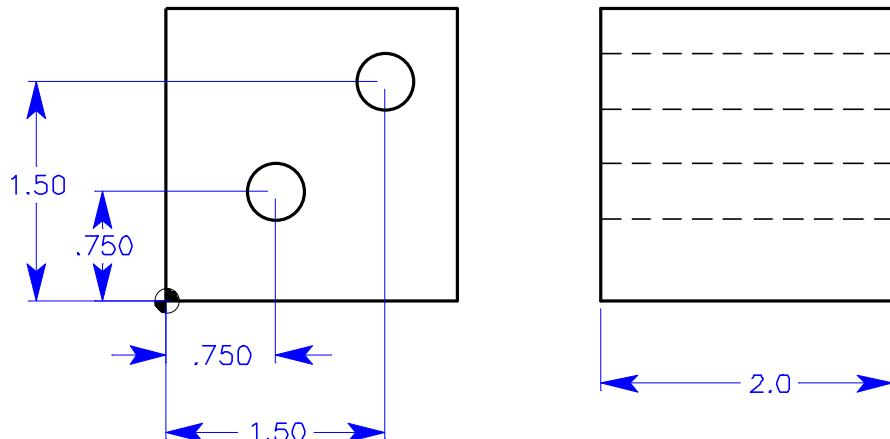
**N7 G80 G00 Z1. M09**

**N8 G28 G91 Z0. M05**

**N9 M30**

**G73 HIGH SPEED PECK DRILL CANNED CYCLE USING I,J,K**

If I, J, and K are specified, a different operating mode is selected. The first peck will be in by I, each succeeding peck will be reduced by the J amount, with the minimum peck being defined with K.



(G73 using I J K)

**N1 T3 M06**

**N2 G90 G54 G00 X.75 Y.75**

**N3 S1200 M03**

**N4 G43 H03 Z1. M08**

**N5 G73 G99 Z-2.15 I.5 J.1 K.2 R.1 F8.**

**N6 X1.5 Y1.5**

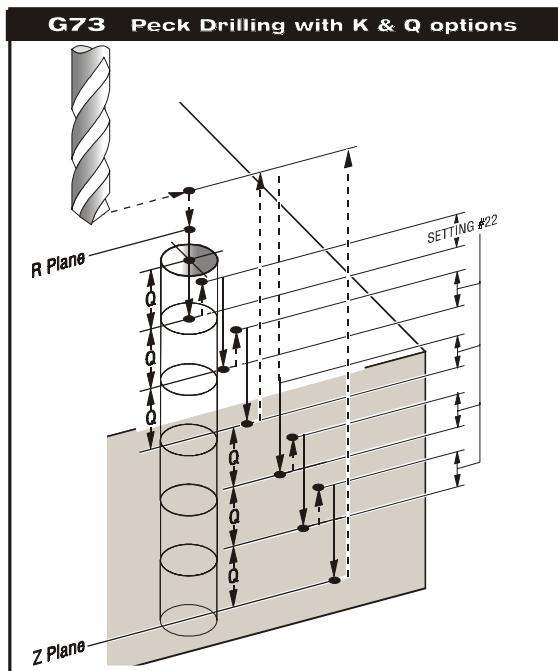
**N7 G80 G00 Z1. M09**

**N8 G28 G91 Z0. M05**

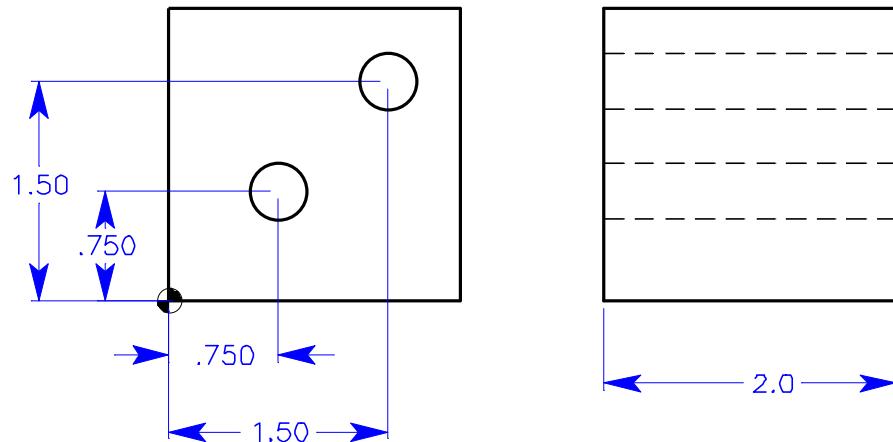
**N9 M30**



## G73 HIGH SPEED PECK DRILL CANNED CYCLE USING K & Q



If K and Q are both specified, a different operating mode is selected for this cycle. In this mode, after a number of pecks of Q distance down in the part totals up to the K amount, the tool is then returned to the R plane. This allows much faster drilling than G83 but still returns to the R plane occasionally to clear chips.



(G73 using K & Q)

**N1 T3 M06**

**N2 G90 G54 G00 X.75 Y.75**

**N3 S1200 M03**

**N4 G43 H03 Z1. M08**

**N5 G73 G99 Z-2.15 Q.2 K1. R.1 F8.**

**N6 X1.5 Y1.5**

**N7 G80 G00 Z1. M09**

**N8 G28 G91 Z0. M05**

**N9 M30**

**G74 REVERSE (LEFT HANDED) TAP CANNED CYCLE**

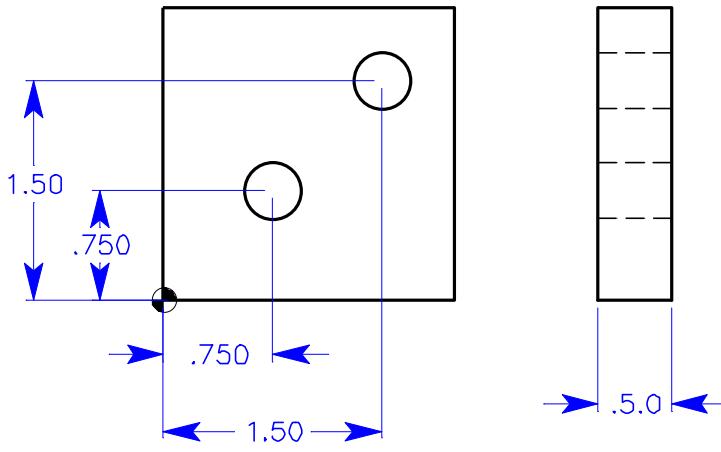
- X** Optional rapid X-axis command
- Y** Optional rapid Y-axis command
- Z** Position of bottom of hole
- R** Rapid position of the R plane
- F** Feed Rate in inches (mm) per minute

You don't need to start the spindle with a M04 for a tap that's using G74 because this cycle will turn on spindle for you automatically, and it will do it quicker.

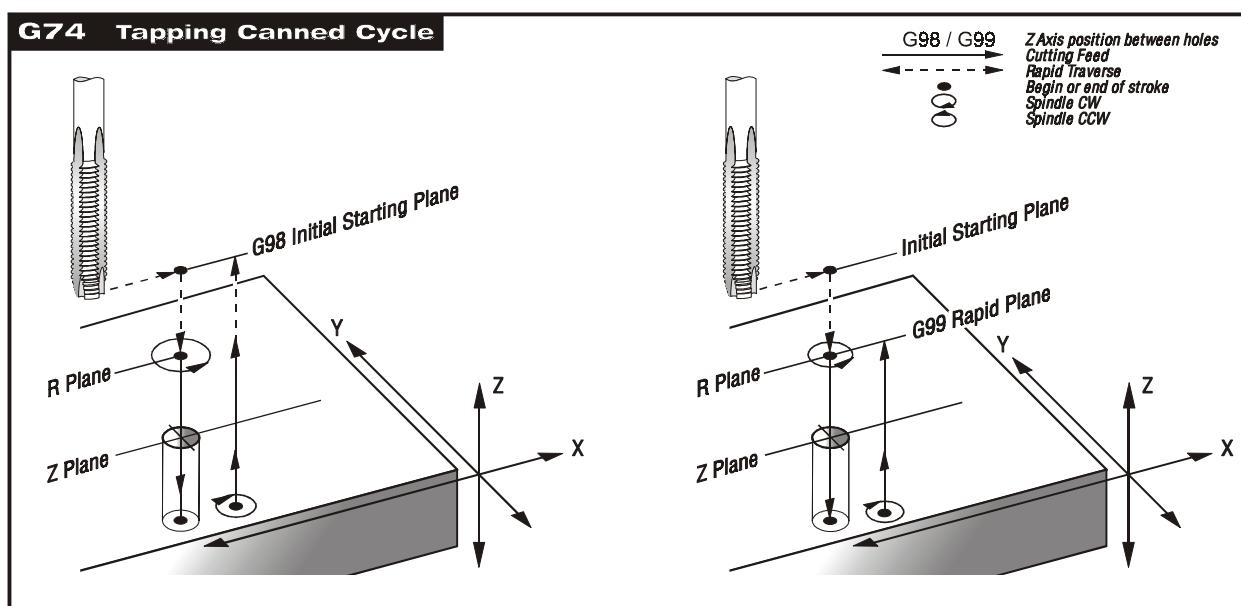
This G code is modal. Use G98 and G99 for the Z position clearance location.

If your using a spindle speed that's in low gear, you may want to command M42 to force it into high gear, because most tapping (a smaller size tap) operations don't need the torque of low gear. And in high gear the tapping operation will perform quicker.

When rigid tapping is used, the ratio between the feed rate and spindle speed must be precisely calculated for the thread pitch being cut.



**N1 T4 M06 (1/2-20 tap)  
N2 G90 G54 G00 X.75 Y.75  
N3 S800 M04 (You don't need an M04. G84 turns on spindle)  
N4 G43 H04 Z1. M08  
N5 G74 G99 Z-.6 R.1 F40.  
N6 X1.5 Y1.5  
N7 G80 G00 Z1. M09  
N8 G28 G91 Z0. M05  
N9 M30**

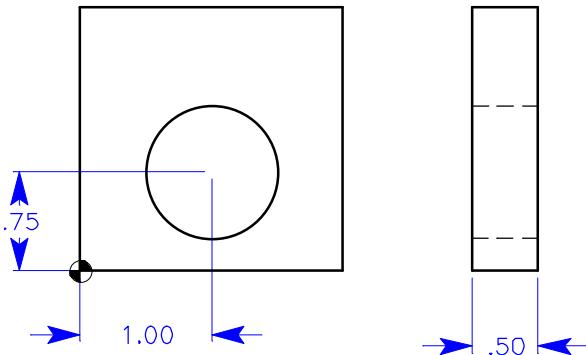


**G76 Fine Boring Canned Cycle**

- X** Optional rapid X-axis command
- Y** Optional rapid Y-axis command
- Z** Position of bottom of hole
- P** The dwell time at the bottom of the hole
- Q** The shift value, always incremental
- I** Optional X-axis shift value, if Q is not specified.
- J** Optional Y-axis shift value, if Q is not specified.
- R** Rapid position of the R plane (*Where you rapid, to start feed*)
- F** Feed Rate in inches (mm) per minute

The Q value distance will shift in the direction set by setting 27. If **Q** is not specified, the optional **I** and **J** values are used to determine the shift direction and distance. And a dwell at Z-depth in this cycle if a **P** command is defined.

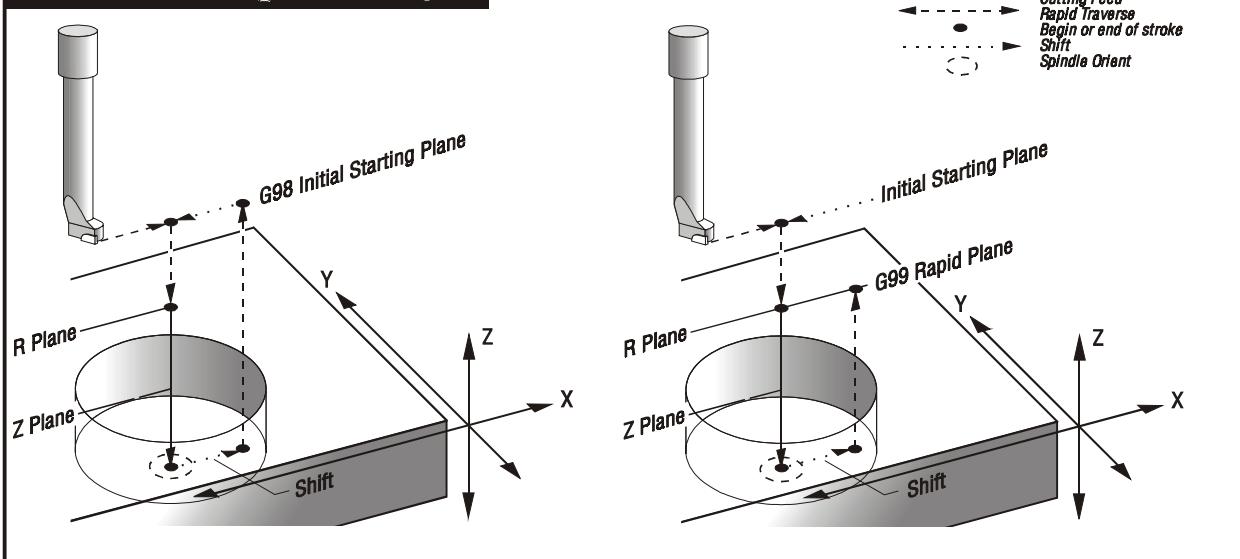
This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. Use G98 and G99 for the Z position clearance location for positioning between holes.



```

N1 T6 M06
N2 G90 G54 G00 X1.0 Y.75
N3 S1200 M03
N4 G43 H06 Z1. M08
N5 G76 G99 Z-.55 P.5 Q.02 R.1 F8
N6 G80 G00 Z1. M09
N7 G28 G91 Z0. M05
N8 M30

```

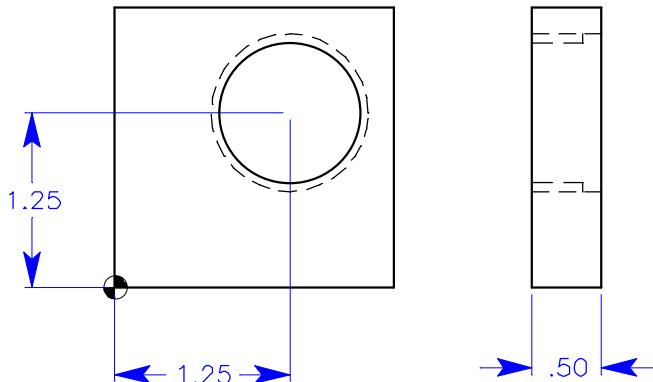
**G76 Fine Boring Canned Cycle**

**G77 BACK BORE CANNED CYCLE**

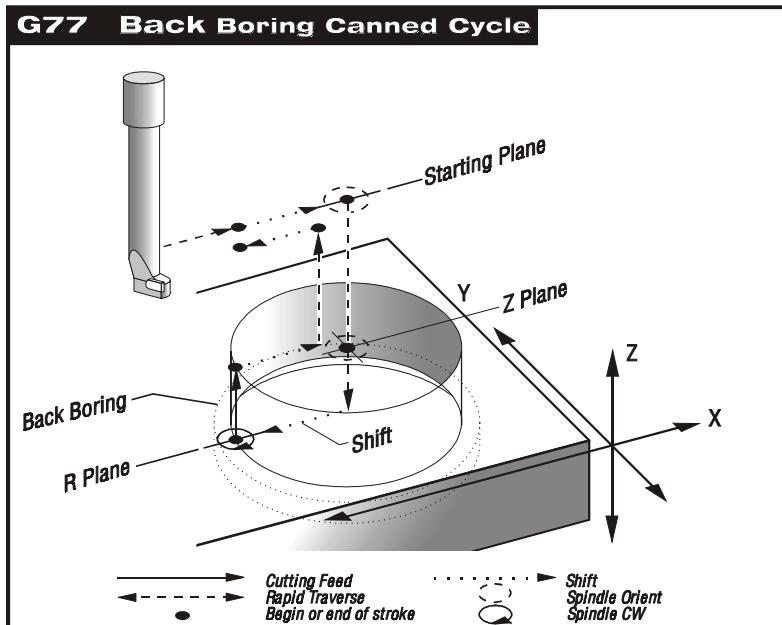
- X** Optional rapid X-axis command
- Y** Optional rapid Y-axis command
- Z** Feed location depth of back Counterbored hole
- R** Rapid position into the bore to the R plane
- Q** The shift value, always incremental
- I** Optional X-axis shift value, if Q is not specified.
- J** Optional Y-axis shift value, if Q is not specified.
- F** Feed Rate in inches (mm) per minute

The Q value distance will shift in the direction set by *setting 27*. If **Q** is not specified, the optional **I** and **J** values are used to determine the shift direction and distance.

This G code is modal so that it is activated every X and/or Y axis move, and it will rapid to that position and then cause this canned cycle to be executed again, until it's canceled. Use G98 and G99 for the Z position clearance location for positioning between holes.



**N1 T7 M06**  
**N2 G90 G54 G00 X1.25 Y1.25**  
**N3 S1200 M03**  
**N4 G43 H07 Z1. M08**  
**N5 G77 G99 Z-.35 R-.6 Q.1 F8.**  
**N6 G80 G00 Z1. M09**  
**N7 G28 G91 Z0. M05**  
**N8 M30**



This *cycle* will shift the X and/or Y axis prior to and after cutting in order to clear the tool while entering and exiting the part. The Q value shift direction is set with *setting 27*. If Q is not specified, the optional I and J values can be used to determine shift direction and distance. The tool will rapid down to the R clearance depth and feed up to the Z command depth for back counterbore.



## CANNED CYCLE AUXILIARY FUNCTIONS

### G98 Canned cycle initial Point Return

This G code is modal and changes the way canned cycles operate. With G98, the canned cycle will return to the initial starting point between each new X and/or Y location during the canned cycle command.

### G99 Canned cycle R Plane Return

This G code is modal and changes the way canned cycles operate. With G99, the canned cycle will return to the R plane between each new X and/or Y location during canned cycle command.

**N1 M06**

**N2 G90 G54 G00 X1.5 Y-.5 (1)**

**N3 S1200 M03**

**N4 G43 H01 Z1. M08** (Initial start point is Z1.0)

**N5 G83 G99 Z-1.2 Q.2 R.1 F8.** (G99 Position to R.1 rapid point)

**N6 X.5 Y-.75 (2)**

**N7 Y-2.25 (3)**

**N8 G98 X1.5 Y-2.5** (Position to initial start point after drilling hole) (4)

**N9 G99 X3.5 R-.4** (Position to rapid point after drilling hole) (5)

**N10 X4.5 Y-2.25 (6)**

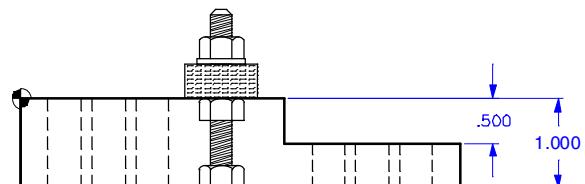
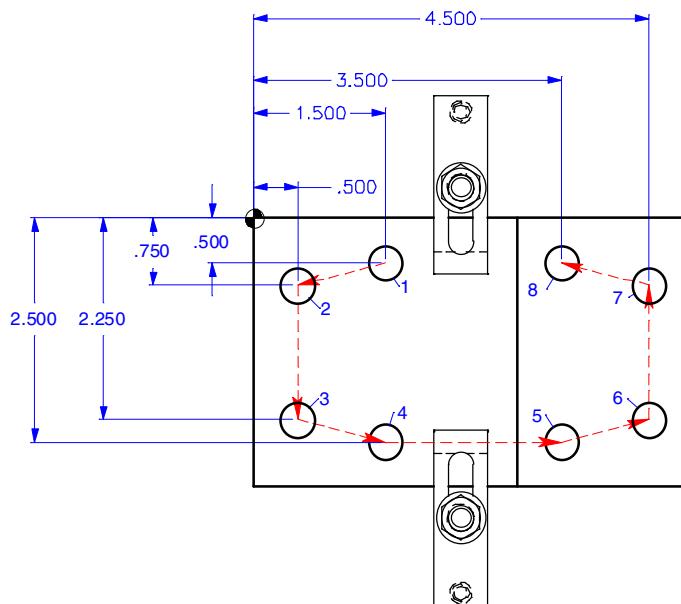
**N11 Y-.75 (7)**

**N12 X3.5 Y-.5 (8)**

**N13 G80 G00 Z1. M09**

**N14 G28 G91 Z0. M05**

**N15 M30**





## BOLT HOLE PATTERNS

There are three G codes that provide patterns usually used for bolt holes. These are G70, G71, and G72. These G codes must be used with one of the canned cycles G73, G74, G76, G77, or G81-G89. To define the beginning of an angle, the start is CCW from horizontal going from 0 to 360.0 degrees starting from Three o' clock. Using a minus value reverses the direction to CW for the starting position from Three o' clock.

### G70 Bolt Hole Circle

- I Radius
- J Starting angle (0 to 360.0 degrees CCW from three o' clock)
- L Number of holes evenly spaced around the circle

The tool must be positioned at the center of the circle either in a previous block or in the G70 block. G70 belongs to Group zero and thus is non-modal. For a G70 to work correctly, a canned cycle should be active so that at each of the positions, some type of drill or tap cycle is performed.

### G71 Bolt Hole Arc

- I Radius
- J Starting angle (Degrees CCW from three O' clock)
- K Angular spacing of holes (+ or -)
- L Number of holes

This G code is similar to G70 except that it is not limited at one complete circle. G71 belongs to Group zero and thus is non-modal. For a G71 to work correctly, a canned cycle should be active so that at each of the positions, some type of drill or tap cycle is performed.

### G72 Bolt Holes Along an Angle

- I Distance between holes
- J Angle of line (Degrees CCW from three o' clock)
- L Number of holes

This G code drills L holes in a straight line at the specified angle. It operates similarly to G70 and G71. G72 belongs to Group zero and thus is non-modal. For a G72 to work correctly, a canned cycle should be active so that at each of the positions, some type of drill or tap cycle is performed.



## BOLT HOLE CIRCLE

You must position to the center of the bolt hole circle in the XY axis coordinates, either in the G70 command block, or define the XY coordinates in a previous command block.

If an L0 is on the canned cycle line, the cycle will not execute that command until the control reads the next line for a G70 command, so as not to drill a hole in the center of a bolt hole circle. Or you can combine the drill cycle with the G70 on the same line, and it will also, not drill a hole in the center.

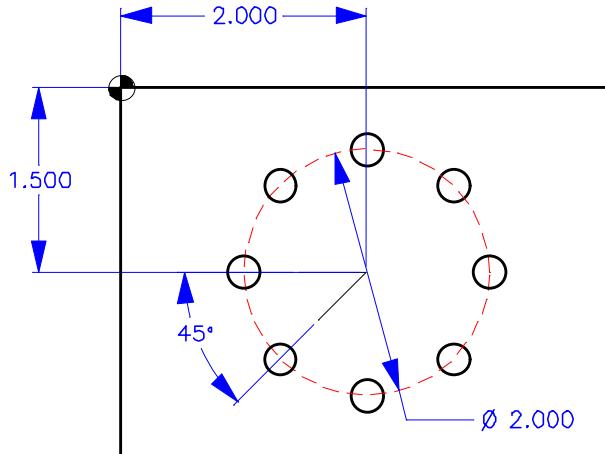
Be sure to use a decimal point with angle command codes.

### G70 BOLT HOLE CIRCLE COMMAND

I = Radius of the bolt circle

J = Starting angle from three o'clock

L = Number of holes (evenly spaced)



**N1 T1 M06**

**N2 G90 G54 G00 X2. Y-1.5** (Center position of bolt hole circle)

**N3 S1200 M03**

**N4 G43 H01 Z1. M08**

**N5 G81 G99 Z-.45 R.1 F8. L0** (L0 will not execute this command)

**N6 G70 I1.0 J0. L8** (until control reads next location.)

**N7 G80 G00 Z1. M09**

**N8 G28 G91 Z0. M05**

**N9 M30**



## BOLT HOLE ARC

You must position to the center of the bolt hole arc in the XY axis coordinates, either in the G71 command block, or define the XY coordinates in a previous command block.

If an L0 is on the canned cycle line, the cycle will not execute that command until the control reads the next line for a G71 command, so as not to drill a hole in the center of a bolt hole arc. Or you can combine the drill cycle with the G71 on the same line, and it will also not drill a hole in the center.

Be sure to use a decimal point with angle command codes.

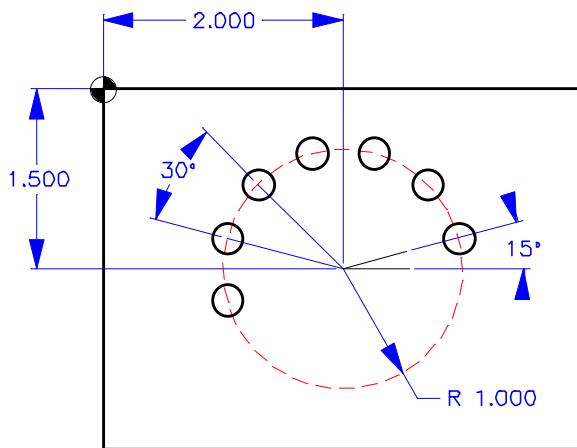
### G71 BOLT HOLE ARC COMMAND

I = Radius of the bolt circle

J = Starting angle from three o'clock

K = Angular spacing between holes

L = Number of holes (evenly spaced)



**N1 T2 M06**

**N2 G90 G54 G00 X2. Y-1.5** (Center position of bolt hole arc)

**N3 S1200 M03**

**N4 G43 H02 Z1. M08**

**N5 G81 G99 Z-.45 R.1 F8. L0** (L0 will not execute this command)

**N6 G71 I1.0 J15. K30. L7** (until control reads next location)

**N7 G80 G00 Z1. M09**

**N8 G28 G91 Z0. M05**

**N9 M30**



## BOLT HOLES ALONG AN ANGLE

If an L0 is on the canned cycle line, the cycle will not execute that command until the control reads the next line for the G72 command. Or you can combine the G72 with the drill cycle on the same line.

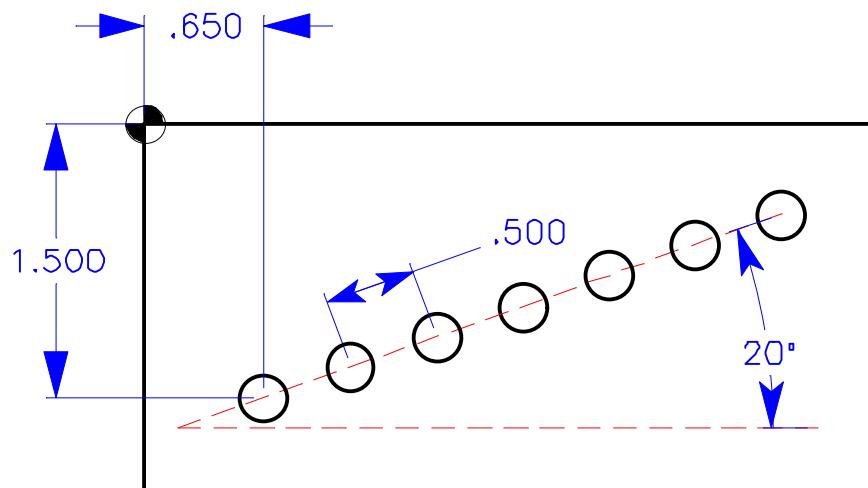
Be sure to use a decimal point with angle command codes.

### G72 BOLT HOLES ALONG AN ANGLE COMMAND

I = Distance between holes

J = Angle from three o'clock

L = Number of holes (evenly spaced)



**N1 T3 M06**

**N2 G90 G54 G00 X.65 Y-1.5** (Start position of bolt holes along an angle)

**N3 S1200 M03**

**N4 G43 H03 Z1. M08**

**N5 G81 G99 Z-.45 R.1 G72 I.5 J20. L7 F8.**

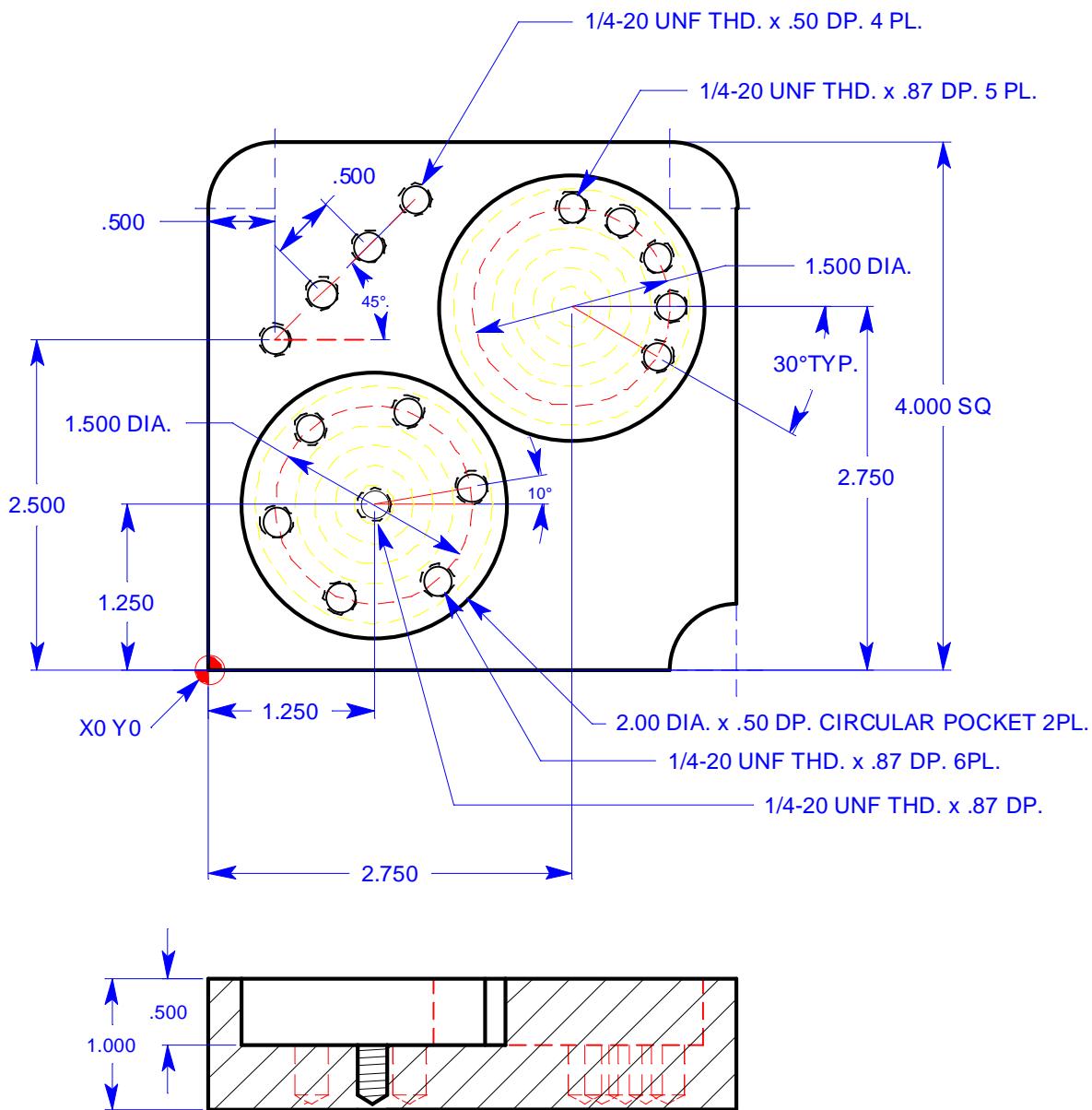
**N7 G80 G00 Z1. M09**

**N8 G28 G91 Z0. M05**

**N9 M30**



## CANNED CYCLE EXERCISE #3



Program a #7 (.201 Dia.) Drill at 2600 RPM at 14.5 feedrate to a Z-.95 drill tip depth for the Bolt Hole Circle and Bolt Hole Arc and change it to a Z-.6 drill tip depth for the Bolt Holes Along an Arc. Define the rapid plane .100 up from the bottom of the circular pocket of the *Bolt Hole Circle* and *Bolt Hole Arc* and change the rapid plane for the *Bolt Holes along an Angle* back to R.1 above Z zero top of part.

NOTE: If an L0 is on the canned cycle line, the cycle will not execute that command until the control reads the next location or the bolt hole command in the following line so as not to drill a hole in the center of bolt hole circle or bolt hole arc.



## CANNED CYCLE EXERCISE #3

**O00070 (CANNED CYCLE EXERCISE #3)**

**T6 M06 (#7 .201 DIA. CARBIDE STUB DRILL)**

**G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_ (X Y center position of bolt circle)**

**S\_\_\_\_ M\_\_\_\_**

**G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_ M\_\_\_\_ (position to Z.1 for the initial point)**

**G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ R\_\_\_\_ F\_\_\_\_ (G81 with G98 *initial point return*)**

**G\_\_\_\_ I\_\_\_\_ J\_\_\_\_ L\_\_\_\_ (bolt hole circle command)**

**X\_\_\_\_ Y\_\_\_\_ L\_\_\_\_ (BOLT HOLE ARC center, but no hole here)**

**G\_\_\_\_ I\_\_\_\_ J\_\_\_\_ K\_\_\_\_ L\_\_\_\_ (bolt hole ARC command)**

**G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_ Z\_\_\_\_ I\_\_\_\_ J\_\_\_\_ L\_\_\_\_ R\_\_\_\_ (Change the R  
plane for B.H.A.A.)**

**G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_ (Cancel canned cycle)**

**G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_**

**M30 (End of Program)**

**NOTE:** The circular pockets are .500 deep, so the rapid R-plane should be defined .100 above the bottom surface of pockets (R-.4), so as not to drill air from top of part down to bottom surface of pocket. But be careful, because the R-plane needs to be changed back to R.1 to the rapid plane above part surface when your positioning over to drill and tap Bolt Holes Along an Angle.



## SUBROUTINES

A subprogram is a separate program called up by another program. The use of subprograms can significantly reduce the amount of programming on some parts. Subroutines allow the CNC programmer to define a series of commands which might be repeated several times in a program and, instead of repeating them many times, they can be "called up" when needed. A subroutine call is done with M97 or M98 and a Pnnnn. The P code command identifies the O program number being used when executed with M98 or an N sequence number to identify the block where a local subroutine starts when executed with M97.

Local subroutines are called with an M97. This can be easier to use than the M98 because the subroutine is contained within the main program without the need to define a separate Onnnn program. With local subroutines, you define an M30 for the end of your main program portion. And after the M30 list all of your subroutines starting with a N sequence number to define the beginning of a local subroutine. And then end every subroutine with an M99 to send the control back to the very next line in the main program after the subroutine call.

This subroutine call causes the blocks in the subroutine to be executed just as if they were included in the main program. Then to return back to the main program you need to end the subroutines with an M99 which sends it back to the next line after the subroutine call in the main program.

Another important feature of a "subroutine call" is that the M97 and M98 block may also include an L (loop) or repeat count. If there is an Ln with the subroutine call it is repeated that number of times before the main program continues with the next block.

The most common use of subroutines is in the definition of a series of holes which may need to be center drilled, peck drilled, tapped, and/or chamfered. If a subroutine is defined that consists only of the X-Y position of the holes, the main program can define the canned cycles, and the hole locations can be called up in the subroutine to do each of the tool operations. Thus, the X-Y positions can be entered only once and used several times for each tool.

**O00080(PROGRAM CALLING SUB-PROGRAM WITH AN M98)****T1 M06 (5/8 DIA. SPOT DRILL)****G90 G54 G00 X0.5 Y-0.875****S1406 M03****G43 Z1. H01 M08****G81 G99 Z-0.25 R0.1 F7. (G81 Drilling Canned Cycle)****M98 P0081 (Call Sub-Program 00081)****T2 M06 (27/64 DIA. DRILL)****G90 G54 G00 X0.5 Y-0.875****S2082 M03****G43 Z1. H02 M08****G83 G99 Z-1.2 Q0.25 R0.1 F12.5 (G83 Peck Drilling Canned Cycle)****M98 P0081 (Call Sub-Program 00081)****T3 M06 (1/2-20 TAP)****G90 G54 G00 X0.5 Y-0.875****S750 (G84 will turn on the spindle, so no M03 is needed)****G43 Z1. H03 M08****G84 G99 Z-1.15 R0.1 F37.5 (G84 Tapping Canned Cycle)****M98 P0081 (Call Sub-Program 00081)****G28 G91 Y0.****M30 (End of Main Program)**

(Sub-routine)  
(Listing all the hole positions)

**O0081**

**X1.5 Y-.5** To return back to the  
main program you need  
to end the sub-program  
**X3.5** with an M99 which  
**X4.5 Y-.875** sends it back to the next  
**Y-2.125** line after the sub-pro-  
**X3.5 Y-2.5** gram call in the main  
**X1.5** program.  
**X.5 Y-2.125**

**G80 Z1.0 M09****G28 G91 Z0. M05**

**M99** (End of sub-program and  
returns back to the next line  
in the main program after  
the M98 sub-program call.)

**O00090 (PROGRAM CALLING LOCAL SUB-ROUTINE WITH M97)****T01 M06 (5/8 DIA. SPOT DRILL)****G90 G54 G00 X0.5 Y-0.875****S1406 M03****G43 H01 Z1. M08****G81 G99 Z-0.25 R0.1 F7. (G81 Drilling Canned Cycle)****M97 P0091 (Call Local Sub-Routine N0091)****T02 M06 (27/64 DIA. DRILL)****G90 G54 G00 X0.5 Y-0.875****S2802 M03****G43 H02 Z1. M08****G83 G99 Z-1.2 Q.25 R0.1 F12.5 (G83 Peck Drilling Canned Cycle)****M97 P0091 (Call Local Sub-Routine N0091)****T03 M06 (1/2-13 TAP)****G90 G54 G00 X0.5 Y-.875****S750 (G84 will turn on the spindle so, M03 is not needed)****G43 H03 Z1. M08****G84 G99 Z-1.15 R.1 F37.5 (G84 Tapping Canned Cycle)****M97 P0091 (Call Local Sub-Routine N0091)****G28 G91 Y0.****M30 (End of program)****N0091 (Local subroutine example listing all the hole locations)****X1.5 Y0.5****X3.5****X4.5 Y-0.875****Y-2.125****X3.5 Y-2.5****X1.5****X0.5 Y-2.125****G80 G00 Z1. M09****G28 G91 Z0. M05****M99 (End of local sub-routine and returns back to the next line in the main program after the M97 sub routine call.)**



## GENERAL PURPOSE POCKET MILLING

### G150 General Purpose Pocket Milling

- X** X position of starting hole (not needed if you're already at that location)
- Y** Y position of starting hole (not needed if you're already at that location)
- P** Subprogram call number defining the geometry of pocket
- I** X-axis shift over cut increment (use I or J not both)
- J** Y-axis shift over cut increment (use I or J not both)
- K** Finishing cut allowance
- G41** Cutter Comp. Left or **G42** Cutter Comp. Right of programmed path
- D** Cutter geometry offset selection number
- Z** Final depth of pocket
- R** R plane position
- Q** Incremental Z-axis depth down, starting from R plane
- F** Feed rate

This G code provides for general purpose pocket milling. The shape of the pocket to be cut must be defined by a series of motions within a subroutine. A series of motions in either the X or Y-axis will be used to cut out the specified shape followed by a finishing pass to clean up the outer edge. Only one of either I or J must be specified. If I is used, the pocket is cut from shifting over in the X-axis and the strokes will be along the Y-axis. If J is used, the pocket is cut from shifting over in the Y-axis and the strokes will be along the X-axis. I and J must be positive numbers. The finishing pass is K and the K command must be a positive number. There is no finishing pass in the Z depth with the K.

Multiple depth of cuts on pocket can be defined to control the cut depth. The feed down to the final Z depth starts from the R plane feeding down by the Q amount, for each pass, until the Z depth is reached. Q must be positive. If an L count is specified on the G150 command line with a G91 and an X and/or Y value, the entire block is repeated again at an incremental X or Y distance L number of times.

The subroutine must define a closed loop area by a series of G01, G02, or G03 motions in X and Y axes and must end with an M99. G codes G90 and G91 can also be used in the subroutine to select absolute or incremental. Any codes other than G, I, J, R, X, or Y are ignored in the subroutine. This subroutine must consist of less than 20 strokes.

Pocket milling may need to begin from a hole which has been previously drilled to the Z depth, in order to plunge the tool on entry of the pocket on the G150. You specify this hole location with X and Y in a previous block or in the G150 command line.



The first motion in the SUBPROGRAM should move from this clear hole to the starting point of the pocket geometry. The final motion in the SUBPROGRAM should close the loop at the same point where you began the pocket geometry, ie., in the example on the following page, the start point of the G150 line is X3.25 Y4.5 and the first move of the of the sub is Y7.0. Therefore the end of the sub must return to X3.25 Y7.0.

If K is specified, the roughing cuts will cut inside the programmed pocket size by K amount. The finishing pass will follow along the pocket geometry edge and is done at the full pocket depth.

**O00100 (G150 POCKET MILLING EXAMPLE)**

**N1** (Tool 1 drills a clearance hole in pocket for end mill to plunge in.)

**T1 M06 (1/2 DIA. DRILL)**

**G90 G56 G00 X3.25 Y4.5** (Starting location for pocket milling routine.)

**S2000 M03**

**G43 H01 Z1.0 M08**

**G83 Z-2.0 R.1 Q0.5 F10.**

**G80 G00 Z1.0 M09**

**G28 G91 Z0 M05**

**N2** (G150 pocket milling doing four cut depths to Z depth)

**T2 M06 (T2 1/2 DIA. E. M.)**

**G90 G56 G00 X3.25 Y4.5** (Starting location for pocket milling routine.)

**S2000 M03**

**G43 Z1.0 H02 M08**

**G01 Z0 F20.** (Feed down to start point in Z axis for increment Q depth)

**G150 P101 J0.4 (or I.4) K0.02 G41 D02 Z-1. Q0.25 R.1 F12.**

**G01 G40 X3.25 Y4.5** (Be sure to cancel cutter comp.)

**G00 Z.1 M09**

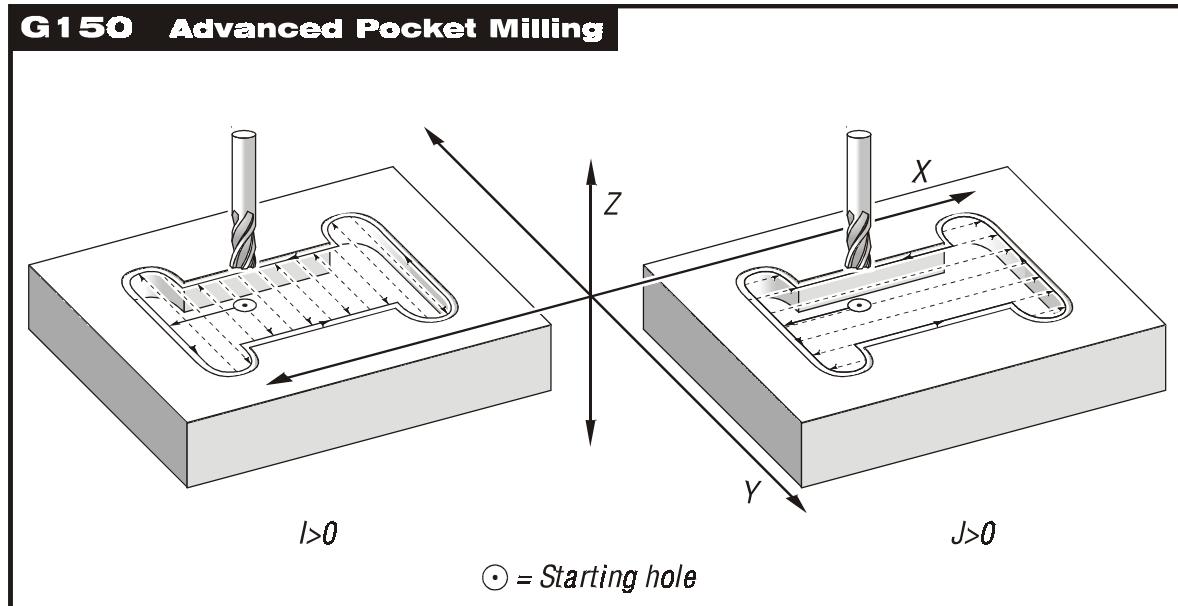
**G28 G91 Y0 Z0 M05**

**M30**

(Sub-routine for the pocket geometry of the G150 pocket milling cycle is on the next page, which is called with P101 on the G150 line which is looking for program O00101.



(SUBPROGRAM)  
**O00101** (G150 POCKET GEOMETRY)  
**G01 Y7.**  
**X1.5**  
**G03 Y5.25 R0.875**  
**G01 Y2.25**  
**G03 Y0.5 R0.875**  
**G01 X5.**  
**G03 Y2.25 R.875**  
**G01 Y5.25**  
**G03 Y7. R0.875**  
**G01 X3.25**  
**M99** (Return to main program)



## G150 NOTES

1. Position to the starting point inside the pocket with end mill. Do not have cutter overlap on any line of the pocket geometry with the entry of end mill.
2. In the G150 command line, it calls up a subprogram with a P command (P1234) which is calling up a separate program (O1234) that defines in it, the geometry of a pocket. This pocket geometry must be defined in 20 moves or less.
3. You can also define an island within the pocket, but remember, you only have 20 moves to define your G150 pocket command.

(G150 notes continued)



## MORE G150 NOTES

4. This pocket geometry "cannot" be after the M30 of a program like the way an M97 is used. It must be in a separate program, which is called up by a G150 with the P command the defines that program number which contains the pocket geometry.
5. When you're defining the pocket geometry in the subprogram. You define it from the starting point that is a location somewhere inside pocket. And then you define a move onto the side of the pocket and around pocket geometry. Close the loop at where you began defining the pocket, end the subprogram there with an M99. Don't try to move back to the starting point where you began.
6. Define the path of the "pocket geometry" in the direction that you wish to cut with your end mill. Define the pocket path clockwise to conventional cut using G42, or counter clockwise to climb cut using G41. And this path will be reversed if you position onto an island inside the pocket to either climb cut or conventional cut with the end mill.
7. When you position onto and off of an island inside a pocket, and you're using cutter compensation (G41 or G42), you will either be left or right of the line, the amount of the cutter compensation in your offset register. And if you position onto the island on the same line or location value as you position off the island, than you will leave a large scallop the size of the radius of your cutter offset. So to eliminate this scallop condition you need to overlap your entry and exit moves onto and off of the island. You need to overlap these entry and exit moves at least the radius amount, of the cutter comp offset being used. Overlapping a little more than the radius amount of the cutter offset being used is usually preferred.
8. The Q value must be defined in a G150 even if your only doing one pass to the Z final depth. And the Q amount is defined from the R plane. So if you only want one pass, and your final depth is Z-.25 and your starting the pocket at R.1 above the part. Than your Q value down will be Q.35 for doing one pass to the final depth. Otherwise if you give it a Q.25 the first depth would be calculated from .1 down the amount of Q.25 which would be at a Z-.15 into the pocket, for a first pass and then it would take another pass at the final depth of Z-.25 for a second pass.
9. If you should execute a G150 pocket milling command in a program and you have an invalid move in your subprogram geometry, the control will alarm out on the G150 command line in the main program. But the error is somewhere within the subprogram, and you need to some how locate the error in the subprogram. Then to locate this error, run the subprogram separately in graphics to locate this error in the subprogram geometry. Being able to run this subprogram in graphics you need to enter a feedrate command at the beginning of the sub-program. Now you will be able to troubleshoot subprogram in graphics. After your done fixing the moves in your subprogram, take the feedrate command back out of subprogram. And verify the fix in graphics of the G150 command, in the main program.

JUNE 2000

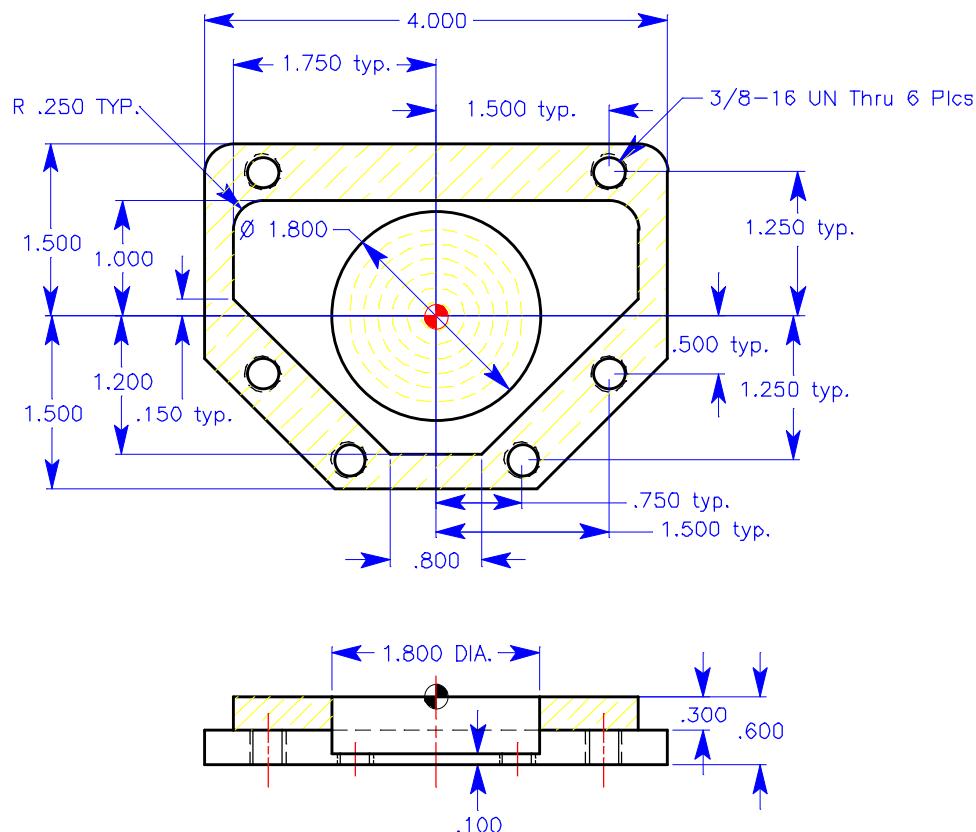
**VF/H5 Series**  
WORKBOOK

PROGRAMMING





## FINAL EXERCISE



**TOOL #1 - 1.00 dia. 4 ftl. End Mill** - finish mill contour at a .3 depth, 1 pass. 250 FPM - .003 chip per tooth S\_\_\_\_\_ F\_\_\_\_\_

**TOOL #2 - 1 3/4 dia. Insert Drill** - rough drill circular pocket to .480 depth using a G81 canned cycle. 250 FPM - .005 chip per tooth  
S\_\_\_\_\_ F\_\_\_\_\_

**TOOL #3 - 5/8 dia. 4 ftl End Mill** - finish 1.800 dia. x .5 depth circular pocket. 250 FPM -.0025 chip per tooth. S\_\_\_\_\_ F\_\_\_\_\_

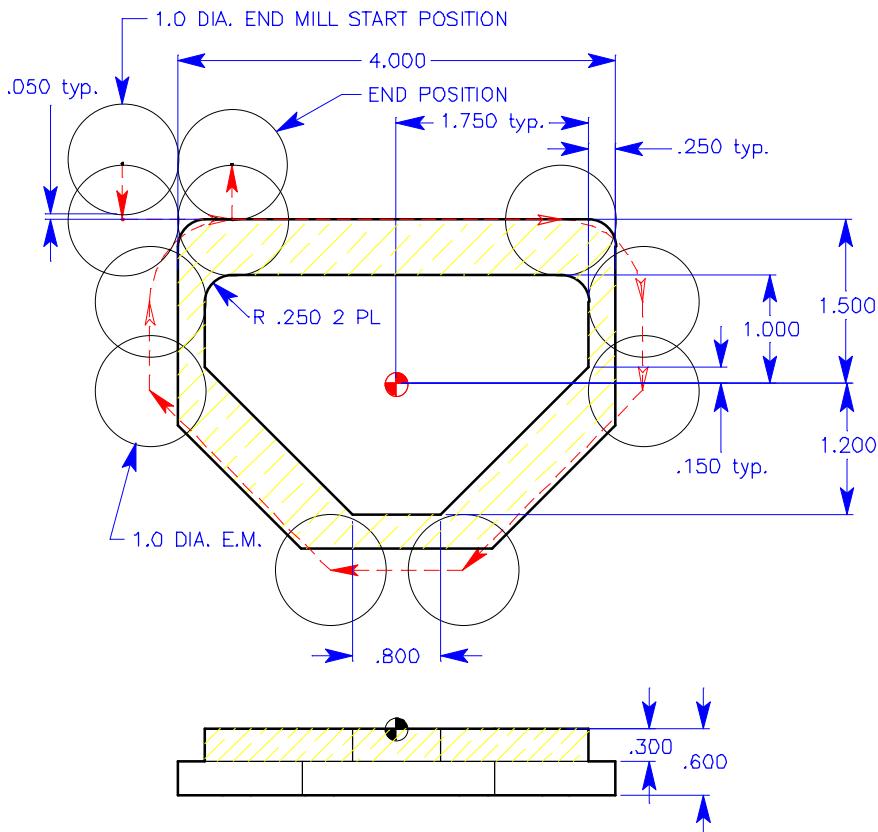
**TOOL #4 - 1/2 dia. Spot Drill** - drill 6 holes .2 depth using G82 canned cycle. 200 FPM - .0035 chip per tooth S\_\_\_\_\_ F\_\_\_\_\_

**TOOL #5 - 5/16 dia. Drill** - drill 6 holes thru using G73 canned cycle 200 FPM - .003 chip per tooth S\_\_\_\_\_ F\_\_\_\_\_

**TOOL #6 - 3/8-16 Tap** - tap 6 holes thru using G84 with a machine that has rigid tapping. Tap at 600 RPM, S\_\_\_\_\_ F\_\_\_\_\_



## FINAL EXERCISE



Be sure to position the cutter to the center of the tool at least half the cutter diameter off of the part surface before activating cutter comp.

### TOOL #1 - 1.00 dia. 4 flt. End Mill.

Climb cut a finish pass around contour shape that is .3 depth 1 pass using cutter comp.

250. FPM x .003 chip per tooth.

### O01010 (FINAL EXERCISE)

T\_\_\_\_ M\_\_\_\_ (1.00 DIA. END MILL)

G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_

S\_\_\_\_ M\_\_\_\_

G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

G\_\_\_\_ Z\_\_\_\_ F30. (Non-cutting fast feed to depth)

G\_\_\_\_ Y\_\_\_\_ D\_\_\_\_ F\_\_\_\_ (Turn on Cutter Comp.)

X\_\_\_\_

G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_ R\_\_\_\_

G\_\_\_\_ Y\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

X\_\_\_\_

X\_\_\_\_ Y\_\_\_\_

Y\_\_\_\_

G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_ R\_\_\_\_

G\_\_\_\_ G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_ (Cancel Cutter Comp.)

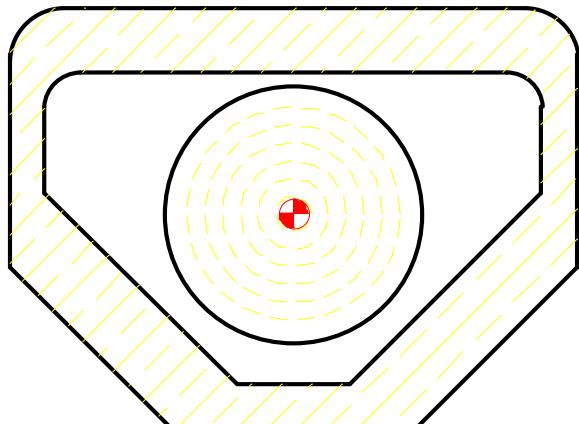
G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_

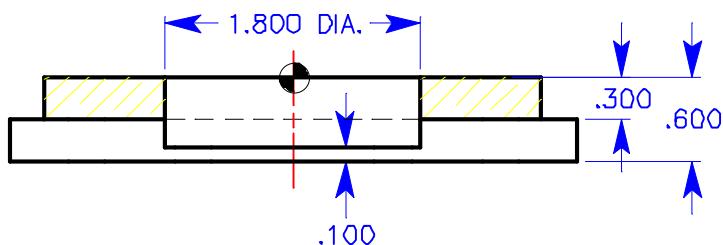
Be sure to position the End Mill at least half the cutter diameter off the part surface, before canceling cutter comp. Cancel to the center of tool.



## FINAL EXERCISE

**TOOL #2 - 1 3/4 Dia. 2 flute**

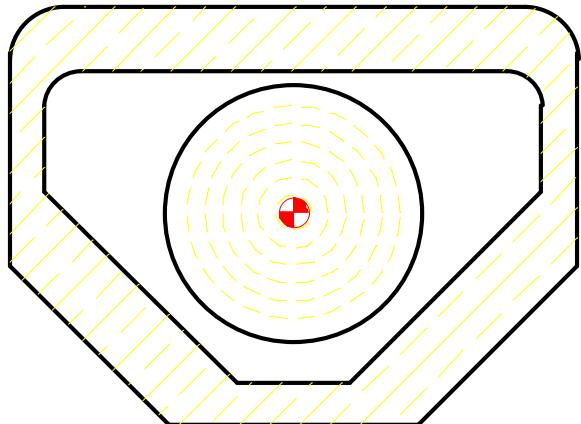
Insert Drill.

Rough the 1.800 circular pocket  
.480 depth using G81 canned cycle.  
250 FPM x .005 chip per tooth

T\_\_\_\_ M\_\_\_\_ (1 3\4 DIA. INSERT DRILL)  
G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_ Y\_\_\_\_  
S\_\_\_\_ M\_\_\_\_  
G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_ M\_\_\_\_  
G\_\_\_\_ Z\_\_\_\_ R\_\_\_\_ F\_\_\_\_  
G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_  
G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_ M\_\_\_\_



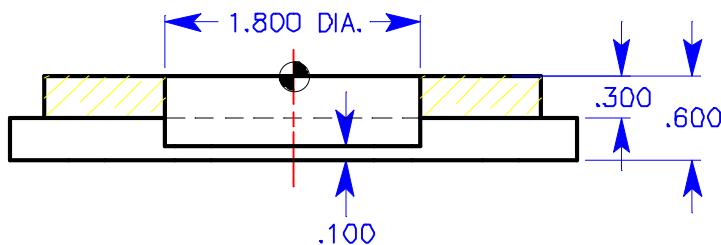
## FINAL EXERCISE



### TOOL #3 - 5/8 dia. 4 flt. End Mill

Mill a 1.800 dia. pocket to a .500 depth using G13 for a circular pocket milling routine.

250 FPM x .0025 chip per tooth



T\_\_\_\_ M\_\_\_\_ (5\8 DIA. 4 FLT. END MILL)

G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_

S\_\_\_\_\_ M\_\_\_\_\_

G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_\_ M\_\_\_\_\_

G\_\_\_\_ Z\_\_\_\_\_ F\_\_\_\_\_ (Plunge Z axis down in part at a feedrate)

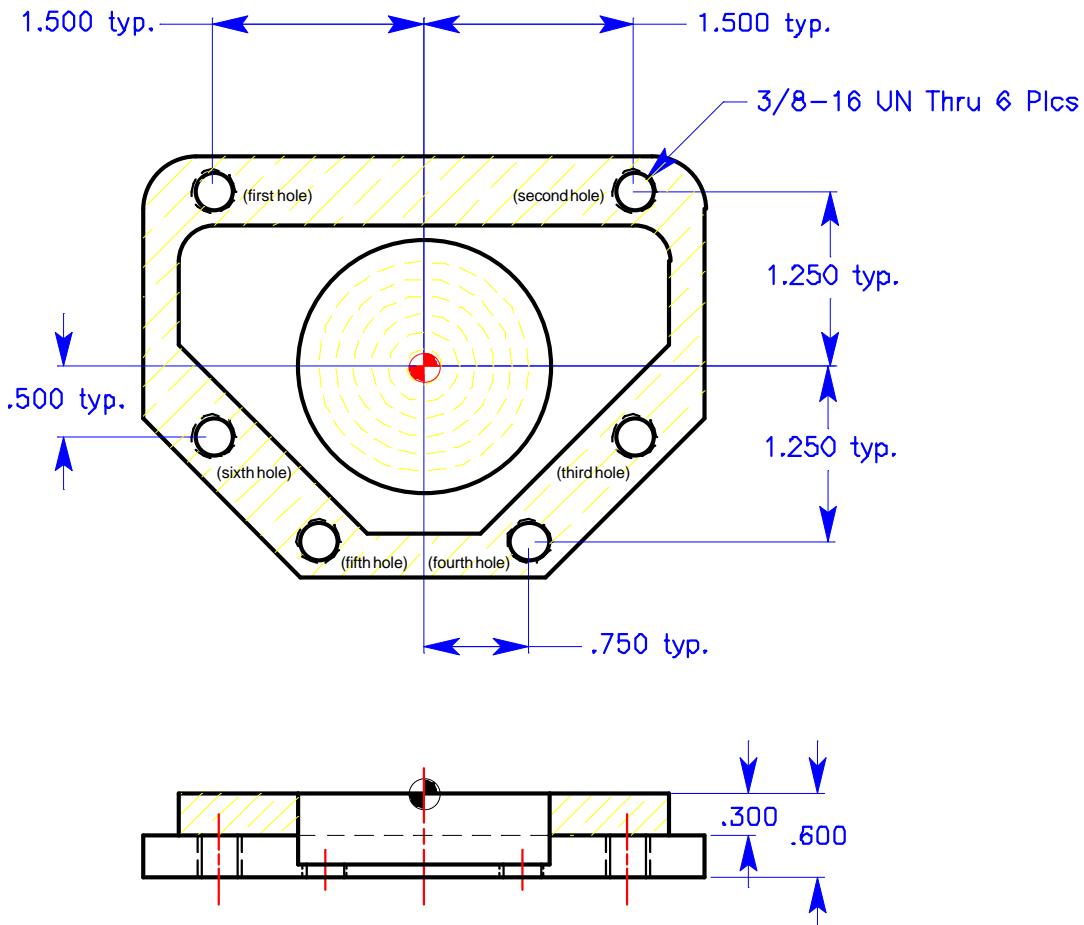
G\_\_\_\_ I\_\_\_\_ K\_\_\_\_ Q\_\_\_\_ Z\_\_\_\_\_ D\_\_\_\_ F\_\_\_\_

G\_\_\_\_ Z\_\_\_\_\_ M\_\_\_\_\_

G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_\_ M\_\_\_\_\_



## FINAL EXERCISE



*The rapid plane for the drill and tap is R-.2 down from the top surface of part.*

### TOOL #4 - 1/2 dia. Spot Drill

Use a G82 to spot drill for 6 holes to a depth of -.4 with a rapid plane of -.2 and a 1/2 second dwell at bottom.

200 FPM x .004 chip per tooth.

### TOOL #5 - 5/16 dia. Drill

Use a G73 to drill 6 holes thru part to a depth of -.75 with a .1 peck depth amount and a -.2 rapid plane.

200 FPM x .003 chip per tooth.

### TOOL #6 is a 3/8-16 Tap

Use a G84 to tap 6 holes thru part to a -.7 depth and a -.2 rapid plane, with a 600 RPM.

Machine is equipped with rigid tapping.



## FINAL EXERCISE

T\_\_\_\_ M\_\_\_\_ (1\2 DIA. SPOT DRILL)  
 G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_  
 S\_\_\_\_\_ M\_\_\_\_\_  
 G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_\_ M\_\_\_\_\_  
 G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_\_ P\_\_\_\_ R\_\_\_\_ F\_\_\_\_\_  
 M\_\_\_\_ P\_\_\_\_

T\_\_\_\_ M\_\_\_\_ (5\16 DIA. DRILL)  
 G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_  
 S\_\_\_\_\_ M\_\_\_\_\_  
 G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_\_ M\_\_\_\_\_  
 G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_\_ Q\_\_\_\_ R\_\_\_\_ F\_\_\_\_\_  
 M\_\_\_\_ P\_\_\_\_

T\_\_\_\_ M\_\_\_\_ (3\8-16 TAP)  
 G\_\_\_\_ G\_\_\_\_ G\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_  
 S\_\_\_\_\_ (M03 not needed with G84)  
 G\_\_\_\_ H\_\_\_\_ Z\_\_\_\_\_ M\_\_\_\_\_  
 G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_\_ R\_\_\_\_ F\_\_\_\_\_  
 M\_\_\_\_ P\_\_\_\_  
**G53 Y0**

**T1 M06** (Change to tool #1 for the next part to be run)  
**M\_\_\_\_** (End of program)

Use G98 and G99 for  
 the Z position clearance  
 location for positioning  
*between holes.*

**G98** Initial Point Return  
**G99** R Plane Return

(SUB-PROGRAM)

**O01011**  
 G\_\_\_\_ X\_\_\_\_\_  
 G\_\_\_\_ Y\_\_\_\_\_  
 G\_\_\_\_ X\_\_\_\_\_ Y\_\_\_\_\_  
 G\_\_\_\_ X\_\_\_\_\_  
 X\_\_\_\_\_ Y\_\_\_\_\_  
 G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_\_ M\_\_\_\_\_  
 G\_\_\_\_ G\_\_\_\_ Z\_\_\_\_\_ M\_\_\_\_\_  
 M\_\_\_\_



PROGRAMMING

**VH5 Series**  
WORKBOOK

JUNE 2000