



Soft Servo
SYSTEMS, INC

Part Programming Manual for ServoWorks
S-100M, S-120M and S-140M

Revision 2.14
© 2010 Soft Servo Systems, Inc.

Warning

The product described herein has the potential – through misuse, inattention, or lack of understanding – to create conditions that could result in personal injury, damage to equipment, or damage to the product(s) described herein. Machinery in motion and high-power, high-current servo drives can be dangerous; potentially hazardous situations such as runaway motors could result in death; burning or other serious personal injury to personnel; damage to equipment or machinery; or economic loss if procedures aren't followed properly. Soft Servo Systems, Inc. assumes no liability for any personal injury, property damage, losses or claims arising from misapplication of its products. In no event shall Soft Servo Systems, Inc. or its suppliers be liable to you or any other person for any incidental collateral, special or consequential damages to machines or products, including without limitation, property damage, damages for loss of profits, loss of customers, loss of goodwill, work stoppage, data loss, computer failure or malfunction claims by any party other than you, or any and all similar damages or loss even if Soft Servo Systems, Inc., its suppliers, or its agent has been advised of the possibility of such damages.

It is therefore necessary for any and all personnel involved in the installation, maintenance, or use of these products to thoroughly read this manual and related manuals and understand their contents. Soft Servo Systems, Inc. stands ready to answer any questions or clarify any confusion related to these products in as timely a manner as possible.

The selection and application of Soft Servo Systems, Inc.'s products remain the responsibility of the equipment designer or end user. Soft Servo Systems, Inc. accepts no responsibility for the way its controls are incorporated into a machine tool or factory automation setting. Any documentation and warnings provided by Soft Servo Systems, Inc. must be promptly provided to any end users.

This document is based on information that was available at the time of publication. All efforts have been made to ensure that this document is accurate and complete. However, due to the widely varying uses of this product, and the variety of software and hardware configurations possible in connection with these uses, the information contained in this manual does not purport to cover every possible situation, contingency or variation in hardware or software configuration that could possibly arise in connection with the installation, maintenance, and use of the products described herein. Soft Servo Systems, Inc. assumes no obligations of notice to holders of this document with respect to changes subsequently made. Under no circumstances will Soft Servo Systems, Inc. be liable for any damages or injuries resulting from any defect or omission in this manual.

Soft Servo Systems, Inc. makes no representation or warranty, expressed, implied, or statutory with respect to, and assumes no responsibility for the accuracy, completeness, sufficiency, or usefulness of the information contained herein. NO IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS OF PURPOSE SHALL APPLY.

Important Notice

The information contained in this manual is intended to be used only for the purposes agreed upon in the related contract with Soft Servo Systems, Inc. All material contained herein is subject to restricted rights and restrictions set forth in the contract between the parties.

These manuals contain confidential and proprietary information that is not to be shared with, nor distributed to, third parties by any means without the prior express, written permission of Soft Servo Systems, Inc. No materials contained herein are to be duplicated or reproduced in whole or in part without the express, written permission of Soft Servo Systems, Inc.

Although every effort and precaution has been taken in preparing this manual, the information contained herein is subject to change without notice. This is because Soft Servo Systems, Inc. is constantly striving to improve its products. Soft Servo Systems, Inc. assumes no responsibility for errors or omissions.

All rights reserved. Any violations of contractual agreements pertaining to the materials herein will be prosecuted to the full extent of the law.

Table of Contents

Warning	i
Important Notice	ii
Table of Contents	iii
List of Tables.....	iv
List of Figures.....	v
Chapter 1: Overview of the ServoWorks Part Programming Language.....	1-1
1.1 Description of the ServoWorks Part Programming Language.....	1-1
1.2 Components of the ServoWorks Part Programming Language.....	1-1
1.3 Numerals in the ServoWorks Part Programming Language.....	1-1
1.4 Blocks of Code and Duplicate Parameters.....	1-3
Chapter 2: Address Descriptions.....	2-1
Chapter 3: Format for Part Programs.....	3-1
3.1 General Format for Part Programs.....	3-1
3.2 Format for Distance and Feedrate.....	3-1
3.3 Block Delete Code.....	3-1
3.4 Comment Code.....	3-2
3.5 Optional Skip Code.....	3-2
3.6 How the ServoWorks S-100M Series Handles Unused Data.....	3-2
Chapter 4: Order of Operations for Part Programming.....	4-1
Chapter 5: Subprogram Functions.....	5-1
5.1 Description of Subprogram Functions.....	5-1
5.2 Required Format for Subprogram Functions.....	5-1
5.3 Possible Parameters That Can Be Used With Subprogram Functions.....	5-1
5.4 Example.....	5-1
5.5 Variations on Subprogram Functions.....	5-2
5.6 Notes for Subprogram Functions.....	5-2
Chapter 6: G Codes (Preparatory Functions).....	6-1
6.1 Modes and Modal Commands.....	6-1
6.2 Summary of G Codes.....	6-2
6.3 Detailed Explanations of G Codes.....	6-6
6.3.1 Rapid Positioning (No Interpolation) (G00).....	6-6
6.3.2 Linear Interpolation (G01).....	6-8
6.3.3 Circular Interpolation (G02, G03).....	6-9
6.3.4 Helical Interpolation (G02, G03).....	6-18
6.3.5 Positive / Negative Exponential Interpolation (G02.3, G03.3).....	6-19
6.3.6 Dwell (G04).....	6-21
6.3.7 Three-Dimensional Dynamic Look-Ahead Contour Control On / Off (G05, G08).....	6-22
6.3.8 Programmable Data Input (G10).....	6-24
6.3.9 XY/ZX/YZ Plane Selection (G17, G18 and G19).....	6-29
6.3.10 Inch / Metric Data Input (G20, G21).....	6-29
6.3.11 Automatic Zero Return to/from Reference Points (G28, G29).....	6-30
6.3.12 Automatic Zero Return To Additional Reference Points (G30).....	6-32
6.3.13 Skip Cutting (G31).....	6-33
6.3.14 Automatic Tool Length Compensation Calibration (G37).....	6-34
6.3.15 Tool Radius Compensations (TRC – G40, G41 and G42).....	6-39
6.3.16 Normal Direction Control (G40.1, G41.1 and G42.1).....	6-41
6.3.17 Positive/Negative Tool Length Compensation (TLC)/ Compensation Cancel (G43, G44, G49).....	6-48
6.3.18 Tool Center Point (TCP) Control / Control Cancel (G43.2, G49) [ServoWorks S-140M Only].....	6-51
6.3.19 Scaling Off / On (G50, G51).....	6-56
6.3.20 Mirror Image Off / On (G50.1, G51.1).....	6-58
6.3.21 Local Coordinate System Selection (G52).....	6-60
6.3.22 Machine Coordinate System Selection (G53).....	6-61
6.3.23 Workpiece Coordinate Selection (G54-G59).....	6-62

6.3.24 Additional Workpiece Coordinate System Selection (G54.1).....	6-63
6.3.25 Exact Stop Check Mode, Continuous Cutting Mode and Continuous Cutting Mode with Block Rollover (G61, G64, G64.1)	6-64
6.3.26 Simple Macro Call (G65).....	6-66
6.3.27 Coordinate System Rotation/Coordinate System Rotation Cancel (G68, G69)	6-67
6.3.28 High-Speed Peck Drilling Cycle (G73).....	6-70
6.3.29 Counter Tapping Cycle (G74).....	6-73
6.3.30 Counter Rigid Tapping Cycle (G74)	6-76
6.3.31 Fine Boring Cycle (G76).....	6-80
6.3.32 Canned Cycle Cancel (G80).....	6-83
6.3.33 Drilling Cycle, Spot Boring (G81).....	6-84
6.3.34 Drilling Cycle (Dwell) (G82).....	6-87
6.3.35 Peck Drilling Cycle (G83).....	6-90
6.3.36 Tapping Cycle (G84).....	6-93
6.3.37 Rigid Tapping Cycle (G84).....	6-96
6.3.38 Boring Cycle (G85).....	6-100
6.3.39 Boring Cycle (Spindle Stop) (G86).....	6-103
6.3.40 Back Boring Cycle (G87).....	6-106
6.3.41 Boring Cycle (Dwell) (G89).....	6-110
6.3.42 Absolute / Incremental Programming (G90, G91)	6-113
6.3.43 Workpiece Coordinate Programming (G92)	6-114
6.3.44 Feed per Minute / Feed per Revolution (G94, G95).....	6-115
6.3.45 Return to Initial Point / R Point in Canned Cycle (G98, G99).....	6-116
6.3.46 Linear Interpolation Feedrate Include / Exclude Rotary Axes (G310, G311).....	6-117
Chapter 7: M Codes (Miscellaneous Codes).....	7-1
Chapter 8: Spindle Functions and S Codes.....	8-1
Chapter 9: Tool Functions and T Codes.....	9-1
9.1 Overview of T Codes.....	9-1
9.2 Tool Offsets	9-1
Chapter 10: Direct Digital Output.....	10-1
Index	I

List of Tables

Table 1-1: Summary of Numeric Interpretation in the ServoWorks Part Programming Language.....	1-2
Table 2-1: Summary of Address Descriptions (1 of 3).....	2-1
Table 2-2: Summary of Address Descriptions (2 of 3).....	2-2
Table 2-3: Summary of Address Descriptions (3 of 3).....	2-3
Table 4-1: Order of Operations.....	4-1
Table 6-1: Summary of G Codes (1 of 4)	6-2
Table 6-2: Summary of G Codes (2 of 4).....	6-3
Table 6-3: Summary of G Codes (3 of 4).....	6-4
Table 6-4: Summary of G Codes (4 of 4).....	6-5
Table 6-5: Rotary Axis No. and Corresponding Axis Names.....	6-43
Table 7-1: Summary of M Codes	7-1

List of Figures

Figure 1-1: Typical Block of Code	1-1
Figure 3-1: Typical Format for Part Programs	3-1
Figure 5-1: Main Program (Example)	5-1
Figure 5-2: OMoveXY.dat (Subprogram Example)	5-2
Figure 6-1: Rapid Traverse	6-6
Figure 6-2: Linear Interpolation	6-8
Figure 6-3: Circular Interpolation	6-9
Figure 6-4: G02/G03 in the XY, ZX and YZ Planes	6-10
Figure 6-5: G02/G03 Limitation	6-11
Figure 6-6: Positive and Negative Arc Radii for Circular Interpolation	6-12
Figure 6-7: Unacceptable Parameters for Circular Interpolation	6-13
Figure 6-8: Arc Movement Produced by Incorrect Parameters for Circular Interpolation	6-13
Figure 6-9: Error – Invalid Center Point for Circular Interpolation	6-14
Figure 6-10: Circle Error Allowance Example #1	6-15
Figure 6-11: Circle Error Allowance Example #3 (1 of 2)	6-16
Figure 6-12: Circle Error Allowance Example #3 (2 of 2)	6-17
Figure 6-13: Helical Interpolation	6-18
Figure 6-14: Exponential Interpolation	6-19
Figure 6-15: Automatic Return to Reference Points (G28)	6-30
Figure 6-16: Automatic Return from Reference Points (G29)	6-30
Figure 6-17: Automatic Return to Additional Reference Points (G30)	6-32
Figure 6-18: Difference between G01 Linear Interpolation and G31 Skip Cutting	6-33
Figure 6-19: Tool Length Compensation Frame in the Tool Compensation Screen of Configuration Mode	6-36
Figure 6-20: G37 Limit	6-36
Figure 6-21: G37 Execution Showing Different G37 Ratios	6-37
Figure 6-22: Tool Radius Offset	6-39
Figure 6-23: Sample Tool Movement during Normal Direction Control	6-41
Figure 6-24: Definition of Angle for Normal Direction Control	6-42
Figure 6-25: Normal Direction Control Frame in the Normal Direction Control Screen of Configuration Mode ..	6-42
Figure 6-26: Normal Direction Control, Example of G41.1	6-44
Figure 6-27: Normal Direction Control, Example of G42.1	6-44
Figure 6-28: Normal Direction Control Example of Angle Limit – Figure 1 of 4	6-45
Figure 6-29: Normal Direction Control Example of Angle Limit – Figure 2 of 4	6-46
Figure 6-30: Normal Direction Control Example of Angle Limit – Figure 3 of 4	6-46
Figure 6-31: Normal Direction Control Example of Angle Limit – Figure 4 of 4	6-47
Figure 6-32: Tool Length Offset	6-48
Figure 6-33: Tool Center Point	6-51
Figure 6-34: Definition of Axis 4 for G43.2, Tool Center Point	6-52
Figure 6-35: Definition of Axis 5 for G43.2, Tool Center Point	6-52
Figure 6-36: TCP Parameters Showing Arm Length and Tool Length	6-53
Figure 6-37: TCP Frame in the Machine Screen of Configuration Mode	6-53
Figure 6-38: The Settings Mode Window – Tool Offset Compensation Screen	6-54
Figure 6-39: TCP Calculations for Axes 4 and 5	6-55
Figure 6-40: Scaling Function, Example #1	6-56
Figure 6-41: Scaling Function, Example #2	6-56
Figure 6-42: Mirror Image Function	6-59
Figure 6-43: G52 Explanation	6-60
Figure 6-44: G52 Example	6-60
Figure 6-45: Workpiece Coordinate Selection Example	6-62
Figure 6-46: Coordinate System Rotation Function	6-67
Figure 6-47: Coordinate System Rotation Example	6-68
Figure 6-48: G73 Parameters (with G98 in effect)	6-70
Figure 6-49: G73 Parameters (with G99 in effect)	6-71

Figure 6-50: G74 Parameters (with G98 in effect)	6-73
Figure 6-51: G74 Parameters (with G99 in effect)	6-74
Figure 6-52: G74 Parameters (with G98 in effect)	6-77
Figure 6-53: G74 Parameters (with G99 in effect)	6-77
Figure 6-54: G76 Parameters (with G98 in effect)	6-80
Figure 6-55: G76 Parameters (with G99 in effect)	6-81
Figure 6-56: G76 Oriented Spindle Stop, Spindle/Tool Shift within Hole.....	6-81
Figure 6-57: G81 Parameters (with G98 in effect)	6-84
Figure 6-58: G81 Parameters (with G99 in effect)	6-85
Figure 6-59: G82 Parameters (with G98 in effect)	6-87
Figure 6-60: G82 Parameters (with G99 in effect)	6-88
Figure 6-61: G83 Parameters (with G98 in effect)	6-90
Figure 6-62: G83 Parameters (with G99 in effect)	6-91
Figure 6-63: G84 Parameters (with G98 in effect)	6-93
Figure 6-64: G84 Parameters (with G99 in effect)	6-94
Figure 6-65: G84 Parameters (with G98 in effect)	6-97
Figure 6-66: G84 Parameters (with G99 in effect)	6-97
Figure 6-67: G85 Parameters (with G98 in effect)	6-100
Figure 6-68: G85 Parameters (with G99 in effect)	6-101
Figure 6-69: G86 Parameters (with G98 in effect)	6-103
Figure 6-70: G86 Parameters (with G99 in effect)	6-104
Figure 6-71: G87 Parameters (with G98 in effect)	6-107
Figure 6-72: G87 Oriented Spindle Stop, Spindle/Tool Shift within Hole.....	6-107
Figure 6-73: G89 Parameters (with G98 in effect)	6-110
Figure 6-74: G89 Parameters (with G99 in effect)	6-111
Figure 6-75: Absolute/Incremental Programming Example	6-113
Figure 6-76 Workpiece Coordinate Programming Example	6-114
Figure 6-77: G94 and G95	6-115
Figure 6-78: Drilling Cycle Showing the Difference between G98 and G99	6-116

Chapter 1: Overview of the ServoWorks Part Programming Language

1.1 Description of the ServoWorks Part Programming Language

The ServoWorks part programming language consists of blocks of code made up of F codes, G codes, M codes, N codes, S codes, T codes, etc.

The most commonly used code is the G code, which is why this coding system is also sometimes referred to as “G code” or “G-code part programming.” The G Codes are a wide range of preparatory codes that prepare the machine to treat the information that follows in a distinct manner and to execute it. In essence, G codes determine the mode of the ServoWorks CNC system. G Codes are described in detail in *Chapter 6: G Codes (Preparatory Functions)*.

M codes define miscellaneous functions related to CNC machine control. These codes work like on/off switches for the functions they control. The M codes are described in detail in *Chapter 7: M Codes (Miscellaneous Codes)*.

S codes define the spindle speed command. S codes are described in *Chapter 8: Spindle Functions and S Codes*.

T codes define the tool selection command. T codes are described in *Chapter 9: Tool Functions and T Codes*.

1.2 Components of the ServoWorks Part Programming Language

The G-code coding system used by ServoWorks is made up of **words**. These words are strung together to form **blocks of code**. A block of code (or “instruction”) is one or more words together that are grouped together on one line.

A word is an **address** followed by numbers. The address defines the meaning of the number that follows the address. In other words, ServoWorks uses the address to determine what the number stands for. A list of addresses recognized by ServoWorks is included in *Chapter 2: Address Descriptions*.

A typical block of code starts with the sequence number of the block, and looks like this:

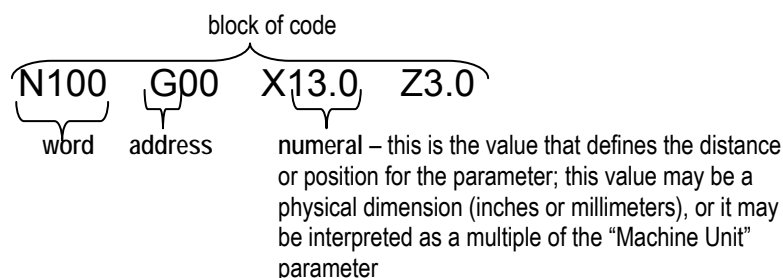


Figure 1-1: Typical Block of Code

1.3 Numerals in the ServoWorks Part Programming Language

The numerals that accompany the parameters X and Z for G codes can be floating point values or integer values.

Floating-point numerals are always absolute values defining the distance or position for that parameter. These values are physical dimensions that you can set to be interpreted as either millimeters or inches, by using the G20/21 inch/metric settings in a part program. (See *Section 6.3.10: Inch / Metric Data Input (G20, G21)*.)

Integer numerals may be interpreted one of two ways, depending on the setting of the “Integer Programming with Machine Unit Enable” parameter specified in ServoWorks S-100M/S-120M/S-140M:

- 1) When the “Integer Programming with Machine Unit Enable” parameter is disabled, integer values in the part program are treated as absolute values defining distance or position – physical values in either mm or inch (depending on G20/21 inch/metric setting in the NC program).
- 2) When the “Integer Programming with Machine Unit Enable” parameter is enabled, integer values are treated as the multiple of the “Machine Unit” parameter. The machine unit is the least input increment for the ServoWorks CNC system – the minimum unit of linear movement or distance that the actuator can be commanded to move.

See the *Reference Manual for ServoWorks CNC Parameters and Functions* for more information on the “Integer Programming with Machine Unit Enable” parameter and the “Machine Unit” parameter.

The following table summarizes the way numerals in the ServoWorks part programming language are interpreted:

DATA TYPE	DATA UNIT WHEN INTEGER PROGRAMMING IS DISABLED	DATA UNIT WHEN INTEGER PROGRAMMING IS ENABLED
Integer	mm or inch	machine unit parameter setting
Floating-Point	mm or inch	mm or inch

Table 1-1: Summary of Numeric Interpretation in the ServoWorks Part Programming Language

As an example, with Integer Programming with Machine Unit Enable = 0, the “14” in “G00 X14 Y14.5” would be interpreted as 14.0 mm or 14.0 inches, and the “14.5” would be interpreted as 14.5 mm or 14.5 inches. With Integer Programming with Machine Unit Enable = 1, the “14” in “G00 X14 Y14.5” would be interpreted as 14.0 machine units, and the “14.5” would still be interpreted as 14.5 mm or 14.5 inches. If the machine unit is set to 0.001 mm, then “X14” will be converted to 0.014 mm.



It is very important that you are aware of this distinction in how integer numerals can be interpreted, or you could get unintended results.

1.4 Blocks of Code and Duplicate Parameters

It's very important for each block to be on a separate line. If you try to put two blocks on the same line, they get combined together. This could have serious consequences: let's say you wanted to move from Point A to Point B in the first block, and from Point B to Point C in the second block. If you put both blocks on the same line, the parameters from the second block are ignored, and you will get movement from Point A to Point B only – there will be no movement to Point C.

Any time a parameter is defined more than once on the same line, the first instance of that parameter is used for the execution of the block. Any other instances of that parameter are ignored. For instance, in the block "N100 G00 X13.0 Z3.0 X2.0 X9.5," only the X13.0 would be considered. The other two instances of the X parameter (X2.0 and X9.5) would be ignored.



CAUTION

If you put two or more blocks on the same line, you are likely to get unintended consequences.

The maximum length of a block of code is 85 characters per line.

Chapter 2: Address Descriptions

The address defines the meaning of the number that follows the address. An address may have more than one meaning. If an address has more than one meaning, the multiple meanings are listed here.

NOTE: ServoWorks S-100M, S-120M and S-140M control three, four or five coordinated CNC axes, respectively, but you can choose from among 6 axis names to name each axis: X, Y, Z, A, B and C. That is why there are often 6 parameters for many G codes (X, Y, Z, A, B and C – coordinate values for the X, Y, Z, A, B and C axes).

ADDRESS	DESCRIPTION
A	A axis absolute or incremental coordinate value designation
B	B axis absolute or incremental coordinate value designation
C	C axis absolute or incremental coordinate value designation
D	<ol style="list-style-type: none"> 1. tool diameter code – integer value: interpreted as an index in the tool offset table (example: “G41 D1”) 2. tool diameter code – floating-point value: interpreted as a nominal tool diameter value (example: “D12.5”) 3. D axis absolute or incremental coordinate value designation
E	E axis absolute or incremental coordinate value designation
F	<ol style="list-style-type: none"> 1. feedrate [NOTE: feedrate is modal – the feedrate specified with this code is effective until a new value is specified; it doesn’t need to be specified for each block.] 2. initial feedrate for exponential interpolation (G02.3, G03.3) 3. cutting feedrate for drilling, tapping or boring (G73, G74, G76, G81, G82, G83, G84, G85, G86, G87, G89)
G	preparatory function (see <i>Chapter 6: G Codes (Preparatory Functions)</i>)
H	address of tool length offset value (G37, G41, G42, G43, G43.2, G44)
I	<ol style="list-style-type: none"> 1. arc center modifier for the X axis (G02, G03) 2. helix angle in exponential interpolation (G02.3, G03.3) 3. scaling factor for X axis (G50, G51)
J	<ol style="list-style-type: none"> 1. arc center modifier for the Y axis (G02, G03) 2. taper angle in exponential interpolation (G02.3, G03.3) 3. scaling factor for Y axis (G50, G51)
K	<ol style="list-style-type: none"> 1. arc center modifier for the Z axis (G02, G03) 2. scaling factor for Z axis (G50, G51) 3. number of times to repeat drilling, tapping or boring cycles (G73, G74, G76, G81, G82, G83, G84, G85, G86, G87, G89)
L	<ol style="list-style-type: none"> 1. data category (G10) 2. number of times to repeat macro (G65) 3. 16-bit direct digital output (see <i>Chapter 10: Direct Digital Output</i>)

Table 2-1: Summary of Address Descriptions (1 of 3)

ADDRESS	DESCRIPTION
M	miscellaneous function (See <i>Chapter 7: M Codes (Miscellaneous Functions)</i>)
N	block / sequence number
P	<ol style="list-style-type: none"> 1. subprogram name call 2. dwell time in milliseconds (G04) 3. 3D-DLACC on/off (G05, G08) 4. data index (G10) 5. axis number (G10, for L20, L30, L108 or L10909) 6. specifies which additional reference point (G30) 7. scaling factor for all axes (G50, G51) 8. number of workpiece coordinate system to select (G54.1) 9. macro number (G65) 10. dwell time at bottom of hole (G74, G76, G82, G84, G87, G89)
Q	<ol style="list-style-type: none"> 1. the error allowance of the circular motion (G02, G03) 2. end point feedrate for exponential interpolation (G02.3, G03.3) 3. depth of cut for each cutting feed (G73, G83) 4. tool shift amount at bottom of hole (G76, G87)
R	<ol style="list-style-type: none"> 1. subprogram repetition 2. arc radius designation (G02, G03) 3. constant for exponential interpolation (G02.3, G03.3) 4. data value (G10) 5. smoothing mode (G10, for L106 or L107) 6. smoothing time (G10, for L108) 7. position loop gain value (G10, for L10909) 8. angular displacement in degrees (G68) 9. distance to point R level from initial level (G73, G74, G76, G81, G82, G83, G84, G85, G86, G87, G89)
S	spindle speed function (See <i>Chapter 8: Spindle Functions and S Codes</i>)
T	tool function (See <i>Chapter 9: Tool Functions and T Codes</i>)
U	U axis absolute or incremental coordinate value designation
V	V axis absolute or incremental coordinate value designation
W	W axis absolute or incremental coordinate value designation
X	<ol style="list-style-type: none"> 1. X axis absolute or incremental coordinate value designation (or end point, center point, coordinate component, etc.) 2. dwell time in seconds (G04) 3. data value for X axis (G10) 4. X surface position when calibrated tool length offset is 0 (G37) 5. X axis tool length compensation (G43, G44) 6. hole position data for the X axis (G73, G74, G76, G81, G82, G83, G84, G85, G86, G87, G89)

Table 2-2: Summary of Address Descriptions (2 of 3)

ADDRESS	DESCRIPTION
Y	<ol style="list-style-type: none"> 1. Y axis absolute or incremental coordinate value designation (or end point, center point, coordinate component, etc.) 2. data value for Y axis (G10) 3. Y surface position when calibrated tool length offset is 0 (G37) 4. Y axis tool length compensation (G43, G44) 5. hole position data for the Y axis (G73, G74, G76, G81, G82, G83, G84, G85, G86, G87, G89)
Z	<ol style="list-style-type: none"> 1. Z axis absolute or incremental coordinate value designation (or end point, center point, coordinate component, etc.) 2. data value for Z axis (G10) 3. Z surface position when calibrated tool length offset is 0 (G37) 4. Z axis tool length compensation (G43, G44) 5. distance to bottom of hole from point R (G73, G74, G76, G81, G82, G83, G84, G85, G86, G89) 6. distance from bottom of hole to point Z (G87)

Table 2-3: Summary of Address Descriptions (3 of 3)

NOTE: D, F, M, S and T are all independent codes, with no parameters (i.e. "F1500" is a valid, single line of code).

Chapter 3: Format for Part Programs

3.1 General Format for Part Programs

A program is made up of blocks of code. A program typically starts with the program number, followed by blocks of code, and ending with either M02 or M30 (explained in *Chapter 7: M Codes (Miscellaneous Functions)*).

A typical program looks like this:

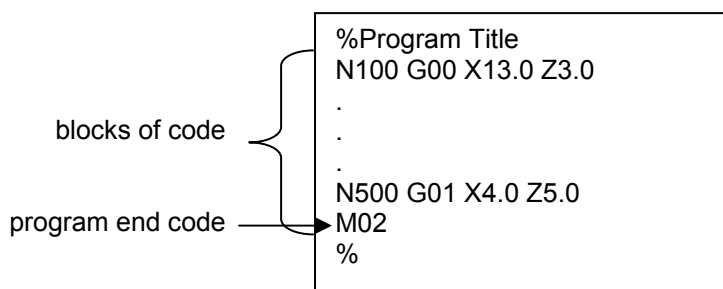



Figure 3-1: Typical Format for Part Programs

Either the “%” symbol or a carriage return is required at the end of all part program files.

 CAUTION
<p>No lower-case letters are recognized except for a subprogram name. If a lower case letter is accidentally included other than for a subprogram name, it will be ignored.</p>

3.2 Format for Distance and Feedrate

All numbers must include a decimal point unless you intend to specify an integer number. To enter negative numbers, use a minus sign (e.g. X-5.0).

3.3 Block Delete Code

Block delete code can be included in part programs as an aid to the programmer and to anyone who may want to use or edit the code at some time in the future. Specify block delete code by placing a “%” at the beginning of a line. Block delete code is not executed – the “%” unconditionally comments out the whole G code line. ””

Block delete code must be on separate lines from blocks of code. A “%” placed anywhere on a line except the beginning of the line will be ignored, and the words following the “%” will be executed, or will cause an error.

An example of block delete code follows:

```
% This is a line of block delete code
```

3.4 Comment Code

Comment code can also be included in part programs as an aid to the programmer and to anyone who may want to use or edit the code at some time in the future. Comment code is specified as comment code by placing it in between parenthesis “(” and “)” anywhere on a G code line.

An example of comment code:

```
N01 G00 X0 Z0 (This is a comment)
```

3.5 Optional Skip Code

Optional Skip Code refers to code that is skipped (not executed) when the Optional Skip switch in Auto Mode is turned on. You can specify a line of code as Optional Skip Code by placing a “/” (forward slash) at the beginning of the line. The forward slash must be the first character in the line of code; otherwise, you will get a parse error.

3.6 How the ServoWorks S-100M Series Handles Unused Data

Unused data is data that is not recognized by ServoWorks as conforming to the Part Programming Language. Such code will be ignored.

Chapter 4: Order of Operations for Part Programming

The order of operations for part programming determines the order in which ServoWorks S-100M series applications process a block of code.

	DESCRIPTION	EXAMPLE
1	N Numbers (Block Numbers)	N1234
2	Comment Statements	(this is a comment)
3	Block Delete Codes	“%”
4	Subprogram Calls	M98
5	Optional Skip Codes	“/”
6	G Code Modals	G01, G90
7	Auxiliary Codes	M08, S1000, T01
8	One-Shot G Codes	G04, G53, G92, G54
9	Motion Commands	X100.0 Y100.0 F3000

Table 4-1: Order of Operations

Chapter 5: Subprogram Functions

5.1 Description of Subprogram Functions

If there are repeated blocks or sequences, then they can be stored as a subprogram. A subprogram can be called from the main program as needed. M98 is used to call the subprogram and M99 is used to return operation from the subprogram back to main program.

The subprogram is the same as a normal program, except that the end of subprogram is M99. The name of subprogram itself always begins with “O” as the first character.

5.2 Required Format for Subprogram Functions

M98 P R
M99

5.3 Possible Parameters That Can Be Used With Subprogram Functions

P – subprogram name (without the first character “O”)
R – number of repetitions (maximum number of repetitions: 999,999)

5.4 Example

```
%  
(Main Program)  
(Subprogram: OMoveXY)  
  
G20 (Inch mode)  
G53 X0 Y0 (Back to machine Zero)  
G92 X0 Y0 (Reset work coordinate)  
#101 = 1.0 (Global macro variable for X movement destination)  
#102 = 2.0 (Global macro variable for Y movement destination)  
M98 PMoveXY R5 (Movement subprogram call)  
M02 (Program end)  
%
```

Figure 5-1: Main Program (Example)

```
%  
OMoveXY  
  
G04 (Exact stop)  
G10 L108 X50 Y50 (Set X and Y axes Acc/Dec time 50ms)  
G90 (Absolute positioning)  
G01 X#101Y#102 F50.0 (Linear move XY)  
G04 (Exact stop)  
G10 L108 X1000 Y1000 (Set X and Y axes Acc/Dec time 1sec)  
G00 X0Y0 (Rapid back to program zero)  
#101 = #101 + 1.0 (Increment X movement destination)  
#102 = #102 + 2.0 (Increment Y movement destination)  
M99 (Subprogram return)  
%
```

Figure 5-2: OMoveXY.dat (Subprogram Example)

5.5 Variations on Subprogram Functions

The subprograms can also call other subprograms (which can in turn call other subprograms). The nesting depth of subprogram calls is up to eight levels.

5.6 Notes for Subprogram Functions

- When R is omitted, then the subprogram is just called once.
- The maximum number of repeated callings is 999,999.
- The main program and its associated subprograms should be stored in same file folder.
- The subprogram call (M98) cannot be with any motion in the same block. If it is, the motion in that block will be ignored.

Chapter 6: G Codes (Preparatory Functions)

6.1 Modes and Modal Commands

G codes are categorized into groups. Most groups are modal groups, as explained in the next paragraph. Only Group 00 is nonmodal.

A **mode** is a particular functioning condition. A **modal group** is a group of two or more related modes. Only one mode in a modal group can be active at a time. The **default mode** is the mode in a modal group that is automatically activated if no other mode from that modal group is programmed in the code.

Modal codes are codes that set a mode. That mode stays in effect until it is cancelled by another code from the same modal group. Many G codes are modal. F codes, which specify feedrate, are also modal.

For instance, modal group 01 (selection of a movement system) is made up of four modes: G00 (rapid positioning), G01 (linear interpolation), G02 (clockwise circular interpolation), and G03 (counterclockwise linear interpolation). Only one of these four modes in modal group 01 can be active at one time. G00 is the default mode. Therefore, if ServoWorks S-100M/S-120M/S-140M reads the following code:

```
X200. Z150. F200
```

it will interpret that block of code as applying to rapid positioning mode, because that is the default mode for the selection of a movement system. If your next block of code is:

```
G01 X5.0 Z157
```

that will change the selection of a movement system to linear interpolation. All position commands will be applied to linear interpolation modes until another mode from modal group 01 is programmed.

Nonmodal codes stay in effect only for the blocks that they are programmed. They are one-shot codes. Afterwards, their function is turned off automatically. All G codes in Group 00 are nonmodal.

More than one G code may be programmed in a block, such as “G00 G90,” but they must be from different groups. If more than one G code from the same group is programmed in one block of code, the one that is programmed last will be executed, and will cancel out the other G codes from that group. For example, “G01 X10.0 G00 Y20.0” will actually move both X and Y in rapid motion (G00) rather than linear interpolation (G01).

6.2 Summary of G Codes

G-CODE	GROUP	DESCRIPTION
G00	01	rapid positioning (rapid traverse)
G01	01	linear interpolation
G02	01	circular or helical interpolation (clockwise)
G03	01	circular or helical interpolation (counterclockwise)
G02.3	01	positive exponential interpolation (for S-120M and S-140M only)
G03.3	01	negative exponential interpolation (for S-120M and S-140M only)
G04	00	dwelling
G05	00	dynamic look-ahead contour control on/off
G08	00	dynamic look-ahead contour control on/off
G10	00	programmable data input
G17	02	XY plane selection
G18	02	ZX plane selection
G19	02	YZ plane selection
G20	06	inch data input
G21	06	metric data input
G28	00	automatic zero return to the reference point
G29	00	automatic return from the reference point
G30	00	automatic return to 2 nd , 3 rd and 4 th reference points
G31	00	skip cutting
G37	00	automatic tool length compensation calibration

Table 6-1: Summary of G Codes (1 of 4)

G-CODE	GROUP	DESCRIPTION
G40	07	tool radius compensation cancel
G40.1	19	normal direction control cancel
G41	07	tool radius compensation right
G41.1	19	normal direction control left
G42	07	tool radius compensation left
G42.1	19	normal direction control right
G43	08	positive tool length compensation
G43.2	08	tool center point (TCP) control
G44	08	negative tool length compensation
G49	08	tool length compensation cancel/ tool center point (TCP) control cancel
G50	11	scaling off
G51	11	scaling on
G50.1	22	mirror image off
G51.1	22	mirror image on
G52	00	local coordinate system selection
G53	00	machine coordinate system selection
G54	14	workpiece coordinate 1 selection
G54.1	14	additional workpiece coordinate selection system
G55	14	workpiece coordinate 2 selection
G56	14	workpiece coordinate 3 selection
G57	14	workpiece coordinate 4 selection
G58	14	workpiece coordinate 5 selection
G59	14	workpiece coordinate 6 selection

Table 6-2: Summary of G Codes (2 of 4)

G-CODE	GROUP	DESCRIPTION
G61	15	exact stop check mode
G64	15	continuous cutting mode
G64.1	15	continuous cutting mode with block rollover
G65	00	custom macro simple call
G68	16	coordinate system rotation
G69	16	coordinate system rotation cancel
G73	09	high-speed peck drilling cycle
G74	09	counter tapping cycle / rigid tapping
G76	09	fine boring cycle
G80	09	canned cycle cancel
G81	09	drilling cycle, spot boring
G82	09	drilling cycle (dwell)
G83	09	peck drilling cycle
G84	09	tapping cycle / rigid tapping
G85	09	boring cycle
G86	09	boring cycle (spindle stop)
G87	09	back boring cycle
G89	09	boring cycle (dwell)
G90	03	absolute programming
G91	03	incremental programming
G92	00	workpiece coordinate programming
G94	05	feed per minute mode
G95	05	feed per revolution mode

Table 6-3: Summary of G Codes (3 of 4)

G-CODE	GROUP	DESCRIPTION
G98	10	return to initial point in canned cycle
G99	10	return to R point in canned cycle
G310	31	linear interpolation feedrate include rotary axes
G311	31	linear interpolation feedrate exclude rotary axes

Table 6-4: Summary of G Codes (4 of 4)

In addition to these preset G codes, you can create customized G codes using custom G code macro calls. See *Section 6.5: Custom Macro Calls Using G Codes, M Codes, S Codes or T Codes* in the *ServoWorks CNC Macro Programming Manual*.

6.3 Detailed Explanations of G Codes

Note that the following explanations assume that the tool is moving and that the part is not moving, rather than the other way around.

6.3.1 Rapid Positioning (No Interpolation) (G00)

Description

The G00 command is used for rapid positioning of a tool. This command will move a tool to a specified position in the workpiece system. The tool will move along each axis at its rapid traverse rate (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*) for that axis until it reaches the specified position on that axis. Acceleration and deceleration are based on the “Smoothing Time” parameter (see *Chapter 7: Smoothing Parameters (Acceleration/Deceleration)* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).

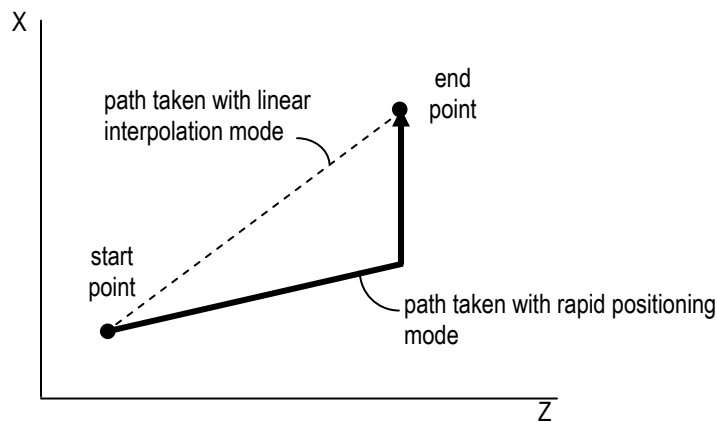


Figure 6-1: Rapid Traverse



Do not use this code for cutting, as the path from point to point is uncertain, and depends upon the rapid traverse rate for the machine tool set by the machine tool builder.

Required Format

G00 X Y Z A B C

Parameters Typically Used With G00

X – coordinate value for the X axis
Y – coordinate value for the Y axis
Z – coordinate value for the Z axis
A – coordinate value for the A axis
B – coordinate value for the B axis
C – coordinate value for the C axis

NOTE: these coordinate values could be absolute or incremental, relative to the part origin or to the current tool position, depending upon whether G90 or G91 is in effect)

Example

G00 X30.5 Y0.0 Z-5.0

ServoWorks S-100M/S-120M/S-140M Parameters Which Affect G00

The NC and Machine Parameter “G00 Perform Linear Interpolation” (see *Section 3.4.2: G00 Perform Linear Interpolation (Rapid Feed Type)* in the *Reference Manual for ServoWorks CNC Parameters and Functions*) affects how G00 codes are executed. If “G00 Perform Linear Interpolation” is enabled, then G00 will be executed with linear interpolation at the highest possible speed (constrained by the rapid feedrate of each axis). If “G00 Perform Linear Interpolation” is disabled, then G00 will be executed as normal rapid traverse, with no interpolation.

The NC and Machine Parameter “Rapid Feedrate” (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*) affects the speed at which the G00 movement is executed.

Other G-Codes Which Affect G00

The G90 and G91 codes specifying absolute and incremental programming, respectively, affect whether the coordinates you specify will be treated as absolute coordinates (relative to the part origin) or incremental coordinates (relative to the current tool position).

6.3.2 Linear Interpolation (G01)

Description

The G00.1 command is used for rapid positioning of a tool. Like G00, this command will move a tool to a specified position in the workpiece system. The tool will move along each axis at its rapid traverse rate (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*) until it reaches the specified position on that axis, with acceleration and deceleration set by the Maximum Acceleration/Deceleration rate parameter (see *Section 17.2.1: Maximum Acceleration/Deceleration Rate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*) for that axis (instead of using an acceleration/deceleration rate calculated in accordance with the “Smoothing Time” parameter).

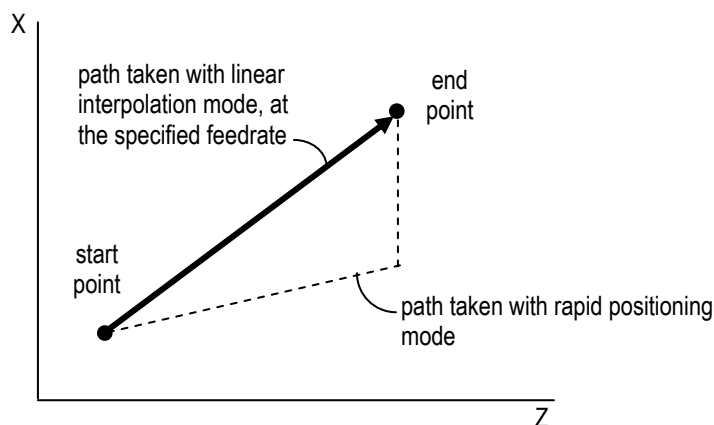


Figure 6-2: Linear Interpolation

Required Format

G01 X Y Z A B C F

Parameters Typically Used With G01

X – coordinate value for the X axis

Y – coordinate value for the Y axis

Z – coordinate value for the Z axis

A – coordinate value for the A axis

B – coordinate value for the B axis

C – coordinate value for the C axis

F – feedrate (of the movement along the linear tool path) [NOTE: the feedrates for the individual axes are calculated so as to produce the specified feedrate along the linear path.]

NOTE: these coordinate values could be absolute or incremental, relative to the part origin or to the current tool position, depending upon whether G90 or G91 is in effect)

Example

G01 X30.5 Z-5.0 F10

Other G-Codes Which Affect G01

The G90 and G91 codes specifying absolute and incremental programming, respectively, affect whether the coordinates you specify will be treated as absolute coordinates (relative to the part origin) or incremental coordinates (relative to the current tool position).

6.3.3 Circular Interpolation (G02, G03)

Description

The G02 and G03 commands are used for moving a tool along two or more axes to provide a circular path (arc). G02 specifies clockwise circular interpolation, and G03 specifies counterclockwise interpolation. This command will move a tool to a specified position in the workpiece system. The movement of the multiple axes is at a specified feedrate.

You must specify either the center of the arc in incremental values relative to the tool start point (with the correct positive or negative sign), *OR* you must also specify the arc radius.

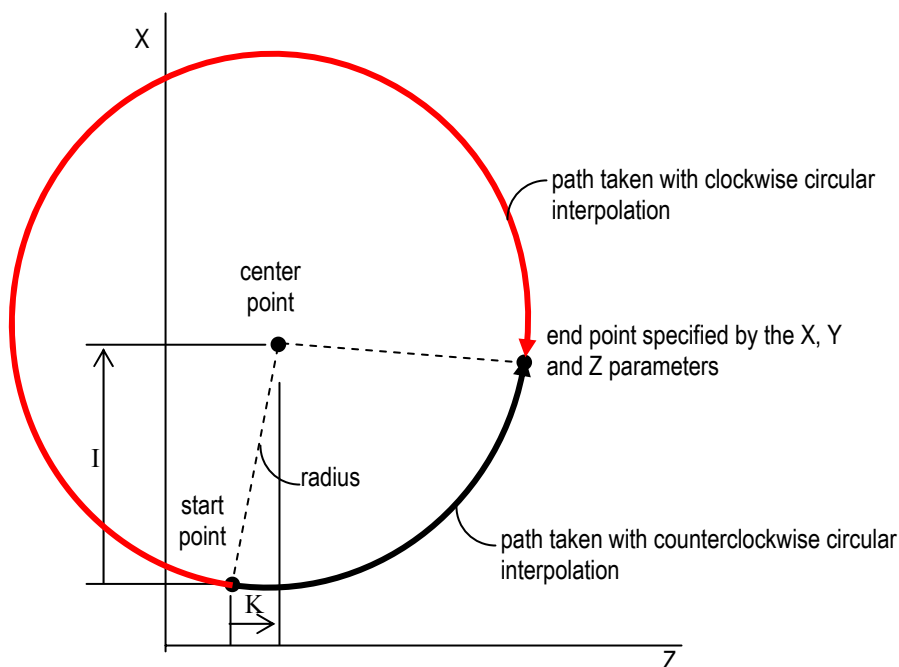


Figure 6-3: Circular Interpolation

G02: Clockwise circular interpolation

G03: Counterclockwise circular interpolation

Required Format

$$G17 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X Y \begin{Bmatrix} R \\ I J \end{Bmatrix} F Q$$

$$G18 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Z X \begin{Bmatrix} R \\ K I \end{Bmatrix} F Q$$

$$G19 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Y Z \begin{Bmatrix} R \\ J K \end{Bmatrix} F Q$$

Parameters Typically Used With G02 and G03 for Circular Interpolation

X – coordinate value for the X axis
Y – coordinate value for the Y axis
Z – coordinate value for the Z axis
I – arc center modifier for the X axis
J – arc center modifier for the Y axis
K – arc center modifier for the Z axis
R – arc radius designation
F – feedrate (of the movement along the circular tool path) [NOTE: the feedrates for the individual axes are calculated so as to produce the specified feedrate along the circular path.]
Q – the error allowance of the circular motion, in millimeters or inches. [NOTE: If the feedrate is too fast, and causes the circular motion to exceed the error allowance specified by the “Q” parameter, the feedrate will be clamped.]



NOTE: If you specify the R parameter (arc radius designation), then you cannot specify the I, J or K parameters (arc center modifiers), or you will get an error. Similarly, if you specify the I, J and/or K parameters (arc center modifiers), then you cannot specify the R parameter (arc radius designation), or you will get an error.

Examples

G18 G02 X30.5 Z-5.0 R10.5 F10 Q0.1
G18 G03 X21.0 Z-7 I17.0 K-3.0 F10 Q0.1

Other G-Codes Which Affect G02 and G03

The G90 and G91 codes specifying absolute and incremental programming, respectively, affect whether the coordinates you specify will be treated as absolute coordinates (relative to the part origin) or incremental coordinates (relative to the current tool position).

The G17, G18 and G19 codes specify which plane the movement will occur in (the XY, ZX and YZ planes, respectively). If you specify G17 for XY plane selection, and then you use a G02 or G03 command with X and Z parameters, instead of X and Y parameters, you will get an error.

G02 and G03 in the XY, ZX and YZ Planes

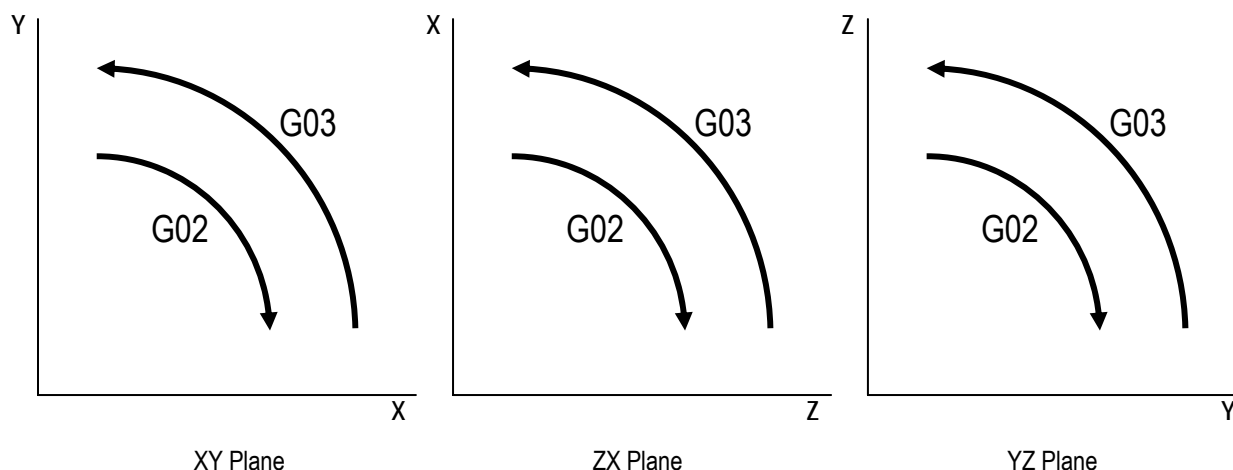


Figure 6-4: G02/G03 in the XY, ZX and YZ Planes

Limitations

- Circular interpolation can only be performed in a two-axis plane. It is not possible to apply circular interpolation to three or more axes.
- In order to avoid the G code programming error “Invalid center point for circular interpolation,” the difference between r_1 and r_2 (as shown in the following example) has to meet this condition:

$$|r_1 - r_2| < \sqrt{(m_1)^2 + (m_2)^2}$$

as demonstrated with the following example:

```
X10 Y10
G17 G3 Y30 I5 J10
%
```

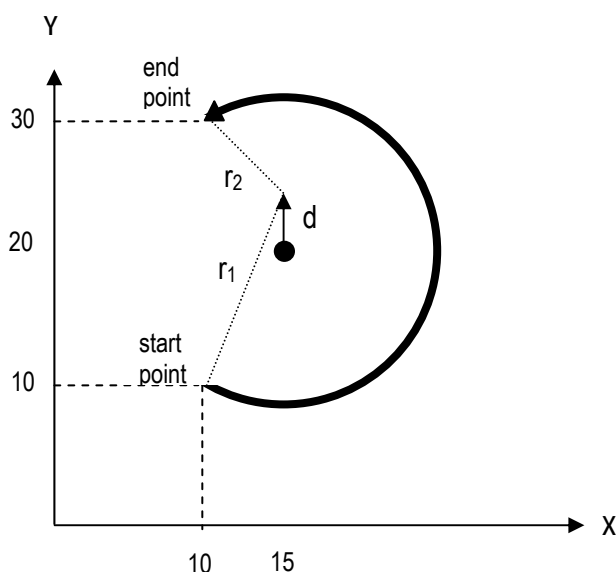


Figure 6-5: G02/G03 Limitation

where:

- r_1 = the distance from the start point to the specified center point
- r_2 = the distance from the end point to the specified center point
- m_1 = the machine unit value of the X axis (specified in the “Axis Ctrl” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see Section 3.2.2: Machine Unit (Minimum Resolution) in the Reference Manual for ServoWorks CNC Parameters and Functions)
- m_2 = the machine unit value of the Y axis (specified in the “Axis Ctrl” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see Section 3.2.2: Machine Unit (Minimum Resolution) in the Reference Manual for ServoWorks CNC Parameters and Functions)
- d = the difference between the true center point and the specified center point

Variations

- 1) To program a full circle, omit coordinates for the end point (or specify an end point which is the same as the start point), and specify the center points of the circle using the I, J and K arc center modifiers. The end point will then be considered the same as the start point, and a 360° arc (a circle) will be specified.



CAUTION

If you specify an end point that is the same as the start point, and you specify an arc radius with the R parameter, you will get an error.

- 2) The arc radius may be specified as either positive or negative. A positive arc radius will produce an arc of less than or equal to 180°, while a negative arc radius will produce an arc of equal to or greater than 180°, as shown in the following figure:

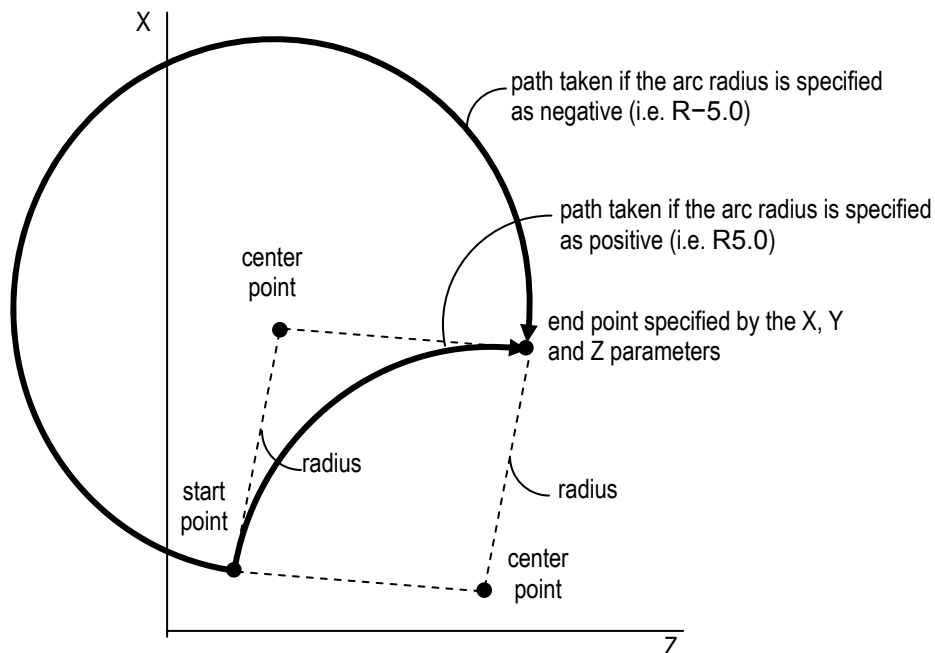


Figure 6-6: Positive and Negative Arc Radii for Circular Interpolation

- 3) If you specify the arc radius using the R parameter, the end point must be along the arc specified by that radius.



CAUTION

If you specify the arc radius with the R parameter, and the end point is not along the arc specified by that radius, you will get an error. This only occurs when the distance between the end point and the start point is greater than twice the specified radius. In this case, no center point can be calculated from which to base the arc, because no arc with the radius specified could be wide enough to bridge the two points.

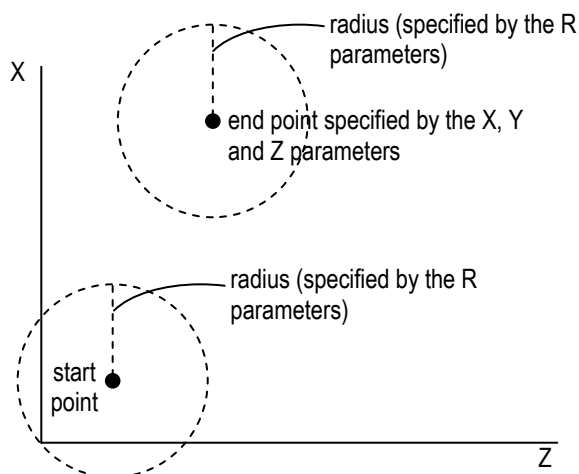


Figure 6-7: Unacceptable Parameters for Circular Interpolation

- 4) If you specify the center points of the circle using the I, J and K parameters (arc center modifiers), and the end point specified by the X, Y and Z parameters is not on the arc, an arc movement will still be performed, followed by a linear movement to reach the end point.

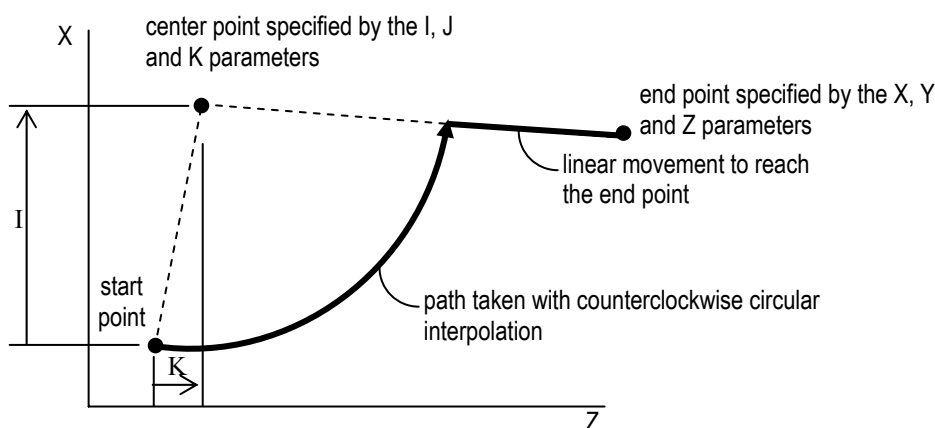


Figure 6-8: Arc Movement Produced by Incorrect Parameters for Circular Interpolation

Circle Error Allowance Parameter

Overview

The ServoWorks S-100M series (S-100M, S-120M and S-140M applications) allow you to specify a “circle error allowance” parameter in Configuration Mode. This “circle error allowance” parameter specifies the maximum error of the end position (when the circular interpolation end point is not on the circle) that is tolerated before an “Invalid center point for circular interpolation” error message is shown.

To set this parameter, go into Configuration Mode, click the “NC” button, and you will see the “Circle Error Allowance” parameter in the “NC Programming Options” frame.

Difference Between Circle Error Allowance and Q

The Q parameter specified for G02/G03 commands is called the “error allowance of the circular motion.” Despite the similarity of the names, the “circle error allowance” is different than the Q parameter specified with the G02/G03 command:

- 1) The Q parameter is a part programming parameter used in a part program with the G02/G03 command, and can be set to a different value with each G02/G03 command; the “circle error allowance” is a program setting within the GUI application, in Configuration Mode, that would apply to the execution of an entire part program.
- 2) The Q parameter addresses an error allowance caused by a feedrate that is too high, in which the position feedback exceeds some distance of the programmed path; the “circle error allowance” parameter addresses an error created by a problem within the G code programming, where the R or I/J/K parameters are specified improperly (i.e. G02 X100 I49).

Description

The “circle error allowance” parameter allows the user to specify the maximum distance between the specified endpoint and the arc drawn by the circular interpolation command. Any circular interpolation commands for which the maximum distance between the endpoint and the arc exceeds the circle error allowance results in an invalid endpoint error:

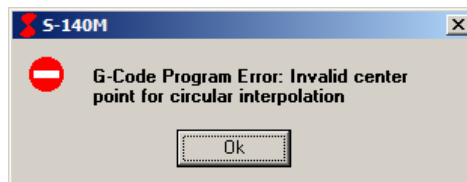


Figure 6-9: Error – Invalid Center Point for Circular Interpolation

The distance between the endpoint and the arc is calculated differently depending upon whether the radius is specified using R or whether the circle center point is specified using I, J, and K.

If the radius is specified using R:

- 1) If R is greater than or equal to half the distance between the starting point and the end point, then there exists an arc in which both the starting point and the endpoint are on the arc, and the arc will be traced regardless of the circle error allowance.
- 2) If R is less than half the distance between the starting point and the end point, then:
 - a. If the difference between R and half the distance between the starting point and the end point is less than the circle error allowance, an arc will be drawn.
 - b. If the difference between R and half the distance between the starting point and the end point is greater than the circle error allowance, an error will be displayed.

If the circle center point is specified using I, J, and K, the position and radius of the circle is fixed with respect to the starting point:

- 1) If the difference between the distance from the starting point to the center point, and the distance from the end point to the center point, is less than the circle error allowance, then an arc will be drawn.
- 2) If the difference between the distance from the starting point to the center point, and the distance from the end point to the center point, is greater than the circle error allowance, then an error will be displayed.

Shape of the Arc

In either of the above cases, if the circle error allowance is set high enough to accommodate a difference between the specified arc and end point, the center point and radius of the arc will be modified so that the arc will pass through the starting point and the end point. The end point is unmodified so that the remainder of the program can run normally.

The new radius is set to be the average of the distance between the starting point of the arc and the center point and the distance between the end point of the arc and the center point. If the radius is specified using R, the new radius is set to be half the distance from the starting point to the end point. The new center point can then be fixed to a position for which the distance between the starting point and the new center point and the distance between the end point and the new center point equals the new radius. This means that if the radius is specified using R, then the arc drawn will necessarily be a half-circle.

Example #1

```
G91 G00 X0. Y0.  
G02 X100. R49.
```

If the circle error allowance is greater than the difference between R (49 mm) and half the average distance between the starting point and the end point (50 mm), or 1 mm, then the arc is drawn. The radius of the arc is 50 mm, and the center point of the arc is at (50.0):

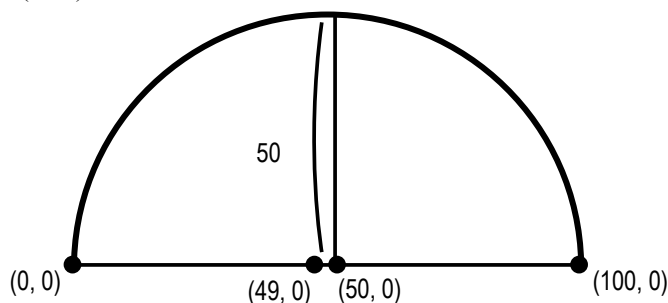


Figure 6-10: Circle Error Allowance Example #1

If the circle error allowance is less than 1 mm, an error will be displayed, and program execution will stop.

Example #2

```
G91 G00 X0. Y0.  
G02 X100. I49.
```

In this case, if the circle error allowance is greater than the difference between the distance between the starting point and the center point, and the distance between the end point and the center point, or 2 mm, then the arc will be drawn. The shape of the arc will be the same as the arc in Example #1.

If the circle error allowance is less than 2 mm, an error will be displayed, and program execution will stop.

Example #3

G91 G00 X0. Y0.
G02 X100. Y0. J-100.

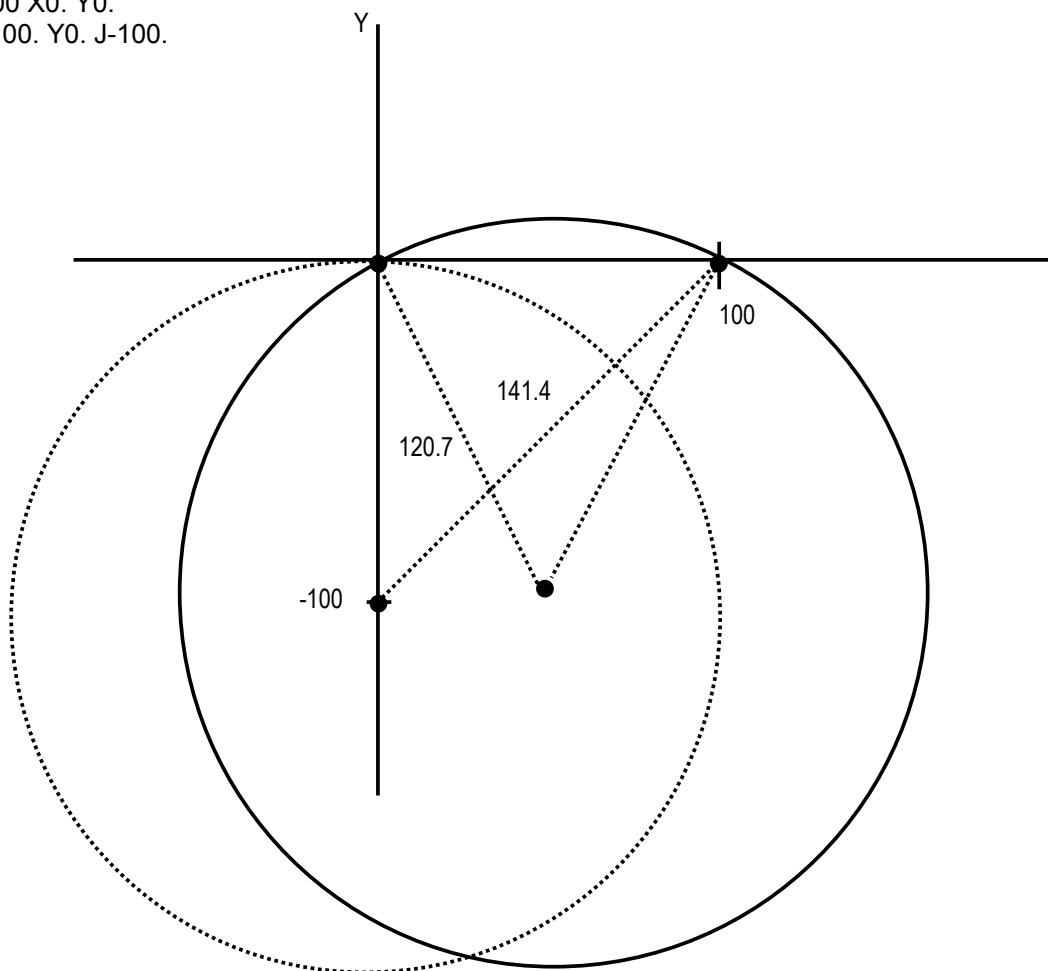


Figure 6-11: Circle Error Allowance Example #3 (1 of 2)

In this more complex example, an arc is drawn with the starting point at (0, 0), the center point at (0, -100), and the end point at (100, 0). This means that the distance from the starting point to the center point is 100, and the distance from the end point to the center point is $100\sqrt{2} = 141.4$. Thus, the difference between the distances to the center point is $141.4 - 100 = 41.4$, and any circle error allowance values below 41.4 will result in an error. Otherwise, the average distance is obtained $((141.4 + 100)/2 = 120.7)$ and a circle with radius equal to the average distance is used to draw the arc. In the figure, the dotted circle represents the original circle and the bold circle represents the new circle with a slightly larger radius used to obtain the arc. The actual arc drawn is shown in the following screenshot:

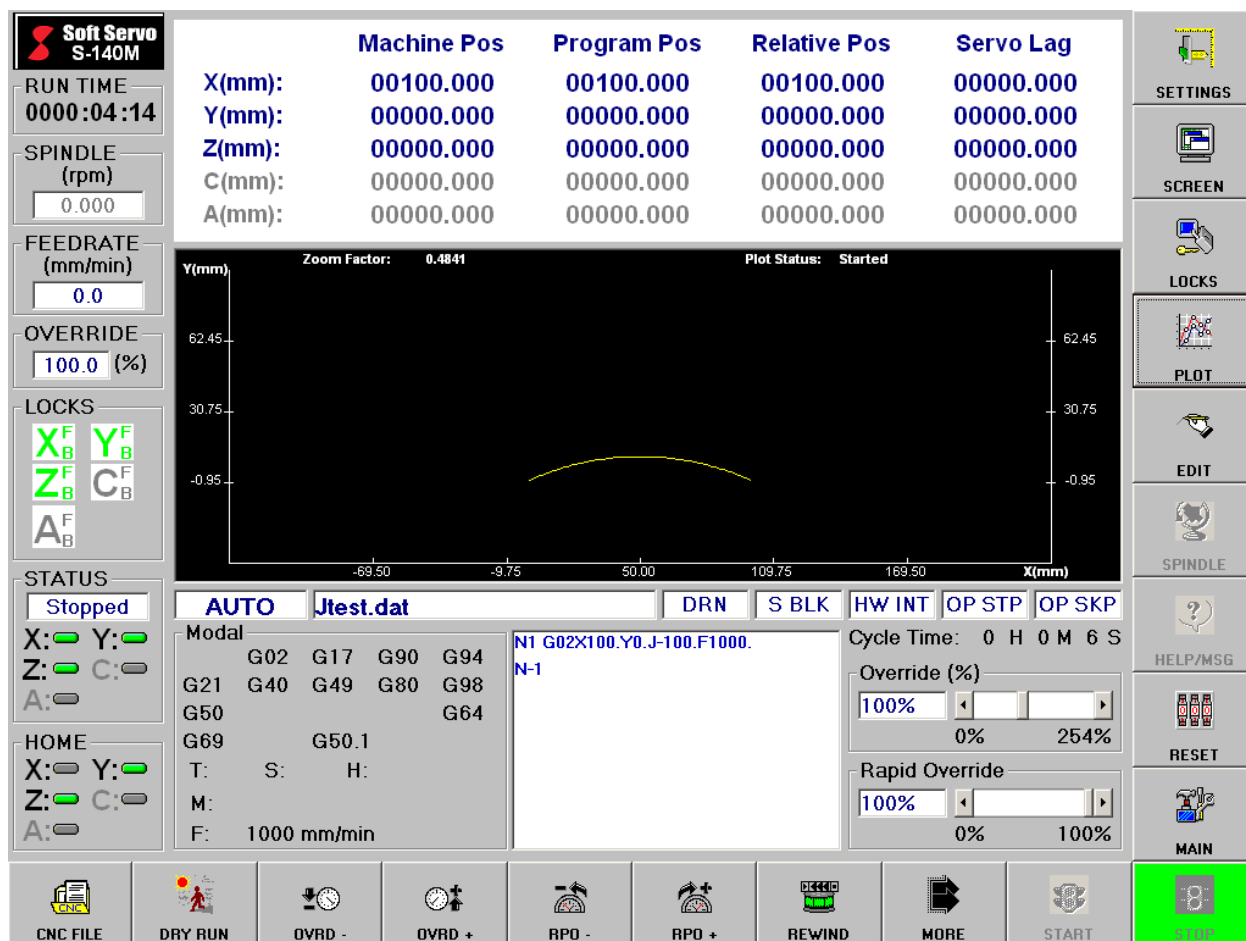


Figure 6-12: Circle Error Allowance Example #3 (2 of 2)

6.3.4 Helical Interpolation (G02, G03)

Description

The G02 and G03 commands can be used for helical interpolation, as well as for circular interpolation. With helical interpolation, the tool moves in a spiral: circular interpolation of two axes is synchronized with circular interpolation of two other optional axes.

You specify a helical interpolation command in the same way as a circular interpolation command, except for two differences:

- 1) The addition of coordinate values for one or two additional axes.
- 2) The feedrate that you specify with the F parameter is not the feedrate along the arc of the helical spiral; it is the tangential feedrate along the projected arc of the circular interpolation path. Therefore, the feedrate along the arc of the helical spiral is as follows:

$$F \times \frac{\text{length of the arc along the helical path}}{\text{length of the arc along the tangential path}}$$

See Section 6.3.3: *Circular Interpolation (G02, G03)* for other G codes that affect G02 and G03, and for variations and for warnings.

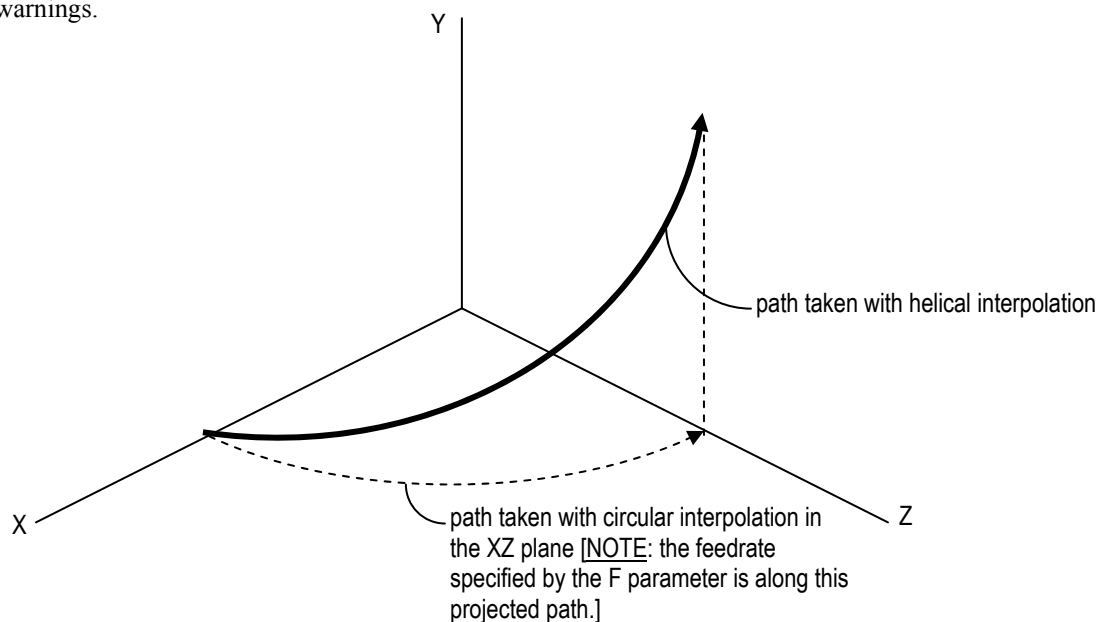


Figure 6-13: Helical Interpolation

Required Format

G17 { G02
G03 } X Y { R
I J } Z/A F

G18 { G02
G03 } Z X { R
K I } Y/A F

G19 { G02
G03 } Y Z { R
J K } X/A F

Parameters Typically Used With G02 and G03 for Helical Interpolation

X – coordinate value for the X axis
Y – coordinate value for the Y axis
Z – coordinate value for the Z axis
A – coordinate value for the A axis
I – arc center modifier for the X axis
J – arc center modifier for the Y axis
K – arc center modifier for the Z axis
R – arc radius designation

NOTE: these coordinate values could be absolute or incremental, relative to the part origin or to the current tool position, depending upon whether G90 or G91 is in effect)

F – feedrate (of the movement along the *projected* circular tool path, not along the helical tool path) **[NOTE:** the feedrates for the individual axes are calculated so as to produce the specified feedrate along the projected circular path.]

NOTE: If you specify the R parameter (arc radius designation), then you cannot specify the I, J or K parameters (arc center modifiers), or you will get an error. Similarly, if you specify the I, J and/or K parameters (arc center modifiers), then you cannot specify the R parameter (arc radius designation), or you will get an error.

Examples

G18 G02 X30.5 Z-5.0 R10.5 Y20.0 F10
G19 G03 X21.0 Y-7 J17.0 K-3.0 Z20.0 A4.0 F10

6.3.5 Positive / Negative Exponential Interpolation (G02.3, G03.3)

Description

The G02.3 and G03.3 commands are used for exponential interpolation, in which the rotation of a workpiece changes exponentially with respect to movement about the rotary axis (Axis A). G02.3 specifies positive exponential interpolation, and G03.3 specifies negative exponential interpolation. This command incorporates linear interpolation with respect to the linear axis (Axis X) to enable tapered groove machining with a constant helix angle.

G02.3 and G03.3 are available only for S-120M and S-140M.

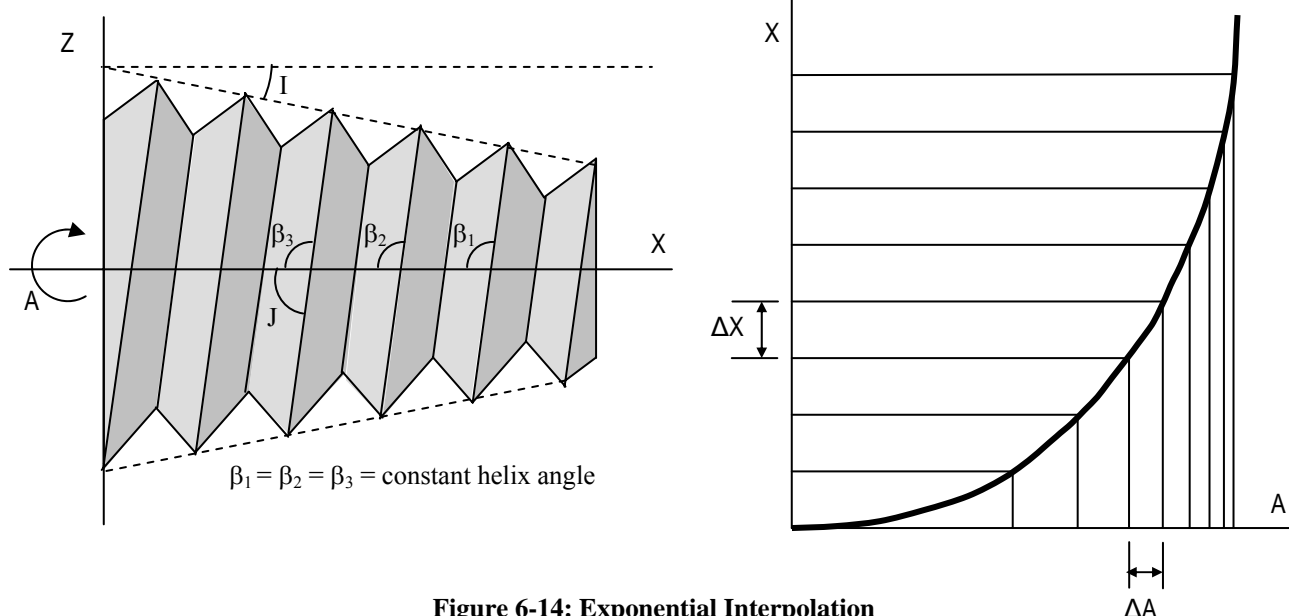


Figure 6-14: Exponential Interpolation

Required Format

$$\left\{ \begin{array}{l} G02.3 \\ G03.3 \end{array} \right\} X Y Z I J R F Q$$

Parameters Typically Used With G02.3 and G03.3

X – end point coordinate value for the X axis
Y – end point coordinate value for the Y axis
Z – end point coordinate value for the Z axis
I – taper angle (range: 0 - 89°)
J – helix angle (range: 0 - 89°)
R – the constant for exponential interpolation (see equations below)
F – initial feedrate (includes the feedrate on the rotary axis)
Q – end point feedrate (includes the feedrate on the rotary axis)

NOTE: these coordinate values could be absolute or incremental, relative to the part origin or to the current tool position, depending upon whether G90 or G91 is in effect)

NOTE: The feedrate varies by interpolation between F and Q along the linear axis travel.

Equations Related to G02.3 and G03.3

travel along the linear axis:
$$X(\theta) = R \left(e^{\frac{\theta}{K}} - 1 \right) \left(\frac{1}{\tan(I)} \right)$$

travel along the rotary axis:
$$A(\theta) = [-1]^{\omega} \left[360 \right] \left(\frac{\theta}{2\pi} \right)$$

where
$$K = \frac{\tan(J)}{\tan(I)}$$

$\omega = 0$ for positive rotation (G02.3) and $\omega = 1$ for negative rotation (G03.3),
and θ is the angle of rotation in radians

When the tool moves from X_1 to X_2 along the linear X axis, the angle moved about the rotation axis is calculated with the following equation:

$$\Delta\theta = K \left[\ln \left(\frac{(X_2)(\tan(I))}{R} + 1 \right) - \ln \left(\frac{(X_1)(\tan(I))}{R} + 1 \right) \right]$$

Example

G02.3 X30.5 Y2.0 Z-5.0 I10.0 J30.0 R1.0 F20 Q10

Other G-Codes Which Affect G02.3 and G03.3

The G90 and G91 codes specifying absolute and incremental programming, respectively, affect whether the coordinates you specify will be treated as absolute coordinates (relative to the part origin) or incremental coordinates (relative to the current tool position).

6.3.6 Dwell (G04)

Description

This command specifies a delay in program execution. Shifting to the next block is delayed for the number of minutes specified by the parameters.

Required Format

$$G04 \left\{ \begin{array}{c} P \\ X \end{array} \right\}$$

Possible Parameters That Can Be Used With G04

P – The time specified without a decimal point (in milliseconds)

X – The time specified with a decimal point (in seconds)

Examples

G04 P5000 (to dwell five seconds)

G04 X3.0 (to dwell three seconds)

6.3.7 Three-Dimensional Dynamic Look-Ahead Contour Control On / Off (G05, G08)

Description

This three-dimensional dynamic look-ahead contour control (3D-DLACC) function is designed for high-speed, precision machining. The essence of this look-ahead function is to generate the position command output pulse in a more intelligent manner than the conventional method of generating the position command, so that the position command follows the programmed path much more accurately.

The problem with the conventional method of generating the position command is that the output pulse generated by the block commands has to go through a smoothing filter to avoid jerky motion. The drawback of the smoothing filter is that the movement of position command is delayed along each axis without coordination. In other words, the integrity of interpolation is broken, and the accuracy of tool-path trajectory is lost.

With the Look-Ahead technology, the CNC controls the acceleration and deceleration of the movement along the commanded trajectory. In other words, the acceleration and deceleration of each axis is coordinated with interpolation. Therefore, the need for a smoothing filter that deteriorates the accuracy of the position command is eliminated, so the final result is an accurately generated position command that faithfully follows the commanded trajectory without jerky motion, and optimized acceleration and deceleration of each axis to achieve the maximum cutting feedrate.

Possible Parameters That Can Be Used With G05, G08

P – Specifies whether to turn 3D-DLACC on or off.

Required Format

There are only four possibilities for the effective usage of G05/G08:

G05 P10000	to turn ON dynamic look-ahead contour control
G05 P0	to turn OFF dynamic look-ahead contour control
G08 P1	to turn ON dynamic look-ahead contour control
G08 P0	to turn OFF dynamic look-ahead contour control

(G05 P10000 is equivalent to G08 P1, and G05 P0 is equivalent to G08 P0)

Example

```
G90G00X0Y0 (X and Y axes return to zero)
G08 P1 (3D-DLACC ON)
G91G01F20000.0 (Incremental linear feed)
X50.0Y100.0
X50.0Y-100.0
X50.0Y100.0
X50.0Y-100.0
G08 P0 (3D-DLACC OFF)
G04 X1.0 (Dwell)
G90G00X0.0Y0.0 (X and Y axes return to zero)
M02 (Program end)
```


This program starts the motion from X-Y plane program zero point, turns on DLACC, performs “saw teeth” shape movements, then turns off 3D-DLACC, and finally returns to program zero point after a dwell.

Notes

All M/S/T codes are supported, and can be commanded when 3D-DLACC is on. This includes M98/M99 subprogram calls.

Limitations

- Any usage of G05/G08 other than the four possibilities listed above will be ignored.
- Only the following supported G codes can be commanded when 3D-DLACC is on:

G00	Rapid traverse
G01	Linear interpolation
G02, G03	CW/CCW circular or helical interpolation
G04	Dwell
G17-G19	XY/ZX/YZ plane selection
G20, G21	Inch/metric data input
G28, G29	Automatic return to/from reference point
G30	Automatic return to 2 nd , 3 rd , 4 th reference points
G50, G51	Scaling off/on
G50.1, G51.1	Mirror image off/on
G52	Local coordinate system selection
G53	Machine coordinate system selection
G54-G59	Workpiece coordinate system 1-6 selection
G54.1	Additional workpiece coordinate system selection
G61	Exact stop check mode
G64	Continuous cutting mode
G64.1	Continuous cutting mode with block rollover
G65	Custom macro simple call
G90	Absolute command programming
G91	Incremental command programming
G92	Workpiece coordinate programming
G310, G311	Linear interpolation feedrate include/exclude rotary axes



NO G CODES OTHER THAN THOSE LISTED ABOVE SHOULD BE COMMANDED WITH 3D-DLACC. If an incompatible G code (a G code not listed above) is used with 3D-DLACC, the motion program will stop at the first instance of an incompatible G code, and you will see an error message.

Reference

It is strongly recommended that you review *Chapter 17: Dynamic Look-Ahead Contour Control Parameters and Usage* in the *Reference Manual for ServoWorks CNC Parameters and Functions* for additional information.

6.3.8 Programmable Data Input (G10)

Required Format

G10 L__ P__ R__

OR

G10 L__ X__ Y__ Z__

Possible Parameters That Can Be Used With G10

L – specifies the data category
P – specifies the data index
R – specifies the data value
X – specifies the data value for the X axis
Y – specifies the data value for the Y axis
Z – specifies the data value for the Z axis

Data Input Categories

The currently supported data input categories (specified by the L parameter) are listed as follows:

- **L10 – Tool Length Compensation**

- Description: Specifies the tool length compensation, specifically, the tool length offset Type A.
- Parameters:
 - P parameter: 1 to 8 (specifies axis number)
 - R parameter: Z axis tool length compensation
- Reference

It is strongly recommended that you review *Section 6.3.17: Positive/Negative Tool Length Compensation/Compensation Cancel (G43, G44, G49)* for additional information.

- **L11 – Tool Radius Compensation**

- Description: Specifies the tool radius compensation, specifically, the tool length offset Type A.
- Parameters:
 - P parameter: 1 to 8 (specifies axis number)
 - R parameter: total value for tool radius compensation (including geometry and wear tool offset values)
- Reference

It is strongly recommended that you review *Section 6.3.15: Tool Radius Compensations (G40, G41 and G42)* for additional information.

- **L20 – Axis Type Settings**

- Description: Specifies the axis type of up to 8 axes; supports both types of G10 formats.

- Parameters:

P parameter: 1 to 8 (specifies axis number)

R parameter: Axis type

Valid Values for R Parameter:

1	Normal
2	Rotary
3	Rotary Single Turn (Rotary ST)
10	PLC Axis

- Examples

G10 L20 P2 R10 // Changes axis 2 (Y) into a PLC axis

G10 L20 X2 Y10 // Changes axis X into a rotary axis and axis Y into a PLC axis

- Reference

It is strongly recommended that you review *Section 2.2.2: Axis Type* in the *Reference Manual for ServoWorks CNC Parameters and Functions* for additional information.

- **L30 – In Position Width**

- Description: Specifies the in position width of a particular axis; supports both types of G10 formats.

- Parameters:

P parameter: 1 to 8 (specifies axis number)

R parameter: In position width

- Examples

G10 L30 P2 R0.1 // Changes the in position width of axis 2 (Y) to 0.1

G10 L30 X0.2 Y0.1 // Changes the in position width of axis X to 0.2 and the in position width of axis Y to 0.1

- Reference

It is strongly recommended that you review *Section 3.2.3: In Position Width* in the *Reference Manual for ServoWorks CNC Parameters and Functions* for additional information.

- **L106 – Smoothing (Acc/Dec) Mode Settings For Rapid Motion (G00)**

- Description: Specifies the smoothing (acceleration/deceleration) mode setting of all axes in rapid motion (G00)

- Parameters:

P parameter: Unused

R parameter: Smoothing (acceleration/deceleration) mode

Valid Values for R Parameter:

NO_SMOOTHING	0	// No velocity smoothing
SMOOTH_LINEAR	1	// Trapezoidal velocity profile
SMOOTH_BELLSHAPE	2	// Bell-Shaped velocity profile
SMOOTH_EXPONENTIAL	3	// Exponential velocity profile



It is invalid to set smoothing mode before acceleration/deceleration motion has finished. We recommend using G04 exact stop before every G10 L106 command.

- Reference

It is strongly recommended that you review *Chapter 7: Smoothing Parameters (Acceleration/Deceleration)* in the *Reference Manual for ServoWorks CNC Parameters and Functions* for additional information.

- **L107 – Smoothing (Acc/Dec) Mode Settings For Cutting Motion (G01)**

- Description: Specifies the smoothing (acceleration/deceleration) mode setting of all axes in cutting motion (G01)

- Parameters:

P parameter: Unused

R parameter: Smoothing (acceleration/deceleration) mode

Valid Values for R Parameter:

NO_SMOOTHING	0	// No velocity smoothing
SMOOTH_LINEAR	1	// Trapezoidal velocity profile
SMOOTH_BELLSHAPE	2	// Bell-Shaped velocity profile
SMOOTH_EXPONENTIAL	3	// Exponential velocity profile



It is invalid to set smoothing mode before acceleration/deceleration motion has finished. We recommend using G04 exact stop before every G10 L107 command.

- Reference

It is strongly recommended that you review *Chapter 7: Smoothing Parameters (Acceleration/Deceleration)* for additional information.

- **L108 – Smoothing (Acc/Dec) Time Settings**

- Description: Specifies the smoothing (acceleration/deceleration) time setting for each axis; supports both types of G10 formats.

- Unit: milliseconds

- Parameters:

P parameter: Axis number (Range: 1 ~ 16)

R parameter: Smoothing (acceleration/deceleration) time value.

Valid Values for R Parameter: vary depending on servo loop update rate, as follows:

1 KHz or lower: 0 ~ 5000 (ms)

2 KHz: 0 ~ 2500 (ms)

4 KHz: 0 ~ 1250 (ms)

- Examples:

G04 (Exact stop)

G10 L108 X50 Y50 (Sets X and Y axes smoothing time to 50 ms)

G10 L108 P1 R1000 (Sets axis X smoothing time to 1000 ms)



It is invalid to set smoothing time before acceleration/deceleration motion has finished. We recommend using G04 exact stop before every G10 L108 command.

- **L10909 – Position Loop Gain Settings**

- Description: Specifies the position loop gain setting of each axis; supports both types of G10 formats.

- Unit: Hz

- Parameters:

P parameter: Axis number (Range: 1 ~ 16)

R parameter: Position loop gain value (Range: 0 ~ 999999.9 Hz)

- Notes: Position loop gain can be set at any time in the program.

- Examples:

G10 L10909 X10 Y5 (Sets X axis position loop gain to 10 Hz, and Y axis to 5 Hz)

G10 L10909 P1 R15 (Sets X axis position loop gain to 15 Hz)



A high position loop gain value could lead to system vibration and instability, which might cause damage to the machine. Please be very careful in setting values. Refer to *Section 10.2: Position Loop Gain* in the *Reference Manual for ServoWorks CNC Parameters and Functions*.

Torque Control with G10 (MECHATROLINK II/III only)

G10 can also be used for torque control of any Sigma III or Sigma V servo axis in a MECHATROLINK II/III servo network. Torque control allows the user to send torque commands to set the torque of individual axes to a percentage of the axes' rated torque. To start torque control, use G10 to set the axis type of an axis to a special "torque axis." Subsequent G01 commands to that axis will be interpreted as torque commands.

Example:

```
N10 G00 X10.0 Y10.0 Z10.0
N20 G10 L20 P3 R12
N30 G01 X50.0 Y100.0 Z20.0 F3000.0
N40 Z0.0
N50 G10 L20 P1 R12
N60 X50.0
N70 G04 X3.0
N80 G10 L20 P1 R1
N90 G10 L20 P3 R1
N100 X200.0 Y200.0 Z200.0
```

N20: Change Z axis to torque axis.

N30: Position command sent to X and Y; torque command sent to Z. The F parameter only applies to X and Y. The Z axis will output 20% of its rated torque until X and Y reach their target positions.

N40: Set Z axis torque to 0.

N50: Change X axis to torque axis.

N60, N70: Set X axis to apply 50% torque for 3 seconds.

N80: Revert X axis to normal axis.

N90: Revert Y axis to normal axis.

N100: Move X, Y, and Z to their commanded positions.

Notes:

-This feature is currently only available for Sigma III and Sigma V servo drives running on the MECHATROLINK II/III platform.

- If the block before G10 (change to torque axis) does not perform an in-position check, the axis is switched to a torque axis at the point when the block before G10 starts decelerating. This means that a torque command will be applied before the block before G10 reaches its target position. If you need torque to be applied after reaching the target position, use in-position check, Dwell, etc.

- Torque commands have units of %. This value is a percentage of the rated torque that you want the servo drive to apply.

-If you are using Sigma-V, the rated torque is automatically obtained without any user input, but if you are using Sigma-III, the rated torque must be manually entered into the .ini file by the user. For Sigma-III drives (drives set to ServoVer=3 in the .ini file), add the following to the description of the drives in the .ini file: TorqueSpecMax=x, where x is the rated torque of your servo packs.

6.3.9 XY/ZX/YZ Plane Selection (G17, G18 and G19)

Description

The G17, G18 and G19 commands are used to select the plane in which specified movement is to occur.

G17: Selects the XY plane

G18: Selects the ZX plane

G19: Selects the YZ plane

Possible Parameters That Can Be Used With G17, G18 and G19

None.

Example

G17 G02 X5.2 Y-7.0 R3.3 F500

Notes

In a system with more than four axes, the “X” refers to an X axis or some other axis parallel to the X axis. Similarly, “Y” and “Z” refer to either the Y and Z axes, or axes parallel to the Y and Z axes.

Other G-Codes Which Affect G17, G18 and G19

The G17, G18 and G19 commands are used in conjunction with G02 and G03, and G02.3 and G03.3.

Default

G17 is the default mode when none of the G17, G18 or G19 codes has been programmed.

6.3.10 Inch / Metric Data Input (G20, G21)

Description

The G20 and G21 commands are used to specify which measurement system you will use to specify coordinates, distances, radii, etc.

G20: Input distances will be in inches

G21: Input distances will be in millimeters

The G20 and G21 commands are typically used at the beginning of a program. All subsequent commands including parameters specifying distance are affected by the G20 and G21 commands.

Possible Parameters That Can Be Used With G20 and G21

None.

Limitations

The G20 and G21 commands only apply to linear axes – axes whose movement is linear, and for which distance is specified in inches or millimeters. The selection of an inch or a metric measurement system has no effect on rotation axes, whose values are specified in degrees.

Default

G21 is the default mode when neither G20 nor G21 has been programmed.

6.3.11 Automatic Zero Return to/from Reference Points (G28, G29)

Description

The G28 and G29 commands are used to return to and to return from the reference point. The reference points are set in ServoWorks S-100M/S-120M/S-140M.

G28: Return to the reference position

G29: Return from the reference position

The G28 command is used to return the tool from the start point (the current tool position) to the reference point, via an intermediate point, the coordinates of which are specified by the parameters.

The G29 command is used to return the tool from the reference point (the current tool position) to a destination point specified by the parameters, via an intermediate point specified by the G28 command.

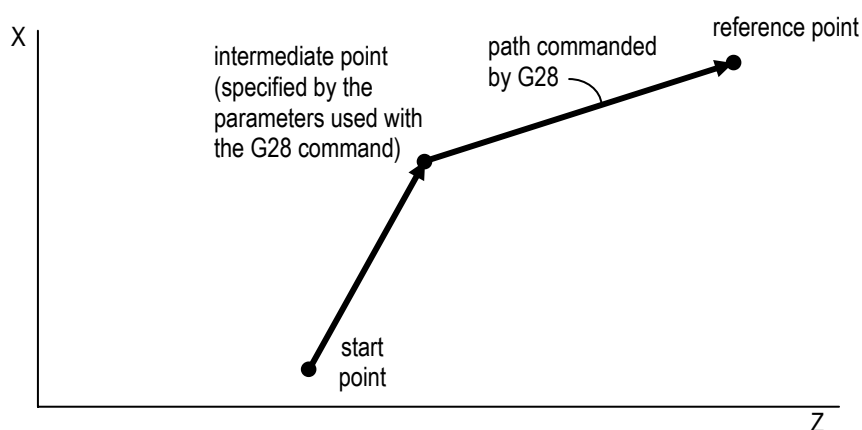


Figure 6-15: Automatic Return to Reference Points (G28)

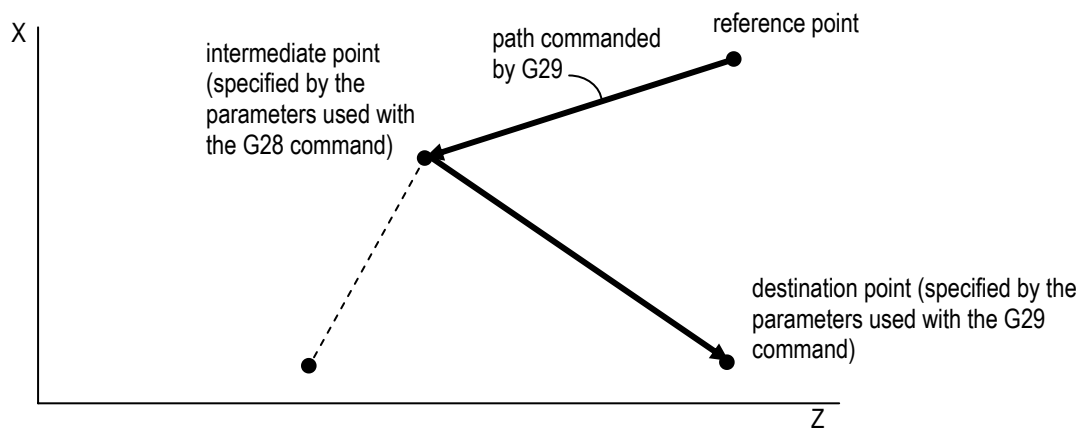


Figure 6-16: Automatic Return from Reference Points (G29)

Required Format

$\left\{ \begin{matrix} \text{G28} \\ \text{G29} \end{matrix} \right\} X Y Z A B C$

Possible Parameters That Can Be Used With G28 and G29

X – coordinate value for the X axis
Y – coordinate value for the Y axis
Z – coordinate value for the Z axis
A – coordinate value for the A axis
B – coordinate value for the B axis
C – coordinate value for the C axis

NOTE: these coordinate values could be absolute or incremental, relative to the part origin or to the current tool position, depending upon whether G90 or G91 is in effect)

Example

G28 X0.5 Z7.8
G29 X7.5 Z2.1

Other G-Codes Which Affect G28 and G29

The G90 and G91 codes specifying absolute and incremental programming, respectively, affect whether the coordinates you specify will be treated as absolute coordinates (relative to the part origin) or incremental coordinates (relative to the current tool position).

6.3.12 Automatic Zero Return To Additional Reference Points (G30)

Description

The G30 command is used to return the tool from the start point (the current tool position) to an additional, specified reference point, via an intermediate point, the coordinates of which are specified by the parameters. The additional reference points are set in ServoWorks S-100M/S-120M/S-140M (see *Section 9.11: Reference Position #2*, *Section 9.12: Reference Position #3* and *Section 9.13: Reference Position #4* in the *Reference Manual for ServoWorks CNC Parameters and Functions* for additional information).

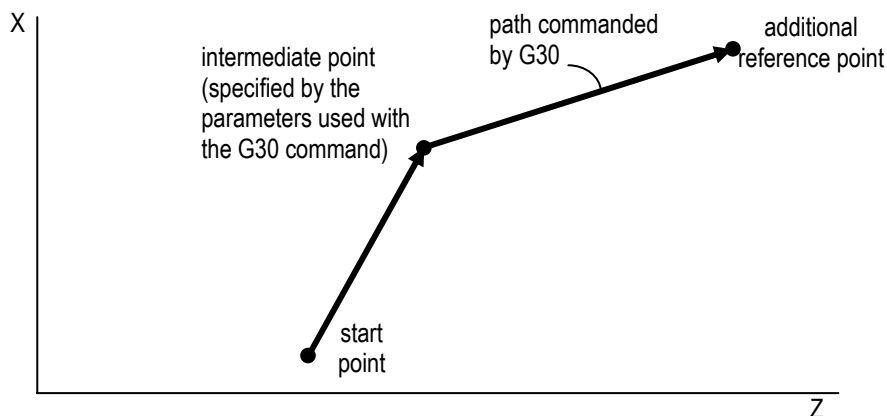


Figure 6-17: Automatic Return to Additional Reference Points (G30)

Required Format

G30 X Y Z A B C P

Possible Parameters That Can Be Used With G30

X – coordinate value for the X axis

Y – coordinate value for the Y axis

Z – coordinate value for the Z axis

A – coordinate value for the A axis

B – coordinate value for the B axis

C – coordinate value for the C axis

P – specifies which additional reference point (P2, P3, etc.) – if not specified, P2 is the default

NOTE: these coordinate values could be absolute or incremental, relative to the part origin or to the current tool position, depending upon whether G90 or G91 is in effect)

Example

G30 P3 X0.5 Z7.8

G29 X7.5 Z2.1

Other G-Codes Which Affect G30

The G90 and G91 codes specifying absolute and incremental programming, respectively, affect whether the coordinates you specify will be treated as absolute coordinates (relative to the part origin) or incremental coordinates (relative to the current tool position).

6.3.13 Skip Cutting (G31)

Description

This one-shot command for linear interpolation allows you to specify commands in a part program that should be cancelled (not completed) if the G6.6 PLC Skip Signal switch changes (either rising or falling edge) while this command is being executed. Once the G6.6 PLC Skip Signal switch changes, command execution terminates and the next block of the part program starts to be executed.

Like the G01 command, the G31 command is used for moving a tool along two or more axes to provide a linear (straight line) path. This command will move a tool to a specified position in the workpiece system (as long as the G6.6 PLC Skip Signal switch doesn't change). The movement of the multiple axes is at a specified feedrate, which is the feedrate in the direction of movement, not the feedrate of an individual axis. The CNC calculates the linear trajectory, and synchronizes multiple motors to maintain a straight path between two points.

Required Format

G31 X Y Z A B C F

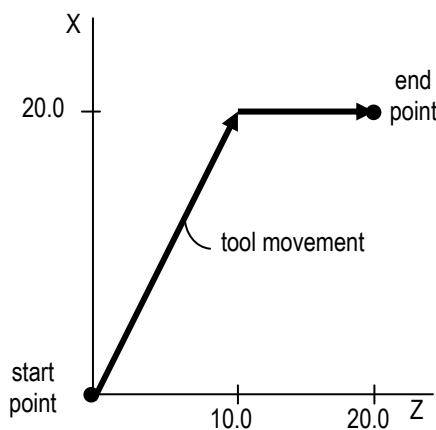
Parameters Typically Used With G31

X – coordinate value for the X axis
Y – coordinate value for the Y axis
Z – coordinate value for the Z axis
A – coordinate value for the A axis
B – coordinate value for the B axis
C – coordinate value for the C axis

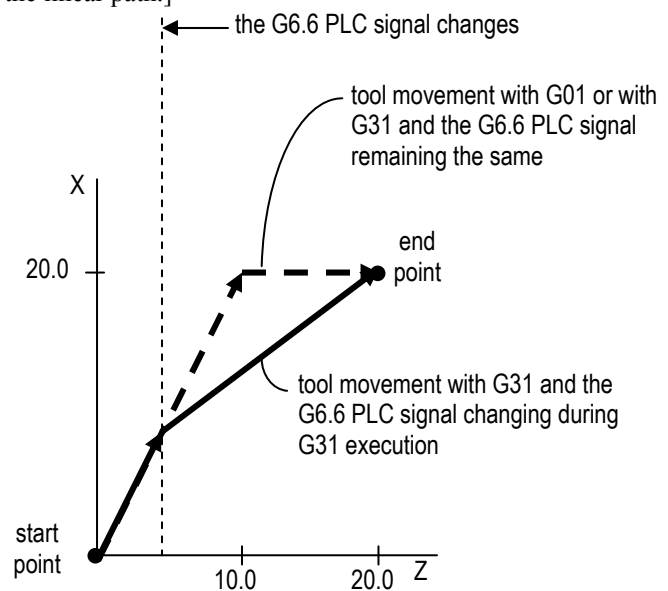
NOTE: these coordinate values could be absolute or incremental, relative to the part origin or to the current tool position, depending upon whether G90 or G91 is in effect)

F – feedrate (of the movement along the linear tool path); **[NOTE:** the feedrates for the individual axes are calculated so as to produce the specified feedrate along the linear path.]

Example



Blocks of Code with G01:
N10 G91 G01 X20.0 Z10.0;
N11 Z10.0;



Blocks of Code with G31:
N10 G91 G31 X20.0 Z10.0;
N11 Z10.0;

Figure 6-18: Difference between G01 Linear Interpolation and G31 Skip Cutting

Limitations

G31 is valid for one block only.

Other G-Codes Which Affect G31

The G90 and G91 codes specifying absolute and incremental programming, respectively, affect whether the coordinates you specify will be treated as absolute coordinates (relative to the part origin) or incremental coordinates (relative to the current tool position).

6.3.14 Automatic Tool Length Compensation Calibration (G37)

Description

G37 allows you to automatically and accurately calibrate the tool length offset value by lowering the appropriate tool axis toward the surface until the surface is detected via a sensor signal. You are free to choose any method of sensing you wish, as long as a signal is triggered and sent to the G201.4 address of PLC when the surface is detected.

For the Panasonic Realtime Express (RTEX) interface system (both the RTE and RT4 versions), there is an additional option for the surface detection signal: the high-speed first input of the third axis servo drive in the RTEX interface system. In this case, both the G201.4 PLC signal and the A4N servo drive signal are monitored simultaneously. If either signal is set “high,” the surface detection signal will be tripped.

You must first specify the address of an existing tool length offset value using G43/G44 (see *Section 6.3.17: Positive/Negative Tool Length Compensation /Compensation Cancel (G43, G44, G49)* for more information). For instance, if you want to use the tool length offset value for Tool #1 (in the Tool Offset Compensation Screen of Settings Mode in ServoWorks S-100M/S-120M/S-140M), you would execute the block of code “G43 H1” before the block containing G37. The final tool length offset is equal to the tool length offset value of the specified tool address, plus the calibrated tool length offset value from G37.

When executing G37, you must specify an axis and the theoretical surface position. The specified axis moves toward the theoretical surface position, but does not stop moving after it reaches the theoretical surface position. The axis stops when the surface detection signal is triggered, either before, after, or exactly when the axis reaches the specified surface position:

- When the surface detection signal is triggered *before* the axis reaches the theoretical surface position, a negative tool length offset calibration is applied, equal to the distance between the position where the axis stopped and the theoretical surface position.
- When the surface detection signal is triggered *after* the axis passes the theoretical surface position, a positive tool length offset calibration is applied, equal to the distance between the position where the axis stopped and the theoretical surface position.
- When the surface detection signal is triggered exactly when the axis reaches the theoretical surface position, no tool length offset calibration is applied.

Required Format

Tool Length Offset Type A: $\left\{ \begin{array}{c} G43 \\ G44 \end{array} \right\} Z H$
G37 Z

Tool Length Offset Type B: G17 $\left\{ \begin{array}{c} G43 \\ G44 \end{array} \right\} Z H$
G37 Z

G18 $\left\{ \begin{array}{c} G43 \\ G44 \end{array} \right\} Y H$
G37 Y

G19 $\left\{ \begin{array}{c} G43 \\ G44 \end{array} \right\} X H$
G37 X

Tool Length Offset Type C: $\left\{ \begin{array}{c} G43 \\ G44 \end{array} \right\} \alpha H$
G37 {X or Y or Z}

Possible Parameters That Can Be Used With G37

X or Y or Z – indicates the theoretical surface position when the calibrated tool length offset is 0 (note that each axis requires a specific “Tool Length Compensation Type” setting and/or plane selection [see *Section 6.3.17: Positive/Negative Tool Length Compensation /Compensation Cancel (G43, G44, G49)* for more information].

H – address of the tool length offset value (which is set in Settings Mode of ServoWorks S-100M/S-120M/S-140M).

(For example, H1 is the address for the tool length offset value of Tool #1.)

Example

G43 H1
G37 Z10.

The tool will move toward Z10. and continue moving down after it reaches Z10., but stop when the surface detection signal is triggered OR if the specified range limit is reached (see the explanation in the following section).

G37 Program Parameters in ServoWorks S-100M/S-120M/S-140M

The ServoWorks S-100M/S-120M/S-140M Configuration Mode allows you to set parameters for the execution of the G37 code, in the Tool Length Compensation frame of the Tool Compensation Screen, shown in the following figure:

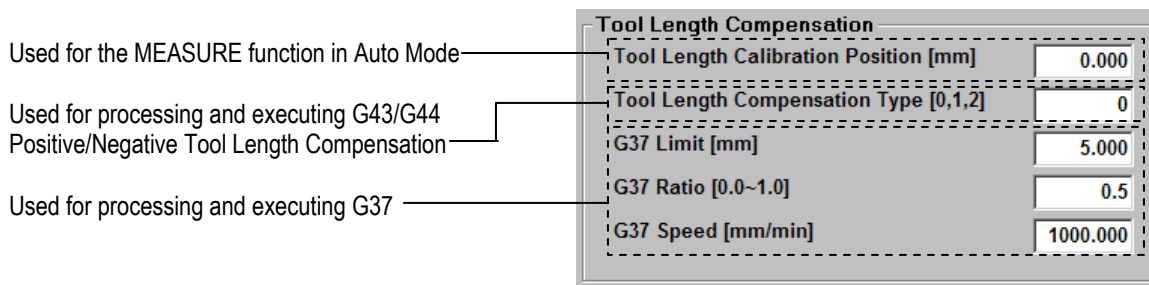


Figure 6-19: Tool Length Compensation Frame in the Tool Compensation Screen of Configuration Mode

Three program parameters apply directly to the G37 execution by the ServoWorks G-Code Parser. A fourth program parameter (“Tool Length Compensation Type”) applies to G43/G44, which must be programmed just prior to G37, and so is explained here also. [The fifth program parameter has nothing to do with G37.]

An explanation of the three parameters that apply to G37 follows:

- G37 Limit** – The valid range of tool length offset calibration values is between the positive G37 Limit and the negative G37 Limit, as measured from the theoretical surface position. For example, if the G37 Limit is 2.0 mm, then the tool length offset calibration value must be less than 2.0 mm and greater than -2.0 mm. In ServoWorks S-100M/S-120M/S-140M, an error message will be displayed if the surface detection signal is triggered before the axis reaches the surface position minus the G37 Limit, or if the axis reaches the surface position plus G37 Limit without the surface detection signal having been triggered.

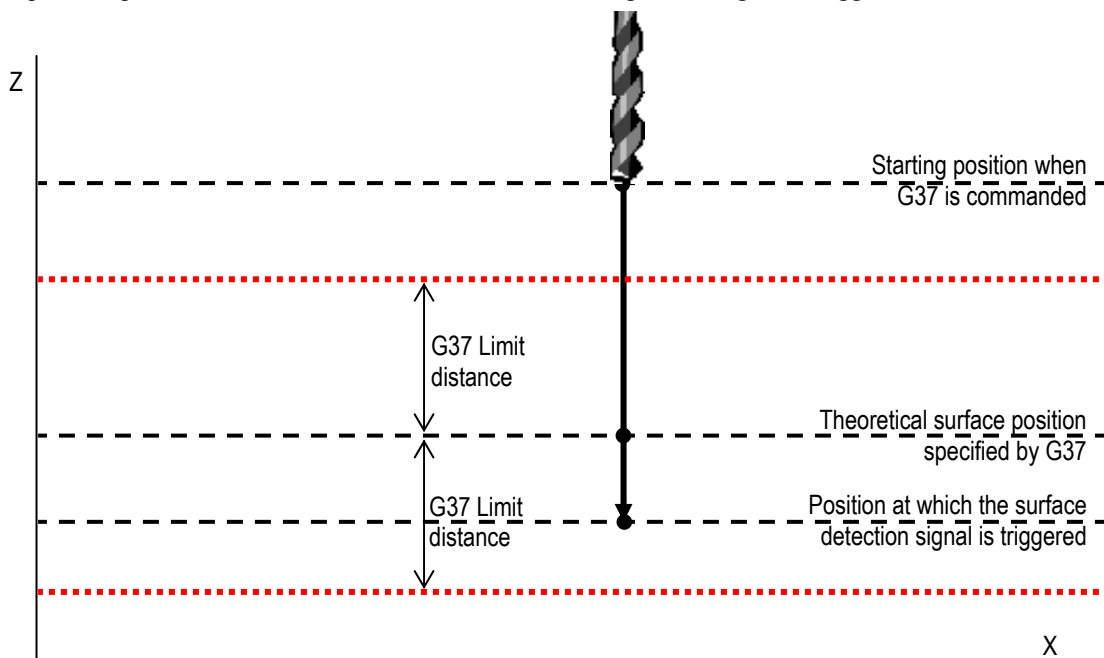


Figure 6-20: G37 Limit

- G37 Ratio** – When G37 is executed, the distance from the starting position to the theoretical surface position (specified by G37) is broken into two movements at two feedrates. The first movement is executed at the rapid traverse rate (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*) for that axis. The second movement is executed at the G37 Speed set in the Tool Length Compensation Frame shown in Figure 6-19 (see the explanation in the next bullet point). The G37 Ratio specifies the fraction of the distance from the starting position to the theoretical surface position at which movement occurs at the rapid traverse feedrate. The remainder of the distance from the starting position to the theoretical surface position occurs at the G37 Speed. **[NOTE:** The distance from the theoretical surface position to where the surface detection signal is triggered, is always at the G37 feedrate.]

If the G37 Ratio is 0, no movement will occur at the rapid traverse feedrate; the entire movement toward the surface position will be at the G37 Speed. Then the axis will continue at the G37 speed until the surface detection signal is triggered.

If the G37 Ratio is 1, the axis will move the entire distance from the starting position to the theoretical surface position at the rapid traverse feedrate, and assuming no surface detection signal was detected prior to the theoretical surface position, continue moving at the G37 Speed until the surface detection signal is triggered.

If the G37 Ratio is 0.5, the axis will move half the distance from the starting position to the theoretical surface position at the rapid traverse feedrate, then (and assuming no surface detection signal was detected at this point) continue moving at the G37 Speed until the surface detection signal is triggered.

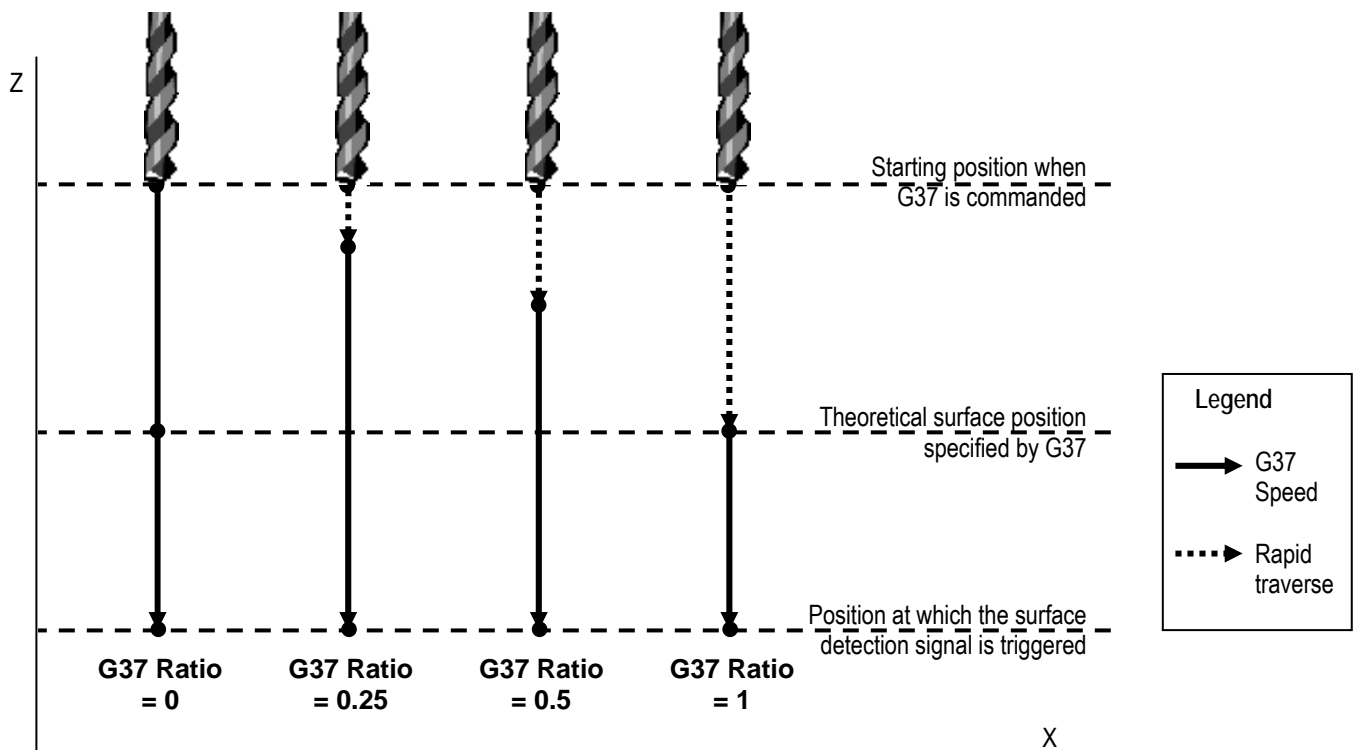


Figure 6-21: G37 Execution Showing Different G37 Ratios

- **G37 Speed** – The feedrate to be used while probing for the surface detection signal. Some fraction of the initial movement toward the surface may not be at the G37 Speed, but at the rapid traverse feedrate, as dictated by the G37 Ratio. [See the explanation above.]
- **Tool Length Compensation Type** – When the Tool Length Compensation Type is 0, all tool length offsets are applied to the Z axis (only the Z axis may be used with G37). When the Tool Length Compensation Type is 1, the tool length offsets are applied to the axis perpendicular to the selected plane. For example, if the ZX plane is selected (by the G-Code G18), the tool length offsets are applied to the Y axis (only the Y axis may be used with G37). [NOTE: When the Tool Length Compensation Type is 2, G37 isn't supported, and you will get an error.]

Notes

- Any calibration offset obtained by G37 is added to the tool length offset of the tool address specified by G43:

$$\begin{aligned} \text{Final compensation value} = & (\text{tool length offset of the tool address specified by G43}) \\ & + [(\text{coordinates of the position at which the surface detection signal is triggered}) - \\ & (\text{coordinates of the theoretical surface position specified by G37})] \end{aligned}$$

- Tool length compensation calibration cancel (and tool length offset cancel) can be programmed in one of two ways:
 - 1) Programming a G49 block of code. (This cancels offsets along all axes.)
 - 2) Programming a G43/G44 block of code with H0. (This cancels offsets along only the specified axis, perpendicular to the specified plane.)

Example

Consider the case where the tool length offset value for Tool #1 is 5.0 mm, and the following code is executed:

```
G43 H1
G37 Z20.
```

Furthermore, assume that the surface detection signal is triggered when the tool is at the position $Z = 19.5$ mm (before it reaches the theoretical surface position of $Z = 20.$ mm). This will set the tool length offset calibration value to be -0.5 mm. The total tool length offset is then

$$5.0 \text{ mm} + (-0.5 \text{ mm}) = 4.5 \text{ mm}.$$

Assume the following code is executed:

```
G01 Z0.
```

This will move the Z axis to $0 \text{ mm} + 4.5 \text{ mm} = 4.5 \text{ mm}$.

If you want the tool length offset to only equal the calibration offset obtained by G37, you must choose a tool for which the tool length offset value is 0.

6.3.15 Tool Radius Compensations (TRC – G40, G41 and G42)

Description

G40: Tool radius compensation (TRC) cancel

G41: Tool radius compensation (TRC) right

G42: Tool radius compensation (TRC) left

TRC enables the contour of the part to be programmed directly without taking the dimensions of the tool into account.

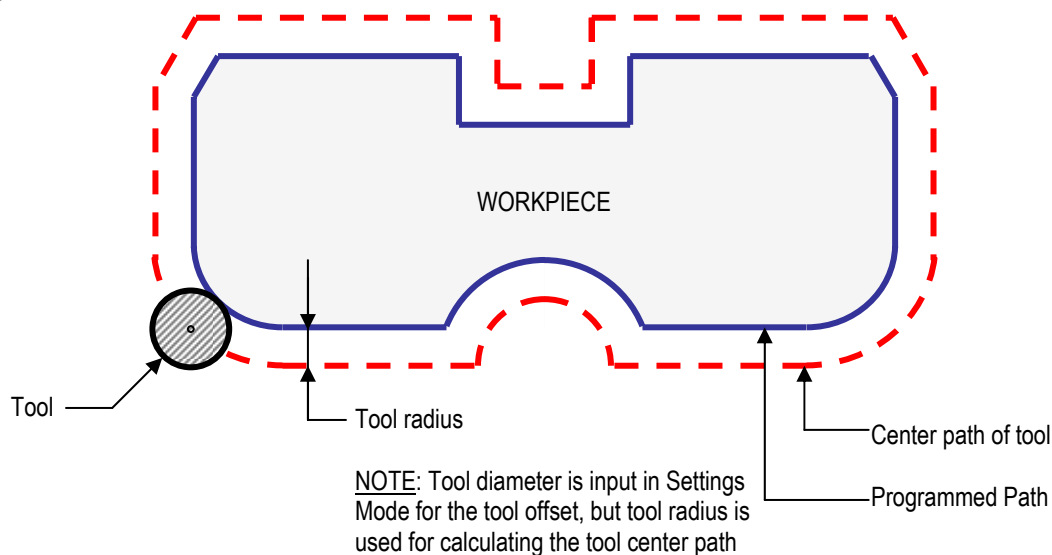


Figure 6-22: Tool Radius Offset

Required Format

G41 H##

...

G40

OR

G42 H##

...

G40

Possible Parameters That Can Be Used With G40 – G42

H – address of the tool length offset value (which is set in Settings Mode of ServoWorks S-100M/S-120M/S-140M).
(For example, H1 is the address for the tool length offset value of Tool #1.)

Other Codes Which Affect G41 and G42

A D code specifies a tool offset (i.e. "D1"). [See page 2-1 for more information on D codes.]

Limitations

- Look-ahead buffer: 100 blocks
- Only motion (G00, G01, G02, G03) and non-motion (G04, M, S, B, and F codes) are eligible in TRC mode
- The 1st and the last motions of TRC mode cannot be G02 or G03
- Scaling (G51), and mirror image (G51.1) must not be used when tool radius compensation is on

Default

G40 is the default mode when none of the G40, G41 or G42 codes has been programmed.

Reference

See *Section 13.2.1: Tool Radius Compensation* in the *Reference Manual for ServoWorks CNC Parameters and Functions* for additional information.

6.3.16 Normal Direction Control (G40.1, G41.1 and G42.1)

Description

Normal direction control is a function that keeps the tool oriented at a constant angle relative to the cutting direction. With normal direction control, a tool with a rotation axis can be controlled so that it is kept constantly perpendicular to the tool path, as the tool moves in the X-Y plane. ServoWorks S-100M/S-120M/S-140M calculates the required geometry modifications to apply to tool movement, to keep the tool normal to its path at all times.

Normal direction control is ideal for clean and functional bevel cutting and chamfering, particularly rotary bevel cutting. This function greatly simplifies the programming of the beveled shapes and chamfers that are common in plasma cutting and fuel cutting applications.

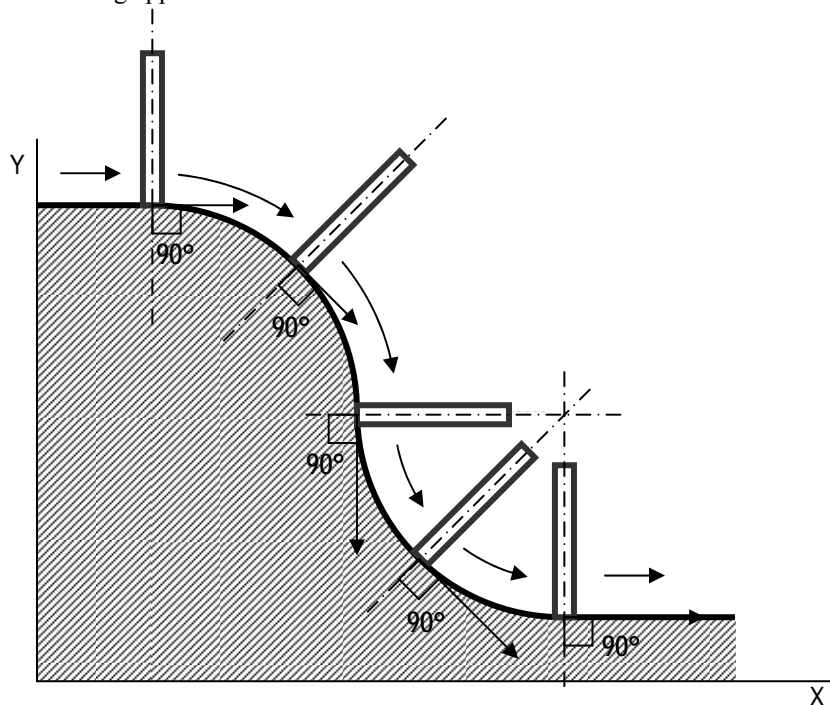


Figure 6-23: Sample Tool Movement during Normal Direction Control

Without normal direction control, if you wanted the tool to point at a constant angle relative to the cutting direction, you would have to manually insert commands to rotate the tool axis into linear and circular interpolation G codes, by finding the angle that the tool will face before and after interpolation, in addition to the direction the tool should turn during the interpolation, for each line of G code. With normal direction control, you can write NC programs that will automatically insert the appropriate rotation parameters into any interpolation G codes that are commanded.

G40.1: Normal direction control cancel

G41.1: Normal direction control left

G42.1: Normal direction control right

The angle for normal direction control is measured from the positive X axis, in the counterclockwise direction, as shown in the figure below:

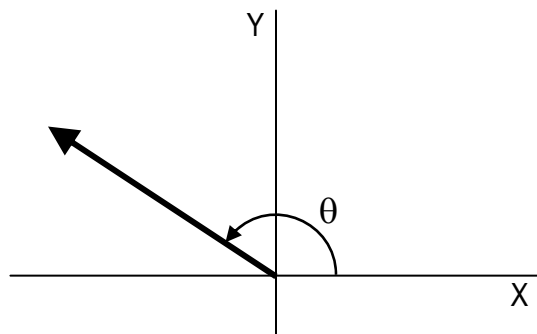


Figure 6-24: Definition of Angle for Normal Direction Control

The “left” and “right” in the two modes of normal direction control designate the side the tool is at, relative to the substrate being cut, from the perspective of looking in the direction of motion. For example, when cutting straight eastwards using G01X100, if the tool is on the north and the substrate on the south, G41.1 should be used, while if the tool is on the south and the substrate on the north, G42.1 should be used. When normal direction control is on, the tool face will be perpendicular to the cutting direction at all times.

Default

G40.1 is the default mode when none of the G40.1, G41.1 and G42.1 codes has been programmed.

Required Format

G41.1 H##

...

G40.1

OR

G42.1 H##

...

G40.1

Possible Parameters That Can Be Used With G40.1, G41.1 and G42.1

None.

G41.1 and G42.1 Program Parameters in ServoWorks S-100M/S-120M/S-140M

The ServoWorks S-100M/S-120M/S-140M Configuration Mode allows you to set four parameters for the execution of the G41.1 and G42.1 normal direction control code, in the Normal Direction Control frame of the Normal Direction Control Screen, shown in the following figure:

Normal Direction Control	
Rotary Axis No.	5
Rotation Feedrate [deg/min]	1000.0
Angle Limit [deg]	10.0
Length Limit [mm]	10.0

Figure 6-25: Normal Direction Control Frame in the Normal Direction Control Screen of Configuration Mode

An explanation of the four parameters that apply to G41.1 and G42.1 follows:

- **Rotary Axis No.** – This number designates which axis controls the angle of the tool, as indicated in the following table:

Axis Name	Rotary Axis No.
Axis 1	1
Axis 2	2
Axis 3	3
Spindle Axis	4
Axis 4	5
Axis 5	6
Axis 6	7
Axis 7	8

Table 6-5: Rotary Axis No. and Corresponding Axis Names

- **Rotation Feedrate** – This specifies the feedrate in effect when normal direction control causes the rotary axis to rotate. Note that this parameter is only used when the rotary axis rotates in position, such as at a sharp corner or the starting point of an arc. When there is continuous rotation together with tool movement, which normally happens during circular interpolation, the X and Y axes feedrates override this parameter.
- **Angle Limit** – This specifies the minimum angle of rotation that can be added to a block by the normal direction control function. All angular rotations less than the Angle Limit are ignored.

NOTE: This parameter is useful when small rotations will not affect the operation of the machine tool and can be ignored. Note that the ignored angles accumulate in that the angle of rotation becomes larger and larger as more angles are ignored for sequences of angles of the same concavity. For example, when cutting an octagon from the outside, the tool normally would rotate 45° per corner. If the Angle Limit is set to 60°, the first corner is ignored because a rotation of 45° is less than 60°. However, on the second corner, the tool rotation would now be 90° since the first rotation was ignored and hence the tool must rotate 45° + 45° to achieve the target orientation. Thus, a 90° rotation is performed on this second corner.

- **Length Limit** – This specifies the length of a line (including ones cut using circular interpolation) for which angular rotation should occur. For G code blocks that cut a shorter distance than the Length Limit, no angular rotation will be inserted by normal direction control function. [NOTE: This parameter is very similar to the Angle Limit parameter.]

Examples Demonstrating the Difference Between G41.1 and G42.1

Assume G41.1 is called, and then a circle is cut. The arc starts at the top of the circle and continues clockwise to the 10 o'clock to 11 o'clock position of the circle. The arrows on the figure indicate the direction that the machine tool is pointing. First, after the machine moves to the starting point of the arc, the rotary axis is rotated to 270° to position the tool perpendicular to the arc. Notice that the tool is now situated so that it is pointing toward the arc from the left side of the view toward the direction of movement. Normal direction control modifies the block drawing the arc appropriately so that the rotary axis rotates in a way that it is always pointing perpendicular to the arc, as labeled in the following figure. At the end of the arc, the tool has rotated to the 315° position.

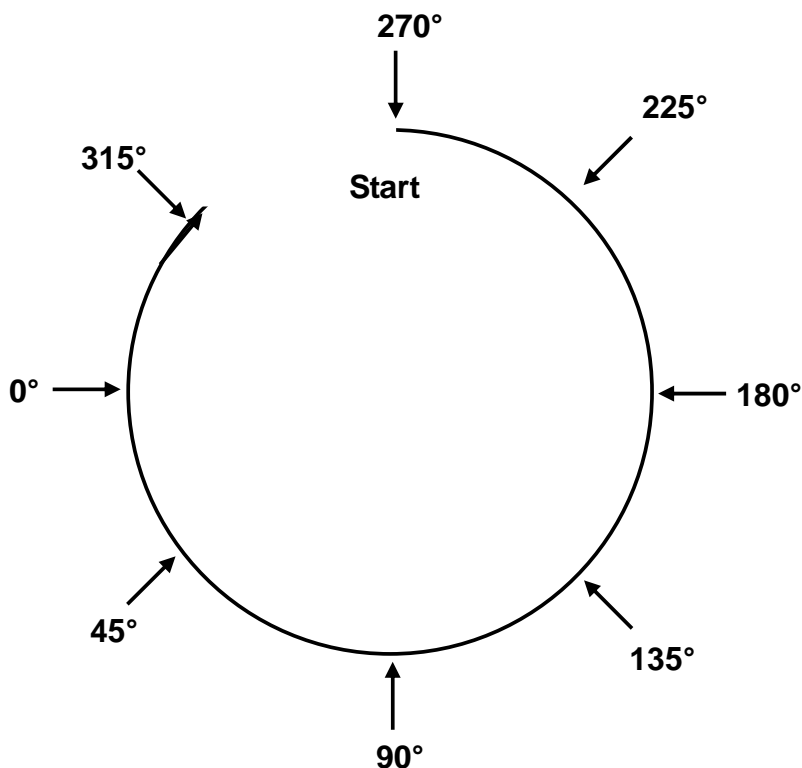


Figure 6-26: Normal Direction Control, Example of G41.1

What happens if we cut the same arc using G42.1 instead of G41.1? The tool is designated to be at the right side of the view toward the direction of movement. This means that the arc is cut from the inside. At the starting point, the rotary axis rotates to 90°, and as the arc is cut, the angle decreases to 0° and further down from 360° to 135°, where the arc is completed. Notice that at all times, the tool is facing perpendicular to the substrate.

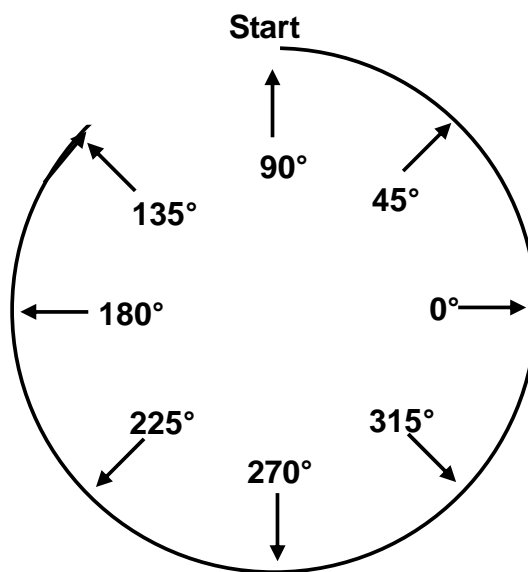


Figure 6-27: Normal Direction Control, Example of G42.1

It is important to realize that G42.1 does not necessarily always cut inside the shape and G41.1 the outside of the shape. Indeed, if we cut the arc in the other direction, from the 10 o'clock to 11 o'clock direction counterclockwise to the 12 o'clock position, G41.1 would position the tool inside the arc and G42.1 would position the tool outside the arc.

Example Demonstrating the Effects of Various Values of the Angle Limit Parameter

The following example demonstrates the effects of varying values for the Angle Limit parameter on the rotations that the tool undergoes while cutting.

The shape shown in the following figure will be cut using various values for the Angle Limit parameter. The shape is cut in the clockwise direction starting from the point marked "Start."

In the first instance, we will demonstrate what happens when the Angle Limit is set to 0° (see the following figure). At the starting position, the tool is pointing in the 0° direction. First, the normal direction control function moves the rotary axis to 270° before drawing the first line. After the first line is drawn, the rotary axis rotates 45° to the 225° position to place the tool perpendicular to the second line. At the end of the second line, the tool is rotated 135° to the 90° position. At the end of the third line, the tool is rotated 90° to the 0° position. At the end of the fourth line, the tool returns to the starting position.

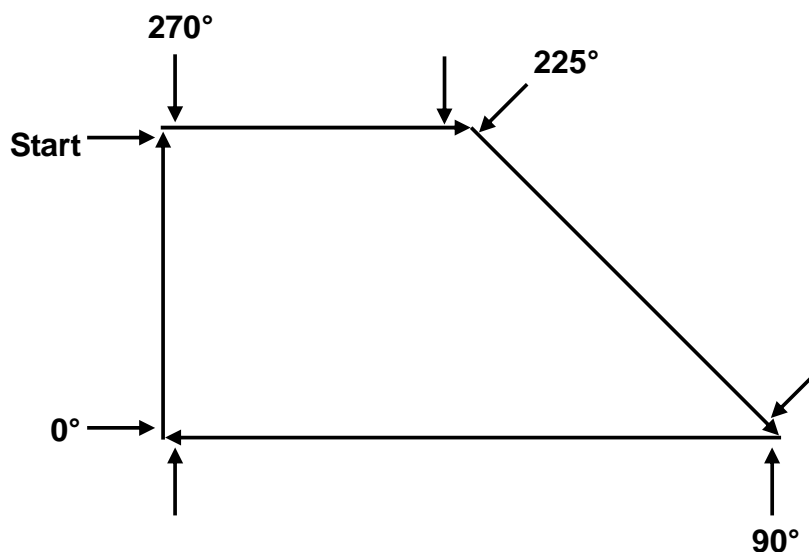


Figure 6-28: Normal Direction Control Example of Angle Limit – Figure 1 of 4

Since no rotations less than 45° are used to cut this shape, an Angle Limit value of less than 45° will have no effect, cutting the shape in the manner described above.

Let's consider a second instance when the Angle Limit is set to 46° (shown in the following figure). The rotation between the first and second line is ignored since it is only 45° . At the corner of the second and third lines, the tool rotates 180° from the 270° position to the 90° position. The rotation at the corner of the third and fourth lines remains unaffected, since the rotation is 90° , greater than the 46° Angle Limit.

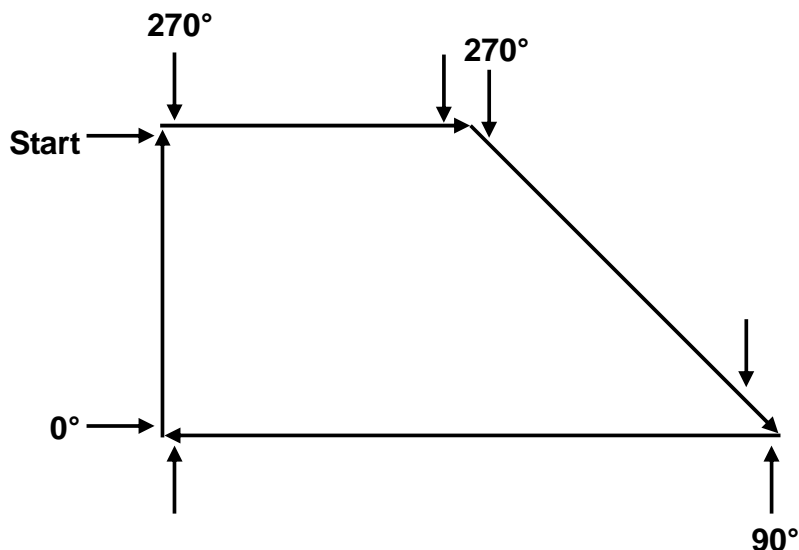


Figure 6-29: Normal Direction Control Example of Angle Limit – Figure 2 of 4

Let's consider a third instance when the Angle Limit is set to 91° . An Angle Limit of 91° is slightly more complicated. Since the first rotation at the start of the first line is a 90° rotation, the rotation is ignored. This means that at the corner of the first and second lines, where there normally is a 45° rotation, the tool must rotate an extra 90° for a total of 135° . Since 135° is greater than the Angle Limit of 91° , a rotation is made at that point. At the corner of the second and third lines, a 135° rotation is performed. At the corner of the third and fourth lines, a 90° rotation normally occurs, but since 90° is less than the Angle Limit, no rotation occurs. At the end of the run, the rotary axis faces 90° . This example is a strange case, probably only useful for illustrative purposes.

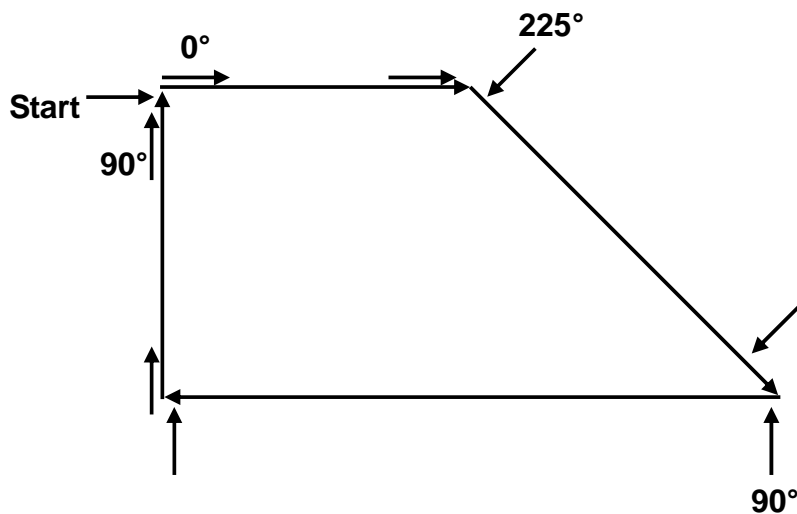


Figure 6-30: Normal Direction Control Example of Angle Limit – Figure 3 of 4

Finally, let's consider a fourth instance when the Angle Limit is set to 181° . We will get no angle movement at all. This is because all angle movements can be performed in 180° or less (if we take the shorter path), so at no point can there be a rotation of over 181° .

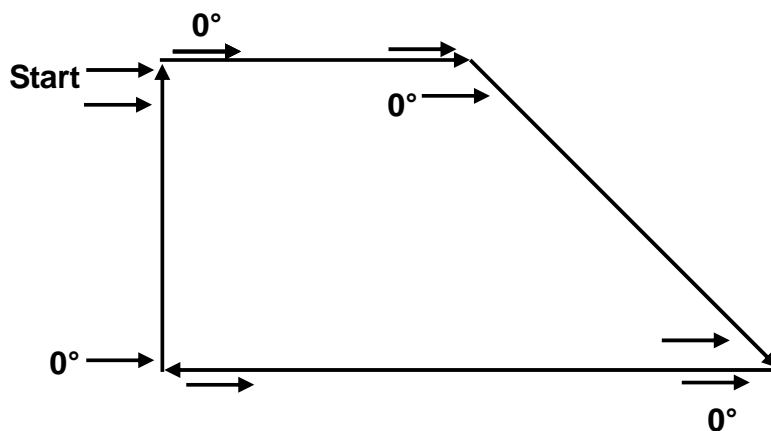



Figure 6-31: Normal Direction Control Example of Angle Limit – Figure 4 of 4

Limitations

The “Axis type” of the tool must be “Rotary” or “Rotary ST.” The “Axis Type” parameter is set in the “Machine” screen of Configuration Mode. You must use the  button to display the settings for Axes 4 and 5.

6.3.17 Positive/Negative Tool Length Compensation (TLC)/ Compensation Cancel (G43, G44, G49)

Description

Tool length compensation accounts for the difference between the actual tool position and the theoretical tool position. When T codes are executed in a part program, the program position is actually shifted to account for tool offset (as opposed to compensating each actual movement command to account for tool offset).

G43: Positive tool length compensation
G44: Negative tool length compensation
G49: Tool length compensation cancel

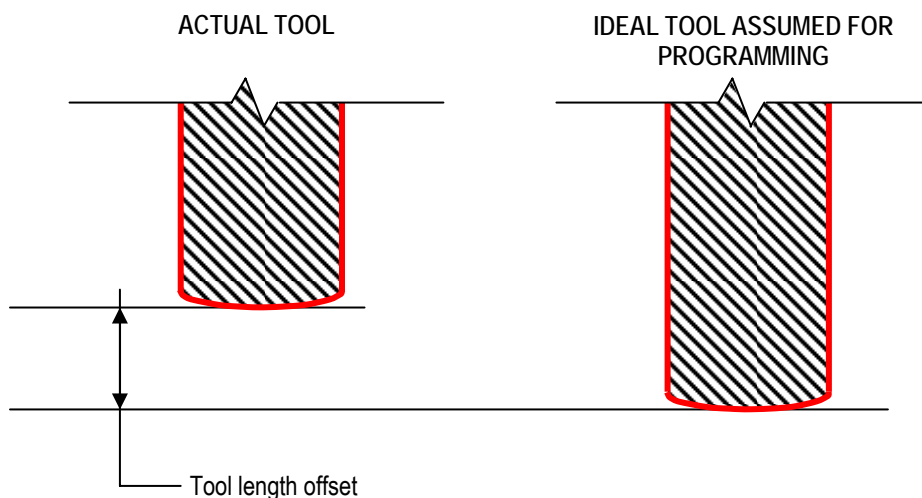


Figure 6-32: Tool Length Offset

With G43, the tool length offset value is ADDED to the coordinates of the end position specified by the program (regardless of whether the absolute command [G90] or the incremental command [G91] is in effect). If the tool length offset value is a negative number, adding this negative number causes the tool to move in the negative direction.

With G44, the tool length offset value is SUBTRACTED to the coordinates of the end position specified by the program (regardless of whether the absolute command [G90] or the incremental command [G91] is in effect). If the tool length offset value is a negative number, subtracting this negative number causes the tool to move in the positive direction.

There are three methods of setting tool length offsets:

- 1) Tool Length Offset Type A (for tool length compensations in the Z-axis direction)
- 2) Tool Length Offset Type B (for tool length compensations in the X-, Y- or Z-axis directions)
- 3) Tool Length Offset Type C (for tool length compensations along any specified axis)

Required Format

The format type is set using a “Tool Length Compensation Type” parameter in the Tool Length Compensation Frame in the Tool Compensation Screen of Configuration Mode in ServoWorks S-100M/S-120M/S-140M. “0” indicates tool length offset type A, “1” indicates tool length offset type B and “2” indicates tool length offset type C, explained as follows:

Tool Length Offset Type A: $\left\{ \begin{matrix} G43 \\ G44 \end{matrix} \right\} Z H$

Tool Length Offset Type B: $G17 \left\{ \begin{matrix} G43 \\ G44 \end{matrix} \right\} Z H$

$G18 \left\{ \begin{matrix} G43 \\ G44 \end{matrix} \right\} Y H$

$G19 \left\{ \begin{matrix} G43 \\ G44 \end{matrix} \right\} X H$

Tool Length Offset Type C: $\left\{ \begin{matrix} G43 \\ G44 \end{matrix} \right\} \alpha H$



CAUTION

For Tool Offset Type C, you must use BOTH G49 and H0 to cancel the offset, or you will generate an alarm.

Possible Parameters That Can Be Used With G43, G44

X – indicates X axis tool length compensation

Y – indicates Y axis tool length compensation

Z – indicates Z axis tool length compensation

α – indicates the axis for tool length compensation (i.e., X, Y, Z, A, B or C)

H – address of the tool length offset value (which is set in Settings Mode of ServoWorks S-100M/S-120M/S-140M).
(For example, H1 is the address for the tool length offset value of Tool #1.)

Example

G43 Z50.5 H3 (If H3 corresponds to the tool length offset value of 10.0 for tool #3, then the tool will move to 50.5 + 10.0 = 60.5)

Variations

To program tool length offsets along two or more axes, use a separate block of code for each axis:

G18 G43 Y H1

G19 G43 X H5

G43/G44 Program Parameters in ServoWorks S-100M/S-120M/S-140M

There is a “Tool Length Compensation Type” parameter in the Tool Length Compensation Frame in the Tool Compensation Screen of Configuration Mode in ServoWorks S-100M/S-120M/S-140M. (See Figure 6-19.) This parameter has three valid values, each of which indicates a method of setting tool length offsets:

- **0** – Tool Length Offset Type A (for tool length compensations in the Z-axis direction)
- **1** – Tool Length Offset Type B (for tool length compensations in the X-, Y- or Z-axis directions)
- **2** – Tool Length Offset Type C (for tool length compensations along any specified axis)

The default value for this parameter is “0.”

Notes

Tool length offset cancel can be programmed in one of two ways:

- 3) Programming a G49 block of code. (This cancels offsets along all axes.)
- 4) Programming a G43/G44 block of code with H0. (This cancels offsets along only the specified axis, perpendicular to the specified plane.)

Default: 0



CAUTION

To turn on the TLC while the TCP is on will be the cause of unexpected error. Do not command the TLC if the TCP is on.

6.3.18 Tool Center Point (TCP) Control / Control Cancel (G43.2, G49) [ServoWorks S-140M Only]

Description

Tool center point control compensates for the distance between the actual tool position (i.e. the tip of the nozzle) and the tool center point (the theoretical tool position) — the program position is actually shifted to account for the tool center point, as opposed to compensating each actual movement command in the part program. The actual trajectory of G code is applied to the tool center point, and the tool center point acts independently from axis movement or angle movement. The tool center point moves at the feedrate specified by the part program, along the path specified by the part program.

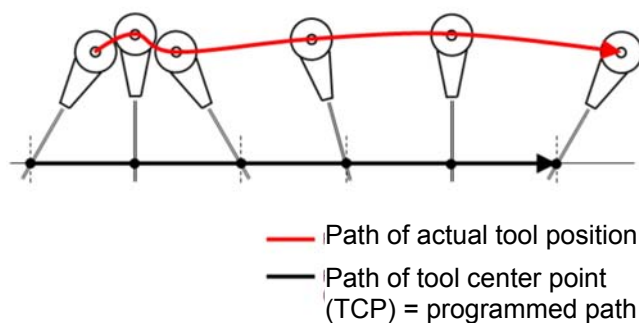


Figure 6-33: Tool Center Point

The tool center point (TCP) control feature simplifies the programming of complex workpieces and allows you to cut complex three-dimensional objects by manipulating the machine tool in five dimensions: X, Y, Z, the angle on the XY plane, and the angle away from the Z axis. By specifying the machine arm length and tool length, the machine tool can be easily rotated while preserving the tool center point, the point of contact between the machine tool and the substrate. The tool center point is a constant offset from the machine tool position in the direction specified by the machine tool angles. When the machine tool is rotated, the machine tool X, Y, and Z coordinates will change in order to maintain the same tool center point position.

When G43.2 is executed, the TCP function uses parameters that are set in the ServoWorks S-140M interface. Future rotations and manipulations of Axis 4 and Axis 5 will automatically preserve the tool center point, instead moving the tool X, Y, and Z coordinates appropriately.

G43.2: Tool center point control on
G49: Tool center point control cancel

The angle on the XY plane and the angle away from the Z axis are called Axis 4 and Axis 5, respectively. These two axes can be used to define any angular orientation of the machine tool. Axis 4 is the angle on the XY plane, measured counterclockwise from the positive X axis to the machine tool. Axis 5 is the angle away from the Z axis, measured from the negative Z axis to the machine tool.

Axis 4 and Axis 5 are shown as follows:

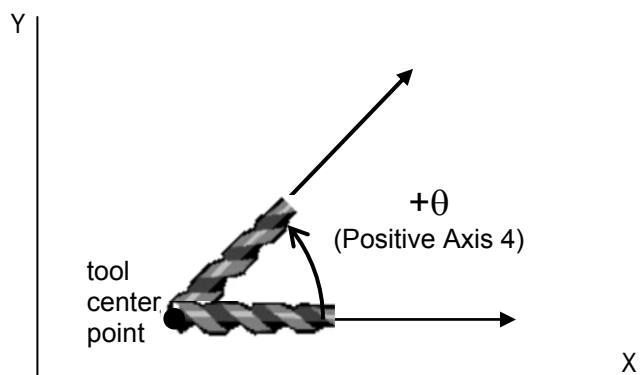


Figure 6-34: Definition of Axis 4 for G43.2, Tool Center Point

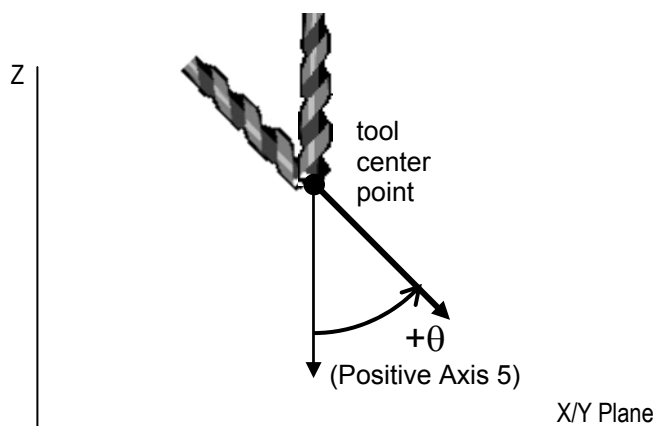


Figure 6-35: Definition of Axis 5 for G43.2, Tool Center Point

Required Format

G43.2 H##

...

G49

Possible Parameters That Can Be Used With G43.2

H – address of the tool length offset value (which is set in Settings Mode of ServoWorks S-140M). (For example, H1 is the address for the tool length offset value of Tool #1.)

G43.2 Program Parameters in ServoWorks S-100M/S-120M/S-140M

There are two parameters that describe the machine tool dimensions that need to be set in the ServoWorks S-140M GUI, for the execution of the G43.2 code:

- **Arm Length**
 - Set in the “Machine” screen of Configuration Mode
 - Fixed value shared by all tools
 - Labeled “P” in the figure below
- **Tool Length**
 - Set in Settings Mode, in the Tool Offset Compensation screen
 - Invoked using G43.2 H##, where ## is replaced by the tool number
 - Can be changed dynamically while running an NC program
 - Labeled “P2” in the figure below

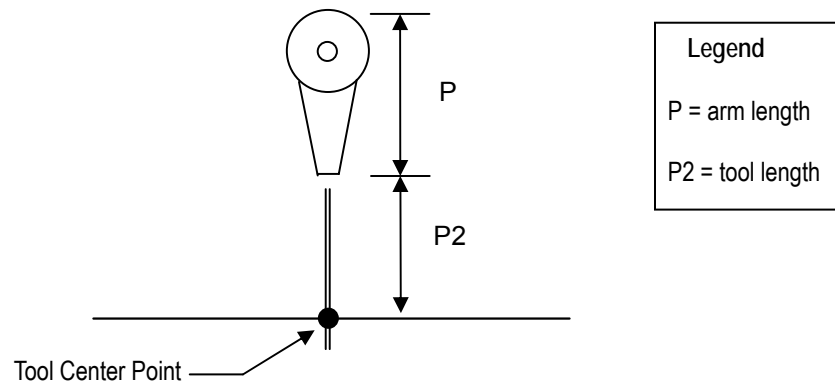


Figure 6-36: TCP Parameters Showing Arm Length and Tool Length

Arm Length is set in the “Machine” screen of Configuration Mode, in the “TCP” frame, as shown in the following figure:

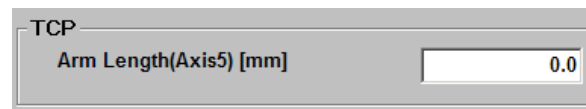


Figure 6-37: TCP Frame in the Machine Screen of Configuration Mode

Tool lengths are set in Settings Mode, in the Tool Offset Compensation Screen, shown as follows:

Workpiece Coordinate System					Tool Offset Compensation				
Tool Offset Values									
Tool #	Length (mm)		Diameter (mm)		Tool #	Length (mm)		Diameter (mm)	
	Geom	Wear	Geom	Wear		Geom	Wear	Geom	Wear
1	0.000	0.000	0.000	0.000	12	0.000	0.000	0.000	0.000
2	0.000	0.000	0.000	0.000	13	0.000	0.000	0.000	0.000
3	0.000	0.000	0.000	0.000	14	0.000	0.000	0.000	0.000
4	0.000	0.000	0.000	0.000	15	0.000	0.000	0.000	0.000
5	0.000	0.000	0.000	0.000	16	0.000	0.000	0.000	0.000
6	0.000	0.000	0.000	0.000	17	0.000	0.000	0.000	0.000
7	0.000	0.000	0.000	0.000	18	0.000	0.000	0.000	0.000
8	0.000	0.000	0.000	0.000	19	0.000	0.000	0.000	0.000
9	0.000	0.000	0.000	0.000	20	0.000	0.000	0.000	0.000
10	0.000	0.000	0.000	0.000	21	0.000	0.000	0.000	0.000
11	0.000	0.000	0.000	0.000	22	0.000	0.000	0.000	0.000

Page 1/12

Figure 6-38: The Settings Mode Window – Tool Offset Compensation Screen

Example

G43.2 H1
G01 C45.
G49
G43.2 H2
G01 B-45.
G49

In this example, Axis 4 (labeled C) is rotated 45 degrees using tool length 1, and Axis 5 (labeled B) is rotated -45 degrees using tool length 2.

Limitations

- The TCP function does not work for ServoWorks S-100M and S-120M.
- The “Axis Type” parameter for both Axis 4 and Axis 5 must be set to “Rotary” in order for the axes to be represented in terms of angles so that TCP can be enabled.

The “Axis Type” parameter is set in the “Machine” screen of Configuration Mode. You must use the



button to display the settings for Axes 4 and 5.

- Tool center point is intended to facilitate the execution of NC programs requiring complex three dimensional tool movements. As such, it is only usable in AUTO or MDI mode.
- DLACC (dynamic look-ahead contour control – invoked through G05 or G08) is not compatible with TCP. Instead, you should use the Corner Deceleration function and the Velocity Control in Circular Interpolation function to maintain smooth and accurate cutting with TCP.

- To turn on the TCP automatically enable the TLC to set the tool length. Setting tool length 0mm will be the same as not having the tool length offset, but using the TCP function.

Notes

Following are equations related to the calculations used in tool center point control:

Tool Center Point: X, Y, Z

$L = \text{arm length} + \text{tool length}$

Tool Base Point: X', Y', Z'

$X' = X - (L)(\sin(\theta_b))(\cos(\theta_c))$

$Y' = Y - (L)(\sin(\theta_b))(\sin(\theta_c))$

$Z' = Z + (L)(\cos(\theta_b))$

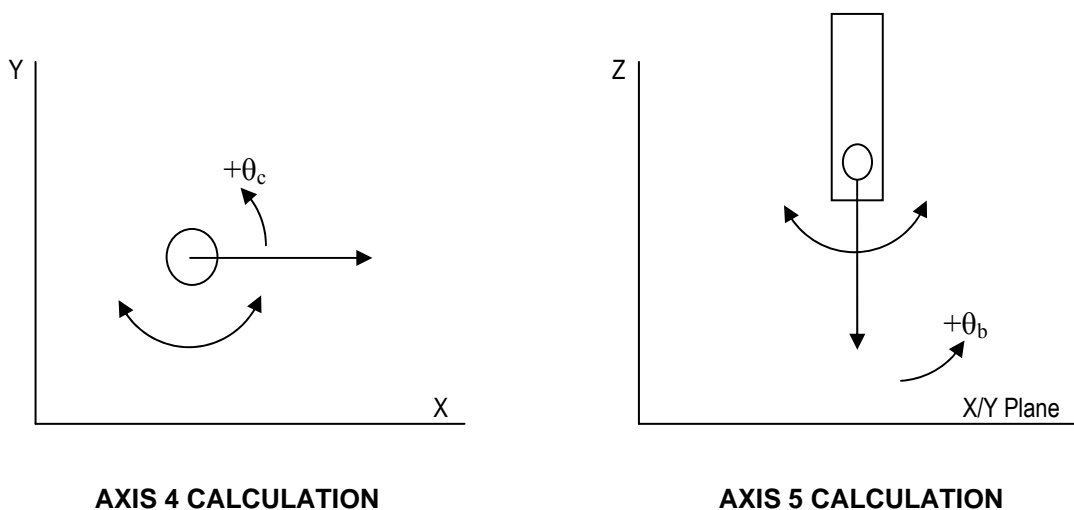


Figure 6-39: TCP Calculations for Axes 4 and 5

CAUTION

To enable the TLC while the TCP is enabled will be the cause of unexpected error.
 Do not command the TLC if the TCP function is enabled.

6.3.19 Scaling Off / On (G50, G51)

Description

The scaling function allows you to scale commands (and therefore parts) larger or smaller, for either all axes together, or for individual axes.

G50: Scaling off

G51: Scaling on

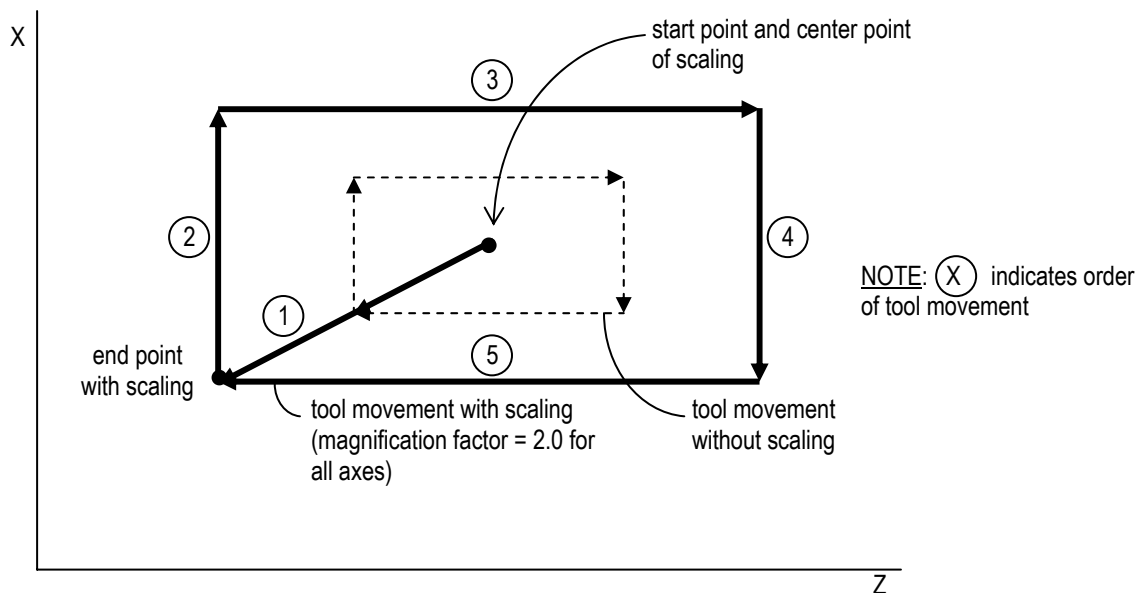


Figure 6-40: Scaling Function, Example #1

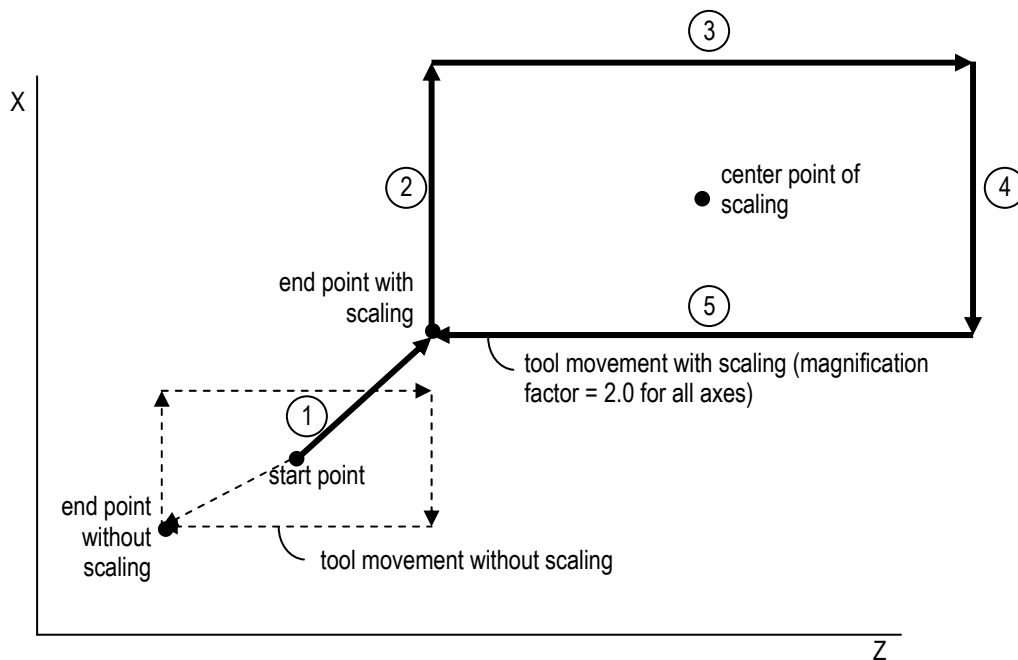


Figure 6-41: Scaling Function, Example #2

Required Format

G51 X Y Z P;

.
.
.
.

} Scaling
mode is in
effect

G50;

G51 X Y Z I J K;

.
.
.
.

} Scaling
mode is in
effect

G50;

Possible Parameters That Can Be Used With G50, G51

X – absolute coordinate value for the X axis center point of scaling

Y – absolute coordinate value for the Y axis center point of scaling

Z – absolute coordinate value for the Z axis center point of scaling

P – specifies the magnification of scaling (0.001 – 999.999) for all axes

I – specifies the magnification of scaling (0.001 – 999.999) for the X axis

J – specifies the magnification of scaling (0.001 – 999.999) for the Y axis

K – specifies the magnification of scaling (0.001 – 999.999) for the Z axis

NOTE: If you specify the P parameter (overall magnification factor), then you cannot specify the I, J or K parameters (individual axis magnification factor), or you will get an error. Similarly, if you specify the I, J and/or K parameters (individual axis magnification factor), then you cannot specify the P parameter (overall magnification factor), or you will get an error.

Examples

G51 X10.0 Y10.0 Z10.0 P2.0

G50;

G51 X10.0 Y10.0 Z10.0 I2.0 J2.0 K1.0

G50;

Limitations

Tool radius compensation (G41, G42) and mirror image (G51.1) must not be used when scaling is on.

Notes

- If I, J and/or K parameters are included in a command, but not all three of the I, J and K parameters are included, then the magnification factor which has not been specified remains at its previous setting from the last time G51 was executed. For instance, if I and J are specified, but K is not, K is assumed to remain at its previous setting.
- If none of the P, I, J and K parameters are specified, an error will be issued.
- If X, Y and Z parameters are omitted from the command, the current position where G51 is commanded becomes the center for scaling subsequent commands.

Default

G50 is the default mode when neither G50 nor G51 has been programmed.

6.3.20 Mirror Image Off / On (G50.1, G51.1)

Description

With the mirror image function, you can command the execution of the mirror image of a set of commands for one or more axes. This is particularly helpful if you have a part that is symmetrical in one or more directions; you would only need to program part of the whole part in a sub program, and then call the subprogram one or more times with the mirror image function on.

G50.1: Mirror image off

G51.1: Mirror image on

Required Format

```
G51.1 X Y Z;
.
.
.
.
G50.1 X Y Z;
```

} Mirror
image mode
is in effect

NOTE: With G50.1, mirror image is cancelled only for the axes included as parameters.

Possible Parameters That Can Be Used With G50.1, G51.1

X –coordinate value for the X axis center point of the mirror image
Y –coordinate value for the Y axis center point of the mirror image
Z –coordinate value for the Z axis center point of the mirror image

NOTE: these coordinate values could be absolute or incremental, relative to the part origin or to the current tool position, depending upon whether G90 or G91 is in effect)

Example

The following example demonstrates the usefulness of the mirror image function by using a subprogram for a piece of the part, and then calling that subprogram in the main program with two different axes to mirror the part about.

Main Program:

```
%
N0010 G00 G90;
N0020 M98 P100;
N0030 G51.1 X40.0;
N0040 M98 P100;
N0050 G51.1 Z40.0;
N0060 M98 P100;
N0070 G50.1 X0;
N0080 M98 P100;
N0090 G50.1 Z0;
%
```

Subprogram O100.dat:

```
%
G00 G90 X50.0 Z30.0;
G01 X70.0;
G01 Z20.0;
G01 X60.0;
G01 Z10.0;
G01 X50.0;
G01 Z30.0;
M99;
%
```

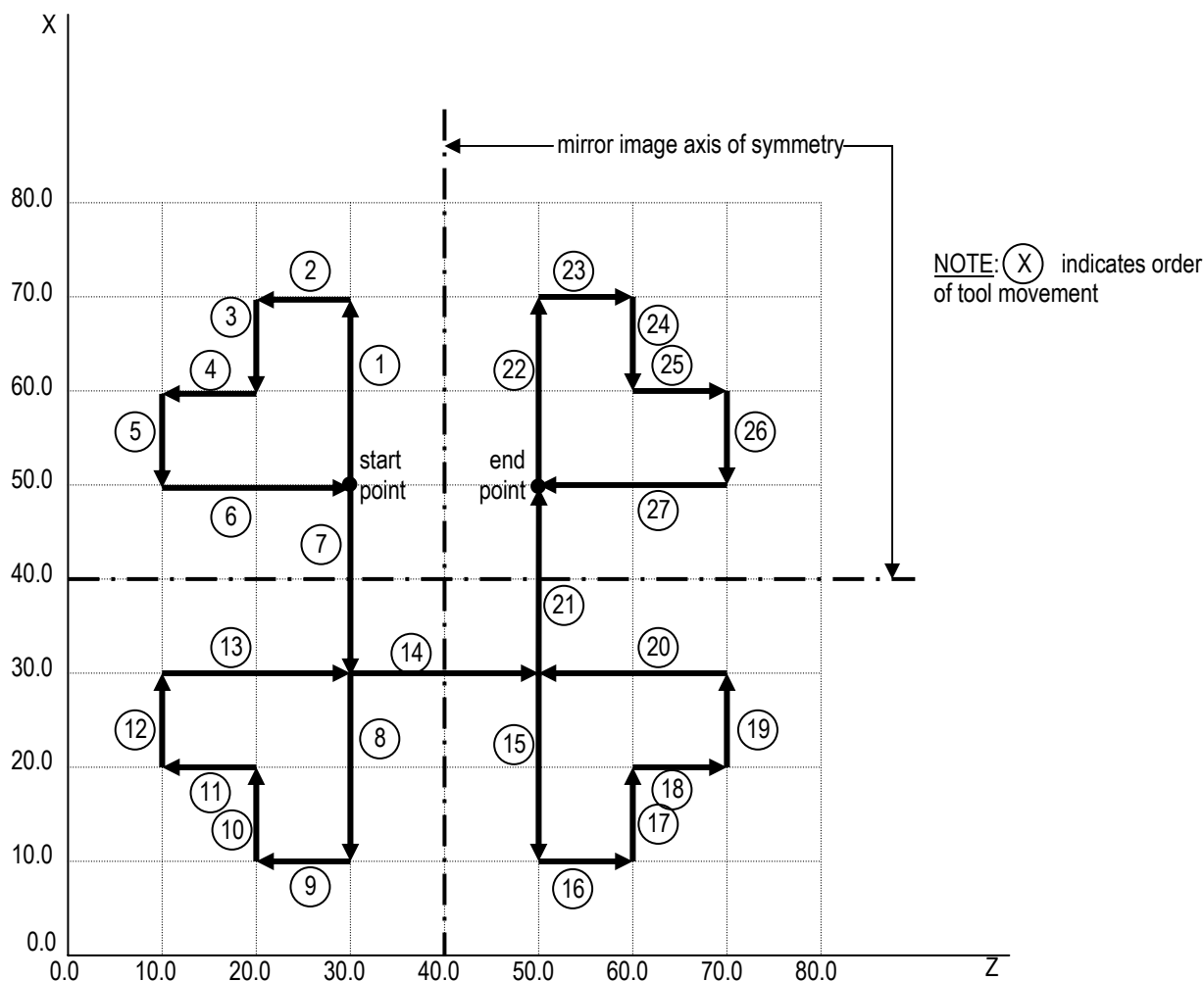


Figure 6-42: Mirror Image Function

Limitations

Tool radius compensation (G41, G42) and scaling (G51) must not be used when mirror image is on.

Notes

If the G51.1 command is specified with only one parameter (one axis), then the rotation direction is reversed for arc commands.

Other G-Codes Which Affect G50.1, G51.1

The G90 and G91 codes specifying absolute and incremental programming, respectively, affect whether the coordinates you specify will be treated as absolute coordinates (relative to the part origin) or incremental coordinates (relative to the current tool position).

Default

G50.1 is the default mode when neither G50.1 nor G51.1 has been programmed.

6.3.21 Local Coordinate System Selection (G52)

Description

This command establishes a local coordinate system for the current workpiece coordinate system, *within* that workpiece coordinate system. The coordinates set by the parameters become the new zero position for the current workpiece coordinate system.

Required Format

G52 X Y Z

Possible Parameters That Can Be Used With G52

X – X Coordinate Component (incremental value, diameter or radius programming)

Y – Y Coordinate Component (incremental value)

Z – Z Coordinate Component (incremental value)

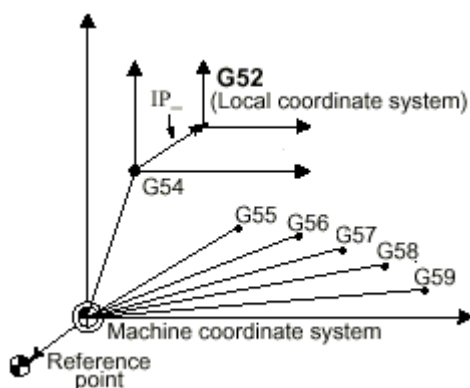


Figure 6-43: G52 Explanation

Example

G52 X5.6 Z8.6

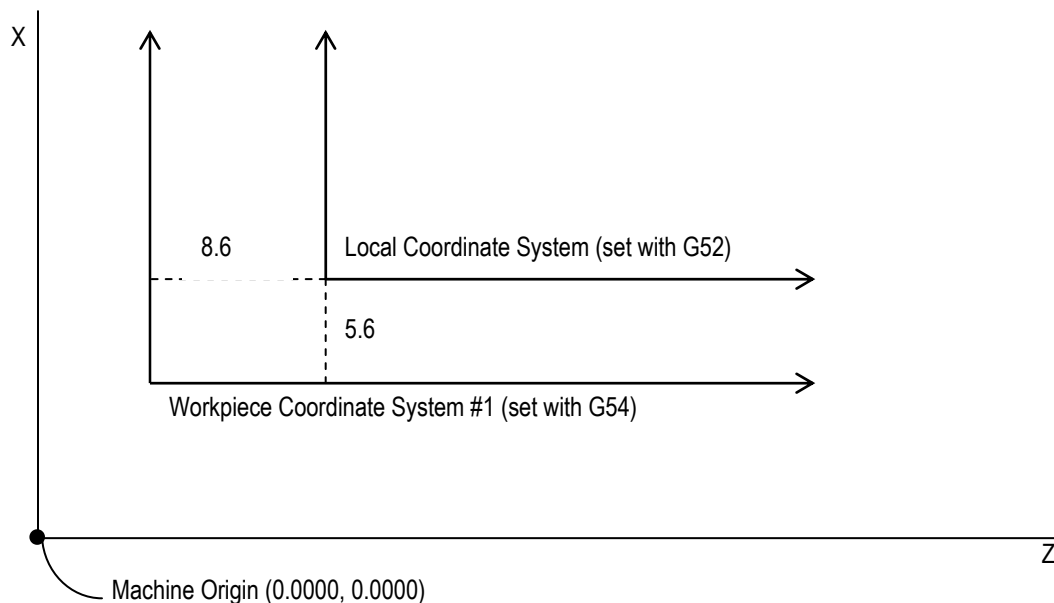


Figure 6-44: G52 Example

Limitations

- G52 will only shift the current workpiece coordinate system (not all coordinate systems).
- You can set different local coordinate systems for each workpiece coordinate system.

Variations

- G10 L2 P0 could be used to change the External Workpiece Zero Point Offset Value (to shift all coordinate systems).
- G10 L2 P1 ~ P6 could be used to change the corresponding Workpiece Zero Point Offset Value.

Notes

To cancel the local coordinate system, and return to the workpiece coordinate system, you must use the G52 command with each parameter set to zero. In other words, if you set up your local coordinate system using “G52 X5.6 Z8.6,” you must use the block “G52 X0 Z0” to cancel the local coordinate system.



This command only works with absolute coordinates. You must not try to specify incremental coordinates, or this G52 code will be ignored.

6.3.22 Machine Coordinate System Selection (G53)

Description

When this command is executed, the coordinate system is set to the machine coordinate system for this one block of code only, and the tool is moved to these coordinates (relative to the machine origin) by rapid positioning. Each consecutive block of code will be relative to the coordinate system that was in effect prior to this block of code.

Required Format

G53 X Y Z A B C

Possible Parameters That Can Be Used With G53

X – coordinate value for the X axis
 Y – coordinate value for the Y axis
 Z – coordinate value for the Z axis
 A – coordinate value for the A axis
 B – coordinate value for the B axis
 C – coordinate value for the C axis

Example

G53 X0.0 Y0.0 Z0.0 A0.0 (which will move the tool back to the machine origin)

6.3.23 Workpiece Coordinate Selection (G54-G59)

Description

These codes are used to select different coordinate systems that correspond to different workpieces. Up to six coordinate systems can be selected using G54-G59:

G54	Workpiece Coordinate System #1
G55	Workpiece Coordinate System #2
G56	Workpiece Coordinate System #3
G57	Workpiece Coordinate System #4
G58	Workpiece Coordinate System #5
G59	Workpiece Coordinate System #6

By setting the coordinate system in such a way that the zero point is set to a fixed piece on the workpiece, you make programming that workpiece very simple. You can use the same blocks of G code to cut six different workpieces, simply by moving the reference point.

Possible Parameters That Can Be Used With G54 – G59

None. The coordinates of the workpiece coordinate systems are input in Settings Mode of ServoWorks S-100M/S-120M/S-140M, and are labeled “Workpiece Zero Offset 1 (G54)” through “Workpiece Zero Offset 6 (G59).”

Example

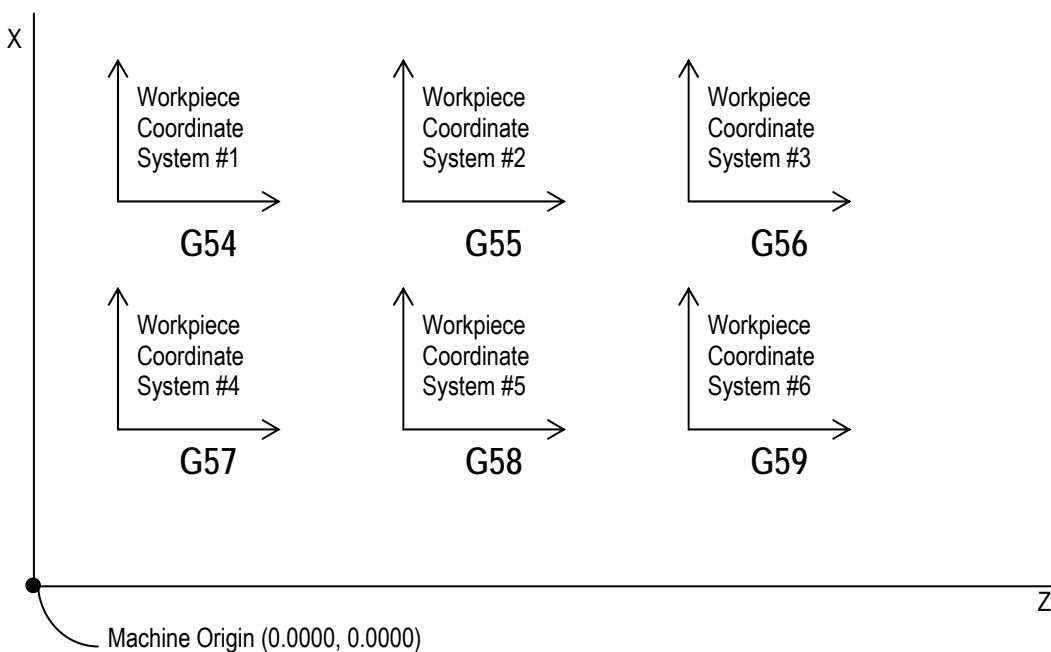


Figure 6-45: Workpiece Coordinate Selection Example

Notes

The G54 – G59 codes are used to *select* (not *set up*) different coordinate systems. These six coordinate systems need to be set up in advance of using these codes to select the coordinate systems.

Reference

You can input the workpiece coordinate systems using Settings Mode in ServoWorks S-100M/S-120M/S-140M. See the *Operator's Manual for ServoWorks S-100M, S-120M and S-140M*.



You should check the workpiece coordinate system setup on your machine tool prior to running a part program that uses the G54 – G59 codes.



If you should call a G54, G55, G56, G57, G58 or G59 code with any parameters, they will be ignored.



G54 – G59 blocks of code should appear on separate lines, with no parameters.

6.3.24 Additional Workpiece Coordinate System Selection (G54.1)

Description

Up to 58 additional different workpiece coordinate systems can be used and selected with G54.1, in addition to the six coordinate systems that can be selected using G54-G59.

Required Format

G54.1 P

Possible Parameters That Can Be Used With G54.1

P – the number of the workpiece coordinate system you want to select (1-58) – these coordinates are input in Settings Mode of ServoWorks S-100M/S-120M/S-140M

Example

G54.1 P18 ← selects the coordinates labeled “Workpiece Zero Offset 24 (G54.1 P18)”

6.3.25 Exact Stop Check Mode, Continuous Cutting Mode and Continuous Cutting Mode with Block Rollover (G61, G64, G64.1)

Description

In Exact Stop Check Mode, all move commands (such as G00, G01, G02 and G03) are executed with the tool decelerating at the end point, and with an in-position check after each move command. The next block is not executed until the in-position check is performed.

Continuous Cutting Mode is used mainly to cancel G61, Exact Stop Check Mode. In Continuous Cutting Mode, no in-position check is made at the end of each block. Instead, program execution continues automatically with the next block.

In Continuous Cutting Mode with Block Rollover, no tool deceleration occurs and no in-position check is made at the end of each block. Instead, program execution continues automatically with the next block (similar to G64/Continuous Cutting Mode).

Continuous Cutting Mode with Block Rollover uses “block rollover” technology to provide more accurate feedrate control and smoother motion, especially for high-speed, small segment contour control.

G61: Exact Stop Check Mode

G64: Continuous Cutting Mode

G64.1: Continuous Cutting Mode with Block Rollover

Detailed Description of the Block Rollover Function (G64.1)

When a single block of code cannot be completed in one servo loop/cycle time (as determined by the interpolation rate of the servo interface system), the remainder of that block of code is completed in subsequent servo loop(s). When the last servo loop for a single block of code only needs to move a small portion of the distance moved in the previous servo loop(s), but in the same cycle time, the velocity is adjusted to slow axis movement for this lesser distance in the same time frame. The block rollover function prevents this change in velocity by moving the remainder of one block to the next block and recalculating the interpolation for the next block.

For example, suppose you have a part program consisting of a series of 0.25 mm segments, and you run the machine at 6000 mm/min. 6000 mm/min is the equivalent of 0.1 mm/ms. Let's assume the servo interface platform has an interpolation rate of 1 ms, so the ServoWorks CNC Engine interpolates the trajectory every 1 ms, meaning that our CNC moves the axes 0.1 mm every interpolation cycle. If the CNC happens to execute a block of 0.25 mm, it takes three interpolation cycles (= 3 ms) to execute the block. But, as you can see, the motion behaves like this: in the first millisecond, it moves 0.1 mm; in the second millisecond, it also moves 0.1 mm; and in the third millisecond (the third cycle) it moves 0.05 mm. As a result, the velocity is not very smooth, but the trajectory can be kept accurate.

If the smoothing time (T_s) is big, then the irregular velocities (0.1 mm/ms, 0.1 mm/ms and 0.05 mm/ms) will be smoothed out, but at the cost of losing the trajectory accuracy. But, if you reduce T_s for more accuracy, then the ServoWorks CNC Engine strictly follows this unsmooth velocity (0.1 mm/ms, 0.1 mm/ms and 0.05 mm/ms), causing an unsmooth, "jerky" motion. Obviously, if the velocity commands are so unsmooth, the servos cannot catch up with these frequent changes of the velocity commands, resulting in huge servo errors, causing a huge trajectory error, which is obviously undesirable. For milling, this would create bad traces on the milling surface.

In the above case, with block rollover, the ServoWorks CNC Engine moves 0.1 mm only for two cycles, and adds the remainder of 0.05 mm to the next block (let's assume it is 0.25 mm), making a 0.3 mm segment. In this way, the ServoWorks CNC Engine can follow the trajectory more accurately while making the velocity very smooth.

Possible Parameters That Can Be Used With G61, G64, G64.1

None.

Example

G61 G01 X9.3 Z2.7 F500

Notes

Use Exact Stop Check Mode when sharp edges are required for the corners of a workpiece. Use Continuous Cutting Mode or Continuous Cutting Mode with Block Rollover at all other times.

Default

G64 is the default mode when none of the G61, G64 or G64.1 codes has been programmed.

6.3.26 Simple Macro Call (G65)

Description

Using customized macros is very similar to calling a subprogram from a main motion program, except with the added advantage that you can use variables in macros. This means that macros are more reusable than subprograms. For instance, you can have one macro for a bolt hole, and just change the variables (arguments) you assign when you call the macro (with customized macro instructions). If you were using a subprogram for programming a bolt hole, you would need a different subprogram for each size bolt hole.

The format of a macro is the same format as a subprogram:

```
O Macro number;
.
.
.
.
.
M99;
%
```

} body of macro

A simple macro call is a one-shot command in which a macro is called once using G65 (although it may be executed more than once, depending upon the L parameter, the number of times the macro is to be repeated).

Required Format

G65 P L <argument assignment>

Possible Parameters That Can Be Used With G65

P – macro number

L – number of times to repeat the execution of the macro (1 by default)

In addition to these two parameters, macros are called with argument assignments, which are not considered to be parameters.

Example

G65 P100 Z20.0 R2.5 F500

Limitations

- G65 must be specified before any argument.
- The macro program (the subroutine) must be in the same file as the part program file that calls the macro program.

Reference

See the *ServoWorks CNC Macro Programming Manual* for additional information on writing and using macros. This section deals only with calling and canceling macros.

6.3.27 Coordinate System Rotation/Coordinate System Rotation Cancel (G68, G69)

Description

The coordinate system rotation command allows you to rotate a programmed shape in two dimensions. This is particularly helpful if you have a workpiece that has been rotated at some angle from the programmed position on the machine; it allows you to modify a part program using a rotation command, rather than rewrite a part program. This coordinate system rotation command is also helpful when creating identical shapes that are rotated from a center.

G68: Coordinate system rotation

G69: Coordinate system rotation cancel

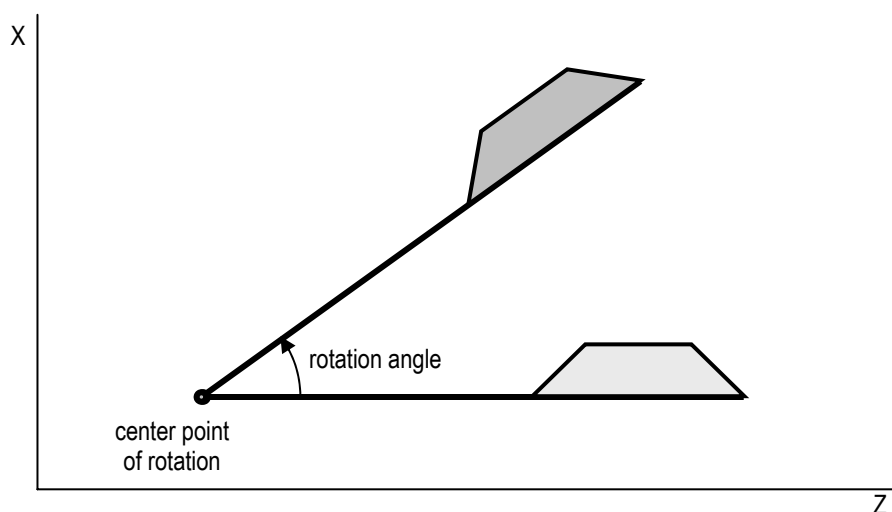


Figure 6-46: Coordinate System Rotation Function

Required Format

$$\left\{ \begin{matrix} G17 \\ G18 \\ G19 \end{matrix} \right\} G68 \left\{ \begin{matrix} X Y \\ Z X \\ Y Z \end{matrix} \right\} R;$$

.
 .
 .
 .
 } Coordinate system rotation mode is in effect [NOTE: G17, G18 and G19 must NOT be designated again]
 G69;

Possible Parameters That Can Be Used With G68, G69

X – absolute coordinate value for the X axis center point of rotation for values specified subsequent to G68

Y – absolute coordinate value for the Y axis center point of rotation for values specified subsequent to G68

Z – absolute coordinate value for the Z axis center point of rotation for values specified subsequent to G68

R – specifies the angular displacement in degrees [NOTE: a positive value denotes counterclockwise rotation; a negative value denotes clockwise rotation.]

Example

The following example demonstrates the usefulness of the coordinate system rotation function by using a subprogram for a piece of the part, and then calling that subprogram in the main program at several different angles.

Main Program:

```
%
G53
G18
G00 G90 X0. Y0.
M98 P100
G68 X0. Y0. R45
M98 P100
G68 X0. Y0. R90
M98 P100
G68 X0. Y0. R135
M98 P100
G68 X0. Y0. R180
M98 P100
G68 X0. Y0. R225
M98 P100
G68 X0. Y0. R270
M98 P100
G68 X0. Y0. R315
M98 P100
M02
%
```

Subprogram O100.dat:

```
%O100
G00 X0. Y0.
G00 G91 X10. Y40.
Y70.
X-20.
Y-70.
X20.
G90
G69
M99
%
```

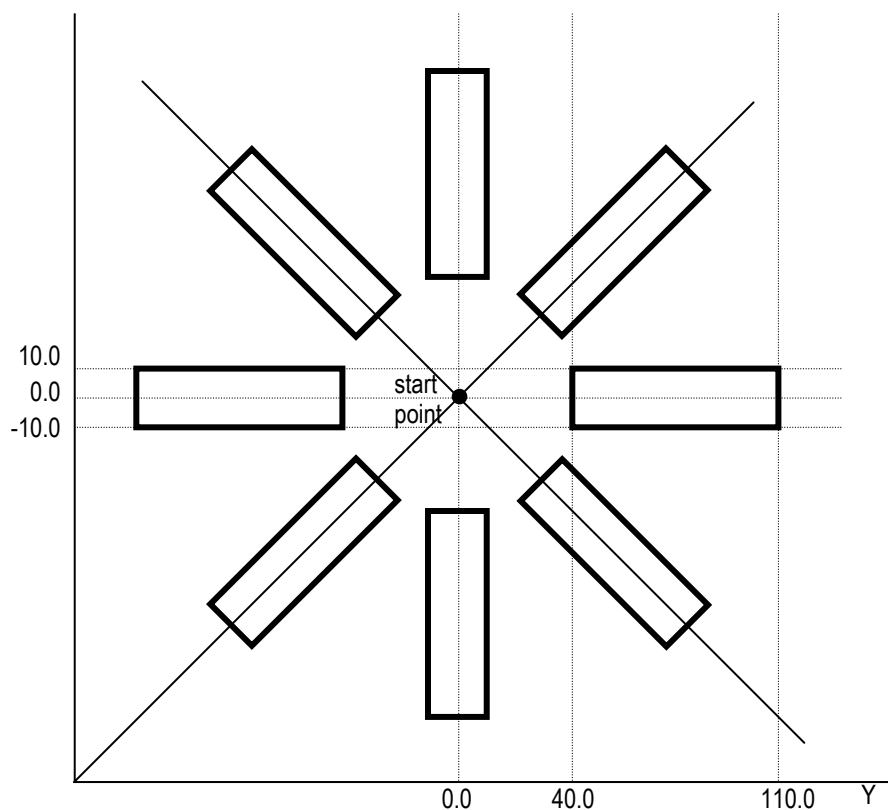


Figure 6-47: Coordinate System Rotation Example

Limitations

- G codes related to reference position return (G28, G29 and G30), and G codes for coordinate system selection (G52, G53, G54-G59, G54.1) must not be used when coordinate system rotation mode is on.
- If you are using tool radius compensation (G41, G42), AND scaling mode (G51) AND coordinate system rotation mode, the commands should be specified in the following order:

```
G51;    (starts scaling mode)
G68;    (starts coordinate system rotation mode)
.
.
G41;    (starts tool radius compensation mode)
.
.
```

- If you are using scaling mode (G51) AND coordinate system rotation mode, but NOT tool radius compensation mode (G41, G42), the commands should be specified in the following order:

```
G51;    (starts scaling mode)
G68;    (starts coordinate system rotation mode)
.
.
G69;    (ends coordinate system rotation mode)
G50;    (ends scaling mode)
```

Notes

- If X, Y and Z parameters are omitted from the command, the current position where G68 is commanded becomes the center of rotation.
- When R is not specified, R is assumed to be 0.0 degrees.
- Tool radius compensation (G41, G42), mirror image (G51.1), and scaling (G51) are executed after the coordinate system is rotated.
- If scaling mode (G51) is effective with a coordinate system rotation mode command is issued, the coordinate value of the center of rotation will be scaled, but not rotation angle R.

Default

G68 is the default mode when neither G68 nor G69 has been programmed.

6.3.28 High-Speed Peck Drilling Cycle (G73)

Description

With high-speed peck drilling, the tool feeds in to the peck distance or depth of cut, then retracts a small pre-determined distance (which is the chip-breaking process), and then feeds to the next peck, taking the tool deeper. This intermittent feeding along the Z axis allows chips to be removed from the hole easily for efficient drilling.

G73 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect.

Required Format

G73 X Y Z R Q F K;

Possible Parameters That Can Be Used With G73

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

Q – depth of cut (for each cutting feed)

F – cutting feedrate for drilling

K – number of times to repeat the entire high-speed peck drilling cycle (the default value for “K” is “1”)

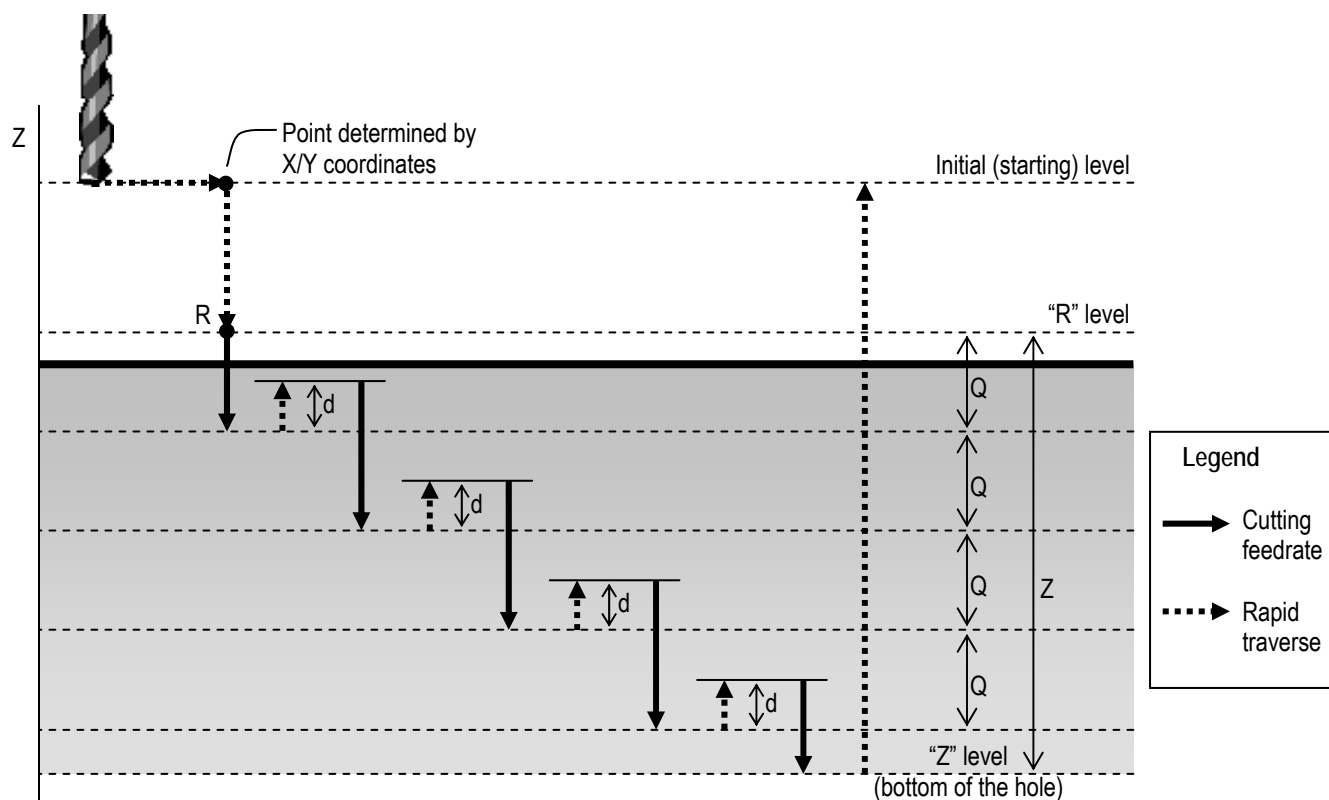


Figure 6-48: G73 Parameters (with G98 in effect)

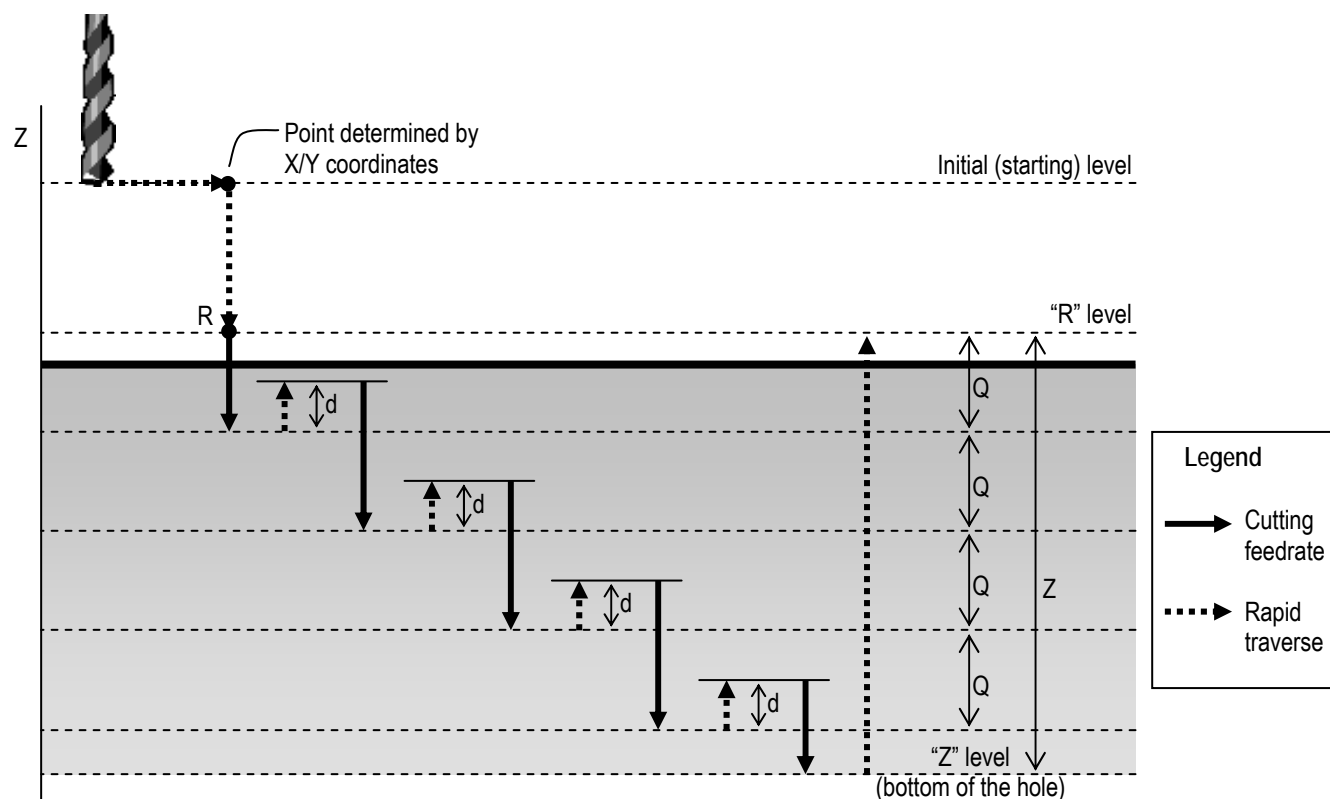


Figure 6-49: G73 Parameters (with G99 in effect)

Example of G73

M3 S500; Starts spindle rotation.
G91 G99 G73 X200 Y200 Z-100 R-20 Q10 F200;
Positions the drill at X=200, Y=200; drills hole #1 with a series of cutting feeds; returns to R level.
Y250; Moves the drill position to X=200, Y=250; drills hole #2 with a series of cutting feeds; returns to R level.
Y300; Moves the drill position to X=200, Y=300; drills hole #3 with a series of cutting feeds; returns to R level.
G98 Y350; Moves the drill position to X=200, Y=350; drills hole #4 with a series of cutting feeds; returns to initial level.

Limitations

- To change the drilling axis, you must first cancel the G73 canned cycle.
- If the G73 block of code does not contain X, Y, Z, R, or any other axes, the G73 peck drilling cycle is not performed.
- Q and R parameters cannot be stored as modal data when they are specified in a block that does not perform drilling.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G73, the G73 block is not executed.
- Tool offsets are ignored in a G73 peck drilling cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- **Implementation of G73:**
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - For the first cutting feed, the tool drills in the depth of cut at the cutting feedrate, and then retracts the specified distance in rapid traverse.
 - For the second and subsequent cutting feeds, the tool drills in the depth of cut beyond the end of the previous cutting feed, and then retracts the specified distance in rapid traverse.
 - At some point, the bottom of the hole is reached.
 - The tool retracts to the initial or R level (depending upon whether G98 or G99 is in effect) at rapid traverse.
- All rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).
- “d” is the retract vector, a parameter specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 3.3.2: Retract Vector* in the *Reference Manual for ServoWorks CNC Parameters and Functions*). The smaller the “d” value that still allows chips to be removed, the more efficient the cycle.
- “Q,” the depth of cut for each cutting feed, must be specified as an incremental value.
- If “K” is not specified, the G73 peck drilling cycle is only performed once.
- When G73 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins drilling.
- When “K” is used to specify multiple G73 peck drilling cycles, any M code specified in the G73 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G73 peck drilling cycle, the tool length offset is applied during the first positioning operation (moving the drill to point R).

Other G-Codes Which Affect G73

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G73 cycle.
- The G73 high-speed peck drilling cycle can be cancelled by G80 (canned cycle cancel).

CAUTION

Before specifying a G73 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).

CAUTION

The Q parameter is stored as modal data within canned cycles. Specify this value carefully, because it is also used as the spindle/tool shift amount for G76 and G87, and the depth of cut for G83.

CAUTION

You must specify a positive value for Q: a negative value will be ignored.

6.3.29 Counter Tapping Cycle (G74)

Description

Tapping is the operation of cutting threads inside a previously drilled hole using a tap. (The hole must be drilled smaller than the root diameter of the thread.) With a counter tapping cycle, the spindle turns counterclockwise until it reaches the bottom of the hole, at which point it begins to rotate clockwise and retract, forming a reverse thread.

G74 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect. When it reaches either the initial point or the R point, the spindle switches from clockwise back to counterclockwise.

Required Format

G74 X Y Z R P F K;

Possible Parameters That Can Be Used With G74

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

P – dwell time (in milliseconds) at the bottom of the hole

F – cutting feedrate for tapping

K – number of times to repeat the entire counter tapping cycle (the default value for “K” is “1”)

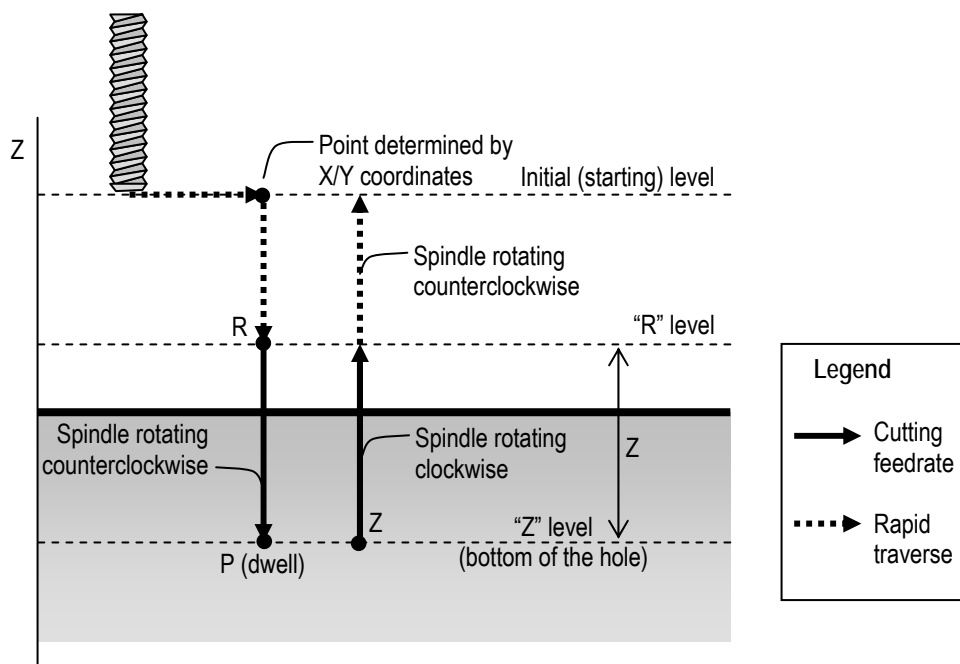


Figure 6-50: G74 Parameters (with G98 in effect)

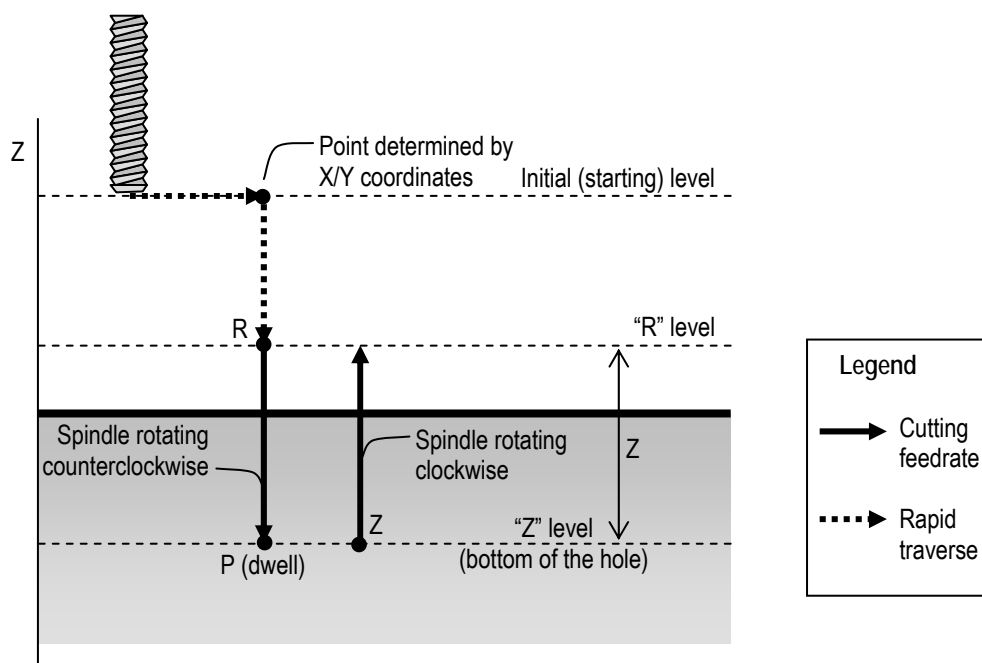


Figure 6-51: G74 Parameters (with G99 in effect)

Example of G74

M4 S500; Starts spindle rotation in the counterclockwise direction.
G90 G99 G74 X200 Y200 Z-100 R-20 P500 F200;
Positions the spindle at X=200, Y=200; taps hole #1 with spindle rotation in the counterclockwise direction; dwells for 500 ms; returns to R level with spindle rotation in the clockwise direction.
Y250;
Moves the spindle position to X=200, Y=250; taps hole #2 with spindle rotation in the counterclockwise direction; dwells for 500 ms; returns to R level with spindle rotation in the clockwise direction.
Y300;
Moves the spindle position to X=200, Y=300; taps hole #3 with spindle rotation in the counterclockwise direction; dwells for 500 ms; returns to R level with spindle rotation in the clockwise direction.
G98 Y350;
Moves the spindle position to X=200, Y=350; taps hole #4 with spindle rotation in the counterclockwise direction; dwells for 500 ms; returns to initial level with spindle rotation in the clockwise direction.

Limitations

- To change the drilling axis, you must first cancel the G74 canned cycle.
- If the G74 block of code does not contain X, Y, Z, R, or any other axes, the G74 counter tapping cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform tapping.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G74, the G74 block is not executed.
- Tool offsets are ignored in a G74 counter tapping cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G74:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The spindle rotates counterclockwise from point R to point Z (the bottom of the hole) at cutting feedrate.
 - At the bottom of the hole, the spindle dwells as specified, and changes the direction of rotation from counterclockwise to clockwise.
 - The spindle retracts at the cutting feedrate from the bottom of the hole to point R while rotating clockwise, creating a reverse thread.
 - If G99 is in effect, the cycle is complete at this point. If G98 is in effect, the spindle retracts to the initial point at rapid traverse, while still rotating.
- All rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).
- When G74 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins tapping.
- When “K” is used to specify multiple G74 counter tapping cycles, any M code specified in the G74 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G74 counter tapping cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).



CAUTION

Before specifying a G74 cycle, start spindle rotation in the counterclockwise direction with an M code (miscellaneous function) or an S code (spindle function).



CAUTION

A feed hold command issued while G74 is executing does not stop the machine until the return operation of the G74 counter tapping cycle is complete.



CAUTION

Feedrate overrides are ignored for G74 counter tapping cycles.

Other G-Codes Which Affect G74

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G74 cycle.
- The G74 counter tapping cycle can be cancelled by G80 (canned cycle cancel).

6.3.30 Counter Rigid Tapping Cycle (G74)

Description

G74 can be used for both a normal canned counter tapping cycle, or for a counter rigid tapping cycle. In order to use G74 for rigid tapping, an “M29” code must be coded before G74 to enable Rigid Mode. [“M29” is for “Spindle Rigid Mode On,” as defined in the default LadderWorks PLC sequence program – SoftServo_M.lad – included with your ServoWorks CNC product.]

Tapping is the operation of cutting threads inside a previously drilled hole using a tap. (The hole must be drilled smaller than the root diameter of the thread.) With a counter tapping cycle, the spindle turns counterclockwise until it reaches the bottom of the hole, at which point it begins to rotate clockwise and retract, forming a reverse thread.

“Rigid Mode” refers to a mode whereby the spindle motor is controlled as if it were a servo motor, to allow a faster counter tapping cycle. With this synchronized counter tapping, the machine spindle rotates in synchronization with the feed rate to match the exact pitch of the thread being cut or formed. At this time, rigid tapping is limited to the servo spindle axis type (a servo motor and drive used as a spindle); it is not available for an inverter spindle at this time.

G74 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect. When it reaches either the initial point or the R point, the spindle stops.

Required Format

M29 S
G74 X Y Z R P F K;

OR

M29 S G74 X Y Z R P F K;

Possible Parameters That Can Be Used With G74

X – coordinate value for the X axis (hole position data)
Y – coordinate value for the Y axis (hole position data)
Z – distance to the bottom of the hole from point R and the position of the bottom of the hole
R – distance to point R level from the initial level
P – dwell time (in milliseconds) at the bottom of the hole and at point R when a return is made
F – cutting feedrate for tapping
K – number of times to repeat the entire counter tapping cycle (the default value for “K” is “1”)

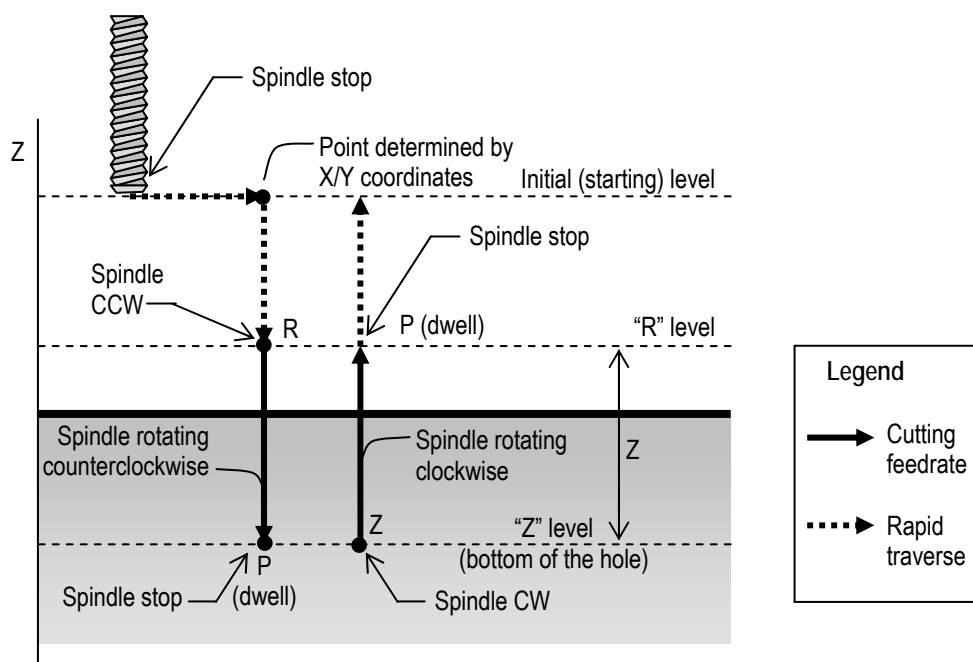


Figure 6-52: G74 Parameters (with G98 in effect)

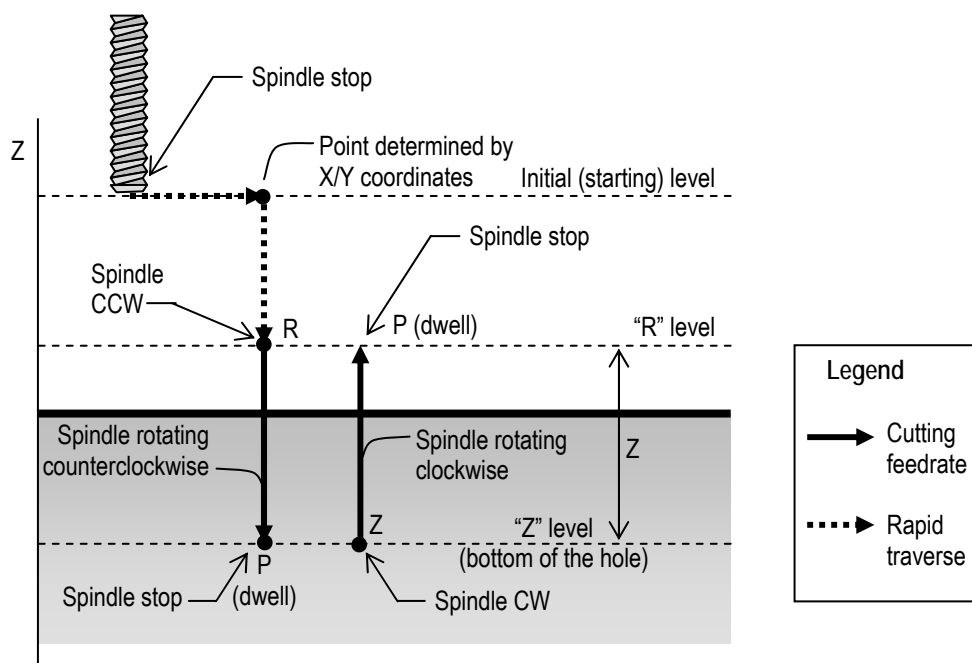


Figure 6-53: G74 Parameters (with G99 in effect)

Example of G74

M29 S100; Establishes rigid tapping mode and stops the spindle. Control is switched from open loop (S axis) spindle control (velocity control) to closed loop (C axis) spindle control (position control). Performs a spindle orientation operation.

G90 G95 G99 G74 X200 Y200 Z-100 R-20 P500 F200;
 Positions the spindle at X=200, Y=200; starts spindle rotation in the counterclockwise direction; taps hole #1 with spindle rotation in the counterclockwise direction; the spindle stops upon reaching the bottom of the hole; dwells for 500 ms; returns to R level with spindle rotation in the clockwise direction; spindle stops; dwells for 500 ms.

Y250; Moves the spindle position to X=200, Y=250; taps hole #2 with spindle rotation in the counterclockwise direction; the spindle stops upon reaching the bottom of the hole; dwells for 500 ms; returns to R level with spindle rotation in the clockwise direction; spindle stops; dwells for 500 ms.

Y300; Moves the spindle position to X=200, Y=300; taps hole #3 with spindle rotation in the counterclockwise direction; the spindle stops upon reaching the bottom of the hole; dwells for 500 ms; returns to R level with spindle rotation in the clockwise direction; spindle stops; dwells for 500 ms.

G98 Y350; Moves the spindle position to X=200, Y=350; taps hole #4 with spindle rotation in the counterclockwise direction; the spindle stops upon reaching the bottom of the hole; dwells for 500 ms; returns to R level with spindle rotation in the clockwise direction; spindle stops; returns to initial level with spindle stopped.

G80; Cancels the rigid tapping cycle. Control is switched back to open loop (S axis) spindle control (velocity control).

Limitations

- Counter rigid tapping is limited to one axis type: servo spindle (a servo motor and drive used as a spindle).
- To change the drilling axis, you must first cancel the G74 canned cycle.
- If the G74 block of code does not contain X, Y, Z, R, or any other axes, the G74 counter tapping cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform tapping.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G74, the G74 block is not executed.
- Tool offsets are ignored in a G74 counter tapping cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G74:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The spindle rotates counterclockwise from point R to point Z (the bottom of the hole) at cutting feedrate.
 - At the bottom of the hole, the spindle stops, dwells as specified, and the spindle starts clockwise rotation.
 - The spindle retracts at the cutting feedrate from the bottom of the hole to point R, while rotating clockwise, creating a reverse thread.
 - The spindle stops, and dwells as specified.
 - If G99 is in effect, the cycle is complete at this point. If G98 is in effect, the spindle retracts to the initial point at rapid traverse, with the spindle stopped.
- The thread lead (the distance between threads, measured on a plane parallel to the centerline of the threaded part) varies depending upon whether G94 or G95 is in effect.
 - In G94 feed per minute mode, thread lead = feedrate / spindle speed
Example: (100 mm/min) / (50 rot/min) = 2.0 mm thread lead
 - In G95 feed per revolution mode, thread lead = feedrate.
- All rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).

- When G74 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins tapping.
- When “K” is used to specify multiple G74 counter tapping cycles, any M code specified in the G74 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G74 counter tapping cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).

**CAUTION**

Before specifying a G74 cycle, start spindle rotation in the counterclockwise direction with M29 (miscellaneous function for rigid mode) and an S code (spindle function).

**CAUTION**

A feed hold command issued while G74 is executing does not stop the machine until the return operation of the G74 counter tapping cycle is complete.

**CAUTION**

Feedrate overrides are ignored for G74 counter tapping cycles.

Other G-Codes Which Affect G74

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G74 cycle.
- The G94 and G95 codes specifying feed per minute or feed per revolution mode, respectively, affect how the feedrates you specify are processed and how the machine spindle is synchronized with the Z axis.
- The G74 counter rigid tapping cycle can be cancelled by G80 (canned cycle cancel).

6.3.31 Fine Boring Cycle (G76)

Description

Boring improves the surface finish and cuts the parts to size – the tool follows its own path, fixing any deviation done by drilling. This fine boring cycle bores the hole precisely with a tool attached to the spindle – the spindle turns clockwise until it reaches the bottom of the hole, where it performs an oriented spindle stop. The tool is moved away from the machined surface of the workpiece, and the spindle and tool are retracted.

G76 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect.

Required Format

G76 X Y Z R Q P F K:

Possible Parameters That Can Be Used With G76

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

O – tool shift amount at the bottom of the hole

P – dwell time (in milliseconds) at the bottom of the hole

F – cutting feedrate for boring

K – number of times to repeat the entire fine boring cycle (the default value for “K” is “1”)

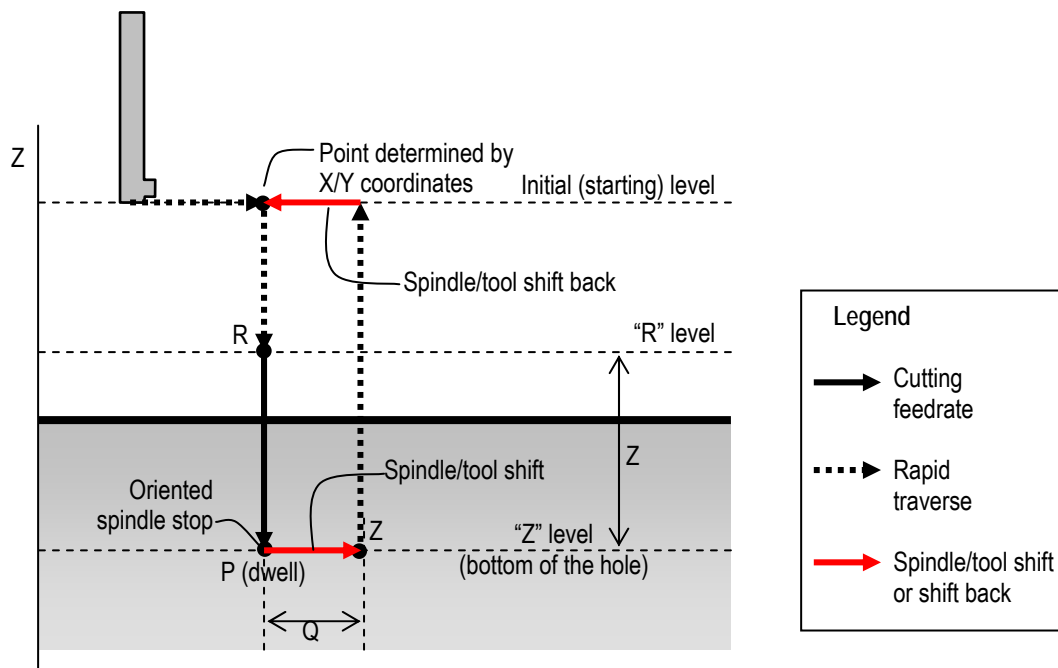


Figure 6-54: G76 Parameters (with G98 in effect)

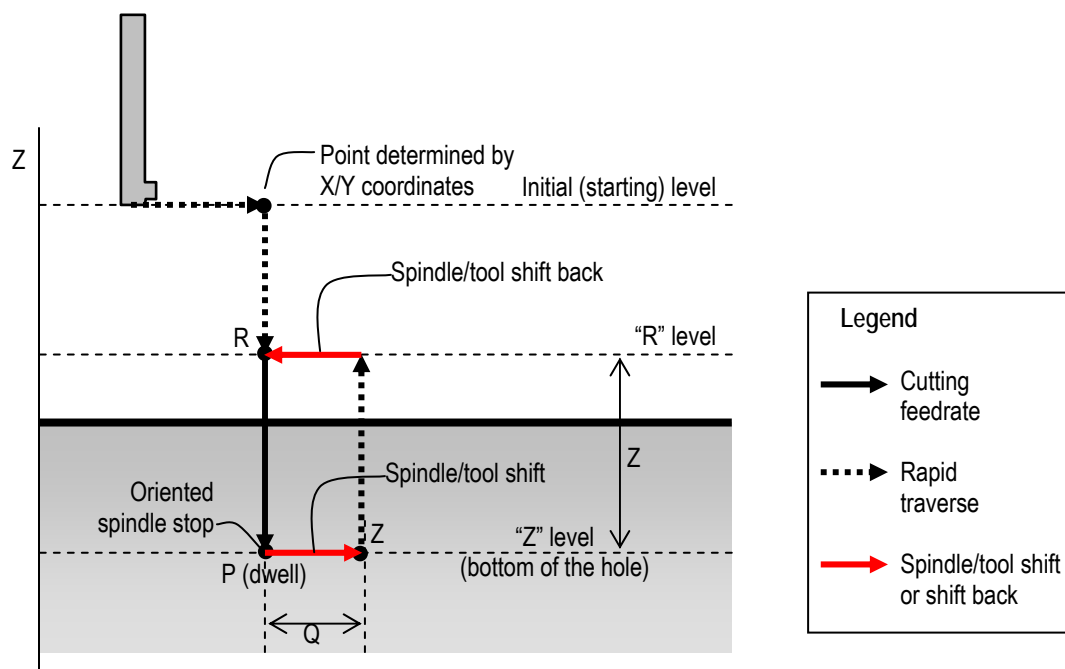


Figure 6-55: G76 Parameters (with G99 in effect)

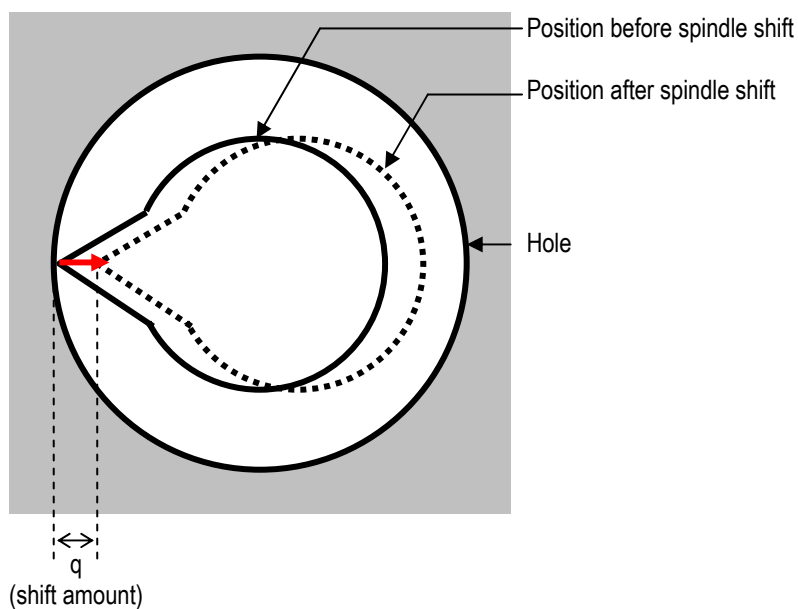


Figure 6-56: G76 Oriented Spindle Stop, Spindle/Tool Shift within Hole

Example of G76

M4 S500; Starts spindle rotation.
 G90 G99 G76 X200 Y200 Z-100 R-20 Q4 P100 F200;
 Positions the spindle at X=200, Y=200; bores hole #1; dwells for 100 ms; shifts the spindle/tool 4mm; returns to R level.
 Y250; Moves the spindle position to X=200, Y=250; bores hole #2; dwells for 100 ms; shifts the spindle/tool 4mm; returns to R level.
 Y300; Moves the spindle position to X=200, Y=300; bores hole #3; dwells for 100 ms; shifts the spindle/tool 4mm; returns to R level.
 G98 Y350; Moves the spindle position to X=200, Y=350; bores hole #4; dwells for 100 ms; shifts the spindle/tool 4mm; returns to initial level.

Limitations



- To change the boring axis, you must first cancel the G76 canned cycle.
- If the G76 block of code does not contain X, Y, Z, R, or any other axes, the G76 fine boring cycle is not performed.
- Q must be specified as a positive value. If you assign a negative value to Q, the negative sign is ignored, and the value is treated as positive. You set the “Shift Direction” in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M, as either “No Shift,” “Shift +” or “Shift –.” [See *Section 3.3.1: Shift Direction in the Reference Manual for ServoWorks CNC Parameters and Functions.*]
- Q and R parameters cannot be stored as modal data when they are specified in a block that does not perform boring.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G76, the G76 block is not executed.

Notes

- Implementation of G76:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The spindle bores from point R to point Z (the bottom of the hole).
 - At the bottom of the hole, the spindle dwells the specified time. Then the spindle is stopped at a fixed rotation position. The tool is moved in the direction opposite to the tool tip (as shown in Figure 6-56) prior to retraction, to prevent any damage to the machined surface.
 - The tool retracts with rapid traverse to either the initial level or to the R level, depending upon whether G98 or G99 is in effect.
 - The spindle/tool shifts back before proceeding to the next block.
- All rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate in the Reference Manual for ServoWorks CNC Parameters and Functions.*).
- When G76 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins boring.
- When “K” is used to specify multiple G76 fine boring cycles, any M code specified in the G76 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G76 fine boring cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).

Other G-Codes Which Affect G76

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G76 cycle.
- The G76 fine boring cycle can be cancelled by G80 (canned cycle cancel).

 CAUTION
<p>The Q parameter is stored as modal data within canned cycles. Specify this value carefully, because it is also used as the depth of cut for G73 and G83, and the spindle/tool shift amount for G87.</p>
 CAUTION
<p>Before specifying a G76 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).</p>

6.3.32 Canned Cycle Cancel (G80)

Description

Cancels all canned cycles: G73 high-speed peck drilling cycle, G74 counter tapping cycle, G74 rigid counter tapping cycle, G76 fine boring cycle, G81 drilling cycle (spot boring), G82 drilling cycle (dwell), G83 peck drilling cycle, G84 tapping cycle, G84 rigid tapping cycle, G85 boring cycle, G86 boring cycle (spindle stop), G87 back boring cycle and G89 boring cycle (dwell).

Clears the values for parameters R and Z, and all other drilling data. In incremental mode, R will be set to zero, and Z will be set to zero.

Required Format

G80;

Possible Parameters That Can Be Used With G80

None.

6.3.33 Drilling Cycle, Spot Boring (G81)

Description

G81 is used for a normal drilling cycle where the tool drills straight to the bottom of the hole at a continuous cutting feedrate, then retracts.

G81 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect.

Required Format

G81 X Y Z R F K;

Possible Parameters That Can Be Used With G81

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

F – cutting feedrate for drilling

K – number of times to repeat the entire drilling cycle (the default value for “K” is “1”)

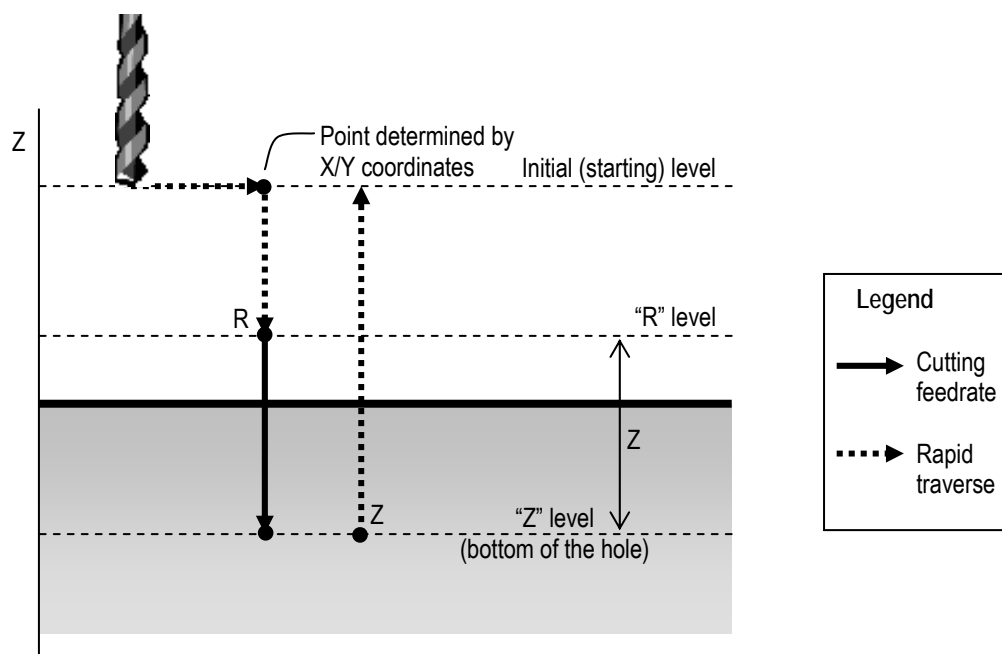


Figure 6-57: G81 Parameters (with G98 in effect)

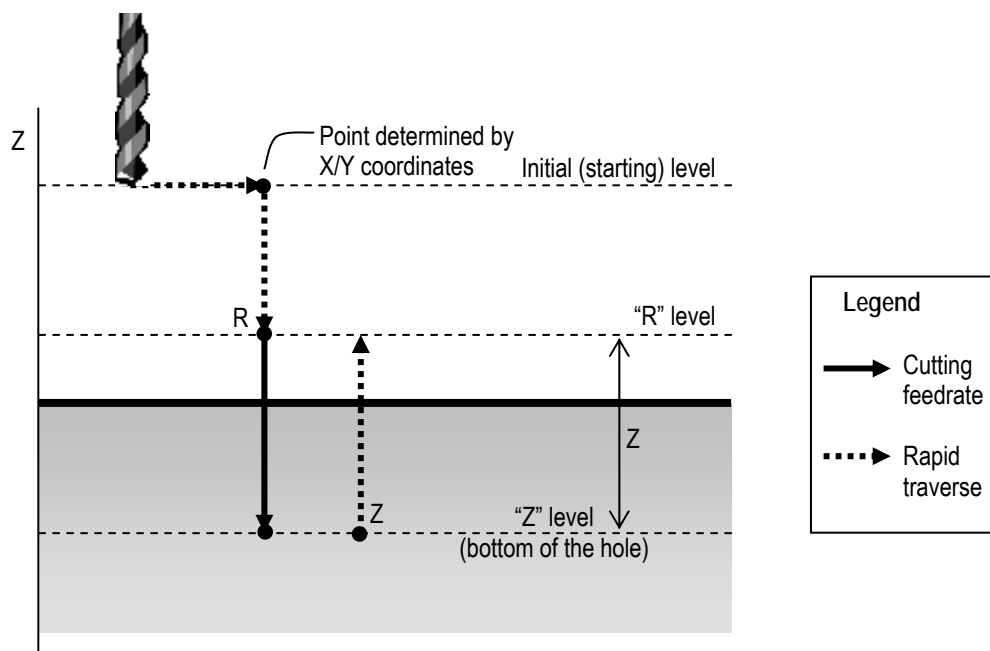


Figure 6-58: G81 Parameters (with G99 in effect)

Example of G81

M4 S500; Starts spindle rotation.
G90 G99 G81 X200 Y200 Z-100 R-20 F200;
Positions the drill at X=200, Y=200; drills hole #1; returns to R level.
Y250;
Moves the drill position to X=200, Y=250; drills hole #2; returns to R level.
Y300;
Moves the drill position to X=200, Y=300; drills hole #3; returns to R level.
G98 Y350;
Moves the drill position to X=200, Y=350; drills hole #4; returns to initial level.

Limitations

- To change the drilling axis, you must first cancel the G81 canned cycle.
- If the G81 block of code does not contain X, Y, Z, R, or any other axes, the G81 drilling cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform drilling.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G81, the G81 block is not executed.
- Tool offsets are ignored in a G81 drilling cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G81:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The tool drills from point R to point Z (the bottom of the hole).
 - The tool retracts at rapid traverse to either the initial level or to the R level, depending upon whether G98 or G99 is in effect.
- All rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see Section 4.3: *Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).
- When G81 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins drilling.

- When “K” is used to specify multiple G81 drilling cycles, any M code specified in the G81 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G81 drilling cycle, the tool length offset is applied during the first positioning operation (moving the drill to point R).

Other G-Codes Which Affect G81

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G81 cycle.
- The G81 drilling cycle (spot boring) can be cancelled by G80 (canned cycle cancel).

**CAUTION**

Before specifying a G81 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).

6.3.34 Drilling Cycle (Dwell) (G82)

Description

G82 is used for a normal drilling cycle where the tool drills straight to the bottom of the hole at a continuous cutting feedrate, dwells for a specified length of time, then retracts. The G82 drilling cycle with dwell performs more accurate drilling of holes with respect to hole depth than the G81 drilling cycle without dwell.

G82 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect.

Required Format

G82 X Y Z R P F K;

Possible Parameters That Can Be Used With G82

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

P – dwell time (in milliseconds) at the bottom of the hole

F – cutting feedrate for drilling

K – number of times to repeat the entire drilling cycle (the default value for “K” is “1”)

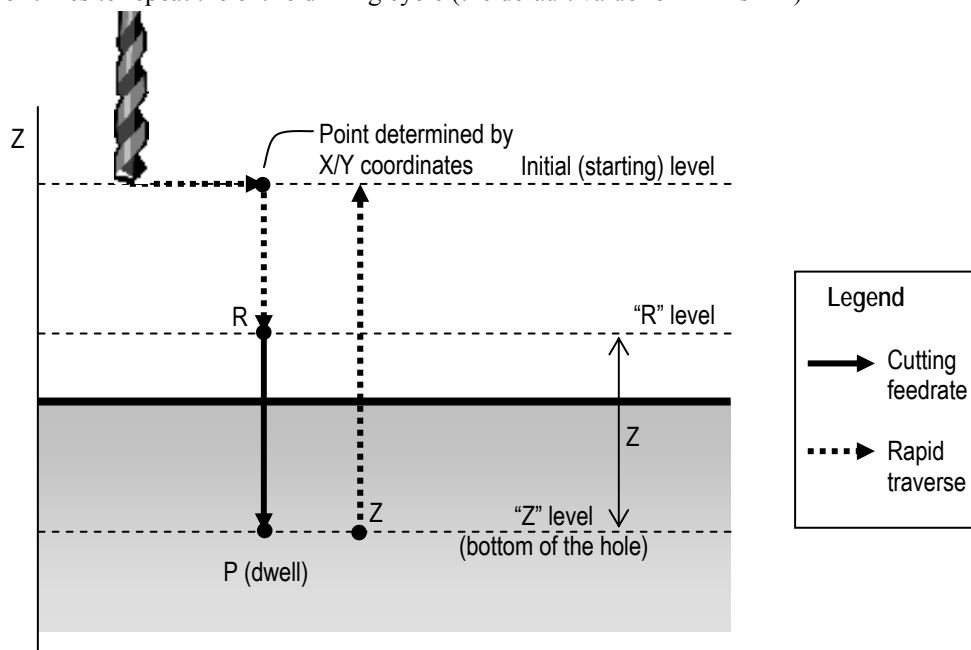


Figure 6-59: G82 Parameters (with G98 in effect)

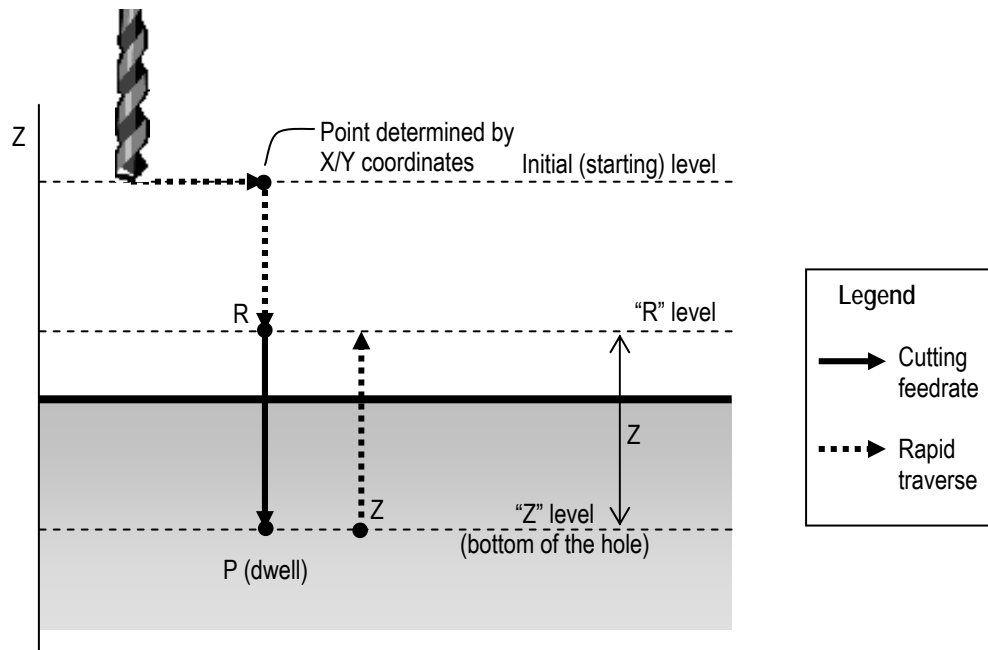


Figure 6-60: G82 Parameters (with G99 in effect)

Example of G82

M4 S500;	Starts spindle rotation.
G90 G99 G82 X200 Y200 Z-100 R-20 P500 F200;	Positions the drill at X=200, Y=200; drills hole #1; dwells for 500 ms; returns to R level.
Y250;	Moves the drill position to X=200, Y=250; drills hole #2; dwells for 500 ms; returns to R level.
Y300;	Moves the drill position to X=200, Y=300; drills hole #3; dwells for 500 ms; returns to R level.
G98 Y350;	Moves the drill position to X=200, Y=350; drills hole #4; dwells for 500 ms; returns to initial level.

Limitations

- To change the drilling axis, you must first cancel the G82 canned cycle.
- If the G82 block of code does not contain X, Y, Z, R, or any other axes, the G82 drilling cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform drilling.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G82, the G82 block is not executed.
- Tool offsets are ignored in a G82 drilling cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G82:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The tool drills from point R to point Z (the bottom of the hole).
 - At the bottom of the hole, the tool dwells as specified.
 - The tool retracts at rapid traverse to either the initial level or to the R level, depending upon whether G98 or G99 is in effect.
- All rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).

- When G82 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins drilling.
- When “K” is used to specify multiple G82 drilling cycles, any M code specified in the G82 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G82 drilling cycle, the tool length offset is applied during the first positioning operation (moving the drill to point R).

Other G-Codes Which Affect G82

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G82 cycle.
- The G82 drilling cycle (dwell) can be cancelled by G80 (canned cycle cancel).

**CAUTION**

Before specifying a G82 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).

6.3.35 Peck Drilling Cycle (G83)

Description

With peck drilling, the tool feeds in to the peck distance or depth of cut, then retracts to the point R level to remove shavings from the hole, and then feeds to the next peck, taking the tool deeper. In the second and all subsequent cutting feeds, cutting feed is performed from a distance “d” above where the last cutting feed ended, to a distance “Q” below where the last cutting feed ended.

G83 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect.

Required Format

G83 X Y Z R Q F K;

Possible Parameters That Can Be Used With G83

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

Q – depth of cut (for each cutting feed)

F – cutting feedrate for drilling

K – number of times to repeat the entire peck drilling cycle (the default value for “K” is “1”)

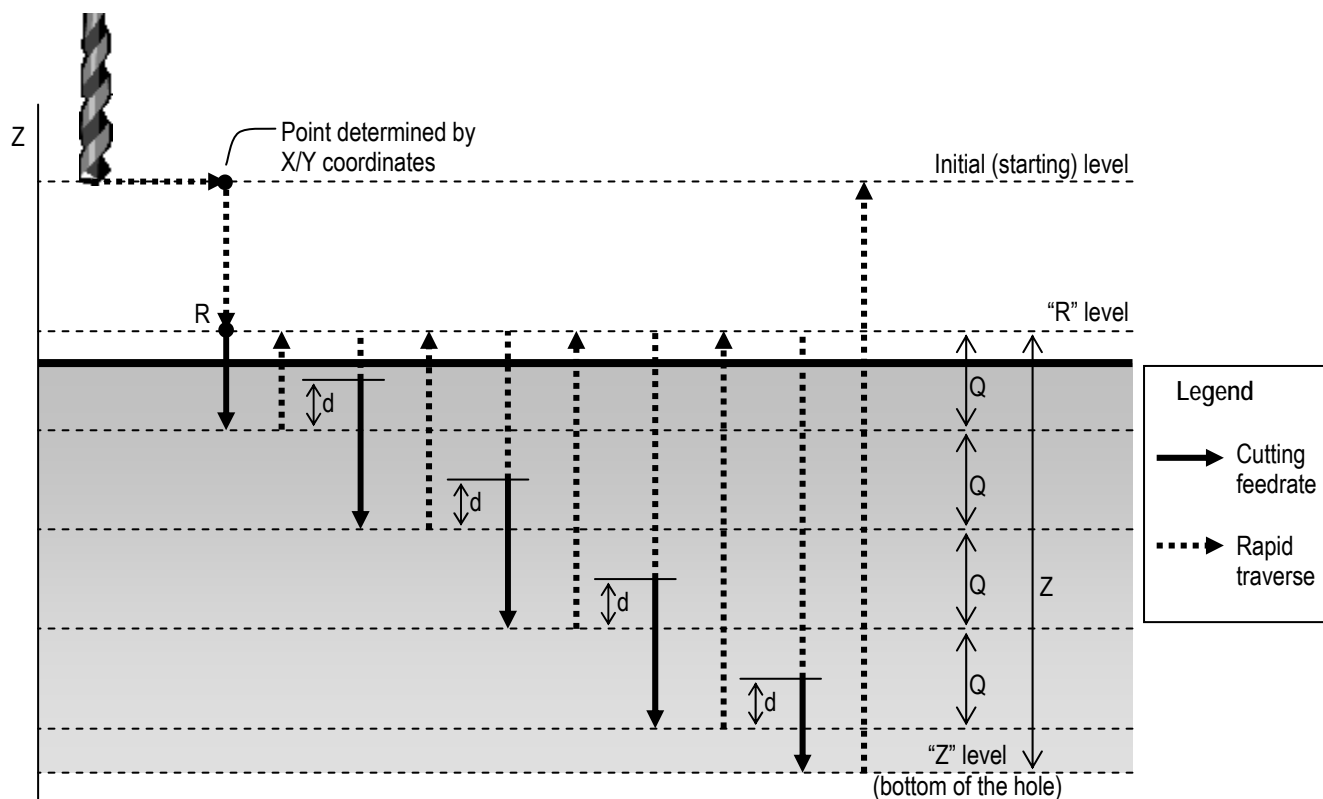


Figure 6-61: G83 Parameters (with G98 in effect)

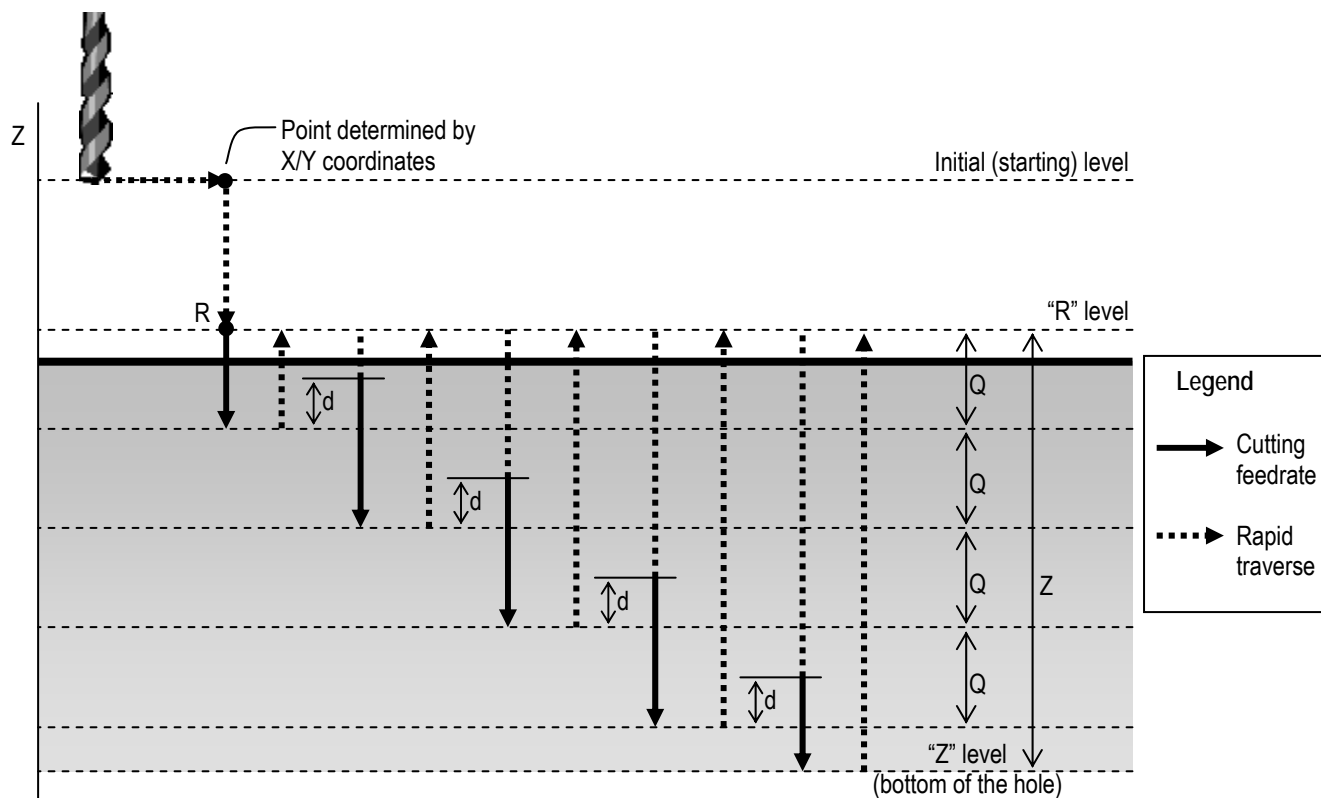


Figure 6-62: G83 Parameters (with G99 in effect)

Example of G83

M3 S500; Starts spindle rotation.
G91 G99 G83 X200 Y200 Z-100 R-20 Q10 F200; Positions the drill at X=200, Y=200; drills hole #1 with a series of cutting feeds; returns to R level.
Y250; Moves the drill position to X=200, Y=250; drills hole #2 with a series of cutting feeds; returns to R level.
Y300; Moves the drill position to X=200, Y=300; drills hole #3 with a series of cutting feeds; returns to R level.
G98 Y350; Moves the drill position to X=200, Y=350; drills hole #4 with a series of cutting feeds; returns to initial level.

Limitations

- To change the drilling axis, you must first cancel the G83 canned cycle.
- If the G83 block of code does not contain X, Y, Z, R, or any other axes, the G83 peck drilling cycle is not performed.
- Q and R parameters cannot be stored as modal data when they are specified in a block that does not perform drilling.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G83, the G83 block is not executed.
- Tool offsets are ignored in a G83 peck drilling cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G83:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - For the first cutting feed, the tool drills from point R down a distance Q, and then retracts back to point R at rapid traverse.
 - For the second and all following cutting feeds, the tool moves at rapid traverse from point R down a distance “d” above the end position of the previous cutting feed, drills from a distance “d” above the end position of the previous cutting feed to a distance Q below the end position of the previous cutting feed, then retracts back to point R at rapid traverse.
 - At some point, the bottom of the hole is reached.
 - The tool retracts to the initial or R level (depending upon whether G98 or G99 is in effect) at rapid traverse.
- All rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate in the Reference Manual for ServoWorks CNC Parameters and Functions*).
- “d” is the retract vector, a parameter specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 3.3.2: Retract Vector in the Reference Manual for ServoWorks CNC Parameters and Functions*). The smaller the “d” value that still allows chips to be removed, the more efficient the cycle.
- “Q,” the depth of cut for each cutting feed, must be specified as an incremental value.
- If “K” is not specified, the G83 peck drilling cycle is only performed once.
- When G83 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins drilling.
- When “K” is used to specify multiple G83 peck drilling cycles, any M code specified in the G83 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G83 peck drilling cycle, the tool length offset is applied during the first positioning operation (moving the drill to point R).

Other G-Codes Which Affect G83

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G83 cycle.
- The G83 peck drilling cycle can be cancelled by G80 (canned cycle cancel).



Before specifying a G83 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).



The Q parameter is stored as modal data within canned cycles. Specify this value carefully, because it is also used as the spindle/tool shift amount for G76 and G87, and the depth of cut for G73.



You must specify a positive value for Q: a negative value will be ignored.

6.3.36 Tapping Cycle (G84)

Description

Tapping is the operation of cutting threads inside a previously drilled hole using a tap. (The hole must be drilled smaller than the root diameter of the thread.) With a tapping cycle, the spindle turns clockwise until it reaches the bottom of the hole, at which point it begins to rotate counterclockwise and retract, forming a thread.

G84 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect. When it reaches either the initial point or the R point, the spindle switches from counterclockwise back to clockwise.

Required Format

G84 X Y Z R P F K;

Possible Parameters That Can Be Used With G84

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

P – dwell time (in milliseconds) at the bottom of the hole

F – cutting feedrate for tapping

K – number of times to repeat the entire tapping cycle (the default value for “K” is “1”)

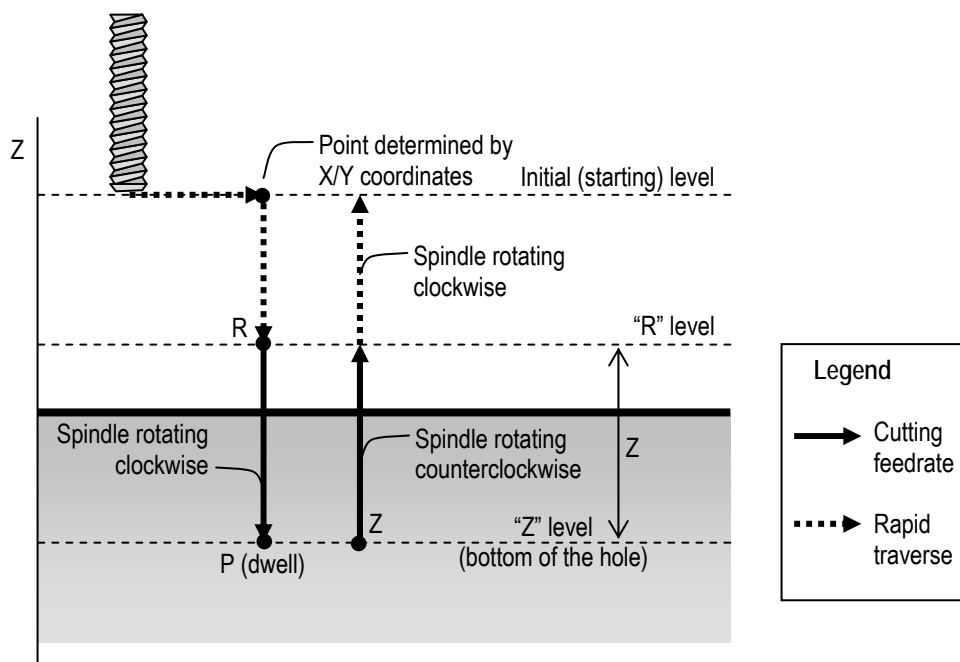


Figure 6-63: G84 Parameters (with G98 in effect)

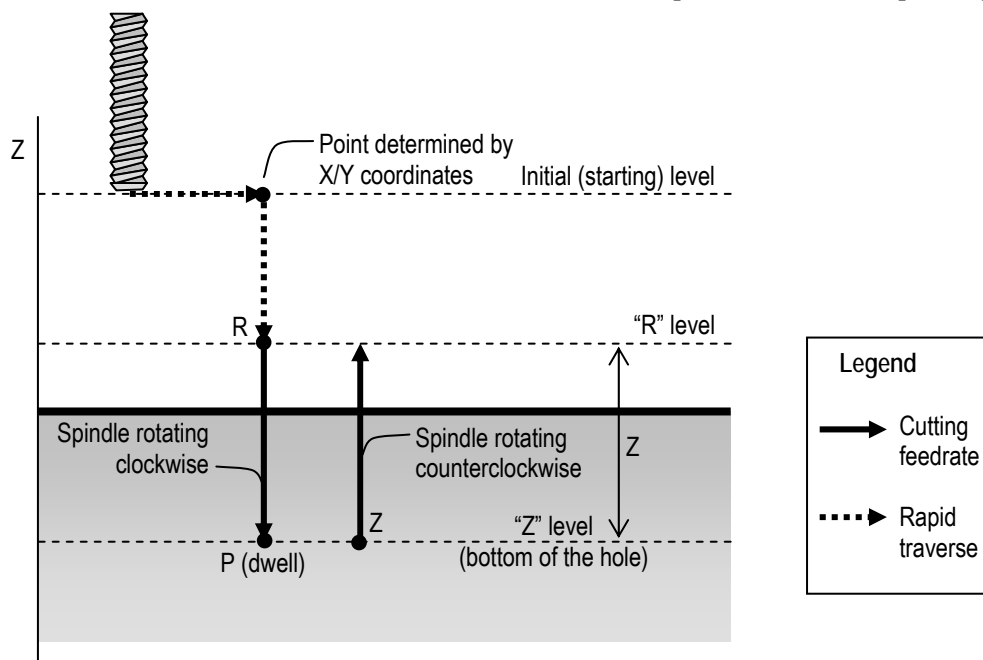


Figure 6-64: G84 Parameters (with G99 in effect)

Example of G84

M4 S500; Starts spindle rotation in the clockwise direction.
G90 G99 G84 X200 Y200 Z-100 R-20 P500 F200;
Positions the spindle at X=200, Y=200; taps hole #1 with spindle rotation in the clockwise direction; dwells for 500 ms; returns to R level with spindle rotation in the counterclockwise direction.
Y250;
Moves the spindle position to X=200, Y=250; taps hole #2 with spindle rotation in the clockwise direction; dwells for 500 ms; returns to R level with spindle rotation in the counterclockwise direction.
Y300;
Moves the spindle position to X=200, Y=300; taps hole #3 with spindle rotation in the clockwise direction; dwells for 500 ms; returns to R level with spindle rotation in the counterclockwise direction.
G98 Y350;
Moves the spindle position to X=200, Y=350; taps hole #4 with spindle rotation in the clockwise direction; dwells for 500 ms; returns to initial level with spindle rotation in the counterclockwise direction.

Limitations

- To change the drilling axis, you must first cancel the G84 canned cycle.
- If the G84 block of code does not contain X, Y, Z, R, or any other axes, the G84 tapping cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform tapping.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G84, the G84 block is not executed.
- Tool offsets are ignored in a G84 tapping cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G84:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The spindle rotates clockwise from point R to point Z (the bottom of the hole) at cutting feedrate.
 - At the bottom of the hole, the spindle dwells as specified, and changes the direction of rotation from clockwise to counterclockwise.
 - The spindle retracts at the cutting feedrate from the bottom of the hole to point R, while rotating counterclockwise, creating a thread.
 - If G99 is in effect, the cycle is complete at this point. If G98 is in effect, the spindle retracts to the initial point at rapid traverse, while still rotating.
- Rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate in the Reference Manual for ServoWorks CNC Parameters and Functions*).
- When G84 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins tapping.
- When “K” is used to specify multiple G84 tapping cycles, any M code specified in the G84 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G84 tapping cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).

Other G-Codes Which Affect G84

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G84 cycle.
- The G84 tapping cycle can be cancelled by G80 (canned cycle cancel).



CAUTION

Feedrate overrides are ignored for G84 tapping cycles.



CAUTION

A feed hold command issued while G84 is executing does not stop the machine until the return operation of the G84 tapping cycle is complete.



CAUTION

Before specifying a G84 cycle, start spindle rotation in the clockwise direction with an M code (miscellaneous function) or an S code (spindle function).

6.3.37 Rigid Tapping Cycle (G84)

Description

G84 can be used for both a normal canned tapping cycle, or for a rigid tapping cycle. In order to use G84 for rigid tapping, an “M29” code must be coded before G84 to enable Rigid Mode. [“M29” is for “Spindle Rigid Mode On,” as defined in the default LadderWorks PLC sequence program – SoftServo_M.lad – included with your ServoWorks CNC product.]

Tapping is the operation of cutting threads inside a previously drilled hole using a tap. (The hole must be drilled smaller than the root diameter of the thread.) With a tapping cycle, the spindle turns clockwise until it reaches the bottom of the hole, at which point it begins to rotate counterclockwise and retract, forming a thread.

“Rigid Mode” refers to a mode whereby the spindle motor is controlled as if it were a servo motor, to allow a faster tapping cycle. With this synchronized tapping, the machine spindle rotates in synchronization with the feed rate to match the exact pitch of the thread being cut or formed. At this time, rigid tapping is limited to the servo spindle axis type (a servo motor and drive used as a spindle); it is not available for an inverter spindle at this time.

G84 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect. When it reaches either the initial point or the R point, the spindle stops.

Required Format

M29 S

G84 X Y Z R P F K;

OR

M29 S G84 X Y Z R P F K;

Possible Parameters That Can Be Used With G84

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R and the position of the bottom of the hole

R – distance to point R level from the initial level

P – dwell time (in milliseconds) at the bottom of the hole and at point R when a return is made

F – cutting feedrate for tapping

K – number of times to repeat the entire tapping cycle (the default value for “K” is “1”)

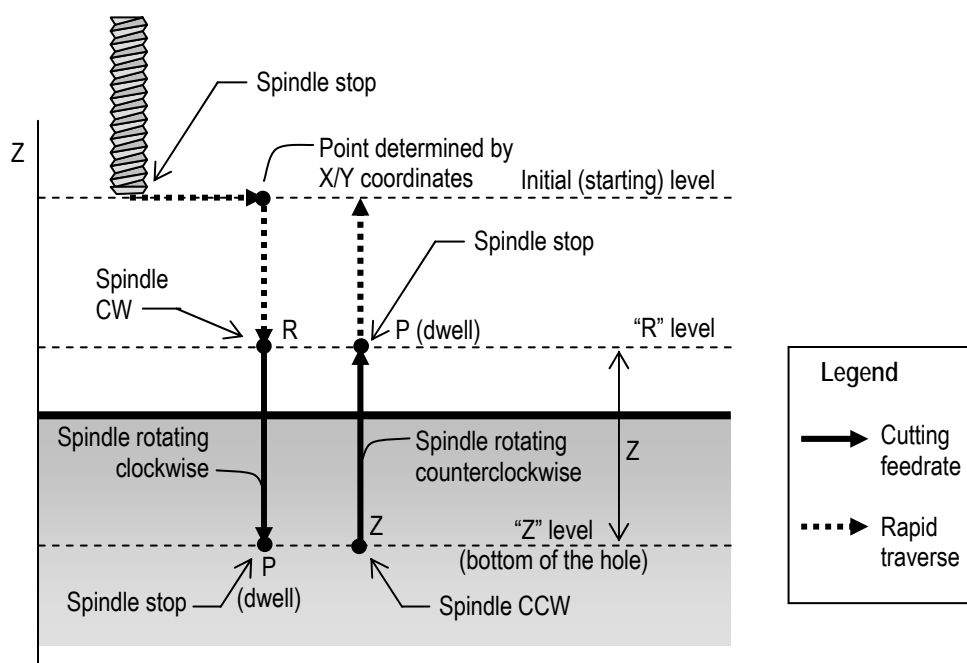


Figure 6-65: G84 Parameters (with G98 in effect)

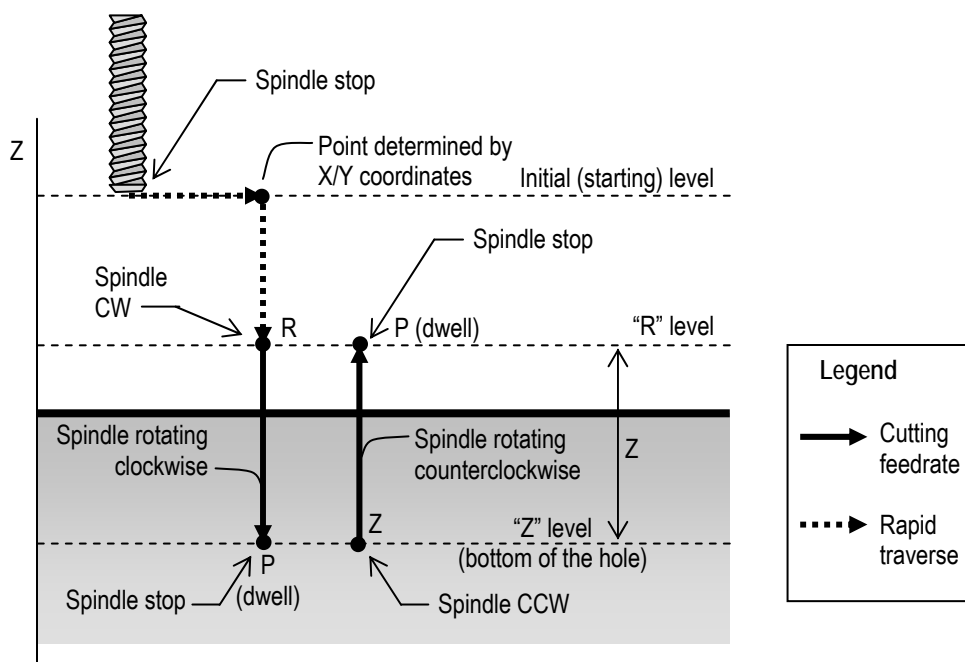


Figure 6-66: G84 Parameters (with G99 in effect)

Example of G84

M29 S100; Establishes rigid tapping mode and stops the spindle. Control is switched from open loop (S axis) spindle control (velocity control) to closed loop (C axis) spindle control (position control). Performs a spindle orientation operation.

G90 G95 G99 G84 X200 Y200 Z-100 R-20 P500 F200;
 Positions the spindle at X=200, Y=200; starts spindle rotation in the clockwise direction; taps hole #1 with spindle rotation in the clockwise direction; the spindle stops upon reaching the bottom of the hole; dwells for 500 ms; returns to R level with spindle rotation in the counterclockwise direction; spindle stops; dwells for 500 ms.

Y250; Moves the spindle position to X=200, Y=250; taps hole #2 with spindle rotation in the clockwise direction; the spindle stops upon reaching the bottom of the hole; dwells for 500 ms; returns to R level with spindle rotation in the counterclockwise direction; spindle stops; dwells for 500 ms.

Y300; Moves the spindle position to X=200, Y=300; taps hole #3 with spindle rotation in the clockwise direction; the spindle stops upon reaching the bottom of the hole; dwells for 500 ms; returns to R level with spindle rotation in the counterclockwise direction; spindle stops; dwells for 500 ms.

G98 Y350; Moves the spindle position to X=200, Y=350; taps hole #4 with spindle rotation in the clockwise direction; the spindle stops upon reaching the bottom of the hole; dwells for 500 ms; returns to R level with spindle rotation in the clockwise direction; spindle stops; returns to initial level with spindle stopped.

G80; Cancels the rigid tapping cycle. Control is switched back to open loop (S axis) spindle control (velocity control).

Limitations

- Rigid tapping is limited to one axis type: servo spindle (a servo motor and drive used as a spindle).
- To change the drilling axis, you must first cancel the G84 canned cycle.
- If the G84 block of code does not contain X, Y, Z, R, or any other axes, the G84 tapping cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform tapping.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G84, the G84 block is not executed.
- Tool offsets are ignored in a G84 tapping cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G84:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The spindle rotates clockwise from point R to point Z (the bottom of the hole) at cutting feedrate.
 - At the bottom of the hole, the spindle stops, dwells as specified, and the spindle starts counterclockwise rotation.
 - The spindle retracts at the cutting feedrate from the bottom of the hole to point R, while rotating counterclockwise, creating a thread.
 - The spindle stops, and dwells as specified.
 - If G99 is in effect, the cycle is complete at this point. If G98 is in effect, the spindle retracts to the initial point at rapid traverse, with the spindle stopped.
- The thread lead (the distance between threads, measured on a plane parallel to the centerline of the threaded part) varies depending upon whether G94 or G95 is in effect.
 - In G94 feed per minute mode, thread lead = feedrate / spindle speed
Example: (100 mm/min) / (50 rot/min) = 2.0 mm thread lead
 - In G95 feed per revolution mode, thread lead = feedrate.
- Rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).

- When G84 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins tapping.
- When “K” is used to specify multiple G84 tapping cycles, any M code specified in the G84 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G84 tapping cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).

Other G-Codes Which Affect G84

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G84 cycle.
- The G94 and G95 codes specifying feed per minute or feed per revolution mode, respectively, affect how the feedrates you specify are processed and how the machine spindle is synchronized with the Z axis.
- The G84 counter rigid tapping cycle can be cancelled by G80 (canned cycle cancel).



CAUTION

Feedrate overrides are ignored for G84 tapping cycles.



CAUTION

A feed hold command issued while G84 is executing does not stop the machine until the return operation of the G84 tapping cycle is complete.



CAUTION

Before specifying a G84 cycle, start spindle rotation in the clockwise direction with M29 (miscellaneous function for rigid mode) and an S code (spindle function).

6.3.38 Boring Cycle (G85)

Description

G85 is used for a straightforward boring cycle in which both the boring and the retraction are performed at the cutting feedrate. Boring improves the surface finish and cuts the parts to size – the tool follows its own path, fixing any deviation done by drilling. This G85 boring cycle bores the hole precisely with a tool attached to the spindle – the spindle turns clockwise until it reaches the bottom of the hole, and continues turning clockwise as it retracts.

G85 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect.

Required Format

G85 X Y Z R Q P F K;

Possible Parameters That Can Be Used With G85

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

F – cutting feedrate for boring

K – number of times to repeat the entire boring cycle (the default value for “K” is “1”)

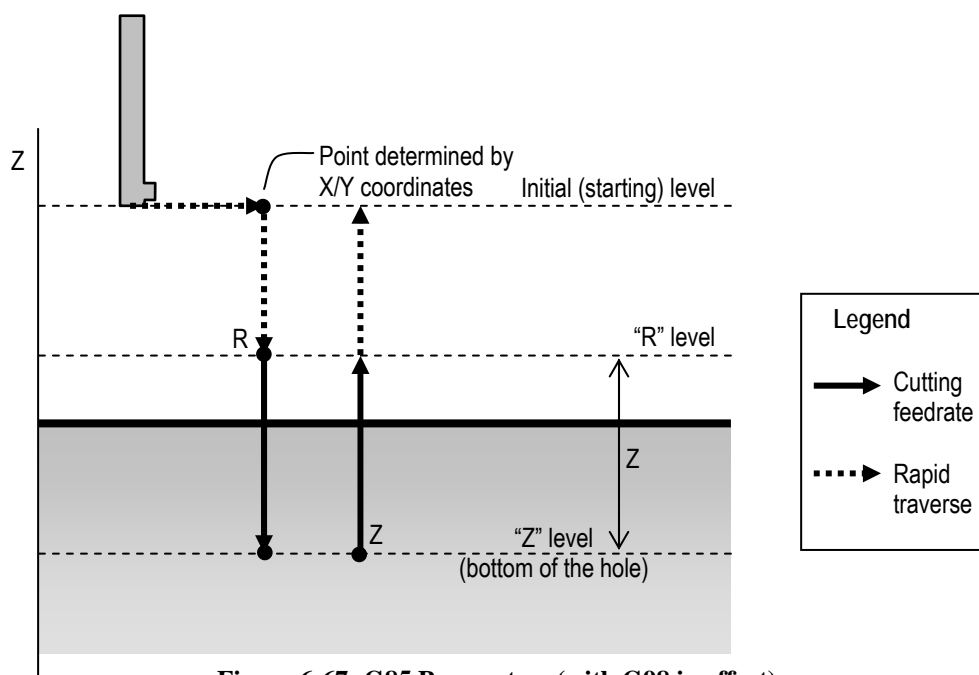


Figure 6-67: G85 Parameters (with G98 in effect)

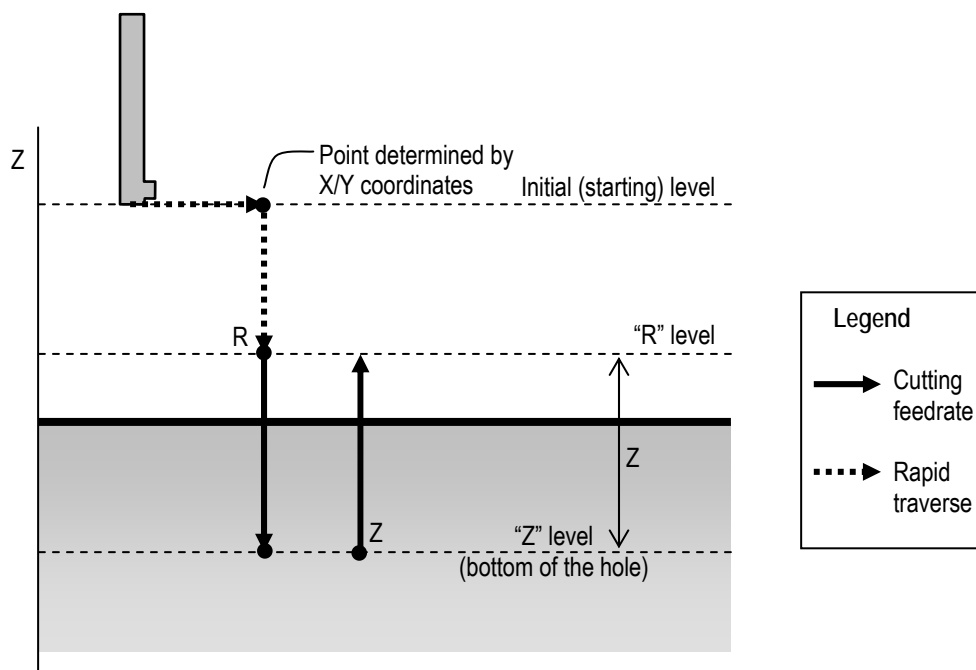


Figure 6-68: G85 Parameters (with G99 in effect)

Example of G85

M4 S500; Starts spindle rotation.
G90 G99 G85 X200 Y200 Z-100 R-20 F200; Positions the spindle at X=200, Y=200; bores hole #1; returns to R level at cutting feedrate.
Y250; Moves the spindle position to X=200, Y=250; bores hole #2; returns to R level at cutting feedrate.
Y300; Moves the spindle position to X=200, Y=300; bores hole #3; returns to R level at cutting feedrate.
G98 Y350; Moves the spindle position to X=200, Y=350; bores hole #4; returns to R level at cutting feedrate, and then initial level at rapid traverse.

Limitations

- To change the boring axis, you must first cancel the G85 canned cycle.
- If the G85 block of code does not contain X, Y, Z, R, or any other axes, the G85 boring cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform boring.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G85, the G85 block is not executed.
- Tool offsets are ignored in a G85 boring cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G85:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The spindle bores from point R to point Z (the bottom of the hole) at the cutting feedrate.
 - The tool retracts from point Z to point R at the cutting feedrate.
 - If G99 is in effect, the cycle is complete at this point. If G98 is in effect, the spindle retracts to the initial point at rapid traverse, while still rotating.

- Rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).
- When G85 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins boring.
- When “K” is used to specify multiple G85 boring cycles, any M code specified in the G85 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G85 boring cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).

Other G-Codes Which Affect G85

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G85 cycle.
- The G85 boring cycle can be cancelled by G80 (canned cycle cancel).

**CAUTION**

Before specifying a G85 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).

6.3.39 Boring Cycle (Spindle Stop) (G86)

Description

G86 is used for a straightforward boring cycle in which the boring is performed at the cutting feedrate, the spindle stops rotation at the bottom of the hole, and retraction is performed with rapid traverse. Boring improves the surface finish and cuts the parts to size – the tool follows its own path, fixing any deviation done by drilling. This G86 boring cycle bores the hole precisely with a tool attached to the spindle – the spindle turns clockwise until it reaches the bottom of the hole, and stops rotating as it retracts.

G86 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect.

Required Format

G86 X Y Z R F K;

Possible Parameters That Can Be Used With G86

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

F – cutting feedrate for boring

K – number of times to repeat the entire boring cycle (the default value for “K” is “1”)

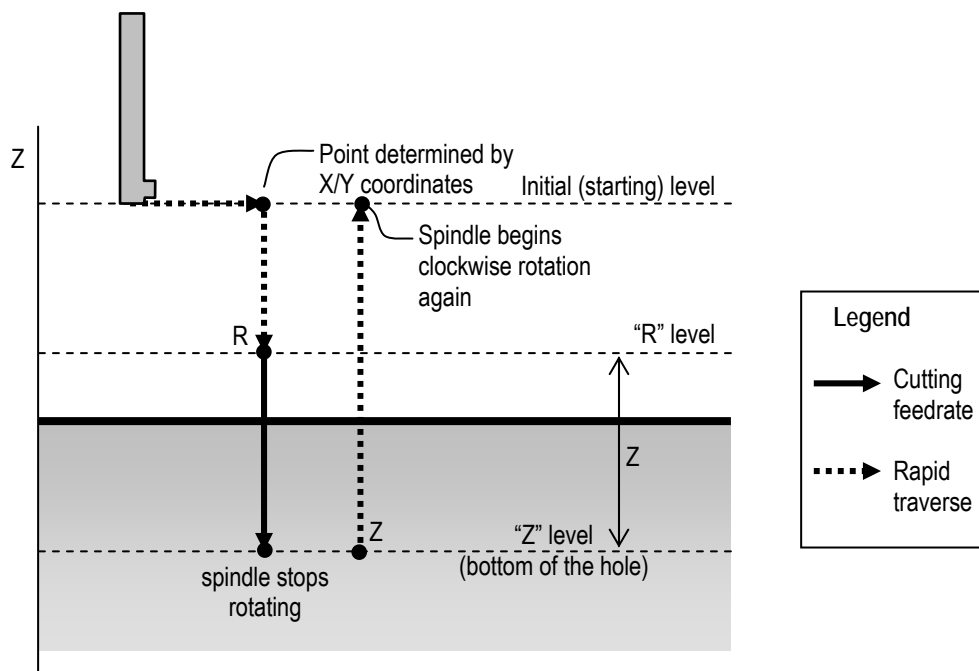


Figure 6-69: G86 Parameters (with G98 in effect)

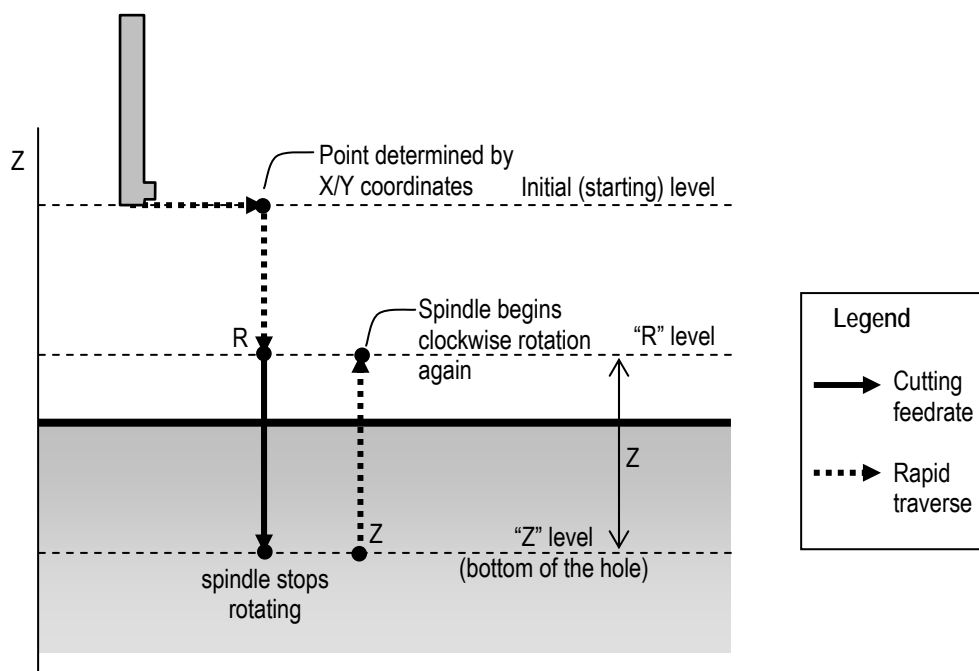


Figure 6-70: G86 Parameters (with G99 in effect)

Example of G86

M4 S500; Starts spindle rotation.
G90 G99 G86 X200 Y200 Z-100 R-20 F200;
Positions the spindle at X=200, Y=200; bores hole #1; spindle stops rotation; returns to R level; spindle begins clockwise rotation again.
Y250; Moves the spindle position to X=200, Y=250; bores hole #2; spindle stops rotation; returns to R level; spindle begins clockwise rotation again.
Y300; Moves the spindle position to X=200, Y=300; bores hole #3; spindle stops rotation; returns to R level; spindle begins clockwise rotation again.
G98 Y350; Moves the spindle position to X=200, Y=350; bores hole #4; spindle stops rotation; returns to initial level; spindle begins clockwise rotation again.

Limitations

- To change the boring axis, you must first cancel the G86 canned cycle.
- If the G86 block of code does not contain X, Y, Z, R, or any other axes, the G86 boring cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform boring.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G86, the G86 block is not executed.
- Tool offsets are ignored in a G86 boring cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G86:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The spindle bores from point R to point Z (the bottom of the hole).
 - At the bottom of the hole, the spindle rotation is stopped.
 - The tool retracts at rapid traverse to either the initial level or to the R level, depending upon whether G98 or G99 is in effect.
 - The spindle begins rotating again before proceeding to the next block.

- Rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).
- When G86 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins boring.
- When “K” is used to specify multiple G86 boring cycles, any M code specified in the G86 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G86 boring cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).

Other G-Codes Which Affect G86

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G86 cycle.
- The G86 boring cycle (spindle stop) can be cancelled by G80 (canned cycle cancel).

**CAUTION**

Before specifying a G86 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).

6.3.40 Back Boring Cycle (G87)

Description

G87 is used for a boring cycle in which the boring is performed starting at the bottom of the workpiece, and boring up, not down. Oriented spindle stops and spindle/tool shifts are performed several times in the cycle.

Boring improves the surface finish and cuts the parts to size – the tool follows its own path, fixing any deviation done by drilling. This G87 boring cycle bores the hole precisely with a tool attached to the spindle – the spindle performs an oriented spindle stop, the tool is moved away from the machined surface of the workpiece and is lowered into the hole and out the other side of the workpiece at rapid traverse. Then the spindle rotates clockwise to bore from the bottom of the hole (the opposite side of the workpiece) to point Z, where it performs another oriented spindle stop. The tool is moved away from the machined surface of the workpiece, and the spindle and tool are retracted.

Required Format

G87 X Y Z R Q P F K;

Possible Parameters That Can Be Used With G87

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance from the bottom of the hole (point R) to point Z

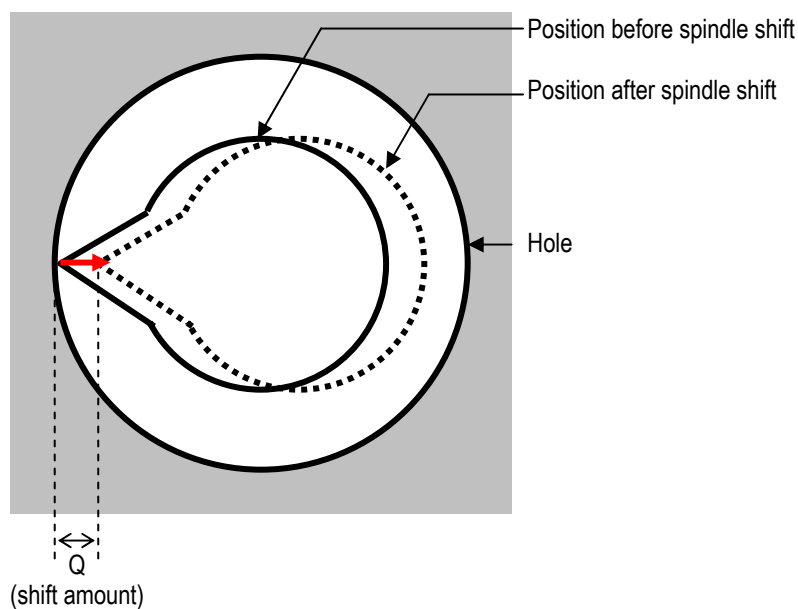
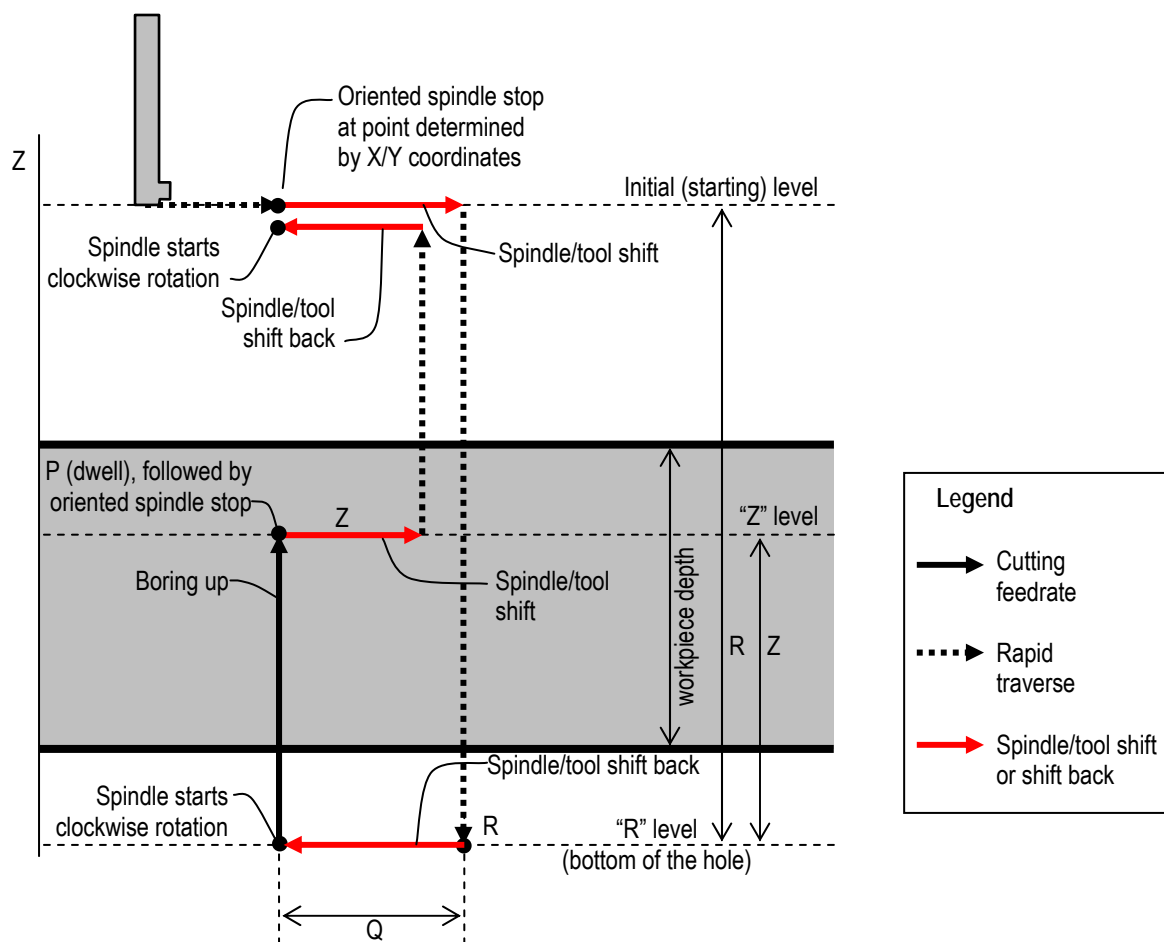
R – distance to point R level (bottom of the hole) from the initial level

Q – tool shift amount at the bottom of the hole

P – dwell time (in milliseconds) at the bottom of the hole

F – cutting feedrate for boring

K – number of times to repeat the entire back boring cycle (the default value for “K” is “1”)



Example of G87

M4 S500; Starts spindle rotation.
 G90 G87 X200 Y200 Z-100 R-20 Q4 P100 F200;
 Positions the spindle at X=200, Y=200; performs an oriented spindle stop; shifts the tool/spindle 4 mm; moves the tool to the bottom of hole at rapid traverse; shifts the tool/spindle back; bores hole #1 from point R to point Z; dwells for 100 ms; performs another oriented spindle stop; shifts the spindle/tool 4mm; returns to initial level at rapid traverse; shifts the tool/spindle back; starts the spindle clockwise rotation.
 Y250;
 Positions the spindle at X=200, Y=2050; performs an oriented spindle stop; shifts the tool/spindle 4 mm; moves the tool to the bottom of hole at rapid traverse; shifts the tool/spindle back; bores hole #1 from point R to point Z; dwells for 100 ms; performs another oriented spindle stop; shifts the spindle/tool 4mm; returns to initial level at rapid traverse; shifts the tool/spindle back; starts the spindle clockwise rotation.
 Y300;
 Positions the spindle at X=200, Y=300; performs an oriented spindle stop; shifts the tool/spindle 4 mm; moves the tool to the bottom of hole at rapid traverse; shifts the tool/spindle back; bores hole #1 from point R to point Z; dwells for 100 ms; performs another oriented spindle stop; shifts the spindle/tool 4mm; returns to initial level at rapid traverse; shifts the tool/spindle back; starts the spindle clockwise rotation.
 Y350;
 Positions the spindle at X=200, Y=350; performs an oriented spindle stop; shifts the tool/spindle 4 mm; moves the tool to the bottom of hole at rapid traverse; shifts the tool/spindle back; bores hole #1 from point R to point Z; dwells for 100 ms; performs another oriented spindle stop; shifts the spindle/tool 4mm; returns to initial level at rapid traverse; shifts the tool/spindle back; starts the spindle clockwise rotation.

Limitations

- To change the boring axis, you must first cancel the G87 canned cycle.
- If the G87 block of code does not contain X, Y, Z, R, or any other axes, the G87 back boring cycle is not performed.
- Q must be specified as a positive value. If you assign a negative value to Q, the negative sign is ignored, and the value is treated as positive. You set the “Shift Direction” in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M, as either “No Shift,” “Shift +” or “Shift –.” [See *Section 3.3.1: Shift Direction* in the *Reference Manual for ServoWorks CNC Parameters and Functions*.]
- Q and R parameters cannot be stored as modal data when they are specified in a block that does not perform boring.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G87, the G87 block is not executed.
- Tool offsets are ignored in a G87 back boring cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- **Implementation of G87:**
 - The tool positions along the X and Y axes at rapid traverse.
 - The spindle stops at the fixed rotation position.
 - The tool is moved in the direction opposite to the tool tip (as shown in Figure 6-72) prior to positioning at point R, to prevent any damage to the machined surface.
 - The tool moves down to the bottom of the hole (position R) at rapid traverse.
 - The tool is then shifted back, in the direction of the tool tip, and the spindle begins clockwise rotation.
 - The spindle bores from point R (the bottom of the hole) to point Z, in an upward direction, at the cutting feedrate.
 - At point Z, the spindle dwells, then is stopped at a fixed rotation position.
 - The tool is moved in the direction opposite to the tool tip (as shown in Figure 6-72) prior to retraction, to prevent any damage to the machined surface.
 - The tool then retracts to the initial level at rapid traverse.
 - At the initial level, the tool is shifted back, in the direction of the tool tip, and the spindle begins clockwise rotation before proceeding to the next block.
- Rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate in the Reference Manual for ServoWorks CNC Parameters and Functions*).
- When G87 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins boring.
- When “K” is used to specify multiple G87 back boring cycles, any M code specified in the G87 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G87 back boring cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).

Other G-Codes Which Affect G87

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of most canned cycles. However, G87 only works with G98; if you specify G87 with G99, you will get an error.
- The G87 back boring cycle can be cancelled by G80 (canned cycle cancel).

CAUTION

The Q parameter is stored as modal data within canned cycles. Specify this value carefully, because it is also used as the depth of cut for G73 and G83, and the spindle/tool shift amount for G76.

CAUTION

Before specifying a G87 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).

CAUTION

If you specify G87 with G99, you will get an error (see below).

6.3.41 Boring Cycle (Dwell) (G89)

Description

G89 is used for a straightforward boring cycle in which both the boring and the retraction are performed at the cutting feedrate. Boring improves the surface finish and cuts the parts to size – the tool follows its own path, fixing any deviation done by drilling. This G89 boring cycle bores the hole precisely with a tool attached to the spindle – the spindle turns clockwise until it reaches the bottom of the hole, dwells, and continues turning clockwise as it retracts.

G89 is implemented differently depending upon whether G98 (return to initial point in canned cycle) or G99 (return to R point in canned cycle) is in effect.

Except for the dwell at the bottom of the hole, this cycle is the same as the G85 boring cycle.

Required Format

G89 X Y Z R P F K;

Possible Parameters That Can Be Used With G89

X – coordinate value for the X axis (hole position data)

Y – coordinate value for the Y axis (hole position data)

Z – distance to the bottom of the hole from point R

R – distance to point R level from the initial level

P – dwell time (in milliseconds) at the bottom of the hole

F – cutting feedrate for boring

K – number of times to repeat the entire boring cycle (the default value for “K” is “1”)

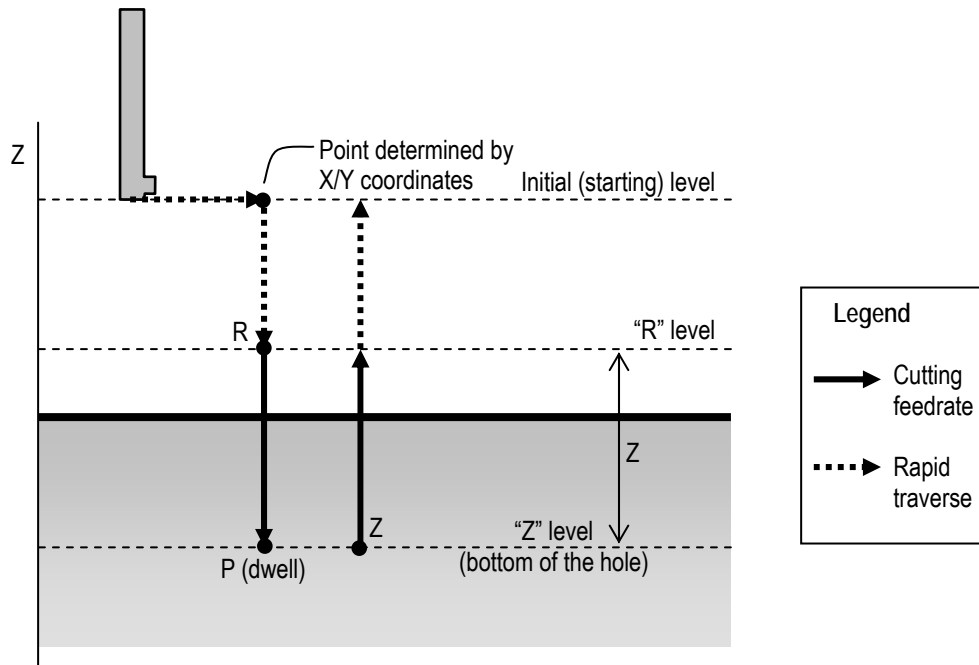


Figure 6-73: G89 Parameters (with G98 in effect)

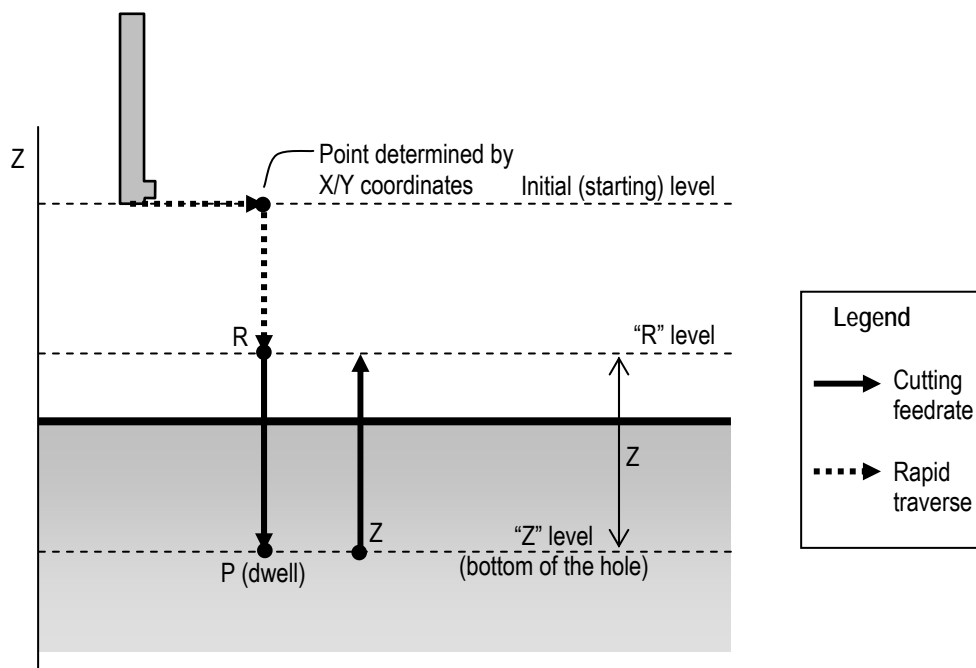


Figure 6-74: G89 Parameters (with G99 in effect)

Example of G89

M4 S500; Starts spindle rotation.
G90 G99 G89 X200 Y200 Z-100 R-20 P100 F200;
Positions the spindle at X=200, Y=200; bores hole #1; dwells for 100 ms; returns to R level.
Y250; Moves the spindle position to X=200, Y=250; bores hole #2 at cutting feedrate; dwells for 100 ms; returns to R level at cutting feedrate.
Y300; Moves the spindle position to X=200, Y=300; bores hole #3 at cutting feedrate; dwells for 100 ms; returns to R level at cutting feedrate.
G98 Y350; Moves the spindle position to X=200, Y=350; bores hole #4 at cutting feedrate; dwells for 100 ms; returns to R level at cutting feedrate; returns to initial level at rapid traverse.

Limitations

- To change the boring axis, you must first cancel the G89 canned cycle.
- If the G89 block of code does not contain X, Y, Z, R, or any other axes, the G89 boring cycle is not performed.
- An R parameter cannot be stored as modal data when it is specified in a block that does not perform boring.
- If a G code from modal group 01 (G00, G00.1, G01, G02, G03 or G02.3) is specified in the same block as G89, the G89 block is not executed.
- Tool offsets are ignored in a G89 boring cycle, except for tool length offsets that are applied to the initial positioning to point R.

Notes

- Implementation of G89:
 - The tool positions along the X and Y axes, and moves to point R, at rapid traverse.
 - The spindle bores from point R to point Z (the bottom of the hole) at the cutting feedrate.
 - At the bottom of the hole, the spindle dwells.
 - The tool retracts from point Z to point R at the cutting feedrate.
 - If G99 is in effect, the cycle is complete at this point. If G98 is in effect, the spindle retracts to the initial point at rapid traverse, while still rotating.

- Rapid traverse is performed at the rapid feedrate specified in the “NC Settings” parameters tab in Configuration Mode in the S-100M/S-120M/S-140M (see *Section 4.3: Rapid Feedrate* in the *Reference Manual for ServoWorks CNC Parameters and Functions*).
- When G89 and an M code are specified in the same block, the M code is executed during the first positioning operation, before the S-100M/S-120M/S-140M begins boring.
- When “K” is used to specify multiple G89 boring cycles, any M code specified in the G89 block of code is only executed for the first hole; for all subsequent holes, the M code is not executed.
- When a tool length offset (G43, G44 or G49) is specified in a G89 boring cycle, the tool length offset is applied during the first positioning operation (moving the spindle to point R).

Other G-Codes Which Affect G89

- G98 and G99 (Return to Initial Point / R Point in Canned Cycle) determine the end position of the G89 cycle.
- The G89 boring cycle (dwell) can be cancelled by G80 (canned cycle cancel).

**CAUTION**

Before specifying a G89 cycle, start spindle rotation with an M code (miscellaneous function) or an S code (spindle function).

6.3.42 Absolute / Incremental Programming (G90, G91)

Description

The G90 and G91 commands are used to tell the machine whether you'll be commanding movement with absolute coordinates or incremental coordinates.

G90: absolute programming (absolute coordinates, relative to the part origin)

G91: incremental programming (incremental coordinates, relative to the current tool position)

Possible Parameters That Can Be Used With G90, G91

None.

Example

Suppose you want to program the movement from start point to end point shown in the following figure, with G01 (linear interpolation).

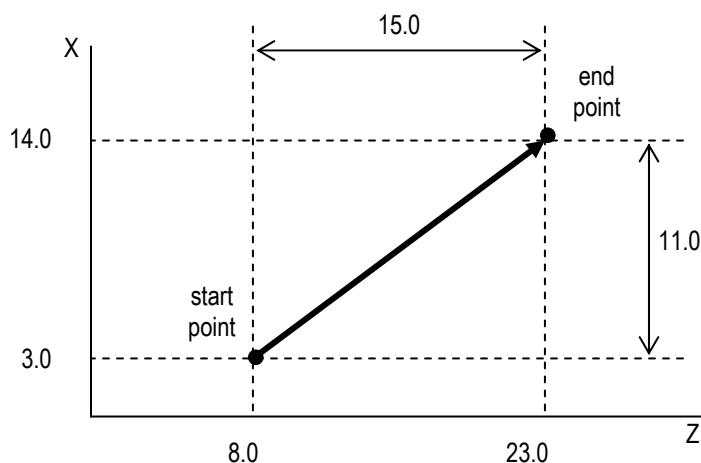


Figure 6-75: Absolute/Incremental Programming Example

You could program this movement with either G90 or G91, as shown:

G90 G01 X14.0 Z23.0

OR

G91 G01 X11.0 Z15.0

Default

G90 is the default mode when neither G90 nor G91 has been programmed.

6.3.43 Workpiece Coordinate Programming (G92)

Description

The G92 command is used to set up a workpiece coordinate system. With this command, you are setting the zero point at specified coordinates that are on the workpiece. This makes programming simple, because all commands related to that workpiece are relative to a point on the workpiece, rather than relative to the machine origin.

Required Format

G92 X Y Z A B C

Possible Parameters That Can Be Used With G92

X – coordinate value for the X axis
Y – coordinate value for the Y axis
Z – coordinate value for the Z axis
A – coordinate value for the A axis
B – coordinate value for the B axis
C – coordinate value for the C axis

Example

G92 X5.6 Z8.2

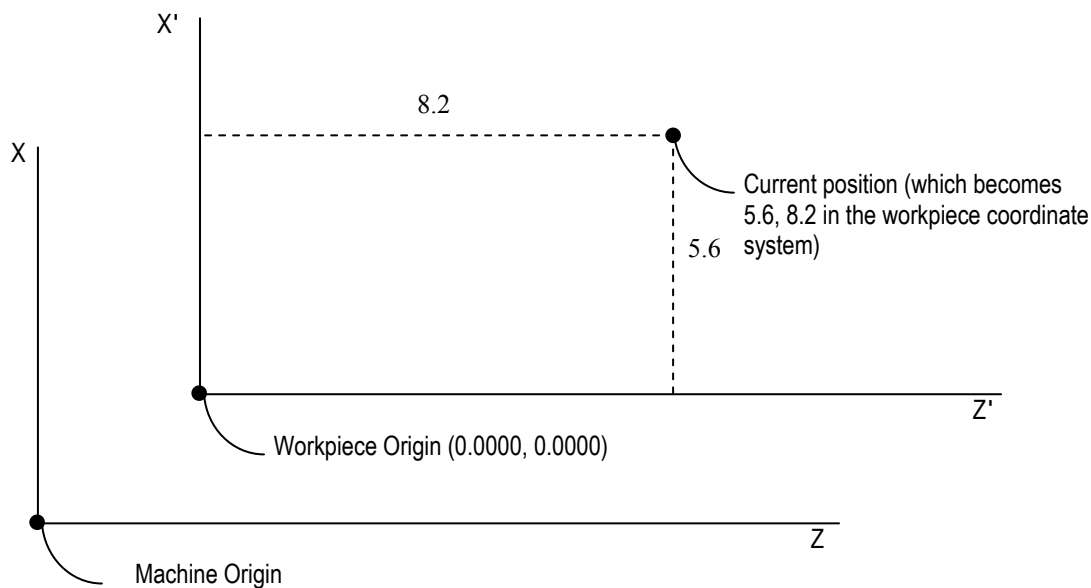


Figure 6-76 Workpiece Coordinate Programming Example

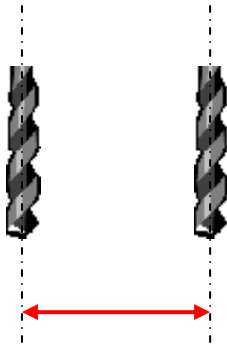
6.3.44 Feed per Minute / Feed per Revolution (G94, G95)

Description

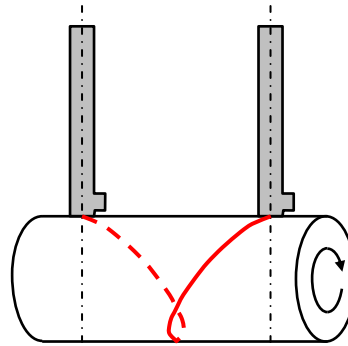
The G94 and G95 commands are used to specify which feed mode will be used to specify feedrates (with F codes).

G94: Feed per minute mode (millimeters/minute or inches/minute)

G95: Feed per revolution mode (feed amount per spindle revolution – millimeters/revolution or inches/revolution)



G94: Feed amount per minute



G95: Feed amount per revolution

Figure 6-77: G94 and G95

Possible Parameters That Can Be Used With G94 and G95

None.

Default

G94 is the default mode when neither G94 nor G95 has been programmed.

6.3.45 Return to Initial Point / R Point in Canned Cycle (G98, G99)

Description

The G98 and G99 commands are used to tell the machine whether want the tool to return to the initial level or the R level in a canned cycle (G73, G74, G76, G81, G82, G83, G84, G85, G87 and G89).

G98: return to initial level/initial point at the end of a canned cycle

G99: return to the R level/R point at the end of a canned cycle

Possible Parameters That Can Be Used With G98, G99

None.

Example

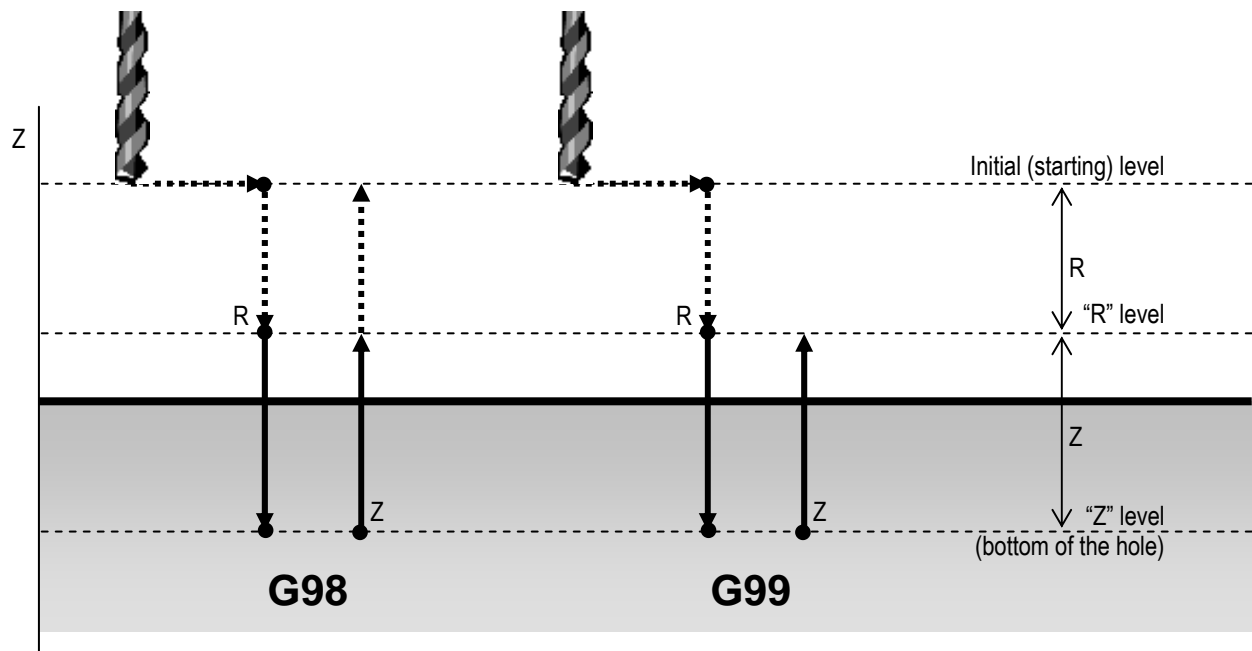


Figure 6-78: Drilling Cycle Showing the Difference between G98 and G99

Default

G98 is the default mode when neither G98 nor G99 has been programmed.

6.3.46 Linear Interpolation Feedrate Include / Exclude Rotary Axes (G310, G311)

Description

G310 and G311 are used to specify whether the specified feedrate includes rotary axes or not.

G310: linear interpolation feedrate includes rotary axes

G311: linear interpolation feedrate excludes rotary axes

Possible Parameters That Can Be Used With G310, G311

None.

Examples

Assume axis A is a rotary axis, and axes X, Y, Z are normal axes (linear table driven by a ball screw).

The following G-codes:

G310;

G91 G01 X30.0 Y40.0 A50.0 F3000

Will cause each axis to move at the speed shown below:

$$\text{X axis speed} = \frac{3000 \times 30.0}{\sqrt{30.0^2 + 40.0^2 + 50.0^2}} = 1,272.8 \text{ mm/min}$$

$$\text{Y axis speed} = \frac{3000 \times 40.0}{\sqrt{30.0^2 + 40.0^2 + 50.0^2}} = 1,697.1 \text{ mm/min}$$

$$\text{A axis speed} = \frac{3000 \times 50.0}{\sqrt{30.0^2 + 40.0^2 + 50.0^2}} = 2,121.3 \text{ deg/min}$$

While the following G-codes:

G311;
G91 G01 X30.0 Y40.0 A50.0 F3000

Will cause each axis to move at the speed shown below:

$$\text{X axis speed} = \frac{3000 \times 30.0}{\sqrt{30.0^2 + 40.0^2}} = 1,800.0 \text{ mm/min}$$

$$\text{Y axis speed} = \frac{3000 \times 40.0}{\sqrt{30.0^2 + 40.0^2}} = 2,400.0 \text{ mm/min}$$

$$\text{A axis speed} = \frac{3000 \times 50.0}{\sqrt{30.0^2 + 40.0^2}} = 3,000.0 \text{ deg/min}$$

Default

G310 is the default mode when neither G310 nor G311 has been programmed.

Chapter 7: M Codes (Miscellaneous Codes)

If there is more than one possible meaning, both meanings are listed.

M-CODE	DESCRIPTION
M00	program temporary stop
M01	program optional stop
M02	program end
M30	program end and rewind
M98	subprogram call from a main program (ex.: M98 PTest_1 R3 calls subprogram OTest_1.dat in the same folder as the parent program, and repeats it three times)
M99	return to main program from a subprogram OR returns the flow of execution to the beginning of the main program (if used in the main program)

Table 7-1: Summary of M Codes

Except for M98 and M99, all M codes listed here are programmed in the default LadderWorks PLC sequence program provided with your ServoWorks S-100M series product. In order to use these M codes (i.e. M00, M01, etc.), you must put the relevant PLC programming code for each M code in the customized PLC sequence program for your machine. We recommend you use the default LadderWorks PLC sequence program as the starting point for your customized PLC sequence program.

In addition to these preset M codes, you can create customized M codes using custom M code macro calls. See *Section 6.5: Custom Macro Calls Using G Codes, M Codes, S Codes or T Codes* in the *ServoWorks CNC Macro Programming Manual*.

NOTE: General M codes are processed after any motion specified in the same block is completed. Therefore, if you want your M code to be executed first, specify it in a separate line of code prior to the block of code specifying the motion you want it to precede.

Chapter 8: Spindle Functions and S Codes

Description

S codes are 5-digit codes used to specify either the spindle speed (RPM) directly or the constant surface speed (the relative speed between the tool tip and the workpiece).

Required Format

For a single spindle, or Spindle #1 for multiple spindles: Sxxxxx

For multiple spindles: Sxx_yyyyy

Parameters

For a single spindle: xxxxx – spindle speed (integer)

For multiple spindles: xx – spindle number (1 to 16)
yyyyy – spindle speed (integer)

Examples

S2000 (sets Spindle #1 to a speed of 2000 RPMs)
S2_1500 (sets Spindle #2 to a speed of 1500 RPMs)

Notes

- The S code is sent to PLC through binary format (F22 ~ F25).
- For each new S code, a one-shot S strobe signal (F07.2) is also sent to PLC.
- For multiple spindles, the underscore symbol “_” is required to connect spindle number and spindle speed parameters.
- For servo type spindles (please refer to the *Reference Manual for ServoWorks CNC Parameters and Functions* for a description of the NC and Machine parameter, “Axis Type”), the S code would start the spindle rotation directly. A positive spindle speed would turn the spindle in the CW direction, and a negative spindle speed would turn the spindle in the CCW direction. Spindle speed zero would stop the spindle. (Refer to the *Reference Manual for ServoWorks CNC Parameters and Functions* for a description of how to preset the appropriate spindle turning direction – see the Motor/Drive parameter called “Motor Polarity”).

Other Codes Which Affect S Codes

- M03 rotates the spindle in the CW direction (for inverter type and single spindle only)
- M04 rotates the spindle in the CCW direction (for inverter type and single spindle only)
- M05 stops the spindle (for inverter type and single spindle only)

Variations

- In addition to these preset S codes, you can create customized S codes using custom S code macro calls. See *Section 6.5: Custom Macro Calls Using G Codes, M Codes, S Codes or T Codes* in the *ServoWorks CNC Macro Programming Manual*.

- You can use S codes without having a spindle in your ServoWorks S-100M/S-120M/S-140M system, IF you are using version 3.0 or higher. If you use an S code with no spindle, you will NOT get an error; your ServoWorks application will just write to PLC F data. [NOTE: ServoWorks S-100M/S-120M/S-140M versions older than version 3.0 WILL generate an error message if S codes are used when Axis 4 has been set to “Unused.”] For more information, see *Section 7.9: Including a Spindle in a MECHATROLINK Interface System* in the *ServoWorks CNC Setup and Integration Manual for the MECHATROLINK Interface System* or *Section 5.10: Including an Inverter Spindle in an RTEX Interface System* in the *ServoWorks CNC Setup and Integration Manual for the Panasonic Realtime Express (RTEX) Interface System*.

Chapter 9: Tool Functions and T Codes

9.1 Overview of T Codes

Description

T codes are 3-digit codes used to select the tool and associated offset number.

Required Format

Txxx

Parameters

xxx – tool number (and offset number) 1 ~ 256 (3-digit)

Notes

- T0 is used to cancel any active tool offset value
- Only tool number xxx is sent to PLC through binary format (F26.0 ~ F26.7)
- For each new tool number, a one-shot T strobe signal (F07.3) is also sent to PLC

Variations

In addition to this preset T code, you can create customized T codes using custom T code macro calls. See *Section 6.5: Custom Macro Calls Using G Codes, M Codes, S Codes or T Codes* in the *ServoWorks CNC Macro Programming Manual*.

9.2 Tool Offsets

In the Tool Offset Compensation Table (in Settings Mode of ServoWorks S-100M/S-120M/S-140M), there are 256 sets of offsets. Each set of offsets consists of the following:

- Length Geometry Offset (diameter or radius programming)
- Length Wear Offset (diameter or radius programming)
- Radius Geometry Offset
- Radius Wear Offset

When the specified tool offset number is selected, the offset values for Length and Radius are the sum of the corresponding Geometry Offset and the corresponding Wear Offset, respectively:

$$\text{Offset Data} = \text{Geometry Offset} + \text{Wear Offset}$$

The Tool Radius Offset is used as Tool Radius Compensation (TRC) with G40, G41 or G42.

Notes

- When a T code is commanded, the associated Tool Radius Offset will become effective. But the Tool Length Offset will not be effective unless either G43 or G44 modal is active and an H code is called explicitly (refer to *Section 6.3.17: Positive / Negative Tool Length Compensation/Compensation Cancel (G43, G44, G49)*).
- If a T code is programmed in the same line with motion commands, they will be processed in parallel. And the motion will be processed with the new tool radius offset.

Example

N00100 T1 G00 X1.0 Y2.0

The Tool #1 signal will be sent to PLC, and tool radius offset #1 will become active. At the same time, motion “G00 X1.0 Y2.0” will be processed with tool radius offset #1.

The above is equivalent to:

N00100 T1
N00110 G00 X1.0 Y2.0

Chapter 10: Direct Digital Output

Description

Direct Digital Output is a feature that allows the user to set the value of a 16-bit region of the PLC output space using G code instead of the PLC ladder. The range of PLC output addresses that can be used with this feature is Y80-Y99. From this range, the user chooses a 2-byte region to be controlled by G code.

Required Format

L

Parameters

L – A 16-bit value to output to the chosen output address space. The range of valid values is 0 to 65535.

Direct DOUT Address Parameter

The NC parameter Direct DOUT Address is used to specify the starting address of the 2-byte long region within the PLC output address space to control using Direct Digital Output. The range of valid values for this parameter is 80 to 98. For example, if Direct DOUT Address is set to 81, the output address space Y81.0~Y82.7 will be the address space controlled by Direct Digital Output.

Other G Codes Which Affect Direct Digital Output

Direct Digital Output can also be used in the same block as most other G codes. In this case, the digital output will be applied at the beginning of the G code block containing a Direct Digital Output (L) command. For example, if the G code block “G01 X100.0 F1000.0 L500” is executed, 500 will be output to the output address space at the same time that X starts moving.

Direct Digital Output does not affect G codes that already use the letter L as a parameter, such as G10 and G65. When using these G codes, any instances of L in the G code block will not trigger Direct Digital Output.

Notes

- If you have an analog output device in the servo network, you can use this feature to output an analog voltage through G code. In this case, you need to set L to the digital output that corresponds to the analog voltage you want to output.
- Once you use this feature, the PLC will no longer control the affected output address space until you restart the application. You will not be able to observe or process the output values through the PLC.

Index

- % 3-1
- (3-2
- 3D-DLACC 6-22
- 3D-DLACC on/off..... 2-2
- A 2-1
- A axis absolute or incremental coordinate value
 - designation 2-1
- absolute programming 6-113
- absolute values..... 1-1
- additional reference point 2-2
- additional reference points..... 6-32
- additional workpiece coordinate system selection..... 6-63
- address 1-1
- address descriptions..... 2-1
- address of tool length offset value 2-1
- Angle Limit 6-43
- angular displacement 2-2
- arc center modifier for the X axis 2-1
- arc center modifier for the Y axis 2-1
- arc center modifier for the Z axis..... 2-1
- arc center modifiers 6-10
- arc radius 6-9
- arc radius designation 2-2, 6-10
- argument assignments..... 6-66
- Arm Length 6-53
- automatic tool length compensation calibration .. 6-34
- automatic zero return to / from the reference points 6-30
- automatic zero return to additional reference points 6-32
- axis number 2-2
- axis type settings..... 6-25
- B 2-1
- B axis absolute or incremental coordinate value
 - designation 2-1
- back boring cycle..... 6-106
- block delete code 3-1
- block number 2-2
- block of code 1-1, 3-1
- block rollover..... 6-64
- blocks of code..... 1-3
- boring..... 6-80, 6-106
- boring cycle 6-100, 6-103
- boring cycle (dwell)..... 6-110
- C 2-1
- C axis absolute or incremental coordinate value
 - designation 2-1
- canned cycle cancel 6-83
- carriage return..... 3-1
- center of the arc 6-9
- chip-breaking process 6-70
- circle error allowance 6-13
- circular interpolation..... 6-9
- circular path 6-9
- clockwise circular interpolation..... 6-9
- coding system 1-1
- comment code..... 3-2
- comments..... 3-1
- constant for exponential interpolation 2-2
- constant helix angle 6-19
- constant surface speed 8-1
- continuous cutting mode..... 6-64
- continuous cutting mode with block rollover 6-64
- contour control on / off..... 6-22
- coordinate programming..... 6-114
- coordinate system rotation..... 6-67
- coordinate system selection 6-60, 6-61
- coordinates..... 6-113
- Corner Deceleration function 6-54
- counter rigid tapping cycle 6-76
- counter tapping cycle..... 6-73
- counterclockwise interpolation..... 6-9
- custom G code macro calls 6-5
- custom M code macro calls 7-1
- custom S code macro calls..... 8-1
- custom T code macro calls 9-1
- customized G codes 6-5
- customized M codes 7-1
- customized S codes..... 8-1
- customized T codes 9-1
- cutting feedrate for drilling, tapping or boring 2-1
- cutting mode 6-64
- cycle cancel 6-83
- d 6-72, 6-92
- D 2-1, 6-39
- data category..... 2-1, 6-24
- data index..... 2-2, 6-24
- data input 6-29
- data input categories 6-24
- data value..... 2-2
- data value for X axis 2-2
- data value for Y axis..... 2-3
- data value for Z axis 2-3
- decimal point 3-1
- default mode 6-1
- delay in program execution 6-21
- direct digital output..... 2-1, 10-1
- direct dout address..... 10-1
- direction control..... 6-41
- distance 3-1
- distance from bottom of hole to point Z 2-3
- distance to bottom of hole from point R 2-3
- DLACC 6-54
- drilling cycle..... 6-90

drilling cycle (dwell)	6-87	G43, G44	6-48, 9-1
drilling cycle, spot boring	6-84	G43.2, G49	6-51
duplicate parameters	1-3	G43/G44	6-34
dwell	6-21, 6-110	G49	6-48
Dwell	6-87	G50, G51	6-56
dwell time	2-2	G50.1, G51.1	6-58
dwell time at bottom of hole	2-2	G52	6-60
dwell time in seconds	2-2	G53	6-61
dynamic look-ahead contour control on / off	6-22	G54.1	6-63
end point feedrate for exponential interpolation	2-2	G54-G59	6-62
error allowance of the circular motion	2-2	G6.6 PLC skip signal	6-33
exact stop check mode	6-64	G61	6-64
exponential interpolation	6-19	G64	6-64
external workpiece zero point offset value	6-61	G64.1	6-64
F 2-1		G65	6-66
F codes	1-1	limitations	6-66
feed per minute	6-115	G68, G69	6-67
feed per revolution	6-115	G73	6-70, 6-83
feedrate	2-1, 3-1	G74	6-73, 6-76, 6-83
fine boring cycle	6-80	G76	6-80, 6-83
floating point values	1-1	G80 6-72, 6-75, 6-79, 6-83, 6-86, 6-89, 6-92, 6-95, 6-99, 6-102, 6-105, 6-109, 6-112	
format for distance and feedrate	3-1	G81	6-83, 6-84
format for part programs	3-1	G82	6-83, 6-87
G 2-1		G83	6-83, 6-90
G codes	1-1	G84	6-83, 6-93, 6-96
summary	6-2	G85	6-83, 6-100
G codes, customized	6-5	G86	6-83, 6-103
G00	6-6	G87	6-83, 6-106
G00 perform linear interpolation	6-7	G89	6-83, 6-110
G01	6-8	G90	6-7, 6-8, 6-10, 6-20, 6-31, 6-34, 6-59
G02	6-9, 6-18, 6-29	G90, G91	6-113
G02.3	6-19, 6-29	G91	6-7, 6-8, 6-10, 6-20, 6-31, 6-34, 6-59
G03	6-9, 6-18, 6-29	G92	6-114
G03.3	6-19, 6-29	G94	6-79, 6-98, 6-99, 6-115
G04	6-21	G95	6-79, 6-98, 6-99, 6-115
G05	6-22	G98 6-70, 6-72, 6-73, 6-75, 6-76, 6-79, 6-80, 6-83, 6-84, 6-86, 6-87, 6-89, 6-92, 6-93, 6-95, 6-96, 6-99, 6-100, 6-102, 6-103, 6-105, 6-109, 6-110, 6-112, 6-116	
G08	6-22	G99 6-70, 6-72, 6-73, 6-75, 6-76, 6-79, 6-80, 6-83, 6-84, 6-86, 6-87, 6-89, 6-92, 6-93, 6-95, 6-96, 6-99, 6-100, 6-102, 6-103, 6-105, 6-109, 6-110, 6-112, 6-116	
G10	6-24	G-code motion programming	1-1
G17	6-10	H 2-1	
G17, G18 and G19	6-29	helical interpolation	6-18
G18	6-10	helix angle	6-19
G19	6-10	helix angle in exponential interpolation	2-1
G20, G21	6-29	high-speed peck drilling cycle	6-70
G20/21	1-1	high-speed, precision machining	6-22
G201.4 address of PLC	6-34	hole position data for the X axis	2-2
G28, G29	6-30	hole position data for the Y axis	2-3
G30	6-32	I 2-1	
G31	6-33	in position width	6-25
G310, G311	6-117	inch / metric data input	6-29
G37	6-34		
G37 Limit	6-36		
G37 Ratio	6-37		
G37 Speed	6-38		
G40, G41 and G42	6-39		
G40, G41 or G42	9-1		
G40.1, G41.1 and G42.1	6-41		

incremental programming.....	6-113	miscellaneous codes	7-1
individual axis magnification factor	6-57	miscellaneous function	2-2
initial feedrate for exponential interpolation.....	2-1	miscellaneous functions.....	1-1
in-position check.....	6-64	modal commands.....	6-1
instruction	1-1	modal group.....	6-1
Integer Programming with Machine Unit Enable..	1-2	modal groups	6-1
integer values.....	1-1	mode	6-1
integers	1-2	motion programming language.....	1-1
intermittent feeding.....	6-70	N 2-2	
inverter spindle	6-76, 6-96, 8-2	N codes	1-1
J 2-1		negative exponential interpolation.....	6-19
K 2-1		negative numbers.....	3-1
L 2-1		nesting depth of subprogram calls	5-2
L10 – tool length compensation	6-24	no interpolation.....	6-6
L106 – smoothing (acc/dec) mode settings for rapid motion (G00)	6-26	NO_SMOOTHING	6-26
L107 – smoothing (acc/dec) mode settings for cutting motion (G01)	6-26	nonmodal	6-1
L108 – smoothing (acc/dec) time settings	6-27	nonmodal codes	6-1
L10909 – position loop gain settings.....	6-27	normal direction control	6-41
L11 – tool radius compensation.....	6-24	number of repetitions.....	5-1
L20 – axis type settings	6-25	number of times to repeat drilling, tapping or boring cycles	2-1
L30 – in position width.....	6-25	number of times to repeat macro	2-1
LadderWorks PLC sequence program.....	7-1	number of workpiece coordinate system to select ..	2-2
length geometry offset.....	9-1	numeral.....	1-1
Length Limit.....	6-43	numerals	1-1
length of a block of code	1-3	numeric interpretation.....	1-2
length wear offset	9-1	O 5-1	
linear interpolation.....	6-8, 6-113	offset number.....	9-1
linear interpolation feedrate include / exclude rotary axes	6-117	offset value	9-1
local coordinate system selection	6-60	one-shot codes	6-1
look-ahead function	6-22	optional skip code.....	3-2
lower case letters	3-1	order of operations.....	4-1
M 2-2		oriented spindle stop.....	6-81, 6-106
M code processing	7-1	overall magnification factor.....	6-57
M codes	1-1, 7-1	P 2-2	
M00	7-1	parentheses	3-2
M01	7-1	part program format.....	3-1
M02	3-1, 7-1	part programming language.....	1-1
M03	8-1	peck drilling cycle	6-70, 6-90
M04	8-1	plane selection	6-29
M05	8-1	PLC – G201.4 address.....	6-34
M29	6-76, 6-96	PLC sequence program.....	7-1
M30	3-1, 7-1	PLC skip signal.....	6-33
M98	5-1, 7-1	position loop gain settings	6-27
M99	5-1, 7-1	position loop gain value.....	2-2
machine coordinate system selection.....	6-61	positive exponential interpolation.....	6-19
machine origin	6-114	positive/negative tool length compensation/ compensation cancel	6-48
machine unit	1-1	precision machining.....	6-22
macro call	6-66	preparatory codes.....	1-1
macro number	2-2	preparatory function	2-1
magnification	6-56, 6-57	preparatory functions.....	6-1
maximum length of a block of code	1-3	program end.....	7-1
mirror image	6-40, 6-57	program end and rewind	7-1
mirror image off / on	6-58	program end code	3-1
		program format.....	3-1

program number	3-1	smoothing (acc/dec) time settings	6-27
program optional stop	7-1	smoothing mode	2-2
program rewind	7-1	smoothing time	2-2
program temporary stop	7-1	spindle	8-2
program title	3-1	spindle functions	8-1
programmable data input	6-24	Spindle Rigid Mode On	6-76, 6-96
programming	1-1	spindle speed (RPM)	8-1
Q 2-2, 6-72		spindle speed command	1-1
R 2-2		spindle speed function	2-2
radius compensations	6-39	spindle stop	6-103
radius geometry offset	9-1	spindle/tool shift within hole	6-81
radius wear offset	9-1	spiral	6-18
rapid feedrate	6-7	spot boring	6-84
rapid positioning	6-6	subprogram call from a main program	7-1
rapid traverse 6-72, 6-75, 6-78, 6-82, 6-85, 6-88, 6-92, 6-95, 6-98, 6-102, 6-105, 6-109, 6-112		subprogram functions	5-1
rapid traverse rate	6-6, 6-37	subprogram name	5-1
reference points	6-30, 6-32	subprogram name call	2-2
repeated callings	5-2	subprogram repetition	2-2
return to initial point in canned cycle	6-116	summary of G codes	6-2
return to main program	7-1	surface detection signal	6-34
return to R point in canned cycle	6-116	surface position	6-34
rewind	7-1	surface speed	8-1
Rigid Mode	6-76, 6-96	SWITCH AXES button	6-47, 6-54
rigid tapping	6-76	symmetrical part	6-58
rigid tapping cycle	6-96	T 2-2	
Rotary Axis No.	6-43	T codes	1-1, 9-1
Rotary axis type	6-54	taper angle in exponential interpolation	2-1
Rotation Feedrate	6-43	tapered groove machining	6-19
S 2-2		tapping	6-73, 6-76, 6-96
S codes	8-1	tapping cycle	6-93
scaling	6-40, 6-59	TCP	6-51
scaling factor for all axes	2-2	theoretical surface position	6-34
scaling factor for X axis	2-1	thread lead	6-98
scaling factor for Y axis	2-1	three-dimensional dynamic look-ahead contour control on / off	6-22
scaling factor for Z axis	2-1	Tool Center Point Control	6-51
scaling off / on	6-56	tool diameter code	2-1
separate line	1-3	tool function	2-2
sequence number	1-1, 2-2	tool functions	9-1
sequence program	7-1	Tool Length	6-53
ServoWorks part programming language	1-1	tool length compensation	6-24, 6-34
Settings Mode Window	6-54	Tool Length Compensation Type	6-35, 6-38, 6-50
sharp edges	6-65	tool length compensation/ compensation cancel ..	6-48
shrinking	6-56	tool length offset	6-48
signal, surface detection	6-34	tool length offset type	6-48, 6-50
simple macro call	6-66	tool length offset value	6-34
skip cutting	6-33	tool offsets	9-1
skip signal	6-33	tool orientation	6-41
small segment contour control	6-64	tool radius compensation	6-24, 6-57, 6-59, 9-1
SMOOTH_BELLSHAPE	6-26	tool radius compensations	6-39
SMOOTH_EXPONENTIAL	6-26	tool radius offset	6-39, 9-1
SMOOTH_LINEAR	6-26	tool selection command	1-1
smoothing (acc/dec) mode settings for cutting motion (G01)	6-26	TRC	6-39, 9-1
smoothing (acc/dec) mode settings for rapid motion (G00)	6-26	unused data	3-2
		Velocity Control in Circular Interpolation function ..	6-54

word.....	1-1	XY/ZX/YZ plane selection.....	6-29
words	1-1	Y 2-3	
workpiece	6-114	Y axis absolute or incremental coordinate value	
workpiece coordinate programming	6-114	designation	2-3
workpiece coordinate selection.....	6-62	Y axis tool length compensation.....	2-3
workpiece coordinate system.....	6-60	Y surface position.....	2-3
workpiece coordinate systems	6-63	Z 2-3	
workpiece zero point offset value.....	6-61	Z axis absolute or incremental coordinate value	
X 2-2		designation	2-3
X axis absolute or incremental coordinate value		Z axis tool length compensation	2-3
designation	2-2	Z surface position	2-3
X axis tool length compensation.....	2-2	zero point	6-114
X surface position.....	2-2		