



Haas Factory Outlet

A Division of Productivity Inc

Lathe Series Training Manual

Haas CNC Lathe Programming



Revised 051512

This Manual is the Property of Productivity Inc

*The document may not be reproduced without the express written permission of
Productivity Inc.*

*The content must not be altered, nor may the Productivity Inc name be removed
from the materials.*

*This material is to be used as a guide to operation of the machine tool. The
Operator is responsible for following Safety Procedures as outlined by their
instructor or manufacturer's specifications.*

To obtain permission, please contact trainingmn@productivity.com.

Haas Lathe Programming Training Manual

Table of Contents

HAAS LATHE PROGRAMMING TRAINING MANUAL TABLE OF CONTENTS	1
INTRODUCTION	5
THE COORDINATE SYSTEM	6
MACHINE HOME	7
ABSOLUTE AND INCREMENTAL POSITIONING	9
PROGRAMMING	14
PROGRAM FORMAT	15
DEFINITIONS WITHIN THE FORMAT.....	16
ALPHABETICAL ADDRESS CODES	17
PREPARATORY FUNCTIONS (G CODES)	22
MISCELLANEOUS FUNCTIONS (M CODES)	24
MACHINE DEFAULTS	25
LATHE PROGRAMMING	26
MACHINING CYCLES FOR THE LATHE	29
DRILLING, TAPPING AND BORING CANNED CYCLES.....	29
MACHINE CYCLES FOR TURNING AND GROOVING	30
LINEAR/CIRCULAR MOVEMENT-CREATING TOOL PATHS	31
INTERPOLATION COMMANDS	33
LINEAR INTERPOLATION EXERCISE	35
CIRCULAR INTERPOLATION COMMANDS.....	36
CIRCULAR INTERPOLATION MOTION EXERCISES	39
CIRCULAR INTERPOLATION EXERCISE	41
MANUALLY PROGRAMMING TOOL NOSE COMPENSATION	42
CALCULATING COMPENSATION FOR A RADIUS ON YOUR PART	42
RADIUS CALCULATION.....	43
EXTERNAL RADIUS CALCULATION	44
INTERNAL RADIUS CALCULATION.....	44
CALCULATING COMPENSATION FOR “AN ANGLE” ON YOUR PART	48
TOOL NOSE RADIUS CALCULATION DIAGRAM	49
TOOL NOSE COMPENSATION – TAPER CALCULATION.....	50
MISCELLANEOUS G CODES	61
REFERENCE POINT DEFINITION AND RETURN	61
SPINDLE SPEED COMMANDS.....	62

WORK COORDINATE SYSTEM SELECTION	63
FEED COMMAND FUNCTIONS.....	63
TOOL NOSE COMPENSATION G CODES.....	64
TOOL NOSE COMPENSATION PROGRAMMING	66
TOOL NOSE COMPENSATION CONCEPTS	68
IMAGINARY TOOL TIP AND DIRECTION.....	69
USING TOOL NOSE COMPENSATION	70
CANNED CYCLES AND ADDITIONAL G CODES.....	78
TOOL NOSE COMPENSATION IN CANNED CYCLES.....	84
TYPE 1 DETAILS	92
TYPE II DETAILS	92
TYPE I DETAILS.....	97
DRILLING, BORING AND TAPPING CANNED CYCLES.....	116
CANNED CYCLES FOR DRILLING AND TAPPING.....	117
M CODE DETAILED DESCRIPTION	131



Productivity Inc®

For more information on Additional Training Opportunities
or our Classroom Schedule

Contact the Productivity Inc Applications Department in Minneapolis:

☎ 763.476.8600

Visit us on the Web: www.productivity.com

Click on the Training Registration Button

✉ trainingmn@productivity.com

Introduction

Welcome to Productivity, Inc., your local Haas Factory Outlet (H.F.O.) for the Haas Lathe Programming Class. This class is intended to give a basic understanding of the set-up and operation of a Haas Turning Center.

In an "NC" (Numerically Controlled) machine, the tool is controlled by a code system that enables it to be operated with minimal supervision and with a great deal of repeatability. "CNC" (Computerized Numerical Control) is the same type of operating system, with the exception that a computer monitors the machine tool.

The same principles used in operating a manual machine are used in programming an NC or 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.

The operation of the SL-Series Turning Center requires that a part program be designed, written, and entered into the memory of the control. There are several options for getting these programs to the control. RS-232 (serial port with a computer), 3.5" Floppy Disk, Ethernet / Networking/ and USB are all viable ways to transmit and receive programs.

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.

At Productivity, we have two classes that pertain to Haas Turning Centers: Lathe Operator and Lathe Programming.

We have two classes to fill the different needs of our customers as not all people that require training require programming training.

This manual can be used as both an operator's manual and as a programmer's manual. It is intended to give a **basic** understanding of CNC programming and its 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.

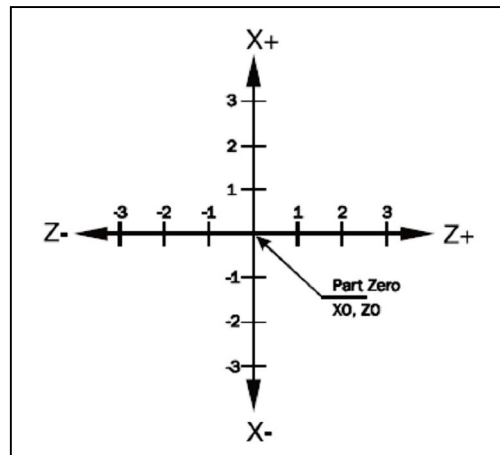
Revised CK 5/15/12

The Coordinate System

All CNC machines move tools to specific locations described by coordinate systems. With lathes the coordinate system can be simply described as two number lines that intersect.

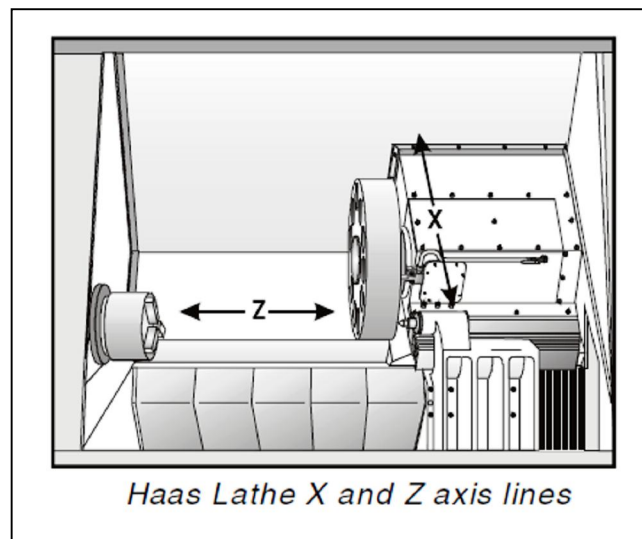
The illustration below shows two number lines that intersect at a location described as reference zero or Absolute Zero.

With lathes the vertical number line is called the **X-axis**. The horizontal number line is called the **Z-axis**.



The movements of the lathe are described by the illustration below. The up and down motion or X-axis corresponds to the vertical number line. The Z-axis or side to side motion corresponds to the horizontal number line. The intersection of the two lines is **Absolute Zero**.

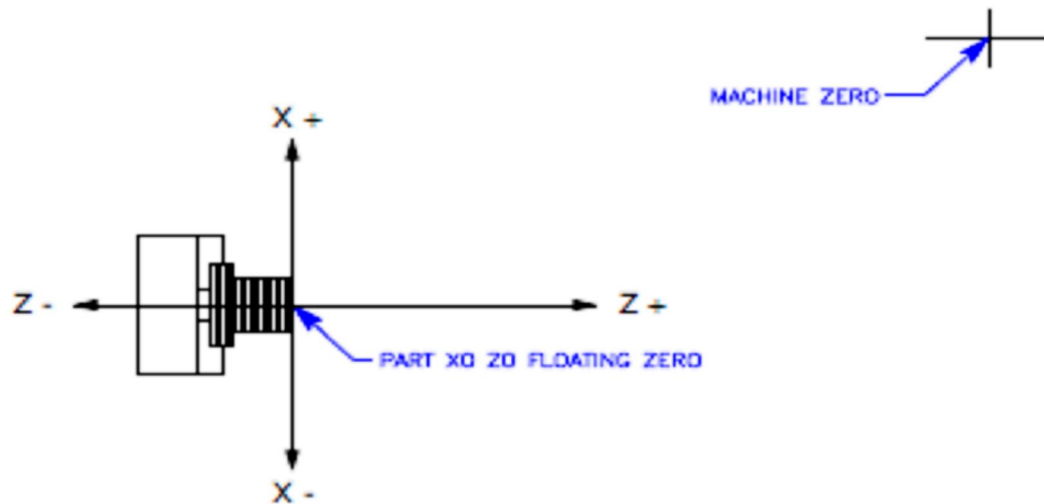
When programming lathes **X0** is always the centerline of the part you are working on. It is the X position on the Z axis that the part rotates around. **Z0** normally is the front finished face of the part.



Machine Home

At start up all Haas machine tools must be returned to zero or taken to what is called a Home Reference Position. A zero return (POWER UP/RESTART) is required when you power on a Haas machine. During a zero return all axes are moved to extreme positive locations until limit switches are reached. The reason the machine does this is to double check its position with the "Home" switches of the machine.

This is crucial to the operation and function of a CNC machine as **all of our programs, locations of fixtures and tooling are based off of machine home**. At home position the **machine coordinates are X0, Z0**. It would not be easy or convenient to write a program using machine coordinates. Instead programs are written with values that would correspond to dimensions found on prints. To do this a secondary floating zero point is established using offsets. This floating zero is referred to as the **PART ZERO or PART ORIGIN**

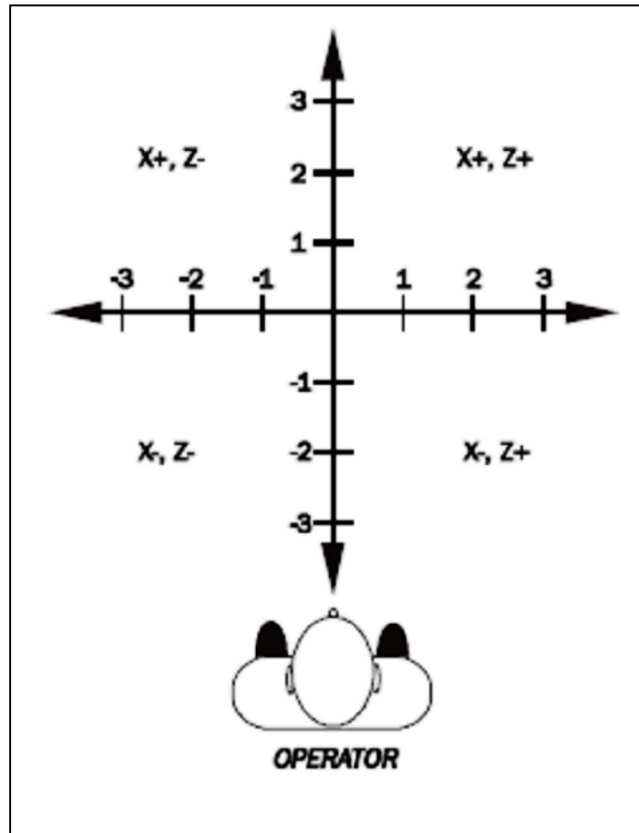


As shown above the centerline of the part becomes (X) zero. Normally the front face of the part is designated as Z (zero). The reason being it is easy to access and easier to program. With this scheme the **centerline of the lathe is always the "X" zero**. The **"Z" zero location** will float with the **face** of different parts.

When we setup a machine, we need to tell the machine the distance different tools at home position are from the part origin. Each tool is manually touched off the face and diameter and thru keystrokes the distances from machine zero to the part zero are saved in X and Z register of the **Tool Offsets Page**. Alternately tools may be set using the tool pre-setter. If the tools are set with a tool pre-setter or other Z reference point not at the Part Zero a Z Work Shift must be done.

A work shift is stored in the **Work Offsets Page** in the Z value. Several different work offsets may be used. The Z value in the Work Offset page is the distance and direction the Part Zero is from the Z reference or tool pre-setter touch off position. Normally X values are always zero. If the face of the part is used to determine the Tool Offsets then the value for Z in the Work Offset page will be zero.

X Axis



The diagram above shows the operator's view of X and Z grid standing in front of the lathe. At the intersection of the X and Z axis is the Part Zero or Reference Point. Note there is four different quadrants with different positive and values for X and Z.

An ST-20 lathe has travel distance of 9.3" in the X and 21" in the Z. Haas lathes have a negative one-inch (-1") travel beyond the centerline of the Z Axis spindle. Note that most of the X values in a part program are going to be positive. Normally the only negative X value is going to be when you face the part off past centerline. As the Z zero is normally the face of the part when you are cutting into the part the Z values will be negative.

Note that the X values on the Haas lathe are in diameters. Therefore an X move from centerline or X0 to X1.0 will only be moving the machine along the X axis 0.5 inch in the positive direction.

Absolute and Incremental Positioning

By using WORK and TOOL OFFSETS a common point on the part is designated as “**PART ZERO**”. This is some point on our part that we can physically find. The programmer uses this point as a base to write the intended movement of the tooling.

Programmers normally use the front end of our finish machined part as (Z Zero) and the centerline of part as (X Zero).

There are two methods used by the programmer to “Steer” our machine.

The first is “**ABSOLUTE POSITIONING**”. Absolute means that X and Z code values are based on the ZERO POINT on the part. If a diameter of 1.0000 inches is needed, it is input as X1.0000. If the print requires facing a shoulder that is 3 inches back from the front of the part, Z-3.0000 in input in the code.

The programmer has another tool available to him called “**INCREMENTAL POSITIONING**”. This is movement based on where the machine is currently sitting. It is also called point to point programming. If a change of half inch smaller diameter is required of the machine from where it is currently sitting U-.5000 is put in the code. If a grooving tool is making a groove that is located $\frac{3}{4}$ ” behind a groove that is already finished, W-.7500 is input.

The letters X&Z represent ABSOLUTE POSTIONING

The letters U&W represent INCREMENTAL POSTIONING

If you are familiar with the mill programming language, absolute and incremental are handled differently for Mills and Lathes. A mill uses G codes (G90 and G91) to go back and forth between the two. Where as a lathe uses the different letters to differentiate them.

QUESTION:

Why doesn't a lathe take G90 and G91 like a mill?

ANSWER:

A lathe has the unique possibility to do Absolute AND Incremental moves AT THE SAME TIME.

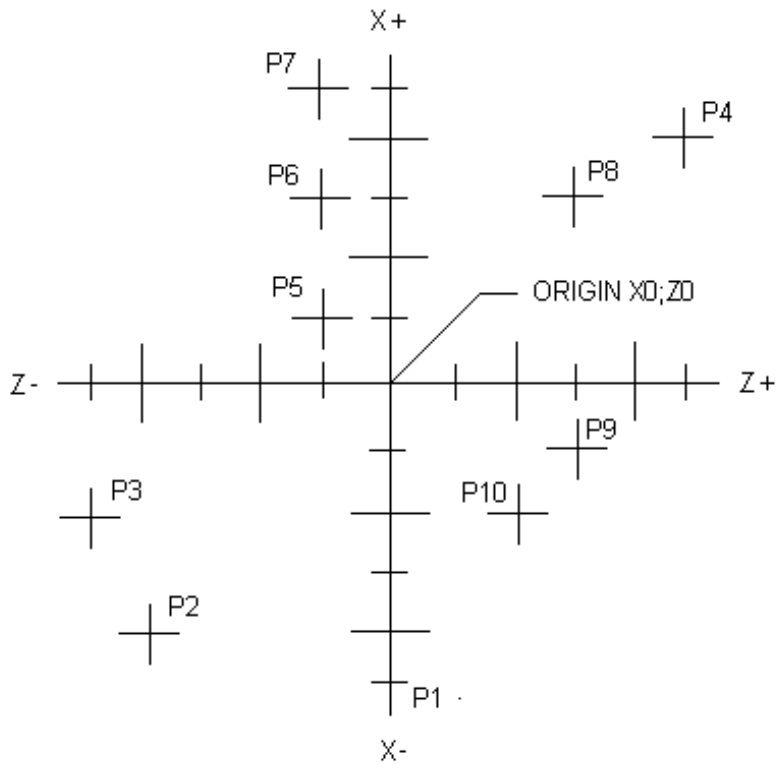
The programmer can place an ABS. letter and an INC. letter on a line of code together, and is most commonly used in making tapers and radius moves.

G01 X2.000 W-.25 (Move X in Absolute, Z in Incremental)

or

G01 U.5000 Z-.5000 (Move X in Incremental, Z in Absolute)

THE CARTESIAN COORDINATE SYSTEM



Lathe program exercise. Give X and U values as Diameters

ABSOLUTE

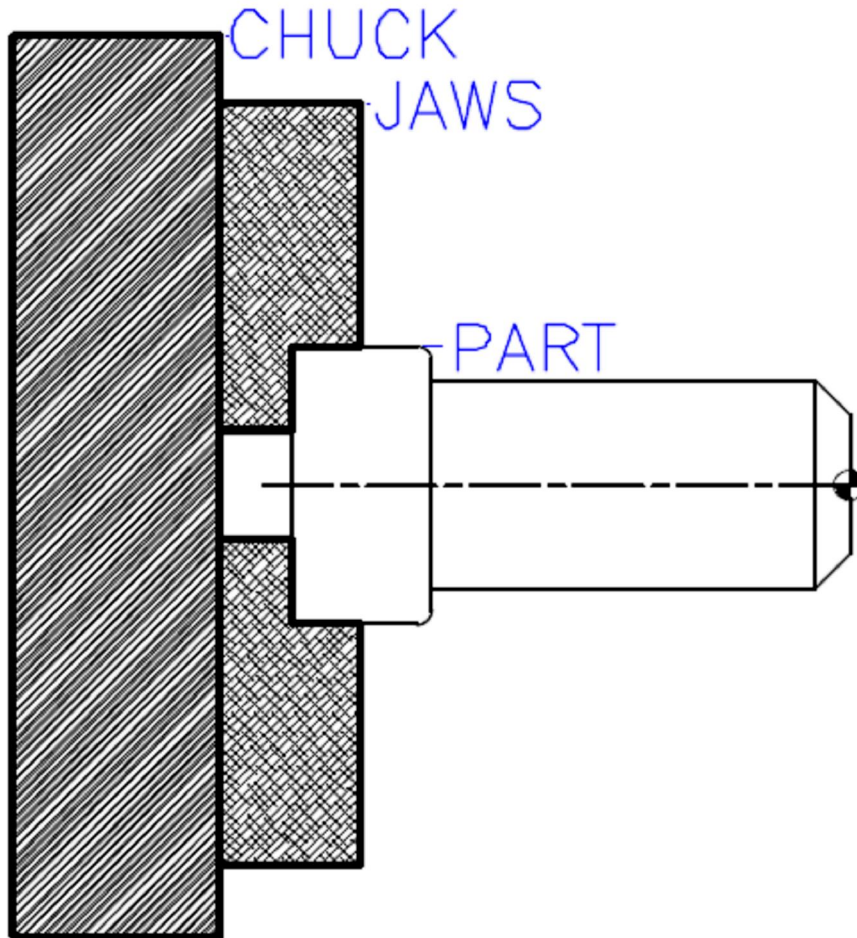
P1	X-5.0	Z0
P2		
P3		
P4		
P5		
P6		
P7		
P8		
P9		
P10		

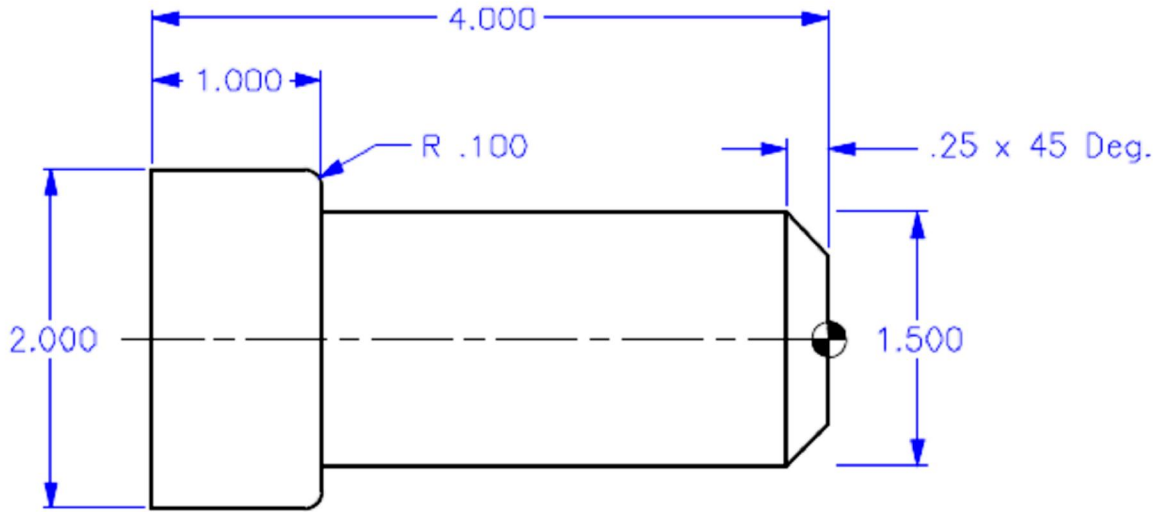
INCREMENTAL

P1	U-5.0	W0
P2		
P3		
P4		
P5		
P6		
P7		
P8		
P9		
P10		

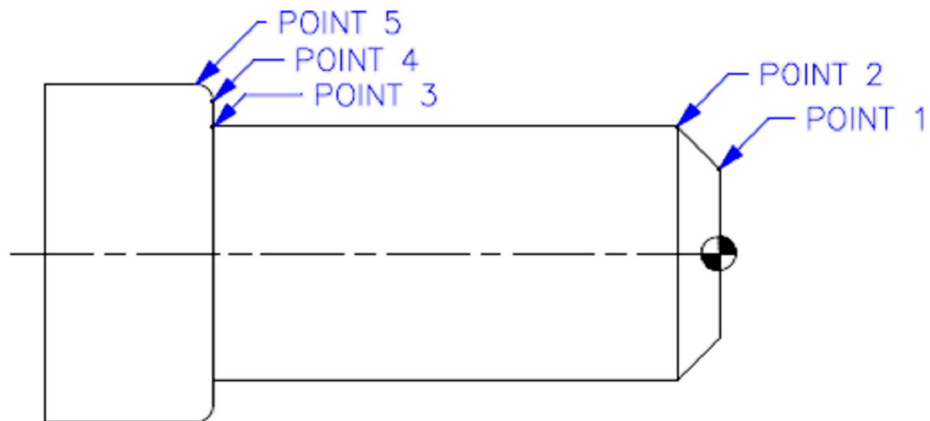
Typical Lathe Part

Below is a typical lathe part sitting in a chuck. Note the \oplus symbol. It denotes the floating zero determined by the programmer.





Above is the print for the part on previous page. Knowing X and Z coordinates for key positions below is the first step in programming.



Give the X and Z coordinates for the part below. Note the X values are diameters on Haas Lathes not radii.

	X	Z
Point 1		
Point 2		
Point 3		
Point 4		
Point 5		

Programming

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 codes commands.

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

The Haas lathe has two (2) linear axes named X and Z. the X-axis moves the tool turret toward and away from the spindle center line, while the Z axis moves the tool turret along the spindle axis. The machine zero position is where the tool is at the right corner of the work cell farthest away from the spindle axis. Motion in the X-axis will move the table toward the spindle centerline for negative numbers and away from spindle center for positive numbers. Motion in the z-axis will move the tool toward the spindle chuck for negative numbers and away from the chuck for positive numbers.

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 =      TURN THE SPINDLE ON AND SELECT THE RPM.
LINE #3 =      TURN THE COOLANT ON.
LINE #4 =      RAPID TO THE STARTING POSITION OF THE PART.
LINE #5 =      CHOOSE THE PROPER FEED RATE AND MAKE THE CUT(S)
LINE #6 =      TURN OFF THE SPINDLE AND THE COOLANT.
LINE #7 =      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.

Program Format

There are no positional requirements for a line CNC code. That means the different codes in a program may be in any order on a line of code. However some standard rules are followed so the code is easier to view. It will also make it easier for the machinist to understand when he runs the part if the program has a standard level of organization. Some standard rules are followed:

X and Z values are positioned in alphabetical order and grouped together

G and M codes may be placed anywhere on a line but convention is that the G codes come first and the M codes come at the end of the block. This makes sense as the last thing to happen on a line is the M-function. The G codes are completed first then the M code is performed on any given line.

On Haas machines only one M code is allowed on a block of code.

Command codes are first given by a letter then a number. Some codes like X,Z and F require decimal points. Others like S and G require an integer (a number with no fractional part).

.

Definitions Within the Format

Character: A single alphanumeric character value or the “+” and “-” sign.

Word: A series of characters defining a single command such as “X” displacement or “F” Feed rate. Unique letters are assigned as the first character of a word, and each letter has either a plus (+) or minus (-) sign value of numbers.

Block: Series of words defining a single instruction. An instruction may consist of a single linear or circular motion, plus additional information such as a feed rate or stop command. A block of code is on one single line.

Positive Signs: If a number value following the address command letters, such as I, K, R, U, W, X, Z, is positive, the plus sign need not be programmed. If the number value is negative, it must be programmed with a minus (-) sign.

Leading Zero's: There is no need to program zeros proceeding a number.

Example: G00 (G0) and M01 (M1).

Trailing zeros however must be programmed.

Example: M30 not M3, G70 or G7.

Modal Commands: Once a particular word, such as G, X, Z, F, S, T, and M is programmed, it is not necessary to repeat the word in the following blocks until a different word or change of value is required in subsequent blocks of information.

Non-Modal Commands: A non-modal command is one that is active only in the program block in which it is issued. M00 program stop is an example of a non-modal command.

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 feed rate along a straight line.
- 3) Move the tool along and arc at a feed rate in a clockwise direction.
- 4) Move the tool along an arc at a feed rate in a counterclockwise direction.
- 5) Move the tool through a series of repetitive operations controlled by “fixed cycles” such as, spot drilling, boring, and tapping.

Miscellaneous Functions: “M” codes cause an action to occur at the end of the block. Only one M-Code is allowed in each block.

Sequence Numbers: Sequence numbers are codes N1 through N9999 and are only used to locate a certain block or line within a CNC program. A program may be input without sequence numbers.

Alphabetical Address Codes

The following is a list of the Address Codes used in programming the lathe:

A FOURTH AXIS ROTARY MOTION

The A address character is normally 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. The smallest magnitude is 0.001 degrees, the most negative value is -8380.000, and the largest value is 8380.000 degrees.

The A axis on the **Haas lathe** is currently **reserved** for the **tool turret** and is hidden to the programmer. The units of rotation indicate tool positions such that 1.000 represents tool #1.

B LINEAR B-AXIS MOTION

The B address character is used to specify absolute motion for the B axis. It specifies a position or distance along the B axis. It is either in inches with four fractional positions or millimeters with three fractional positions. It is followed by a signed number between -8380.00 and 8380.00. If no decimal point is entered, the last digit is assumed to be 1/10000 inches or 1/1000 millimeters.

On the Haas lathes the B axis is either travel of the tail stock or sub-spindle.

C FIFTH AXIS ROTARY MOTION

The C address character is used to specify motion for the optional external fifth, 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. The smallest magnitude is 0.001 degrees, the most negative value is -8380.00, and the largest value is 8380.00 degrees.

D NOT USED, OPTIONAL MACRO PARAMETER

E FEED RATE, 6 PLACE PRECISION (SAME AS F)

The E address character is used to select feed rate applied to any interpolating G codes or canned cycles. The unit is in inches per revolution or mm per revolution. Up to six fractional positions can be specified. The default of units/revolution (G99) can be changed to units/minute with G98. For YASNAC and FANUC control compatibility, use the E code when 5 or 6 place precision is desired.

F FEED RATE

The F address character is used to select feed rate applied to any interpolating G codes or canned cycles. The unit is in inches per revolution or mm per revolution. The default of units/revolution (G99) can be changed to units/minute with G98. Traditionally, the F code was capable of only 4-fractional position accuracy; but on this control you can specify F to six place accuracy. Code E and F are equivalent.

G PREPARATORY FUNCTIONS (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 255. 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 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. More than one G code can be placed in a block in order to specify all of the setup conditions for an operation. See the following section (Preparatory Functions (G Codes)) for a detailed list of G codes. The G codes are specified in an A/B/C format to show the three different G coding systems. Each system uses a different G code for the same function. In this manual, programming is described based on the A system, however, Setting 34 (G CODE SYSTEM) can be used to select another system.

H NOT USED, OPTIONAL MACRO PARAMETER

I CANNED CYCLE AND CIRCULAR OPTIONAL DATA

The I address character is used to specify data used for some canned cycles and circular motions. It is either in inches with four fractional positions or mm with three fractional positions. It is followed by a signed number in inches between -8380.00 and 8380.00 for inches or between -8380.00 and 8380.00 for metric.

J CANNED CYCLE AND CIRCULAR OPTIONAL DATA

The J address character is used to specify data used for some canned cycles and circular motions. It is formatted just like the I data.

K CANNED CYCLE AND CIRCULAR OPTIONAL DATA

The K address character is used to specify data used for some canned cycles and circular motions. It is formatted just like the I data.

L LOOP COUNT FOR REPEATED CYCLES

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

M M CODE MISCELLANEOUS FUNCTIONS

The M address character is used to specify an M code for a block. These codes are used to control miscellaneous machine functions. Not that only one M code is allowed per block of the CNC program and all M codes are performed at the end of the 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, M98 and M99 functions may reference an N line number.

O PROGRAM NUMBER PROGRAM NAME PARENTHESIS

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 an Onnnnn identification in the first block; it cannot be deleted. Altering the O in the first block causes the program to be renamed. A program may have only one Onnnnn command a colon (:) may be used in the place of O, but is always displayed as "O". "O" Programs may be deleted by highlighting in the List Program page by using the DEL key or ERASE PROG key. Up to 500 programs may be stored in the List Program page.

P DELAY TIME OR PROGRAM 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 G04 dwell) or a program name (for a M97), the value may be either a positive number without decimal point up to 9999. If it is used as a time, it may be a positive decimal with fraction between 0.001 and 1000.00.

Q CANNED CYCLE OPTIONAL DATA

The Q address character is used in canned cycles and is always a positive number in inches/mm between 0 and 100.0.

R CANNED CYCLE AND CIRCULAR OPTIONAL DATA

The R address character is used in canned cycles and circular interpolation. It is either in inches with four fractional positions or mm with three fractional positions. It is followed by a signed number in inches between -8380.00 and 8380.00 for inches or between -8380.00 and 8380.00 for metric. It is 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. 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 the spindle is running, a gear change operation will occur and the spindle will continue running at the new speed. A decimal is not allowed with a S command.

T TOOL SELECTION CODE

The T address selects a tool and offset while initiating the tool change process. The format is Txxyy. The xx selects the tool turret position. The number depends on the location of the tool with respect to the turret position. It is a positive number between 1 and the total turret tools (i.e. Parameter 65). yy selects the tool offset and is a positive number between 1 and 50.

U INCREMENTAL X AXIS MOTION

The U address character is used to specify motion for the X-axis. It specifies an incremental position or distance along the X-axis relative to the current machine position. It is either in inches with four fractional positions or mm with three fractional positions. It is followed by a signed number in inches between -838.0000 and 838.0000 for inches or between -8380.000 and 8380.000 for metric. If no decimal point is entered, the last digit is assumed to be 1/100000 inches or 1/1000 mm.

V OPTIONAL MACRO PARAMETER

W INCREMENTAL Z AXIS MOTION

The W address character is used to specify motion for Z-axis. It specifies an incremental position or distance along the Z-axis relative to the current machine position. It is formatted the same as address U.

X LINEAR X-AXIS MOTION

The X address character is used to specify absolute 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 between -8380.00 and 8380.00 for inches or -8380.00 and 8380.00 for metric. If no decimal point is entered, the last digit is assumed to be 1/10000 inches or 1/1000 mm.

Y NOT USED, OPTIONAL MACRO PARAMETER

Z LINEAR Z-AXIS MOTION

The Z address character is used to specify absolute motion for the Z-axis. It specifies a position or distance along the Z-axis. It is either in inches with four fractional or mm with three fractional positions. It is followed by a signed number in inches between -8380.00 and 8380.00 for inches or between -8380.00 and 8380.00 for metric. If no decimal point is entered, the last digit is assumed to be 1/10000 inches or 1/100

Preparatory Functions (G Codes)

- 1) G codes come in groups with each group having a specific group number.
- 2) A G code from the same group can be replaced by another G code in the same group. This establishes modes of operation. The universal rule here is that 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 G code from the same group.
- 4) There are non-modal G codes which, once called, are effective only in the calling block and are immediately forgotten by the control.

The rules above govern the use of the machine code used for programming the Haas control. The concept of grouping codes and the rules that apply will have to be remembered if we are to effectively program the machine tool. The following is a brief description on some of the codes most basic to the operation of the machine.

(*) denotes a default code.

The following is a summary of G codes.

<u>Code:</u>	<u>Group:</u>	<u>Function:</u>
G00	01*	Rapid Motion (VF2 1000 in/min, SS 1400 in/min)
G01	01	Linear Interpolation Motion
G02	01	CW Interpolation Motion
G03	01	CCW Interpolation Motion
G04	00	Dwell (non-modal)
G09	00	Exact Stop (non-modal)
G10	00	Programmable Offset Setting (non-modal)
G17	02	XY Plane Selection (used with live tooling)
G18	02*	ZX Plane Selection (Plane used on lathes)
G19	02	YZ Plane Selection (used with live tooling)
G20	06	Inch Programming Selection (setting #9)
G21	06	Metric Programming Selection
G28	00	Machine Home in Rapid Traverse
G29	00	Set Return to Reference Point
G31	00	Skip Function
G40	07*	Tool Nose Compensation Cancel
G41	07	Tool Nose Compensation Left
G42	07	Tool Nose Compensation Right
G50	11	Spindle Speed Clamp, Global Coordinate
G51	11	Cancel G50 Offset (Yasnac)
G52	00	Child Coordinate System, M30, Reset Cancels
G53	00	Non-Modal Machine Coordinate Selection

G54	12*	Select Work Coordinate System 1
G55	12	Select Work Coordinate System 2
G56	12	Select Work Coordinate System 3
G57	12	Select Work Coordinate System 4
G58	12	Select Work Coordinate System 5
G59	12	Select Work Coordinate System 6
G61	13	Exact Stop Modal
G64	13*	G61 Cancel
G65	00	Macro Subroutine Call
G70	00	Finishing Cycle
G71	00	O.D./I.D. Stock Removal Cycle
G72	00	End Face Stock Removal Cycle
G73	00	Irregular Path Stock Removal Cycle
G74	00	End Face Grooving Cycle, Peck Drilling
G75	00	O.D./I.D. Grooving cycle, Peck Drilling
G76	00	Thread Cutting Cycle, Multiple Pass
G80	09*	Canned Cycle Cancel
G81	09	Drill Canned Cycle
G82	09	Spot Drill Canned Cycle
G83	09	Peck Drill Canned Cycle
G84	09	Tapping Canned Cycle
G85	09	Boring Canned Cycle
G86	09	Bore/Stop Canned Cycle
G87	09	Bore/Manual Retract Canned Cycle
G88	09	Bore/Dwell Canned Cycle
G89	09	Bore Canned Cycle
G90	01	O.D.I.D. Turning cycle, Modal
G92	01	Thread Cutting Cycle, Modal
G94	01	End Face Cutting cycle, Modal
G96	12	Constant Surface Speed On
G97	12*	Constant Surface Speed Cancel
G98	05	Feed per Minute
G99	05*	Feed per Revolution
G102	00	Programmable Output to RS-232
G103	00	Block Look Ahead Limit (P0-P15)
G110-G129	12	Extra Work Coordinate System 7 through 26 older Machines
G154P1 to G154P99	12	Extra Work offsets newer machines
G184	09	Reverse Tap Canned Cycle
G187	00	Accuracy Control for High Speed Machining

Miscellaneous Functions (M Codes)

M-codes are non-axes moving commands; also called *machine functions*.

The format for M-codes is the letter “M” followed by two numbers, for example M01.

Only one M-code is allowed on a single block of code.

M-codes are the last thing to take effect on a line of code.

M-codes are usually placed at the end of a block of code.

M00	Program Stop Stops spindle, axes movement, turns off coolant On pressing CYCLE START button program will continue on from where it stopped
M01	Optional program stop Works exactly the same as a M00 Is only active when optional stop key activates it on control panel
M02	End of program – cannot continue Most common way is using M30
M03	Start spindle forward (Clockwise) Must be accompanied by a spindle speed
M04	Start spindle reverse (Counterclockwise) Must have a spindle speed (S value)
M05	Spindle Stop
M08	Coolant on Command
M09	Coolant off Command
M10	Clamp Spindle Chuck
M11	Unclamp Spindle Chuck
M21	Tailstock Forward
M22	Tailstock Reverse
M23	Thread Chamfer On during a G76 or G92 cycle

M24	Thread Chamfer Off
M30	Program end and rewind to beginning of program
M41	Low Gear
M42	High Gear
M88	High Pressure Coolant On
M89	High Pressure Coolant Off

NOTE: Only one "M" code can be used per line. The "M" code will be the last item of code to be performed, regardless of where it is located in the line.

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 entered for G54 if no other work coordinate code was specified in the actual program. The machine automatically recognizes the G54 column upon start-up. This is known as a default.

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

G00	Rapid Traverse
G18	X, Z Circular Plane Selection
G40	Cutter Compensation Cancel
G54	Work Coordinate Zero #1 (1 of 26 available)
G64	Exact Stop Cancel
G80	Canned Cycle Cancel
G97	Constant Surface Speed Cancel
G99	Feed Per Revolution

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

Lathe Programming

Program format, or program style, is an important part of CNC machining. Each individual will format their programs differently. The point is that a programmer needs to be consistent and efficient. For example:

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

The **first line** or block in a program should be a return to machine zero (using G28). Any tool change should be after a return to machine zero. Although this is not necessary, it is a good safety measure. With extended live tools, tail stocks or sub-spindles significant clearance issues appear. As a result, programmers will take the machine home first in X with either G28 U0 or G53 X0 commands. Then take the machine home in Z with a G28 W0 or G53 Z0 command. Another option is to take the machine to a safe Z index position. This is done by using machine coordinates, G53 Z-5.0 as an example.

No safety line? Examples in this book do not have safety lines. A possible safety line could be:

G18 G20 G40 G54 G80 G97 G99

CAD/CAM systems will by default output a safety line in the beginning of a program or before a tool change. This is done to make sure the machine is in a safe start condition. Some programmers however don't employ a safety line in lathes the reasons are given below:

G18: G18 is the default command for designating which plane (X,Z plane) a radius is thrown. If an arc is attempted in a G17(XY plane) or a G19 (YZ plane) the Haas lathe will alarm out.

G20: G20(inch) and G21(mm) make sure setting #9 is set to inch or metric. Normally most shops are either inch or metric. If the wrong setting is active it will be clearly seen in graphics or the machine will give an over travel alarm.

G40: G40 cancels cutter compensation codes G41 or G42. Cutter compensation should be cancelled after it is used. If it is not cancelled and the program is run in graphics alarm 349 PROG STOP W/O CANCEL CUTTER COMP. As a result the machine will pick up the mistake anyway.

G54: Work offsets are not used as in mills. If no pre-setter is used many times there are no values in the work offsets register. All the Z offsets are reflected in the Tool Geometry offsets. If no work offsets are used some programs will not have G54 in them. G54 is however the default value and if not changed will always be active

G80: Cancels canned cycles. If it is missing in a program a G00, G01, M30 or RESET will automatically cancel any can cycle.

G97: Constant spindle speed. A G97 should be programmed at the beginning and end of every tool in a program with a spindle speed.

G99: Feed per revolution: This is the default. Normally lathes are always in G99 mode, they are never changed to G98 (Feed per minute). The only time G98 would be used if live tooling is being used. If you are in G98 mode when G99 is required the feed rates are so small (.009) the machine would just move very slow taking forever to complete the command. If you are in G99 mode and a typical Feed/Minute say 10 inches/minute the machine at a speed of 200 RPM the machine would interpret this as 2000 inch per minute way above the feed limit on the machine and alarm out.

So it is up to the individual programmer to decide if a safety line is used.

The **second line** of code indexes the turret to the appropriate tool giving proper geometry or tool shifts.

The **third line** may contain a spindle speed maximum, G50, for the lathe during any operation..

The **fourth line** or block should cancel any constant surface speed mode (G97) and specify a spindle speed command (S____) along with a spindle ON clockwise command (M03). This speed should correspond to a constant surface speed (G96) in line N6 at the diameter indicated in N5. If not the lathe will hesitate until it gets to the proper speed and sound strange.

The **fifth line** should contain a preparatory code (G00) for a rapid movement to a start X and Z position. It should specify the work coordinate being used, and to turn on the coolant (M08).

The **sixth line** may optionally specify a Constant Surface Speed (G96) and a surface feet per minute (SFM) defined with an (S____) command.

An example of the first five lines of a program might look like this:

```
N1 G28;  
N2 T101;  
N3 G50 S2000;  
N4 G97 S1146 M03;  
N5 G54 G00 X1.5 Z.02 M08;  
N6 G96 S450;
```

This format is a good example and defines a commonly used program style.

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

Notes on Program Structure

A CNC program consists of a one or more blocks of commands. A block of code may be considered a line of text. At the end of the block is a semicolon (;). This denotes the end of the text or block and is also called a **(EOB)** end of block symbol. When at the machine and manually entering a program pressing the **EOB key** will enter the semicolon symbol and move down to the next line in the program.

Programs must begin and end with a percentage symbol (%). These symbols will not be seen on the controls of the Haas. If however you save the program to say a USB device and look at the program using a computer they will be there. If you are trying to load a program into the CNC without percentage symbols at the beginning and end of a program the CNC machine will alarm out and not accept the program.

After the first % sign the next line of the program requires a program name using the letter O and a four digit number on older machines or 5 digit number on newer machines. This symbol identifies a particular program and is used to call up the correct program. On the program number line, text in parenthesis, example (P/N 15466 JD shaft) may further be used to give the program a title. Text inside the () is not seen or used by the machine. This text will show up in list program screen of the memory.

Programs may also contain back slashes (/). This denotes an optional block. When **BLOCK DELETE** is active the program will ignore any commands on any line containing a (/).

Machining Cycles for the Lathe

A machining cycle is used to simplify the programming of a part. Machine cycles are defined for the most common machining operations. There are two types: machining cycles for turning and grooving and canned cycles for drilling, tapping, and boring.

Drilling, Tapping and Boring Canned Cycles

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

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

These cycles are modal, which remain in effect after they are defined and executed in the Z-axis for each positioning of X axes in a program. Some of the cycle command values can be changed after these cycles have been defined. The command values most often changed during a cycle are the R plane value and the Z depth value. These modal cycles will be canceled with the G80, G01 or G00 commands. The X-axis moves in these modal machine cycles are performed as rapid moves.

The operation of a canned cycle will vary according to whether incremental (U, W) or absolute (X, Z) is specified. Incremental motion is often useful in a canned cycle. If a loop count (Lnn code number) is defined within the block, the canned cycle will repeat that many times with an incremental U (X-axis move) move between each cycle.

The following is a list of the canned cycles that can be used on the HAAS lathe:

G80	Canned Cycle Cancel
G81	Drill Canned Cycle
G82	Spot Drill Canned Cycle
G83	Peck Drill Canned Cycle
G84	Tapping Canned Cycle
G85	Bore in Bore out Canned Cycle
G86	Bore in Rapid out Canned Cycle
G87	Bore with Manual Retract Canned Cycle
G88	Bore in Dwell with Manual Retract Canned Cycle
G89	Bore in Dwell Bore out Canned Cycle

Machine Cycles for Turning and Grooving

The following is a list of the canned cycles that can be used for turning and grooving for the HAAS lathe controls.

G70	Finishing Cycle
G71	O.D./I.D. Stock Removal cycle
G72	End Face Stock Removal Cycle
G73	Irregular Path Stock Removal Cycle
G74	End Face Grooving or Turning with Chip Break Cycle
G75	O.D./I.D. Grooving or Turning with Chip Break cycle
G76	Thread Cutting Cycle, Multiple Pass
G90	O.D./I.D. Turning Cycle Modal
G92	Thread Cutting Cycle Modal
G94	End Face Cutting Cycle Modal

A machine cycle is used to simplify programming of a part. Machine cycles are defined for the most common, repetitive operations such as turning, facing, threading and grooving. There are both modal and non-modal machine cycles. Modal cycles, such as turning cycle G90, remain in effect after they are defined. After any subsequent X or Z-axis positioning, the canned cycle is executed again. Modal canned cycles remain in effect until canceled by a G80, G00, end of program, or RESET. Non-modal machine cycles are effective only for the block that contains them, but will perform a series of machining moves to perform that command.

Linear/Circular Movement-Creating Tool Paths

CNC programs consist of lines and arcs. This unit will discuss the proper methods to create tool paths applied to CNC programs to create the desired results in addition to creating tool paths with and without cutter compensation.

Objectives:

Upon completion of this unit, the student will:

- 1) Understand the major differences in G01 and G00.
- 2) Understand and apply the five criteria needed to produce an arc.
- 3) Understand the principles of programming and applications with and without cutter compensation in addition to the advantages and disadvantages of each.
- 4) Be capable of producing a tool path program, containing lines and arcs, with and without cutter compensation.
- 5) Be capable of determining feeds and speeds given an SFM and cutting tool, and be capable of determining a feed rate given an RPM and chip load.
- 6) Be able to integrate a tool path into a part program.
- 7) Understand the rules governing the use of cutter compensation.
- 8) Have a basic understanding of the concept of arc in/arc out and some of its applications.

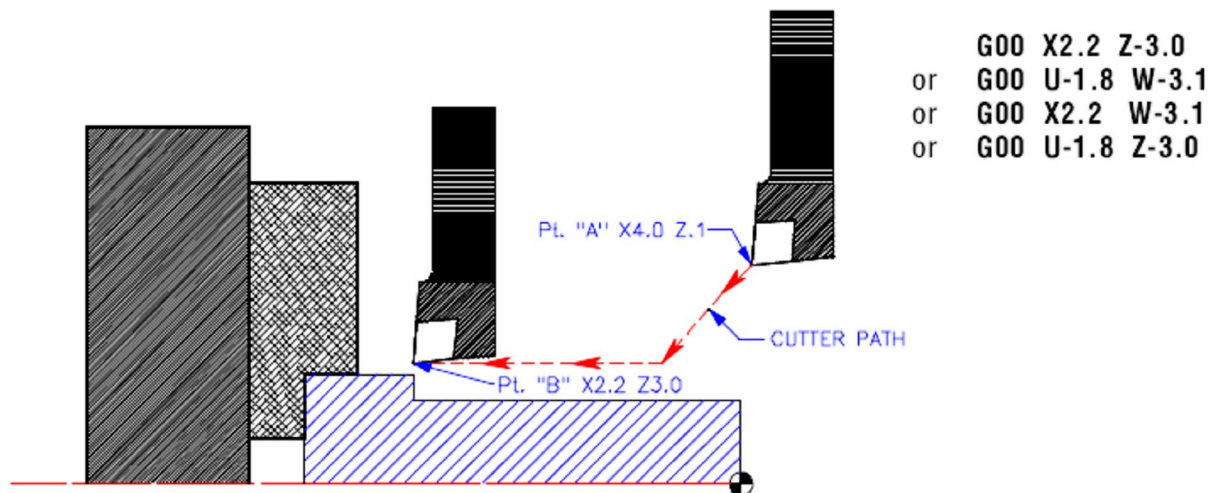
G00 Rapid Motion Positioning

X	Absolute X-axis rapid motion command
Z	Absolute Z-axis rapid motion command
U	Incremental X-axis rapid motion command
W	Incremental Z-axis rapid motion command
B	Absolute tailstock rapid motion command

The G code is used to cause a rapid traverse of the two axes of the machine from one programmed point to the next programmed point. The auxiliary (tailstock) B-axis can also be moved with a G00. This G code is modal so that all following blocks will be in rapid motion until another group 01 is specified.

Generally, rapid motions done together in both X and Z-axes will not be in a straight line from one program location to the next program location. All of the axes specified are moved at the same speed (710 IPM), but will not necessarily complete their motions at the same time. Each axis slide is activated independently and the axis with the shortest distance to move will reach its endpoint first; thus, giving the impression of a 45-degree angle move. The control will wait until all motions are complete. These rapid moves may be made in ABSOLUTE or INCREMENTAL coordinate command values, which will change how those values are interpreted. The "U" letter address relates to X-axis incremental moves and the "W" letter address relates to Z-axis incremental moves.

To move from point "A" to point "B", the programmed line can be either:



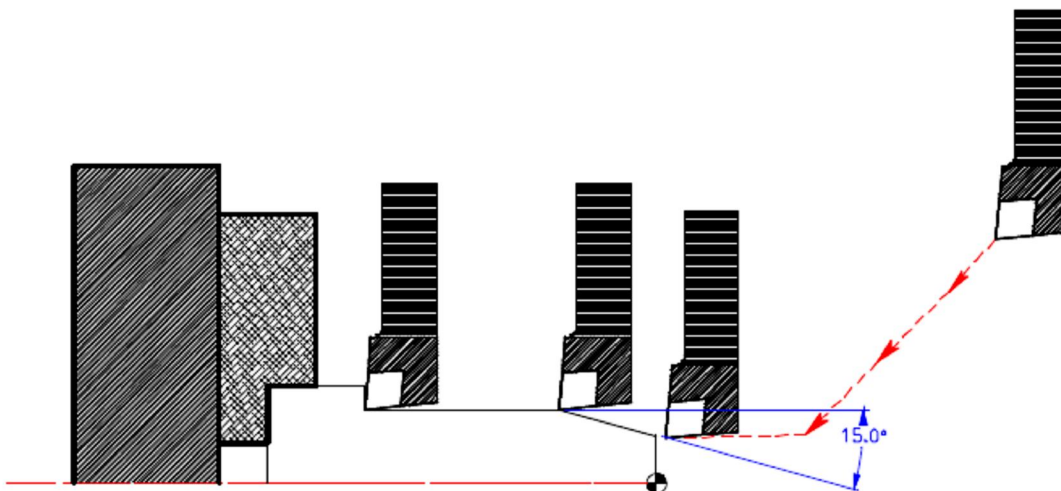
Interpolation Commands

G01 Linear Interpolation Motion

X	Absolute X-axis feed motion command
Z	Absolute Z-axis feed motion command
U	Incremental X-axis feed motion command
W	Incremental Z-axis feed motion command
B	Absolute tailstock feed motion command
F	Feed rate

This G code provides for straight-line (linear) motion from point-to-point. Motion can occur in one or two axes. Both axes will start and finish motion at the same time to move the tool along a straight-line path parallel to an axis or at a slope (angled) line. The speeds of all axes are controlled so that the feed rate specified is achieved along the actual path. The F (Feedrate) command is modal and may be specified in a previous block. These moves may be made in ABSOLUTE or INCREMENTAL coordinate command values, which will change how those values are interpreted. The “U” letter address relates to X-axis incremental moves and the “W” letter address relates to Z-axis incremental moves. Only those axes specified are moved in either absolute XZ or incremental UW commands.

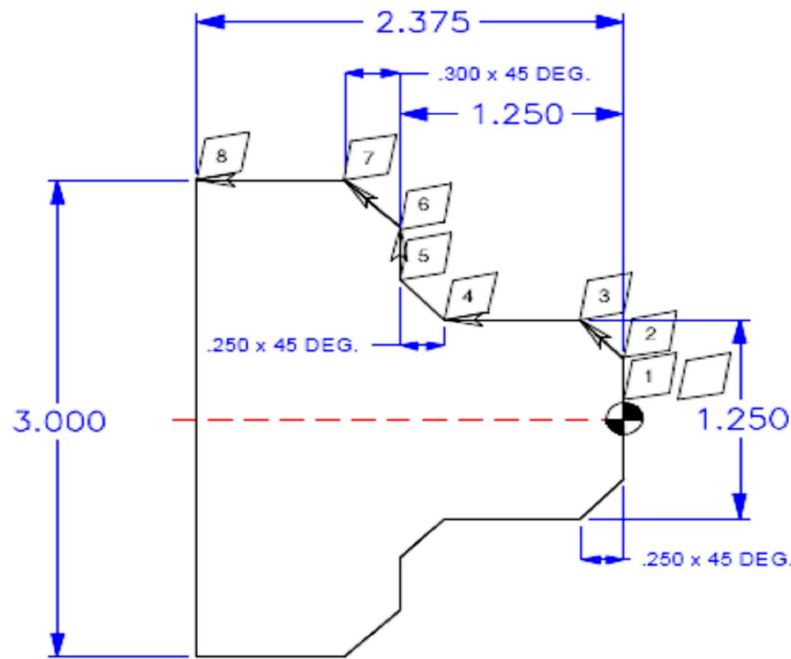
G00 X.9105 Z0.1 (ABSOLUTE) G01 X1.5 Z-1.1 F.006 Z-3.0 X2.0	G00 X.9105 Z0.1 (INCREMENTAL) G01 U.5895 W-1.1 F.006 W-1.9 U.5	G00 X.9105 Z0.1 (ABS. AND INC.) G01 X1.5 W-1.1 F.006 Z-3.0 U.5
------------------------------------------------------------------------	----------------------------------------------------------------------------	----------------------------------------------------------------------------



Linear Interpolation Exercise

FORMAT: G01 (Linear interpolation [feed] Corner Chamfering for Absolute or Incremental).

With what we have learned about G01, we can move a cutter along the part profile by a series of Absolute X and Z-axis movements or Incremental U (X-axis) and W (Z-axis) movements, which can be a combination of both X and Z absolute movements and U and W incremental movements. Define the actual points around this part using both incremental and absolute.



ABSOLUTE PROGRAMMING

```

G00 X.25 Z.1
N1 G__ Z0. F.006
N2 X_____
N3 X_____ Z-_____
N4 Z-_____
N5 X_____ Z-_____
N6 X_____
N7 X_____ Z-_____
N8 Z-_____
    
```

INCREMENTAL AND ABSOLUTE PROGRAMMING

```

G00 X.25 Z.1
N1 G__ Z0. F.006
N2 X_____
N3 U_____ W-_____
N4 Z-_____
N5 U_____ Z-_____
N6 X_____
N7 X_____ W-_____
N8 Z-_____
    
```

Circular Interpolation Commands

G02 CW Circular Interpolation Motion

X	Absolute X-axis Arc end point motion command
Z	Absolute Z-axis Arc end point motion command
U	Incremental X-axis Arc end point motion command
W	Incremental Z-axis Arc end point motion command
I	Distance from start point to arc center in the X-axis
K	Distance from start point to arc center in the Z-axis
R	Radius of circle (If I and K are not used)
F	Feed rate

G03 CCW Circular Interpolation Motion

X	Absolute X-axis Arc end point motion command
Z	Absolute Z-axis Arc end point motion command
U	Incremental X-axis Arc end point motion command
W	Incremental Z-axis Arc end point motion command
I	Distance from start point to arc center in the X-axis
K	Distance from start point to arc center in the Z-axis
R	Radius of circle (If I and K are not used)
F	Feed rate

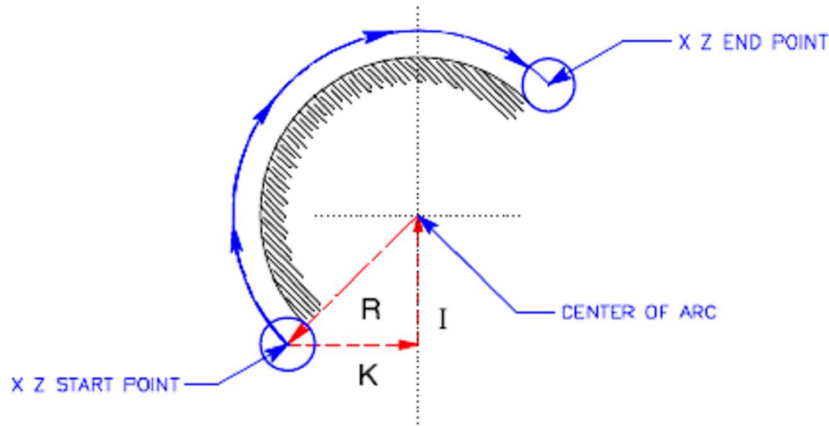
G03 will generate a counterclockwise circular motion, but is otherwise defined the same as a G02 clockwise circular motion.

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: plane selection, arc start position coordinates, rotation direction, arc end position coordinates, and arc center coordinates or arc radius.

There are two basic command formats for defining circular interpolation depending on whether the **I and K method** or the **R method** is used to define the arc center.

R is the distance from the starting point to the center of the arc. X or Z is required to specify an endpoint different from the starting point. 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**.

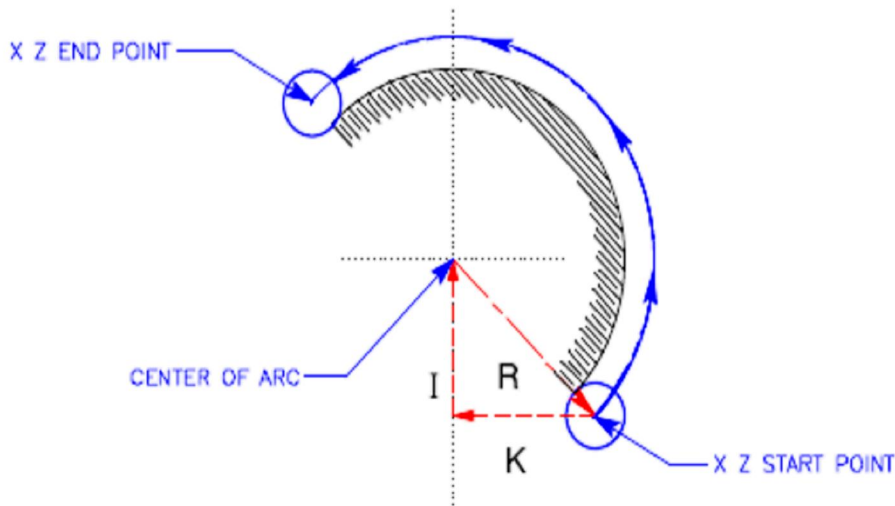
G02 CW Circular Interpolation Motion (Example)



Five pieces of information are required for executing a circular interpolation command.

Plane selection command	G17	Arc parallel to XY-plane (not available)
Plane selection command	G18	Arc parallel to ZX-plane (default)
Plane selection command	G19	Arc parallel to YZ-plane (not available)
Arc start position coordinates	X,Z	Coordinates to the start position of arc
Rotation direction	G02 G03	Clockwise interpolation direction Counterclockwise interpolation direction
Arc end position Absolute or Arc end position Incremental	X,Z U,W	Coordinates of the end position of arc defined from the part zero or part origin Incremental distance and direction from start Point of arc to end point of arc in X and Z- axes
I and K method (arc center coordinates)	I,K	Incremental distance and direction from start position of arc to the arc center for X and Z-axes. "I" is for the X-axis; "K" is for the Z-axis. "I" is a radial value.
R method (arc radius)	R	Arc radius value of arc.

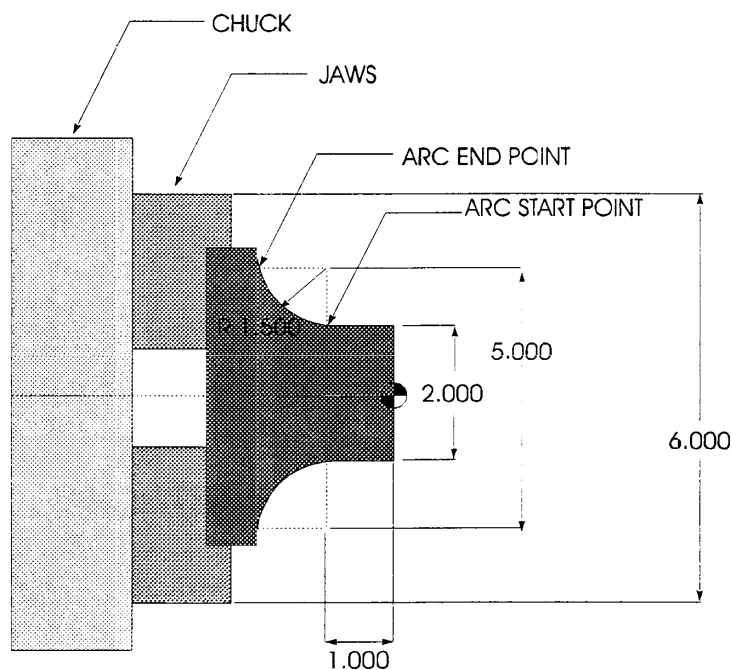
G03 CWW Circular Interpolation Motion (Example)



To move the cutter in a clockwise direction from the START POINT to the END POINT, the programming coding would be like this:

```

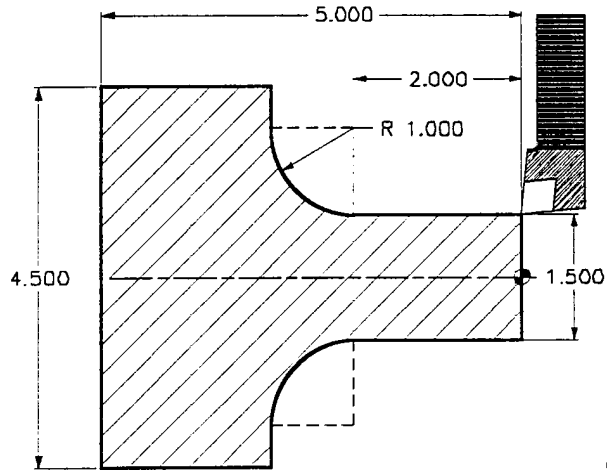
N11.....
N12 G00 X0. Z1.          (tool rapids to X0 Z1 in front of part)
N13 G01 Z0. F012        (tool feeds to Z0. face of part)
N14 X2.                  (tool feeds up in X-axis to a 2. dia.)
N15 Z-1.                 (tool at 2 dia. machines to the arc start point at Z-1.)
N16 G02 X5. Z-2.5 R1.5  (cuts the 1.5 radius to the arc end point at X5. Z-2.5) (I+1.5 K0)
N17 G01 X6.0            (tool feeds up in X-axis to a 6 dia.)
N18.....
    
```



Circular Interpolation Motion Exercises

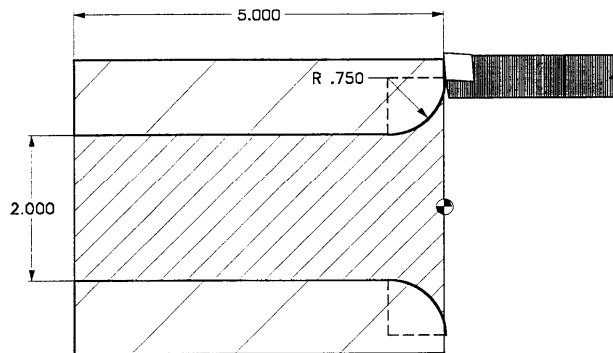
Exercise #1: Exercise for G01 and G02

From the tool's current position, program the cutter path to turn the 1.500 diameter up to the START POINT of the 1.000 radius and then interpolate the radius. Finish the cut by feeding up to the 4.500 diameter.



```

N9 G00 X1.5 Z.100
N10 G01 Z-_____ F.01
N11 G____ X_____ Z_____ I_____ K_____
N12 G____ X_____
    
```



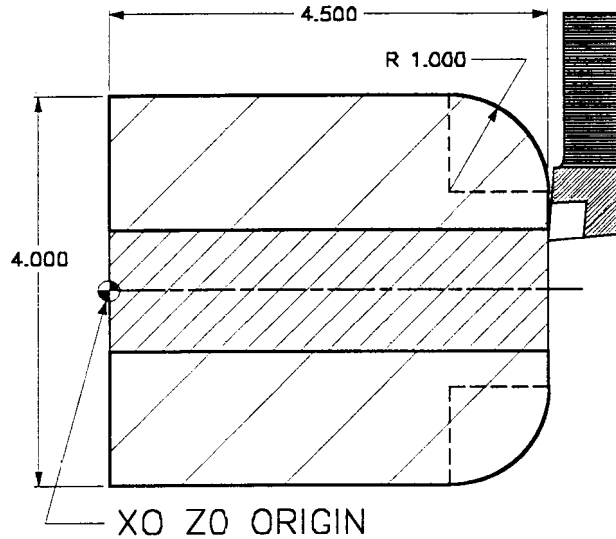
From the tool's current position, feed into face, face down to the START POINT of radius, machine the radius down into the bore, and then bore through to the end of the part.

```

N19 G00 X4.0 Z.10
N20 G01 Z0. F.01
N21 X_____
N22 G____ X_____ Z_____ I_____ K_____
N23 G____ Z_____
    
```

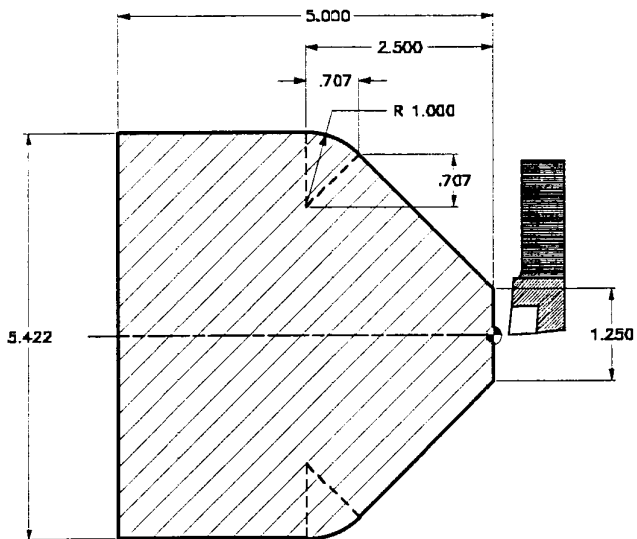
Exercise #2: Exercise for G01 and G03

From the tool's current position, program the cutter path to feed up the face to the START POINT of the 1.0 radius, machine the radius and then turn the 4.000 diameter to the end of the part.



```
N29 G00 X1.0 Z4.6
N30 G01 Z4.5 F.01
N31 X2.0
N32 G__ X__ Z__
I__ K__
N33 G__ Z__
```

From the tool's current position, move into the face, feed up the Feed to face of part at X0, feed up to the 45 degree angle Start going to the START POINT of the radius, machine the start point then the 1.00 radius and turn the 5.422 diameter back to the end of the part. Back to 5.0



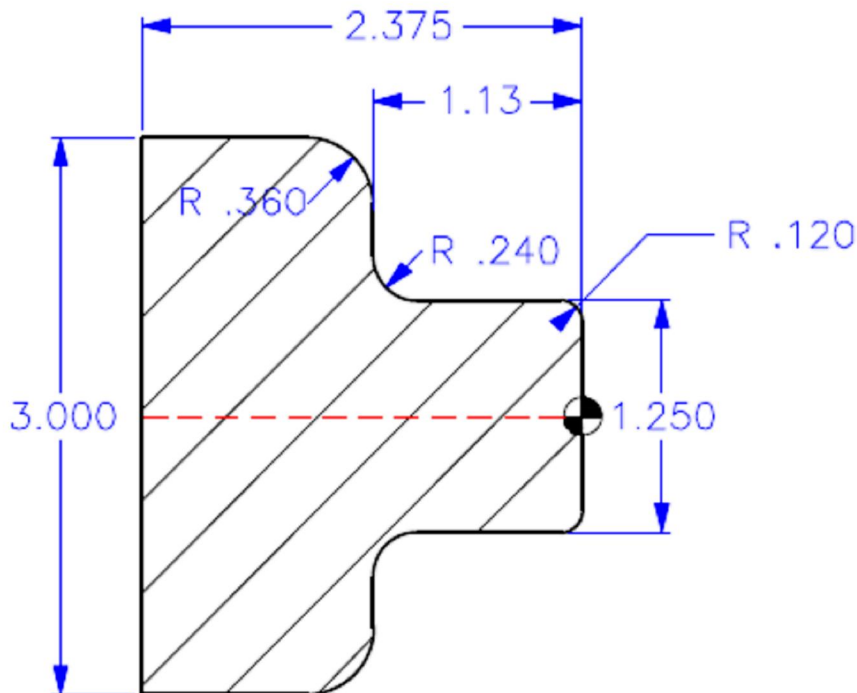
```
N39 G00 X0 Z.100
N40 G01 Z__ F.01
N41 G0__ X__
N42 G01 X__ Z__
N43 G__ X__ Z__ I__
K__
N44 G 01 Z__
```

Circular Interpolation Exercise

FORMAT: G02 and G03 (Circular Interpolation Clockwise/Counterclockwise) using "R" (radius) in place of "I" and "K".

It is sometimes easier to program the "R" (radius) word in place of the "I" and "K" words. The "I" and "K" words are used to define the incremental distance and direction from START POINT to the ARC CENTER, which can also be done by using the "R" word in place of "I" and "K".

NOTE: Any radius being cut using the "R" word needs a positive (+) value for a radius of 180 degrees or less. For a radius larger than 180 degrees, it needs to have a minus (-) sign with a radius value.



Program Example **(G02 & G03 using I and K)**

```
G01 Z0  
X1 .01  
G03 X1.25 Z-.12 I ____ K ____  
G01 Z-.89  
G02 X1.73 Z-1.13 I ____ K ____  
G01 X2.28  
G03 X3. Z-1.49 I ____ K ____  
G01 Z-2.375
```

Program Example **(G02 & G03 using R)**

```
G01 Z0  
X1 .01  
G03 X1.25 Z-.12 R ____  
G01 Z-.89  
G02 X1.73 Z-1.13 R ____  
G01 X2.28  
G03 X3. Z-1.49 R ____  
G01 Z-2.375
```

Manually Programming Tool Nose Compensation

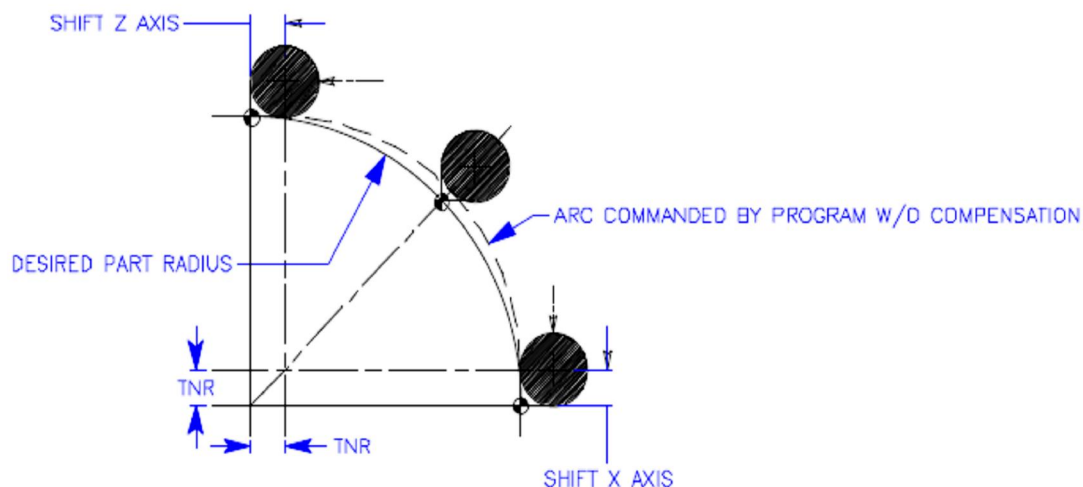
This section is to help those who have not had the time or patience to learn to use TNC. Our suggestion is, if you want to save time and money to learn how to use TNC.

Calculating Compensation for a Radius on Your Part

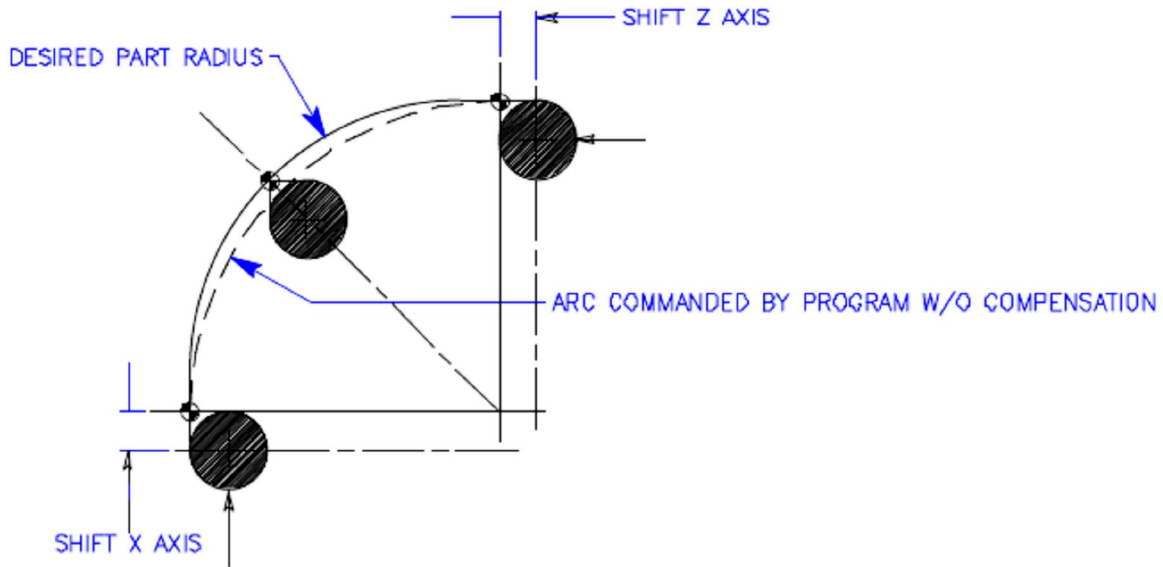
When you program a straight line in either X or Z, the tool tip touches the part at the same point where you touched your original tool offsets in X and Z. However, when you program a radius, the theoretical tool tip does not touch the part radius. Where the tip actually touches the part is dependent upon the radius of the tool and the point around the radius being cut. If you were to try to program a part without using any compensation and you programmed to the finish radius size of your part, you would see a smaller radius on the outside corners of your part and larger radiuses on the fillet radiuses of your part. The amount that will be either a smaller corner radius or a larger fillet radius will be the amount that is the radius of the tool you are machining.

Refer to the illustrations on the pages while reading the text below.

For a 90-degree corner radius, calculate the correct tool path position for a tool with a .031 TNR in your program. Manually calculate the compensated path by adding .031 to the radius to be machined. You will also need to re-calculate the start point and end point of this larger programmed radius.



For a 90-degree fillet radius, calculate the correct tool path position for a tool with a .031 TNR in your program. Manually calculate the compensated path by subtracting .031 from the radius to be machined. You will also need to re-calculate the start point and end point of this smaller programmed radius.

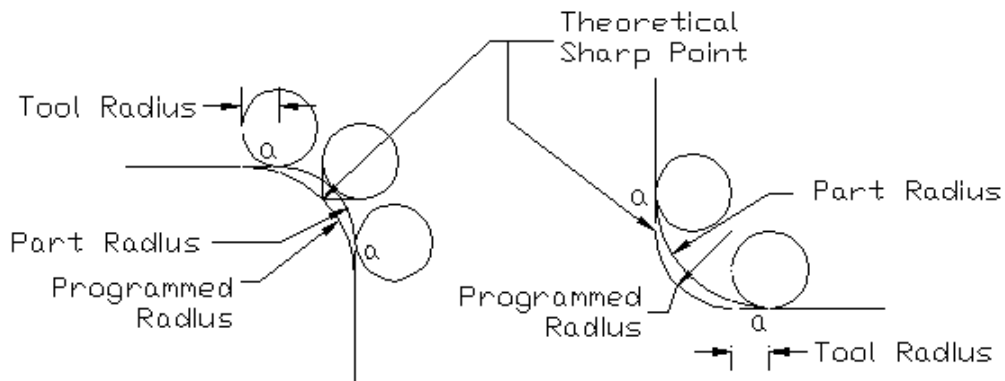


Radius Calculation

When a radius is required TNRC has to be applied. The basic rule is as follows:

- An external radius has the Tool Nose Radius added to it.**
- An internal radius has the Tool Nose Radius subtracted from it.**

The example below shows this:



External Radius Calculation

If a radius of .25 is required on the external corner of a part and the tool nose radius is .031 the programmed radius will be:

$$.25 + .031 = .281$$

If the radius is on a 1.0 diameter and at the Z0 face of the part the program will follow the example below.

G1 X0 Z0	Start of program
G1 X.438	$1.0 - (.281 \times 2) = .438$
G3 X1.0 Z-.281 R.281	Move to the X and Z axis end point
G1 Z-?	Next axis parallel move

Internal Radius Calculation

If a radius of .25 is required on the internal corner of a part and the tool nose radius is .031 the programmed radius will be:

$$.25 - .031 = .219$$

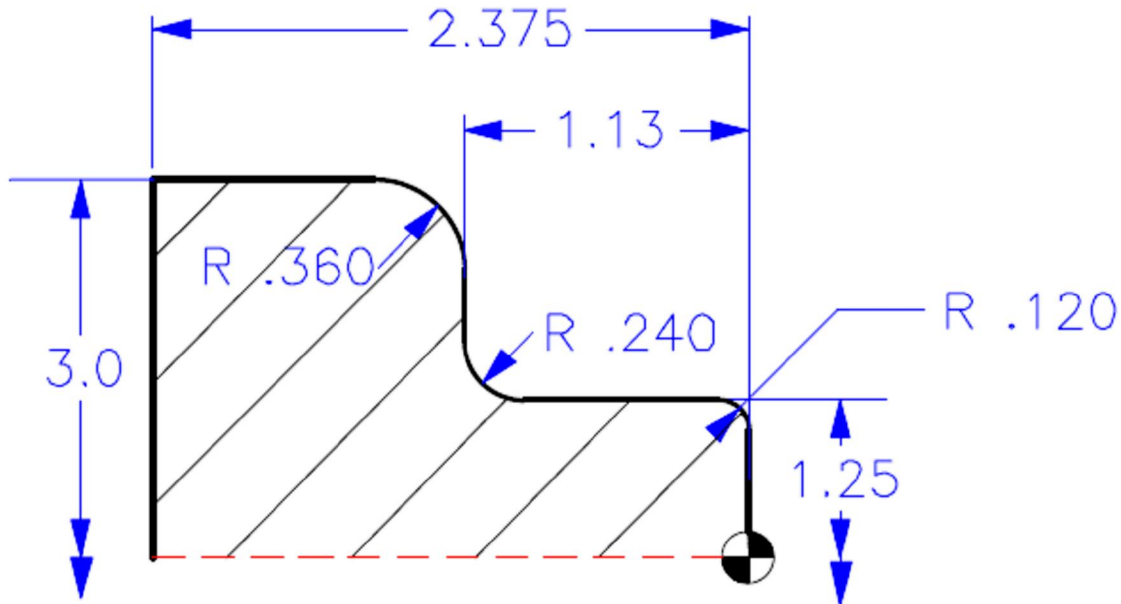
If the radius is on a 1.0 diameter and at the Z0 face of the part the program will follow the example below.

G1 X1. Z0
G1 Z-.781
G2 X 1.438 Z-1. R.219
G1 X-?

Note: When calculating the end points in a part program you must **double the X axis** calculation to allow for both sides of the part. Any trigonometry **calculation is only generating a radial value** for X.

Circular Interpolation Calculation

Manually calculate compensation for a lathe tool with a .031 TNR to machine a radius on your part.



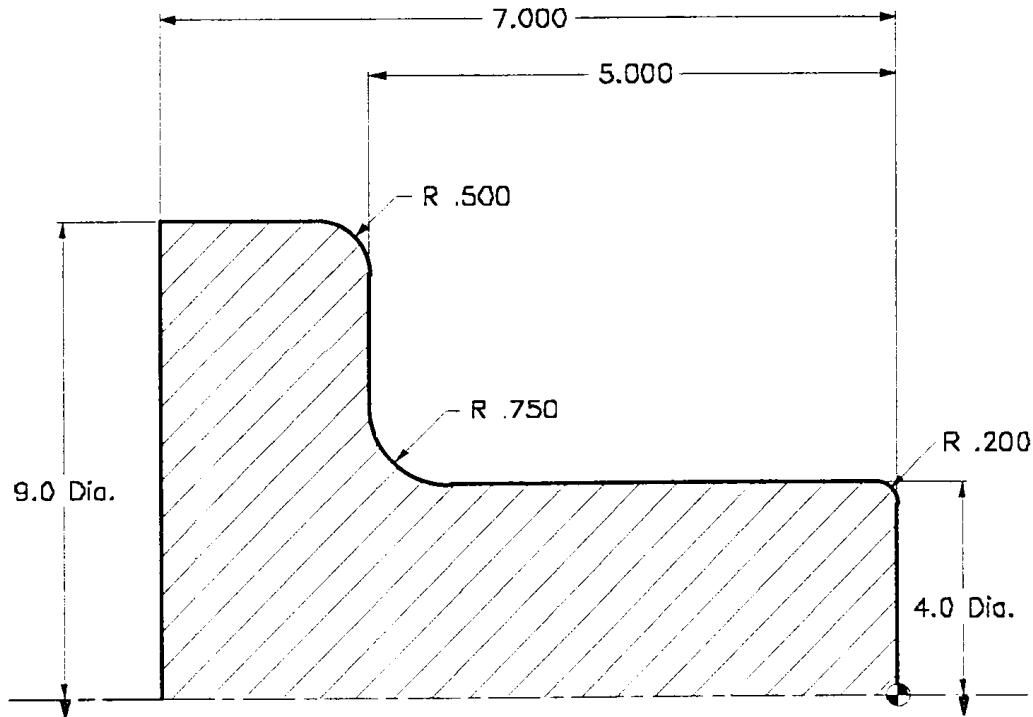
Manual Programming A Radius (G02 & G03 using I and K)

```
O0102
G28
T101
G97 S1200 M03
G54 G00 X.92 Z.05
G01 Z0. F.01
X.948 F.006
G03 X1.25 Z-.151 I0. K-.151
G01 Z-.921
G02 X1.668 Z-1.13 I.209 K0.
G01 X2.218
G03 X3. Z-1.521 I0. K-.391
G01 Z-2.375
G00 U.01 Z1.0
G28
```

Manual Programming A Radius (G02 & G03 using R)

```
O0103
G28
T101
G97 S1200 M03
G54 G00 X.92 Z.05
G01 Z0. F.01
X.948 F.006
G03 X1.25 Z-.151 R.151
G01 Z-.921
G02 X1.668 Z-1.13 R.209
G01 X2.218
G03 X3. Z-1.521 R.391
G01 Z-2.375
G00 U.01 Z1.0
G28
```


Circular Interpolation Exercise



Fill in the blanks to complete one finish pass of the cutter path for the above part defining the actual line of part "without" using cutter compensation (G41 or G42) and adding in the actual radius compensation for a .031 radius O.D. turning tool.

O0010

(Circular Interpolation "Without" Cutter Compensation)

N1 (Finish O.D.)

G28

T101 (O.D. Turning Tool .031 TNR)

G50 S2200

G97 S291 M03

G54 G00 X4.2 Z0.1 M08

G96 S750

G00 Z0.

G01 X-0.062 F0.004

G0 X3.5 Z0.02

G01 Z0. F0.005

X_____

G03 X_____ Z_____ R_____

G01 Z_____

G02 X_____ Z_____ R_____

G01 X_____

G03 X_____ Z_____ R_____

G01 Z_____

G00 U0.02 Z1. M09

G97 S320

G28

M30

Calculating Compensation for “An Angle” On Your Part

When you program a straight line in either X or Z, the tool tip touches the part at the same point where you touched your original tool offsets in X and Z. However, when you program a chamfer or an angle, the tip does not touch the part at those same points. Where the tip actually touches the part is dependent upon the degree of angle being cut and the size insert you are using. If you were to try to program your part without using any compensation you would see overcutting and undercutting on your part.

The following pages contain tables and illustrations demonstrating how to calculate the compensation in order to program your part accurately. There are two tables with 1/32 and 1/64 insert radius values in both the X and Z-axes.

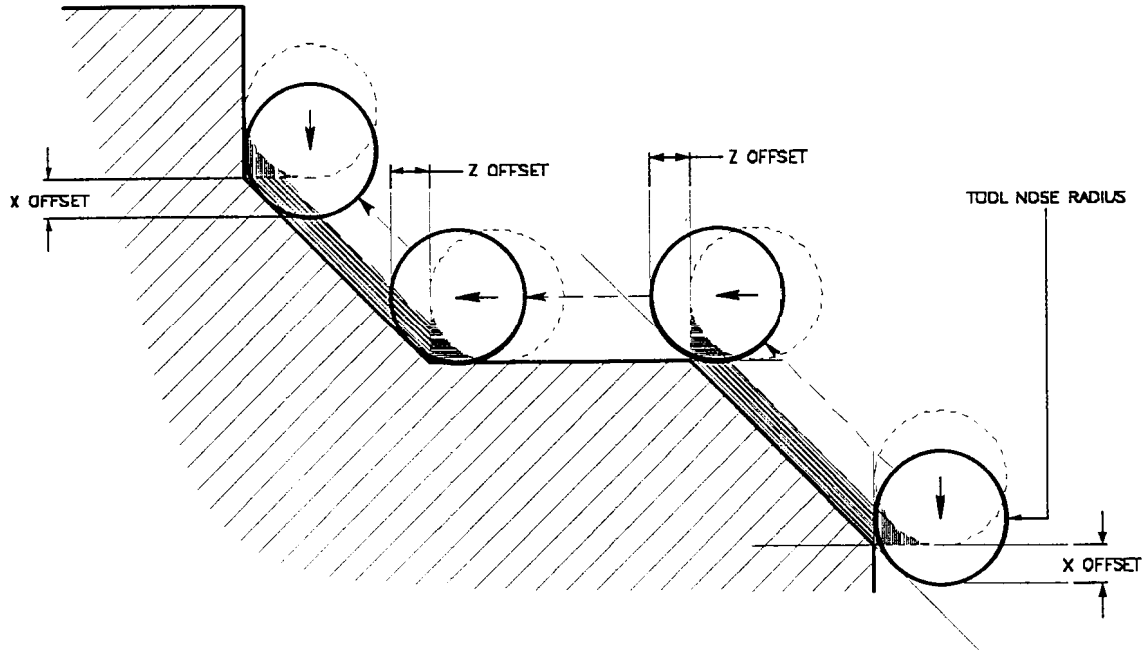
Along with the charts is an illustration example of compensation using offset values added to your program moves. Next is a program exercise using these values to define your part for machining an angle.

Taper Calculation

To simplify tool radius compensation on tapers a chart is provided that gives a range of radii and angle between 0°-45° if a radius greater than 45° is encountered simply use the opposite angle in the triangle and reverse the Xc and Zc values.

Tool Nose Radius Calculation Diagram

Machining a chamfer when cutter compensation is NOT used on the control requires that calculations must be made for the tool tip geometry for the programmed moves on your part angles.



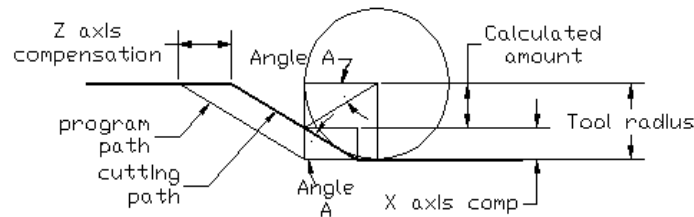
<u>ANGLE</u>	<u>X-OFFSET</u> .015 N/R	<u>Z-OFFSET</u> .015 N/R	<u>X-OFFSET</u> .031 N/R	<u>Z-OFFSET</u> .031 N/R
15	.0072	.0136	.0146	.0271
30	.0132	.0114	.0264	.0229
45	.0184	.0092	.0366	.0183
60	.0228	.0066	.0458	.0132

X-OFFSET numbers above are diametric.

Tool Nose Compensation – Taper Calculation

If an angle has to be calculated the formula is described below:

$$\text{TNRC for X} = R - (\tan \theta \times R)$$



If the angle is 30° and the radius is .03125, the X-axis calculation is:

$$\begin{aligned} X &= R - (\tan \theta \times R) \\ X &= .03125 - (\tan 30 \times .03125) \\ X &= .03125 - (.57735 \times .03125) \\ X &= .03125 - (.01804) \\ X &= .013207 \end{aligned} \quad (\text{Note this is a radial number})$$

$$\text{TNRC for Z} = \frac{\text{Compensation in X}}{\tan \theta}$$

The Z-axis calculation is:

$$\begin{aligned} Z &= \frac{.013207}{.57735} \\ Z &= .02287 \end{aligned}$$

If these calculations are compared to your chart, the numbers match.

In practice, it is far easier to use a chart and apply one axis of compensation, and then calculate the end point of the other axis. This method ensures that the correct angle is machined, and any rounding of numbers is taken into consideration.

The following tables are taken from SL-Series Programming and Operation Manual.

Note that Xc values do not have to be multiplied by 2 as they are already calculated as diametric numbers.

TOOL RADIUS AND ANGLE CHART 1/32 RADIUS

NOTE: THE MEASUREMENT CALCULATED IS BASED ON PART DIAMETER

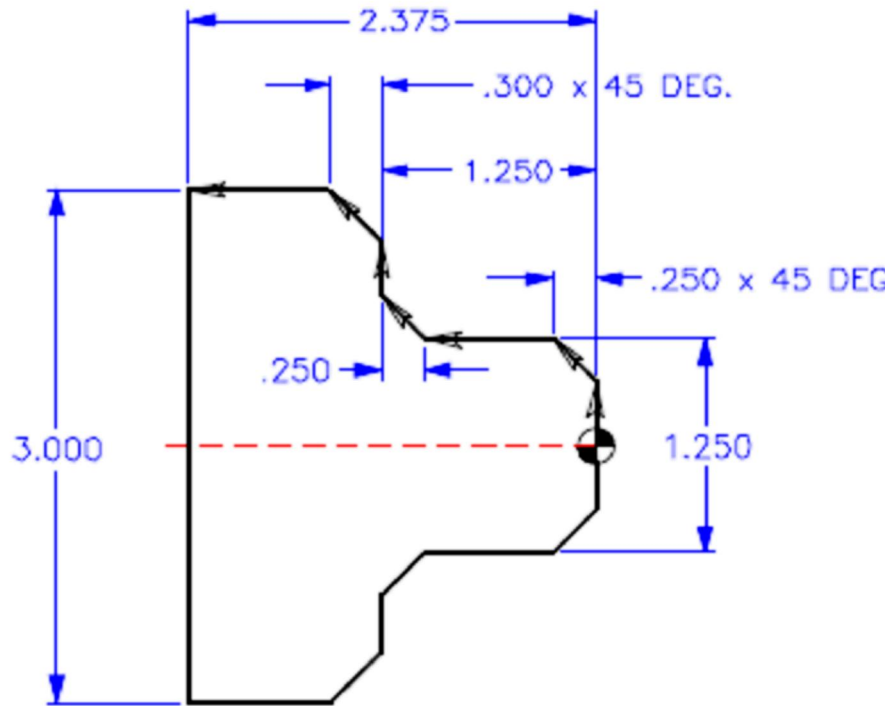
ANGLE	Xc CROSS	Zc LONGITUDINAL	ANGLE	Xc CROSS	Zc LONGITUDINAL
1.	.0010	.0310	46.	.0372	.0180
2.	.0022	.0307	47.	.0378	.0177
3.	.0032	.0304	48.	.0386	.0173
4.	.0042	.0302	49.	.0392	.0170
5.	.0052	.0299	50.	.0398	.0167
6.	.0062	.0296	51.	.0404	.0163
7.	.0072	.0293	52.	.0410	.0160
8.	.0082	.0291	53.	.0416	.0157
9.	.0092	.0288	54.	.0422	.0153
10.	.01	.0285	55.	.0428	.0150
11.	.0011	.0282	56.	.0434	.0146
12.	.0118	.0280	57.	.0440	.0143
13.	.0128	.0277	58.	.0446	.0139
14.	.0136	.0274	59.	.0452	.0136
15.	.0146	.0271	60.	.0458	.0132
16.	.0154	.0269	61.	.0464	.0128
17.	.0162	.0266	62.	.047	.0125
18.	.017	.0263	63.	.0474	.0121
19.	.018	.0260	64.	.0480	.0117
20.	.0188	.0257	65.	.0486	.0113
21.	.0196	.0255	66.	.0492	.0110
22.	.0204	.0252	67.	.0498	.0106
23.	.0212	.0249	68.	.0504	.0102
24.	.022	.0246	69.	.051	.0098
25.	.0226	.0243	70.	.0514	.0094
26.	.0234	.0240	71.	.052	.0090
27.	.0242	.0237	72.	.0526	.0085
28.	.025	.0235	73.	.0532	.0081
29.	.0256	.0232	74.	.0538	.0077
30.	.0264	.0229	75.	.0542	.0073
31.	.0272	.0226	76.	.0548	.0068
32.	.0278	.0223	77.	.0554	.0064
33.	.0286	.0220	78.	.056	.0059
34.	.0252	.0217	79.	.0564	.0055
35.	.03	.0214	80.	.057	.0050
36.	.0306	.0211	81.	.0576	.0046
37.	.0314	.0208	82.	.0582	.0041
38.	.032	.0205	83.	.0586	.0036
39.	.0326	.0202	84.	.0592	.0031
40.	.0334	.0199	85.	.0598	.0026
41.	.034	.0196	86.	.0604	.0021
42.	.0346	.0193	87.	.0608	.0016
43.	.0354	.0189	88.	.0614	.0011
44.	.036	.0186	89.	.062	.0005
45.	.0366	.0183			

TOOL RADIUS AND ANGLE CHART 1/64 RADIUS

NOTE: THE X MEASUREMENT CALCULATED IS BASED ON PART DIAMETER

ANGLE	Xc CROSS	Zc LONGITUDINAL	ANGLE	Xc CROSS	Zc LONGITUDINAL
1.	.0006	.0155	46.	.00186	.0090
2.	.0001	.0154	47.	.0019	.0088
3.	.0016	.0152	48.	.0192	.0087
4.	.0022	.0151	49.	.0196	.0085
5.	.0026	.0149	50.	.0198	.0083
6.	.0032	.0148	51.	.0202	.0082
7.	.0036	.0147	52.	.0204	.0080
8.	.0040	.0145	53.	.0208	.0078
9.	.0046	.0144	54.	.021	.0077
10.	.0050	.0143	55.	.0214	.0075
11.	.0054	.0141	56.	.0216	.0073
12.	.0060	.0140	57.	.022	.0071
13.	.0064	.0138	58.	.0222	.0070
14.	.0068	.0137	59.	.0226	.0068
15.	.0072	.0136	60.	.0228	.0066
16.	.0078	.0134	61.	.0232	.0064
17.	.0082	.0133	62.	.0234	.0062
18.	.0086	.0132	63.	.0238	.0060
19.	.0090	.0130	64.	.024	.0059
20.	.0094	.0129	65.	.0244	.0057
21.	.0098	.0127	66.	.0246	.0055
22.	.0102	.0126	67.	.0248	.0053
23.	.0106	.0124	68.	.0252	.0051
24.	.011	.0123	69.	.0254	.0049
25.	.0014	.0122	70.	.0258	.0047
26.	.0118	.0120	71.	.0260	.0045
27.	.012	.0119	72.	.0264	.0043
28.	.0124	.0117	73.	.0266	.0041
29.	.0128	.0116	74.	.0268	.0039
30.	.0132	.0114	75.	.0272	.0036
31.	.0136	.0113	76.	.0274	.0034
32.	.014	.0111	77.	.0276	.0032
33.	.0142	.0110	78.	.0280	.0030
34.	.0146	.0108	79.	.0282	.0027
35.	.015	.0107	80.	.0286	.0025
36.	.0154	.0103	81.	.0288	.0023
37.	.0156	.0104	82.	.029	.0020
38.	.016	.0102	83.	.0294	.0018
39.	.0164	.0101	84.	.0296	.0016
40.	.0166	.0099	85.	.0298	.0013
41.	.017	.0098	86.	.0302	.0011
42.	.0174	.0096	87.	.0304	.0008
43.	.0176	.0095	88.	.0308	.0005
44.	.018	.0093	89.	.031	.0003
45.	.0184	.0092			

**Angular Straight Line Interpolation Exercise
- Manually Adding Radius Compensation**

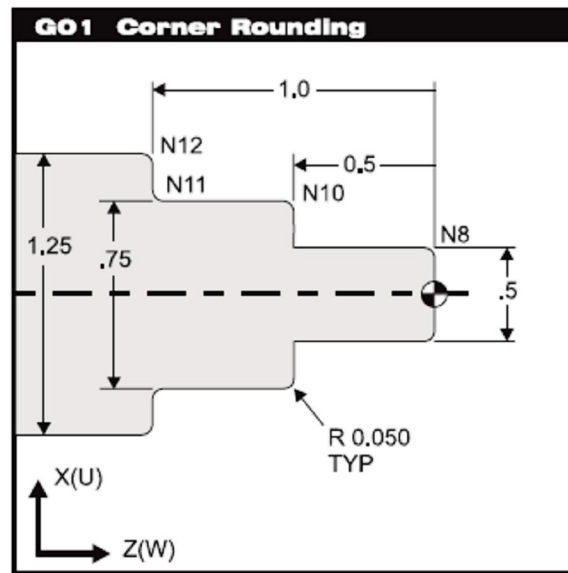


**Manual Programming a Finish Pass
With an Angle using a .031 TNR:**

```

O0020
T101
G97 S1200 M03
G54 G00 X.85 Z.1 M08
G01 Z0. F.03
X_____ F.01 (Face)
G00 X_____ Z.02
G__ Z0. F.006
X_____
X_____ Z-_____
Z-_____
X_____ Z-_____
X_____
X_____ Z-_____
Z-_____
G00 U.01 Z1.0 M09
G28
  
```

AUTOMATIC G01 CORNER ROUNDING



AUTOMATIC CORNER ROUNDING PROGRAM USING R EXAMPLE

```

O00042 (Linear G01 with Radius using R)
N1 G53 G00 X0. Z0. T0
N2 T101 (O.D. TOOL x .031 TNR)
N3 G50 S3000
N4 G97 S3000 M03
N5 G54 G00 X0.3 Z0.1 M08
N6 G96 S390
N7 G42 G01 Z0. F0.01
N8 G01 X0.5 R-0.05
N9 G01 Z-0.5
N10 G01 X0.75 R-0.05
N11 G01 Z-1.0 R0.05
N12 G01 X1.25 R-0.05
N13 G01 Z-1.5
N14 G40 G00 U0.01 Z0.1 M09
N15 G53 G00 X0. Z0. T0
N16 M30
    
```

(The R command is for a 90-degree corner radius from X into Z, or Z into X.)

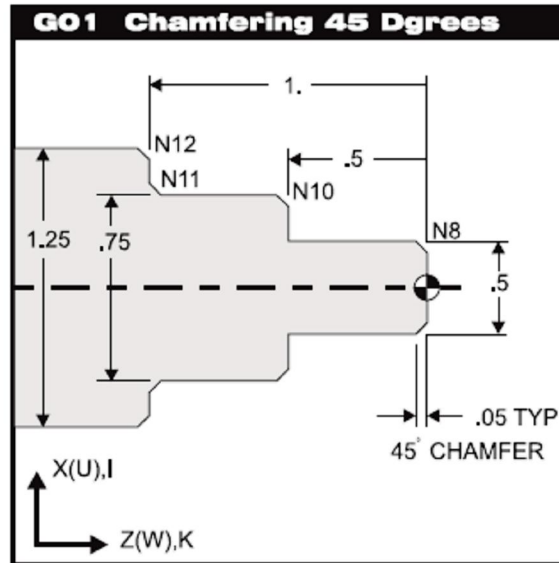
(R+ is to machine a radius into the plus direction on an axis.)

(R- is to machine a radius into the minus direction on an axis.)

Rules:

- 1) The linear G01 block must be a single X(U) or Z(W) move with an R that is perpendicular to the previous move for corner rounding.
- 2) When using R for corner rounding, do not use I, K, or A.
- 3) Caution: This Radius (R) command is not supported in any of the roughing passes of a G71 or G72 canned cycles. The last pass of the G71 and G72 will be executed with the radius defined with R in the G01 command.
- 4) A G70 or G73 will support this type of radius command.
- 5) The + or - sign define the direction of the axis your moving into at the end of the arc move.

AUTOMATIC G01 CHAMFERING 45 DEGREE ANGLES



AUTOMATIC CHAMFERING PROGRAM EXAMPLE

O00043 (Linear G01 with 45 Degree Chamfer using I or K)

```

N1 G53 G00 X0. Z0. T0
N2 T101 (O.D. TOOL x .031 TNR)
N3 G50 S3000
N4 G97 S3000 M03
N5 G54 G00 X0.3 Z0.1 M08
N6 G96 S390
N7 G42 G01 Z0. F0.01
N8 G01 X0.5 K-0.05
N9 G01 Z-0.5
N10 G01 X0.75 K-0.05
N11 G01 Z-1.0 I0.05
N12 G01 X1.25 K-0.05
N13 G01 Z-1.5
N14 G40 G00 U0.01 Z0.1 M09
N15 G53 G00 X0. Z0. T0
N16 M30
    
```

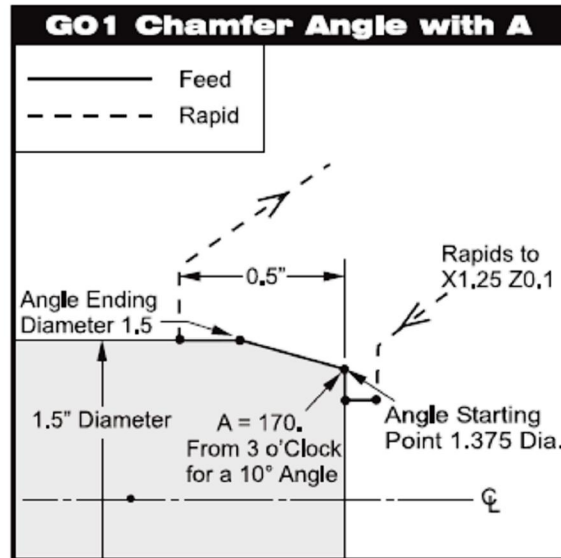
(I is for the 45-degree chamfer size from Z into X. I+ is up, I- is down into the X axis.)

(K is for the 45-degree chamfer size from X into Z. K- is left, K+ is right into the Z axis.)

Rules:

- 1) The linear G01 block must be a single X(U) or Z(W) move with an I or K that is perpendicular to the previous move for chamfering.
- 2) When using I or K for chamfering, do not use A or R.
- 3) Caution: This chamfer (I,K) command is not supported in any of the roughing passes of a G71 or G72 canned cycles. The last pass of the G71 and G72 will be executed with the chamfer defined with I and K in the G01 command.
- 4) A G70 or G73 will support this type of chamfer command.
- 5) The + or - sign defines the direction of the axis your moving into at the end of chamfer move.

AUTOMATIC G01 CHAMFERING WITH ANGLE



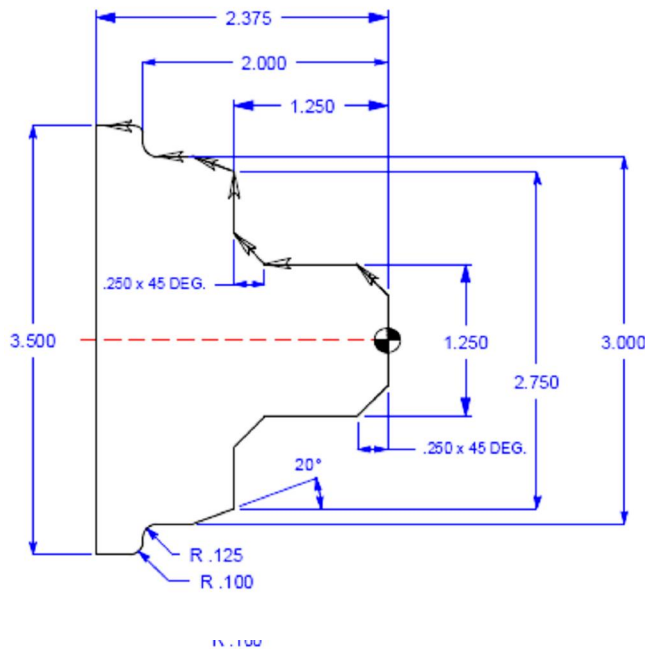
AUTOMATIC CHAMFERING PROGRAM EXAMPLE

```
O00044 (Linear G01 Chamfer 10 Degree Angle using A)
N1 G53 G00 X0. Z0. T0
N2 T101 (O.D. TOOL x .031 TNR)
N3 G50 S2800
N4 G97 S1490 M03
N5 G54 G00 X1.25 Z0.1 M08
N6 G96 S390
N7 G42 G01 Z0. F0.01
N8 G01 X1.375 (Start point)
N9 G01 X1.5 A170. (10 Degree Angle using A)
N10 Z-0.5
N11 G40 G00 U0.01 Z0.1 M09
N12 G53 G00 X0. Z0. T0
N13 M30
```

Rules:

- 1) The linear G01 block must be a single X(U) or Z(W) move that is perpendicular to the previous move with an A to do a specific angle.
- 2) When using A for an angle, do not use I, K or R.
- 3) Caution: This angle (A) command is "not supported in any of the roughing passes of a G71 or G72 canned cycles, though the last pass in the G71 and G72 will be executed with the angle defined with A in the G01 command.
- 4) A G70 or G73 will support this type of chamfer command.
- 5) You can use a minus value to define an angle clockwise from three o'clock: A-30. = A150. and A-45. = A135.
- 6) Be sure to enter in a decimal point for angles.

CHAMFER & CORNER ROUNING WITH G01 EXERCISE



This G code provides for straight line (linear) motion from point to point that can have a chamfer or radius between 2 blocks of linear motion that intersect at a 90 degree angle. This type of command is programmed with a chamfer or corner rounding move by providing a letter command of I, K, R, or A in a G01 command block.

Note: This G01 Code using I, K, R, or A, to do an arc or angle, is not recognized in the roughing passes of the canned cycles G71 or G72.

Interpolation with G01, G02 and G03

O00045

N11 G53 G00 X0. Z0. T0

N12 T101 (O.D. TOOL x .031 TNR)

N13 G50 S2600

N14 G97 S2057 M03

N15 G54 G00 X0.65 Z0.1

N16 G96 S350

N17 G42 G01 Z0. F0.006

N18 X.75

N19 X1.25 Z-0.25

N20 Z-1.0

N21 X1.75 Z-1.25

N22 X2.75

N23 X3. Z-1.5934 (Calculated Z move)

N24 Z-1.875

N25 G02 X3.25 Z-2.0 R0.125

N26 G01 X3.3

N27 G03 X3.5 Z-2.1 R0.1

N28 G01 Z-2.375

N29 G00 G40 U0.01 Z0.1 M09

N30 G53 G00 X0. Z0. T0

Interpolation with G01 using I, K, R & A

O00050

N31 G53 G00 X0. Z0. T0

N32 T101 (O.D. TOOL x .031 TNR)

N33 G50 S2600

N34 G97 S2057 M03

N35 G54 G00 X0.65 Z0.1

N36 G96 S350

N37 G___ G42 Z0. F0.006

N38 X_____ K_____ (K+ or - ?)

N39 Z-_____ I_____ (I+ or - ?)

N40 X_____ (To X start point for A)

N41 X_____ A_____ (A+ or - ?)

N42 Z-_____ R_____ (R+ or - ?)

N43 X_____ R_____ (R+ or - ?)

N44 Z-_____

N45 G40 G00 U0.01 Z0.1 M09

N46 G53 G00 X0. Z0. T0

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; coolant and spindle will remain on. If the P has a fraction part, the delay is in milliseconds (0.01 seconds); otherwise, the delay is in seconds.

EXAMPLE: G04 P1.0 (for a delay of 1.0 second)
 or
 G04 P25 (for a delay of 25 milliseconds)

INCH / METRIC SELECTION (G20, G21)

Selecting between inch and metric programming can only be done from the Setting page, Setting 9. When changing the setting from inches to metric all work offsets and tool offsets will be converted to mm. The machine will not convert the programs however to metric. This must be done by the programmer.

The G codes G20 and G21 are sometimes used to select between inch and metric. In the HAAS control, the G20 (inch) and G21 (mm) codes can only be used to make sure that the inch/metric setting is set correctly for that program.

Reference Point Definition and Return

G28 RETURN TO MACHINE ZERO AND CANCEL OFFSET (FANUC)

The G28 code is used to machine zero position on all axes. If an X or Z-axis is specified on the same block, only those axes will move and return to the machines' zero point. If X or Z specifies a different location than the current position, then the movement to machine zero will be through the specified point. If no X or Z is specified, all axes will be moved directly to machine zero along with the tailstock. If you do not want to position through an intermediate point while specifying an X or Z-axis to position to machine home, then use an incremental U0 and/or W0 command 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 will go directly to machine zero. G28 also cancels tool offsets.

G51 RETURN TO MACHINE ZERO AND CANCEL OFFSET (YASNAC)

G51, used on a YASNAC control, cancels out any existing tool wear and work coordinate shift offsets and then returns to the machine zero position.

Spindle Speed Commands

G50 SPINDLE SPEED CLAMP

G50 can be used to clamp maximum spindle speed. The control will not allow the spindle to exceed the S address value specified in the most recent G50 command. This is most often used in constant surface feed mode G96.

N1	G50 S2500;	(Spindle RPM will not exceed 2500 RPM)
N2	G96 S650 M03;	(Enter constant surface speed mode, spindle on)

The effect of centrifugal force on its gripping force, unbalanced condition of the work piece, etc., can be constrained by programming a G50 code.

The MAXIMUM spindle speed is designated by the "S" word along with the G50 preparatory command and must be in the same block.

Once the MAXIMUM spindle speed is established, any direct RPM programmed in G97 (direct revolution per minute) or control-calculated RPM from the G96 (constant surface footage mode) exceeding the RPM established by the G50 block is ignored and the G50 "S" word spindle speed is used.

The second use of G50 is for older equipment not having work zero and geometry offset capabilities. This will not be covered because it is not desirable or as easy to use.

G96 CONSTANT SURFACE SPEED ON

G96 commands the control to maintain a constant surface speed of the part relative to the tool tip. Surface speed is based on the distance of the tool tip to the spindle center (radius of the cut). Constant surface speed is maintained by adjusting the spindle speed based on the radius of the cut. The current S code is used to determine the surface speed. G96 is modal.

G97 CONSTANT SURFACE SPEED OFF

G97 commands the control to NOT adjust the spindle speed based on the radius of cut. It is used to cancel any current G96 command. When G97 is in effect, any S command is in revolution per minute (RPM). G97 is modal.

Work Coordinate System Selection

The HAAS CNC lathe control supports both YASNAC and FANUC coordinate systems. Work coordinates, together with tool offsets, can be used to position a part program anywhere within the work cell with great flexibility.

G52 CHILD OR LOCAL COORDINATE SYSTEM

G52 selects the child coordinate system. It is non-modal and FANUC compatible. M30, Reset or power down will remove any values in G52

G53 NON-MODAL MACHINE COORDINATE SELECTION

G53 temporarily cancels work coordinates offset and uses the machine coordinate system. It is non-modal, therefore, the next block will revert to the work coordinate that was active.

G54-59 SELECT COORDINATE SYSTEM #1 THRU #6

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

G110-G129 COORDINATE SYSTEM #7 THRU #26 OR G154P1 to G154 P99

Feed Command Functions

G98 FEED PER MINUTE

G98 affects how the F address code is interpreted. The value of F indicates inches per minute when Setting 9 is set to INCH and millimeters-per-minute when Setting 9 is set to METRIC. This code is modal.

$$\text{IPM} = \text{Current RPM} \times \text{IPR}$$

G99 FEED PER REVOLUTION

G99 affects how the F address is interpreted. The value of F indicates inches-per-revolution of the spindle when Setting 9 is set to INCH and millimeters-per-revolution of the spindle when Setting 9 is set to METRIC. This code is modal and is the default feed mode.

$$\text{IPR} = \text{IPM} / \text{Current RPM}$$

Tool Nose Compensation G Codes

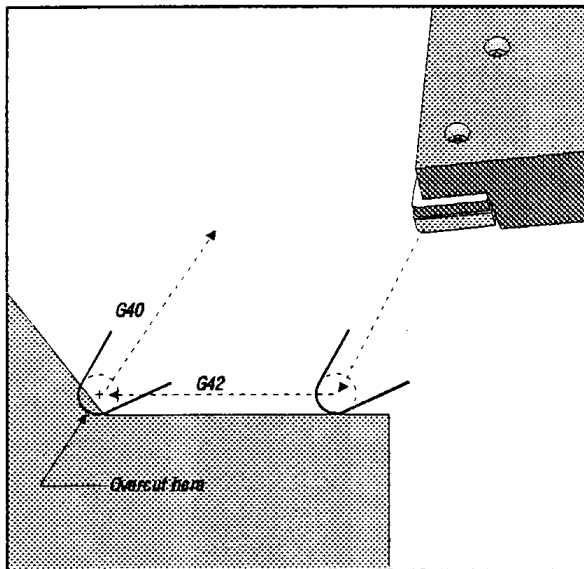
G40 TOOL NOSE COMPENSATION CANCEL

X	X-axis absolute departure location
Z	Z-axis absolute departure location
U	X-axis incremental departure distance
W	Z-axis incremental departure distance
I	X-axis intersection vector direction, (radius)
K	Z-axis intersection vector direction

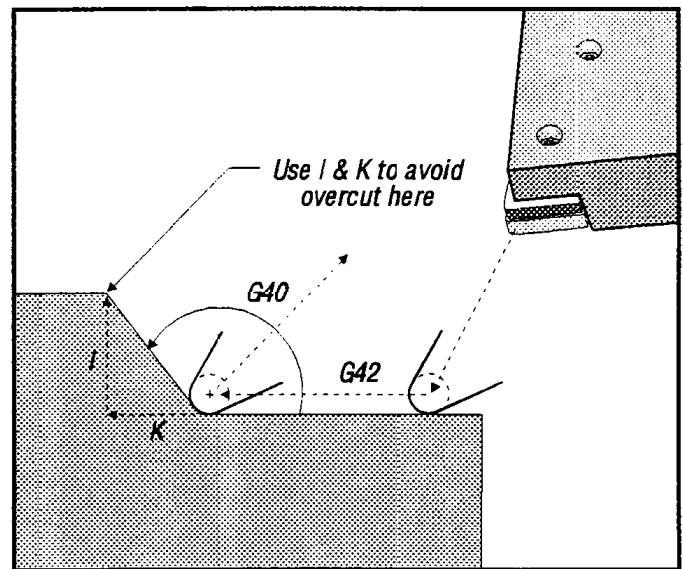
G40 cancels G41 or G42. Programming Txx00 also cancels tool nose compensation. You must always cancel tool nose compensation before the end of a program.

The departure target location usually does not correspond with a point on the part. In many cases, overcutting or undercutting can occur. Figure 3 illustrates this.

When address codes I and K are used in a G40 departure block, the control will use these values as an intersection vector for the end point of the last, completely compensated, motion stroke. Figure 4 illustrates where I and K lie in relation to the departure stroke. Usually I and K lie along a face of the machined part.



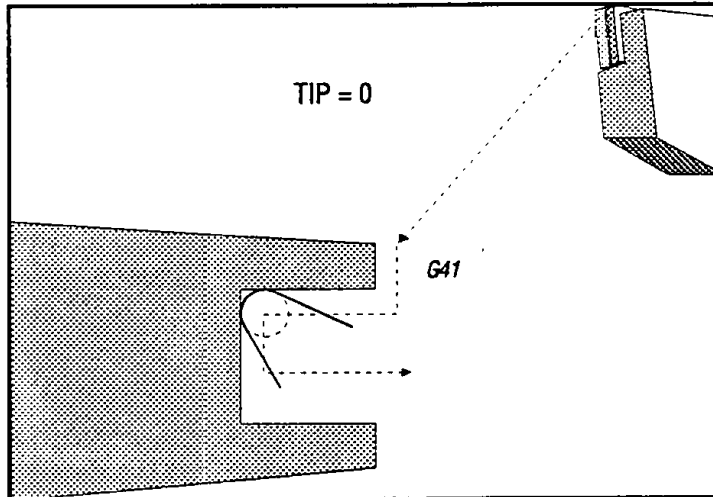
G40 without I and K



G40 with I and K

G41 TOOL NOSE COMPENSATION – LEFT

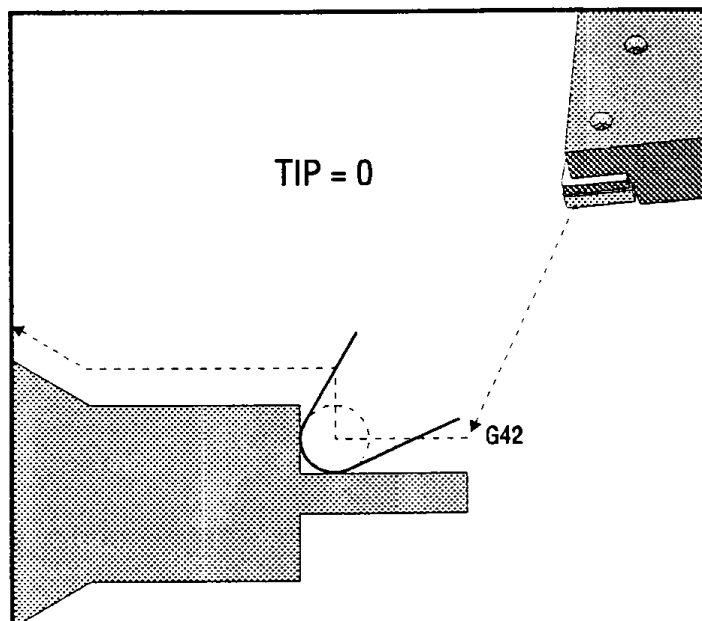
G41 will select tool nose compensation left; that is, the tool is moved to the left of the programmed path to compensate for the size of a tool. A tool offset must be selected with a Tnnxx code, where xx corresponds to the offsets to be used with the tool. If a negative radius is specified, tool nose compensation will operate as though a G42 was programmed.



G41 TNC Left

G42 TOOL NOSE COMPENSATION – RIGHT

G42 will select tool nose compensation right; that is, the tool is moved to the right of the programmed path to compensate for the size of a tool. A tool offset must be selected with a Tnnxx code, where xx corresponds to the offsets to be used with the tool.



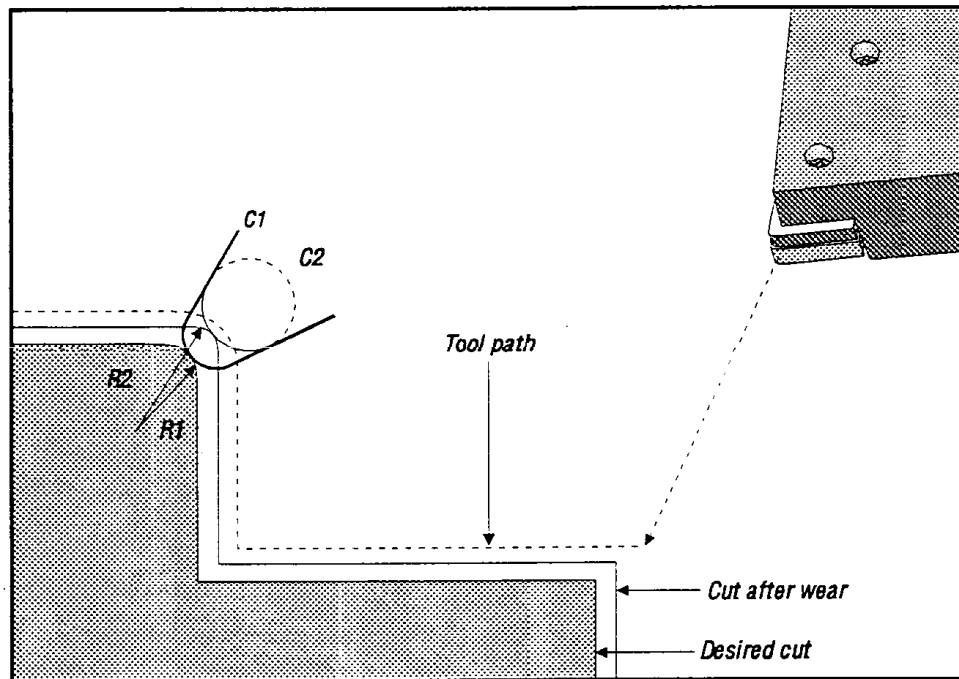
Tool Nose Compensation Programming

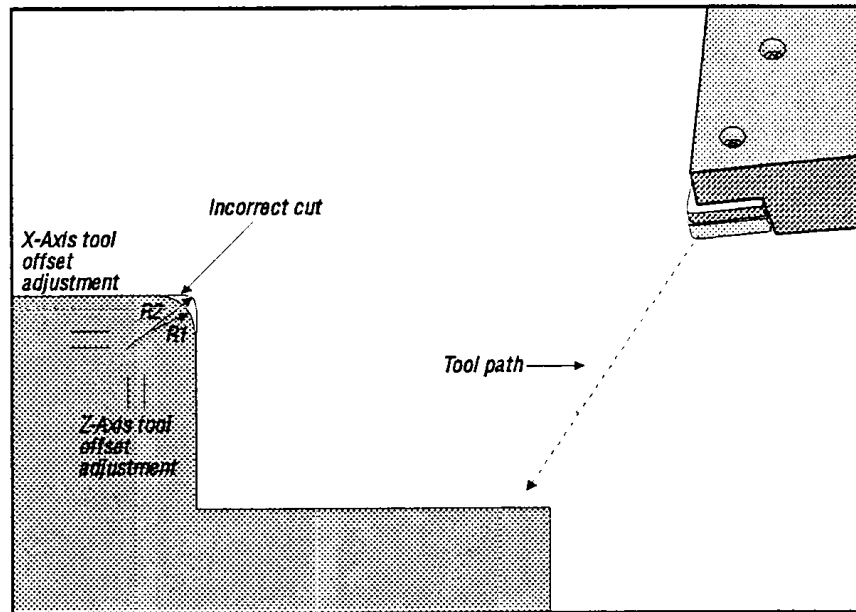
Overview

Tool nose compensation is a feature that allows the user to adjust a programmed tool path for normal cutter wear or in response to differing cutter tool nose radius sizes. The user can do this by entering minimal offset data at runtime without any additional programming effort.

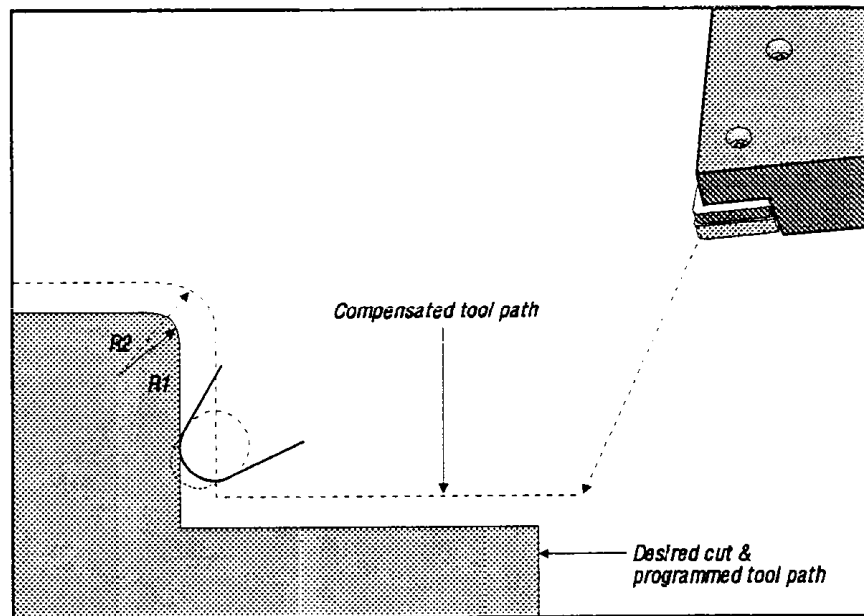
When to Use Tool Nose Compensation

Tool nose compensation is used when the tool nose radius changes and cutter wear is to be accounted for with curved surfaces or tapered cuts. Tool nose compensation generally does not need to be used when programmed cuts are solely along the X-axis (diameters) or Z-axis (faces). For angled and circular cuts, as the tool nose radius changes, undercutting or overcutting can occur. In the figure below, suppose that immediately after setup, C1 is the radius of the cutter that cuts the programmed tool path. As the cutter wears to C2, the operator might adjust the tool geometry offset to bring the part length and diameter to dimension. If this were done, as shown in Figure 8, a smaller radius would occur. If tool nose compensation is used, an incorrect cut does not occur. The control will automatically adjust the programmed path based on the offset for tool nose radius as set up in the control. The control will alter or generate code to cut the proper part geometry.





Two cuts overlaid to show cutting error

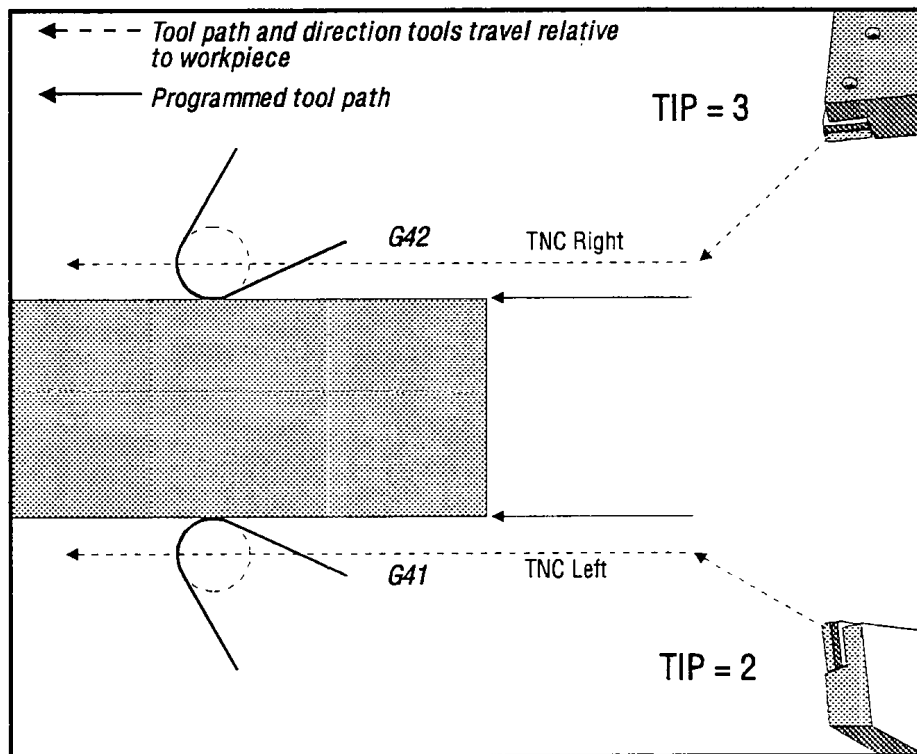


Path generated when tool nose compensation is used.

Note that in the above figure, the programmed path coincides with the final part dimension. Although parts do not have to be programmed using tool nose compensation, it is the preferred method because it makes program problems easier to detect and resolve.

Tool Nose Compensation Concepts

Tool nose compensation works by shifting the PROGRAMMED TOOL PATH to the right or to the left. The programmer will usually program the tool path to the current manufacturing operation finish size. When tool nose compensation is used, the control will compensate for the diameter of a tool based on special instructions written into the program. Two G-code commands are used to do this for compensation within a two-dimensional plane. G41 commands the control to shift to the left of the programmed tool path, and G42 commands the control to shift to the right of the programmed tool path. Another command, G40, is provided to cancel any shift made by tool nose compensation. G40, G41, and G42 are described in detail later in the "Using Tool Nose Compensation G Codes" section.



Shift direction

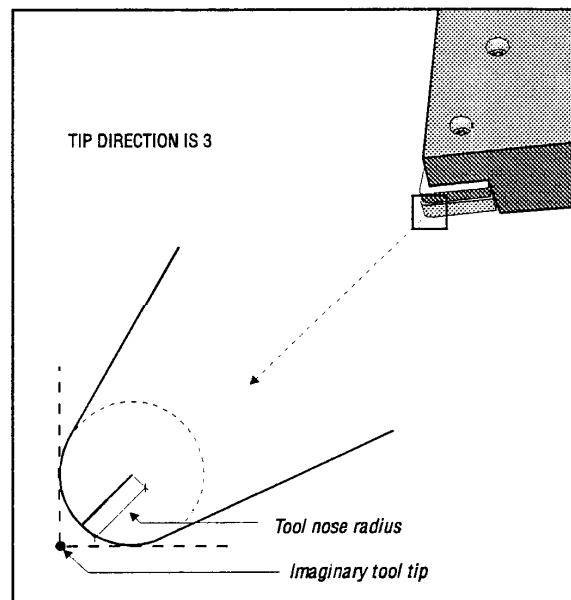
The shift direction is based on the direction of the tool movement relative to the tool and the side of the part it is located. When thinking about which direction the compensated shift will occur in tool nose compensation, imagine yourself standing on the line of the part to be cut, looking in the direction the cutter is traveling. Which side of the line is the cutter going to pass you on, left or right? G41 will shift the tool tip on the left and G42 will shift the tool tip on the right. For a lathe, this means that normal O.D. turning, which is machining from the face of the part toward the chuck will require a G42 for the correct tool compensation, while normal I.D. turning will require a G41.

Imaginary Tool Tip and Direction

Tool nose compensation assumes that a compensated tool has a radius at the tool tip that it must compensate for. This is known as the TOOL NOSE RADIUS (TNR). For a lathe, it is not easy to determine the center of a tool nose radius. The cutting edges are set when a tool is touched off to record tool geometry. The control can calculate where the center of the tool radius is by using the tool geometry offset information, the radius of the tool, the shift direction the cutter is expected to cut, and the direction the cutter is pointing. The X and Z-axis' geometry offsets intersect at a point, called the IMAGINARY TOOL TIP, which determines the tool tip direction. The TOOL TIP DIRECTION is determined by a vector originating from the center of the tool radius and extending to the imaginary tool tip.

The tool tip direction of each tool is coded as a single integer from 0 to 9. The tip direction code is found next to the radius offset on the geometry offsets page. It is recommended that a tip direction be specified for all tools using tool nose compensation. Figure 14 is a summary of the tip-coding scheme along with cutter orientation examples.

Note that the tip indicates to the setup person how the programmer intends the tool offset geometry to be measured. For instance, if the setup sheet shows tip direction 8, the programmer intends the tool geometry to be at the edge of and on the centerline of the tool insert.



Imaginary Tool Tip

Using Tool Nose Compensation

Tool nose compensation accomplishes its task by reading ahead one or two blocks to determine how to modify the current block of code. This is referred to as BLOCK LOOKAHEAD or LOOKAHEAD PROCESSING.

When the control is first powered on or in the reset condition, tool nose compensation is not active. Tool nose compensation is turned on in a program by the G41 or G42 command. When this command is executed, the control will look ahead to determine where the first compensated move will be. The first compensated move is generally a move from a non-compensated position to a compensated position and is therefore unusual. This first move is called the APPROACH move and is required when using tool nose compensation. Similarly, a DEPART move is also required. In a DEPART move, the control will move from a compensated position to a non-compensated position. A depart move occurs when tool nose compensation is cancelled with a G40 command or Txx00 command. Although approach and depart moves can be precisely planned, they are generally uncontrolled moves and the tool should not be in contact with the part when they occur.

Follow the steps below when using tool nose compensation.

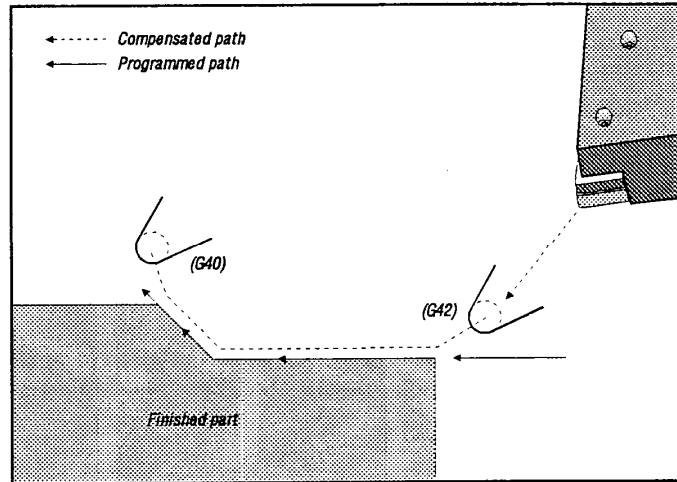
PROGRAM the part to finished dimensions.

- 1. Approach and Departure Moves:** Ensure that there is an approach move for each compensated path that is executed and determine if G41 or G42 is to be used. Ensure that there is also a departure move for each compensated path. Approach and departure moves should be equal to or greater than the tool nose radius of the tool.
- 2. Tool Geometry Offsets:** Set the tool length geometry offsets and clear the length wear offsets of each tool.
- 3. Tool Nose Radius and Wear Offsets:** Select a standard insert (with a defined radius) that will be used for each tool. Set the tool nose radius offset for each compensated tool. Clear the corresponding tool nose wear offset to start at zero for each tool.
- 4. Tool Tip Direction:** Input the tool tip direction for each tool that is using compensation, G41 or G42.
- 5. Check Compensation Geometry:** Debug the program in graphics mode and correct any tool nose compensation geometry problems that may occur. A problem can be detected in two ways: an alarm will be generated indicating compensation interference or the incorrect geometry will be seen generated in graphics mode.
- 6. Run and Inspect First Article:** Adjust wear offsets for the setup part.

Each of the above steps is described in detail in the following sections.

1. Approach And Departure Moves

The first X or Z motion in the same line that contains a G41 or G42 is called the APPROACH move. The first move must be a **linear move**, a **G01** or **G00**. At the start of an approach move, the current position is not compensated. At the end of the approach move, the machine position will be fully compensated. This is shown in the following figure.



Approach and Departure moves

Any line containing a G40 will cancel tool nose compensation. This is called the DEPARTURE move. The last move must be a linear move, a G01 or G00. At the start of a departure move, the current position is fully compensated. At the end of the departure move, the machine position is not compensated.

NOTE: Figure 13 (below) shows the normal condition just prior to cancellation of tool nose compensation. Some geometries will result in overcutting or undercutting of the part. The programmer can control this by including an I and K in the G40 cancellation block. The I and K address codes in a G40 block define a vector the control will use in determining the compensated target position of the previous block. The vector is usually aligned with an edge or wall of the completed part. Figure 13 shows how I and K can correct undesired cutting in a departure move. Refer to the G40 command description for instructions on calculating values of I and K.

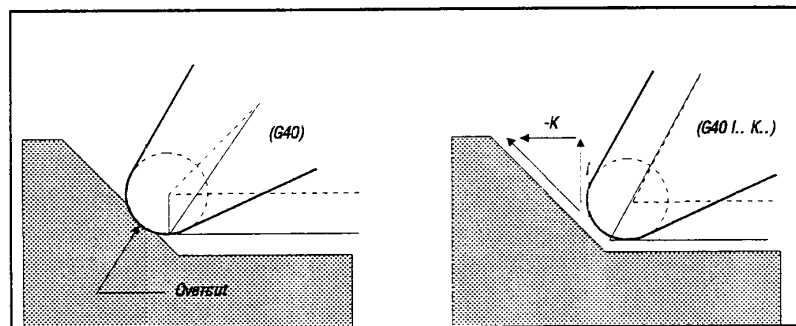


Figure 13: Use of I and K in a G40 block

2. Tool Geometry Offsets

The length geometries offsets of tools that use tool nose compensation are set up in the same manner as tools not using compensation. Refer to the "Setup Procedures" section of your operation manual for details on touching off tools and recording tool-length geometries. When a new tool is set up, the geometry WEAR should be cleared to zero.

Often a tool will exhibit uneven wear. This occurs when particularly heavy cuts occur on one edge of the tool. In this case, it may be desirable to adjust the X or Z GEOMETRY WEAR OFFSETS rather than the RADIUS WEAR OFFSETS. By adjusting X or Z length geometry wear, the operator can often compensate for uneven tool nose wear. Length geometry wear will shift ALL dimensions for a single axis. The part design "may not allow" the operator to compensate for wear by using the length geometry shift. One can determine which wear to adjust by checking several X and Z dimensions on a finished part. Wear that is even will result in similar dimensional changes on the X and Z-axes and suggests that the radius wear offset should be increased. Wear that affects the dimensions on one axis only suggests length geometry wear.

3. Tool Nose Radius And Wear Offsets

Each turning tool that uses tool nose compensation requires a TOOL NOSE RADIUS. The tool nose radius specifies how much the control is to compensate for a given tool. It is determined by the geometry of the tool tip. If standard inserts are being used for the tool, then the tool nose radius is simply the tool tip radius of the insert.

Associated with each tool on the geometry offsets page is a TOOL NOSE RADIUS OFFSET. The column, labeled RADIUS, is where the tool nose radius of each tool is placed. If the value of any tool nose radius offset is set to zero, no compensation will be generated for that tool.

Associated with each radius offset is a RADIUS WEAR OFFSET. It is located on the wear offset page. The control adds the wear offset to the radius offset to obtain an effective radius that will be used for generating compensated values.

Small adjustments to the radius offset during production runs should be placed in the wear offset page. This allows the operator to easily track the wear for a given tool. As a tool is used, the insert will generally wear so that there is a larger radius at the end of the tool. This should place positive values in the wear column. When replacing a worn tool with a new one, the wear offset should be cleared to zero.

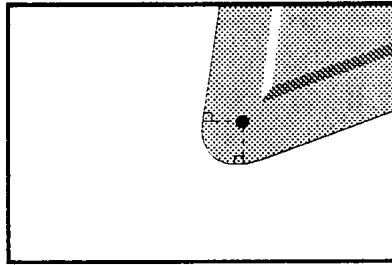
4. Tool Tip Direction

The tool tip direction of each tool is coded as a single integer from 0 to 9. The tip direction code is found next to the radius offset on the geometry offsets page. It is recommended that a tip direction be specified for all tools using tool nose compensation. The following figure is a summary of the tip coding scheme along with cutter orientation examples.

Note that the tip indicates to the setup person how the programmer intends the tool offset geometry to be measured. For instance, if the setup sheet shows tip direction 8, the programmer intends the tool geometry to be at the edge of and on the centerline of the tool insert.

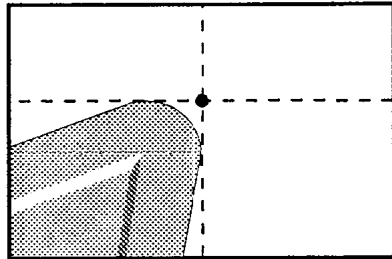
TOOL TIP CODE #	TOOL TIP DIRECTION ORIENTATION	COMMENT TOOL CENTER LOCATION
--------------------	-----------------------------------	---------------------------------

0



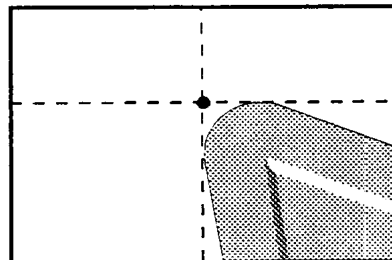
Zero (0) indicates no Specified direction. It is usually not used when tool nose compensation is desired.

1



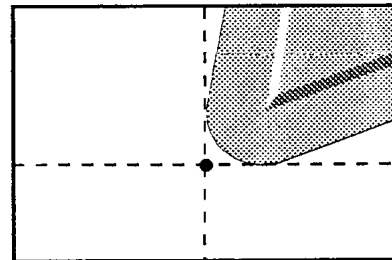
Tool tip off in the X+, Z+ direction.

2



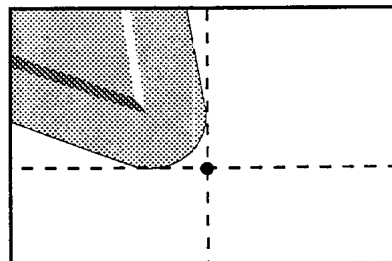
Tool tip off in the X+, Z- direction.

3



Tool tip off in the X-, Z- direction.

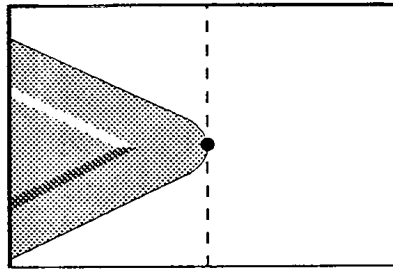
4



Tool tip off in the X-, Z+ direction.

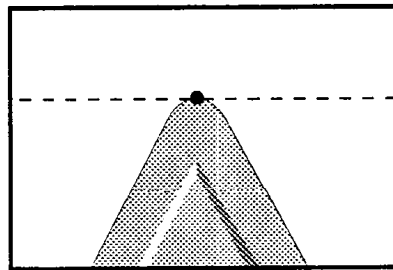
TOOL TIP CODE #	TOOL TIP DIRECTION ORIENTATION	COMMENT TOOL CENTER LOCATION
--------------------	-----------------------------------	---------------------------------

5



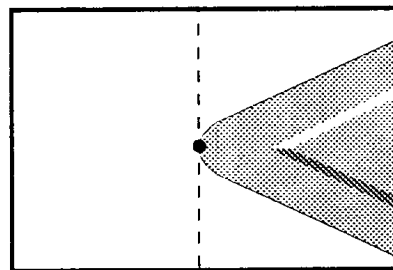
Tool Edge
Z+ direction

6



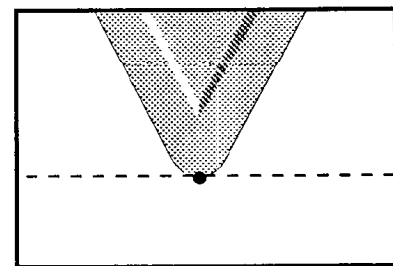
Tool Edge
X+ direction

7



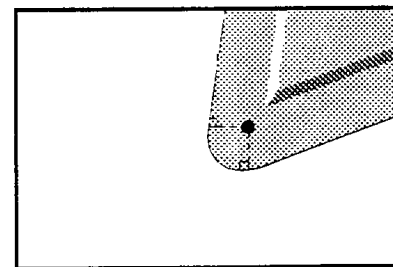
Tool Edge
Z- direction

8



Tool Edge
X- direction

9



Same as tip 0

NOTE: It is important to remember that tool nose compensation values are in terms of radius rather than diameter. This is important in blocks where tool nose compensation is cancelled or turned on. **If the incremental distance of a departure or approach move in a compensated path is not twice the radius of the cutting tool's radius, overcutting may occur.** Always remember that programmed paths are in terms of diameter and allow for twice the tool radius on approach and departure moves. The block of canned cycles requiring a PQ sequence can often be a departure move. The following example illustrates how incorrect programming will result in overcutting.

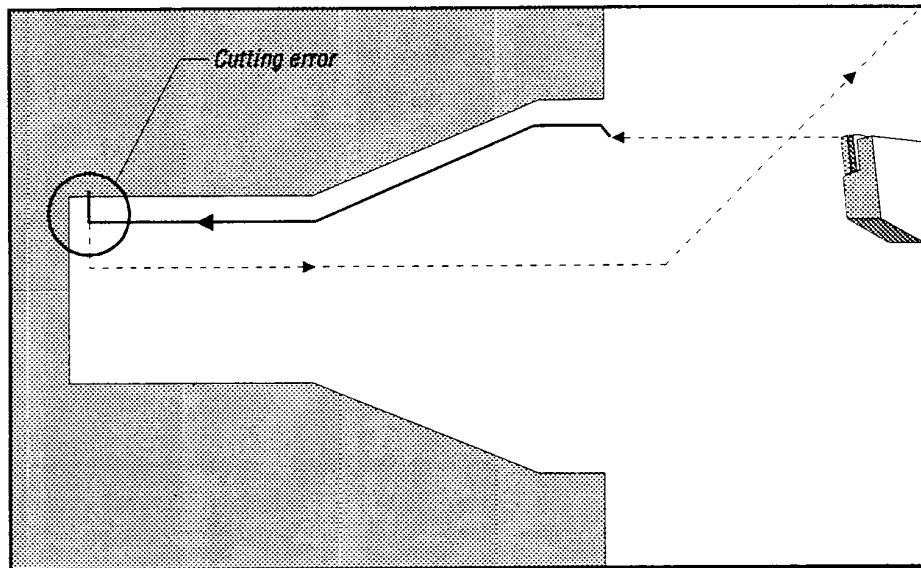
Example

Setting 33 is FANUC:	X	Z	Radius Tip
Tool Geometry Offset 8:	-6.0000 -8.000	0.0156	2

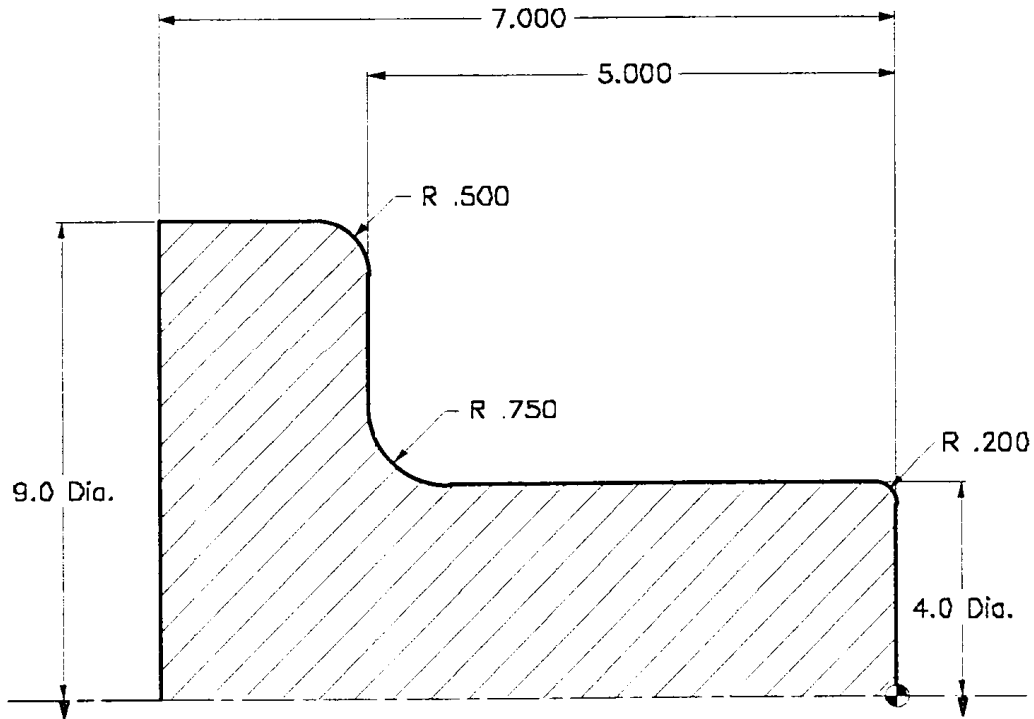
```
O0104;
G28;
T808; (Boring Bar);
G97 2400 M03;
G54 G00 X.49 Z.05;
G51 G01 X.5156 F.004;
Z-.05;
X.3438 Z-.25;
Z-.5;
X.33;
```

Note: Move is less than .032, which is the value required to avoid cut-in with a departure move before TNC is cancelled.

```
G40 G00 X.25;
Z.05;
G28;
M30;
```



Circular Interpolation Exercise with Tool Compensation



Fill in the blanks to complete one finish pass of the cutter path for the above part. Assume the part has already been roughed out. Define the tool path of the part using cutter compensation (G41 or G42) where the tool nose radius is .031 radius for an O.D. turning tool.

O0057

(Circular Interpolation With Cutter Compensation)

N1 (Finish O.D.)

G28

T101 (O.D. Turning Tool .031 TNR)

G50 S2200

G97 S____ M03

G54 G00 X4.2 Z0.1 M08

G96 S750

G00 Z0.

G01 X-0.062 F0.004

G0 X3.5 Z0.10

G01 G____ Z0. F0.005 (**TURN ON COMP**)

X_____

G03 X_____ Z-_____ R_____

G01 Z_____

G02 X_____ Z-_____ R_____

G01 X_____

G03 X_____ Z-_____ R_____

G01 Z-_____

G00 G40 X_____ Z1. M09 (**COMP OFF**)

G97 S320

G28

M30

Canned Cycles and additional G Codes

One of the most common types of codes in CNC programming are “canned cycles.” This unit will define and discuss canned cycles and state some uses and applications.

Objectives:

Upon completion of this unit, the student will:

- 1) Be able to define “canned cycle.”
- 2) Be familiar with common canned cycles, their variables, the proper applications, and the correct use of them.
- 3) Complete final exercise.

Machining Cycles for the Lathe

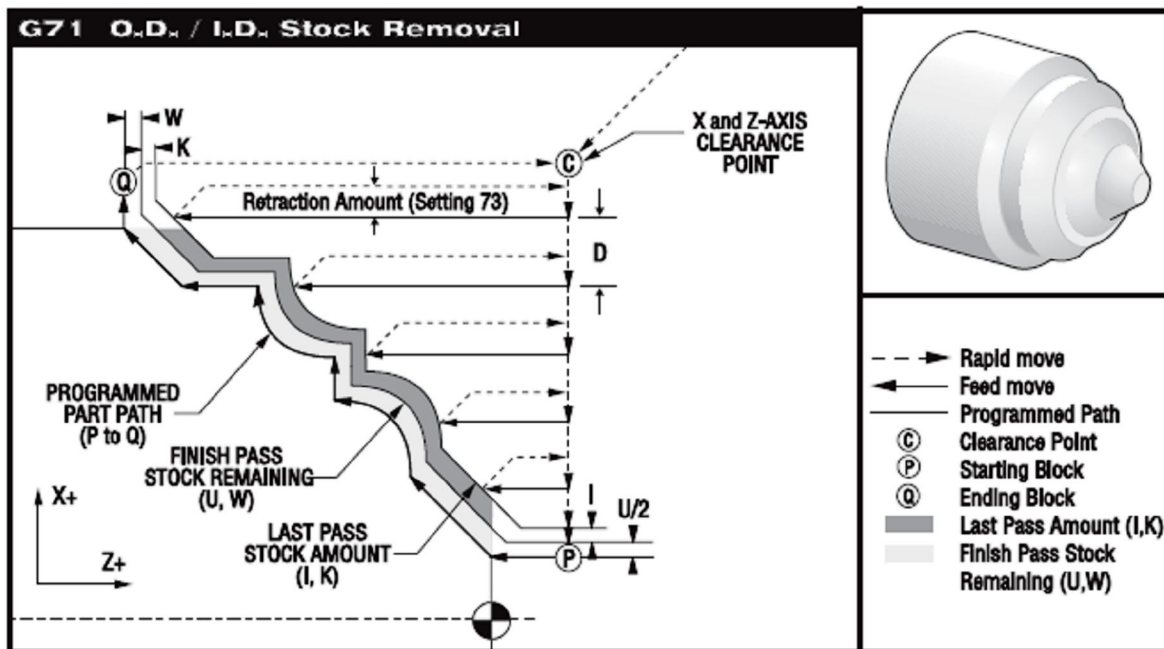
A machining cycle is used to simplify the programming of a part. Machine cycles are defined for the most common machining operations. They can be divided into two types. There are machine cycles for turning and grooving. There are cycles for drilling and tapping and can be either single block canned cycles or modal canned cycles.

The following is a list of the canned cycles that can be used for turning and grooving for the HAAS lathe controls:

G70	Finishing Cycles
G71	O.D./I.D. Stock Removal Cycle
G72	End Face Stock Removal Cycle
G73	Irregular Path Stock Removal Cycle
G74	End Face Grooving or Turning with Chip Break Cycle
G75	O.D./I.D. Grooving or Turning with Chip Break Cycle
G76	Thread Cutting Cycle, Multiple Pass
G90	O.D./I.D. Turning Cycle Modal
G92	Thread Cutting Cycle Modal
G94	End Face Cutting Cycle Modal

G71 O.D./I.D. STOCK REMOVAL CYCLE

- P Starting Block number of path to rough
- Q Ending Block number of path to rough
- U X-axis value and direction of G71 rough stock allowance, diameter
- W Z-axis value and direction of G71 rough stock allowance
- D Depth of cut for each pass of stock removal, positive radius
- *I X-axis stock value and direction for a G71 finish pass, radius
- *K Z-axis stock value and direction for a G71 finish pass
- *S Spindle speed to use throughout G71 PQ block
- *T Tool and offset to use throughout G71 PQ block
- *F Feedrate to use throughout G71 PQ block
- *R1 YASNAC select Type II roughing
- * Indicate optional



This canned cycle will rough out material on a part given the finished part shape. All a programmer needs to do is to define the shape of a part by **programming the finished tool path** and then submitting the path definition to the G71 call by means of a **PQ block designation**. Any feeds, spindle speeds or tools within the block defining the path are ignored by the G71 call. Any F, S or T commands on the G71 line or in effect at the time of the G71, are used throughout the G71 roughing cycle. Usually, a G70 call to the same PQ block definition is used to finish the shape using the programmed feeds, speeds, tools and offsets defined within the PQ block definition.

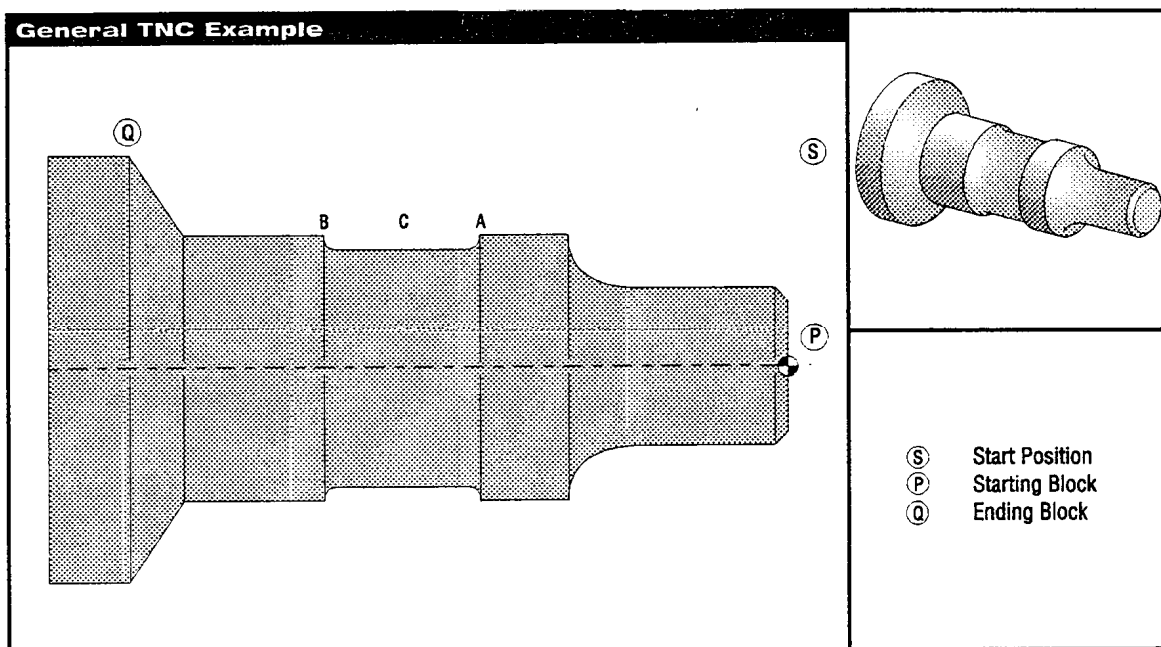
Two types of machining paths are addressed with a G71 command. The first type of path, a **Type I**, is when the X-axis of the programmed path does not change direction. This type of path is called a monotonic path. The second type of path, a **Type II**, allows the X-axis to change direction. For both the first and second type of programmed path, the Z-axis must be monotonous; it cannot change direction.

For FANUC, **Type I** is selected by having only an **X-axis motion** in the block specified by the **P block** in the G71 call. For FANUC, **Type II** is selected by having both an **X-axis** and **Z-axis** motion in the **P block**. When in YASNAC mode, Type II roughing is selected by including R1 on the G71 command block.

G71 consists of a roughing phase and a finishing phase. The roughing and finishing phase are handled slightly differently for types I and types II. Generally, the roughing phase consists of repeated passes along the Z-axis at the specified G71 command line feed rate. The finishing phase consists of a G70 block using the programmed feeds, speeds, tools and offsets defined within the PQ block definition. **The final motion** in either types is a **return to the starting position S**.

The **start position S** is the position of the tool at the time of the G71 call. The Z clearance plane is derived from the Z-axis start position and the sum of W and optional K finish.

Program Example using G70 and G71



Tool Description:

- Tool 1 80 Degree CNMG 432 Roughing insert .0312 Radius
- Tool 2 55 Degree TPG 432 Finishing insert .0312 Radius
- Tool 3 .250 Wide Grooving Tool with .0156R

Tools	OffsetsX	Z	Radius Tip	
T1	01	-8.9650	-12.8470	.0312 3
T2	02	-8.9010	-12.8450	.0312 3
T3	03	-8.8400	-12.8380	.016 3
T3	13	-8.8400	-12.5880	.016 4

Program Example

```
%
O0105 (General TNC Example)
N1
G28 (Return to machine zero for tool change)
T0101 (Select Tool 1 Offset 1)
G50 S1000
G97 S500 M03
G54 G00 X2.1 Z0.1 (Rapid to Point S, Defines Stock)
G96 S300
G71 P10 Q20 U0.02 W0.01 D.1 F0.01 (Rough P to Q using G71 and TNC)
(Define part path PQ sequence)
N10 G42 G00 X0 Z0.1 F.01 (P Block) (G71 type II, TNC approach)
G01 Z0 F.005
X0.65
X0.75 Z-0.05
Z-0.75
G02 X1.25 Z-1. R0.25
G01 Z-1.5
G02 X1. Z-1.625 R0.125
G01 Z-2.5
G02 X1.25 Z-2.625 R0.125
G01 Z-3.5
X2. Z-3.75
N20 G00 G40 X2.1 (TNC departure) (Q block)
G97 S500
G28 (Zero for tool change clearance)
M01

N2
G28 (Return to machine zero for tool change)
T202 (Select Tool 2 Offset 2)
G50 S1000
G97 S750 M03
G00 X2.1 Z0.1 (Rapid to point S)
G96 S400
G70 P10 Q20 (Finish P to Q using G70 and TNC)
G97 S750
G28 (Return to machine zero for tool change)
M01
```

N3	
G28	(Return to machine zero for tool change)
T303	(Select Tool 3 Offset 3)
G50 S1000	(Groove to Point B using Offset 3)
G97 S500 M03	
G00 G42 X1.5 Z-2.0	(Rapid to Point C)(TNC approach)
G96 S200	
G01 X1. F0.003	
G01 Z-2.5	
G02 X1.25 Z-2.625 R0.125	
G40 G01 X1.5	(TNC Depart)
	(Move to right side of groove, to Point A, using Offset 13)
T313	(Change offset to other side of tool)
G00 G41 X1.5 Z-2.125	(Move to Point C)(TNC approach)
G01 X1. F0.003	
G01 Z-1.625	
G03 X1.25 Z-1.5 R0.125	(A)
G40 G01 X1.6	(TNC Depart)
G97 S500	
G28	(Zero for tool change clearance)
M30	
%	

Note: Two offsets must be used when finishing the groove using tool nose compensation. Here, the same tool is used, but the tool cuts on different sides along the Z-axis. Offset 3 Z length geometry is set on the left side of the tool whereas offset 13 Z length geometry is set on the right side of the tool.

Tool Nose Compensation in Canned Cycles

This section describes how tool nose compensation works when a canned cycle is used. Refer to the “Canned Cycles” section of this manual for a detailed description of canned cycles. Some canned cycles ignore tool nose compensation; some canned cycles expect a specific coding structure, while other canned cycles perform their own specific canned cycle activity.

The following canned cycles will ignore tool nose radius compensation. It is recommended that tool nose compensation be cancelled prior to executing any of these canned cycles.

G74 End face grooving cycle, peck drilling
G75 O.D./I.D. grooving cycle, peck drilling
G76 Thread cutting cycle, multiple pass
G92 Thread cutting cycle, modal

The following canned cycles work well when a specific programming sequence is used. This programming sequence is called a TEMPLATE. By using the suggested template, the programmer should have no problem using these canned cycles with tool nose compensation. These canned cycles make use of P and Q to identify a path the canned cycle is to work with.

G70 FINISHING CYCLE

Usually, G70 is used following the use of a G71, G72 or G73, but it can be used alone. Below is the template for using tool nose compensation with G70 alone. Note that TNC approach is part of the PQ path definition sequence, whereas TNC departure is after the execution of G70.

G71 and G72 are similar canned cycles with regard to tool nose compensation. The finishing and rough finishing passes of G71 and G72 recognize tool nose compensation; however, the roughing pass of these two G codes does not. The template below can be applied to either G71 or G72.

G73 is similar to G71 and G72. G73 recognizes TNC on all passes.

P Starting Block number of routine to execute
Q Ending Block number of routine to execute

The G70 Finishing cycle can be used to finish cut paths that are defined and roughed out with stock removal cycles G71, G72 and G73.

The G70 requires a beginning block number (P code) and an ending block number (Q code) be specified for the machine code defining the part geometry to be machined.

The G70 cycle is usually used after a G71, G72 or G73 has been performed using the blocks specified by P and Q. All codes in the block defined by P and Q are executed. Any F, S or T codes between the P and Q block are effective. The PQ sequence is searched for in the current program starting from the beginning of the program. The current machine position is saved and remembered as the start position. Then the block starting at P is executed. Processing continues in a normal fashion with blocks following P until a block containing an N code matching the Q code in the G70 calling block is found and executed.

After execution of the Q block, a rapid (G00) is executed returning the machine to the start position saved earlier during G70 initialization. The program then returns to the block following the G70 call.

EXAMPLE G70 WITH TNC

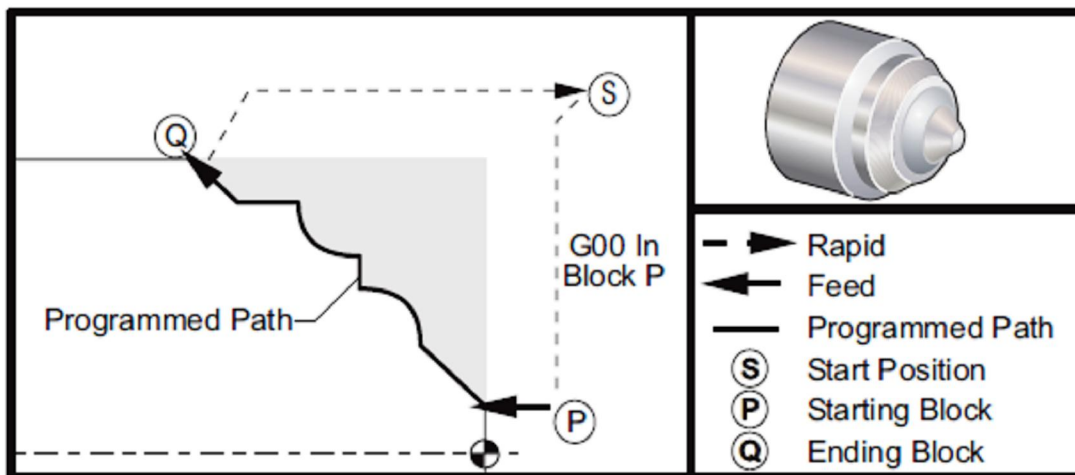
A traditional calling sequence using G70 appears below.

Rough cycle feedrate to rough out part between N10 and N50 for the G71 roughing cycle will be F.012, which is defined in the G71 block.

Finish Feedrate for the G70 finishing cycle will be whatever is defined between N10 and N50.

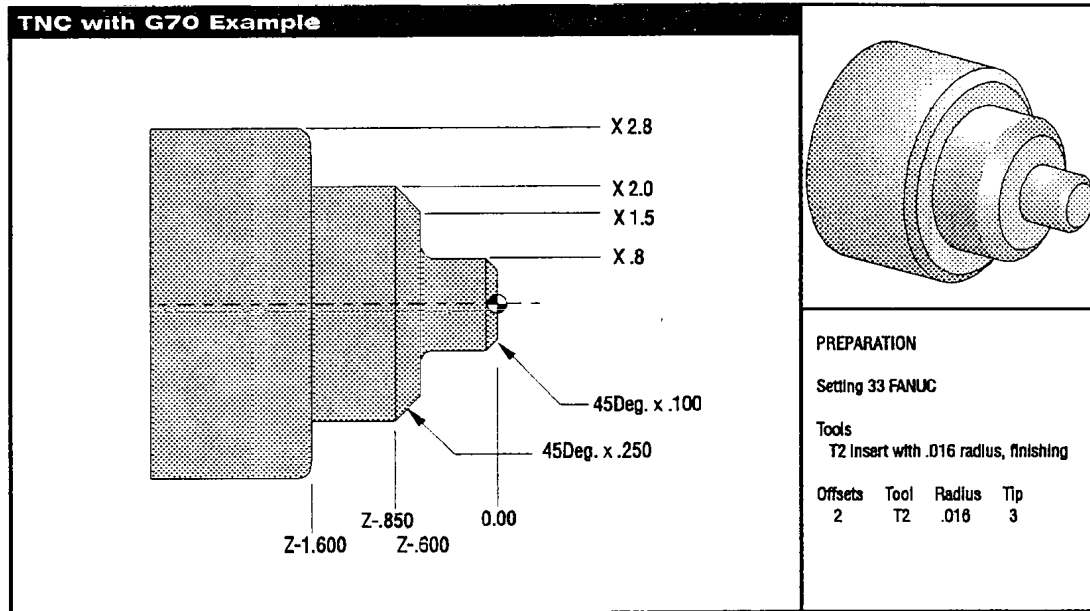
```
G71 P10 Q70 U.01 W.005 D.08 F.012      (Roughing cycle feedrate will be F.012 which is defined
                                         this block)
N10 G00                                (machine code that defines part path geometry)
.... " " " " F.005
.... " " " "
.... " " " "
.... " " " " F.003
.... " " " " F.008
N70
G70 P10 Q70                            (Finish cycle feedrate is defined between N10 to N50).
```

This example demonstrates the use of a G70 finishing canned cycle. The basic shape should be roughed out using a G71, G72, or G73 roughing cycle.



Part program using a G70 with TNC

This example demonstrates tool nose compensation with the G70 finishing cycle. The basic shape has been roughed out using a G71 roughing cycle.



Part drawing for TNC with G70 example Type 1

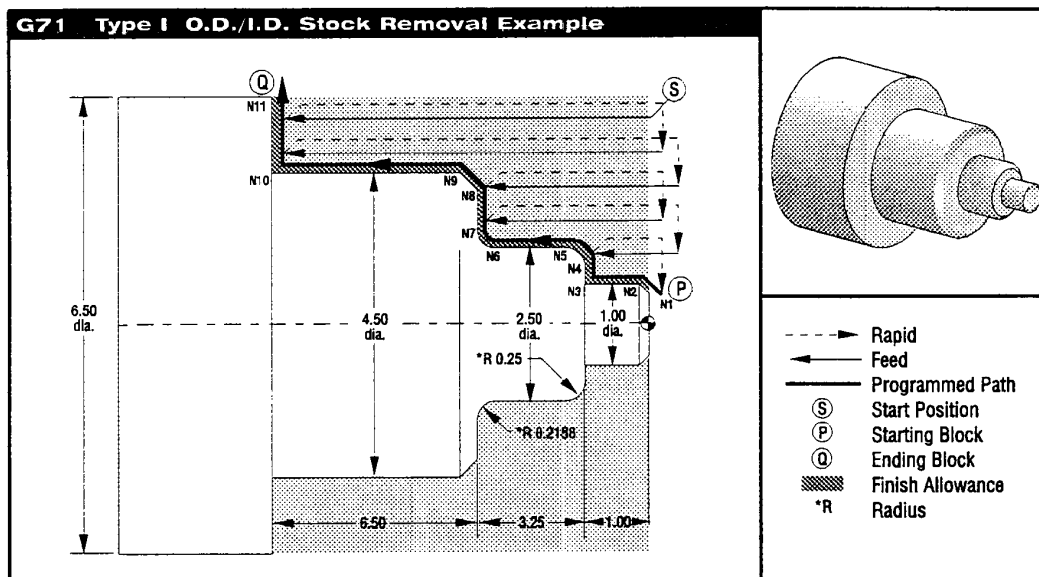
Program Example

Description

```

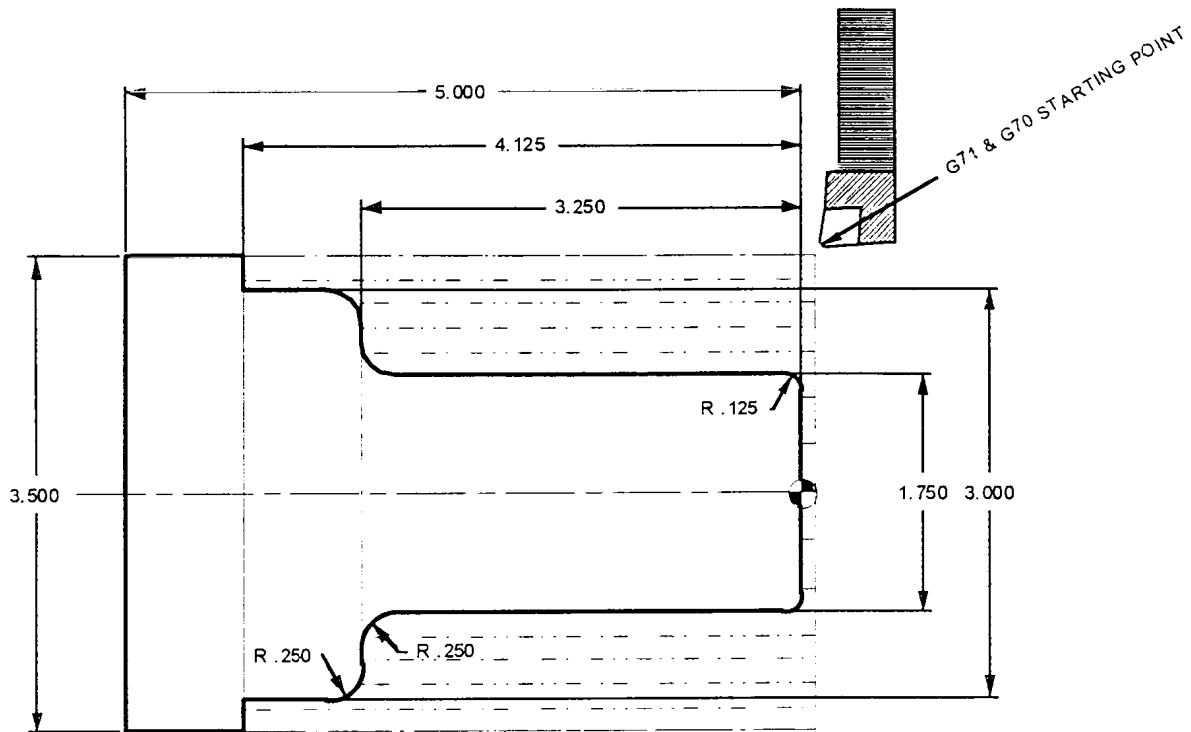
%
O0106
G28
T0202 (Select Tool 2)
G50 S1750
G97 S320 M03
G54 G00 X3. Z0.1 M08
G96 S300
G71 P10 Q20 U.02 W.01 D.1 F.012 (G71 Define part path lines P thru Q)
N10 G42 G00 X0.5 (P Block)(Approach move, turn on tool nose compensation)
G01 Z0. F.012
X.6
X0.8 Z-0.1 F.008
Z-0.5
G02 X1.0 Z-0.6 I0.1
G01 X1.5
X2.0 Z-0.85
Z-1.6
X2.3
G03 X2.8 Z-1.85 K-0.25
G01 Z-2.1
N20 G00 G40 X3.0 (Q)(Departure move turning off tool nose compensation)
G70 P10 Q20 (G70 Finishing Cycle)
G28 (Return to machine zero for tool change clearance)
M30
%
```


Note that the suggested template of the previous section for G70 is used. Also note that compensation is enabled in the PQ sequence, but is cancelled after G70 is completed.



O0107 (O.D. Roughing and Finishing)	(FANUC G71 – TYPE I example)
G28	(Rapid to Home Position)
T101 (CNMG 432)	(Tool change & apply offsets)
G50 S2000	(Set Max RPM 2000)
G97 S400 M03	(Spindle On)
G54 G00 X6.6 Z0.1 M08	(Rapid to Start Position, Coolant On)
G96 S630	(Constant surface speed on)
G71 P10 Q20 U0.01 W0.005 D0.15 F0.012	(G71 Roughing Cycle)
N10 G00 X0.6634	(P Block)(Begin definition)
G01 X1. Z-0.1683 F0.004	
Z-1.	
X1.9376	
G03 X2.5 Z-1.2812 R0.2812	
G01 Z-40312	
G02 X2.9376 Z-4.25 R0.2188	
G01 X3.9634	
X4.5 Z-4.5183	
Z-10.75	
N20 X6.5	(Q Block)(End definition)
G97 S400 M09	
G28	(Rapid to tool change position)
T202	(Finish tool)
G50 S2500	
G97 S400 M03	
G54 G00 X6. Z0.1 M08	
G96 S730	
G70 P10 Q20	(G70 Finishing Cycle)
G97 S400 M09	
G28	
M30	
%	

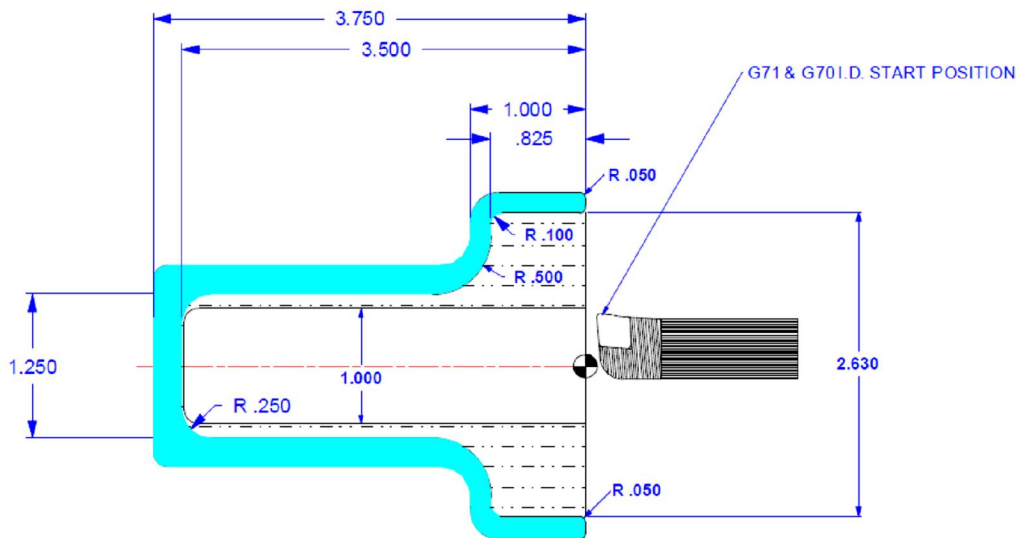
G71 and G70 Exercise with Tool Nose Compensation



Fill in the blank line for tool #1 using a G71 roughing cycle command to define roughing passes for the part geometry defined between N10 and N20. Then, define a finish pass using a G70 finishing cycle command, with tool #2, to define a finish pass for the part geometry defined between N10 and N20. On the G71 command line, leave .010 stock on diameters and .005 on the faces. Take .120 depth of cut at .012 feed.

O0030	(Program number)
N1	(First Operation)
T101 (O.D. Tool X.031 TNR)	(Tool #1 and Offset #1)
G50 S2500	(Spindle speed clamp at 2500 RPM)
G97 S____ M03	(Cancel CCS, define a 400 RPM, spindle ON forward)
G54 G____ X____ Z____ M08	(Rapid X and Z to starting location, coolant ON)
G96 S____	(Turn on CSS to 375)
G71 P____ Q____ U____ W____ D____ F____	(Roughing O.D. G71 cycle command)
N____ G____ G____ X____ Z____	(Pnn starting number, Rapid X and Z axis, cutter comp ON)
G____ Z____ F.006	(G71 Part Geometry)
X____	(G71 Part Geometry)
G____ X____ Z____ R____	(G71 Part Geometry)
G____ Z____	(G71 Part Geometry)
G____ X____ Z____ R____	(G71 Part Geometry)
G____ X____	(G71 Part Geometry)
G____ X____ Z____ R____	(G71 Part Geometry)
G____ Z____ F.003	(G71 Part Geometry)
X____ F.015	(G71 Part Geometry)
N____ G____ X____	(Qnnnn ending number, Cancel Cutter Comp.)
G____ S____ M____	(Cancel CSS, define 400 RPM, spindle ON forward, Coolant Off)
G____	(Return to reference point)
M____	(Optional stop command)
N2	(Second operation)
T____	(Tool #2 and Offset #2)
G____ S____	(Spindle speed clamp at 2500 RPM)
G____ S____ M____	(Cancel CSS, turn on spindle 1200 RPM)
G____ X____ Z____	(Rapid to start position above part)
G____ S____	(Turn on CSS to 650)
G____ P____ Q____	(Define finish pass using part geometry)
G____ S____ M____	(Cancel CSS, define 400 RPM, spindle on , Coolant Off)
G____	(Return to reference point)
M____	(End of program rewind)

G71/G70 TYPE I ROUGHING AND FINISHING AN I.D. WITH TNC EXAMPLE



NOTE: Be sure the tools start position is positioned below the diameter of the part you wish to start roughing out, before defining a G71 on an I.D. with this cycle.

TOOL	OFFSET	RADIUS	TIP
4	04	.032	2

```

O00088 (ID G71 EXTRA)
N1 G53 G00 X0 Z0 T0
N2 T404
N3 G50 S3000
N4 G97 S___0 M03
N5 G54 G00 X___ Z0.1 M08 (ST PT BELOW STK OF ID)
N6 G96 S420
N7 G71 P8 Q18 U-___ W___ D0.08 F0.010 (G71 is MINUS for G71ID)
N8 G41 G00 X___ (N8 start position of part path geometry)
N9 G01 Z0 F0.008
N10 X___ F0.005
N11 G02 X2.63 Z-___ R0.05
N12 G01 Z-___
N13 G0__ X___ Z-0.825 R0.1
N14 G0__ X2.25
N15 G0__ X1.25 Z-1.___ R0.5
N16 G01 Z-3.25
N17 G03 X0.___ Z-___ R0.25
N18 G01 G40 X___
N19 G70 P8 Q18
N20 G97 S1780 M09
N21 G53 G00 X0 Z0 T0
N22 M30
    
```

G71 and G70 may be used on ID work. With G71 and G70 ID work the U value must be negative reflecting the stock left on will be smaller than with OD work. The part needs a hole drilled before using G71. G71 also works good for boring jaws. The above exercise has a 1" hole drilled 3.5" deep.

Type 1 Details

When Type I is specified by the programmer, it is assumed that the tool path is monotonic in the X-axis. Prior to any roughing motion, the tool path designated by PQ is checked for monotonicity and G code compliance. An alarm is generated if a problem is found.

Roughing begins by advancing from the start position S and moving to the first roughing pass. All roughing passes start and end at the Z clearance plane. Each roughing pass X-axis location is determined by applying the value specified in D to the current X location. The direction that D is applied is determined by the signs of U and W. The nature of the movement along the Z clearance plane for each roughing pass is determined by the G code in block P. If block P contains a G00 code, then movement along the Z clearance plane is a rapid mode. If block P contains a G01, then movement will be at the G71 feed rate. Roughing continues until the X-axis position in block P is exceeded.

Each roughing pass is stopped before it intersects the programmed tool path allowing for both roughing and finishing allowances. The tool is then retracted from the material at a 45-degree angle by the distance specified in **setting 73**. The tool then moves in rapid mode to the Z-axis clearance plane.

When roughing is completed, the tool is moved along the tool path to clean up the rough cut. If I and K are specified, an additional rough finish cut parallel to the tool path is performed.

Type II Details

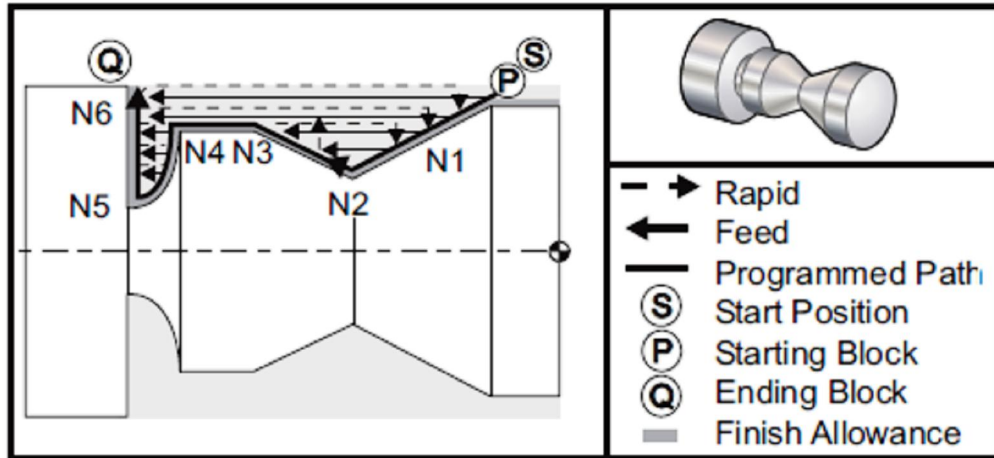
When Type II is specified by the programmer the X-axis PQ path is allowed to vary non-monotonically. In other words, the X-axis can change direction throughout the PQ path. Z must continue along in the same direction as the initial Z direction. The PQ path is checked prior to the start of any cutting and an alarm is generated if a problem exists.

When Setting 33 is set to FANUC, placing a reference to both the X and Z-axis in the block specified by P specifies Type II.

Roughing is similar to Type I except that after each pass along the Z-axis, the tool will follow the path defined by PQ. The tool will then retract parallel to the X-axis by a distance defined in Setting 73 (CAN CYCLE RETRACTION). The Type II roughing method does not leave steps in the part prior to finish cutting and typically results in a better finish.

There is virtually no limit to the number of blocks used to define a Type II PQ path. There is a limit to the number of troughs included in a PQ path definition. A trough can be defined as a change in direction creating a concave surface in the material being cut. If successive troughs are on the same level, there can be an unlimited number of troughs. When troughs are within troughs (nested), there can be no more than 10 levels of trough nesting. An alarm is generated when this limit is exceeded.

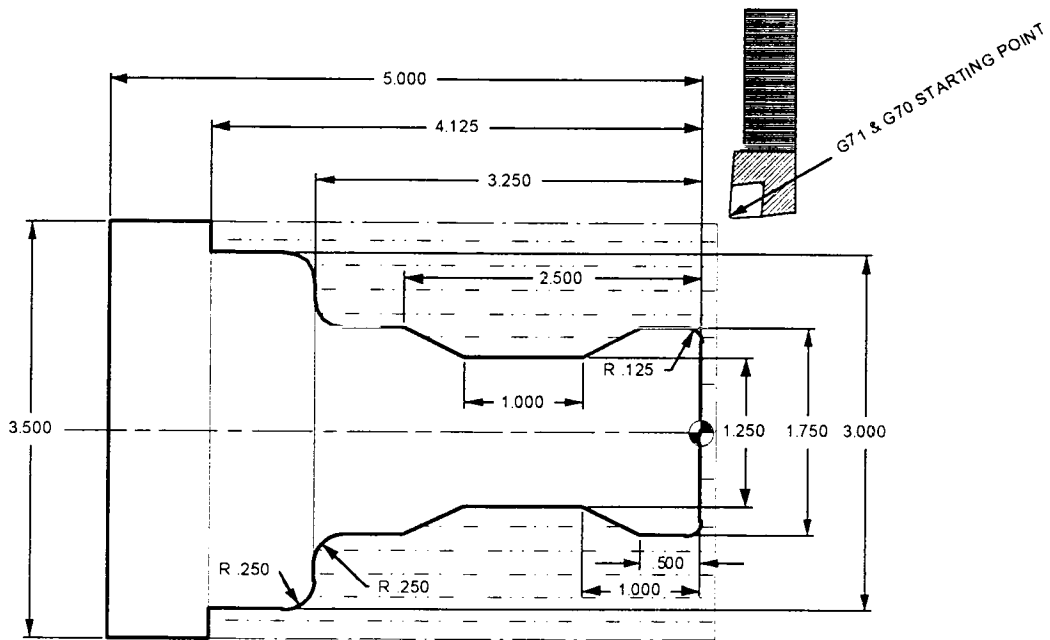
Example G71 Type II Roughing



```

%
O0135                                     (Fanuc G71 Type II roughing)
G28
T101                                       (Roughing Tool)
G97 S1200 M03
G00 G54 X2. Z0.05                          (Start position)
G71 P1 Q6 D.035 U.03 W.01 F.01
N1 G1 X1.5 Z-.5 F.004                       (PQ path definition)
N2 X1.0 Z-1.
N3 X1.5 Z-1.5
N4 Z-2.
N5 G02 X.5 Z-2.5 R.5
N6 G1 X2.
G28
T202                                       (Finishing tool)
G97 S1500 M03
G70 P1 Q6                                   (Finish pass)
G28
M30
%
```

G71 & G72 Type II Roughing Exercise



Fill in the blank line for tool #1 using a G71 Roughing Type II cycle command, to define Roughing passes for the part geometry defined between N10 and N20. Then define a finish pass using a G70 Finishing cycle command, with tool #2. Define a finish pass for the part geometry defined between N10 and N20. On the G71 command line leave .010 stock on diameters and .005 on the faces. Take a .120 depth of cut at .012 feed.

O0040	(Program number)
N1	(First operation)
T101 (O.D. Tool X .031 TNR)	(Tool #1 and Offset)
G50 S2500	(Spindle speed clamp at 2500 RPM)
G97 S400 M03	(Cancel CSS, define a 400 RPM, spindle ON forward)
G54 G___ X___ Z___ M08	(Rapid X & Z to starting location, coolant ON)
G___ S___	(Turn on CSS to 375)
Z___	(Position .005 from end of part)
G___ X___ F.005	(Feed down to rough face end of part)
G___ X___ Z___	(Rapid to start position above part)
G71 P___ Q___ U___ W___ D___ F___	(Roughing O.D. G71 cycle command)
N___ G___ X___ Z___	(Pnn starting number, Rapid X axis, Cutter comp ON)
G___ Z___ F.006	(G71 part geometry)
X___	(G71 part geometry)
G___ X___ Z___ R___	(G71 part geometry)
G___ Z___	(G71 part geometry)
X___ Z___ F.012	(G71 Type II part geometry)
Z___	(G71 Type II part geometry)
X___ Z___	(G71 Type II part geometry)
Z___	(G71 part geometry)
G___ X___ Z___ R___	(G71 part geometry)
G___ X___	(G71 part geometry)
G___ X___ Z___ R___	(G71 part geometry)
G___ Z___ F.003	(G71 part geometry)
X___ F.015	(G71 part geometry)
N___ G___ X___	(Qnnnn ending number, Cancel Cutter Comp)
G___ S___ M___	(Cancel CSS, define 400 RPM, spindle on forward, Coolant Off)
G___	(Return to reference point)
M___	(Optional stop command)
N2	(Second operation)
T___	(Tool #2 and Offset #2)
G___ S___	(Spindle speed clamp at 2500 RPM)
G___ S___ M___	(Cancel CSS, turn on spindle 1200)
G___ X___ Z___	(Rapid to start position above part)
G___ S___	(Turn on CSS to 650)
G___ P___ Q___	(Define finish pass using part geometry)
G___ S___ M___	(Cancel CSS, define 400 RPM, spindle On, coolant off)
G___	(Return to reference point)
M___	(End of program rewind)

G72 End Face Stock Removal Cycle

P	Starting block number of path to rough
Q	Ending block number of path to rough
U	X-axis size and direction of G72 rough allowance, diameter
W	Z-axis size and direction of G72 rough allowance
D	Depth of cut for each pass of stock removal, positive number
*I	X-axis size and direction of G72 finish pass allowance, radius
*K	Z-axis size and direction of G72 finish pass allowance
*S	Spindle speed to use throughout G72 PQ block
*T	Tool and offset to use throughout G72 PQ block
*F	Feed rate to use throughout G72 PQ block

* Indicates optional

This canned cycle will rough out material on a part given the finished part shape. It is similar to G71 but roughs out material along the face of a part. All a programmer needs to do is define the shape of a part by programming the finished tool path and submitting the path definition to the G72 call by means of a PQ block designation. Any feeds, spindle speeds or tools within the block defining the path are ignored by the G72 call. Any F, S or T commands on the G72 are used throughout the G72 roughing cycle. Usually, a G70 call to the same PQ block definition is used to finish the shape using the programmed feeds, speeds, tools and offsets.

Two types of machining paths are addressed with a G72 command. The first type of path (TYPE I) is when the Z-axis of the programmed path does not change direction. This is depicted in Figure 3-12. This type of path is called a monotonic path. The second type of path (TYPE II) allows the Z-axis to change direction. For both the first and second type of programmed path, the X-axis must be monotonic; it cannot change direction. Type I is selected by having only an X-axis motion in the block specified by P in the G71 call. When both an X-axis and Z-axis motion are in the P block, TYPE II roughing is assumed.

The G72 consists of a roughing phase and a finishing phase. The roughing and finishing phase are handled slightly differently for types I and types II. Generally, the roughing phase consists of repeated passes along the X-axis at the specified feed rate. The finishing phase consists of a pass along the programmed tool path to remove excess material left by the roughing phase but to leave finish material for a G70 block with perhaps a finishing tool. The final motion in either types is a return to the starting position S.

In the following, the start position S is the position of the tool at the time of the G72 call. The X clearance plane is derived from the X-axis start position and the sum of U and optional I finish allowances.

Type I Details

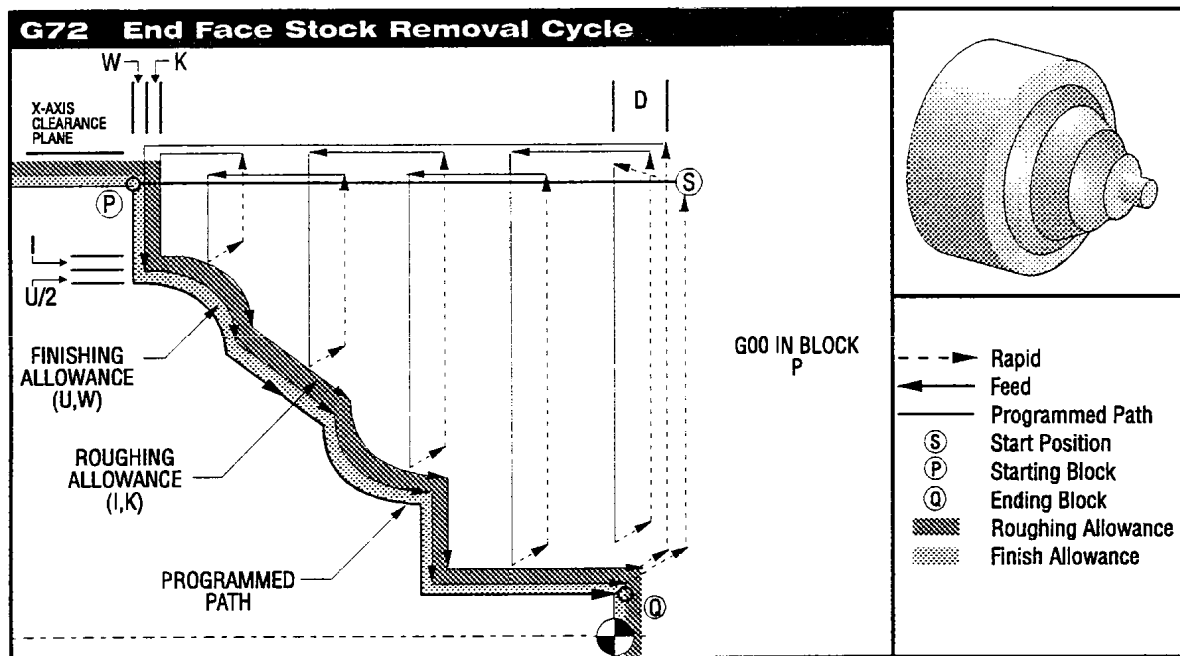
When Type I is specified by the programmer, it is assumed that the tool path is monotonic in the X-axis. Prior to any roughing motion, the tool path is checked for monotonicity and G code compliance. An alarm is generated if a problem is found.

Roughing begins by advancing from the start position S and moving to the first roughing pass. All roughing passes start and end at the X clearance plane. Each roughing pass Z-axis location is determined by applying the value specified in D to the current location. The direction that D is applied, is determined by the signs of U and W.

The nature of the movement along the X clearance plane for each roughing pass is determined by the G code in block P. If block P contains a G00 code, then movement along the X clearance plane is a rapid mode. If block P contains a G01, then movement will be at the G72 feed rate. Roughing continues until the Z-axis position in block P is exceeded.

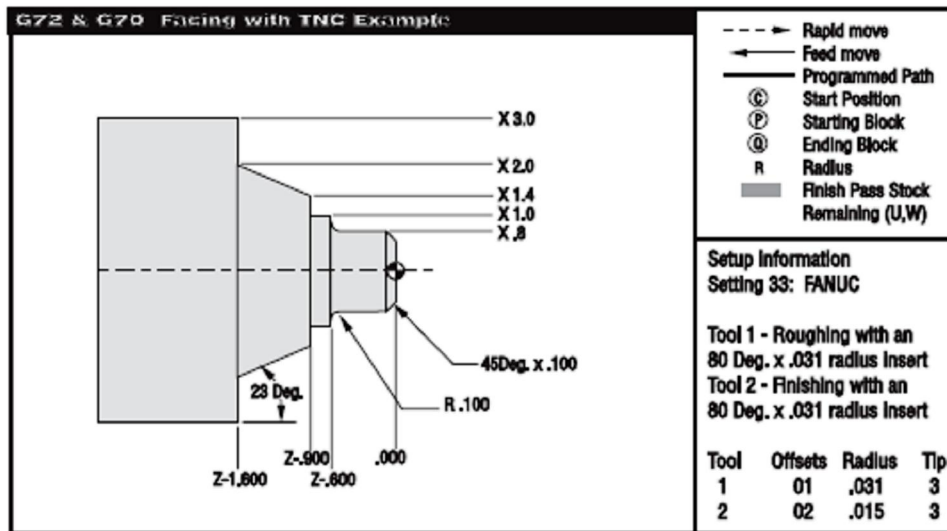
Each roughing pass is stopped before it intersects the programmed tool path allowing for both roughing and finishing allowances. The tool is then retracted from the material at a 45-degree angle by the distance specified in Setting 73. The tool then moves in rapid mode to the X-axis clearance plane.

When roughing is completed, the tool is moved parallel to the tool path to clean up the rough cut. If I and K are specified, an additional rough finish cut parallel to the tool path is performed.



Example: TNC with G72

This example uses tool nose compensation with a G72 roughing canned cycle.



Part Drawing for TNC with G72 example

Program Example

Description

%

O0814

G50 S1000

T0101

(Select Tool 1)

G0 X3.5 Z.1

(Move to point S)

G96 S100 M03

G72 P80 Q180 U.005 W.01 D.05 F.010
(Define part path PQ sequence)

(Rough P to Q with T1 using G72 and TNC)

N80 G41 G0 Z-1.6

(P)(G72 Type I, TNC Approach)

G1 X2.0 F0.005

X1.4 Z-0.9

X1.0

Z-.60

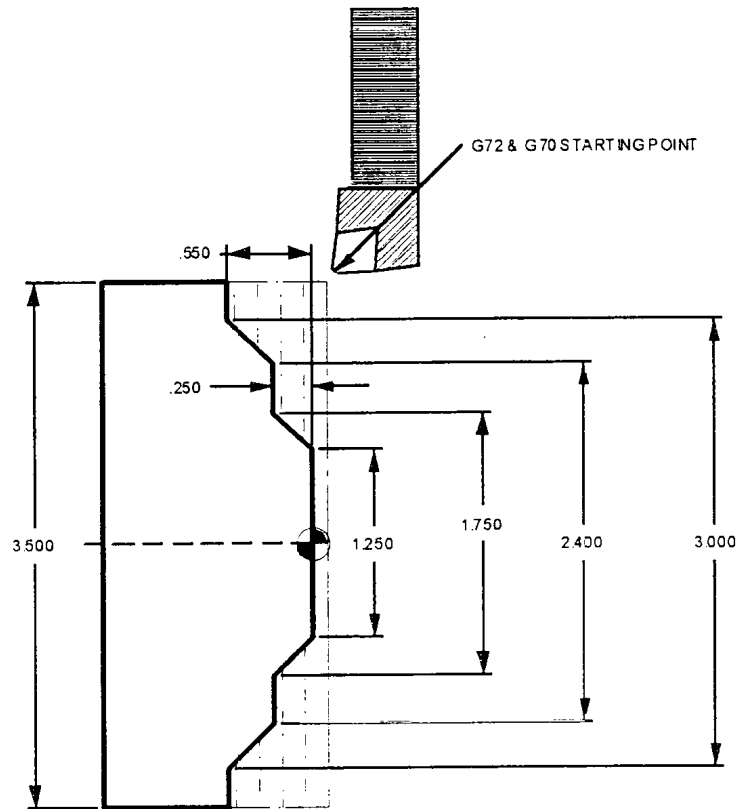
G3 X0.8 Z-0.5 K0.1

Machining a chamfer, when cutter compensation is NOT used on the control, requires that calculations must be made for the tool tip dimensions.

G1 Z-0.1	
X0.6 Z0.0	
N180 X0.0	
G40 G0 X3.1 Z0.1 M5	(TNC departure)
(*****OPTIONAL FINISHING SEQUENCE*****)	
G28	(Zero for tool change clearance)
M01	
N2 G50 S1000	
T0202	(Select tool 2)
G0 X3.5 Z.1	(Move to point S)
G96 S100 M03	
(Finish P to Q with T2 using G70 and TNC)	
G70 P80 Q180	
G0 G40 Z.5 M5	(TNC departure)
G28	(Zero for tool change clearance)
M30	
%	

G72 is used instead of G71 because the roughing strokes in X are longer than the Z roughing strokes of a G71. It is therefore more efficient to use G72. Caution should be used however because of the direction of the force used in machining with face cutting. This is especially true with large parts held on with small jaw engagement and low jaw pressures because of part deformation. These parts may be thrown out of the chuck using face cutting canned cycle.

TYPE I - G72 AND G70 Exercise with Tool Nose Compensation



Fill in the blank lines for Tool #1 using a G72 roughing cycle command to define multiple rough facing passes for the part geometry defined between N10 and N20. Define tool #2 to do a finish pass using a G70 finishing cycle command. On the G72 command line, leave .010 stock on diameters and .005 on the faces. Take .12 depth of cut at .012 feed.

O0050	(Program number)
G28	
T101 (O.D. Turning Tool .031R)	(Tool #1 and Offset #1)
G50 S2500	(Spindle speed clamp at 2500 RPM)
G97 S400 M03	(Cancel CSS, turn on spindle 700 RPM)
G54 G00 X_____ Z_____ M08	(Rapid, X, Z location, Coolant ON)
G_____ S_____	(Turn on CSS to 375)
Z_____	(Start position .050 from face of part)
G72 P___ Q___ U___ W___ D___ F___	(G72 Facing cycle command)
N___ G___ G___ Z_____	(Pnn starting number, Rapid to start, cutter comp ON)
G___ X_____ F.006	(G72 part geometry)
X_____ Z_____	(G72 part geometry)
X_____	(G72 part geometry)
X_____ Z_____	(G72 part geometry)
X_____	(G72 part geometry)
N___ G___ Z_____	(G72 Qnnnn ending number, Cancel cutter comp.)
G___ S_____	(Cancel constant surface speed)
G_____	(Return to reference point)
M_____	(Optional stop command)
T_____	(Tool #2 and Offset #2)
G___ S_____	(Spindle speed clamp at 2500 RPM)
G___ S_____ M_____	(Cancel CSS, turn on spindle 1200RPM)
G___ G___ X_____ Z_____ M_____	(Rapid, X, Z location, Coolant ON)
G___ S_____	(Turn on CSS to 650)
Z_____	(Position to .050 from end of part)
G___ P_____ Q_____	(Define finish pass using part geometry)
G___ S_____	(Cancel constant surface speed)
G___ Z_____ M_____	(Rapid 1.0 away from part in the Z-axis)
G_____	(Return to reference point)
M_____	(End of program rewind)

G73 Irregular Tool Path Stock Removal Cycle

P	Starting block number of path to rough
Q	Ending block number of path to rough
U	X-axis size and direction of G73 finish allowance, diameter
W	Z-axis size and direction of G73 finish allowance
D	Number of cutting passes, positive number, integer
I	X-axis distance and direction from finish cut to first, radius
K	Z-axis distance and direction from finish cut to first
*S	Spindle speed to use throughout G73 PQ block
*T	Tool and offset to use throughout G73 PQ block
F	Feed rate to use throughout G73 PQ block

* Indicates optional

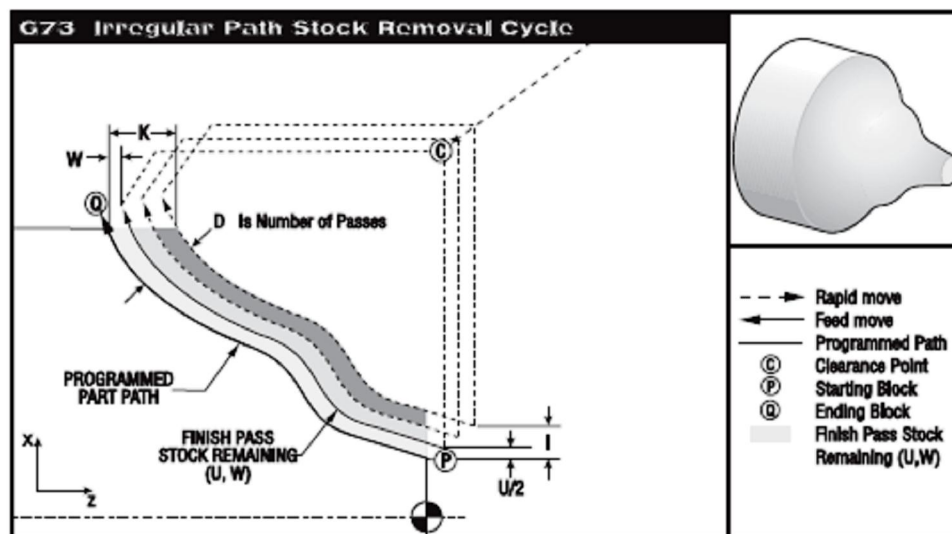
The G73 canned cycle can be used for rough cutting of castings or forgings. The canned cycle assumes that the extra material will follow the profile that is programmed between the tool path P through Q.

Machining either starts at the start position S and rapids or feeds to the first rough cut. The nature of the approach move is based on whether a G00 or G01 is programmed in block P. Machining continues parallel to the programmed tool path. When block Q is reached, a rapid departure move is executed to the Start position plus the offset for the second roughing pass. Roughing passes continue in this manner for the number of rough passes specified in D. After the last roughing pass is completed, the tool returns to the starting position S.

Any feed (F), spindle speed (S) or tool change (T) commands on the lines from P to Q are ignored and any F, S, and T prior to or in the G73 block are in effect.

The programmed tool path P through Q does not have to be monotonic in X or Z, but care has to be taken to ensure that existing material does not interfere with tool movement during approach and departure moves.

This canned cycle is intended for use with the G70 finishing canned cycle.



EXAMPLE TNC WITH G73

This example uses tool nose compensation with a G73 roughing canned cycle.

Preparation: Setting 33 FANUC

Tools:	T1	Insert with .064 radius, roughing
	T2	Insert with .016 radius, finishing
Offsets	Tool	Radius Tip
1	T1	.064 3
2	T2	.016 3

Program Example

Description

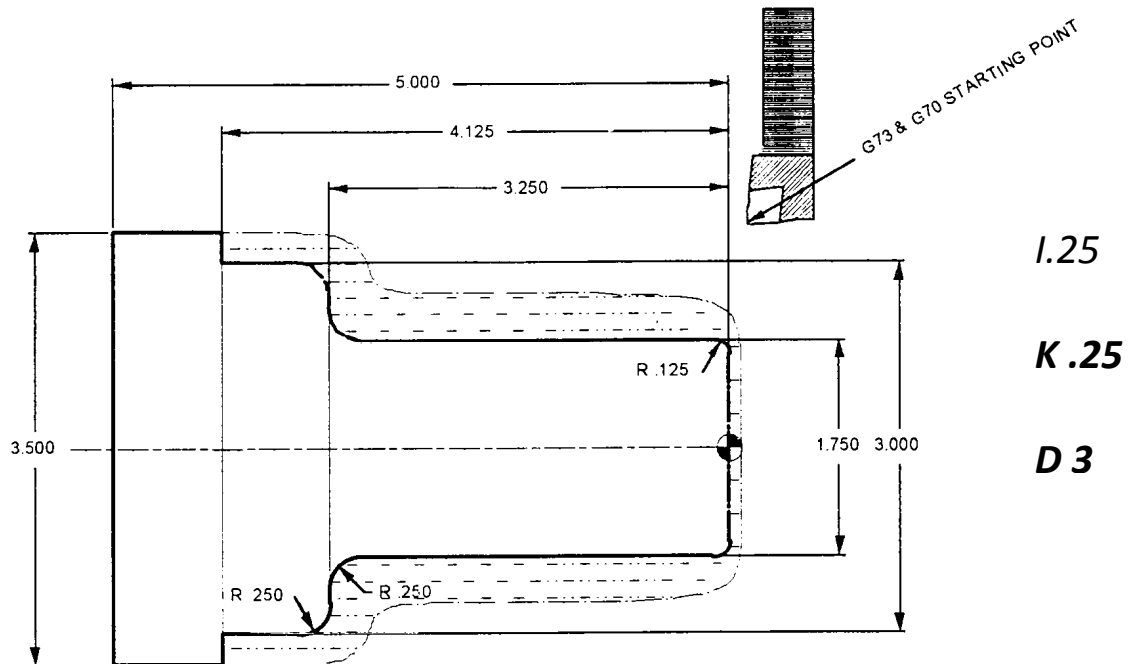
```

%
O00815
G50 S1000
T101
G00 X3.5 Z.1
G96 S250 M03
(Rough P to Q with T1 using G73 and TNC)
G73 P80 Q180 U.01 W.005 I0.3 K0.15 D4 F.012
(Define part path PQ sequence)
N80 G42 G00 X0.6
G01 Z0. F0.1
X0.8 Z-0.1 F.005
Z-0.5
G02 X1.0 Z-0.6 I0.1
G01 X1.4
X2.0 Z-0.9
Z-1.6
X2.3 G03 X2.8 Z-1.85 K-0.25
N180 G40 G01 Z-2.1
G00 X3.0 Z0.1 M05

(*****OPTIONAL FINISHING SEQUENCE*****)
G28
M01
N2 G50 S1000
T0202
G00 X3.0 Z.1
G96 S250 M03
(Finish P to Q with T2 using G70 and TNC)
G70 P80 Q180
G00 G40 Z.5 M05
G28
M30
%
```

(Example section 8.10 example 5)
 (Select Tool 1)
 (Move to point S)
 (P)(G72 Type I, TNC Approach)
 (Q)
 (TNC departure)
 (Zero for tool change clearance)
 (Select tool 2)
 (Move to point S)
 (TNC departure)
 (Zero for tool change clearance)

G73 and G70 Exercise with Tool Nose Compensation



Fill in the blank line for tool #1 using a G73 roughing cycle command defining roughing passes for the part geometry N10 and N20. Then define a finish pass using a G70 finishing cycle command, with tool #2, to define a finish pass for the part geometry defined between N10 and N20. On the G73 command line leave .010 stock on diameters and .005 on the faces. Take three passes at .012 feed.

O0060	(Program number)
N1	(First operation)
T101 (O.D. Tool X .031 TNR)	(Tool #1 and Offset #1)
G50 S2500	(Spindle speed clamp at 2500 RPM)
G97 S400 M03	(Cancel CSS, define a 400 RPM, spindle ON forward)
G54 G__ X__ Z__ M08	(Rapid X & Z to starting location, coolant ON)
G__ S__	(Turn on CSS to 375)
Z__	(Position .005 from end of part)
G__ X__ F.005	(Feed down to rough face End of part)
G__ X__ Z__	(Rapid to start position above Part)
G73 P__ Q__ U__ W__ D__ F__ I__ K__	(Roughing O.D. G73 cycle command)
N__ G__ G__ X__	(Pnn starting number, Rapid X-axis, cutter comp ON)
G__ Z__ F.006	(G73 part geometry)
X__	(G71 part geometry)
G__ X__ Z__ R__	(G71 part geometry)
G__ Z__	(G71 part geometry)
G__ X__ Z__ R__	(G71 part geometry)
G__ X__	(G71 part geometry)
G__ X__ Z__ R__	(G71 part geometry)
G__ Z__ F.003	(G71 part geometry)
X__ F.015	(G71 part geometry)
N__ G__ X__	(Qnnnn ending number, cancel Cutter comp)
G__ S__ M__	(Cancel CSS, define 400 RPM, spindle ON forward, Coolant Off)
G__	(Return to reference point)
M__	(Optional stop command)
N2	(Second operation)
T__	(Tool #2 and offset #2)
G__ S__	(Spindle speed clamp at 2500 rpm)
G__ S__ M__	(Cancel CSS, turn on spindle 1200 RPM)
G54 G__ X__ Z__ M__	(Rapid, X, Z location, Turn coolant on)
G__ S__	(Turn on CSS to 650)
G__ P__ Q__	(Define finish pass using part geometry)
G__ S__ M__	(Cancel CSS, define 400 RPM, spindle ON forward, Coolant Off)
G__	(Return to reference point)
M__	(End of program rewind)

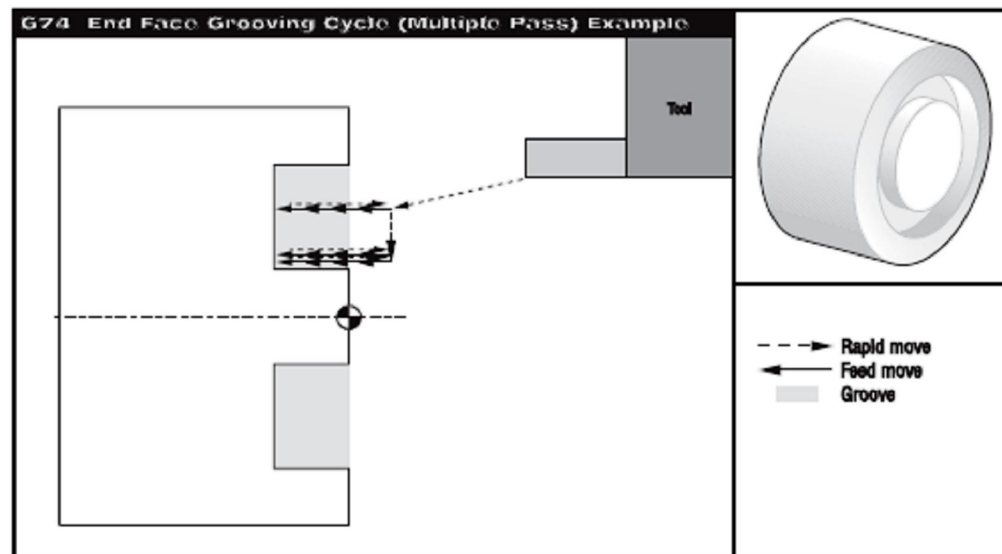
G74 End Face Grooving Cycle, Peck Drilling

Z	Z-axis absolute location total pecking depth, signed
W	Z-axis incremental distance to total pecking depth, signed
*X	X-axis absolute location of furthest peck cycle, signed diameter
*U	X-axis incremental distance to furthest peck, signed diameter
*I	X-axis size of increment between peck cycles, positive radius
*K	Z-axis size of increment between pecks in a cycle
*D	Tool clearance when returning to starting plane, positive
*F	Feedrate
*	Indicates Optional

The G74 canned cycle can be used for grooving on the face of a part for peck drilling or for turning with a chip break. With this canned cycle, either a single pecking cycle can be executed, as for drilling on the spindle centerline, or a series of pecking cycles can be performed.

When an X or U code is added to a G74 block and X is not the current position, then a minimum of two pecking cycles will occur: one at the current location and another at the X location. The I code is the incremental distance between X-axis pecking cycles. Adding an I will perform multiple, evenly spaced, pecking cycles between the starting position S and X. If the distance between S and X is not evenly divisible by I, then the last interval along X will be less than I.

When K is added to a G74 block, then pecking will be performed at each interval specified by K. The peck is a rapid move opposite the direction of feed and the peck distance is obtained from Setting 22. The D code can be used for grooving and turning to provide material clearance when returning to starting plane S.

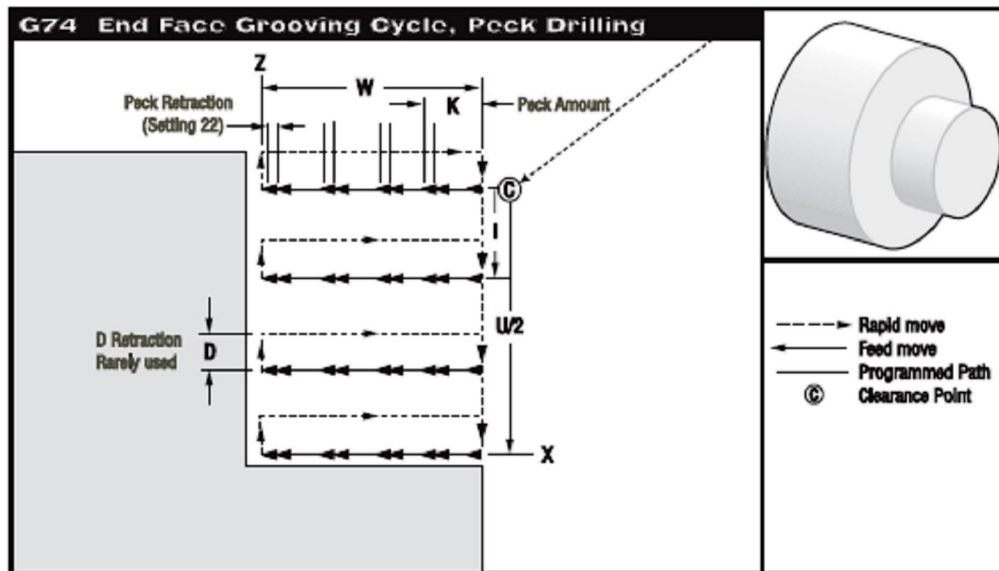


Program Example

Description

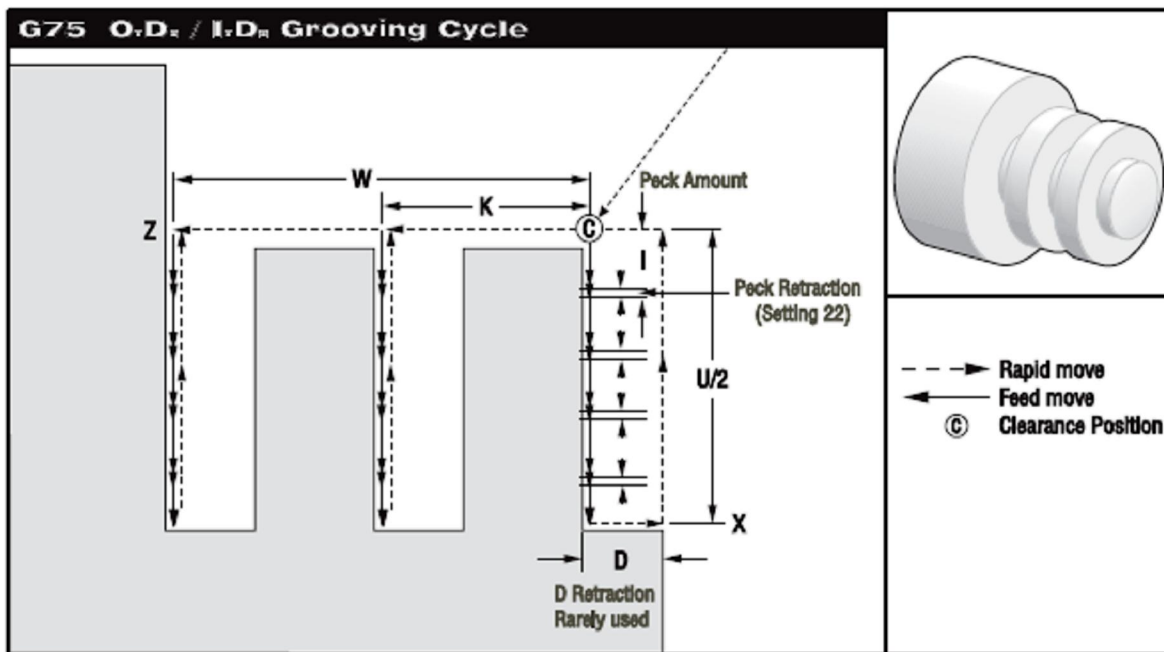
```
%  
O0074  
T101 M03 S750  
G00 X3. Z0.05  
G74 I0.2 K0.1 X1.75 Z-0.5 F0.01  
G28  
M30  
%
```

(Rapid to start position)
(Face grooving cycle multiple pass)



G75 O.D./I.D. Grooving Cycle, Peck Drilling

Z	Z-axis absolute location to furthest peck cycle, signed
W	Z-axis incremental distance to furthest peck cycle, signed
*X	X-axis absolute location total pecking depth, signed diameter
*U	X-axis incremental distance to total pecking depth
*D	Tool clearance when returning to starting plane, positive
*I	X-axis size of increment b
*K	
*F	Feed Rate
*	Indicates optional



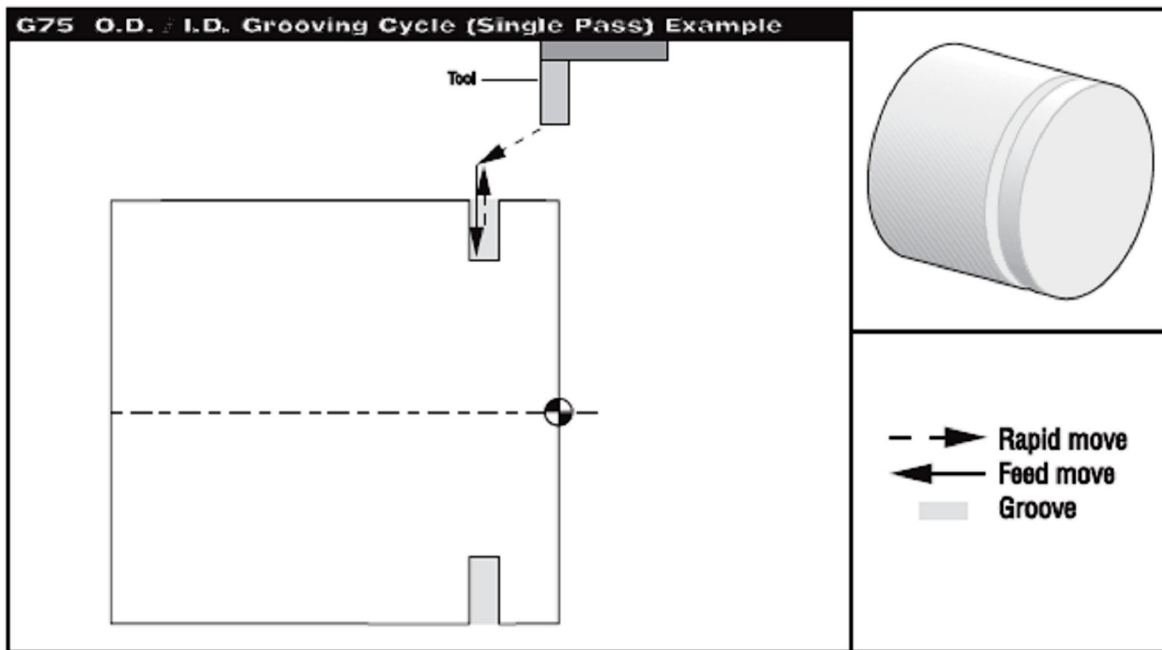
The G75 canned cycle can be used for grooving an outside diameter with a chip break. With this canned cycle, either a single pecking cycle can be executed (as for a single groove), or a series of pecking cycles can be performed (as for multiple grooves).

When adding a Z or W code to a G75 block and Z is not the current position, then a minimum of two pecking cycles will occur: one at the current location and another at the Z location. The K code is the incremental distance between Z-axis pecking cycles. Adding a K will perform multiple, **evenly spaced, pecking cycles between the starting position S and Z**. If the distance between S and Z is not evenly divisible by K, then the last interval along Z will be less than K.

When I is added to a G75 block, then pecking will be performed at each interval specified by I. The peck is a rapid move opposite the direction of feed and the peck distance is obtained from **Setting 22**.

The D code can be used for grooving to provide material clearance when returning to starting plane S.

G75 Single Pass



O00109 (G75 O.D./I.D. SINGLE PASS GROOVE CYCLE)

(Machine a .25 wide O.D. Groove with .25 Groove Tool)

N1 G28

N2 T505 (.25 WIDE O.D. GROOVE TOOL)

N3 G97 S960 M03

N4 G54 G00 X2.1 Z0.1 M08

(Rapid to clearance point)

N5 Z-0.75

(Rapid to a start point of groove)

N6 G75 X1.75 I.05 F0.005

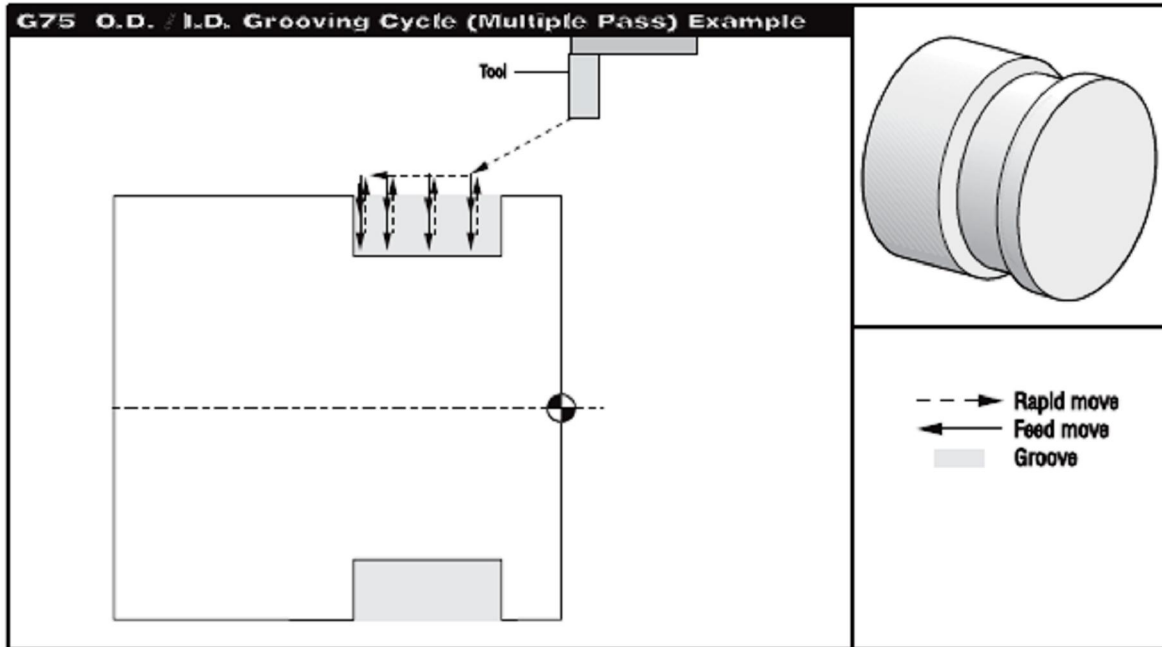
(G75 Single pass O.D. grooving cycle)

N7 M09

N8 G28

N9 M30

G75 Multiple Pass



(Machine a 1. wide O.D. Groove with .25 Groove Tool)
 O00110 (G75 O.D./I.D. MULTIPLE PASS GROOVING CYCLE)
 N1 G28
 N2 T505 (.25 WIDE O.D. GROOVE TOOL)
 N3 G97 S960 M03
 N4 G54 G00 X2.1 Z0.1 M08 (Rapid to front of part)
 N5 Z-0.75 (Rapid to start point of groove)
 N6 G75 X1.75 Z-1.5 I0.05 K0.2 F0.005 (G75 Multiple pass O.D. grooving cycle)
 N7 M09
 N8 G28
 N9 M30

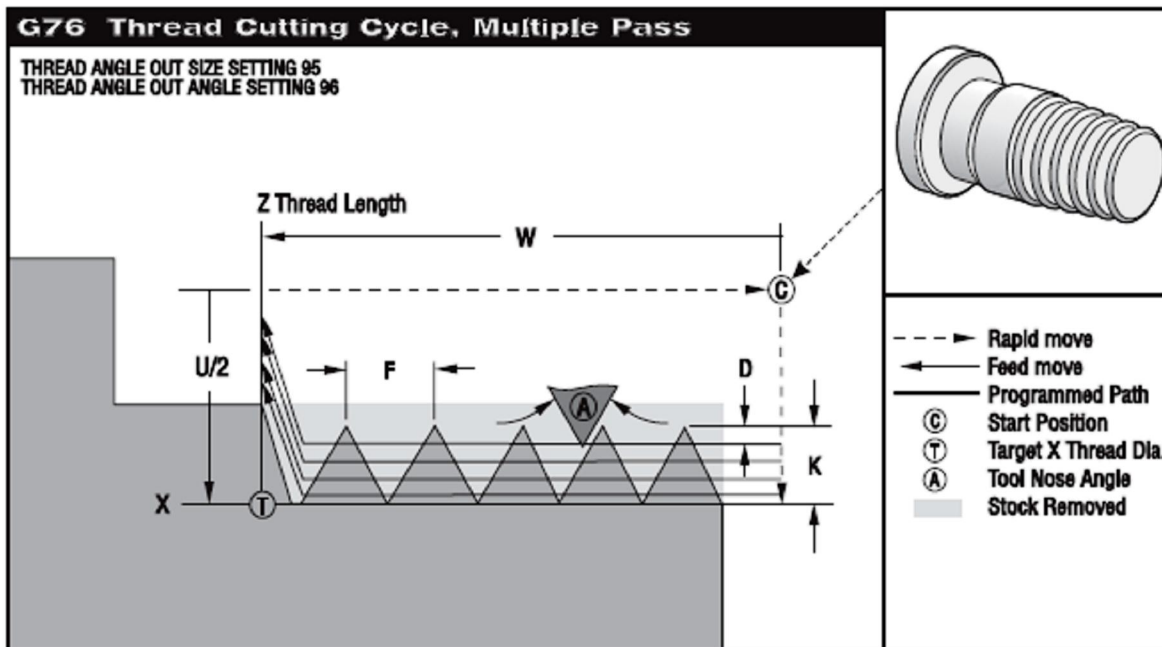
SETTING 22 (CAN CYCLE DELTA Z) - As the groove tool pecks deeper into the part, with each peck value of I, it pulls back a constant specified distance above the bottom of the groove created by the previous peck to break the chip. That specified distance it pulls back is defined in Setting 22.

G76 Thread Cutting Cycle, Multiple Pass

G00 X1.1 Z0.3

G76 X0.913 Z-.85 K.0451 D.0136 A58 F0.0714 (1-14 THREAD Minor .9132)

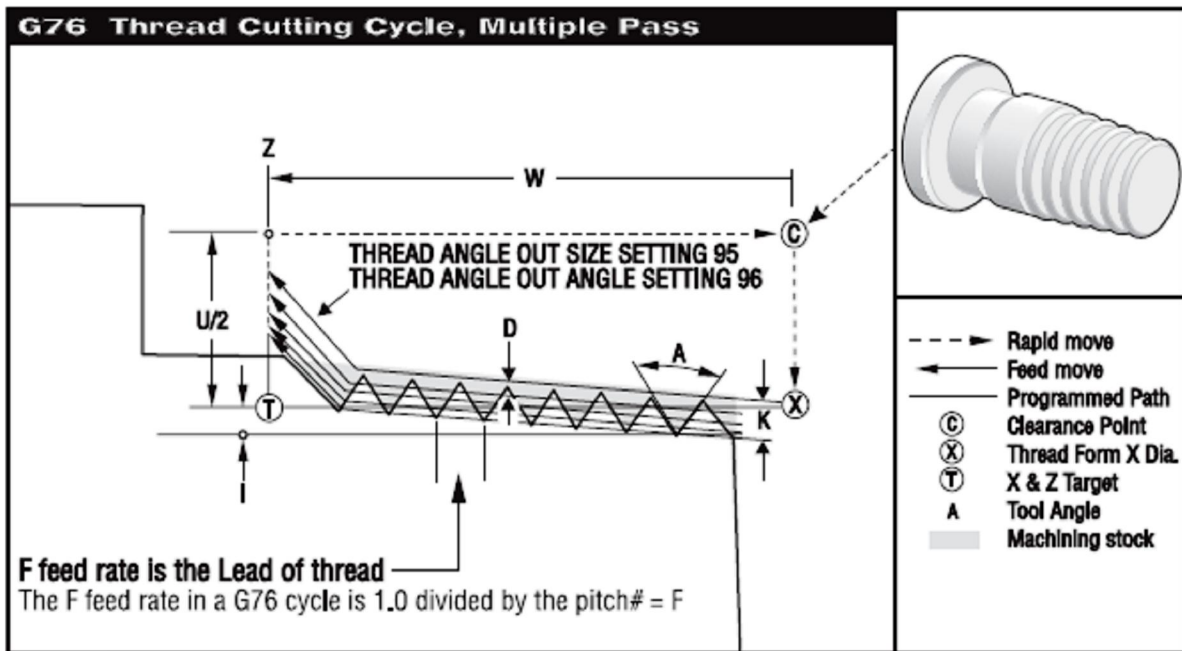
- D First pass cutting depth
- K Thread height, defines limit of multiple passes, radius measure
- *X X-axis absolute location, maximum thread I.D.
- *Z Z-axis absolute location, maximum thread length
- *U X-axis incremental distance, start to maximum thread I.D.
- *W Z-axis incremental distance, start to maximum thread length
- *I Thread taper amount, radius measure
- *P Subsequent pass positioning algorithm, cutting method
- *F(E) Feedrate (specifies feedrate for threading – also indicates the pitch or lead)
- *A Tool nose angle (from 0 to 120 degrees – if not used, then defaults to 0)
- * Indicates Optional



The G76 canned cycle can be used for threading both straight or tapered (pipe) threads. With G76, a programmer can easily command multiple cutting passes along the length of the thread. The nature of tool load and wear can be controlled by using the P code. The P code can specify which side the tool cuts on and it can specify how much material will be cut.

The **height of the thread** is specified in K. The height of the thread is defined as the distance from the crest to the root of the thread. K must agree with the direction that the X-axis is being cut. The actual depth of cut will be K less the finish allowance. **Setting 86** (THREAD FINISH ALLOWANCE) is this amount and is defaulted to 0.

The **thread taper** amount is specified in **I**. It is measured from the target position X, Z at **point T** to the beginning **position C**. A **conventional O.D. taper thread** will have a **negative I** value.

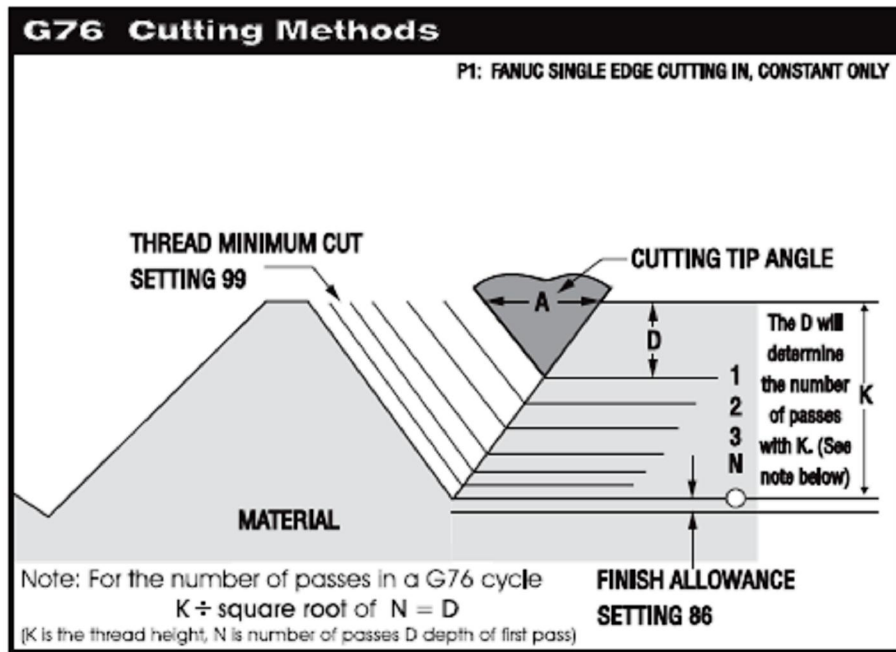


The **depth of the first cut** through the thread is specified in **D**, which also determines the number of passes over the thread based on the value of **K** and the cutting method used.

The depth of the last cut through the thread can be controlled with **Setting 99** (THREAD MINIMUM CUT). For any of the methods specified in **P**, the last cut will never be less than this value. The default value is .001 inches/.01 mm. On most carbon and alloy steels the value of minimum pass of .005 will produce acceptable finishes. For some materials .001 minimum pass will improve the surface finish of the thread.

At the end of the thread, an optional chamfer is performed. The size and angle of the chamfer is controlled with **Setting 95** (THREAD CHAMFER SIZE) and **Setting 96** (THREAD CHAMFER ANGLE). The chamfer size is designated in number of threads, so that if 1.000 is recorded in Setting 95 and the feedrate is .05, then the chamfer will be .05. A chamfer can improve the appearance and functionality of threads that must be machined up to a shoulder. If relief is provided for at the end of the thread, then the chamfer can be eliminated by specifying 0.000 for the chamfer size in Setting 95. The default value for Setting 95 is 1.000 and the default angle for the thread (Setting 96) is 45 degrees. M23 commands chamfer on. M24 commands chamfer off. M23 is the default value.

The **tool nose angle** for the thread is **A**. The value can run from 0 to 120. If no A is given 0 degrees is assumed. If no A value is given then the thread is created with a radial cut. With a radial cut both sides of the threading tool are cutting. This creates a v shaped chip which may be hard to evacuate. This may cause the insert to chip. Also chattering may occur as the entire tool is engaged at the end of the thread cycle.



If **A** has a **60** degree value, the same as the thread cutting tool, the Haas machine will make a flank cut. In a flank cut just the leading edge of the tool will do the cutting. Each successive pass will in feed the tool at 30 degrees as shown above. The advantage of using a flank cut is that the chip will flow out of thread form area easier than a radial cut. This also reduces a burr from forming on the trailing edge of the tool. To avoid rubbing on the trailing edge of the tool a modified flank cut is recommended. With a modified flank cut the value of **A** is **58** (In feed angle 29 degrees). A channel chip is formed but because of uneven thickness on the chip flows out similar to a flank cut. The disadvantage to flank or modified flank cutting is torn or poor finish may result when soft or gummy materials are cut such as low carbon steels, aluminum and stainless steels.

P Options:

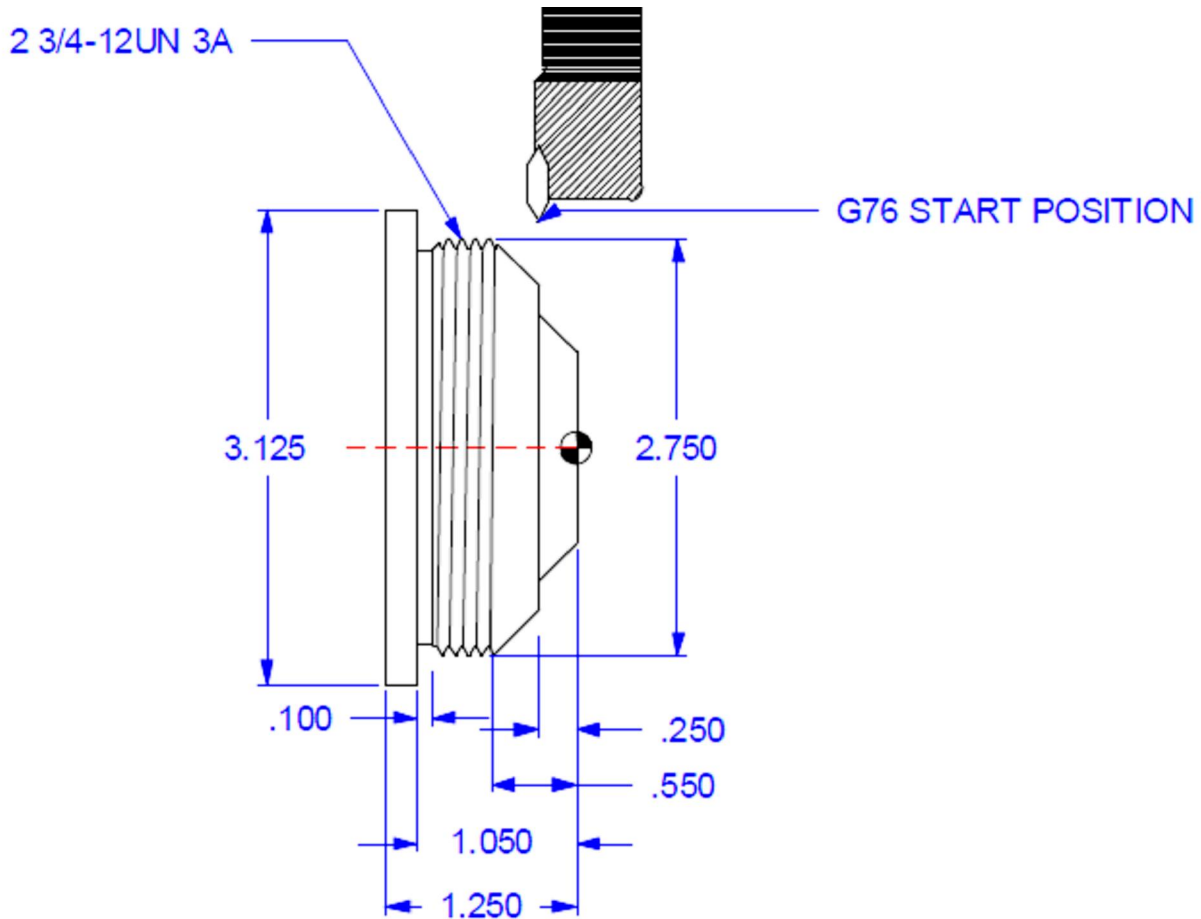
- P1:** Single cutting, cutting amount constant (default)
- P2:** Double edge cutting, cutting amount constant
- P3:** Single edge cutting, cutting depth constant
- P4:** Double edge cutting, cutting depth constant

Alternating flank cutting is possible with P2 and P4. Alternating flank cutting will increase tool life as both sides of the inserts are being used equally. Option **P2** may give superior results according to Haas.

Example of G76 code for a 1-8 Thread:

```
T404
S50 S2000
G97 S1200 M03
G00 X1.2 Z.3 M08
G76 X.8492 Z-1.50 K.0676 D.0169 A58 P2 F.125
G00 G28 U0
G28 W0.
M30
```


Thread Programming Exercise



2 3/4-12UN 3A Thread

Major Diameter 2.7500/2.7386 Minor Diameter 2.6478

```

O0010
N1 G28
N2 T101 (Thread Tool)
N3 G97 S590 M03
N4 G54 G00 X_____ Z_____ M08 (Rapid X and Z above part)
N5 Z_____ M_____ (Z-.25 is start of thread, M code for no chamfer)
N6 G76 X_____ Z_____ K_____ D_____ A___ P___ F____ (12 passes,
alternating flank, in feed cut 29 degrees)
N7M_____ (Turn Coolant off)
N8 G_____ U_____ (Take Home in X)
N9 G_____ W_____ ( Take Home in Z)
N10 M_____ (End of Program)
    
```

Drilling, Boring and Tapping Canned Cycles

These canned cycles are used to define and simplify programming for the most common Z-axis repetitive operations. Once selected, a canned cycle is active until canceled with a G80 code. There are six operations involved in every canned cycle:

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

Modal canned cycles remain in effect after they are defined and executed for each positioning of X axes in the program. Some of the canned cycle command values can also be changed after the canned cycle is defined. The commands most often changed during a canned cycle are the R plane value and the Z depth value. Modal canned cycles will be canceled with the G80, G01 or G00 command. Positioning moves during a canned cycle is performed as a rapid motion.

The operation of a canned cycle will vary according to whether incremental (U,W) or absolute (X, Z) is specified. Incremental motion is often useful in a canned cycle. If a loop count (Lnn code number) is defined within the block, the canned cycle will repeat that many times with an incremental U or W move between each cycle.

A canned cycle will only be executed in the Z-axis when positioning to a new X-axis position during a canned cycle command.

Canned Cycles for Drilling and Tapping

The following is a list of the canned cycles that can be used for drilling and tapping for the HAAS lathe controls.

Canned Cycles

G80	Canned Cycle Cancel
G81	Drill Canned Cycle
G82	Spot Drill Canned Cycle
G83	Peck Drill Canned Cycle
G84	Tapping Canned Cycle
G85	Bore in Bore out Canned Cycle
G86	Bore in Rapid out Canned Cycle
G87	Bore with Manual Retract Canned Cycle
G88	Bore in Dwell with Manual Retract Canned Cycle
G89	Bore in Dwell Bore out Canned Cycle

G Code	Z Drilling Direction	Operation at Bottom of Hole	Retraction Z Direction	Application
G81	Feed	None	Rapid	Spot drilling
G82	Feed	Dwell	Rapid	Counter boring
G83	Intermittent feed	None	Rapid	Peck drilling
G84	Feed	Spindle CCW	Feed	Tapping cycle
G85	Feed	None	Feed	Boring cycle
G86	Feed	Spindle stop	Rapid	Boring cycle
G87	Feed	Spindle stop	Manual/rapid	Back cycle
G88	Feed	Dwell, then spindle stop	Manual/rapid	Boring cycle
G89	Feed	dwell	feed	Boring cycle

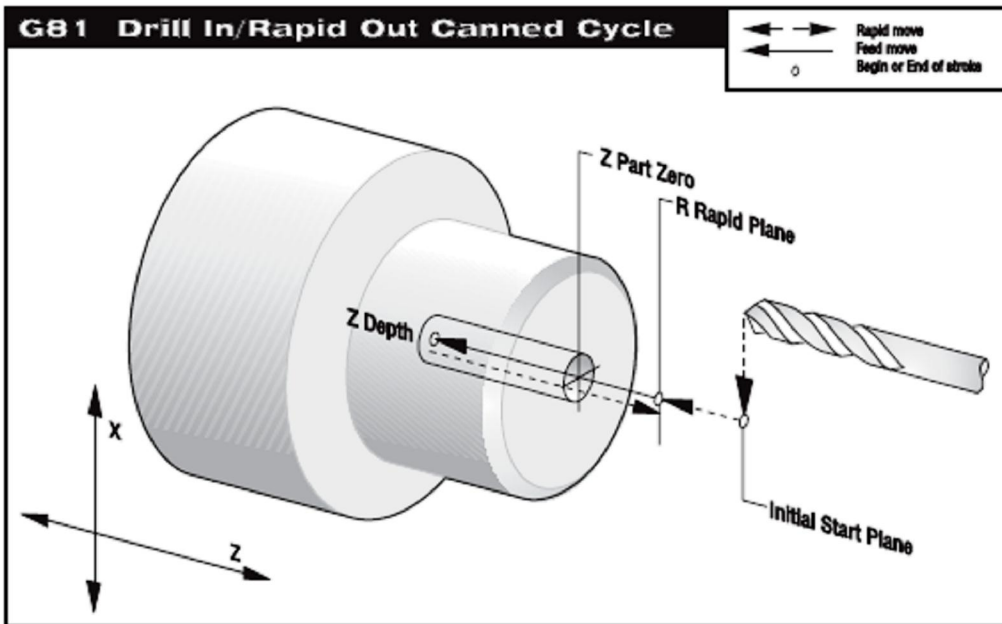
A canned cycle is presently limited to operations in the Z-axis. That is, only the G18 plane is allowed. This means that the canned cycle will be executed in the Z-axis whenever a new position is selected in the X-axis. The following is a summary of the canned cycles used on HAAS lathe controls.

G80 CANNED CYCLE CANCEL

The G code is modal; deactivating all canned cycles until a new one is selected. Note that use of G00 or G01 will also cancel a canned cycle.

G81 DRILL CANNED CYCLE

X Optional X-axis motion command
 Z Position of bottom of hole
 R Position of the R plane
 F Feed Rate



TOOL DESCRIPTION	TOOL	OFFSET	RADIUS	TIP
1/2 DIA. DRILL	1	01	0	0

O00119 (G81 Drilling)

N1 G28

N2 T101 (1/2 DIA. DRILL) (Tool 1 Offset 1)

N3 G97 S1450 M03

N4 G54 G00 X0. Z1. M08 (Rapid to Initial Start Point)

N5 G81 Z-0.625 R0.1 F0.005 (G81 Drilling Cycle)

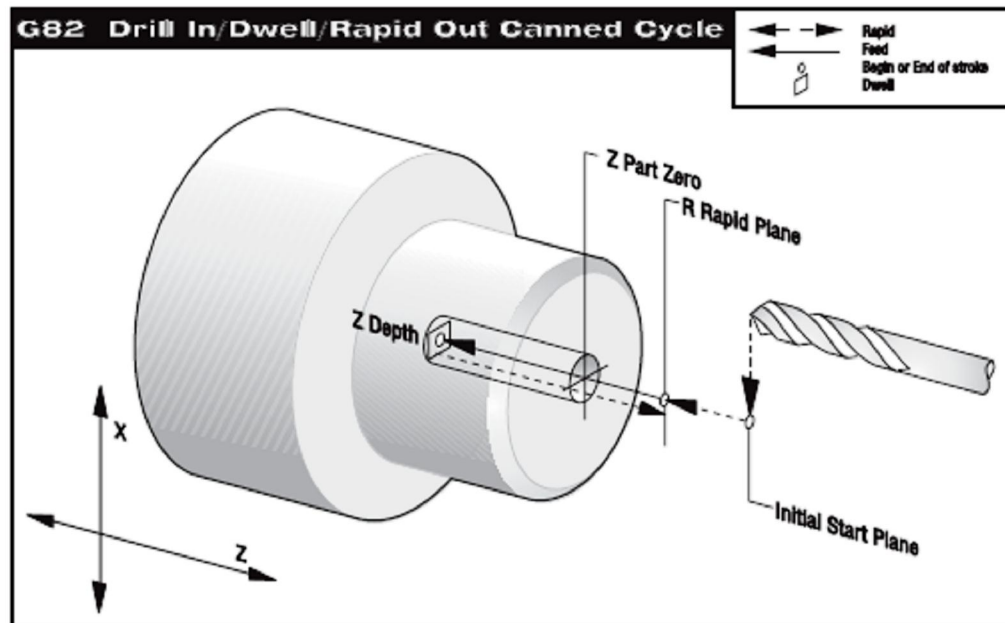
N6 G80 G00 Z1.0 M09

N7 G28

N8 M30

G82 SPOT DRILL COUNTER BORE CANNED CYCLE

- X Optional X-axis motion command
- Z Position of bottom of hole
- P The dwell time at the bottom of the hole
- R Position of the R plane
- F Feed Rate



TOOL DESCRIPTION	TOOL	OFFSET	RADIUS	TIP
1/2 DIA. FLAT BOTTOM DRILL	2	02	0	0

O00120 (G82 Drilling with a Dwell)

N1 G28

N2 T202 (1/2 DIA. FLAT BOTTOM DRILL) (Tool 2 Offset 2)

N3 G97 S1450 M03

N4 G54 G00 X0. Z1. M08 (Rapid to Initial Start Point)

N5 G82 Z-0.625 P0.5 R0.1 F0.005 (G82 Drill with a Dwell at Z Depth Cycle)

N6 G80 G00 Z1.0 M09

N7 G28

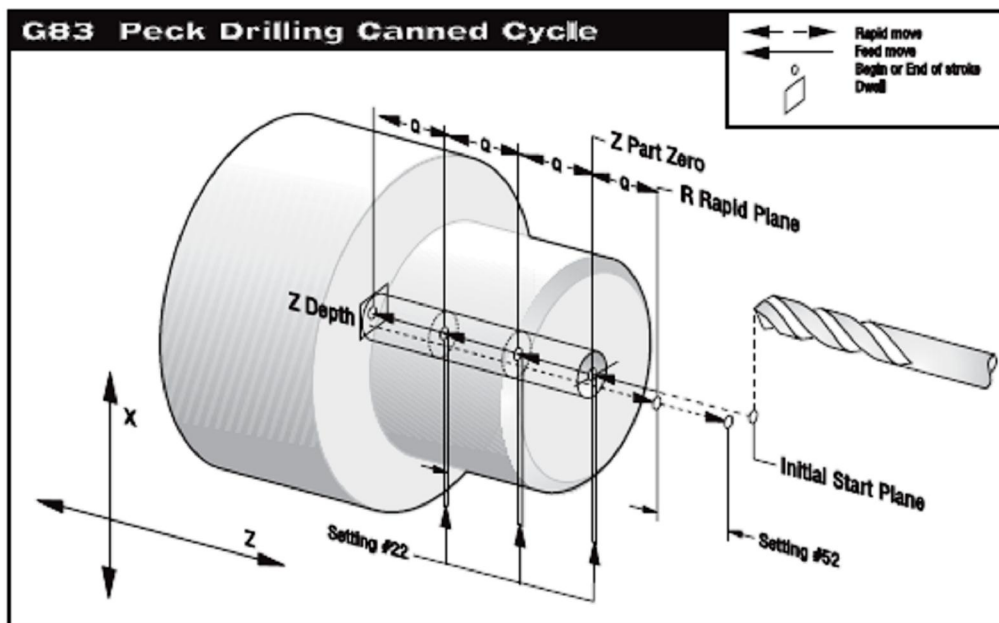
N8 M30

G83 DEEP HOLE PECK DRILLING CANNED CYCLE

X	Optional X-axis motion command
Z	Position of bottom of hole
I	Optional size of first cutting depth
J	Optional amount to reduce cutting depth each pass
K	Optional minimum depth of cut
Q	The cut-in value, always incremental
R	Position of the R plane
F	Feedrate

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

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 ensure that the chip clear motion actually allows the chips to get out of the hole. 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.



```
N4 G54 G00 X0. Z1. M08 (Rapid to Initial Start Point)
N5 G83 Z-1.5 Q0.2 R0.1 F0.005 (G83 Peck Drilling Cycle with Q)
```

The operation of this cycle is different if the rigid tapping option is installed. When rigid tapping is used, the ration between the feed rate and spindle speed must be precisely the thread pitch being cut.

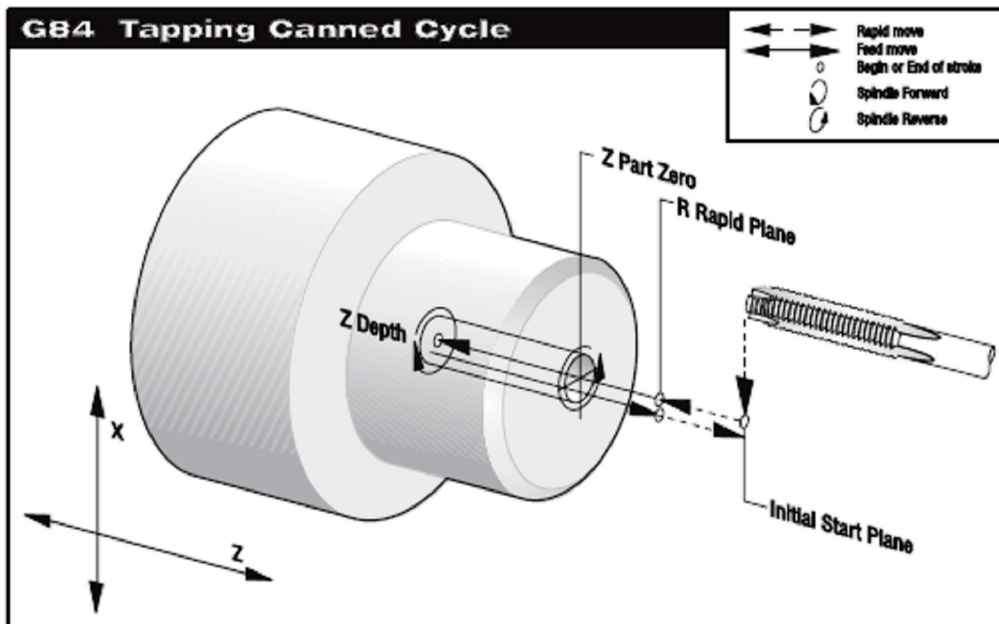
You do not need to start the spindle CW before this canned cycle. The control does this automatically.

G84 TAPPING CANNED CYCLE

X	Optional X-axis motion command
Z	Position of bottom of hole
R	Position of the R plane
F	Feed rate

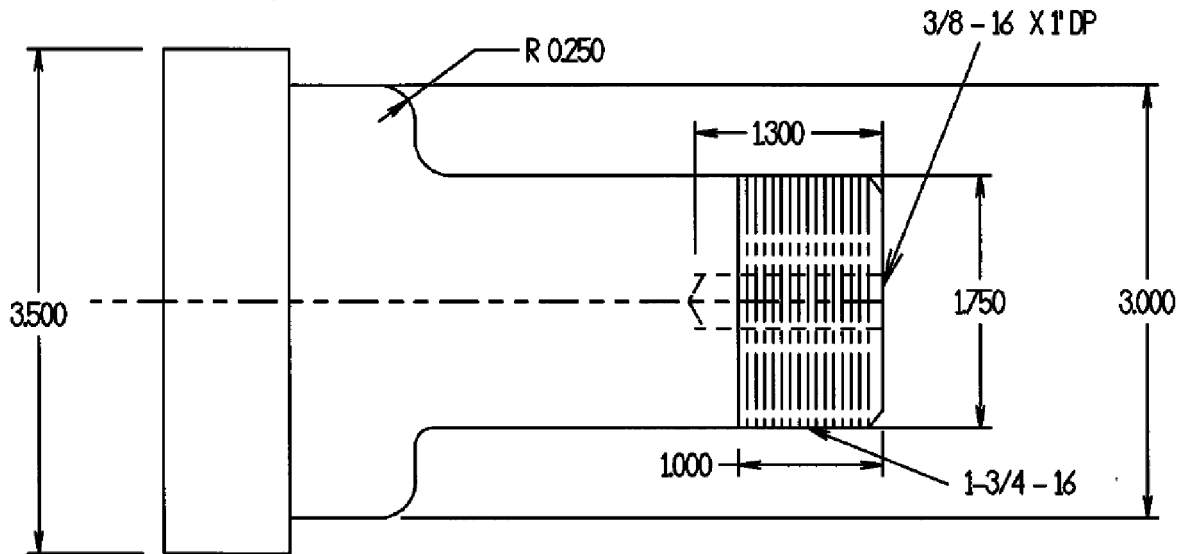
Feed rate is the **Lead** of the thread. Feed Rate = 1/ Threads per inch

G184 Reverse Tapping Canned Cycle (the same as a G84, but is used for left handed taps)



```
N2 T404 (3/8-16 TAP)      (Tool 4 Offset 4)
N3 G97 S650 M05          (G84 will turn on the spindle for you)
N4 G54 G00 X0. Z1. M08   (Rapid to Initial Start Point)
N5 G84 Z-0.5 R0.2 F0.0625 (G84 Tapping Cycle)
```


Turn Threads, Drill and Tap Exercise



Turn Thread 1-3/4 -16 UN 2A 1.0" back from face at a sfm=300

Major diameter 1.748, Minor diameter 1.6739

Drill and Tap 3/8-16 x 1.0" deep

Drill high speed 5/16 drill 1.3 deep F.006"/rev at sfm=80

Peck drill at depth of the diameter of the drill

Tap at 200 rpm

O0096

(OD THREAD)

G__ S____

G__

T404

G__ S____ M__

G__ G__ X____ Z__ M__

G__ X____ Z-____ K____ D____ A__ F____

M09

G28

M01

Clamp speed at 2000

Take Home

Start Spindle

Start Position, Turn on coolant

Thread cycle

T1111 (5/16 DRILL)

G__ S__ M__

G__ G__ X__ Z__ M__

G__ X__ Z__ R__ Q__ F__

G__

G00 Z0.2

G28

M09

M01

T1010 (3/8-16 TAP)

G97 S__ M__

G54 G__ X__ Z__ M08

G__ X__ Z__ R0. F0.0__

G80

G28

M09

M30

Start spindle

Rapid to start position, Turn on coolant

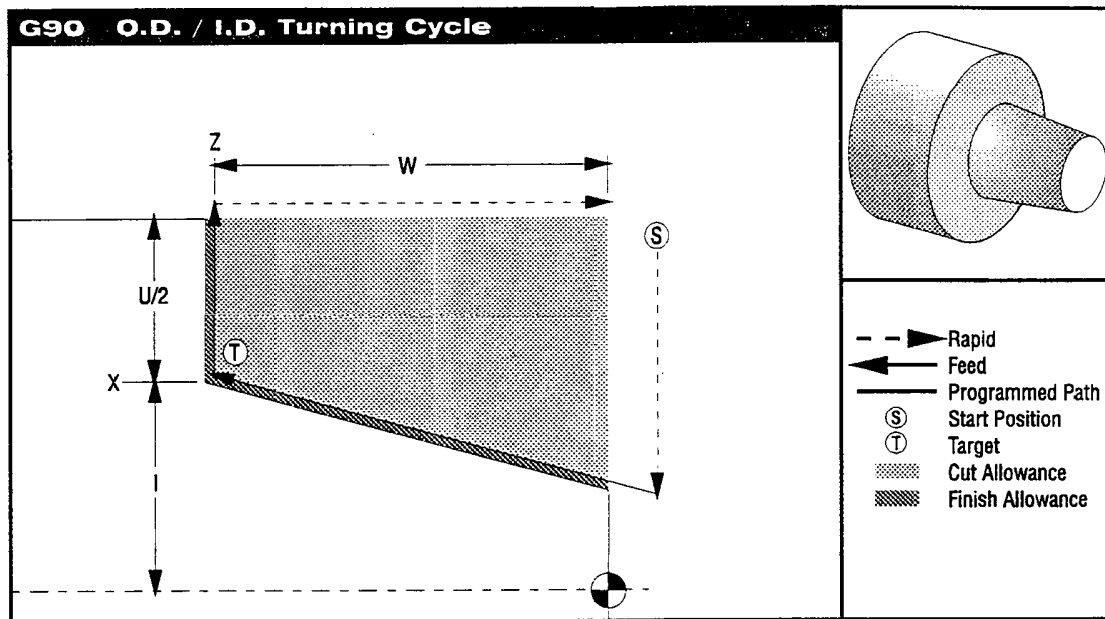
Drill Peck cycle

Cancel Canned cycle

Start spindle

Rapid to start position, turn on coolant

Tap cycle



G90 O.D./I.D. Turning Cycle

Group 01

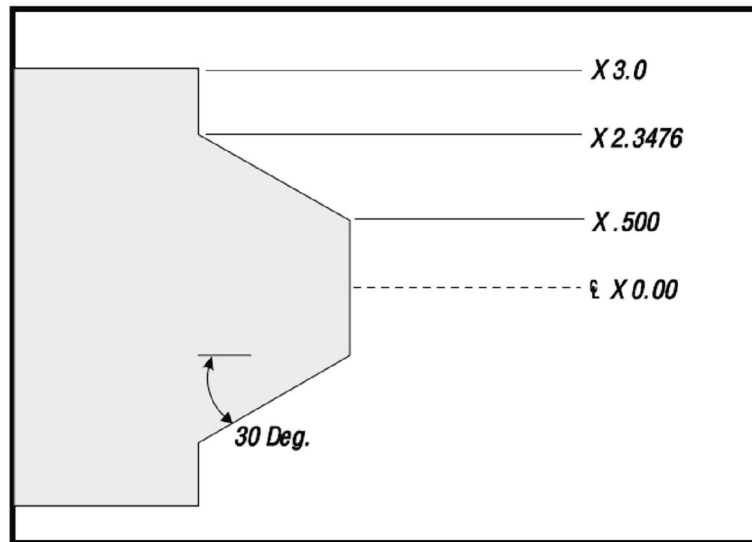
- X X-axis absolute location of target
- Z Z-axis absolute location of target
- *I Optional distance and direction of X-axis taper, radius
- *U X-axis incremental distance to target, diameter
- *W Z-axis incremental distance to target
- F (E) Feedrate
- * indicates optional

G90 is a modal canned cycle. It can be used for simple turning. Since it is modal, you can do multiple passes for turning by specifying the X locations of successive passes.

Straight turning cuts can be made by specifying X, Z, and F. By adding I, a taper cut can be made. The amount of taper is referenced from the target; I is added to the value of X at the target.

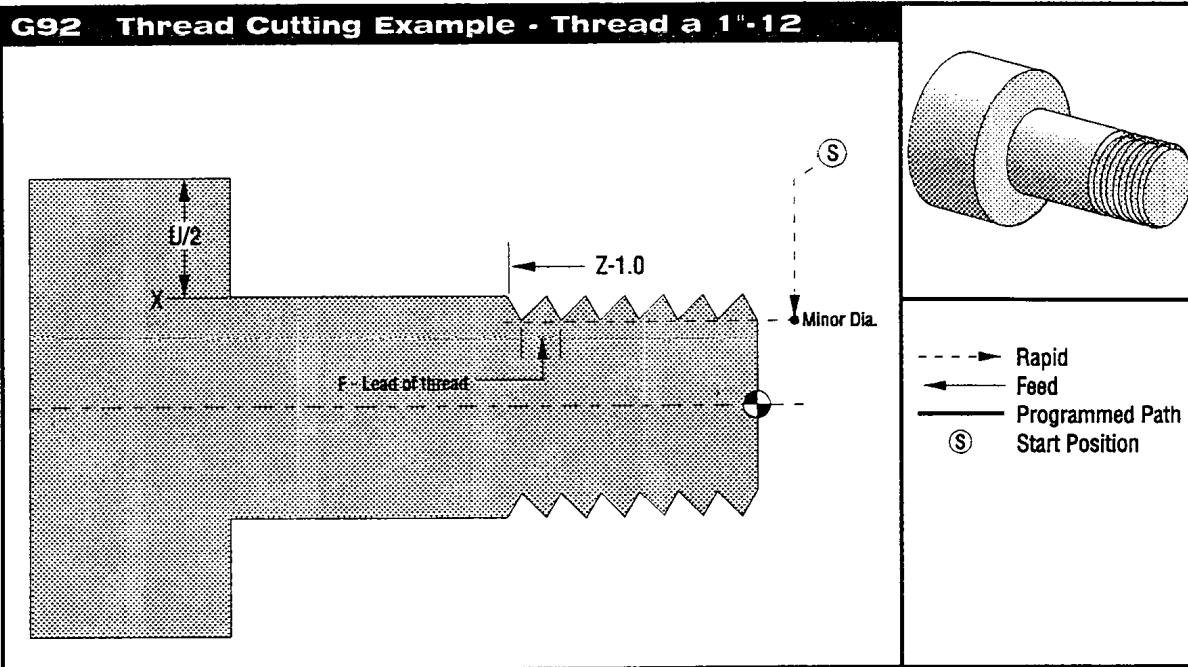
Any of the four ZX quadrants can be programmed by varying U, W, X, and Z. The taper can be positive or negative. Selecting the sign direction is not intuitive.

EXAMPLE: G90 MODAL TURNING CYCLE WITH TNC



This example uses tool nose compensation with a G90 modal rough turning cycle.

```
O00131 (G90 Modal Turning with TNC)
N11 G28
N12 T101 (O.D. TURNING TOOL)
N13 G50 S3000
N14 G97 S480 M03
N15 G54 G00 X3.1 Z1. M08 (Rapid to Start Point)
N16 G96 S390
N17 Z0.1
N18 G90 G42 X2.8 Z-1.6001 I-0.9238 F0.01 (Rough 30 Deg. angle to X2.3476
N19 X2.65 (Additional Pass) Dia. using G90 and TNC)
N20 X2.55 (Additional Pass)
N21 X2.45 (Additional Pass)
N22 X2.355 (Additional Pass)
N23 X2.3476 (Additional Pass)
N24 G00 G40 X3.1 Z1. M09 (TNC Departure)
N25 M05
N26 G28
N27 M30
```

G92 Thread Cutting Cycle

Group 01

- X X-axis absolute location of target
- Z Z-axis absolute location of target
- *I Optional distance and direction of X-axis taper, radius
- *U X-axis incremental distance to target, diameter
- *W Z-axis incremental distance to target
- F (E) Feedrate, the lead of the thread

* indicates optional

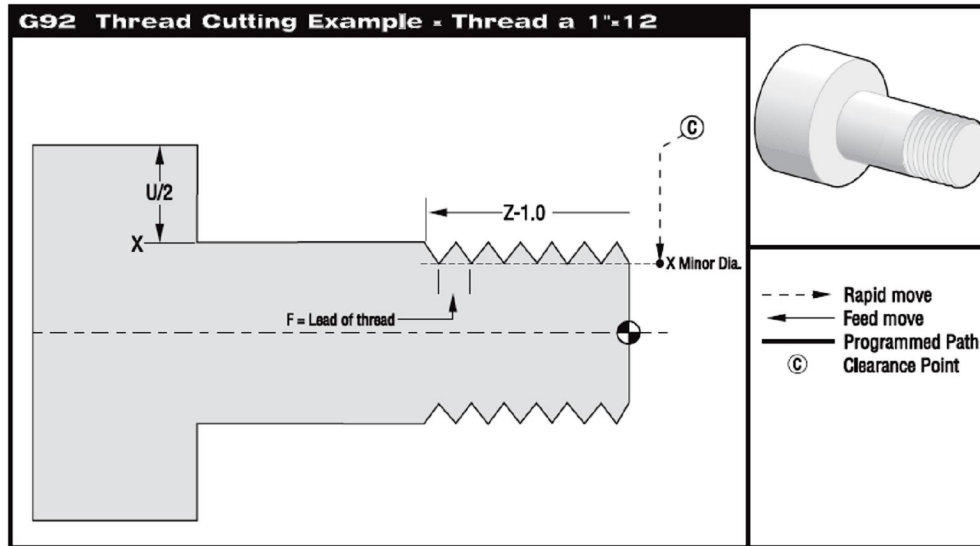
G92 is a modal canned cycle. It can be used for simple threading. Since it is modal, you can do multiple passes for threading by specifying the X locations of successive passes.

Straight threads can be made by specifying X, Z, and F. By adding I, a pipe or taper thread can be cut. The amount of taper is referenced from the target; I is added to the value of X at the target.

At the end of the thread, an automatic chamfer is executed before reaching the target default. This chamfer is one thread at 45 degrees. These values can be changed with Setting 95 and Setting 96.

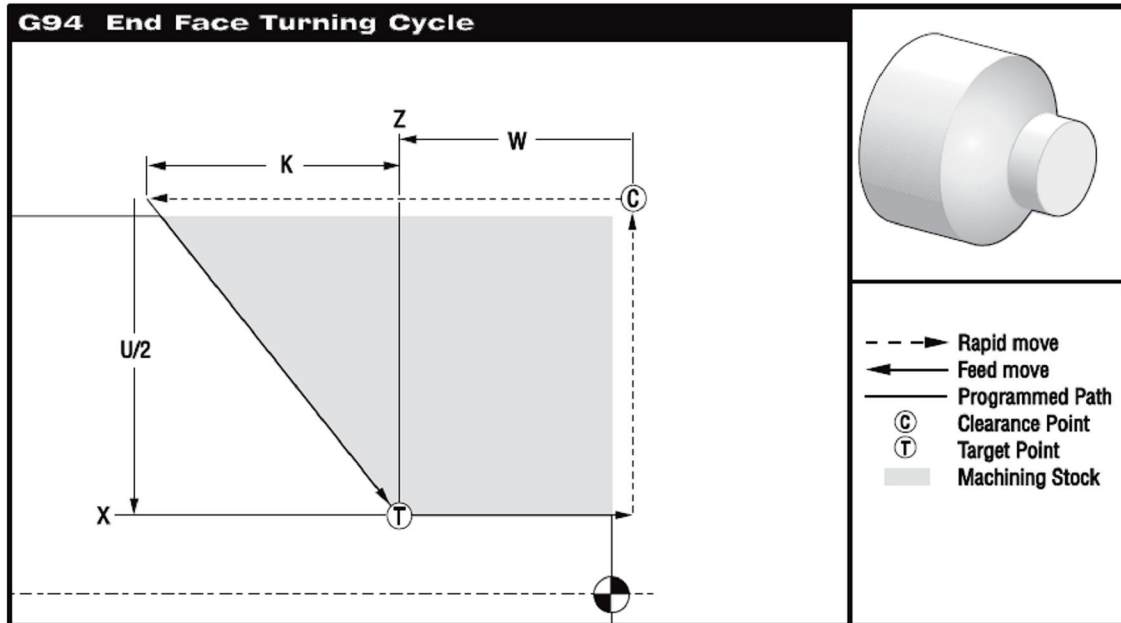
Any of the four ZX quadrants can be programmed by varying U, W, X, and Z. The taper can be positive or negative. Selecting the sign direction is not intuitive. The figure shows a few examples of the values required for machining in each of the four quadrants.

EXAMPLE: G92 MODAL THREADING CYCLE



```

O00133 (G92 Modal Threading)
N10 (1.0-12UN Thread)
N11 G28
N12 T404 (O.D. THREADING TOOL)
N13 G97 S825 M03
N14 G54 G00 X1.1 Z1. M08 (Rapid to Start Point)
N15 Z0.25
N16 G92 X.98 Z-1.05 F0.08333 M23 (First Pass of a G92 O.D. Thread Cycle)
N17 X.96 (Additional Pass)
N18 X.94 (Additional Pass)
N19 X.935 (Additional Pass)
N20 X.93 (Additional Pass)
N21 X.925 (Additional Pass)
N22 X.9225 (Additional Pass)
N23 X.92 (Additional Pass)
N24 X.9175 (Additional Pass)
N25 X.9155 (Additional Pass)
N26 X.915 (Additional Pass)
N27 X.9148 (Additional Pass)
N28 G00 X1.1 Z1. M09
N29 M05
N30 G28
N31 M30
    
```



G94 End Face Cutting Cycle

Group 01

- X X-axis absolute location of target
- Z Z-axis absolute location of target
- *U X-axis incremental distance to target, diameter
- *W Z-axis incremental distance to target
- *K Optional distance and direction of Z-axis coning
- F (E) Feedrate

* indicates optional

G94 is a modal canned cycle. It can be used for simple end facing. Since it is modal, you can do multiple passes for facing by specifying the Z locations of successive passes.

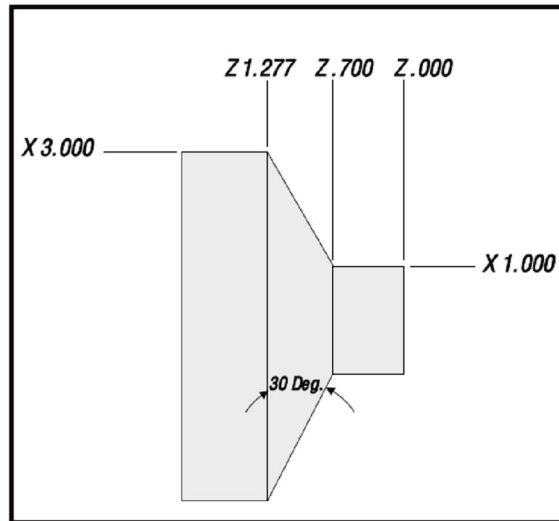
Straight end facing cuts can be made by specifying X, Z, and F. By adding K, a conical face can be cut. The amount of coning is referenced from the target. K is added to the value of X at the target.

Any of the four ZX quadrants can be programmed by varying U, W, X, and Z. The coning can be positive or negative. Selecting the sign direction is not intuitive.

EXAMPLE 7 TNC WITH G94

This example uses tool nose compensation with a G94 modal, rough turning cycle.

EXAMPLE: G94 MODAL END FACING WITH TNC



This example uses tool nose compensation with a G94 modal rough facing cycle.

```
O00135 (G94 Modal End Facing with TNC example)
N11 G28
N12 T101 (O.D. FACING TOOL)
N13 G50 S3000
N14 G97 S480 M03
N15 G54 G00 X3.1 Z1. M08 (Rapid above part)
N16 G96 S390
N17 Z.1 (rapid to start point)
N18 G94 G41 X1.0 Z-0.3 K-0.5774 F0.01 (Rough 30 Deg. angle to X1. and
N19 Z-0.4 (Additional Pass) Z-0.7 using G94 and TNC)
N20 Z-0.5 (Additional Pass)
N21 Z-0.6 (Additional Pass)
N22 Z-0.69 (Additional Pass)
N23 Z-0.7 (Additional Pass)
N24 G40 G00 X3.1 Z1. M09 (Cancel TNC)
N25 M05
N26 G28
N27 M30
```

M Code Detailed Description

M00 STOP PROGRAM

The M00 code is used to stop a program. It also stops the spindle, turns off the coolant, and stops interpretation look ahead processing. The program pointer will advance to the next block and stop. A cycle start will continue program operation from the next block.

M01 OPTIONAL PROGRAM STOP

The M01 code is identical to M00 except that it only stops if OPTIONAL STOP is turned on from the front panel. A cycle start will continue program operation from the next block.

M02 PROGRAM END

The M02 code will stop program operation the same as the M00, but does not advance the program pointer to the next block. It will not reset the program pointer to the beginning of the program as an M30 does.

M03 SPINDLE FORWARD

The M03 code will start the spindle moving in a clockwise direction at whatever speed was previously set. The block will delay until the spindle reaches about 90% of commanded speed. If bit 31 of parameter 209 (CNCR SPINDLE) is set to 1, then this command is executed at the beginning of block execution rather than the end as most M codes are.

M04 SPINDLE REVERSE

The M04 code will start the spindle moving in a counterclockwise direction at whatever speed was previously set. The block will delay until the spindle reaches about 90% of commanded speed. If bit 31 of a parameter 209 (CNCR SPINDLE) is set to 1, then this command is executed at the beginning of block execution rather than the end as most M codes are.

M05 SPINDLE STOP

The M05 code stops the spindle. The block is delayed until the spindle slows below 10 RPM.

M08 COOLANT ON

The M08 code will turn on the coolant. Note that the M code is performed at the end of a block, so that if a motion is commanded in the same block, the coolant is turned on after the motion.

M09 COOLANT OFF

The M09 code will turn off the coolant.

M10 CLAMP CHUCK

The M10 code is used to clamp the chuck. It is only used when M11 is used to unclamp the chuck.

M11 UNCLAMP CHUCK

The M11 code will unclamp the chuck.

M17 TURRET ROTATION ALWAYS FORWARD

The M17 code is modal and forces the turret to rotate in the forward direction when a tool change is made. Setting 97 TOOL CHANGE DIRECTION needs to be switched from SHORTEST to M17/M18.

T1010 M17:

This command will advance the tool turret in a FORWARD direction to position #10. Because M17 is modal, any subsequent T command will cause the turret to rotate in the forward direction to the commanded tool.

M18 TURRET ROTATION ALWAYS REVERSE

The M18 code is modal and forces the turret to rotate in the forward direction when a tool change is made. Setting 97 TOOL CHANGE DIRECTION needs to be switched from SHORTEST to M17/M18.

T1010 M18:

This command will advance the tool turret in a REVERSE direction to position #10. Because M18 is modal, any subsequent T command will cause the turret to rotate in the reverse direction to the commanded tool.

M21-M22 OPTIONAL USER M

The M21 through M22 codes are optional for user interfaces.

M25-M28 OPTIONAL USER M

The M25 through M28 codes are optional for user interfaces.

M30 PROG END AND REWIND

The M30 code is used to stop a program. It also stops the spindle and turns off the coolant. The program pointer will be reset to the first block of the program and stop. The parts counters displayed on the current commands display is also incremented. M30 will also cancel tool length offsets.

M31 CHIP CONVEYOR FORWARD

M31 starts the chip conveyor motor in the forward direction.

M32 CHIP CONVEYOR REVERSE

M32 starts the chip conveyor motor in the reverse direction.

M33 CHIP CONVEYOR STOP

M33 stops chip conveyor motion.

M51- M58 OPTIONAL USER M ON

The M51 through M58 codes are optional for user interfaces.

M61- M68 OPTIONAL USER M OFF

The M61 through M68 codes are optional for user interfaces.

M75 SET G35 OR G136 REFERENCE POINT

M76 DISABLE DISPLAYS

M77 ENABLE DISPLAYS

M78 ALARM IF SKIP SIGNAL FOUND

M79 ALARM IF SKIP SIGNAL NOT FOUND

M85 OPEN AUTOMATIC DOOR

M86 CLOSE AUTOMATIC DOOR

M97 LOCAL SUB-PROGRAM CALL

This code is used to call a subroutine referenced by a line N number within the same program. A Pnnnn code is required and must match a line number within the same program. This is useful for simple subroutines within a program and does not require the complication of a separate program. The subroutine must end with an M99. An L count on the M97 block will repeat the subroutine call that number of times.

M98 SUB PROGRAM CALL

This code is used to call a subroutine. The Pnnnn code is the number of the program being called. The Pnnnn code must be in the same block. The program by the same number must already be loaded into memory and it must contain an M99 to return to the main program. An L count can be put on the line containing the M98 and will cause the subroutine to be called L times before continuing to the next block.

M99 SUB PROGRAM RETURN OR LOOP: This code is used to return to the main program from a subroutine or macro. It will also cause the main program to loop back to the beginning without stopping if it is used in other than a subprogram without a P code. If an M99 Pnnnn is used, it will cause a jump to the line containing Nnnnn of the same number.