User Manual

SZGH-CNC990TDc(series) Lathe Control System -PROGRAMMING- V2.2

BOOK1

Shenzhen Guanhong Automation CO.,LTD
Website: www.szghauto.com
Add:Room 503 Anxin Building, No 536 Shenhui Road, Liuyue community, Henggang Street ,Longgang District, Shenzhen City,Guangdong Province, ChinaProvince, China
Post code: 518115

Warnings and Notes as Used in this Publication

Warning

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use.
In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.

Caution

Caution notices are used where equipment might be damaged if care is not taken.

Note

Notes merely call attention to information that is especially significant to understanding and operating the equipment.

This document is based on information available at the time of its publication. While efforts have been made to be accurate, the information contained herein does not purport to cover all details or variations in hardware or software, nor to provide for every possible contingency in connection with installation, operation, or maintenance. Features may be described herein which are not present in all hardware and software systems. GE Fanuc Automation assumes no obligation of notice to holders of this document with respect to changes subsequently made.

Shenzhen Guanhong Automation 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 warranties of merchantability or fitness for purpose shall apply.

©Copyright 2016 Shenzhen Guanhong Automation Co.,Ltd.
All Rights Reserved.

SAFETY PRECAUTIONS

This section describes the safety precautions related to the use of CNC units. It is essential that these precautions be observed by users to ensure the safe operation of machines equipped with a CNC unit (all descriptions in this section assume this configuration). Note that some precautions are related only to specific functions, and thus may not be applicable to certain CNC units.

Users must also observe the safety precautions related to the machine, as described in the relevant manual supplied by the machine tool builder. Before attempting to operate the machine or create a program to control the operation of the machine, the operator must become fully familiar with the contents of this manual and relevant manual supplied by the machine tool builder.

1 Definition of Warning , Caution, and Note

This manual includes safety precautions for protecting the user and preventing damage to the machine. Precautions are classified into Warning and Caution according to their bearing on safety. Also, supplementary information is described as a Note. Read the Warning, Caution, and Note thoroughly before attempting to use the machine.

WARNING

Applied when there is a danger of the user being injured or when there is a danger of both the user being injured and the equipment being damaged if the approved procedure is not observed.

CAUTION

Applied when there is a danger of the equipment being damaged, if the approved procedure is not observed.

NOTE

The Note is used to indicate supplementary information other than Warning and Caution.

Read this manual carefully, and store it in a safe place！

2 GENERAL WARNINGS AND CAUTIONS

Warning

1. Never attempt to machine a workpiece without first checking the operation of the machine.
Before starting a production run, ensure that the machine is operating correctly by performing a trial run using, for example, the single block, feedrate override, or machine lock function or by operating the machine with neither a tool nor workpiece mounted. Failure to confirm the correct operation of the machine may result in the machine behaving unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.
2. Before operating the machine, thoroughly check the entered data. Operating the machine with incorrectly specified data may result in the machine behaving unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.
3. Ensure that the specified feedrate is appropriate for the intended operation. Generally, for each machine, there is a maximum allowable feedrate. The appropriate feedrate varies with the intended operation. Refer to the manual provided with the machine to determine the maximum allowable feedrate. If a machine is run at other than the correct speed, it may behave unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.
4. When using a tool compensation function, thoroughly check the direction and amount of Compensation. Operating the machine with incorrectly specified data may result in the machine behaving unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.
5. The parameters for the CNC and PMC are factory-set. Usually, there is not need to change them. When, however, there is not alternative other than to change a parameter, ensure that you fully understand the function of the parameter before making any change. Failure to set a parameter correctly may result in the machine behaving unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.
6. Immediately after switching on the power, do not touch any of the keys on the MDI panel until the position display or alarm screen appears on the CNC unit. Some of the keys on the MDI panel are dedicated to maintenance or other special operations. Pressing any of these keys may place the CNC unit in other than its normal state. Starting the machine in this state may cause it to behave unexpectedly.
7. The operator's manual and programming manual supplied with a CNC unit provide an overall description of the machine's functions, including any optional functions. Note that the optional functions will vary from one machine model to another. Therefore, some functions described in the manuals may not actually be available for a particular model. Check the specification of the machine if in doubt.
8. Some functions may have been implemented at the request of the machine–tool builder. When using such functions, refer to the manual supplied by the machine–tool builder for details of their use and any related cautions.

NOTE: Programs,parameters, and macro variables are stored in nonvolatile memory in the CNC unit. Usually, they are retained even if the power is turned off. Such data may be deleted inadvertently, however, or it may prove necessary to delete all data from nonvolatile memory as part of error recovery.To guard against the occurrence of the above, and assure quick restoration of deleted data, backup all vital data, and keep the backup copy in a safe place.

3 WARNINGS AND CAUTIONS RELATED TO PROGRAMMING

This section covers the major safety precautions related to programming. Before attempting to perform programming, read the supplied operator's manual and programming manual carefully such that you are fully familiar with their contents.

Warning

1.Coordinate system setting

If a coordinate system is established incorrectly, the machine may behave unexpectedly as a result of the program issuing an otherwise valid move command.

Such an unexpected operation may damage the tool, the machine itself, the workpiece, or cause injury to the user.

2. Positioning by nonlinear interpolation

When performing positioning by nonlinear interpolation (positioning by nonlinear movement between the start and end points), the tool path must be carefully confirmed before performing programming. Positioning involves rapid traverse. If the tool collides with the workpiece, it may damage the tool, the machine itself, the workpiece, or cause injury to the user.

3. Function involving a rotation axis

When programming polar coordinate interpolation or normal–direction (perpendicular) control, pay careful attention to the speed of the rotation axis. Incorrect programming may result in the rotation axis speed becoming excessively high, such that centrifugal force causes the chuck to lose its grip on the workpiece if the latter is not mounted securely. Such mishap is likely to damage the tool, the machine itself, the workpiece, or cause injury to the user.

4. Inch/metric conversion

Switching between inch and metric inputs does not convert the measurement units of data such as the workpiece origin offset, parameter, and current position. Before starting the machine, therefore, determine which measurement units are being used. Attempting to perform an operation with invalid data specified may damage the tool, the machine itself, the workpiece, or cause injury to the user.

5. Constant surface speed control

When an axis subject to constant surface speed control approaches the origin of the workpiece coordinate system, the spindle speed may become excessively high. Therefore, it is necessary to specify a maximum allowable speed. Specifying the maximum allowable speed incorrectly may damage the tool, the machine itself, the workpiece, or cause injury to the user.

6. Stroke check

After switching on the power, perform a manual reference position return as required. Stroke check is not possible before manual reference position return is performed. Note that when stroke check is disabled, an alarm is not issued even if a stroke limit is exceeded, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the user.

7. Absolute/incremental mode

If a program created with absolute values is run in incremental mode, or vice versa, the machine may behave unexpectedly.

8. Plane selection

If an incorrect plane is specified for circular interpolation, helical interpolation, or a canned cycle, the machine may behave unexpectedly. Refer to the descriptions of the respective functions for details.

9. Compensation function

If a command based on the machine coordinate system or a reference position return command is issued in compensation function mode, compensation is temporarily canceled, resulting in the unexpected behavior of the machine. Before issuing any of the above commands, therefore, always cancel compensation function mode.

4 WARNINGS AND CAUTIONS RELATED TO HANDLING

This section presents safety precautions related to the handling of machine tools. Before attempting to operate your machine, read the supplied operator's manual and programming manual carefully, such that you are fully familiar with their contents.

Warning

1. Manual operation

When operating the machine manually, determine the current position of the tool and workpiece, and ensure that the movement axis, direction, and feedrate have been specified correctly. Incorrect operation of the machine may damage the tool, the machine itself, the workpiece, or cause injury to the operator.

2. Manual reference position return

After switching on the power, perform manual reference position return as required. If the machine is operated without first performing manual reference position return, it may behave unexpectedly. Stroke check is not possible before manual reference position return is performed. An unexpected operation of the machine may damage the tool, the machine itself, the workpiece, or cause injury to the user.

3. Manual handle feed

In manual handle feed, rotating the handle with a large scale factor, such as 100, applied causes the tool and table to move rapidly. Careless handling may damage the tool and/or machine, or cause injury to the user.

4. Disabled override

If override is disabled (according to the specification in a macro variable) during threading or other tapping, the speed cannot be predicted, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the operator.

5. Origin/preset operation

Basically, never attempt an origin/preset operation when the machine is operating under the control of a program. Otherwise, the machine may behave unexpectedly, possibly damaging the tool, the machine itself, the tool, or causing injury to the user.

6. Workpiece coordinate system shift

Manual intervention, machine lock, or mirror imaging may shift the workpiece coordinate system. Before attempting to operate the machine under the control of a program, confirm the coordinate system carefully. If the machine is operated under the control of a program without making allowances for any shift in the workpiece coordinate system, the machine may behave unexpectedly, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the operator.

7. Software operator's panel and menu switches

Using the software operator' s panel and menu switches, in combination with the MDI panel, it is possible to specify operations not supported by the machine operator' s panel, such as mode change, override value change, and jog feed commands.
Note, however, that if the MDI panel keys are operated inadvertently, the machine may behave

unexpectedly, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the user.

8. Manual intervention

If manual intervention is performed during programmed operation of the machine, the tool path may vary when the machine is restarted. Before restarting the machine after manual intervention, therefore, confirm the settings of the manual absolute switches, parameters, and absolute/incremental command mode.

9. Feed hold, override, and single block

The feed hold, feedrate override, and single block functions can be disabled using custom macro system variable #3004. Be careful when operating the machine in this case.

10. Dry run

Usually, a dry run is used to confirm the operation of the machine. During a dry run, the machine operates at dry run speed, which differs from the corresponding programmed feedrate. Note that the dry run speed may sometimes be higher than the programmed feed rate.

11. Cutter and tool nose radius compensation in MDI mode

Pay careful attention to a tool path specified by a command in MDI mode, because tool nose radius compensation is not applied. When a command is entered from the MDI to interrupt in automatic operation in tool nose radius compensation mode, pay particular attention to the tool path when automatic operation is subsequently resumed. Refer to the descriptions of the corresponding functions for details.

12. Program editing

If the machine is stopped, after which the machining program is edited (modification, insertion, or deletion), the machine may behave unexpectedly if machining is resumed under the control of that program. Basically, do not modify, insert, or delete commands from a machining program while it is in use.

5 WARNINGS RELATED TO DAILY MAINTENANCE

WARNING

1. Memory backup battery replacement

When replacing the memory backup batteries, keep the power to the machine (CNC) turned on, and apply an emergency stop to the machine. Because this work is performed with the power on and the cabinet open, only those personnel who have received approved safety and maintenance training may perform this work.

When replacing the batteries, be careful not to touch the high－voltage circuits (marked and fitted with an insulating cover).Touching the uncovered high－voltage circuits presents an extremely dangerous electric shock hazard.

NOTE:The CNC uses batteries to preserve the contents of its memory, because it must retain data such as programs, offsets, and parameters even while external power is not applied. If the battery voltage drops, a low battery voltage alarm is displayed on the machine operator's panel or screen. When a low battery voltage alarm is displayed, replace the batteries within a week. Otherwise, the contents of the CNC's memory will be lost. Refer to the maintenance section of the operator's manual for details of the battery replacement procedure.

2. Absolute pulse coder battery replacement

When replacing the memory backup batteries, keep the power to the machine (CNC) turned on, and apply an emergency stop to the machine. Because this work is performed with the power on and the cabinet open, only those personnel who have received approved safety and maintenance training may perform this work. When replacing the batteries, be careful not to touch the high－voltage circuits (marked and fitted with an insulating cover). Touching the uncovered high－voltage circuits presents an extremely dangerous electric shock hazard.

NOTE:The absolute pulse coder uses batteries to preserve its absolute position. If the battery voltage drops, a low battery voltage alarm is displayed on the machine operator's panel or screen. When a low battery voltage alarm is displayed, replace the batteries within a week. Otherwise, the absolute position data held by the pulse coder will be lost.

3. Fuse replacement

For some units, the chapter covering daily maintenance in the operator’s manual or programming manual describes the fuse replacement procedure.
Before replacing a blown fuse, however, it is necessary to locate and remove the cause of the blown fuse. For this reason, only those personnel who have received approved safety and maintenance training may perform this work.
When replacing a fuse with the cabinet open, be careful not to touch the high－voltage circuits (marked and fitted with an insulating cover).
Touching an uncovered high－voltage circuit presents an extremely dangerous electric shock hazard.

CONTENTS

SAFETY PRECAUTIONS.....II
1 Definition of Warning , Caution, and Note.....II
2 GENERAL WARNINGS AND CAUTIONS.....III
3 WARNINGS AND CAUTIONS RELATED TO PROGRAMMING.....IV
4 WARNINGS AND CAUTIONS RELATED TO HANDLING.....V
5 WARNINGS RELATED TO DAILY MAINTENANCE.....VI
Chapter 1 Preface.....1
1.1 Characteristics.....2
1.2 Technical Specifications.....2
1.3 G Code List.....4
1.4 System operation condition.....5
Chapter 2 Programming.....6
2.1 Basic Concept of Programming.....7
2.1.1 Definition of Coordinate System.....7
2.1.2 Machine Coordinate System and Machine Reference Point.....8
2.1.3 Workpiece Coordinate System and Program Reference Point.....8
2.1.4 Coordinate System.....9
2.1.5 Interpolation.....10
2.1.6 Absolute Programming & Incremental Programming.....12
2.1.7 Diameter Programming & Radius Programming.....13
2.1.8 Cutting Speed & Spindle Speed.....14
2.1.9 Tool Function.....15
2.1.10 Command For Machine Operations.....15
2.2 Configuration of Program.....16
2.3 Main Program & Subprogram.....19
2.4 Program Run.....19
Chapter 3 G INSTRCUTIONS.....20
3.1 INTRODUCTION.....20
3.2 G Code List.....20
3.3 Positioning (Rapid Traverse) (G00).....22
3.4 Linear Interpolation (G01).....24
3.5 Circular Interpolation (G02/G03).....24
3.6 Thread Cutting (G32).....28
3.6.1 Constant Lead Threading.....28
3.6.2 Continous Thread Cutting.....29
3.6.3 Thread Cutting With Variable Lead.....30
3.7 Circular Thread Cutting(G332/G333).....30
3.8 Canned Cycle(G90,G92,G93,G94).....30
3.8.1 Outer Diameter/Internal Diameter Cutting Cycle (G90).....30
3.8.2 Thread Cutting Cycle (G92).....33
3.8.3 Canned Tapping Cycle (G93).....36
3.8.4 End Face Turning Cycle G94.....38

3.8.5 Usage for Canned Cycle....41
3.9 Multiple Repetitive Cycle Instructions(G70~G76)....42
3.9.1 Axial Roughing Turning Cycle (G71)....42
3.9.2 Radial Roughing Facing Cycle (G72)....45
3.9.3 Pattern Repeating Cycle (G73)....46
3.9.4 Finishing Cycle (G70)....48
3.9.5 Usages of G71,G72,G73 & G70....48
3.9.5.1 Example of G71&G70....48
3.9.5.3 Example of G73&G70....49
3.9.6 End Face Peck Drilling Cycle (G74)....51
3.9.7 Outer Diameter/Internal Diameter Drilling Cycle (G75)....52
3.9.8 Multiple Thread Cutting Cycle (G76)....54
3.9.9 Notes On Multiple Repetitive Cycle (G70 ~ G76)....58
3.10 Skip Function(G31,G311)....58
3.11 Block Cycle (G22,G800)....59
3.12 Return to Starting Point (G26,G261~G264)....60
3.13 Save Current Position (G25)....60
3.14 Return to Specified Position (G61,G611~G614)....60
3.15 Return to Reference Position (G28)....61
3.16 Coordinate System....62
3.16.1 Machine Coordiante System (G53)....62
3.16.2 Workpiece Coordinate System....63
3.16.3 Setting a Workpiece Coordinate System(G50)....63
3.16.4 Selecting a Workpiece Coordinate System....64
3.16.5 Changing Workpiece Coordinate System....65
3.16.6 Workpiece Coordinate System Shift....66
3.16.7 Local Coordinate System....67
3.17 Constant Surface Speed Control (G96/G97)....68
3.18 Cutting Feed (G98,G99)....70
3.19 Pole Coordinate Interpolation (G15/G16)....71
3.20 Absolute and Incremental Programming (G990,G991)....72
3.21 Inch/Metric Conversion (G20/G21)....73
3.22 Dwell (G04)....74
3.23 Positioning/Continous Path Processing(G60/G64)....74
3.24 Workpiece Position and Move Command....75
3.25 Details of Tool Nose Radius Compensatation (G40/G41/G42)....78
3.26 Tool Nose Radius Compensation of Offset C....79
3.27 Automatical beveling (I) and smoothing(R)....84
3.28 3D Space Arc Interpolation G06....85
3.29 Macro program instruction(G65/G66/G67)....86
3.29.1 Non-Mode Macro Command G65....86
3.29.2 Mode Macro Command G66/G67....86
3.29.3 Macro Program Instruction....87
3.29.3.1 Input Instruction: WAT....87

3.29.3.2 Output Instruction: OUT 87
3.29.3.3 Assignment Instruction: = ... 88
3.29.3.4 Unconditional Jump: GOTO n 88
3.29.3.5 Conditional Jump 88
3.29.3.6 Loop Command 89
3.29.4 Operators' meaning 90
3.29.5 Arithmetic & Logic Operation 90
3.29.6 Local Variable 91
3.29.7 Global Variable 91
3.29.8 System Variable 91
3.29.9 System Parameter Variable 91
3.29.10 I/O variable 91
Chapter 4 M INSTRCUTIONS 93
4.1 M Function (Auxiliary Function) 93
4.1.1 Program Stop(M00) 93
4.1.2 Optional Stop (M01) 93
4.1.3 End of Program (M02,M30) 93
4.1.4 Cycle of Program (M20) 93
4.1.5 Account of Workpiece(M87) 93
4.1.6 Uncondtional Jump (M97) 93
4.2 Subprogram Configuration 93
4.2.1 Calling of Subprogram (M98) 94
4.2.2 End of Subprogram (M99) 94
4.3 Standard PLC M Command List 95
4.3.1 M Output Command List 95
4.3.1.1 Spindle Control (M03/M04/M05) 96
4.3.1.2 Spindle Gear Shifting(M41/M42/M43/M44) 97
4.3.1.3 Coolant(M08/M09) 97
4.3.1.4 Lubricate(M32/M33) 98
4.3.1.5 Chuck(M10/M11) 98
4.3.1.6 Tailstock(M79/M78) 98
4.3.1.7 Condition Output of Machine Tool(M65/M67/M69) 99
4.3.2 M Input Command List 99
4.4 Analog Speed of Spindle(S , SS) 100
4.5 T Tool Function Command 100
4.6 User-defined macro instruction(G120-G160,M880-M889) 101
4.7 Synthetic instance for programming 102

Chapter 1 Preface

CNC machine tool is an electro-mechanical integrated product, composed of Numerical Control Systems of Machine Tools, machines, electric control components, hydraulic components, pneumatic components, lubricant, coolant and other subsystems (components), and CNC systems of machine tools are control cores of CNC machine tools. CNC systems of machine tools are made up of computerized numerical control(CNC), servo (stepper) motor drive devices, servo (or stepper) motor and etc.

Operational principles of CNC machine tools: according to requirements of machining technology, edit user programs and input them to CNC, then CNC outputs motion control instructions to the servo (stepper) motor drive devices, and last the servo (or stepper) motor completes the cutting feed of machine tool by mechanical driving device; logic control instructions in user programs to control spindle start/stop, tool selections, coolant ON/OFF, lubricant ON/OFF are output to electric control systems of machine tools from CNC, and then the electric control systems control output components including buttons, switches, indicators, relays, contactors and so on. Presently, the electric control systems are employed with Programmable Logic Controller (PLC) with characteristics of compact,convenience and high reliance. Thereof, the motion control systems and logic control systems are themain of CNC machine tools.

SZGH-CNC990TDc series CNC control system is economic type CNC control system for lathe machine , which is developed by Shenzhen Guanhong Automation Co.,Ltd. And we have already made updates basic original CNC990TDc CNC Controller.

Fig1.1 SZGH-CNC990TDc

1.1 Characteristics

- 1) 800*600 8.4 inch real color LCD Display with fully english
- 2) Support ATC function , Macro function and PLC function
- 3) Electric Turret & Binary code Turret & Special Turret, Max: 99 Pcs of tools
- 4) 128MB Memory , 100Mb user store room(Updated!)
- 5) 5MHz Pulse Output Frequency, Max speed is 300m/min(Updated!)
- 6) PLC On-line Display,Monitor & Design(Updated!)
- 7) High anti-jamming switch power(220VAC -> 24VDC & 5VDC)
- 8) Built-in plc programs, which can be edited freely.
- 9) With USB interface, for upgrade & copy programs
- 10) Analog voltage output of 0~10V in two channels, support double spindles
- 11) Adapted servo spindle can realize position,rigid tapping,threading of spindle
- 12) Basic I/Os : 40*24
- 13) Built-in screw compensation
- 14) Support Position Feedback Function & Linear Scale
- 15) Support Absolute type & Increment type servo system
- 16) Support macro variable dialog box & Running program by input point (Updated!)

1.2 Technical Specifications

Max Number of control axes

• Number of control axes: 4 axes (X Z Y(C) A)
• Number of linkage axes: 4 axes
• Number of PLC control axes: 4 axes

Feeding axes function

• Minimum command unit: 0.001mm
• Position command range: +/- 99999.999
• Max speed: 30 m/minFeeding speed:0.001-15m/min
• G00 rapid override: Total 8 levels: 0~150%,real-time adjusting
• Feeding override: Total 16 levels: 0~150%,real-time adjusting
• Spindle override: Total 16 levels: 5%~150%,real-time adjusting
• Interpolation mode: Interpolation of linear ,arc ,thread and rigid tapping
• Auto chamfering

Thread

• Acceleration and deceleration funciton
• Common thread(follow the spindle) / Rigid thread
• Single-headed/Multi thread of straight ,taper and therminal suface in metric system/inch system,wqual and variable pitch thread
• Thread retractin length ,angle and speed characteristics can be set
• Thread pitch:0.1~1000.000mm or 0.1~99 tooth/inch
• Rapid traverse: linear type or S type
• The starting speed,finishing speed and time of acceleration and deceleration are set by parameter

Spindle function

• Analog voltage 0~10V output in two channels ,support two-spindle control

• Spindle encoder feedback in one channel,resolution of spindle encoder can be set
• Spindle speed: It is set by speed parameter,max spindle speed also coressponding to 10V
• Spindle override: Total 16 levels: 5%~150%,real-time adjusting
• Spindle constant surface speed control
• Ragid tapping

Tool Function

•Tool length compensation
• tool nose radius compensation (C type)
• Tool wearing compensation
• Method of setting tools: Tool-setting in fixed position ,trial cutting tool -setting,,auto tool setting
• Tool offset exexuting mode: Rewriting coordinate mode,tool traverse mode

Precision compensation

• Backlash compensation/Pitch error compensation in memory type
• Built-in Thread Compensation

PLC function

• Refresh cycle: 8ms
• PLC program can be altered on PC , download by USB interface
• I/Os : 40*24 I/Os
• Support On-line display,monitor & alter ladder

Man-machine interface

• 8.4" large screen real-color LCD , the resolution is 480 000
• Display in Chinese or English
• Display in two-dimensional tool path
• Real-time clock
• Operation management
• Operate mode:Auto, Manual, MDI, mechanical zero return, MPG/single step。
• Operation authority of multiple-level management
• Alarm record

Edit program

• Program capacity: 128M
• Editing function: program/block/characters research ,rewriting and deleting
• Program format: ISO code,support Macro command programming, programming of relative coordinate ,absolute coordinate and hybrid coordinate
• Calling program: Support macro program ,subprogram

Community function

• RS232: Files of part program can be transmitted
• USB: File operationand file directly processing in flash disk,support PLC programs,flash disk of software upgrade.

Safety function

• Emergency stop
• Hardware travel limit
• Software travel limit
• Data restoring and recovering
• User-defined alarm hint

1.3 G Code List

CODE	Description	CODE	Description
G00	Rapid Positioning	G17	XY plane selection
G01	Linear Interpolation	G18	ZX plane selection
G02	Circular Interpolation CW	G19	YZ plane selection
G03	Circular Interpolation CCW	G65	Macro command non-mode calling
G32	Threading Cutting	G66	Macro command mode calling
G31	Jumping function	G67	Macro program mode calling calling
G311	Jumping function	G40	Tool nose radius compensation cancel
G70	Finishing Cycle	G41	Tool nose radius left compensation
G71	Axial Roughing in Cycle	G42	Tool nose radius right compensation
G72	Radial Roughing in cycle	G26	Return to starting point of program
G73	Close Cutting Cycle	G261	X-axis Return to starting point of program
G74	Axial Grooving Cycle	G262	Y-axis Return to starting point of program
G75	Radial Grooving Cycle	G263	Z-axis Return to starting point of program
G76	Multiple Thread cutting cycle	G264	A-axis Return to starting point of program
G90	Axial Cutting cycle	G265	B-axis Return to starting point of program
G92	Thread cutting cycle	G25	Save curret coordinate value
G93	Canned Tapping Cycle	G61	Return to position of G25
G94	Radial Cutting Cycle	G611	X-axis return to position of G25
G22	Program Cycle	G612	Y-axis return to position of G25
G800	Program Cycle Cancel	G613	Z-axis return to position of G25
G15	Polar coordinate command cancel	G614	A-axis return to position of G25
G16	Polar coordinate command	G615	B-axis return to position of G25
G990	Absolute value programming	G28	Return to home of machine
G991	Incremental value programming	G281	X-axis return to home of machine
G20	Inch input	G282	Y-axis return to home of machine
G21	Millimeter input	G283	Z-axis return to home of machine
G04	Dwell	G284	A-axis return to home of machine
G60	Exact stop & position	G285	B-axis return to home of machine
G64	Continous track processing	M800	C-axis return to zero position of SP-encoder
G50	Set max speed of spindle	M801	C-axis ready to stop
G52	Set local coordinate system	G53	Machine coordinate system
G184	Setup/Offset curret coordinte value	G54	Workpiece coordinate system 1
G185	Set/Offset all coordinate value	G55	Workpiece coordinate system 2
G96	Constant surface speed control	G56	Workpiece coordinate system 3
G97	Constant surface speed control cancel	G57	Workpiece coordinate system 4
G98	Feeding/min	G58	Workpiece coordinate system 5
G99	Feeding/rev	G59	Workpiece coordinate system 6
G06	3D Space Arc Interpolation	Note: System also includes M codes&Other codes.	

1.4 System operation condition

1) Power supplying

AC 220V(+10%/-15%), Frequency 50Hz±2%. Power:≤ 200W.

Note: it must use isolation transform to supply power first input:380V

2) Climate condition

Item	Working Conditions	Storage&Delivery Conditions
Ambient Temperature	0℃~45℃	-40℃~+70℃
Ambient Humidity	≤95%(no freezing)	≤95%(40℃)
Atmosphere Pressure	86kPa~106kPa	86kPa~106kPa
Altitude	≤1000m	≤1000m

3) operation environment :

No excessive flour dust, no acid, no alkali gas and explosive gas, no strong electromagnetic interference.

1.5 Wiring Sketch

Fig1.2 Wiring Sketch for Total CNC Lathe System

Chapter 2 Programming

CNC machine tool is an electro-mechanical integrated product, composed of Numerical Control Systems of Machine Tools, machines, electric control components, hydraulic components, pneumatic components, lubricant, coolant and other subsystems (components), and CNC systems of machine tools are control cores of CNC machine tools. CNC systems of machine tools are made up of computerized numerical control(CNC), servo (stepper) motor drive devices, servo (or stepper) motor and etc.

Software used for controlling SZGH-CNC990TDc Turning Machine CNC system is divided into system software (NC for short) and PLC software (PLC for short). NC system is used for controlling display, communication, edit, decoding, interpolation and acceleration/deceleration, and PLC system for controlling explanations, executions, inputs and outputs of ladder diagrams.

Programming is a course of workpiece contours, machining technologies, technology parameters and tool parameters being edit into part programs according to special CNC programming instructions.CNC machining is a course of CNC controlling a machine tool to complete machining of workpiece according requirements of part programs. Technology flow of CNC machining is as following:

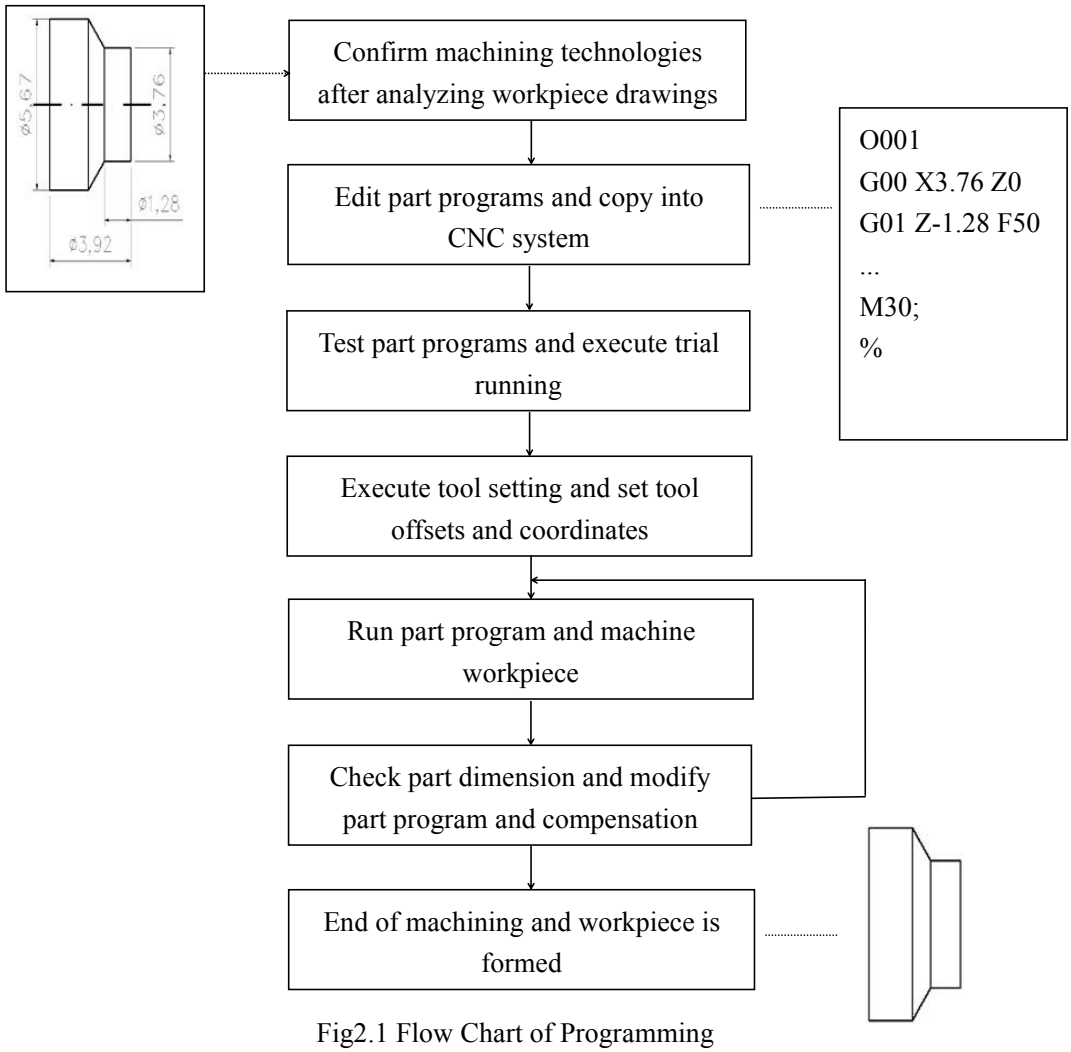


Fig2.1 Flow Chart of Programming

2.1 Basic Concept of Programming

2.1.1 Definition of Coordinate System

Sketch map of CNC turning machine is as following:

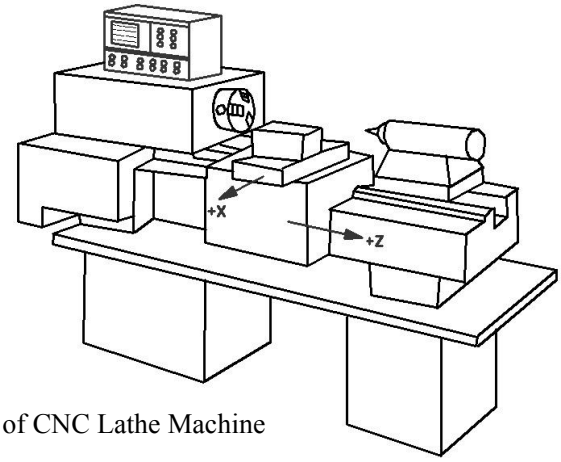


Fig2.2 Sketch Map of CNC Lathe Machine

Operational principles of CNC machine tools: according to requirements of machining technology, edit user programs and input them to CNC, then CNC outputs motion control instructions to the servo (stepper) motor drive devices, and last the servo (or stepper) motor completes the cutting feed of machine tool by mechanical driving device; logic control instructions in user programs to control spindle start/stop, tool selections, coolant ON/OFF, lubricant ON/OFF are output to electric control systems of machine tools from CNC, and then the electric control systems control output components including buttons, switches, indicators, relays, contactors and so on. Presently, the electric control systems are employed with Programmable Logic Controller (PLC) with characteristics of compact, convenience and high reliance. Thereof, the motion control systems and logic control systems are the main of CNC machine tools.

CNC system is employed with a rectangular coordinate system composed of X, Z axis. X axis is perpendicular with axes of spindle and Z axis is parallel with axes of spindle; direction of approach to the workpiece is negative direction and direction are away from workpiece is positive direction.

According to their relative position between the toolpost and spindle, there are 2 kinds of turning coordinate system; one is Front toolpost coordinate sytem(Fig2.3) , the other is Rear toolppost coordinate system(Fig2.4).It shows exactly the opposite direction in X direction but the same direction in Z direction from figures. In the manual, following figures and examples are based on Front toolpost coordinate sytem. **Type of CNC Lathe Machine: P3 in Tool parameter.**

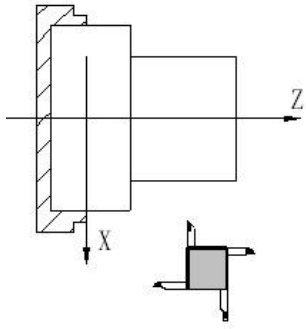


Fig2.3 Front toolpost coordinate system

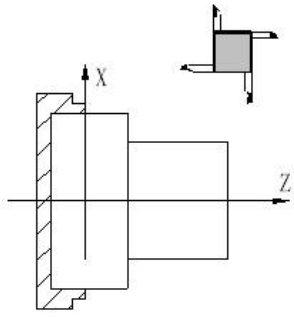


Fig2.4 Rear toolpost coordinate system

2.1.2 Machine Coordinate System and Machine Reference Point

Machine tool coordinate system is a benchmark one used for CNC counting coordinates and a fixed point on the machine tool.

Machine tool origin is named **machine reference point , machine zero** or **home**, which is specified by a reference point return switch on the machine tool. Usually, the reference point return switch is installed on max stroke in X, Z positive direction. The system considers the current coordinates of machine tool as zeroes and sets the machine tool coordinate system according to the current position as the coordinate origin after having executed the machine reference point return.

Machine Reference position is offset point based on machine zero.Offset value is set by P32(X-axis) & P33(Z-axis) in Axis parameter. If P32&P33=0, machine zero & reference position is same. Normally, tool change and programming of absolute zero point as described later are performed at this position.

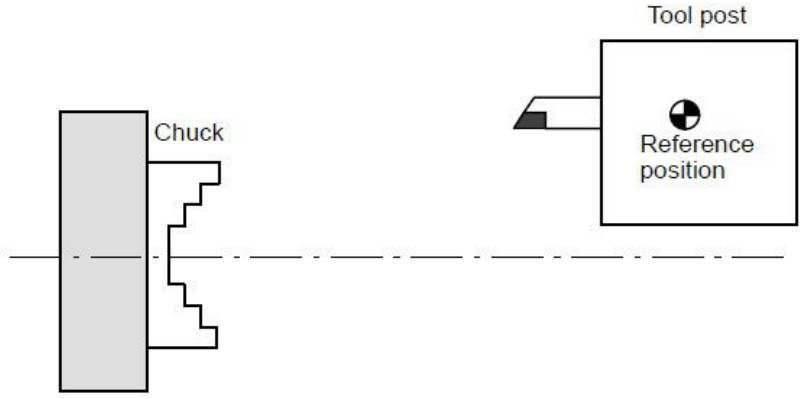


Fig2.5 Reference Position

The system considers the current coordinates of machine tool as zeroes and sets the machine tool coordinate system according to the current position as the coordinate origin after having executed return of machine reference point.

Note: Do not execute the machine reference point return without the reference point switch installed on the machine tool , otherwise movement over limitation of stroke , and broke machine.

2.1.3 Workpiece Coordinate System and Program Reference Point

Workpiece coordinate system is set to a rectangular coordinate system according to part drawings , also named **floating coordinate system**. After the workpiece is clamped on the machine tool, G50 is executed to set an absolute coordinates of tool's current position according to the relative position of tool and workpiece, and so the workpiece system has been created. The current position of tool is named **program reference point** and the tool returns to the position after executing the program reference point return. Usually, Z axis is consistent with the axes of spindle and X axis is placed on the heading or the ending of workpiece. The workpiece will be valid until it is replaced by a new one.

The current position of workpiece coordinate system set by G50 is named the **program reference point** and the system returns to it after executing the program reference point return.

Note: Do not execute the program reference point return without using G50 to set the workpiece coordinate system after power on.

2.1.4 Coordinate System

Coordinate system on part drawing and coordinate specified by CNC coordinate system.

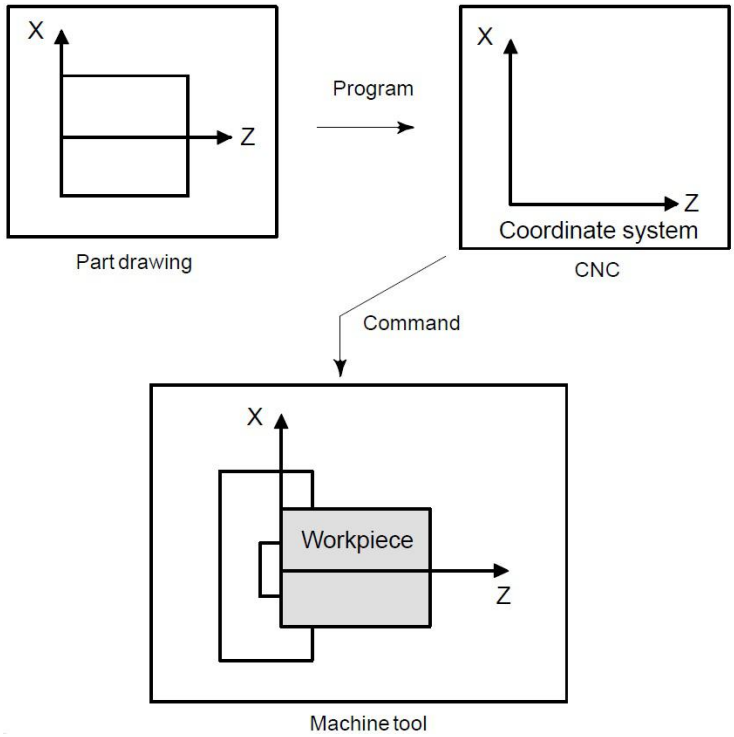


Fig2.6 Coordinate system

The following two coordinate systems are specified at different locations:

1. Coordinate system on part drawing

The coordinate system is written on the part drawing. As the program data, the coordinate values on this coordinate system are used.

2. Coordinate system specified by the CNC

The coordinate system is prepared on the actual machine tool. This can be achieved by programming the distance from the current position of the tool to the zero point of the coordinate system to be set.

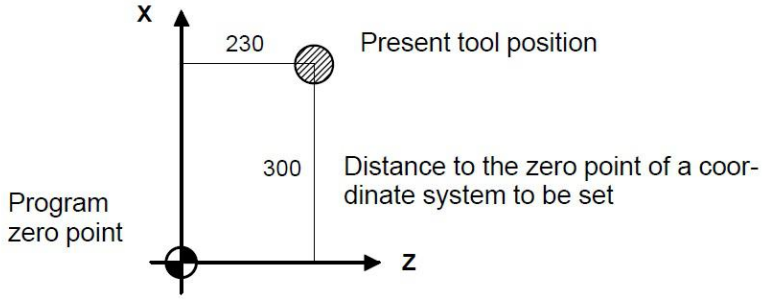


Fig2.7 Coordinate system specified by the CNC

The tool moves on the coordinate system specified by the CNC in accordance with the command program generated with respect to the coordinate system on the part drawing, and cuts a workpiece into a shape on the drawing.

Therefore, in order to correctly cut the workpiece as specified on the drawing, the two coordinate systems must be set at the same position.

The following method is usually used to define two coordinate systems at the same location.

1. When coordinate zero point is set at chuck face

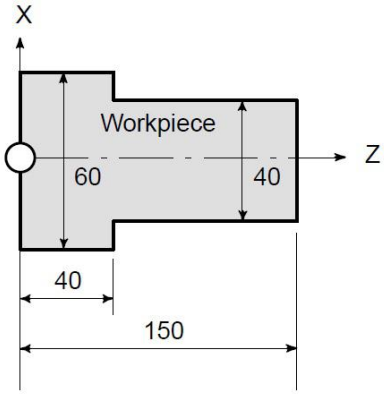


Fig2.8 Coordinates and dimensions on part drawing

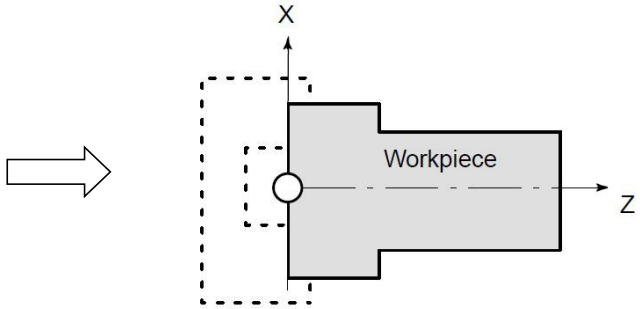


Fig2.9 Coordinate system on lathe as specified by CNC (made to coincide with the coordinate system on part drawing)

2. When coordinate zero point is set at work end face.

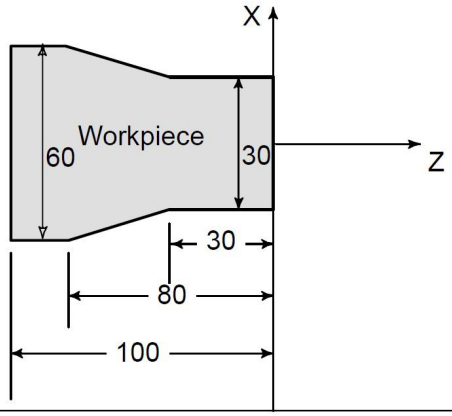


Fig2.10 Coordinates and dimensions on part drawing

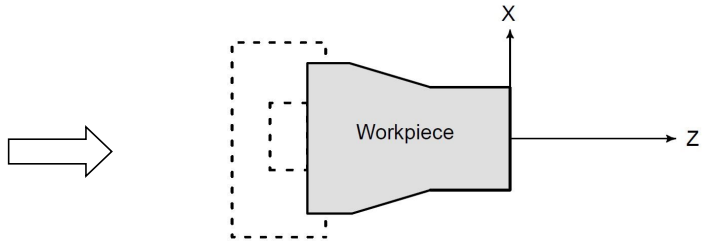


Fig2.11 Coordinates system on lathe as specified by CNC (made to coincide with the coordinate system on part drawing)

2.1.5 Interpolation

The tool moves along straight lines and arcs constituting the workpiece parts figure.

Interpolation is defined as a planar or three dimensional contour formed by path of 2 or multiple axes moving at the same time, also called **Contour control**. The controlled moving axis is called link axis when the interpolation is executed. The moving distance, direction and speed of it are controlled synchronously in the course of running to form the required complex motion path. Fixed point control is defined that the motion path in the course of running are not controlled but end point of one axis or multiple axes moving.

Linear Interpolation: Tool movement alone a straight line

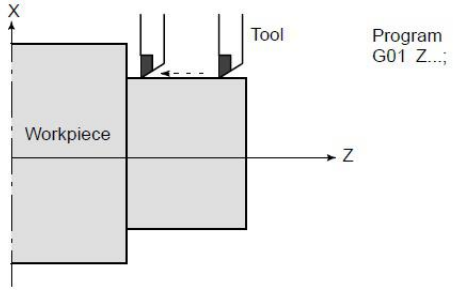


Fig2.12 Tool movement along the straight line which is parallel to Z–axis

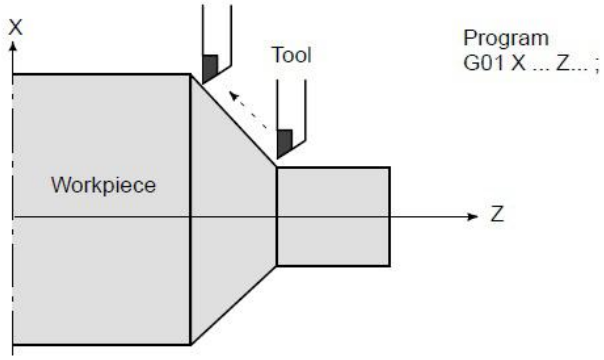


Fig2.13 Tool movement along the taper line

Arc Interpolation: Tool movement along an arc

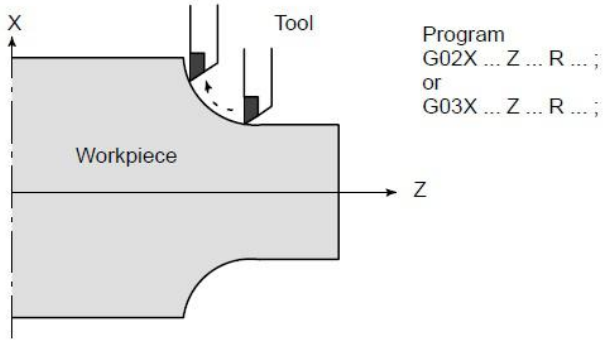


Fig2.14 Tool movement along an arc

Thread Interpolation(Thread Cutting): Threads can be cut by moving the tool in synchronization with spindle rotation. In a program, specify the thread cutting function by G32.

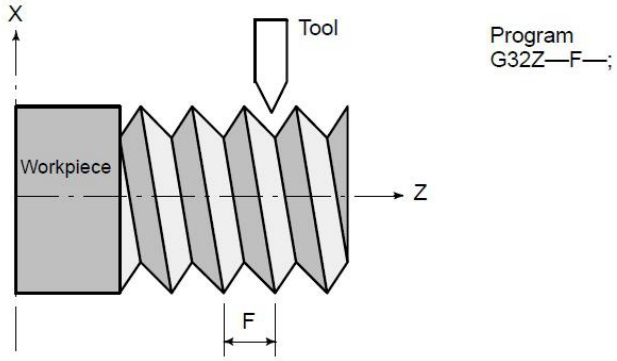


Fig2.15 Straight thread cutting

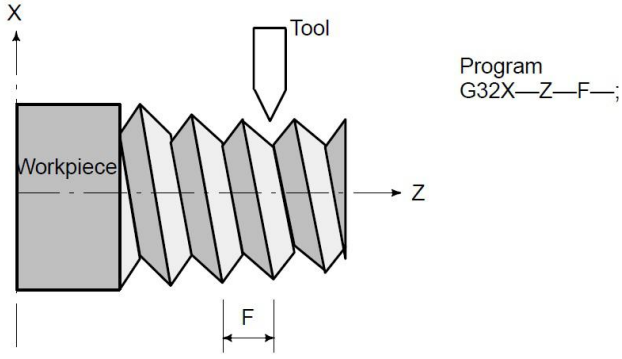


Fig2.16 Taper thread cutting

Feed: Movement of the tool at a specified speed for cutting a workpiece is called the feed.

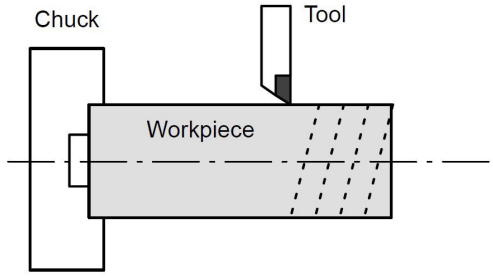


Fig2.17 Feed Function

Feed Rates can be specified by using actual numerics. Eg.: **F2.0** , which can be used to feed the tool 2 mm while the workpiece makes one turn.

The function of deciding the feed rate is called the feed function.

2.1.6 Absolute Programming & Incremental Programming

Methods of command for moving the tool can be indicated by absolute or incremental designation

Absolute command: The tool moves to a point at “the distance from zero point of the coordinate system” that is to the position of the coordinate values.

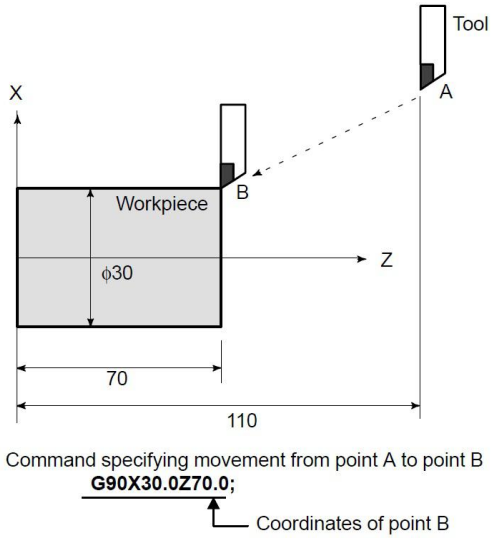


Fig2.18 Absolute Command

Incremental command: Specify the distance from previous tool position to the next tool position.

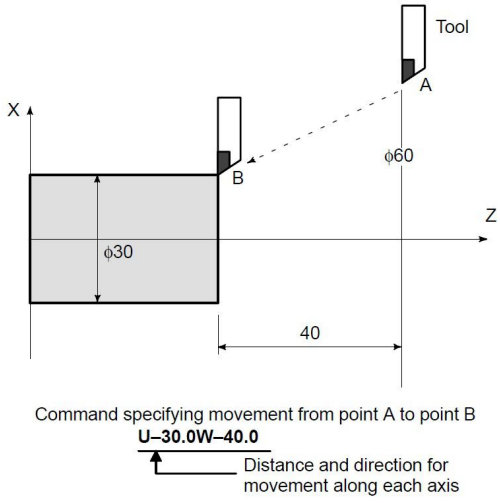


Fig2.19 Incremental command

2.1.7 Diameter Programming & Radius Programming

Since the work cross section is usually circular in CNC lathe control programming, its dimensions can be specified in two ways :

Diameter and Radius

When the diameter is specified, it is called diameter programming and when the radius is specified, it is called radius programming.

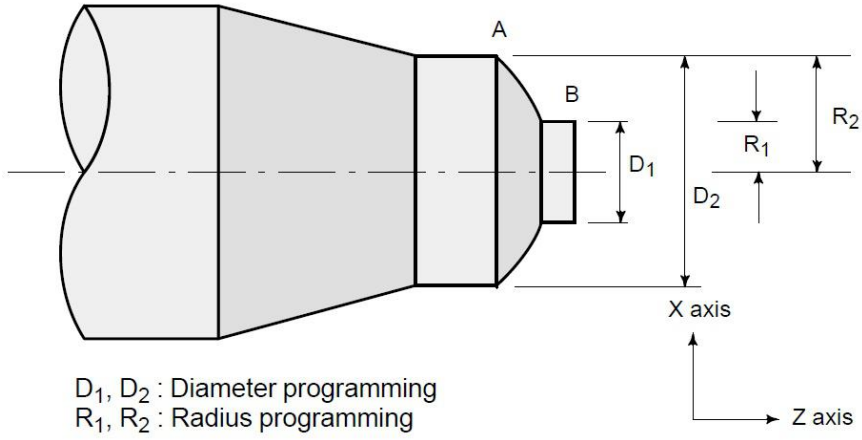


Fig 2.20 Diameter Programming & Radius Programming

Dimensions of the X axis can be set in diameter or in radius. Diameter programming or radius programming is employed independently in each machine. P16 in User parameter is set for diameter command&radius command.

1. Diameter Programming

In diameter programming, specify the diameter value indicated on the drawing as the value of the X axis.

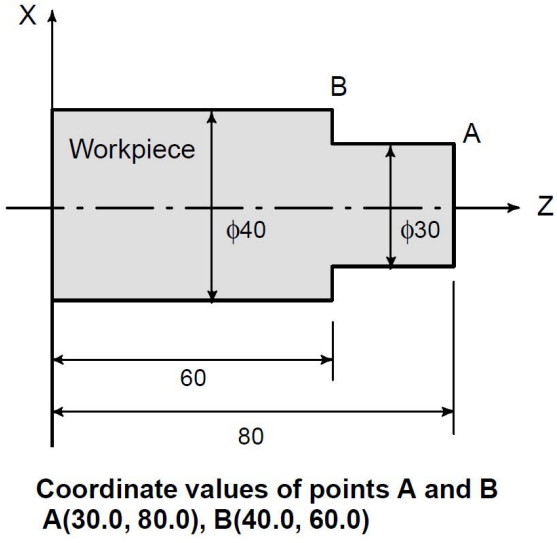
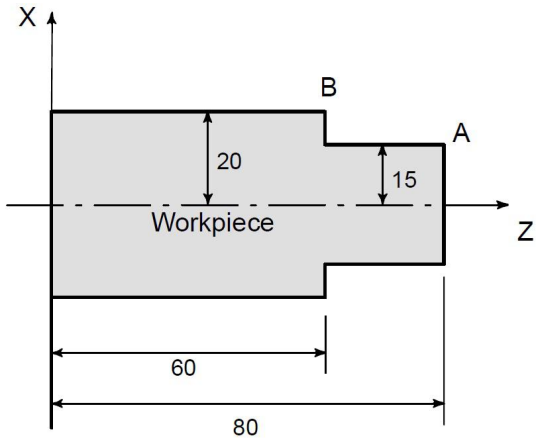


Fig2.21 Diameter Programming

2. Radius Programming

In radius programming, specify the distance from the center of the workpiece, i.e. the radius value as the value of the X axis.



Coordinate values of points A and B
A(15.0, 80.0), B(20.0, 60.0)

Fig2.22 Radius Programming

When using diameter programming, note the conditions listed in the table in the following,

Item	Notes
X axis command	Specified with a diameter value
Incremental command	Specified with a diameter value In the above figure, specifies D2 minus D1 for tool path B to A of Fig2.14.
Coordinate system setting (G50)	Specifies a coordinate value with a diameter value
Component of tool offset value	P16 in User Parameter determines either diameter or radius value
Parameters in canned cycle, such as cutting depth along X axis. (R)	Specifies a radius value
Radius designation in circular interpolation (R, I, K, and etc.)	Specifies a radius value
Feedrate along axis	Specifies change of radius/rev. or change of radius/min.
Display of axis position	Displayed as diameter value

2.1.8 Cutting Speed & Spindle Speed

The speed of the tool with respect to the workpiece when the workpiece is cut is called the cutting speed.

As for the CNC Machine, the cutting speed can be specified by the spindle speed in min^{-1} unit.

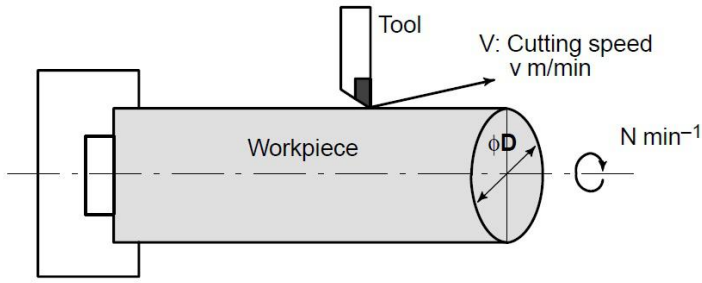


Fig2.23 Cutting Speed

Eg.: <When a workpiece 200 mm in diameter should be machined at a cutting speed of 300

m/min. >

The spindle speed is approximately 478 min − 1, which is obtained from N=1000v/ π D. Hence the following command is required:

S478 ;

Commands related to the spindle speed are called the spindle speed function.

The cutting speed v (m/min) can also be specified directly by the speed value. Even when the workpiece diameter is changed, the CNC changes the spindle speed so that the cutting speed remains constant.

This function is called the constant surface speed control function.

2.1.9 Tool Function

Selection of tool used for various machining.

When drilling, tapping, boring, milling or the like, is performed, it is necessary to select a suitable tool. When a number is assigned to each tool and the number is specified in the program, the corresponding tool is selected.

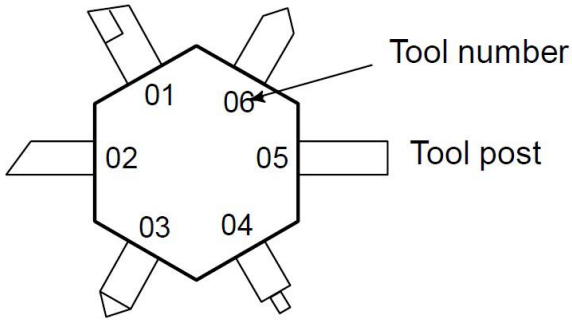


Fig2.24 Tool used for various machining

Example: <When No.01 is assigned to a roughing tool>

When the tool is stored at location 01 of the tool post, the tool can be selected by specifying **T0101**.This is called the tool function

2.1.10 Command For Machine Operations

When machining is actually started, it is necessary to rotate the spindle, and feed coolant. For this purpose, on–off operations of spindle motor and coolant valve should be controlled

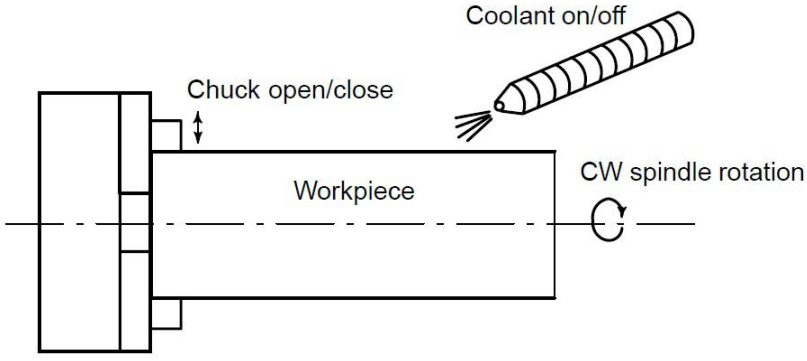


Fig2.25 Command for machine Operations

The function of specifying the on–off operations of the components of the machine is called the miscellaneous function. In general, the function is specified by an M code.

For example, when M03 is specified, the spindle is rotated clockwise at the specified spindle speed.

2.2 Configuration of Program

A group of commands given to the CNC for operating the machine is called the program. User needs to compile part programs according to instruction formats of CNC system.By specifying the commands, the tool is moved along a straight line or an arc, or the spindle motor is turned on and off.

In the program, specify the commands in the sequence of actual tool movements.

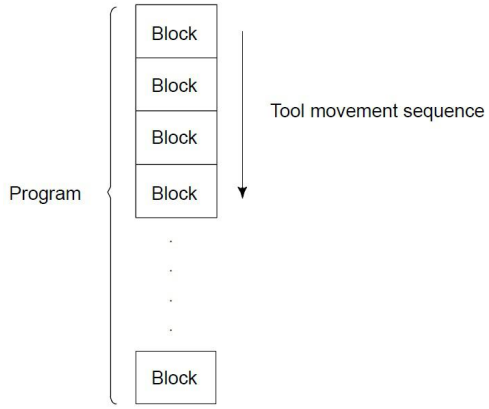


Fig2.26 Configuration of Program

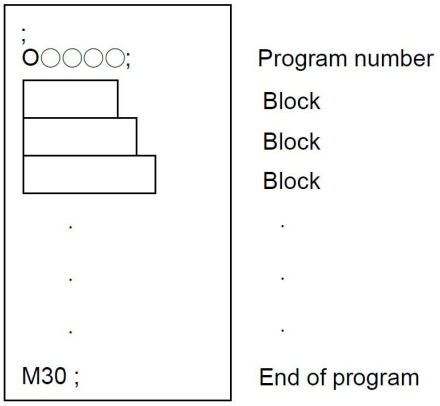


Fig2.27 Program Configuration

Normally, a program number is specified after the end - of - block (;) code at the beginning of the program, and a program end code (M02 or M30) is specified at the end of the program.

See the general structure of program as follows:

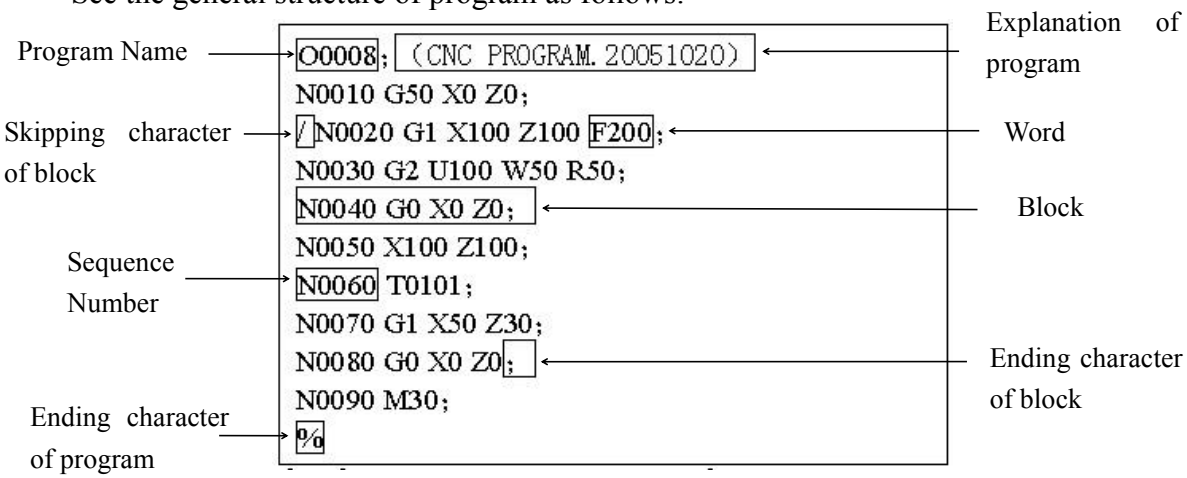


Fig2.28 General Structure of Program

Program Name: consist of alphabet & number (Eg.: O0001). There are countless programs stored in the system.To identify it, each program has only one program name(there is no the same program name).

Note: It doesn't allow exist blank on program name.

Word is the basic instruction unit to command CNC system to complete the control function,composed of an English letter (called instruction address) and the following number (operation instruction with/without sign). The instruction address describes the meaning of its following operation instruction and there may be different meaning in the same instruction address when the different words are combined together. Table 2-1 is Word List of SZGH-CNC990TDc system.

Table 2-1 Word List

Address	Data range	Functions
N	00000~9999	Block number
G	00~99	Preparatory function
	100~150	User-defined G macro function
M	00~99	Auxiliary function output
T	01-99	Tool function
S	0-99999(rpm)	Specify Speed of 1st Spindle
SS	0-99999(rpm)	Specify Speed of 2nd Spindle
F	0.01-15000mm/min	Feedrate per minute
	0.001-500mm/r	Feedrate per rev
	0.1~1000mm	Thread Lead in Metric
X	±99999.999mm	Coordinates in X directtion
	0～9999.999(s)	Dwell time
U	±99999.999mm	Increment in X directtion
	Finish allowance in X direction in G71,G72,G73	
	Cutting depth in G71	
	Moving distance of tool retraction in X direction in G73	
Z	±99999.999mm	Coordinates in Z directtion
W	±99999.999mm	Increment in Z directtion
	Finish allowance in Z direction in G71,G72,G73	
	Cutting depth in G72	
	Moving distance of tool retraction in Z direction in G73	
I	00-99 teeth/inch	Thread Lead in Inch
	±99999.999mm	Vector of arc center relative to starting point I in X direction
K	±99999.999mm	Vector of arc center relative to starting point in Z direction
R	0.001-99999.999mm	Arc radius
	Moving distance of cycle tool retraction in G71,G72	
	Cycle times of roughing in G73	
	Moving distance of tool retraction after Cutting in G74, G75	
	Moving distance of tool retraction after cutting to the end point in G74, G75	
	Finishing allowance in G76	
	Taper in G90, G92, G94, G96	
P	0.001-65s	Dwell time
	0000-99999	Calling subprogram number
	Circular moving distance in X direction in G74, G75	

	Thread cutting parameter in G76	
	Initial block number of finishing in the compound cycle instruction	
L	1~9999	Cycle times
	1~9999	Cycle times of calling subprogram
	1-99	Heads of multi-head thread
T	0000-9999	Tool function
M	00-99	Auxiliary function output, program executed flow, subprogram call
	880-889	User-defined M Macro Function
/	Program skip	

Block: a group of commands at each step of the sequence.

The program consists of a group of blocks for a series of machining. The number for discriminating each block is called the sequence number, and the number for discriminating each program is called the program number.

The block and the program have the following configurations.

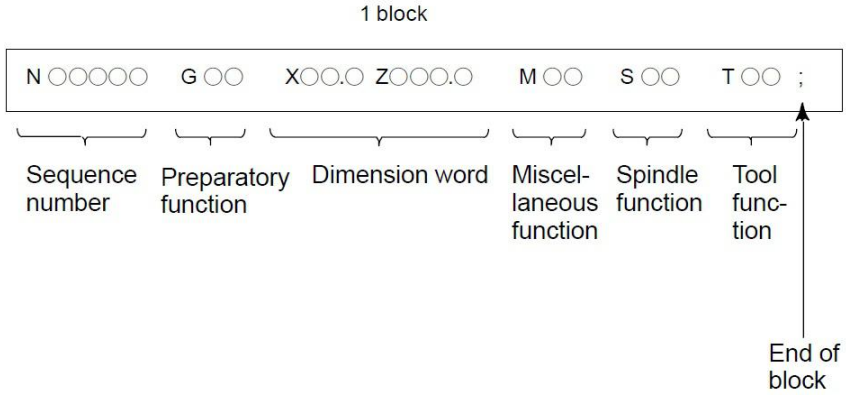


Fig2.29 Block Configuration

A block begins with a sequence number that identifies that block and ends with an end－of－block code.

This manual indicates the end－of－block code by ; (LF in the ISO code and CR in the EIA code).

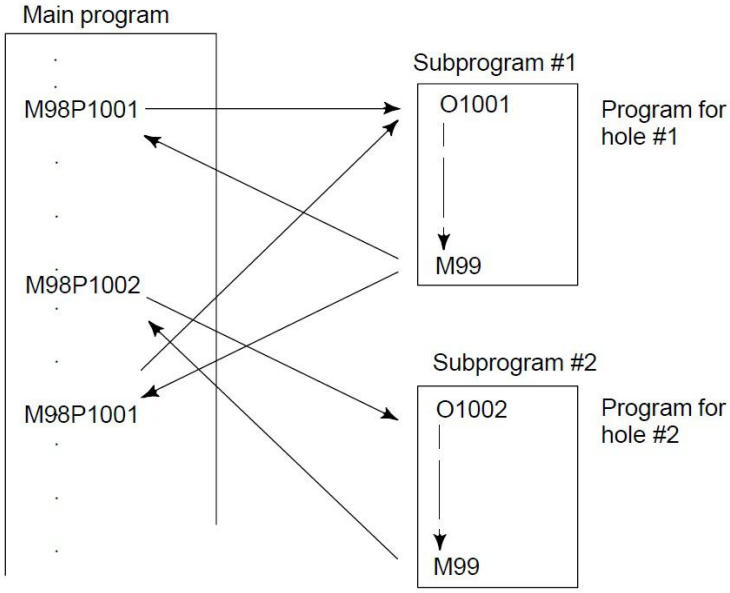
The contents of the dimension word depend on the preparatory function.

In this manual, the portion of the dimension word may be represent as IP_.

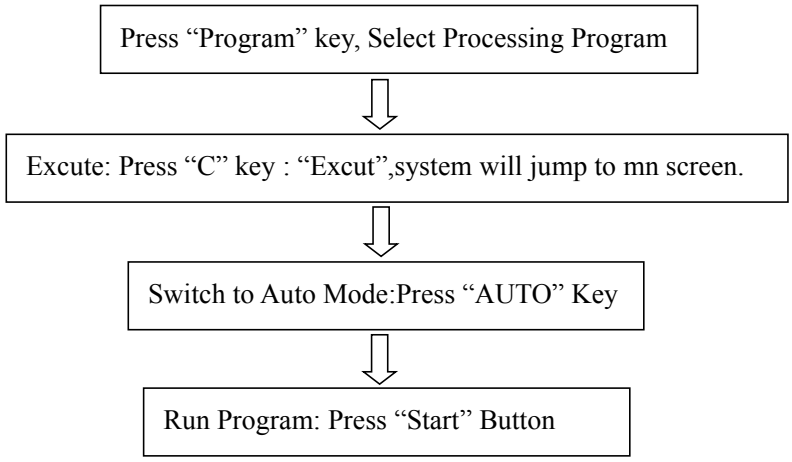
There is only one for other addresses except for N, G, S, T, H, L in one block, otherwise the system alarms. The last word in the same address is valid when there are more N, G, S, T, H, L in the same block. The last G instruction is valid when there are more G instructions which are in the same group in one block.

2.3 Main Program & Subprogram

When machining of the same pattern appears at many portions of a program, a program for the pattern is created. This is called the subprogram. On the other hand, the original program is called the main program. When a subprogram execution command appears during execution of the main program, commands of the subprogram are executed. When execution of the subprogram is finished, the sequence returns to the main program.



2.4 Program Run



Suggestion:Before running program,please compile program ,and ensure program is right.

Chapter 3 G INSTRCUTIONS

3.1 INTRODUCTION

G instruction consists of instruction address G and its following 1～2 bits instruction value, used for defining the motion mode of tool relative to the workpiece, defining the coordinates and so on. Refer to G instructions as Table 3.

G □□

→ Instruction value(00~99,the leading zero can be omitted)

→ Instruction address G

A number of following address G determines the meaning of the comand for the concerned block.

G codes are divided into the following two types

Type	Meaning
One-shot G code	The G code is effective only in the block in which it is specified
Modal G code	The G code is effective unti another G code of the same group is specified.

Eg.: G01 and G00 are modal G codes.

```
G01X_;
    Z_;
    X_;
G00Z_;
```

} **G01 is effective in this range**

3.2 G Code List

1. If CNC enters the clear state ,also when the power is turned on or CNC is reset, the modal G codes change as follows.

1) G codes marked with “ □ ” in Table 3 are enabled ,which is initial modal codes.

2) When system is cleared due to power-on or reset,which ever specified, either G20 or G21 , remains effective.

2. G codes of group 00 are single-shot G codes.

3. G codes of different groups can be specified in the same block.

If G codes of the same group are specified in the same block,the G code specified last is valid.

4. G codes of different groups can be specified in the same block.

If G codes of the same group are specified in the same block, the G code specified last is valid.

5. If a G code of group 01 is specified in a canned cycle, the canned cycle is canceled in the same way as when a G80 command is specified. G codes of group 01 are not affected by G codes for specifying a canned cycle.

6. G codes are displayed for each group number.

7. When a G code not listed in the G code list is specified or a G code that corresponding function is disabled.

Table 3 G Code List

Word	Ground	Functions	Page
G00	01	Positioning(Rapid Traverse)	
G01		Linear Interpolation(Cutting feed)	
G02		Circular Interpolation CW	
G03		Circular Interpolation CCW	
G32	01	Thread Cutting	
G332		Thread Interpolation with Circular CW	
G333		Thread Interpolation with Circular CCW	
G70	00	Finishing Cycle	
G71		Axial Roughing Turning Cycle	
G72		Radial Roughing Facing Cycle	
G73		Pattern Repeating Cycle	
G74		End Face Peck Drilling	
G75		Outer diameter/Internal diameter Grooving Cycle	
G76		Multi Threading Cycle	
G90	01	Outer diameter/internal diameter cutting cycle	
G92		Thread Cutting Cycle	
G93		Canned Tapping Cycle	
G94		Endface turning cycle	
G31	00	Skip function (No alarm)	
G311		Skip function (alarm)	
G22	08	Program Block Cycle	
G800		Program Block Cycle Cancel	
G26	03	ALL-Axis go starting point	
G261		X-Axis go starting point	
G262		Y-Axis go starting point	
G263		Z-Axis go starting point	
G264		A-Axis go starting point	
G61	04	Return G25 coordinate of G25	
G611		Return the coordinate position of X-Axis in G25	
G612		Return the coordinate position of Y(C)-Axis in G25	
G613		Return the coordinate position of Z-Axis in G25	
G614		Return the coordinate position of A-Axis in G25	
G25	00	Save value of current coordinate	
G28	00	Return to reference position	
G281		X-Axis return to reference position	
G282		Y(C)-Axis Return to reference position	
G283		Z-Axis Return to reference position	
G284		A-Axis Return to reference position	
G50	00	Coordinate system setting or max. spindle speed setting	
G52		Local coordinate system setting	
G53		Machine Coordinate System	
G54	14	Workpiece Coordinate System-1 Selection	
G55		Workpiece Coordinate System-2 Selection	
G56		Workpiece Coordinate System-3 Selection	
G57		Workpiece Coordinate System-4 Selection	
G58		Workpiece Coordinate System-5 Selection	

G59		Workpiece Coordinate System-6 Selection	
G184	00	Setup/offset coordinate of current tool	
G185		Setup/offset coordinate of all tools	
G96	02	Constant surface speed control	
G97		Constant surface speed control cancel	
G98	05	Feeding per minute	
G99		Feeding per revolution	
G15	21	Polar coordinate interpolation cancel mode	
G16		Polar coordinate interpolation mode	
G990	12	Absolute programming	
G991		Incremental programming	
G20	06	Input in Inch	
G21		Input in mm	
G65	00	Macro calling	
G66	12	Macro modal call	
G67		Macro modal call cancel	
G04	00	Dwell	
G60	04	Exact Stop & Positioning	
G64		Continous Path Processing	
G40	07	Tool nose radius compensation cancel	
G41		Tool nose radius compensation left	
G42		Tool nose radius compensation right	
G17	16	XpYp plane selection	
G18		ZpXp plane selection	
G19		YpZp plane selection	

3.3 Positioning (Rapid Traverse) (G00)

G00 command moves a tool to the position in the workpiece system specified with an absolute or an incremental command at a rapid traverse rate.

In the absolute command, coordinate value of the end point is programmed.

In the incremental command the distance the tool moves is programmed.

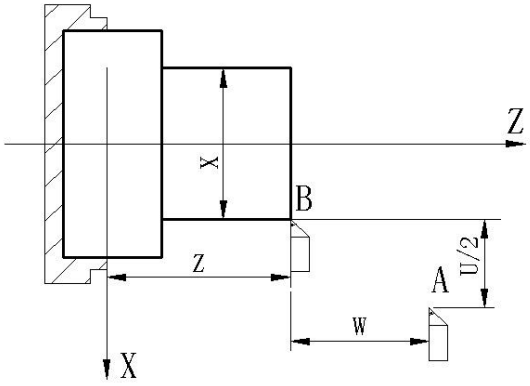


Fig3.1 Coordinate Code of G00

Format: G00 X(U)_ Z(W)_ Y/C(V)_ A_ ;

Either of the following tool paths can be selected according to P9_D6 (Bit 6 of No.9 parameter) in Other parameter.

Nonlinear interpolation positioning

The tool is positioned with the rapid traverse rate for each axis separately. The tool path is normally straight.

Linear interpolation positioning

The tool path is the same as in linear interpolation (G01). The tool is positioned within the shortest possible time at a speed that is not more than the rapid traverse rate for each axis. However, the tool path is not the same as in linear interpolation (G01).

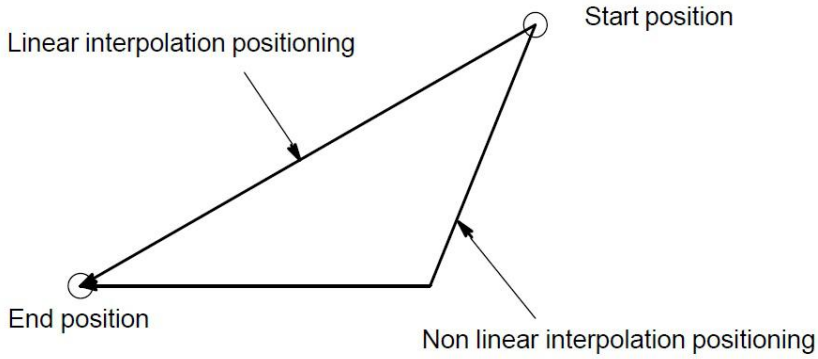


Fig3.2 Mode of Tool Path

P1 & P2 in Speed parameter is set for rapid traverse rate in the G00 command for each axis independently.

The speed rate of G00 can be divided into 5%～100%, total six gears, it can be selected by the key on panel.

G00 is mode instruction, when the next instruction is G00 too, it can be omitted.G00 can be written G0.

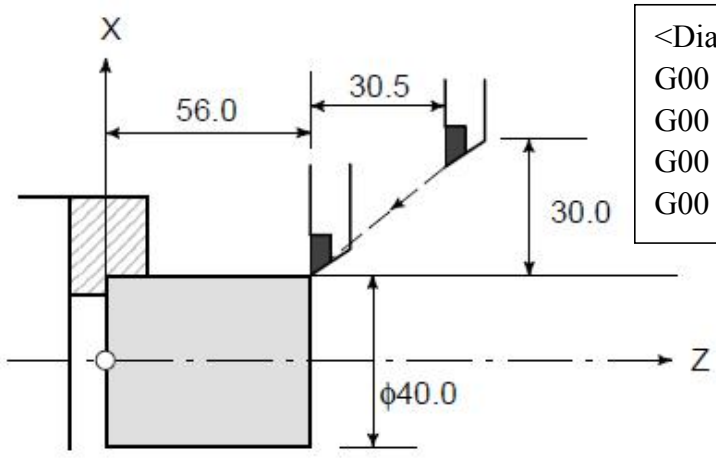
In the positioning mode actuated by G00, the tool is accelerated to a predetermined speed at the start of a block and is decelerated at the end of a block.

Note: 1. When Rotary Axis positioning in absolute programing, G00 is actuated with nearest path ; when in incremental programing, G00 is actuated with arithmetic path.

2. The rapid traverse rate cannot be specified in the address F.

3. Even if linear interpolation positioning is specified, nonlinear interpolation positioning is used in the following cases. Therefore, be careful to ensure that the tool does not foul the workpiece.

Example:



<Diameter Programming>
G00 X40.0 Z56.0 ; (Absolute command)
G00 U-60.0 W-30.5 ; (Incremental command)
G00 U-60.0 Z56.0 ; (Compound programming)
G00 X40.0 W-30.5 ; (Compound programming)

Fig3.3 Positioning of G00

3.4 Linear Interpolation (G01)

A tools move along a line to the specified position at the feedrate specified in F.

Format: G01 X/U_ Z/W_ Y(C)/V_ A_ F_ ;

X,Z,Y(C), A means motion axis.For an absolute command, the coordinates of an end point , and for an incremental command, the distance the tool moves.

F: Speed of tool feed(Feedrate)

The feedrate specified in F is effective until a new value is specified. It need not be specified for each block.

The feedrate commanded by the F code is measured along the tool path.

If the F code is not commanded, the feedrate is regarded as zero.

For feed–per–minute mode under 2–axis simultaneous control, the feedrate for a movement along each axis as follows :

G01αα̲ββ̲ ***Ff*** ;

Feed rate of α axis direction : $F\alpha = \frac{\alpha}{L} \times f$

Feed rate of β axis direction : $F_\beta = \frac{\beta}{L} \times f$

$$L = \sqrt{\alpha^2 + \beta^2}$$

Example: <Diameter Programming>

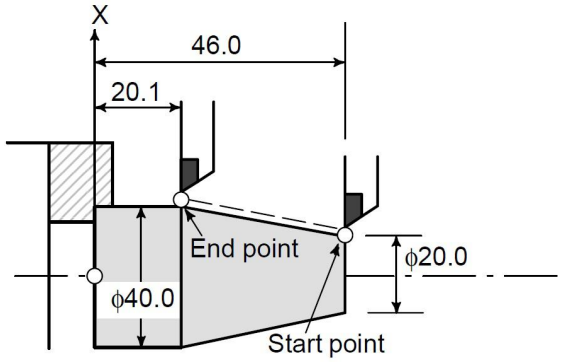


Fig3.4 Linear Interpolation of G01

G01 X40.0 Z20.1 F20 ; (Absolute command)

G01 U20.0 W-25.9 F20 ; (Incremental command)

G01 instruction can also specify movemtn of either X-axis or Z-axis separately.

G01 is F feed rate can be motivated by the panel to override adjusted up or down to adjust the range (0% -150%).

G01 instruction can also be directly written G1.

3.5 Circular Interpolation (G02/G03)

These commands will move a tool along a circular arc.

Format: Arc in the ZpXp plane (Default)

$$\mathbf{G18} \left\{ \begin{matrix} \mathbf{G02} \\ \mathbf{G03} \end{matrix} \right\} \mathbf{Xp\_Zp\_} \left\{ \begin{matrix} \mathbf{I\_K\_} \\ \mathbf{R\_} \end{matrix} \right\} \mathrm{F\_}$$

Code	Description
G17	Specification of arc on XpYp plane
G18	Specification of arc on ZpXp plane(Default)
G19	Specification of arc on YpZp plane
G02	Circular Interpolation Clockwise direction (CW)
G03	Circular Interpolation Counterclockwise direction (CCW)
X/Z/Y	Position of end point in workpiece coordinate
U/W/V	Distance from start point to end point
I	X axis distance from start point to center of an arc with sign(radius value)
K	Z axis distance from start point to center of an arc with sign(radius value)
J	Y axis distance from start point to center of an arc with sign(radius value)
R	Arc radius without sign (always with radius value)
F	Feedrate along the arc

Note: G18 is default set , which can be omitted.

Direction of circular interpolation: When turret in different position of lathe machine, Front toolpost system & Rear toolpost system , the direction of G02&G03 is opposite in these two system.

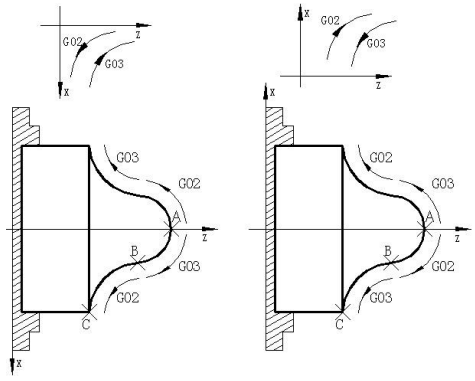


Fig3.5.1 Direction of G02&G03 at different toolpost system

"Clockwise"(G02) and "counterclockwise"(G03) on the ZpXp plane are defined when the XpYp plane is viewed in the positive - to - negative direction of the Yp axis in the Cartesian coordinate system.

Distance moved on an arc: The end point of an arc is specified by address X(U) , Z(W) or Y(V), and is expressed as an absolute or incremental value according to G990 or G991. For the incremental value, the distance of the end point which is viewed from the start point of the arc is specified.

Distance from the start point to the center of arc: The arc center is specified by addresses I and K for the Xp and Zp axes, respectively. The numerical value following I or K, however, is a vector component in which the arc center is seen from the start point, and is always specified as an incremental value irrespective of G990 and G991, as shown below.

I, and K must be signed according to the direction.

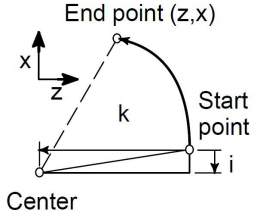


Fig3.5.2 Define & Direction of I & K

I0 and K0 can be omitted.

If the distance is from the end point to the center of arc , which exceeds by the value in a parameter of P41 in Speed parameter(Original value+4).

Full－circle programming: When Xp and Zp are omitted (the end point is the same as the start point) and the center is specified with I and K, a 360° arc (circle) is specified.

Arc radius: The distance between an arc and the center of a circle that contains the arc can be specified using the radius, R, of the circle instead of I and K. In this case, one arc is less than 180°, and the other is more than 180° are considered. If Xp and Zp are all omitted, if the end point is located at the same position as the start point and when R is used, an arc of 0° is programmed.

G02R ; (The cutter does not move.)

Note: 1.When I = 0 or K = 0, they can be omitted; one of I, K or R must be input, otherwise the system alarms.

2.If I, K, and R addresses are specified simultaneously, the arc specified by address R takes precedence and the other are ignored.

3.Processing arc workpiece usually use ball tool(arc tool) in the actual process, it must use function of tool radius compensation in programming, that's G41 G42 instruction.

4.Arc path can be more than and less than 180° when R is commanded, the arc is more than180° when R is negative, and it is less than or equal to 180 ° when R is positive.

Example 1 : Command of circular interpolation X,Z

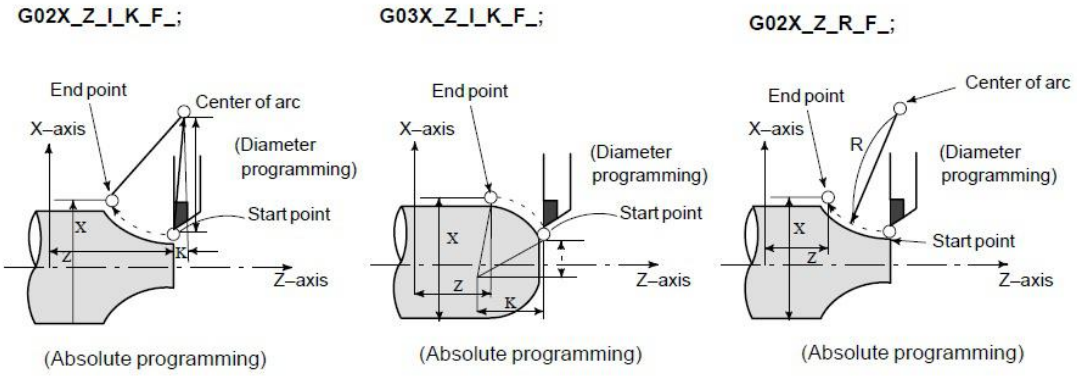
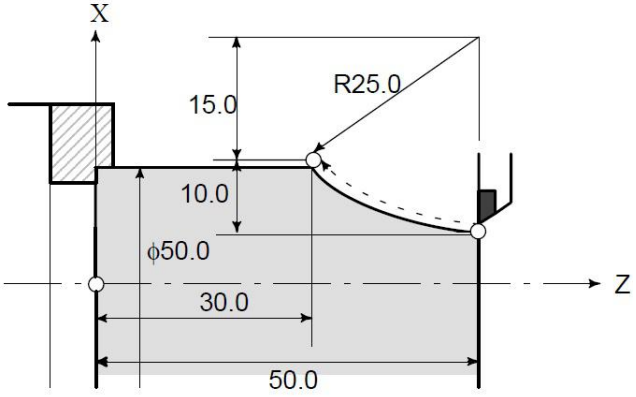


Fig3.5.3 Command of circular interpolation

Example 2: Processing same path with different command.

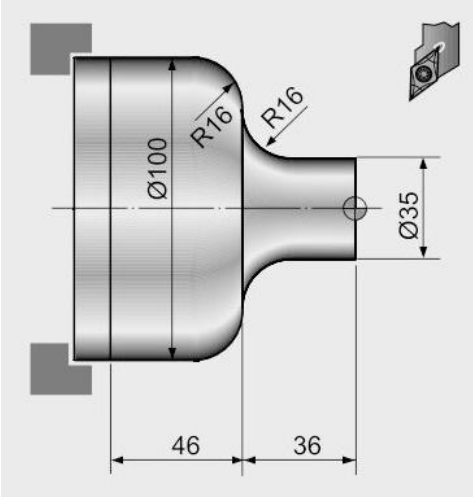


<Diameter Programming>

- G02 X50.0 Z30.0 I25.0 F0.3
- G02 U20.0 W-20.0 I25.0 F0.3
- G02 X50.0 Z30.0 R25.0 F0.3
- G02 U20.0 W-20.0 R25.0 F0.3

Fig3.5.4 Processing same path with different command

Example 3: Application of Circular Interpolation




```
<Program>
N20 G50 S2000 T0300
G96 S200 M03
G42 G00 X35.0 Z5.0 T0303 M08
G01 Z-20.0 F0.2
G02 X67.0 Z-36.0 R16.0
G01 X68.0
G03 X100.0 Z-52.0 R16.0
G01 Z-82.0
G40 G00 X200.0 Z200.0 M09 T03.00
M30
```

Fig3.5.5 Example 3

SZGH-CNC990TD CNC system support many kinds of thread cutting, which includes straight thread,tapered thread,scroll thread,thread cutting with variable lead, Continous thread cutting, multiple-thread cutting, metric/inch single.Length and angle of thread run-out can be changed,multiple-thread is machined by single side to protect tool and improve smooth finish of its surface.

When machine tool needs to do thread cutting, spindle must fix encoder.When transmission of spindle and encoder is not 1:1, we need to set transmission ratio by set P412&P413 in Axis parameter.

P10 in Axia parameter is set for pulses of per rev of spindle(Resolution*Poles).

P412 in Axis parameter is set for teeth of spindle motor.

P413 in Axis parameter is set for teeth of spindle encoder.

X or Z axis traverses to start machine after the system receives spindle signal per rev in thread cutting, and so one thread is machined by multiple roughing, finishing without changing spindle speed.

There is a big error in the thread pitch because there are the acceleration and the deceleration at the starting and ending of thread cutting in X, Z direction, and so there is length of thread lead-in and distance of tool retraction at the actual starting and ending of thread cutting.

The traverse speed of tool in X, Z direction is defined by spindle speed instead of cutting feedrate override in thread cutting when the pitch is defined. The spindle override control is valid in thread cutting. When the spindle speed is changed, there is error in pitch caused by acceleration/deceleration in X, Z direction, and so the spindle speed cannot be changed.

Note:

1. When teeth of spindle motor is more than teeth of spindle encoder , it must match with keysets of our company ;

2. In the process of thread cutting, Feeding speed and override is invalid.

3. In the process of thread cutting, spindle will not stop whatever you operate, if the user operate suspend,the system will suspend after processd this segment.

4. The spindle speed must remain constant from rough cutting through finish cutting. If not, incorrect thread lead will occur.

3.6 Thread Cutting (G32)

Tapered screws and scroll threads in addition to equal lead straight threads can be cut by using a G32 command.

The spindle speed is read from the position coder on the spindle in real time and converted to a cutting feedrate for feed – per minute mode, which is used to move the tool.

3.6.1 Constant Lead Threading

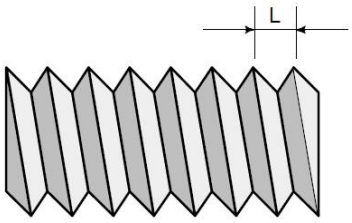


Fig3.6.1 Straight thread

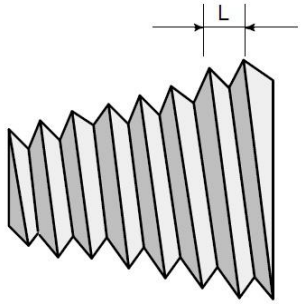


Fig3.6.2 Tapered Screw

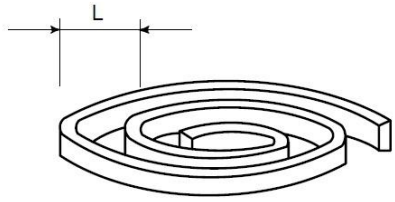


Fig3.6.3 Scroll Thread

Straight thread: only input the direction and length of Z-axis;

Tapper thread: must input the direction and length of X-axis and Z-axis;

Scroll thread(head face thread): only input the direction and length of X-axis;

Format: G32 Z(W)_X(U)_ F(I)_ SP(Q)_

G32 is the spiral interpolation machining instruction.it is modal.

X(U)_ , Z(W)_ is end point in absolutely/correspond coordinate system.

F_: metric lead(pitch) of long axis, is moving distance of long axis when the spindle rotates one rev: 0.1-1000mm; Max Lead=Lines (Resolution) of Spindle encoder /50 mm. After F is executed, it is valid until F with specified pitch is executed again.

I_: teeth per inch,is ones per inch(25.4 mm) in long axis, and also is circles of spindle rotation when the long axis traverses one inch(25.4 mm):0.1~99 tooth/inch. After I is executed, it is valid until I with specified pitch is executed again.

SP_: Initial angle(offset angle)between spindle rotation one rev and starting point of thread cutting: 0~360000(unit: 0.001 degree). SP is non-modal parameter, must be defined every time, otherwise it is 0°.

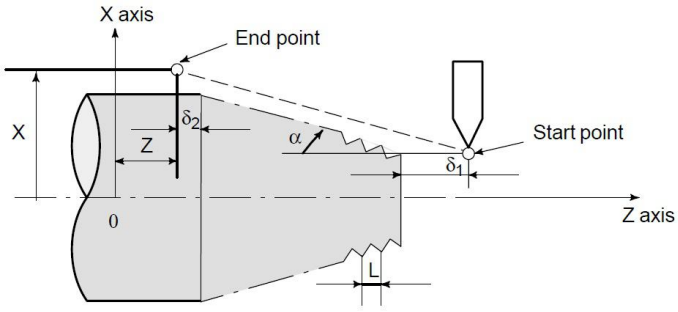
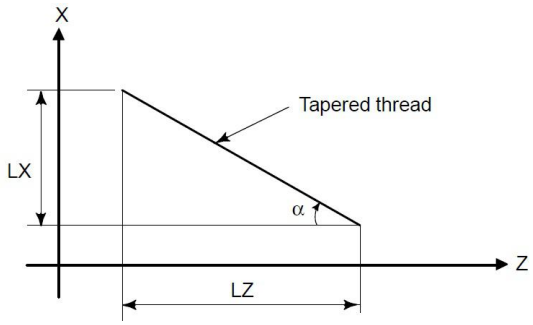


Fig3.6.4 Example of Thread Cutting

In general, thread cutting is repeated along the same tool path in rough cutting through finish cutting for a screw. Since thread cutting starts when the position coder mounted on the spindle outputs a 1–turn signal, threading is started at a fixed point and the tool path on the workpiece is unchanged for repeated thread cutting.



α ≥ 45° , lead is LZ , Z-axis is long axis
α ≤ 45° , lead is LX , X-axis is long axis

Fig3.6.5 LZ & LX of a tapered thread

In general, the lag of the servo system, etc. will produce somewhat incorrect leads at the starting and ending points of a thread cut. To compensate for this, a threading length somewhat longer than required should be specified.

Example1: Straight Thread Cutting

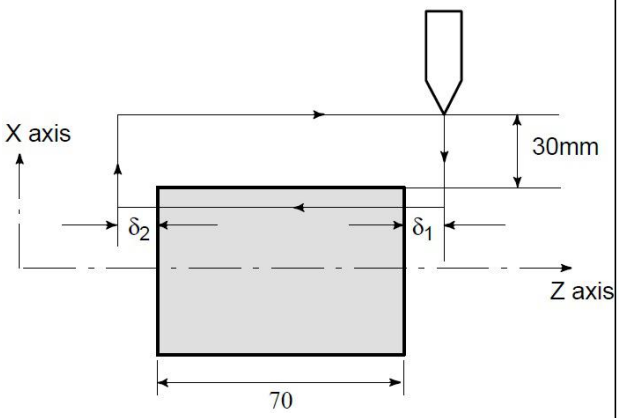


Fig3.6.6 Example of Straight Thread Cutting

Requirement: (Metric input, Diameter programming)
Thread Lead:4mm , δ_1=3mm ;δ_2=3mm ; Depth of cut:1mm(cut twice)

```
G00 U-62.0 ;
G32 W-74.5 F4.0 ;
G00 U62.0 ;
W74.5 ;
U-64.0 ;
(For the second cut,cut 1mm)
G32 W-74.5 ;
G00 U64.0 ;
W74.5 ;
```

Example2: Tapered Thread Cutting

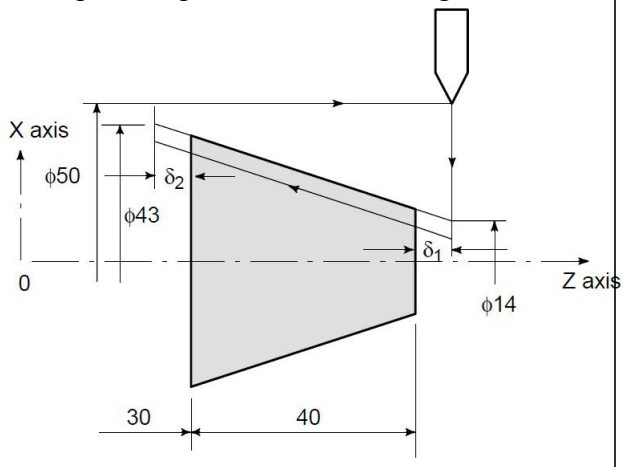


Fig3.6.7 Example of Tapered Thread Cutting

Requirement: (Metric input, Diameter programming)
Thread Lead:3.5mm in the direction of Zaxis,
δ_1=2mm ;δ_2=1mm; Depth Cut(X):1mm (cut twice)

```
G00 X12.0 Z72.0 ;
G32 X-41.0 Z29.0 F3.5 ;
G00 X50.0 ;
Z72.0 ;
X10.0 ;
(For the second cut,cut 1mm)
G32 X39.0 Z29.0 ;
G00 X50.0 ;
Z72.0 ;
```

3.6.2 Continous Thread Cutting

This function for continuous thread cutting is such that fractional pulses output to a joint between move blocks are overlapped with the next move for pulse processing and output (block overlap) .

Therefore, discontinuous machining sections caused by the interruption of move during continuously block machining are eliminated, thus making it possible to continuously direct the block for thread cutting instructions.

Since the system is controlled in such a manner that the synchronism with the spindle does not deviate in the joint between blocks wherever possible, it is possible to performed special thread cutting operation in which the lead and shape change midway.

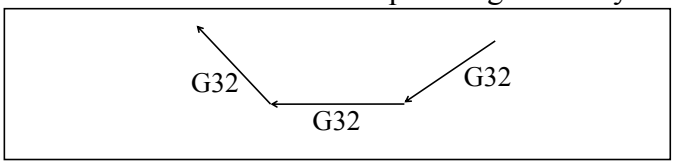


Fig3.6.8 Continuous Thread Cutting

Even when the same section is repeated for thread cutting while changing the depth of cut, this system allows a correct machining without impairing the threads.

3.6.3 Thread Cutting With Variable Lead

Variable-lead thread cutting is finsihed by continuous input G32 command, input a thread length of each program, lead of thread(F) is different. At the second cycle , CNC sytem will not detect the encoder synchronization signal.

The start angle (Q) increment is 0.001 degrees. Note that no decimal point can be specified. Example: For a shift angle of 180 degrees, specify SP180000.

SP180.000 cannot be specified, because it contains a decimal point.

3.7 Circular Thread Cutting(G332/G333)

Format: G332 Z(W)_ X(U)_ R_ F(I)_ SP_;

G332: Clockwise circular thread cutting

Format: G333 Z(W)_ X(U)_ R_ F(I)_ SP_

G333: Counter-Clockwise circular thread cutting

Z(W)_X(U)_: end point of thread cutting

R: radius of circular(negative number means degree over 180º)

F(I)_: lead(pitch) od thread ;

SP_ : start angleThe start angle is not a continuous–state (modal) value. It must bespecified each time it is used. If a value is not specified, 0 is assumed.

Using method refer to G02, G03, G32 instructions.

3.8 Canned Cycle(G90,G92,G93,G94)

To simplify programming, the system defines G instruction of single machining cycle with one block to complete the rapid traverse to position, linear/thread cutting and rapid traverse to return to the starting point.

There are three canned cycles : Outer diameter/internal diameter cutting canned cycle (G90), Thread cutting canned cycle (G92), Canned tapping cycle (G93) , and End face turning canned cycle (G94).

Note: 1. Explanatory diagrams in this chapter uses diameter programming in X axis.

2. In radius programming, changes U/2 with U and X/2 with X.

3.8.1 Outer Diameter/Internal Diameter Cutting Cycle (G90)

From starting point, the cutting cycle of cylindrical surface or taper surface is completed by radial feeding(X axis) and axial(Z axis or X and Z) cutting.

a) Straight Cutting Cycle : Format: G90 X(U)_ Z(W)_ F_ ;

X: absolute coordinates of cutting end point in X direction

U: different value of absolute coordinates between end point and starting point of cutting in X direction

Z: different value of absolute coordinates between end point and starting point of cutting in Z direction

W: different value of absolute coordinates between end point and starting point of cutting in Z direction

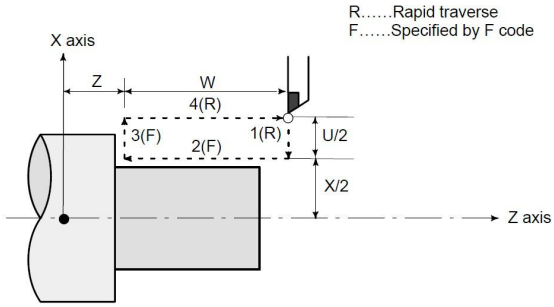


Fig3.8.1 Straight Cutting Cycle

In incremental programming, the sign of the numbers following address U and W depends on the direction of paths 1 and 2. In the Fig3.8.1, the signs of U and W are negative.

In single block mode, operations 1, 2, 3 and 4 are performed by pressing the cycle start button once.

b) Taper Cutting Cycle: Format: G90 X(U)_ Z(W)_ R_ F_ ;

R: different value (radius value) of absolute coordinates between end point and start point of cutting in X direction.

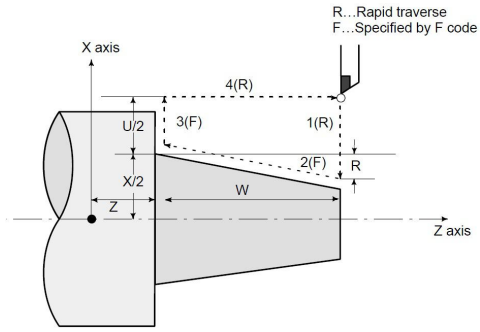
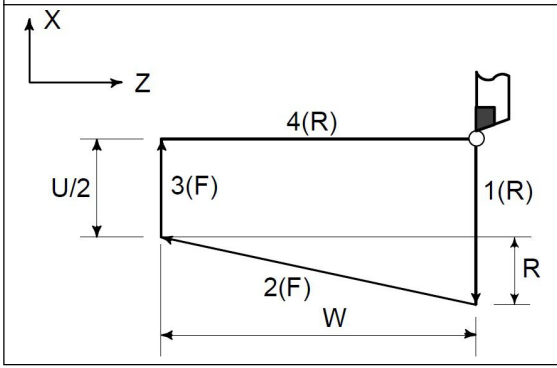
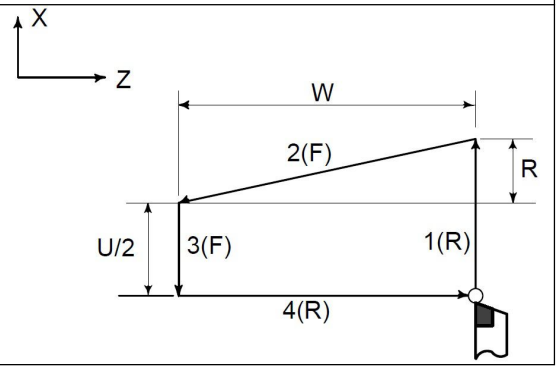
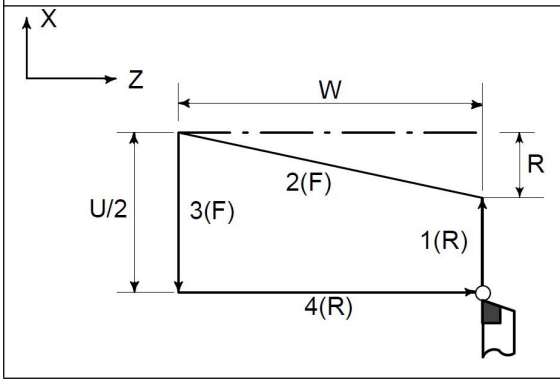
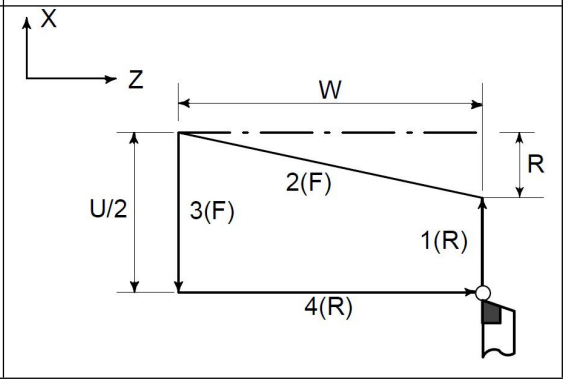


Fig3.8.2 Taper Cutting Cycle

c) Signs of numbers specified in the taper cutting cycle

In incremental programming, the relationship between the signs of the numbers following address U, W, and R, and the tool paths are as follows:

1. U<0 , W<0 , R<0	2. U>0,W<0, R>0
	

3. U<0, W<0, R>0 at |R| ≤ |U/2|	4. U>0, W<0, R<0 at |R| ≤ |U/2|
	

d) Cycle Process & examples

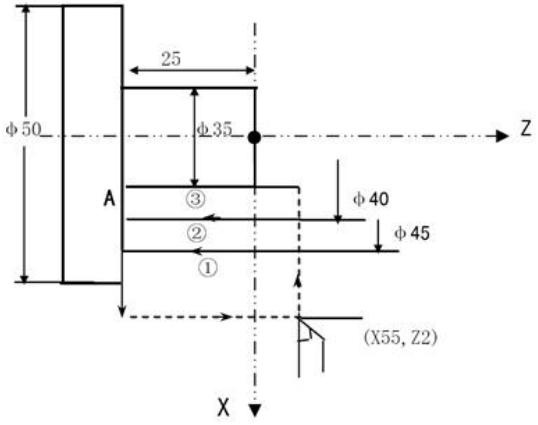
① The tool rapidly traverses from starting point to cutting starting point in X direction;

② Cutting feed(linear interpolation) from the cutting starting point to cutting end point;

③ Retract the tool at feedrate in X direction (opposite direction to the above-mentioned ①), and return to the position which the absolute coordinates and the starting point are the same;

④ The tool rapidly traverses to return to the starting point and the cycle is completed.

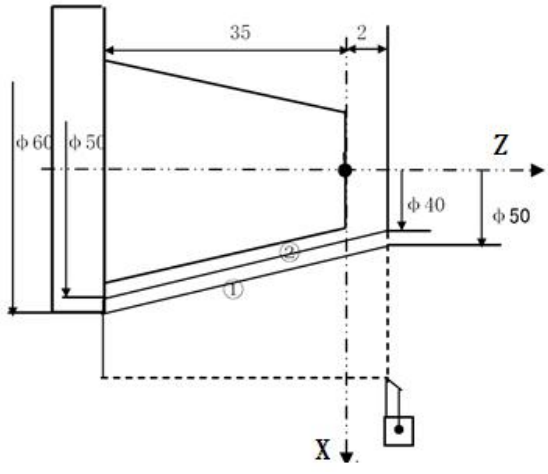
Example1: Use G90 to Process Cylinder Surface




```
N10 T0101 ;
N20 G00 X55 Z4 M03 ;
N30 G01 Z2 F100 M08 ;
N40 G90 X45 Z-25 ;
N50 X40 ;
N60 X35 ;
N70 G00 X100 Z100 ;
N80 T0100 M09 ;
N90 M05 ;
N100 M30 ;
```

Fig3.8.3 Usage of G90

Example: Use G90 to Process Taper Surface




```
N10 M03 S1000 ;
N20 T0101 ;
N30 G00 X65 Z5 ;
N50 G96 S120 ;
N60 G99 G01 Z2 F1 M08 ;
N70 G90 X60 Z-35 R-5 F0.2 ;
N80 X50 ;
N90 G00 G98 X100 Z100 M09 ;
N100 G97 S1000 T0100 ;
N110 M05 ;
N120 M30 ;
```

Fig3.8.4 Usage of G90

3.8.2 Thread Cutting Cycle (G92)

Tool infeeds in radial(X axis) direction and cuts in axial(Z axis or X, Z axis) direction from starting point of cutting to realize straight thread, taper thread cutting cycle with constant thread pitch. Thread run-out in G92: at the fixed distance from end point of thread cutting, the tool executes thread interpolation in Z direction and retracts with exponential or linear acceleration in X direction, and retracts at rapidly traverse speed in X direction after it reaches to end point of cutting in Z direction.

(1) Straight Thread Cutting: G92 X(U)_ Z(W)_ F/I_ ;

X: absolute coordinate of end point of cutting in X direction, unit:mm;

U: different value of absolute coordinate from end point to starting point of cutting in X direction, unit:mm;

Z: absolute coordinate of end point of cutting in Z direction, unit:mm;

W: different value of absolute coordinate from end point to starting point of cutting in X direction, unit:mm;

F=0.001~1000 mm, metric thread pitch. After F value is executed, it is reserved and can be omitted; I=0.1~99 toots/inch, metric thread teeth per inch.

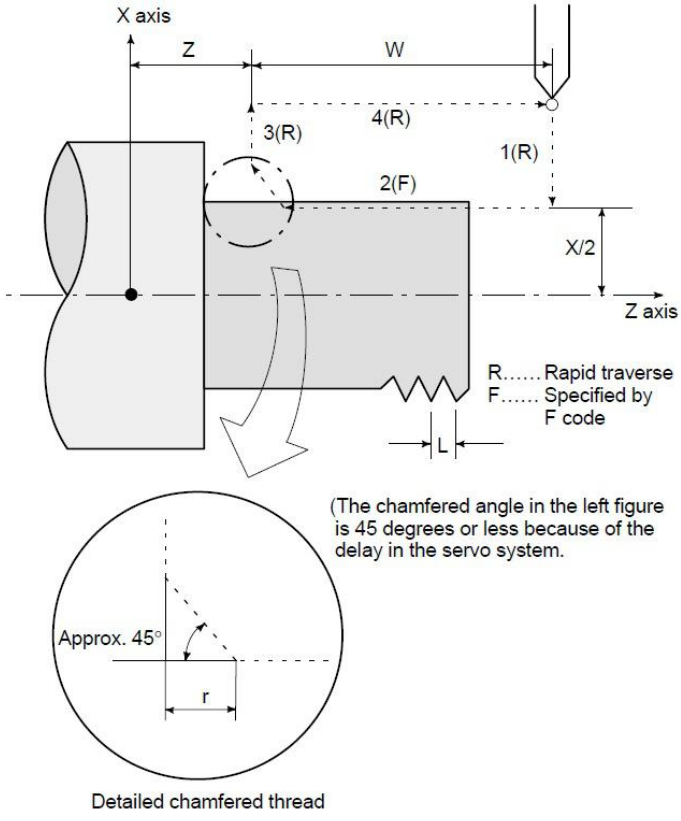
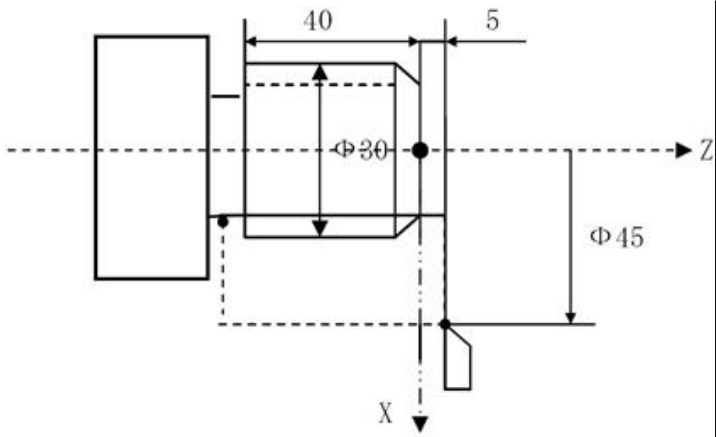


Fig3.8.5 Straight Thread Cutting

In incremental programming, the sign of numbers following addresses U and W depends on the direction of paths 1 and 2. That is , if the direction of path 1 is the negative along the X axis, the value of U is negative. The range of thread leads, limitation of spindle speed, etc. are the same as in G32 (thread cutting).



Program:
N10 M03 S1000;
N20 T0101;
N30 G00 X45 Z5;
N40 G92 X29.2 Z-45 F1.5;
N50 X28.6;
N60 X28.2;
N70 X28.04;
N80 G00 X100 Z50;
N90 T0100 M05;
N100 M30;

Fig3.8.6 Example of Straight Thread Cutting

(2)Taper Thread Cutting : G92 X(U)_ Z(W)_ R_ F/I_ ;

R: different value(R value) of absolute coordinate from end point to starting point of cutting in X direction. unit:mm.

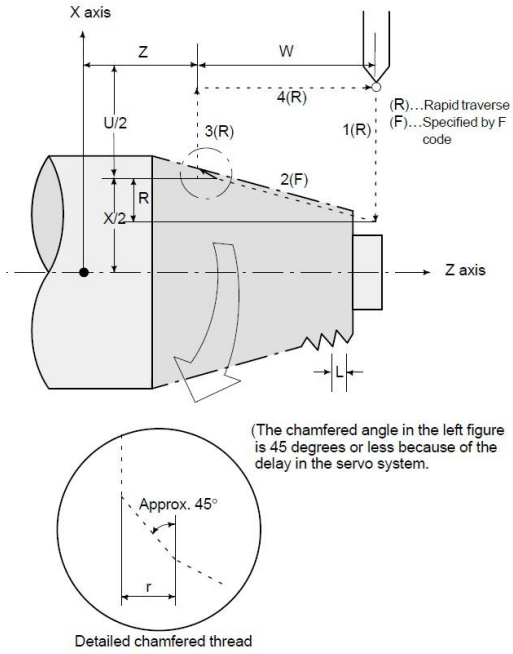


Fig3.8.7 Taper Thread Cutting

Exmaple: Process of taper screw of inner hole,
pitch is 11 tooth/inch,(coning is 1:32)

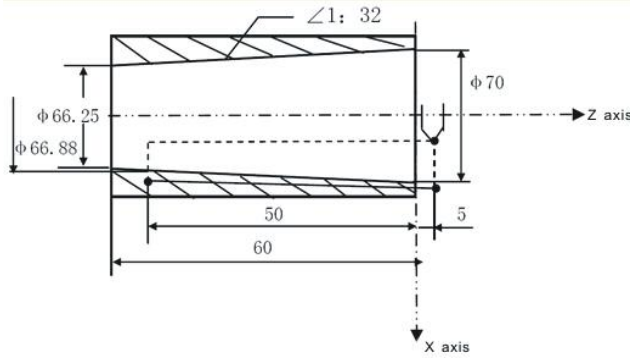


Fig3.8.8 Example of Taper Thread Cutting

Program:
N10 M03 S1000;
N20 T0101;
N30 G00 X55 Z10;
N40 G01 X60 Z5 F100;
N50 G90 X66.25 Z-60 R1.875;
N60 G92 X66.88 Z-50 R1.4 I11;
N70 X66.9 I11;
N80 X67 I11;
N90 X67.4 I11;
N100 X67.6 I11;
N110 X67.8 I11;
N120 G00 X100 Z50;
N130 T0100 M05;
N140 M30;

Note:

1.When processing inch thread, pitch I is non-mode,just be effective in one sentence, so every segments should plus I in thread cycle.

2.Speed of processing thread ,which should be less than 3000mm/min, is Pitch (F) multi Speed of spindle (S).

3.The retract speed of X axis ,which should be less than 5000mm/min, is F*S*P24 (Speed parameter)*0.1 . Eg.:When processing F2, S1200,this value of P24 parameter should be less than 20.

(3) Deceleration or acceleration control in thread cutting cycle:

At the end of thread, because of the index of deceleration control,cause the distance of pitch is inhomogeneous,the higher speed of spindle the longer of inhomogeneous pitch. To reduce the error,should reduce the index of deceleration or acceleration time, but it will cause the motor stuck if match the step motor. In order to solve this problem:

- could choose Z axis according to linear acceleration or deceleration speed constant;
- could choose the X axis with the rapid speed G00 to back tail.

The relevant parameter is as follows (see the chapter of parameter):

In Speed parameter

P22: the acceleration or deceleration constant of Z axis in thread processing

P23: the acceleration or deceleration constant of Y axis in thread processing

P24: The backing tail speed rate of servo motor in thread cycle

P25: The starting speed of servo motor in thread cycle

P26: The maximum backing tail speed of servo motor in thread cycle

(4) Multi Thread Cutting : G92 X_ Z_ F_ L_ [or SP];

L_: Multi threads: 1~100 and it is modal parameter. (the system defaults it is single thread when L is omitted); repeat times of G92 is L.

SP_: Initial angle(offset angle)between spindle rotation one rev and starting point of thread cutting: 0~360000(unit: 0.001 degree). SP is non-modal parameter, must be defined every time, otherwise it is 0°.

Note: Cannot use SP to specify when processing multi threads.

Such as: L03, 3 threads, continuous executing 3 times G92. First time,processing at once when spindle rotate one rev; Second time,after offset of 120 degrees ,begin cutting thread; Third time, after 240º offset, begin cutting thread,

Example:

G92 X50.Z-100 F5 L5 ; at X50, process 5 threads.

X48.5 ; at X48.5, processing 5 threads.

X45 ; at X45, processing 5 threads.

G00 X100 Z100 ;

.....

(5) Back Tail of Thread : G92 X_ Z_ F/I_ P_;

P: volume of backing tail: the default value of P could be set by P20 in User parameter (Default when powering on).

Set unit: P1 means 0.1 pitch ; P10 means 1 pitch .

Scope: 1--225, when the set value beyond to the range is invalid.

(6) Back tail at any angle

When cutting thread without backing fuller, the system must have the function of automatical

backing tail in thread processing to produce a qualified section of thread. Including the program format of backing tail in thread:

G92 X_ Z_ F_ J_ K_P_;

● J, K set the ratio of back tail X, Z. When J2 K1, X is twice faster than Z.

● P: back tail volume. Setting: 0.1 pitch. Set range: 1~255 (beyond to this range is invalid). The default value can be set by No.20 parameter in process parameter (Default when powering on).

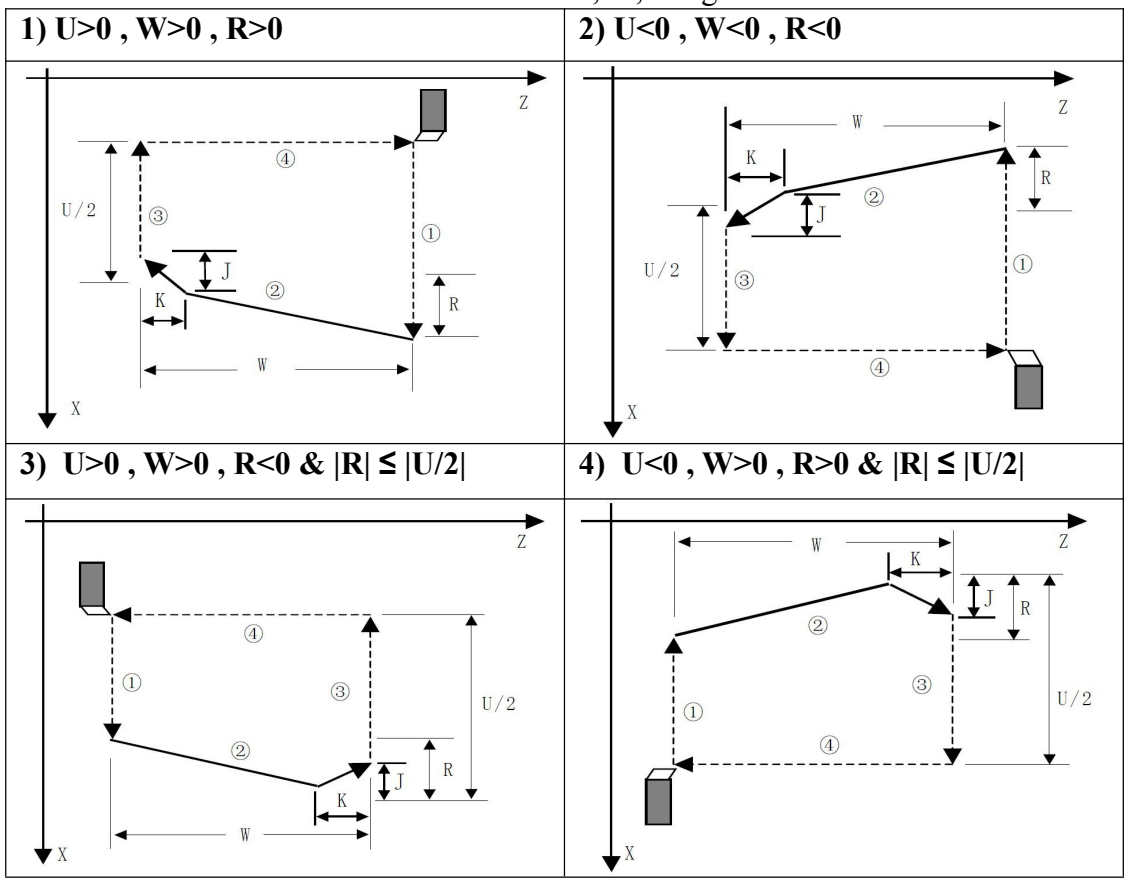
● J, K, P are mode value.

● When executing J0 or K0 in G92, cancel any angle specify, fixed 45 degrees.The default value is 45 degrees when powering on.

● When J K are set to be negative number, or beyond to 65535,it's invalid setting. The range: 1~65535.

(7) Path of G92 Instruction

Relative position between thread cutting end point and starting point with U, W, R and tool path and thread run-out direction with different U, W, R signs.



3.8.3 Canned Tapping Cycle (G93)

Tool path is from starting point to end point and then from end point to starting point. The tool traverses one pitch when the spindle rotates one rev, the pitch is consistent with pitch of tool and there is spiral grooving in internal hole of workpiece and the internal machining can be completed one time.

Canned Tapping Cycle in Z direction Format: G93 Z(W)_ F/I_ ;

Z(W): starting point and end point in Z direction are the same one not to execute the thread cutting when Z or W is not input;

G93 is modal instruction.

F: metric thread pitch

I: teeth per inch thread

Tapping has two kinds of method:

① Tracking the spindle encoder: In Axis parameter: P411=0

Note: spindle must fix with spindle encoder(CN9 plug is connected to sp-encoder).

② Interpolation between spindle servo & Z axis: In Axis Parameter: P405=0, P410=92, P411=4.

Cycle process:

1) Tool infeed in Z negative direction ;
2) Stop spindle (output M05 signal) after the tool reaches the specified end point in Z direction in progamming ;
3) Test spindle after completely stopping ;
4) Spindle rotation with reverse direction automatically ;
5) The tool retracts to starting point in Z direction ;
6) Stop spindle (output M05 signal) ;
7) Spindle recover rotation as before G93.

Example: Tapping M10 * 1.5

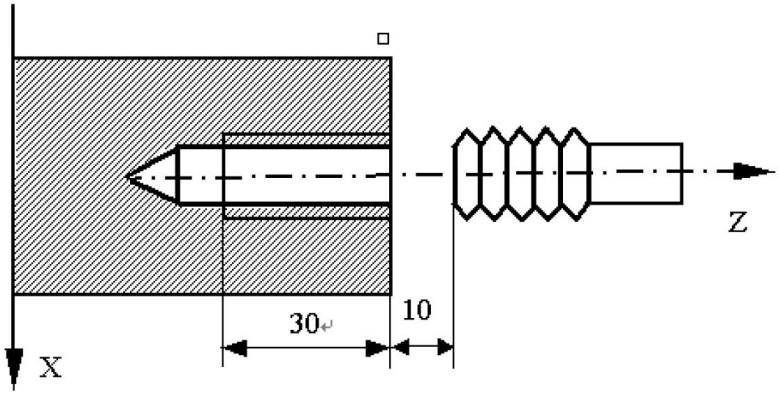


Fig3.8.9 Example of Canned Tapping Cycle

Program: O0011

```
G00 Z90 X0 M03 ;
G93 Z50 F1.5 ;
M03
G00 X60 Z100
M30
```

Example2:

```
G93 Z-100 F5     ; tapping cycle to Z-100;
Z-101            ;  tapping cycle to Z-101;
G00 X50          ;  G00
```

Solution of Canned Tapping Cycle in X direction:

1) P41 in Speed parameter : "compesation mode of arc reverse backlash " , set to 2 , canned tapping cycle in X direction ;

2) Add G19 into this segment when tapping in X direction ; Add G17 into this segment when tapping in Z axis ; Add G18 into next segment after finish tapping.

Example:

G93 G19 X-100 F2 ; Canned Tapping Cycle in X direction
G93 G17 Z-100 F2 ; Canned Tapping Cycle in Z direction
G18 G0 X30 ; Cancel Canned Tapping Cycle

Note:

1. ***If execute G93 after Z moving in positive direction,due to opposite direction, system will make reverse backlahs compensation firstly. We should set P13, reverse backlash parameter in Axis parameter . If configured with stepper motor & stuck , we could set the smaller speed value of reverse backlash compensation , also P41-1 & P41-2. Or input the instruction that let Z axis moves with negative direction before executing G93.***
2. ***The parameters of spindle breaking time will affect the start rotating time after stop. Please pay attention to setting these parameters***
3. ***Z-axis must move in negative direction when tapping.***
4. ***Must start spindle rotating before executing G93.***
5. ***The breaking time of spindle should be short.***
6. ***The rotating speed of spindle should be not too high.***
7. ***For specifying inch thread when specifying I is the same as G32 & G92.***
8. ***When choosing the acceleration and deceleration control mode, if the spindle speed change, there is some delay when making the thread change. So choose the non-speed up or down if require the accuracy. However, configured with stepper motor, the speed of spindle cannot be too high, otherwise it will cause the stuck.***

3.8.4 End Face Turning Cycle G94

From starting point, the cutting cycle of cylindrical surface or taper surface is completed by radial feeding(X axis) and axial(Z axis or X and Z) cutting.

a) Face Cutting Cycle : G94 X(U)_ Z(W)_ F_ ;

X: absolute coordinates of end point of cutting in X direction Unit:mm;

U: different value of absolute coordinates from end point to starting point of cutting in X direction, Unit:mm;

Z: absolute coordinates of end point of cutting in Z direction, Unit:mm;

W: different value of absolute coordinates from end point to starting point of cutting in X direction, Unit:mm;

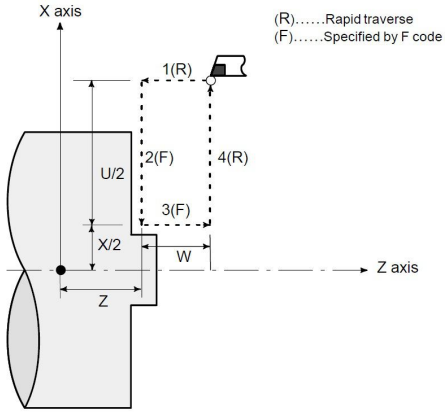


Fig3.8.10 End face loop cutting

In incremental programming, the sign of numbers following addresses U and W depends on the

direction of paths 1 and 2. That is, if the direction of the path is in the negative direction of the Z axis, the value of W is negative.

In single block mode, operations 1, 2, 3, and 4 are performed by pressing the cycle start button once.

Example:

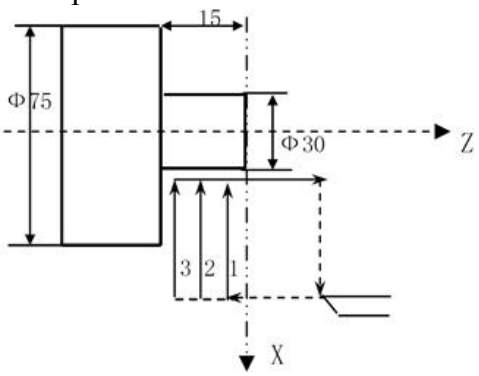


Fig3.8.11 Usage of G94

```
Program:
N10 M03 S1000;
N20 T0101;
N30 G00 X85 Z10 M08;
N40 G01 Z5 F200;
N50 G94 X30 Z-5 F100;
N60 Z－10;
N70 Z－15;
N80 G00 X100 Z60 M09;
N90 T0100 M05;
N100 M30;
```

b)Taper Face Cutting Cycle : G94 X(U)_ Z(W)_ R_ F_;

R_: different value(R value) of absolute coordinates from end point to starting point of cutting in X direction. When the sign of R is not the same that of U, R.

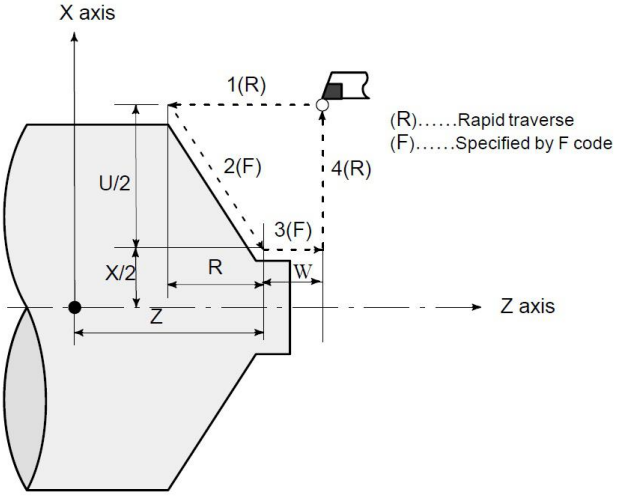
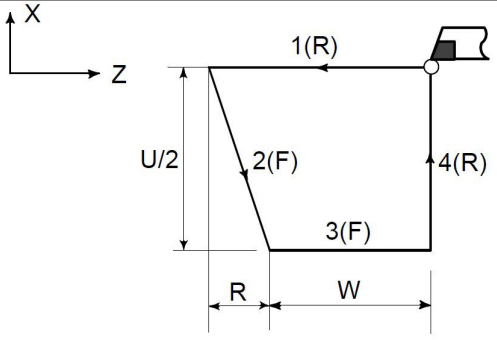
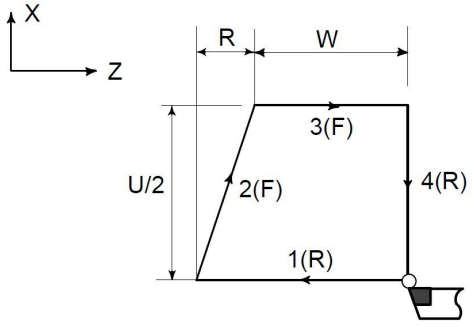
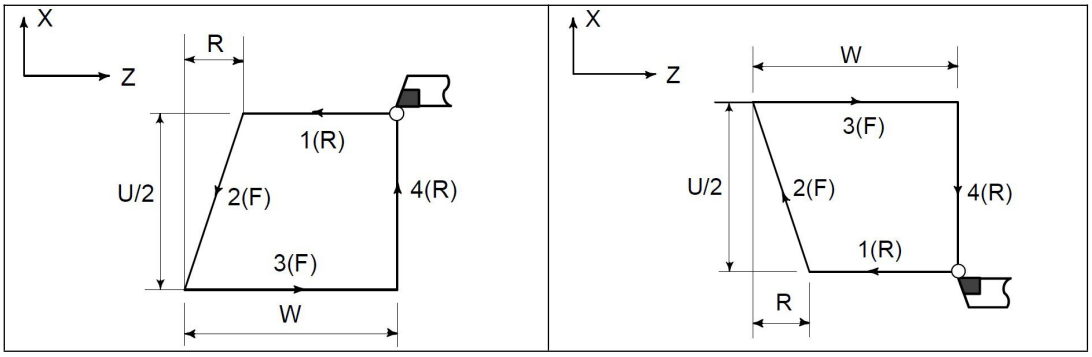


Fig3.8.12 Taper Face Cutting Cycle

c) Signs of numbers specified in the taper cutting cycle

In incremental programming, the relationship between the signs of the numbers following address U, W, and R, and the tool paths are as follows:

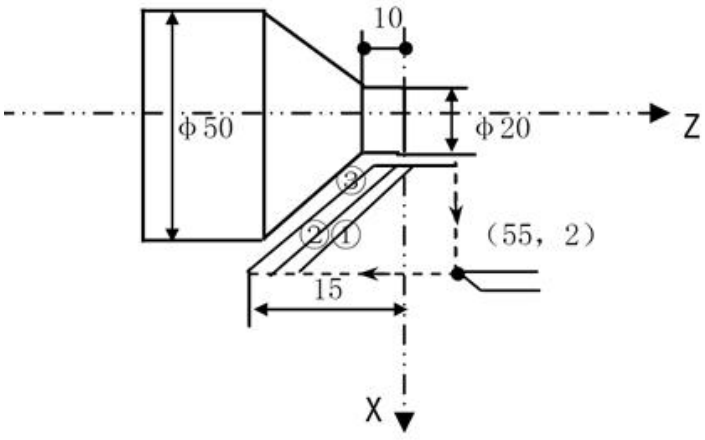
1. U<0 , W<0 , R<0	2. U>0 , W<0 , R
	
3. U<0 , W<0 , R>0 at |R| ≤ |W|	4. U>0 , W<0 , R<0 at |R| ≤ |W|



d) Process of Cycle

1) The tool rapidly traverses from starting point to cutting starting point in Z direction;
2) Cutting feed (linear interpolation) from the cutting starting point to cutting end point;
3) Retract the tool at the cutting feedrate in Z direction (opposite direction to the above-mentioned 1), and return to the position which the absolute coordinates and the starting point are the same;
4) The tool rapidly traverses to return to the starting point and the cycle is completed.

e) Example of Taper Face Cutting Cycle




```
Program
……
N40 G01 X55 Z2 F200;
N50 G94 X20 Z0 R-5 F100;
N60 Z-5;
N70 Z-10;
N80 G00 X Z;
……
(R in N50 program:
R-5=-15-(-10)=-5mm)
```

Fig3.8.13 Example of Taper Face Cutting Cycle

Note:

1. Since data values of X (U), Z (W) and R during canned cycle are modal, if X (U), Z (W), or R is not newly commanded, the previously specified data is effective , except that lead I in Inch thread processing. Thus, when the Z axis movement amount does not vary as in the example below, a canned cycle can be repeated only by specifying the movement commands for the X–axis.

However, these data are cleared, if a one–shot G code expect for G04 (dwell) or a G code in the group 01 except for G90, G92, G94 is commanded.

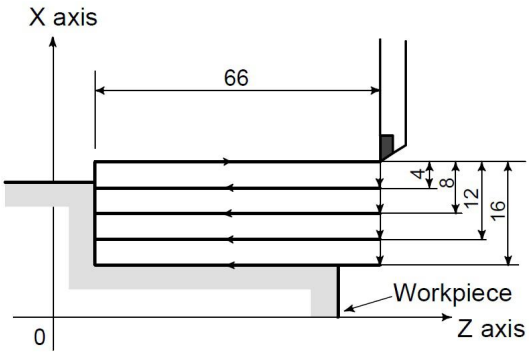
2. The following two applications can be performed.

(1) If an EOB(;) or zero movement commands are specified for the block following that specified with a canned cycle, the same canned cycle is repeated.

(2) Only use "Start" button to run program when input codes in MDI

(3) If the M, S, T function is commanded during the canned cycle mode, both the canned cycle and M, S, or T function can be performed simultaneously. If this is inconvenient, cancel the canned cycle once as in the program examples below (specify G00 or G01) and execute the M, S, or T command. After the execution of M, S, or T terminates, command the canned cycle again.

Example1:



The Cycle in the above figure is executed by the following program.

N030 G90 U-8 .0 W-66.0 F0.4 ;
N031 U-16.0 ;
N032 U-24.0 ;
N033 U-32.0 ;

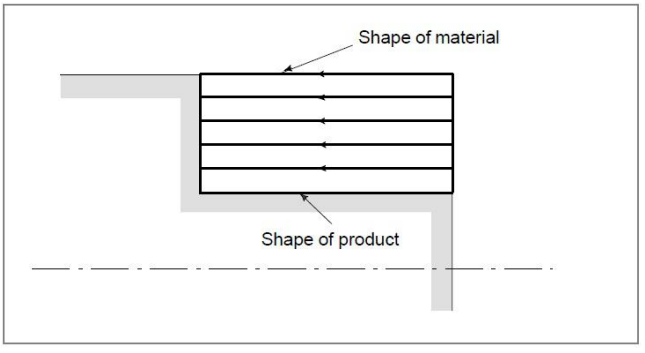
Example2:

N003 T0101 ;
...
N010 G90 X20.0 Z10.0 F0.2
N011 G00 T0202
N012 G90 X20.5 Z10.0

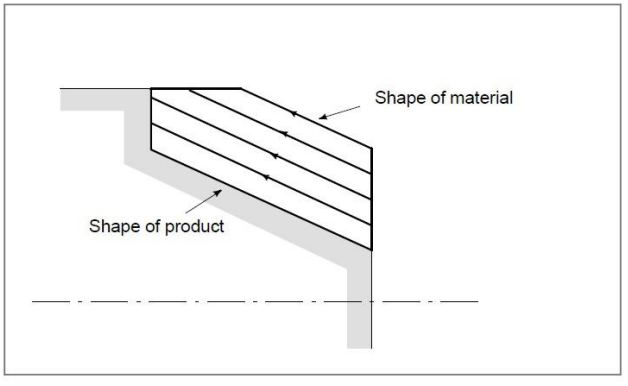
3.8.5 Usage for Canned Cycle

An appropriate canned cycle is selected according to the shape of the material and the shape of the product.

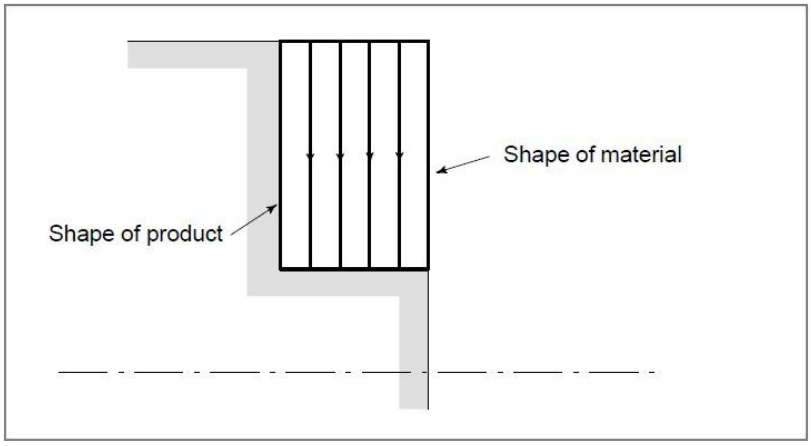
1) Straight Cutting Cycle (G90):



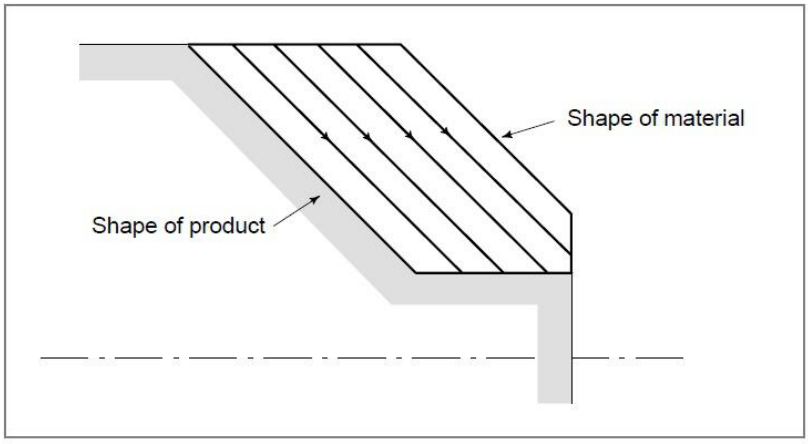
2) Taper Cutting Cycle (G90):



3) Face Cutting Cycle (G94):



4) Taper Face Cutting Cycle (G94):



3.9 Multiple Repetitive Cycle Instructions(G70~G76)

Several types of canned cycles are provided to make programming easier. For instance, the data of the finish work shape describes the tool path for rough machining. And also, a canned cycles for the thread cutting is available.

Multiple cycle instructions of the system includes: Axial roughing turning cycle G71, Radial roughing facing cycle G72, Pattern Repeating Cycle G73, Finishing cycle G70, End face peck drilling cycle G74, Outer/Internal diameter grooving cycle G75 and Multiple thread cutting cycle G76.

When the system executes these instructions, it automatically counts the cutting times and the cutting path according to the programmed path, travels of tool infeed and tool retraction, executes multiple machining cycle (tool infeed → cutting → retract tool → tool infeed), automatically completes the roughing, finishing workpiece and the starting point and the end point of instruction are the same one.

3.9.1 Axial Roughing Turning Cycle (G71)

According to the finishing path, the finishing allowance, the path of tool infeed and tool retract, the system automatically counts the path of roughing， the tool cuts the workpiece in paralleling with Z axis, and the roughing is completed by multiple executing the cutting cycle tool infeed→ cutting→tool retraction. The starting point and the end point are the same one. The instruction is

applied to the formed roughing of non-formed rod.

Format: G71 U(△d) R(e);

G71 P(ns) Q(nf) U(△u) W(△w) F(f) S(s) T(t) ;

N(ns) ..... ;
.....
F___
S___
T___
N(nf) ..... ;

The move command between A and B is specified in the block from sequence number ns to nf .

△d: Depth of cut (radius designation)

Designate without sign. The cutting direction depends on the direction AA'. This designation is modal and is not changed until the other value is designated. Also this value can be specified by the parameter (P1 in User parameter), and the parameter is changed by the program command.

e : Escaping amount

This designation is modal and is not changed until the other value is designated. Also this value can be specified by the parameter (P2 in User parameter), and the parameter is changed by the program command.

ns : Sequence number of the first block for the program of finishing shape.

nf : Sequence number of the last block for the program of finishing shape.

△u: Distance and direction of finishing allowance in X direction (diameter designation). Also this value can be specified by the parameter (P4 in User parameter). Input negative number when processing inner hole.

△w: Distance and direction of finishing allowance in Z direction. Also this value can be specified by the parameter (P5 in User parameter).

F: Feedrate; S: Spindle speed; T: Tool number, tool offset number.

f,s,t : Any F , S, or T function contained in blocks ns to nf in the cycle is ignored, and the F, S, or T function in this G71 block is effective.

If a finished shape of A to A' to B is given by a program as in the figure below, the specified area is removed by Δd (depth of cut), with finishing allowance Δu/2 and Δw left.

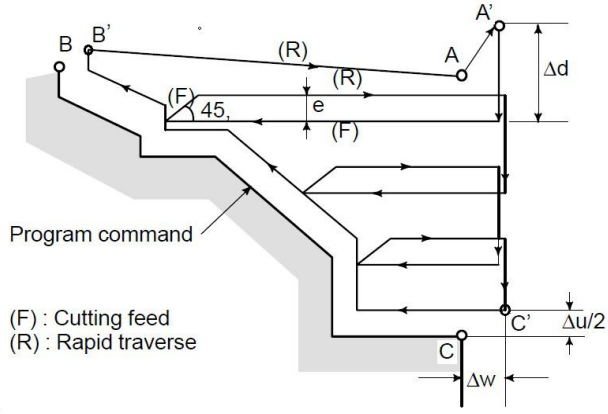


Fig3.9.1 Cutting Path in Axial Roughing Turning Cycle

Execution process:(reference as Fig3.9.1)

① Rapid traverse from A point to A' point , the travel in X direction is Δu, and the travel in Z direction is Δw;

② The travel in X direction from A' point is Δd(tool infeed), ns block is for tool infeed at

rapid traverse speed with G0, is for tool infeed at feedrate F with G71, and its direction of tool infeed is that of A→ C point;

③ Cutting feeds to the roughing path in Z direction, and its direction is the same that of coordinates in Z direction C→B point;

④ The travel of tool retraction is e (45°straight line)at feedrate in X, Z direction, the directions of tool retraction is opposite to that of too infeed;

⑤ Rapid retract at rapid traverse speed in Z direction to the position which is the same that of the coordinates in Z direction of A'-C' point;

⑥ After executing the tool infeed (Δd+e)again in X direction, the end point of traversing tool is still on the middle point of straight line between A' and C'(the tool does not reach or exceed C'), and after executing the tool infeed (Δd+e)again, execute ③;after executing the tool infeed (Δd+e)again, the end point of tool traversing reaches C'point or exceeds the straight line between A' →C'point and execute the tool infeed to C'point in X direction and the execute the next step;

⑦ Cutting feed from C'to B' point along the roughing path;

⑧ Rapid traverse to B' from A point and the program jumps to the next clock following nf block after G71 cycle is ended.

Note:

1. G71 in the use of rough machining cycle,The cycle machining is performed by G71 command with P and Q specification. F, S, and T functions which are specified in the move command between points A and B are ineffective and those specified in G71 block or the previous block are effective. But F S T in program of ns→nf is effectively to fine machining , invalid in rough machining cycle.

2. The tool path between A' and B must be steadily increasing or decreasing pattern in both X and Z axis. When the tool path between A and A' is programmed by G00/G01, cutting along AA' is performed in G00/G01 mode respectively.

3. When the constant surface speed control funtion is enabled, G96 or G97 command specified in the move command between points A and B are ineffective, and that specified in G71 block or the previous block is effective.

4. The tool path between A and A' is specified in the block with sequence number "ns" including G00 or G01, and in this block, a move command in the Z axis cannot be specified.

5. The subprogram cannot be called from the block between sequence number "ns" and "nf".

6. The following four cutting patterns are considered in G71 instruction. All of these cutting cycles are made paralleled to Z axis and the sign of Δu and Δw are as follows:

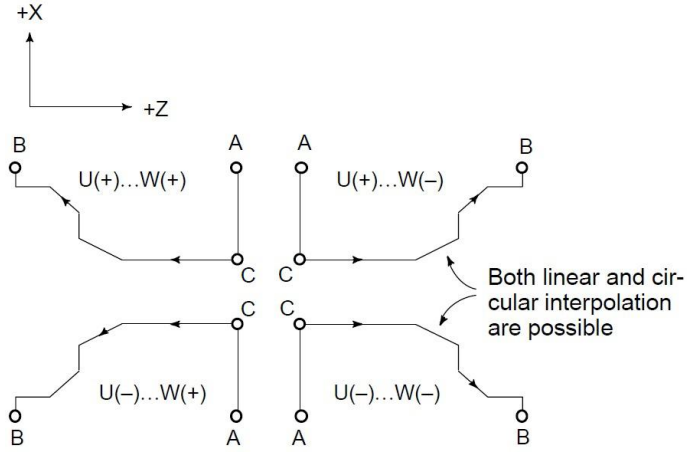


Fig3.9.2 Sign of Δu and Δw in G71

3.9.2 Radial Roughing Facing Cycle (G72)

As shown in the figure below, this cycle is the same as G71 except that cutting is made by a operation parallel to X axis.

According to the finishing path, the finishing allowance, the path of tool infeed and retract tool, the system automatically counts the path of roughing, the tool cuts the workpiece in paralleling with Z axis, and the roughing is completed by multiple executing the cutting cycle tool infeed→cutting feed→tool retraction. The starting point and the end point of G72 are the same one. The instruction is applied to the formed roughing of non-formed rod.

Format: G72 W(△d) R(e) ;

G72 P(ns) Q(ns) U(△u) W(△w) F(f) S(s) T(t) ;

N(ns) ..... ;
.....
F___
S___
T___
N(nf) ..... ;

The move command between A and B is specified in the block from sequence number ns to nf .

The meanings of △d、e、ns、nf、△u、△w、f、s、t are same as those in G71.

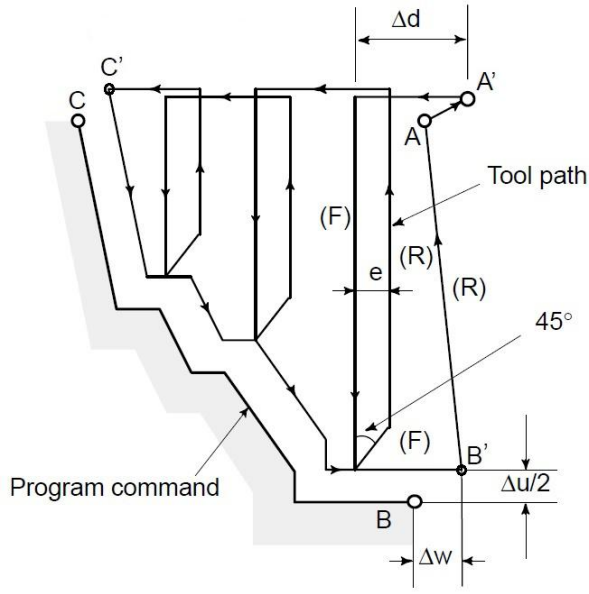


Fig3.9.3 Cutting Path in Radial Roughing Facing Cycle

Execution Process: (reference as Fig3.9.3)

① Rapid traverse from A point to A' point, the travel in X direction is Δu, and the travel in Z direction is Δw;

② The travel in Z direction from A' is Δd(tool infeed), ns block is for tool infeed at rapid traverse speed with G0, is for tool infeed at G72feedrate F in G1, and its direction of tool infeed is that of A→C point;

③ Cutting feeds to the roughing path in X direction, and its direction is the same that of coordinates in X direction C→B point;

④ The travel of tool retraction is e (45º straight line)at feedrate in X, Z direction, the directions of tool retraction is opposite to that of tool infeed ;

⑤ Rapidly retract at rapid traverse speed in X direction to the position which is the same that of the coordinates in Z direction ;

⑥ After executing the tool infeed (Δ d+e)again in Z direction, the end point of traversing tool is still on the middle point of straight line between A' and C'(the tool does not reach or exceed C'), and after executing the tool infeed (Δ d+e)again, execute ③; after executing the tool infeed (Δ d+e)again, the end point of tool traversing reaches C' point or exceeds the straight line between A'➞C' point and
execute the tool infeed to C' point in Z direction and the execute the next step;

⑦ Cutting feed from C'to B' point along the roughing path;

⑧ Rapidly traverse from B' to A point and the program jumps to the next clock following nf block after G71 cycle is completed.

Use G72 to cut the shape, there are four situation. No matter what kind of is the tool parallel the X axis to cut again. △ u, △ w symbols are as follow

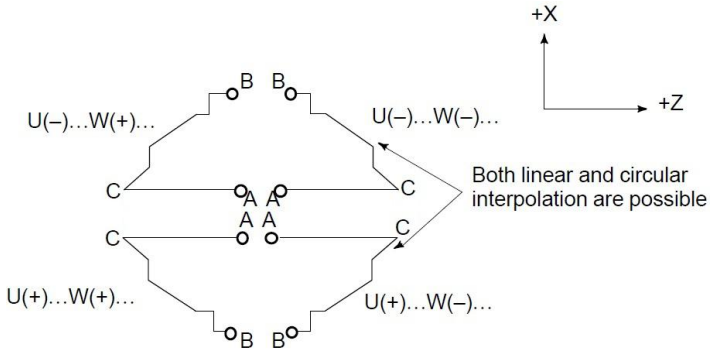


Fig3.9.4 Signs of numbers specified with u and W in G72

The tool path between A and C is specified in the block with sequence number "ns" including G00 or G01, and in this block, a move command in the X axis cannot be specified. The tool path between C and B must be steadily increasing and decreasing pattern in both X and Z axes. Whether the cutting along AC is G00 or G01 mode is determined by the command between A and C.

3.9.3 Pattern Repeating Cycle (G73)

This function permits cutting a fixed pattern repeatedly, with a pattern being displaced bit by bit. By this cutting cycle, it is possible to efficiently cut work whose rough shape has already been made by a rough machining, forging or casting method,etc.The pattern commanded in the program should be as follows:A→A'→B.

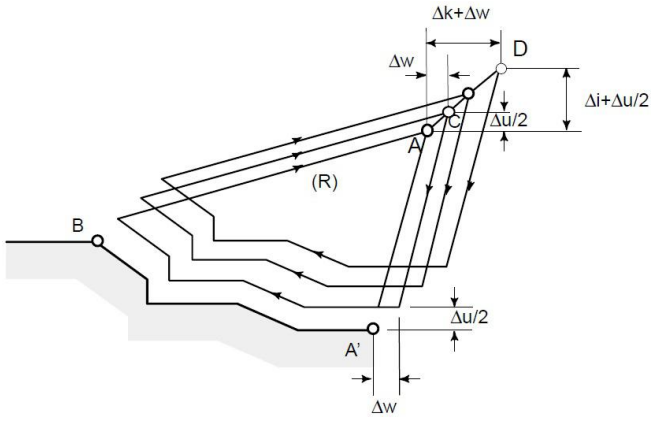


Fig3.9.5 Cutting Path in Pattern Repeating Cycle

Format: G73 U(△i) W(△k) R(d) ;
G73 P(ns) Q (nf) U(△u) W(△w) F(f) S(s) T(t) ;
N(ns) …… ;
…… ;
F_
S_
T_
N(nf) ……;

The move command between A and B is specified in the blocks from sequence number ns to nf.

△i: Distance and direction of relief in the X axis direction (Radius designation).

This designation is modal and is not changed until the other value is designated. Also this value can be specified by P7 in User parameter, and the parameter is changed by the program command.

△k: Distance and direction of relief in the Z axis direction.

This designation is modal and is not changed until the other value is designated. Also this value can be specified by P8 in User parameter, and the parameter is changed by the program command.

d: The number of division.

This value is the same as the repetitive count for rough cutting. This designation is modal and is not changed until the other value is designated. Also, this value can be specified by P6 in User parameter, and the parameter is changed by the program command.

ns : Sequence number of the first block for the program of finishing shape.

nf : Sequence number of the last block for the program of finishing shape.

u : Distance and direction of finishing allowance in X direction (diameter/radius designation)

w : Distance and direction of finishing allowance in Z direction

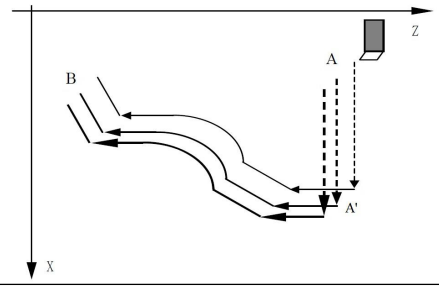
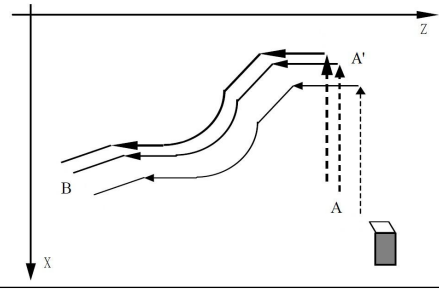
f,s,t : Any F, S, and T function contained in the blocks between sequence number "ns" and "nf" are ignored, and the F, S, and T functions in this G73 block are effective. Others is same as G71.

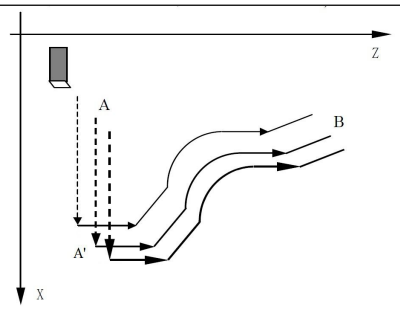
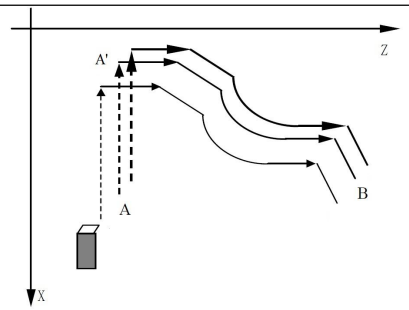
Note: 1. The cycle is according to the program which is between P and Q in G73.The tool backs to A point automatically after finish cycle.

2. Increase or decrease X or Z axis is invalid when using G73.

3. While the values Δi and Δk, or Δu and Δw are specified by address U and W respectively, the meanings of them are determined by the presence of addresses P and Q in G73 block. When P and Q are not specified in a same block, addresses U and W indicates Δi and Δk respectively. When P and Q are specified in a same block, addreses U and W indicates Δu and Δw respectively.

4. The cycle machining is performed by G73 command with P and Q specification.The four cutting patterns are considered. Take care of the sign of Δu, Δw, Δk, and Δi.When the machining cycle is terminated, the tool returns to point A.

1) Δi<0 Δk>0 , △u<0 Δw>0 ;	2) Δi>0 Δk>0 , △u>0 Δw>0 ;
	

3) Δi<0 Δk<0 , △u<0 Δw<0 ;	4) Δi>0 Δk<0 , △u>0 Δw<0 ;
	

3.9.4 Finishing Cycle (G70)

After rough cutting by G71, G72 or G73, the following command permits finishing.

Format: G70 P(ns) Q(nf) ;

(ns) : Sequence number of the first block for the program of finishing shape.

(nf) : Sequence number of the last block for the program of finishing shape.

Note:

1 F, S, and T functions specified in the block G71, G72, G73 are not effective but those specified between sequence numbers "ns" and "nf" are effective in G70.

2 When the cycle machining by G70 is terminated, the tool is returned to the start point and the next block is read.

3 In blocks between "ns" and "nf" referred in G70 through G73, the subprogram cannot be called.

3.9.5 Usages of G71,G72,G73 & G70

3.9.5.1 Example of G71&G70

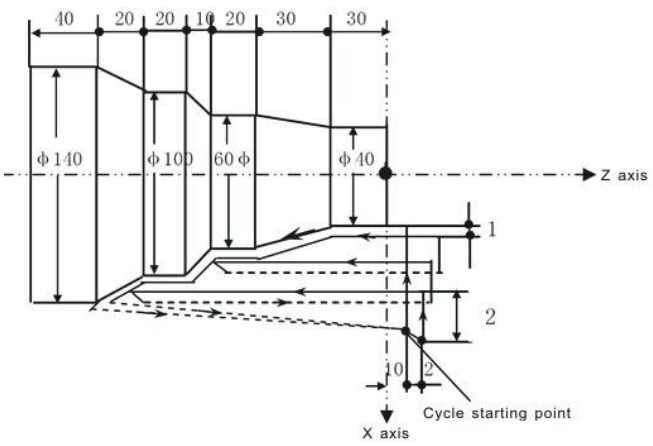


Fig3.9.6 Using of G71&G70

```
N10 M03 S1500;
N20 T0101;
N30 G00 X160 Z10;
N40 G71 U2 R1;
N50 G71 P60 Q120 U2 W1 F100 S2000
N60 G00 X40;
N70 G01 Z-30 F80;
N80 X60 W-30;
N90 W-20;
N100 X100 W-10;
N110 W-20;
N120 X140 W-20;
N130 G70 P60 Q120;
N140 G00 X200 Z50;
N150 T0100 M05;
N160 M30 ;
```

3.9.5.2 Example of G72 & G70

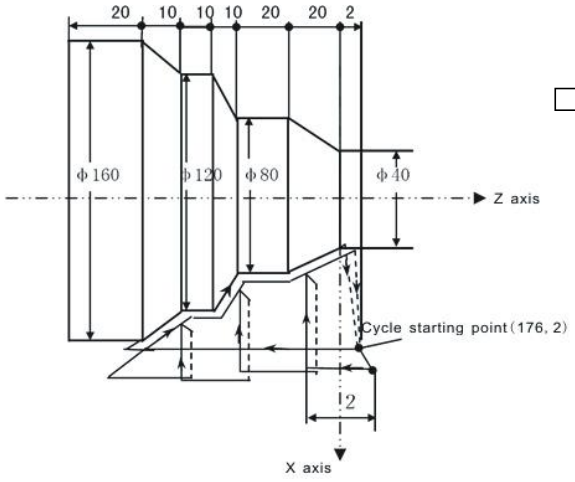


Fig3.9.7 Using of G72&G70

```
N10 M03 S2000;
N20 T0202;
N30 G00 X176 Z2;
N40 G72 W2 R1;
N50 G72 P60 Q120 U2 W1 F100 ;
N60 G00 Z-72;
N70 G01 X160 Z-70 F80;
N80 X120 W10;
N90 W10;
N100 X80 W10;
N110 W20;
N120 X36 W22.08;
N130 G70 P60 Q120;
N140 G00 X200 Z50;
N150 T0200 M05;
N160 M30;
```

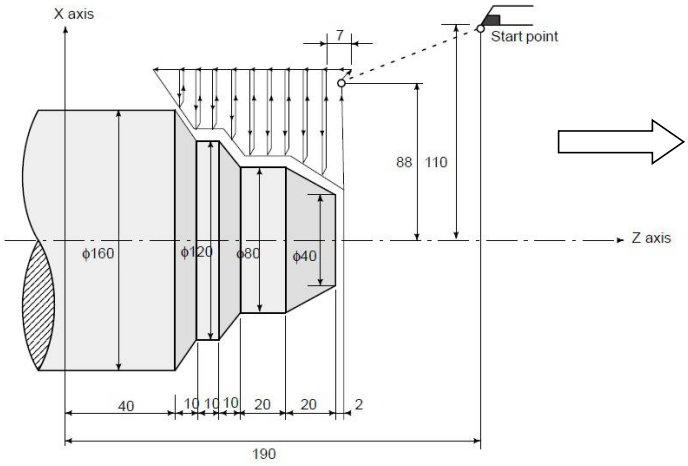


Fig3.9.8 Using of G72&G70

```
(Diameter designation, metric input)
N010 G50 X220.0 Z190.0 ;
N011 G00 X176.0 Z132.0 ;
N012 G72 W7.0 R1.0 ;
N013 G72 P014 Q019 U4.0 W2.0
F0.3 S550 ;
N014 G00 Z58.0 S700 ;
N015 G01 X120.0 W12.0 F0.15 ;
N016 W10.0 ;
N017 X80.0 W10.0 ;
N018 W20.0 ;
N019 X36.0 W22.0 ;
N020 G70 P014 Q019 ;
```

3.9.5.3 Example of G73&G70

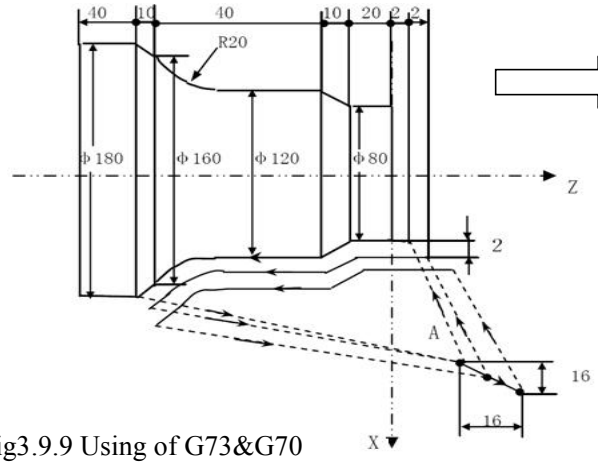


Fig3.9.9 Using of G73&G70

```
N10 M03 S3000;
N20 T0303;
N30 G00 X220 Z40;
N40 G73 U14 W14 R0.010;
N50 G73 P60 Q110 U 4   W2 F100;
N60 G00 X80 Z2;
N70 G01 Z-20 F80;
N80 X120 W-10;
N90 W-20;
N100 G02 X160 W-20 R20;
N110 G01 X180 W-10;
N120 G70 P60 Q110;
N130 G00 X250 Z50;
N140 T0300 M05;
N150 M30;
```

(b)

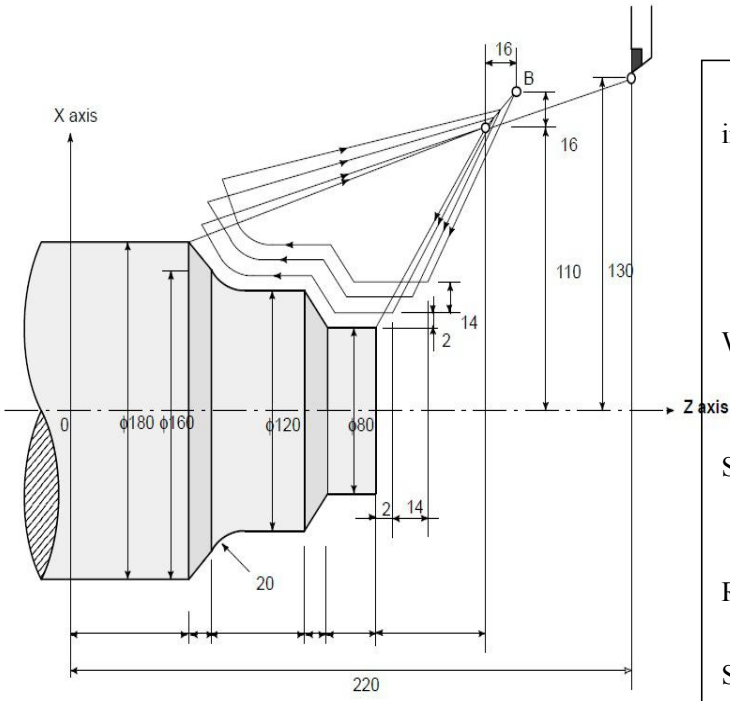


Fig3.9.10 Using of G73 & G70

(Diameter designation, metric input)

N010 G50 X260.0 Z220.0;

N011 G00 X220.0 Z160.0 ;

N012 G73 U14.0 W14.0 R3 ;

N013 G73 P014 Q019 U4.0 W2.0 F0.3 S0180 ;

N014 G00 X80.0 W–40.0 ;

N015 G01 W–20.0 F0.15 S0600 ;

N017 W–20.0 S0400 ;

N018 G02 X160.0 W–20.0 R20.0;

N019 G01 X180.0 W–10.0 S280;

N020 G70 P014 Q019 ;

3.9.6 End Face Peck Drilling Cycle (G74)

The following program generates the cutting path shown in Fig3.9.11. Chip breaking is possible in this cycle as shown below. If X (U) and Pare omitted, operation only in the Z axis results, to be used for drilling.

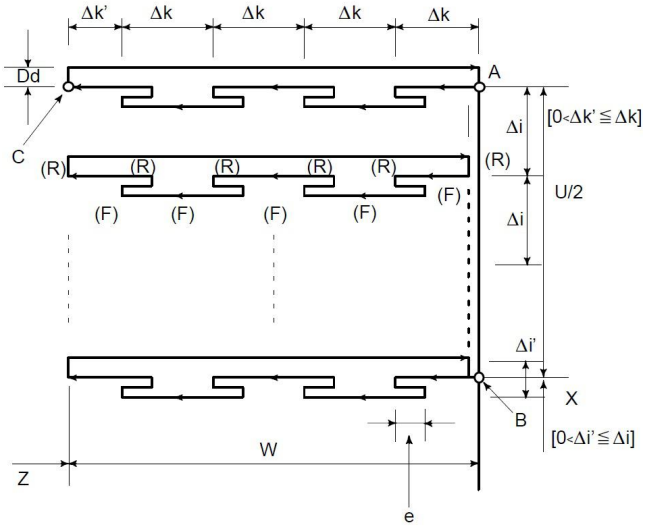


Fig3.9.11 Cutting Path in End Face Peck Drilling Cycle

Format: G74 R(e) ;

G74 X(u) P(△i) Z(w) Q(△k) R(△d) F(f) ;

e: Return amount;

This designation is modal and is not changed until the other value is designated. Also this value can be specified by P10 in User parameter, and the parameter is changed by the program command.

X : X component of point B

U : Incremental amount from A to B

Z : Z component of point C

W : Increment amount from A to C

Δi : Movement amount in X direction (without sign).

Δk : Depth of cut in Z direction (without sign),also can be set by P9 in User parameter. (Unit: um)

Δd : Relief amount of the tool at the cutting bottom. The sign of Δd is always plus(+). However, if address X (U) and Δi are omitted, the relief direction can be specified by the desired sign.

f : Feed rate

NOTE

1. While both e and d are specified by address R, the meanings of them are determined by the present of address X (U). When X(U) is specified, d is used.

2. The cycle machining is performed by G74 command with X(U) specification.

Execution process:(Fig3.9.11)

① Axial (Z axis) cutting feed △k from the starting point of G74, feed in Z negative direction when the coordinates of cutting end point is less than that of starting point in Z direction, otherwise, feed in Z positive direction;

②Axial(Z axis) rapid tool retraction e and its direction is opposite to feeding direction;

③ Cutting feed(Δk+e) again in Z direction, the end point of cutting feed is still in it between startingpoint An of axial cutting cycle and end point of axial tool infeed, cutting feed (Δk+e)again in Z direction and execute ②; after cutting feed (Δk+e)again in Z direction, the end point of cutting feed is on Bn or is not on it between An and Bn cutting feed to Bn in Z direction and then execute ④;

④ Radial(X axis) rapid tool retraction △d（radius value）to Cn , when the coordinates of Bf (cutting end point) is less than that of A (starting point) in X direction, retract tool in X positive, otherwise, retract tool in X negative direction;

⑤ Axial(Z axial) rapid retract tool to Dn, No. n axial cutting cycle is completed. If the current axial cutting cycle is not the last one, execute ⑥ ; if it is the previous one before the last axial cutting cycle, execute ⑦;

⑥ Radial(X axial)rapid tool infeed, and it direction is opposite to ④ retract tool. If the end point of tool infeed is still on it between A and Af (starting point of last axial cutting cycle) after tool infeed(△d+△i) (radius value) in X direction, i.e. n→An+1 and then execute ① (start the next axial cutting cycle); if the end point of tool infeed is not on it between Dn and Af after tool infeed (△d+△i) (radius value) in X direction, rapidly traverse to Af and execute ① to start the first axial cutting cycle;

⑦ Rapidly traverse to return to A in X direction, and G74 is completed.

Example:

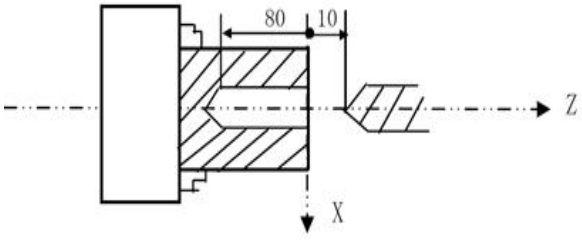


Fig3.9.12 Example of G74

```
N10 G00 X0 Z10;
N20 G74 R2;
N30 G74 Z-80 Q10000
F800;
N40 G00 X50 Z50;
N50 M30;
```

3.9.7 Outer Diameter/Internal Diameter Drilling Cycle (G75)

The following program generates the cutting path shown in Fig3.9.13.

This is equivalent to G74 except that X is replaced by Z. Chip breaking is possible in this cycle, and grooving in X axis and peck drilling in X axis (in this case, Z, W, and Q are omitted) are possible.

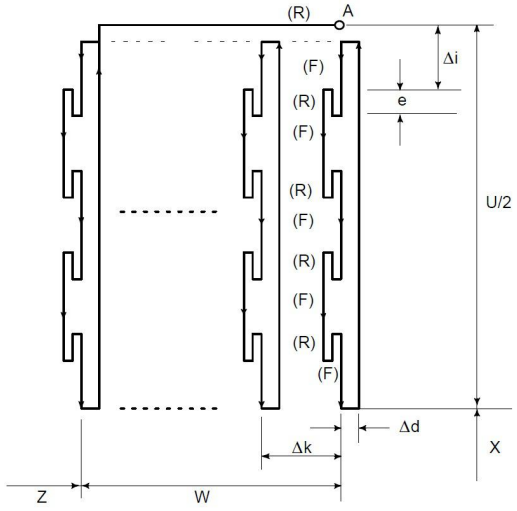


Fig3.9.13 Cutting Path in Outer/Internal Diameter Drilling Cycle

Format: G75 R(e) ;

G75 X(U) P(Δi) Z(w) Q(△k) R(Δd) F(f) ;

X : X component of point B
U : Incremental amount from A to B
Z : Z component of point C
W : Increment amount from A to C

Δi : Movement amount in X direction (without sign),also can be set by P9 in User parameter. (Unit: um)

Δk : Depth of cut in Z direction (without sign),

Δd : Relief amount of the tool at the cutting bottom. The sign of Δd is always plus(+). However, if address X (U) and Δi are omitted, the relief direction can be specified by the desired sign.

f : Feed rate

Both G74 and G75 are used for grooving and drilling, and permit the tool to relief automatically. Four symmetrical patterns are considered,respectively.

Execution process:(Fig3.9.14)

① Radial (X axis) cutting feed △i from the starting point of radial cutting cycle, feed in X negative direction when the coordinates of cutting end point is less than that of starting point in X direction,otherwise, feed in X positive direction;

② Radial(X axis) rapid tool retraction e and its direction is opposite to the feed direction of ①;

③ Cutting feed(Δk+e) again in X direction, the end point of cutting feed is still in it between starting point An of radial cutting cycle and end point of radial tool infeed, cutting feed (Δi+e)again in X direction and execute ②; after cutting feed (Δi+e)again in X direction, the end point of cutting feed is on Bn or is not on it between An and Bn cutting feed to Bn in X direction and then execute ④ ;

④ Axial(Z axis) rapid tool retraction △d（radius value）to Cn , when the coordinates of Bf (cutting end point) is less than that of A (starting point) in Z direction, retract tool in Z positive, otherwise, retract tool in Z negative direction;

⑤ Radial(Z axis) rapid retract tool to Dn, No. n radial cutting cycle is completed. The current radial cutting cycle is not the last one, execute ⑥ ; if it is the previous one before the last radial cutting cycle, execute ⑦;

⑥ Axial(X axis)rapid tool infeed, and it direction is opposite to ④ retract tool. If the end point of tool infeed is still on it between A and Af (starting point of last radial cutting cycle) after tool infeed (△d+△k) (radius value) in Z direction, i.e. Dn→An+1 and then execute ① (start the nextradial cutting cycle); if the end point of tool infeed is not on it between Dn and Af after tool infeed (△d+△k) in Z direction, rapidly traverse to Af and execute ① to start the first radial cuttingcycle;

⑦ Rapidly traverse to return to A in Z direction, and G75 is completed.

Example:

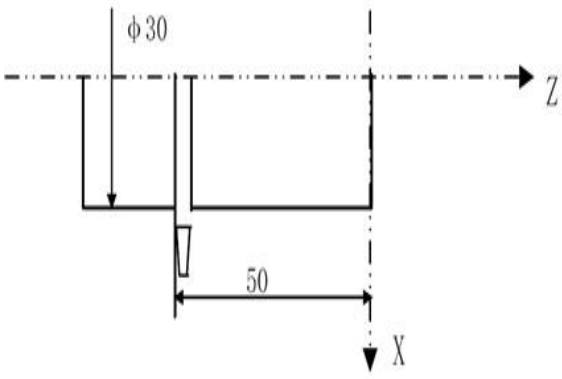


Fig3.9.14 Example of G75

```
N10 M03 S1000;
N20 T0101;
N30 G00 X35 Z-50;
N40 G75 R1;
N50 G75 X-1 P5000 F60;
N60 G00 X100 Z50 M09;
N70 M05;
N80 T0100;
N90 M30;
```

3.9.8 Multiple Thread Cutting Cycle (G76)

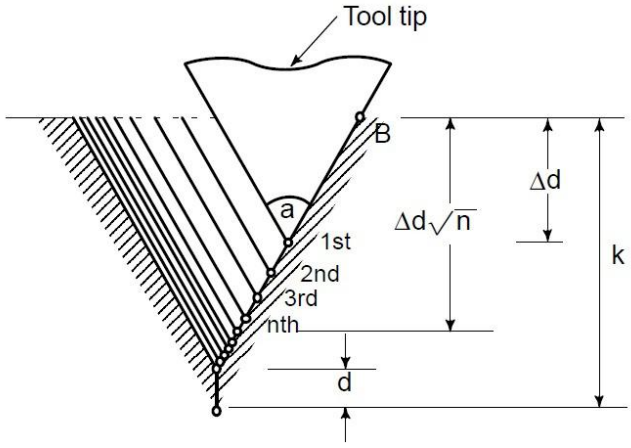


Fig3.9.15 Detailes of Cutting

Format: **G76 P(b)(c)(m)(r)(a) Q(Δdmin) R(d) ;**
G76 X(U)_ Z(W)_ R(i) P(k) Q(Δd) F(I)_ L(L)[or SP];

P actually consists of multiple values which control the thread behavior.

b: 0———Infeed decremently;
1———Equidistant infeed;
2———If the fist feed is too long in digression feed,so divide into two infeed.

c: 0———right enter;
1———left enter;
2———middle enter
3———right and left enter, the first feed is middle.

m: Repetitive count in finishing (1 to 99)

This designation is modal and is not changed until the other value is designated. Also this value can be specified by P11 in User parameter, and the parameterr is changed by the program command.

r: Chamfering amount.

When the thread lead is expressed by L, the value of L can be set from 0.0L to 9.9L in 0.1L increment (2–digit number from 00 to 90). This designation is modal and is not changed until the other value is designated. Also this value can be specified by P12 in User parameter, and the parameter is changed by the program command.

a: Angle of tool tip

One of six kinds of angle, 80°, 60°, 55°, 30°, 29°, and 0°, can be selected,and specified by 2–digit number. This designation is modal and is not changed until the other value is designated. Also this value can be specified by P13 in User Parameter, and the parameter is changed by the program command.

b ,c,m, r, and a are specified by address P at the same time.

Example: When b=2, c=3, m=1, r=1.2K, a=60°, specify as shown below (K is lead of thread).

$P\frac{2}{b}\frac{3}{c}\frac{01}{m}\frac{12}{r}\frac{60}{a}$, coding instructe is : P23011260

Q(Δdmin): Minimum cutting depth. (specified by the radius value)

When the cutting depth of one cycle operation ($\Delta d*\sqrt{n} - \Delta d*\sqrt{n-1}$) becomes smaller than

this limit, the cutting depth is clamped at this value. This designation is modal and is not changed until the other value is designated. Also this value can be specified by P14 in User Parameter, and the parameter is changed by the program command. Unit: um.

R(d): Finishing allowance.

This designation is modal and is not changed until the other value is designated. Also this value can be specified by P15 in User parameter, and the parameter is changed by the program command.

X: Absolute coordinates (unit: mm) of thread end point in X direction;

U: Different value (unit: mm) of absolute coordinates between thread end point and starting point in X direction;

Z: Absolute coordinates (unit: mm) of thread end point in Z direction;

W: Different value (unit: mm) of absolute coordinates between thread end point and starting point in Z direction;

R(i): Difference of thread radius.

If i = 0, ordinary straight thread cutting can be made.

P(k): Height of thread. This value is specified by the radius value.

F: metric thread pitch. (same as G32).0.1~500.000mm.

I: Thread teeth per inch for inch thread. 0.1~99 teeth.

L: multiple thread head numbers.

SP: starting angle: 0-360°, unit is 0.001 degree. No specify means 0 degree.

Δd: Depth of cut in 1st cut (radius value) .Or infeed times.

First cut amount (with G32 threading) in microns; or feed times.

P24 in User parameter is set for meaning of Q (Δ d) of G76. [P24 = 8 ,times of roughing infeed]. When P24 = 8, Q (Δ d) means times that needed to complete the roughing cycle, the default is 1; otherwise Q (Δ d) means that depth of cut in 1st cut. Q (Δ d) , times of infeed, also there are modes of equidistant infeed and decrementally infeed.

The infeed amount and infeed times of roughing in all case are as follows:

1) b = 0, P24 $\neq$ 8, every infeed depth: $\nabla d\sqrt{n}$;

2) b = 0, P24 = 8, each infeed depth is: the same way as a) according to Δ d calculated as: $\frac{(k-d)}{\sqrt{\nabla d}}\sqrt{n}$, infeed times is Δ d;

3) b=1, P24≠8, amount of each infeed: Δ d,times of roughing infeed is (k-d)/ Δ d;

4) b=1, P24=8,the amount of each infeed: (k-d)/ Δ d,roughing feed times for Δ d;

The thread cutting cycle as shown in Fig3.9.16 is programmed by G76 command.

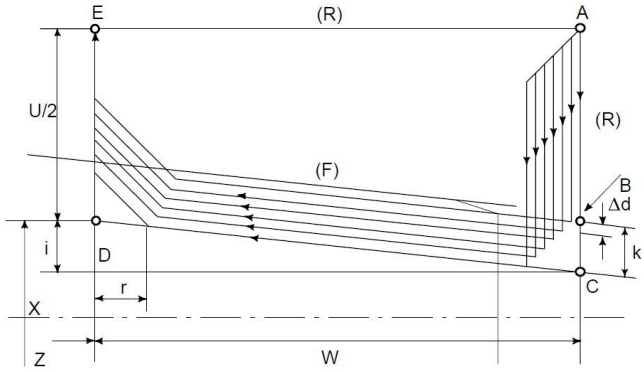


Fig3.9.16 Cutting path in multiple thread cut

Execution process:

① The tool rapidly traverses to B(1st),and the thread cutting depth is △d. The tool only traverses in X direction when a=0; the tool traverses in X and Z direction and its direction is the same that of A→D when a≠0；

② The tool cuts threads paralleling with C→D to the intersection of D→E (r≠0: thread run-out)；

③ The tool rapidly traverses to E point in X direction；

④ The tool rapidly traverses to A point in Z direction and the single roughing cycle is completed；

⑤ The tool rapidly traverses again to tool infeed to B(nth) (is the roughing times), the cutting depth is the bigger value of ($\sqrt{n}\times$△d),($\sqrt{n-1}\times$△d+△dmin) and execute ② if the cutting depth is less than（k-d）; if the cutting depth is more than or equal to（k-d）, the tool infeeds（k-d）to B((n+1)th) ,and then,execute ⑥ to complete the last thread roughing；

⑥ The tool cuts threads paralleling with C→D to the intersection of D→E (r≠0: thread run-out)；

⑦ The tool rapidly traverses to E point in X direction；

⑧ The tool rapidly traverses to A point in Z direction and the thread roughing cycle is completed to execute the finishing；

⑨ After the tool rapidly traverses to B(the cutting depth is k and the cutting travel is d), execute the thread finishing, at last the tool returns to A point and so the thread finishing cycle is completed；

⑩ If the finishing cycle times is less than m, execute ⑨ to perform the finishing cycle, the thread cutting depth is k and the cutting travel is 0; if the finishing cycle times is equal to m, G76 compound thread machining cycle is completed.

Thread Cutting Cycle Retract:

When feed hold is applied during threading in the multiple thread cutting cycle (G76), the tool quickly retracts in the same way as in chamfering performed at the end of the thread cutting cycle. The tool goes back to the start point of the cycle. When cycle start is triggered, the multiple thread cutting cycle resumes.

Note: 1. The meanings of data specified by address P, Q, and R determined by the presence of X(U) and X (W).

2. The cycle machining is performed by G76 command with X (U) and Z (W) specification. By using this cycle , one edge cutting is performed and the load on the tool tip is reduced.Making the cutting depth d for the first path, and dn for the nth path, cutting amount per one cycle is held constant.Four symmetrical patterns are considered corresponding to the sign of each address. The internal thread cutting is available. In the above figure, the feed rate between C and D is specified by address F, and in the other path, at rapid traverse. The sign of incrementaldimensions for the above figure is as follows:

U, W : minus (determined by the direction of the tool path AC and CD.)

R(i) : minus (determined by the direction of the tool path AC.)

P(k) : plus (always)

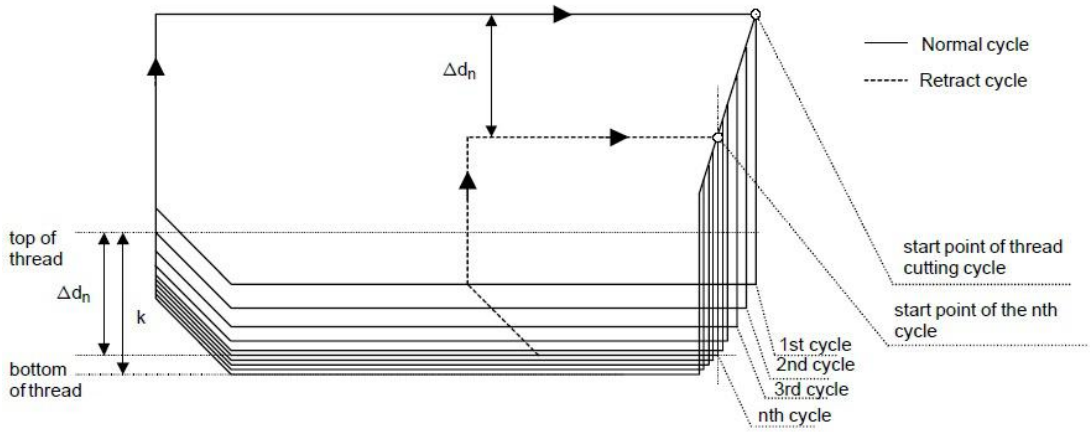
Q(△d) : plus (always)

3. Notes on thread cutting are the same as those on G32 thread cutting and G92 thread cutting cycle.

4. The designation of chamfering is also effective for G92 thread cutting cycle.

5. The tool returns to the cycle start point (cutting depth dn) as soon as the feed hold status isentered

during thread cutting. (dn : cutting depth in nth cut)



6. If start point of thread cutting cycle is close to a workpiece, tool may interfere with the workpiece during retract cycle because of passing along the retract cycle route described at Note 5. Therefore start point of thread cutting cycle must be at least k(height of thread) away from the top of thread.

7. pay attention to cut thread, use G32 to cut is the same as using G92.

8. specify the chamfering amount of thread, it's also effective to the G92 thread cutting circle.

9. when matching step motor, because of the acceleration or deceleration the thread in tail will be inhomogeneous. So should choose the linear acceleration or deceleration to control X axis with G00 to back tail fast.

Example: Multiple Repetitive Cycle G76

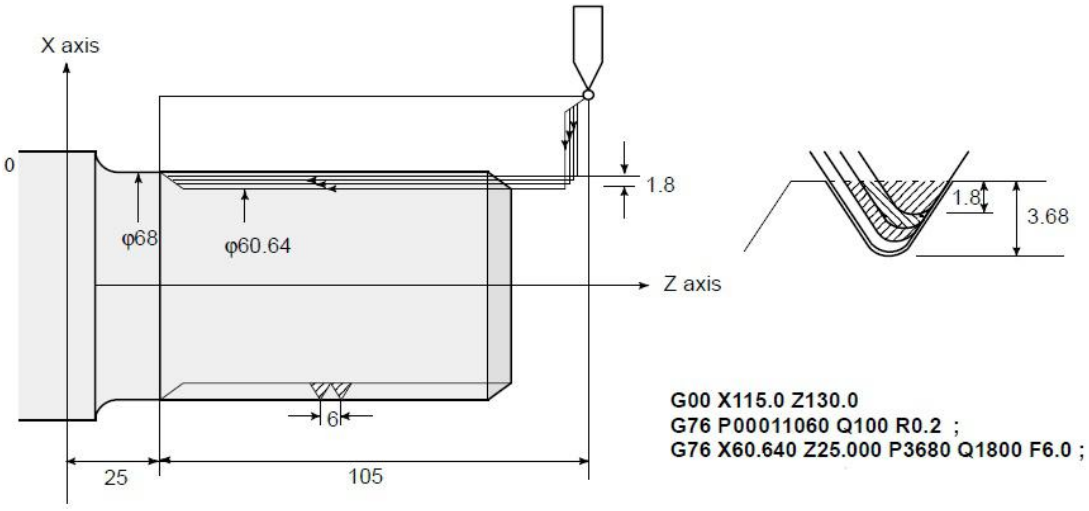


Fig3.9.16 Example of G76

3.9.9 Notes On Multiple Repetitive Cycle (G70 ~ G76)

1. In the blocks where the multiple repetitive cycle are commanded, the addresses P, Q, X, Z, U, W, and R should be specified correctly for each block.

2. In the block which is specified by address P of G71, G72 or G73, G00 or G01 group should be commanded.

3. In MDI mode, G70, G71, G72, or G73 cannot be commanded.G74, G75, and G76 can be commanded in MDI mode.

4. In the blocks in which G70, G71, G72, or G73 are commanded and between the sequence number specified by P and Q, M98 (subprogram call) and M99 (subprogram end) cannot be commanded.

5. In the blocks between the sequence number specified by P and Q, the following commands cannot be specified.

- One shot G code except for G04 (dwell)
- 01 group G code except for G00, G01, G02, and G03
- 06 group G code
- M98 / M99

6. While a multiple repetitive cycle (G70~G76) is being executed, it is possible to stop the cycle and to perform manual operation. But, when the cycle operation is restarted, the tool should be returned to the position where the cycle operation is stopped. If the cycle operation is restarted without returning to the stop position, the movement in manual operation is added to the absolute value, and the tool path is shifted by the movement amount in manual operation.

7. When G70, G71, G72, or G73 is executed, the sequence number specified by address P and Q should not be specified twice or more in the same program.

8. The blocks between the sequence number specified by P and Q on the multiple repetitive cycle must not be programmed by using “Direct Drawing Dimensions Programming” or “Chamfering/Corner R”.

9. G74, G75, and G76 also do not support the input of a decimal point for P or Q. The least input increments are used as the units in which the amount of travel and depth of cut are specified.

10. When #1 = 2500 is executed using a custom macro, 2500.000 is assigned to #1. In such a case, P#1 is equivalent to P2500.

11. Tool nose radius compensation cannot be applied to G71, G72, G73, G74, G75, or G76.

12. The multiple repetitive cycle cannot be executing during DNC operation.

13. Interruption type custom macro cannot be executed during executing the multiple repetitive cycle.

3.10 Skip Function(G31,G311)

Linear interpolation can be commanded by specifying axial move following the G31 command, like G01. If an external skip signal is input during the execution of this command, execution of the command is interrupted and the next block is executed.

The skip function is used when the end of machining is not programmed but specified with a signal from the machine, for example, in grinding. It is used also for measuring the dimensions of a workpiece.

For details of how to use this function, refer to the manual supplied by the

machine tool builder.

Format: G31 IP_ P_ ;
G311 IP_ P_ ;

G31&G311 are One-shot G code.(It is effective only in the block in which it is specified).

IP_: coordinate value ;

P_ : Specify jumping line number & detecting if input point is valid;

P(a)(b)(c)

a: Jumping line number specfied by " N**" ;when missed, stop running current line, and jump to next block and run;

b: 10 or 20 ; 10 means that when input point is valid, skip to specfied line, when input point is invalid, don't skip, keep on running or alarm hint; 20 means that when input point is invalid, skip to specfied line ; when input point is valid, don't skip , keep on running or alarm hint;

c: Specify detecting input point. address of input point, X00~X39

P 56 10 24 } when Input point X24 is valid, stop running current
a b c liune, jump to N56 block and run.

Difference between G31 & G311: When system don't detect signal of specified input point, G31 don't hint alarm & keep on running program; G311 will hint that don't input is valid & stop running, after Press "Enter", it will go on run program ;

The coordinate values when the skip signal is turned on can be used in a custom macro because they are stored in the custom macro system variable as follows:

#5021 X axis coordinate value
#5022 Third axis (Cs axis) coordinate value
#5023 Z axis coordinate value
#5024 4th axis (A axis) coordinate value

Example: G31 X50 Z100 F100 P**33**1022 ;if X22 is valid then go to **N33**(line no.).
G311 X50 Z100 F100 P2021 ;if X21 is invalid then go to next line.Valid-Alarm.

3.11 Block Cycle (G22,G800)

G22 is a program loop instruction, G800 is the end of the cycle instruction. Both must be paired for parts machining process requires repeated occasions. L is the number of cycles, ranging from 1-99999. Cycle instructions can be nested.

Format:

G22 L_
.
. } **Block Cycle**
.
G800 ;End

Example:

```
N0000   M03 M08
N0001   G0   X200 Z200
N0002   G01 W-100 F300
N0003   G22   L6              ; 6 times cycle
N0004   G01   U-22   F100
```

```
N0005   W－11   U6
N0006   W-30
N0007   W-10 U5
N0008   G0   U10
N0009   W51
N0010   G800                    ;loop end
N0011   G26
N0012   M30
```

3.12 Return to Starting Point (G26,G261~G264)

These instructions are used for return back to the starting point of the program.Starting point is coordinate position of N0000 block. The returning speed is same to G00 speed.

Format:

G26 ; All Feeding axes return to starting point.
G261 ; X-axis return to starting point
G262 ; Y(C)-axis return to starting point
G263 ; Z-axis return to starting point
G264 ; A-axis reutrn to starting point

3.13 Save Current Position (G25)

G25 is used for memory current coordinate position of all axes(XZYA), save current position as specified point.

Format: G25 ; Save current coordinate

3.14 Return to Specified Position (G61,G611~G614)

These instructions are used for return to point specified by G25.

G61 ; all axes return to specified point ;
G611 ; X-axis returns to specified point ;
G612 ; Y(C)-axis returns to specified point ;
G613 ; Z-axis returns to specified point ;
G614 ; A-axis returns to specified point ;

Note: If user don't use G25 to save current position, these instructions will return to starting point as G26.

Example:

```
N0000   G0 X20   Z80              ;
N0001   G01 U5 W－16 F200         ;
N0002   W－100                    ;
N0003   G00 U10                   ;
N0004   Z80                       ;
N0005   G25                       ; save current positon (X35,Z80)
N0006   G01 U10 W-30              ;
N0007   G0 X100 Z200              ;
N0008   G61                       ; return to (X35,Z80)
N0009   M2                        ; End of program
```

3.15 Return to Reference Position (G28)

The reference position is a fixed position on a machine tool to which the tool can easily be moved by the reference position return function.For example, the reference position is used as a position at which tools are automatically changed.

If there is machine zero point(hardware switch),it is also reference point; when user set float zero point(software switch) as home and system will take float zero point as reference point.

Reference Position offset of axes to home can be set by parameters.

X-axis offset: P32 in Axis parameters , unit: 0.01mm.

Z-axis offset: P33 in Axis parameters, unit: 0.01mm.

C-axis offset: P116 in Axis parameters, unit: 0.01mm.

A-axis offset: P214 in Axis parameters, unit: 0.01mm.

User could return reference point in Manual (Press “Return” key) or in Auto (Use G28 instruction).

Tools are automatically moved to the reference position via an intermediate position along a specified axis.

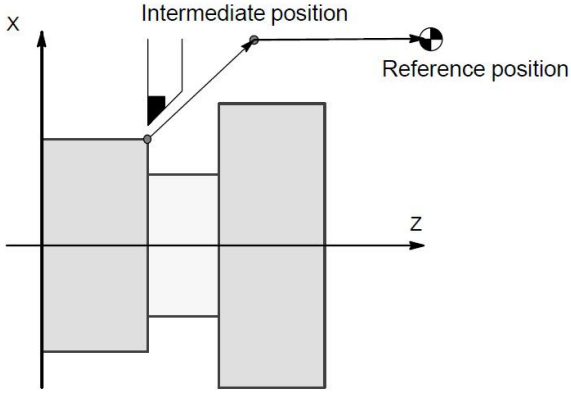


Fig3.15.1 Reference Position Return

Format: G28 IP_;

IP_:Command specifying the intermediate position of all feeding axes (Absolute/incremental command)

G281 ; Only X-axis return to reference position
G282 ; Only Y(C)-axis return to reference position,
G283 ; Only Z-axis return to reference position,
G284 ; Only A-axis return to reference position,

Note: 1. When use G282 instruction, Y(C)-axis return reference position,the system only detect Y0 (PIN2 of CN3 plug),don’t detect Z0 signal of encoder.

2. When use M800, C-axis return to zero position(Z0) of encoder.please check details of M800 in M code chapter.

3. When the G28 insruction is specified when manual return to the reference position has not been performed after the power has been turned on, the movement from the intermediate point is the same as in manual return to the reference position.In this case, the tool moves in the direction for reference position return specified by P28 in Axis parameter. Therefore the specified intermediate position must be a position to which reference position return is possible.

4. When use G28 instruction to return to reference position, if just specify intermediate position of some axes, which can return to reference position, the others axes that don't be specified cannot return to reference position.

5. Before run these codes, reference position must be set well.

6. After return to reference positoin, system will cancel tool compensation automatically.

3.16 Coordinate System

By teaching the CNC a desired tool position, the tool can be moved to the position. Such a tool position is represented by coordinates in a coordinate system. Coordinates are specified using program axes.

When two program axes, the X – axis and Z – axis, are used, coordinates are specified as follows:

X_Z_

This command is referred to as a dimension word.

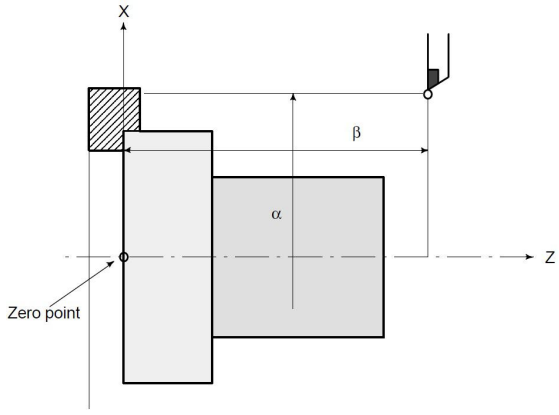


Fig3.16.1 Tool Position Specified by XαZβ

Coordinates are specified in one of following three coordinate systems:

(1) Machine coordinate system

(2) Workpiece coordinate system

(3) Local coordinate system

The number of the axes of a coordinate system varies from one machine to another. So, in this manual, a dimension word is represented as **IP_**.

3.16.1 Machine Coordiante System (G53)

The point that is specific to a machine and serves as the reference of the machine is referred to as the machine zero point. A machine tool builder sets a machine zero point for each machine.

A coordinate system with a machine zero point set as its origin is referred to as a machine coordinate system.

A machine coordinate system is set by performing manual reference position return after power–on (see Chapter5.5.3). A machine coordinate system, once set, remains unchanged until the power is turned off.

G53 IP_ ;

IP_ : absolute dimension word

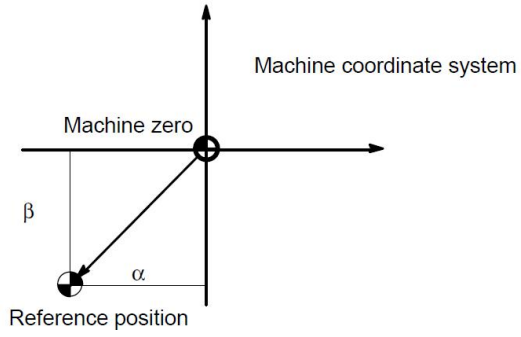
When a position has been specified as a set of machine coordinates, the tool moves to that position by means of rapid traverse. G53, used for selecting the machine coordinate system, is a

one–shot G code. Any commands based on the selected machine coordinate system are thus effective only in the block containing G53. The G53 command must be specified using absolute values. If incremental values are specified, the G53 command is ignored. When the tool is to be moved to a machine–specific position such as a tool change position, program the movement in a machine coordinate system based on G53.

Note: 1. When the G53 command is specified, cancel the tool nose radius compensation and tool offset.

2. Since the machine coordinate system must be set before the G53 command is specified, at least one manual reference position return or automatic reference position return by the G28 command must be performed after the power is turned on. This is not necessary when an absolute–position detector is attached.

When manual reference position return is performed after power–on, a machine coordinate system is set so that the reference position is at the coordinate values of (α, β) set by P32 & P33 in Axis parameter .



3.16.2 Workpiece Coordinate System

A coordinate system used for machining a workpiece is referred to as a workpiece coordinate system. A workpiece coordinate system is to be set with the NC beforehand **(setting a workpiece coordinate system)**.

A machining program sets a workpiece coordinate system **(selecting a workpiece coordinate system)**.

A set workpiece coordinate system can be changed by shifting its origin **(changing a workpiece coordinate system)**.

A workpiece coordinate system can be set using one of three methods:

(1)Method using G50

A workpiece coordinate system is set by specifying a value after G50 in the program.

(2) Automatic setting

A workpiece coordinate system is automatically set when manual reference position return is performed .

(3) Mothod of using G54 to G59

Make settings on the MDI panel to preset six workpiece coordinate systems. Then, use program commands G54 toG59 to select which workpiece coordinate system to use.

When an absolute command is used, a workpiece coordinate system must be established in any of the ways described above.

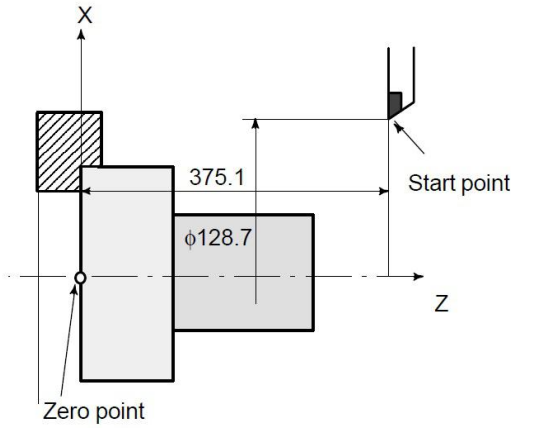
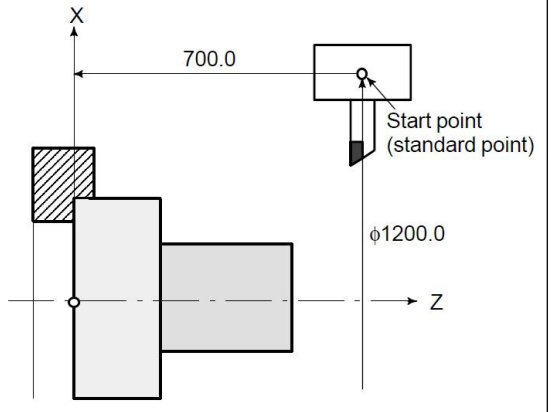
3.16.3 Setting a Workpiece Coordinate System(G50)

Format: G50 IP_ ;

A workpiece coordinate system is set so that a point on the tool, such as the tool tip, is at

specified coordinates. If IP is an incremental command value, the work coordinate system is defined so that the current tool position coincides with the result of adding the specified incremental value to the coordinates of the previous tool position. If a coordinate system is set using G50 during offset, a coordinate system in which the position before offset matches the position specified in G50 is set. And after workpiece coordinate system set well, absolute coordiante position of following commands are based on this coordinate system. Setting solution of G50 workpiece coordiante system(See Chapter of coordiante system set)

Note: In the status of compensation, if use G50 to set the workpiece coordinate system, position before make tool comepensation is based on G50 workpiece coordinate system. Usually cancel the tool compensation first of all before starting program. After return to reference positoin, system will cancel tool compensation automatically.

Example1:	Example2: Base Point
Setting the coordinate system by the G50X128.7Z375.1; command (Diameter designation)	Setting the coordinate system by the G50X1200.0Z700.0; command (Diameter designation)
	

3.16.4 Selecting a Workpiece Coordinate System

The user can choose from set workpiece coordinate systems as described below.

(1)G50 or automatic workpiece coordinate system setting

Once a workpiece coordinate system is selected, absolute commands work with the workpiece coordinate system.

(2)Choosing from six workpiece coordinate systems set in the MDI

By specifying a G code from G54 to G59, one of the workpiece coordinate systems 1 to 6 can be selected.

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

Workpiece coordinate system 1 to 6 are established after reference position return after the power is turned on.

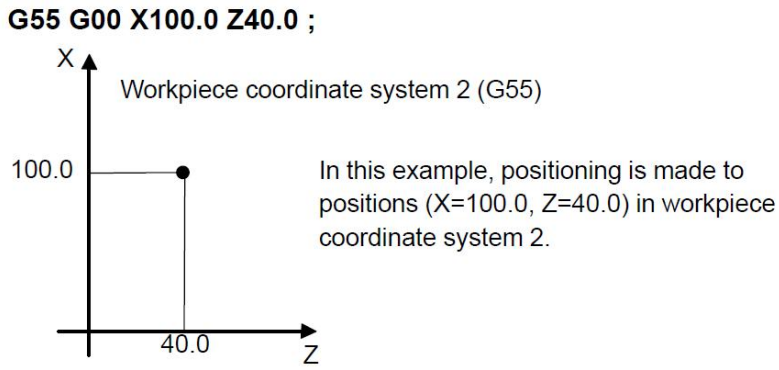


Fig3.16.2 Example of Workpiece Coordinate System

3.16.5 Changing Workpiece Coordinate System

The six workpiece coordinate systems specified with G54 to G59 can be changed by changing an external workpiece zero point offset value or workpiece zero point offset value.

Three methods are available to change an external workpiece zero point offset value or workpiece zero point offset value.

(1) Inputting from the MDI panel (see III－11.4.9)

(2) Programming by G50

(3) Using the external data input function

An external workpiece origin offset can be changed by using a signal input to the CNC,also alter coordinate system in Coor parameter.

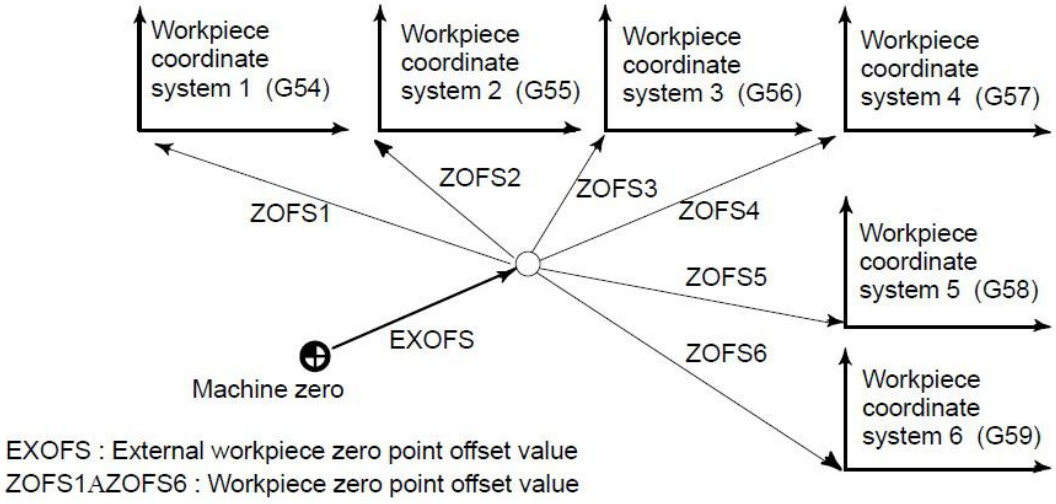


Fig3.16.3 Workpiece Zero Point Offset

Format: Changing by G50 : G50 IP_ ;

By specifying G50IP_;, a workpiece coordinate system (selected with a code from G54 to G59) is shifted to set a new workpiece coordinate system so that the current tool position matches the specified coordinates (IP_).

If IP is an incremental command value, the work coordinate system is defined so that the current tool position coincides with the result of adding the specified incremental value to the coordinates of the previous tool position. (Coordinate system shift)

Then, the amount of coordinate system shift is added to all the workpiece zero point offset values. This means that all the workpiece coordinate systems are shifted by the same amount.

Examples:

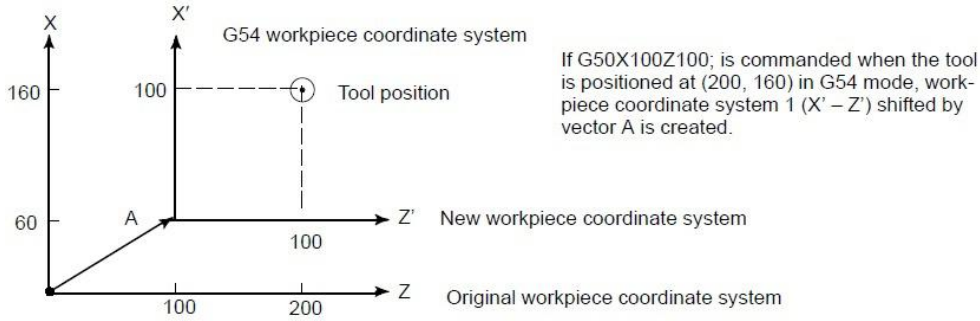


Fig3.16.4 Example1 of Workpiece Coordiante System

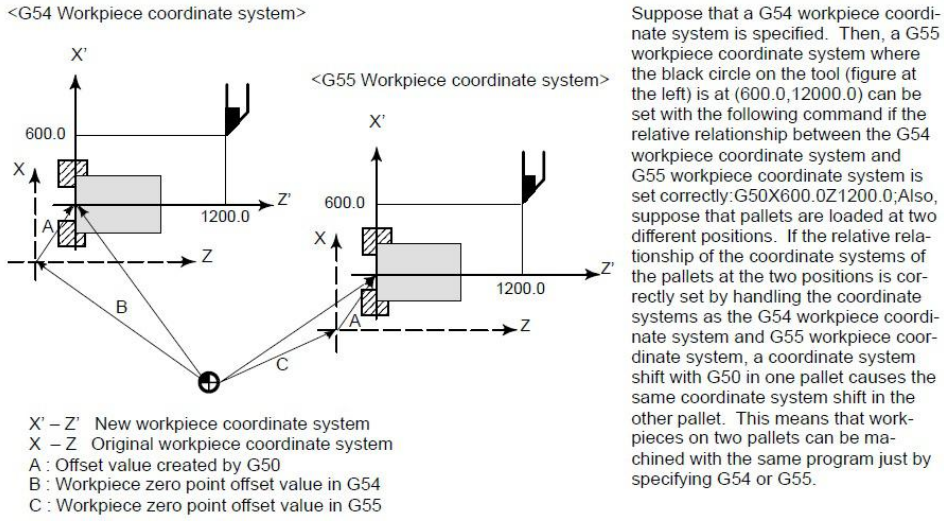


Fig3.16.5 Example2 of Workpiece Coordinate System

3.16.6 Workpiece Coordinate System Shift

When the coordinate system actually set by the G50 command or the automatic system setting deviates from the programmed work system, the set coordinate system can be shifted .

Set the desired shift amount in the work coordinate system shift memory.

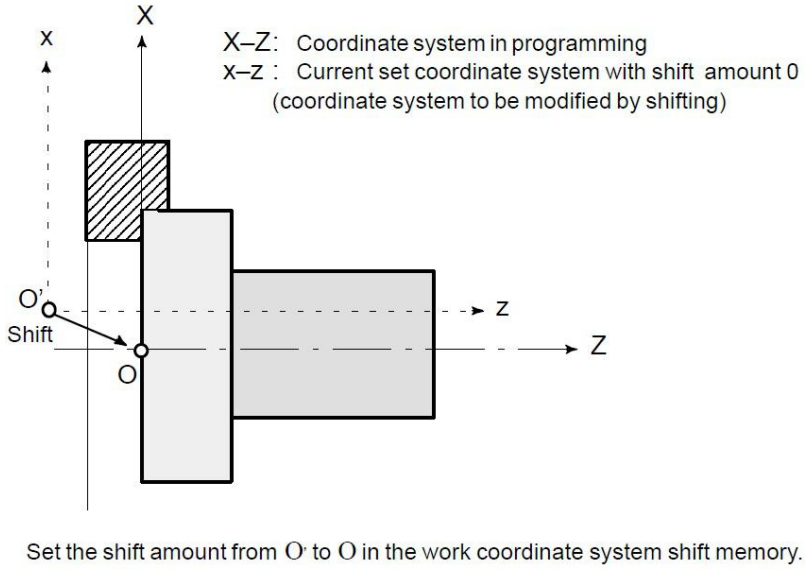


Fig3.16.6 Workpiece Coordinate System Shift

3.16.7 Local Coordinate System

When a program is created in a workpiece coordinate system, a child workpiece coordinate system may be set for easier programming. Such a child coordinate system is referred to as a local coordinate system.

Format: G52 IP_; Setting the local coordinate system

G52 IP 0 ; Canceling of the local coordinate system

IP_: Origin of the local coordinate system

By specifying G52IP_;, a local coordinate system can be set in all the workpiece coordinate systems (G54 to G59). The origin of each local coordinate system is set at the position specified by IP_ in the workpiece coordinate system.

Once a local coordinate system is established, the coordinates in the local coordinate system are used in an axis shift command. The local coordinate system can be changed by specifying the G52 command with the zero point of a new local coordinate system in the workpiece coordinate system.

To cancel the local coordinate system and specify the coordinate value in the workpiece coordinate system, match the zero point of the local coordinate system with that of the workpiece coordinate system.

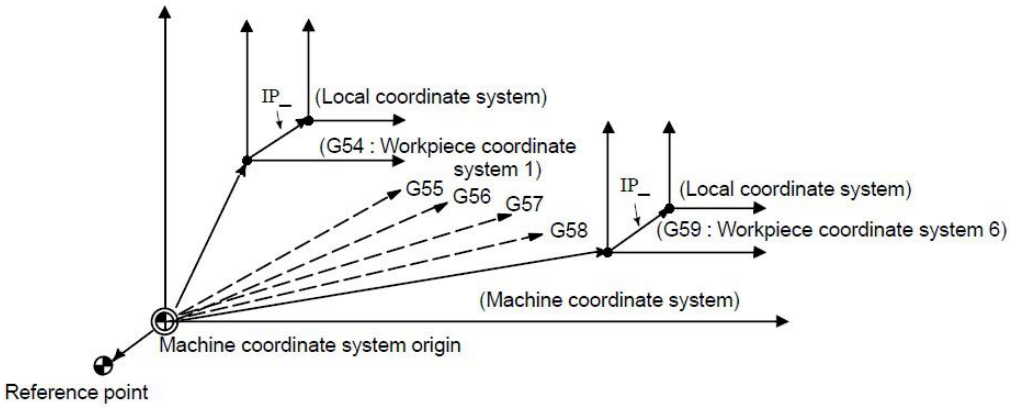


Fig3.16.7 Setting the Local Coordinate System

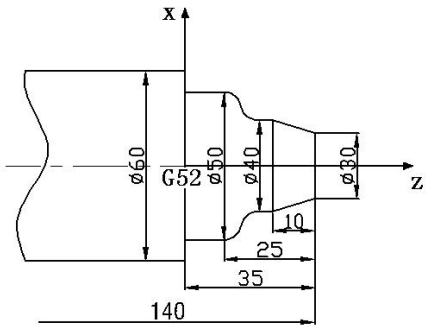
Note:1.The local coordinate system setting does not change the workpiece and machine coordinate systems.

2. When G50 is used to define a work coordinate system, if coordinates are not specified for all axes of a local coordinate system, the local coordinate system remains unchanged. If coordinates are specified for any axis of a local coordinate system, the local coordinate system is canceled.

3. G52 cancels the offset temporarily in tool nose radius compensation.

4. Command a move command immediately after the G52 block in the absolute mode.

Example1:



```
N1 G00 X60 Z20
N2 G52 X0 Z-236
N3 T0101
N4 M03 S800 M08
N5 G01 Z35 F100
N6 X-1
N7 X70
N8 G71 U2 R1
N8 G71 P10 Q15 U0.5 W0.5 F150
N10 G01 X30
N11 X40 Z25
N12 Z20
N13 G02 X50 Z15 R5
N14 G03 X60 Z10 R5
N15 G01 Z0
N16 G00 X70
N17 G52 X0 Z0
N18 M05
N19 M30
```

Example2:Processing 5pcs same workpieces on one workpiece.

```
G22 L5
X50 Z60; workpiece
......
G52 W-50; each Z-axis relative offset 50mm
G800
G52 X0 Z0; cancel the local coordinate system
G0 X100 Z100
M02
```

3.17 Constant Surface Speed Control (G96/G97)

Specify the surface speed (relative speed between the tool and workpiece) following S. The spindle is rotated so that the surface speed is constant regardless of the position of the tool.

Format: G96 SOOOO ; Constant surface speed control command

↑ Surface Speed (m/min or feet/min)

G97 SOOOO ; Constant surface speed control cancel command

↑ Spindle Speed (min^{-1})

G50 S_ ; The maximum spindle speed (min^{-1}) follows S.

G96 (constant surface speed control command) is a modal G code. After a G96 command is specified, the program enters the constant surface speed control mode (G96 mode) and specified S values are assumed as a surface speed. A G96 command must specify the axis along which constant surface speed control is applied. A G97 command cancels the G96 mode.When constant surface speed control is applied, a spindle speed higher than the value specified in G50S_; (maximum spindle speed) is clamped at the maximum spindle speed. When the power is turned on, the maximum spindle speed is not yet set and the speed is not clamped. S (surface speed) commands in

the G96 mode are assumed as S = 0 (the surface speed is 0) until M03 (rotating the spindle in the positive direction) or M04 (rotating the spindle in the negative direction) appears in the program.

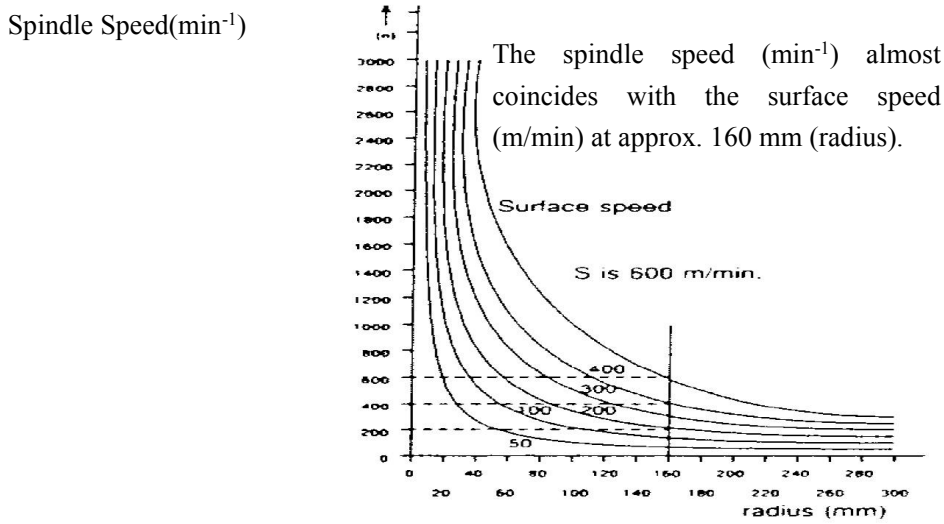


Fig3.17.1 Relation between workpiece radius, spindle speed and surface speed

To execute the constant surface speed control, it is necessary to set a workpiece coordinate system , Z axis, (axis to which the constant surface speed control applies) becomes zero.

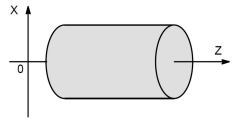
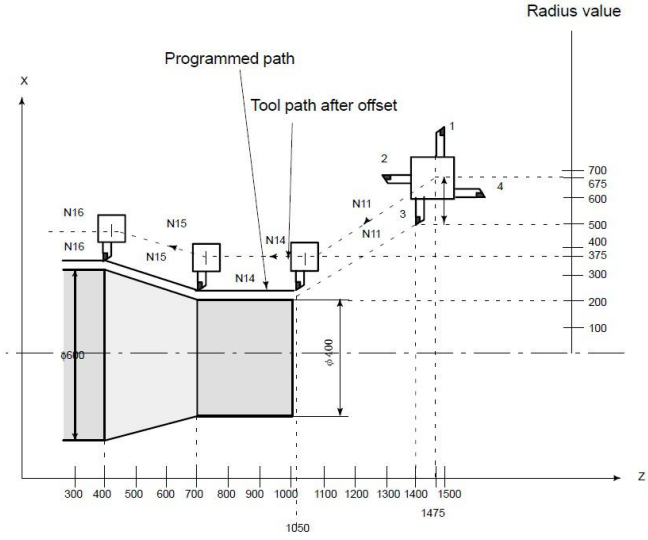


Fig3.17.2 Example of the workpiece coordinate system for constant surface speed control

The constant surface speed control is also effective during threading. Accordingly, it is recommended that the constant surface speed control be invalidated with G97 command before starting the scroll threading and taper threading, because the response problem in the servo system may not be considered when the spindle speed changes.

In a rapid traverse block specified by G00, the constant surface speed control is not made by calculating the surface speed to a transient change of the tool position, but is made by calculating the surface speed based on the position at the end point of the rapid traverse block, on the condition that cutting is not executed at rapid traverse.



Example1:

N8 G00 X1000.0Z1400.0 ;
N9 T33;
N11 X400.0Z1050.0;
N12 G50S3000 ; (Designation of max. spindle speed)
N13 G96S200 ; (Surface speed 200m/min)
N14 G01 Z 700.0F1000 ;
N15 X600.0Z 400.0;
N16 Z … ;

Fig3.17.3 Example of Constant surface speed Control for G00

The CNC calculates the spindle speed which is proportional to the specified surface speed at the position of the programmed coordinate value on the X axis. This is not the value calculated according to the X axis coordinate after offset when offset is valid. At the end point N15 in the example above, the speed at 600 dia. (Which is not the turret center but the tool nose) is 200 m/min. If X axis coordinate value is negative, the CNC uses the absolute value.

Press "absolute coordinate" to change the spindle speed in the status of constant linear speed G96.

Note: 1. use constant linear speed, the spindle must be able to change speed automatically. (such as: Servo Spindle,Inverter Spindle).Max SP-speed is set by parameter.

2. Min SP-speed in condition of G96 can be set by P35 in Speed parameter.

3. Spindle override don't work when processing with constant surface speed.

Example2:

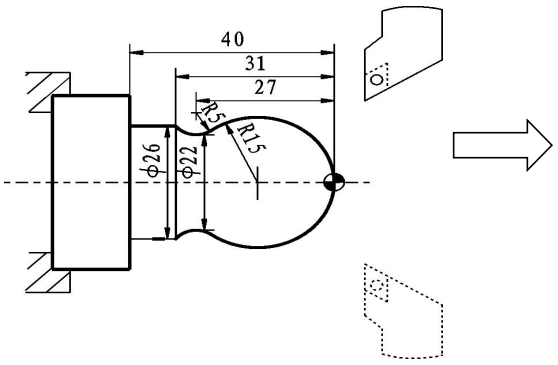


Fig3.17.4 Example2

```
N1 T0102 X40 Z5
N2 M03 S400
N3 G96 S80
N4 G00 X0
N5 G01 Z0 F60
N6 G03 U24 W-24 R15
N7 G02 X26 Z-31 R5
N8 G01 Z-40
N9 X40 Z5
N10 G97 S300
N11 M30
```

3.18 Cutting Feed (G98,G99)

Feedrate of linear interpolation (G01), circular interpolation (G02, G03), etc. are commanded with numbers after the F code.

In cutting feed, the next block is executed so that the feedrate change from the previous block is minimized.

Two modes of specification are available:

1. Feed per minute (G98)

After F, specify the amount of feed of the tool per minute.

2. Feed per revolution (G99)

After F, specify the amount of feed of the tool per spindle revolution.

Format: G98 ; Feed per minute instruction

F_; Feedrate command (mm/min or inch/min)

G99 ; Feed per revolution instruction

F_ ; Feedrate command (mm/rev or inch/rev)

Cutting feed is controlled so that the tangential feedrate is always set at a specified feedrate.

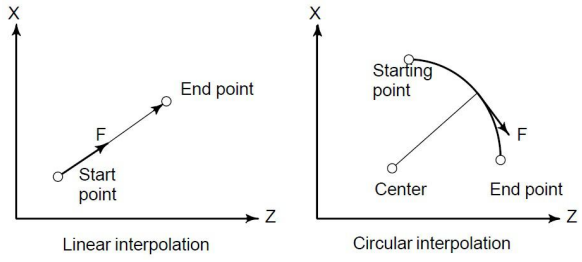


Fig3.18.1 Tangential feedrate (F)

After specifying G98 (in the feed per minute mode), the amount of feed of the tool per minute is to be directly specified by setting a number after F. G98 is a modal code. Once a G98 is specified, it is valid until G99 (feed per revolution) is specified. At power－on, the feed per revolution mode is set.

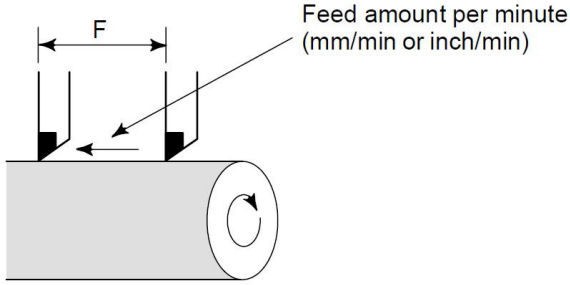


Fig3.18.2 Feed Per Minute

After specifying G99 (in the feed per revolution mode), the amount of feed of the tool per spindle revolution is to be directly specified by setting a number after F. G99 is a modal code. Once a G99 is specified, it is valid until G98 (feed per minute) is specified.

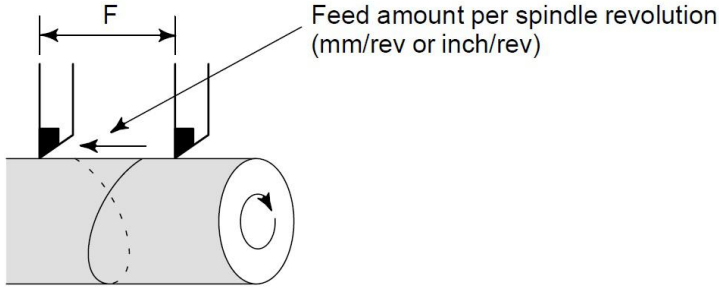


Fig3.18.2 Feed Per Revolution

Note: 1. When using G99 instruction, Spindle must configured with encoder,otherwise system will stay on.

2.When the speed of the spindle is low, feedrate fluctuation may occur. The slower the spindle rotates, the more frequently feedrate fluctuation occurs.

3.19 Pole Coordinate Interpolation (G15/G16)

Polar coordinate interpolation is a function that exercises contour control in converting a command programmed in a Cartesian coordinate system to the movement of a linear axis (movement of a tool) and the movement of a rotary axis (rotation of a workpiece). This method is useful in cutting a front surface and grinding a cam shaft on a lathe.

Polar coordinates input directive allows radius and angle in polar coordinates, the angle of the positive Z direction is counterclockwise turned, while the negative direction is a clockwise turn. Radius with absolute command value instruction (Z), the angle with absolute command (X).

G16 IP_ ; Starts polar coordinate interpolation mode (enables polar coordinate interpolation)

... } Specify linear or circular interpolation using coordinates
... } in a Cartesian coordinate system consisting of a linear
... } axis and rotary axis (virtual axis).

G15 ; Polar coordinate interpolation mode is cancelled (for not performing polar coordinate interpolation)

Explanations:

1. Z specify zero as the origin of the polar coordinate system from the point of measuring the radius of the workpiece coordinate system.

2. IP_: Plane selection & home of polar coordinate system

Z_ : Radius of polar coordinate system

X_ : Angle of polar coordinate system

3. Z set to zero as the origin of the polar coordinate system workpiece coordinate system:

By absolute programming instructions a specified radius (distance between the zero point and programming). Zero set workpiece coordinate system for the origin of the polar coordinate system

G16 starts the polar coordinate interpolation mode and selects a polar coordinate interpolation plane(Fig3.19.1).Polar coordinate interpolation is performed on this plane.

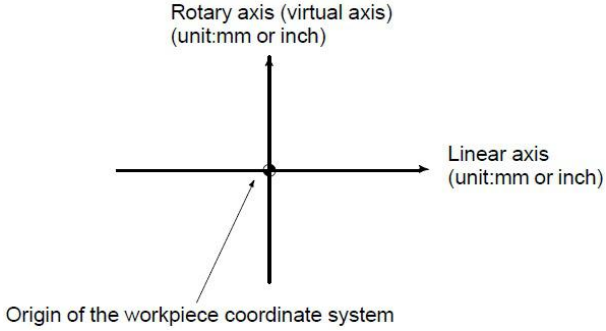


Fig3.19.1 Polar Coordinate Interpolation Plane

When the power is turned on or the system is reset, polar coordinate interpolation is canceled (G15).

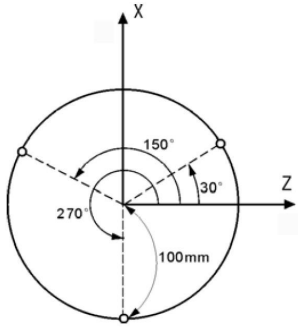


Fig3.19.2 Example of Polar Coordiante System

Example: N1 G16 X0 Z0 ; setup polar coordinate

N2 G01 X30.0 Z100.0 F200.0

N3 X150.0

N4 X270.0

N5 G15 ; cancel polar coordiante system

N6 M02

3.20 Absolute and Incremental Programming (G990,G991)

There are two ways to command travels of the tool; the absolute command, and the incremental command. In the absolute command, coordinate value of the end position is programmed; in the incremental command, move distance of the position itself is programmed. G90 and G91 are used to command absolute or incremental command, respectively.

Absolute programming or incremental programming is used depending on the command used. See following tables.

G code System	A	B or C
Command method	Address word	G990,G991

G code system A

	Absolute command	**Increment command**
X axis move command	X	U
Z axis move command	Z	W
C axis move command	C	V

G code system B or C

Absolute command	**G990 IP_ ;**
Incremental command	**G991 IP_ ;**

Example: Tool movement from point P to point Q (diameter programming is used for the X-axis)

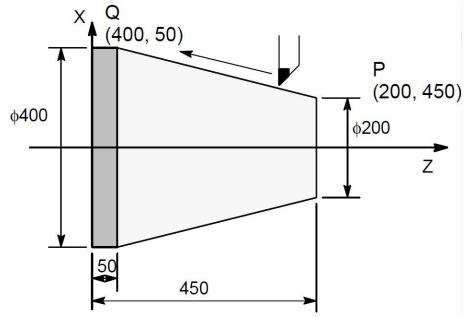


Fig3.20.1 Example of Absolute and Incremental Programming

	G code system A	G code system B or C
Absolute command	X400.0 Z50.0	G90 X400.0 Z50.0
Incremental command	U200.0 W-400.0	G91 X200.0 Z-400.0

Note: 1. Absolute and incremental commands can be used together in a block. In the above example, the following command can be specified : X400.0 W–400.0 ;

2. When both X and U or W and Z are used together in a block, the one specified later is valid.

3. Incremental commands cannot be used when names of the axes are A and B during G code system A is selected.

3.21 Inch/Metric Conversion (G20/G21)

Either inch or metric input can be selected by G code.

Format: G20 ; Inch input
G21 ; mm input

This G code must be specified in an independent block before setting the coordinate system at the beginning of the program. After the G code for inch/metric conversion is specified, the unit of input data is switched to the least inch or metric input increment of increment system IS－B or IS－C. The unit of data input for degrees remains unchanged.

The unit systems for the following values are changed after inch/metric conversion:

- **Feedrate commanded by F code**
- **Positional command**
- **Work zero point offset value**
- **Tool compensation value**
- **Unit of scale for manual pulse generator**
- **Movement distance in incremental feed**
- **Some parameters**

When the power is turned on, the G code is the same as that held before the power was turned off.

Note:

1. G20 and G21 must not be switched during a program.

2. Movement from the intermediate point is the same as that for manual reference position return. The direction in which the tool moves from the intermediate point is the same as the reference position return direction.

3. When the least input increment and the least command increment systems are different, the maximum error is half of the least command increment. This error is not accumulated.

4. The inch and metric input can also be switched using setting of data setting.

3.22 Dwell (G04)

By specifying a dwell, the execution of the next block is delayed by the specified time.

Format: G04 P_ ; or G04 X_; or G04 U_;

P_: Specify a time (decimal point not permitted) , unit: ms (millisecond)

X_: Specify a time (decimal point permitted) , unit: s (second)

U_: Specify a time (decimal point permitted) , unit: s (second)

Example 1:

G04 X1; delay 1s.

G04 P1000; delay 1s.

G04 U1; delay 1s.

Special application:G04 can be accurate stop instruction, such as processing corner kinds of workpiece, it appears over cutting sometimes, if use G04 instruction around the corner, it will clear the over cutting.

Example 2:

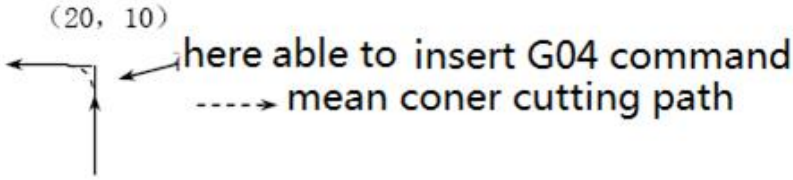


Fig3.22.1 Application of Dwell(G04)

Program: ……

N150 G01 X20 Z10 F100;

N160 G04 P150;（Clear the over cutting）

N170 G01 W-10;

……

Note: Set P21 in User parameter, also by setting intervals between blocks of G01/G02/G03 to clear the over cutting.

3.23 Positioning/Continous Path Processing(G60/G64)

According to process requirements, user can specify the connection way between program block by G60/G64.

Format: G60 ; Acturate Positioning Processing

G64 ; Continous Path Processing

Both G60 and G64 are modal instructions.

3.24 Workpiece Position and Move Command

In tool nose radius compensation, the position of the workpiece with respect to the tool must be specified.G40, G41, and, G42 are modal.

G code	Workpiece Position	Tool Path
G40	(Cancel)	Moving along the programmed path
G41	Right side	Moving on the left side the programmed path
G42	Left side	Moving on the right side the programmed path

The tool is offset to the opposite side of the workpiece.

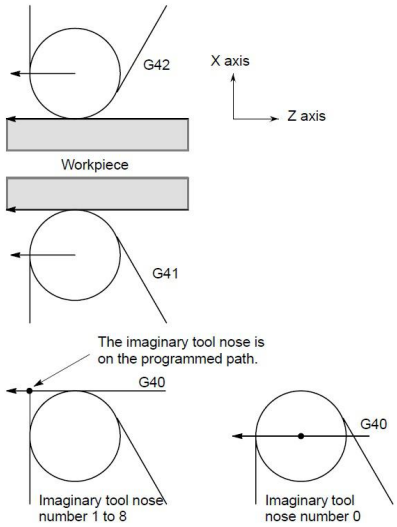


Fig2.24.1 a Workpiece Position of G41/G42

The workpiece position can be changed by setting the coordinate system as shown below.

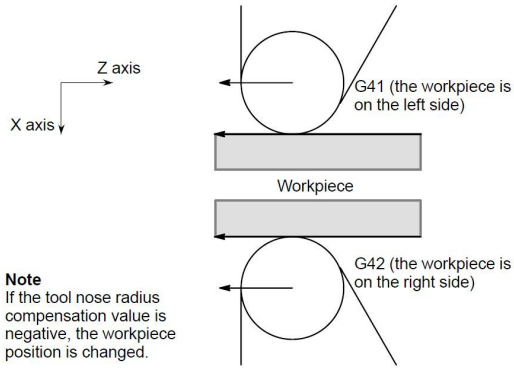


Fig3.24.2 b Workpiece Position of G41/G42

When the tool is moving, the tool nose maintains contact with the workpiece.

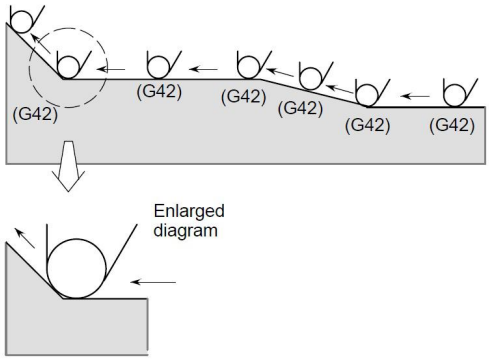


Fig3.24.3 Tool movement when the workpiece position does not change

The workpiece position against the toll changes at the corner of the programmed path as shown in the following figure.

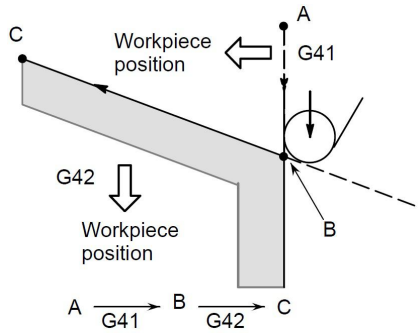


Fig3.24.4 Tool movement when the workpiece position changes

Although the workpiece does not exist on the right side of the programmed path in the above case. the existence of the workpiece is assumed in the movement from A to B. The workpiece position must not be changed in the block next to the start－up block. In the above example, if the block specifying motion from A to B were the start－up block, the tool path would not be the same as the one shown.

The block in which the mode changes to G41 or G42 from G40 is called the start－up block.

G40 _ ;

G41 _ ; (Start-up block)

Transient tool movements for offset are performed in the start–up block. In the block after the start-up block, the tool nose center is positioned Vertically to the programmed path of that block at the start position.

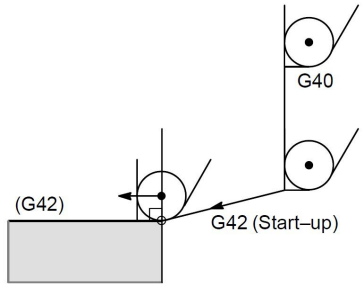


Fig3.24.5 Start-Up

The block in which the mode changes to G40 from G41 or G42 is called the offset cancel block.

G41 _ ;

G40 _ ; (Offset cancel block)

The tool nose center moves to a position vertical to the programmed path in the block before the cancel block. The tool is positioned at the end position in the offset cancel block (G40) as shown below.

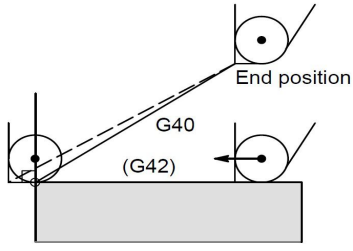


Fig3.24.6 Offset Cancel

When is specified again in G41/G42 mode , the tool nose center is positioned vertical to the programmed path of the preceding block at the end position of the preceding block.

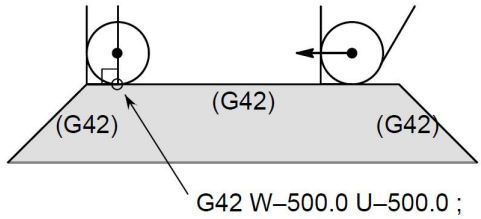


Fig3.24.7 Specification of G41/G42 in G41/G42 mode

In the block that first specifies G41/G42, the above positioning of the tool nose center is not performed.

When you wish to retract the tool in the direction specified by X(U) and Z(W) cancelling the tool nose radius compensation at the end of machining the first block in the figure below, specify the following :

G40 X(U) _ Z(W) _ I _ K _ ;

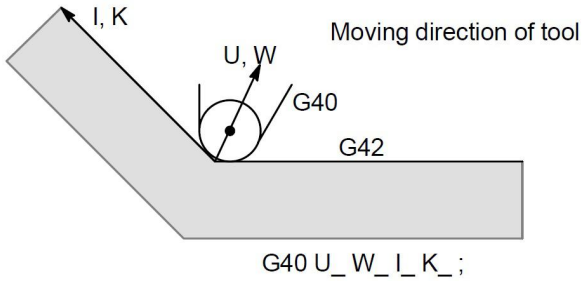


Fig3.24.8 Tool movement when the moving direction of the tool in a block

The workpiece position specified by addresses I and K is the same as that in the preceding block. If I and/or K is specified with G40 in the cancel mode, the I and/or K is ignored.

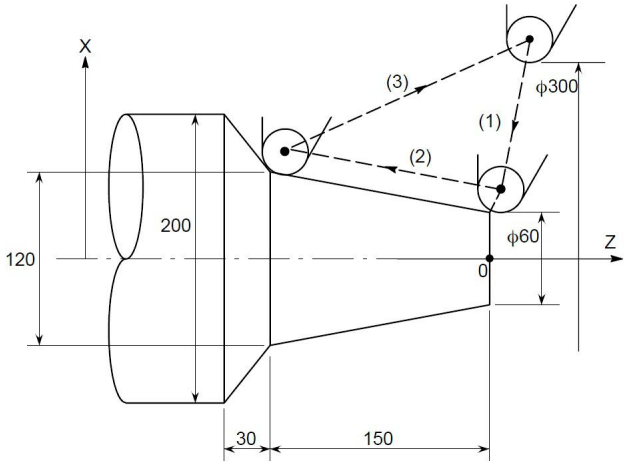
G40 X_ Z_ I_ K_ ;	Tool nose radius compensation
G40 G02 X_ Z_ I_ K_ ;	Circular interpolation

G40 G01 X_ Z_ ;

G40 G01 X_ Z_ I_ K_ ; Offset cancel mode (I and k are ineffective.)

The numeral s followed I and K should always be specified as radius values.

Example



(G40 mode)
1.G42 G00 X60.0 ;
2.G01 X120.0 W–150.0 F10 ;
3.G40 G00 X300.0 W150.0 I40.0 K–30.0 ;

Fig3.24.9 Example of Tool Nose Radius Compensation

Tool nose radius compensation with G90 (outer diameter/internal diameter cutting cycle) or G94 (end face turning cycle) is as follows, :

1. Motion for imaginary tool nose numbers

For each path in the cycle, the tool nose center path is generally parallel to the programmed path.

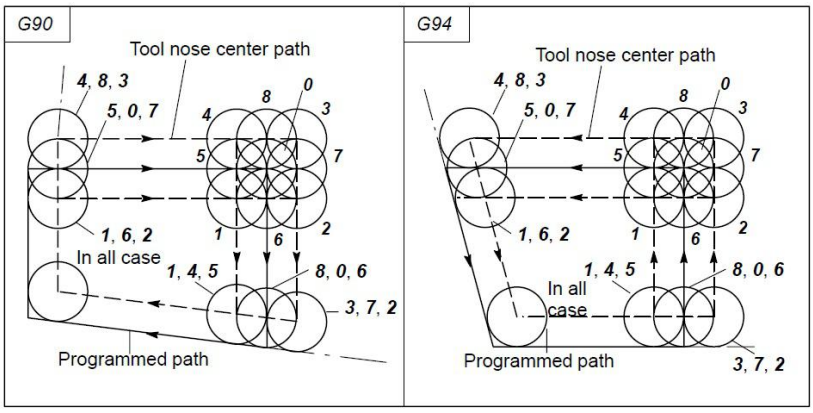


Fig3.24.10 Motion for imaginary tool nose numbers

2. Direction of the offset

The offset direction is indicated in the figure below regardless of the G41/G42 mode.

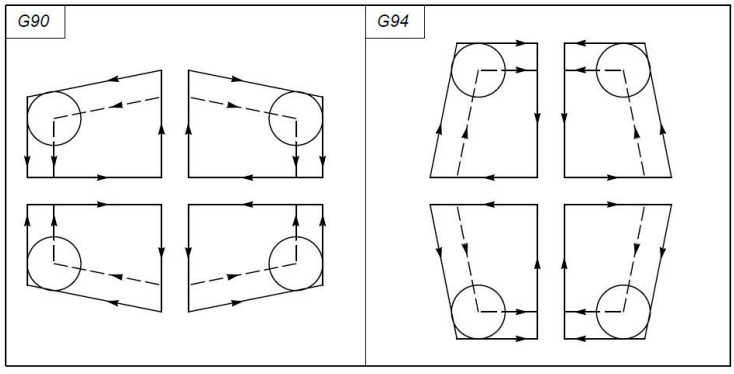
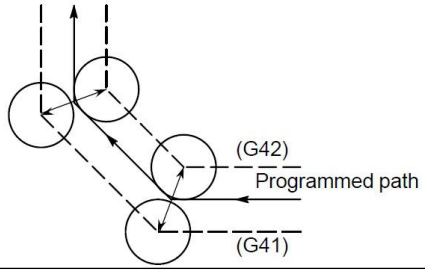
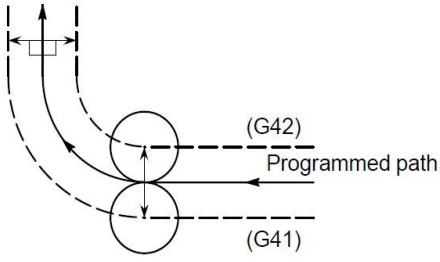


Fig3.24.11 Direction of the offset

Movement after compensation is shown below.

Tool nose radius compensation when chamfering is performed	Tool nose radius compensation when a corner arc is inserted
	

3.25 Details of Tool Nose Radius Compensatation (G40/G41/G42)

The tool nose radius center offset vector is a two dimensional vector equal to the offset value specified in a T code, and the is calculated in the CNC.

Its dimension changes block by block according to tool movement. This offset vector (simply called vector herein after) is internally crated by the control unit as required for proper offsetting and to calculate a tool path with exact offset (by tool nose radius) from the programmed path.

This vector is deleted by resetting.

The vector always accompanies the tool as the tool advances.

Proper understanding of vector is essential to accurate programming.

Read the description below on how vectors are created carefully.

G40, G41 or G42 is used to delete or generate vectors.

These codes are used together with G00, G02, or G32 to specify a mode for tool motion (Offsetting).

G code	Function	Workpiece Position
G40	Tool nose radius compensation cancel	Neither
G41	Left offset along tool path	Left
G42	Right offset along tool path	Right

G41 and G42 specify an off mode, while G40 specifies cancellation of the offset.

The system enters the cancel mode immediately after the power is turned on, when the RESET button on the MDI is pushed or a program is forced to end by executing M02 or M30. (the system may not enter the cancel mode depending on the machine tool.) In the cancel mode, the vector is set to zero, and the path of the center of tool nose coincides with the programmed, path. A program must end in cancel mode. If it ends in the offset mode, the tool cannot be positioned at the end point, and the tool stops at a location the vector length away from the end point.

Note: 1. G40/G41/G42 are modal codes, can cancel each other.

2. G41/G42 without parameters, the compensation number (on behalf of the tool tip radius compensation corresponding values) specified by the T code. Its tip arc offset number and tool offset number corresponding compensation.Establish and canceled .

3. Tool nose radius compensation with G00 or G01 instruction only, not the G02 or G03.Nose radius compensation tool compensation interface "radius compensation", the definition of the turning radius; imaginary tool nose number defines the direction of the tip.

The imaginary tool nose tip number defines the positional relationship between the tool and cutter tip arc center point, which is from 0-9 ten directions, as shown above.

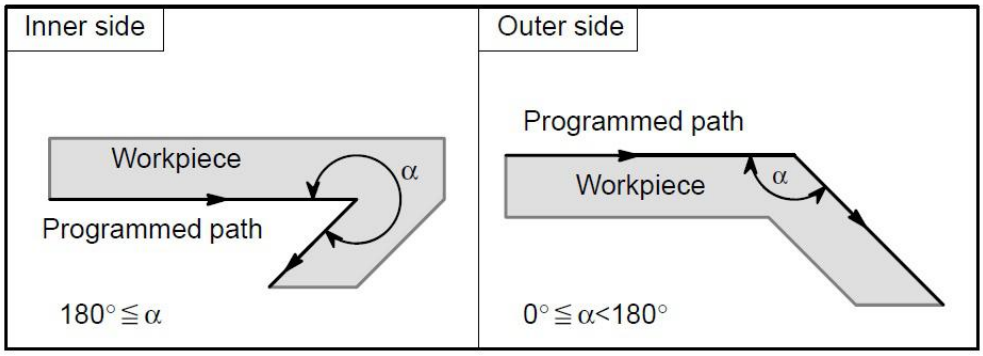
Note: For the definition of the edge position code "Posit tip" See Chapter 6.8 .

3.26 Tool Nose Radius Compensation of Offset C

C means the system calculates the tool trajectory of radius compensation according to the last program line and the next program line.

1) Inner side and outer side

When an angle of intersection created by tool paths specified with move commands for two blocks is over 180º, it is referred to as “inner side.” When the angle is between 0º and 180º , it is referred to as “outer side.”

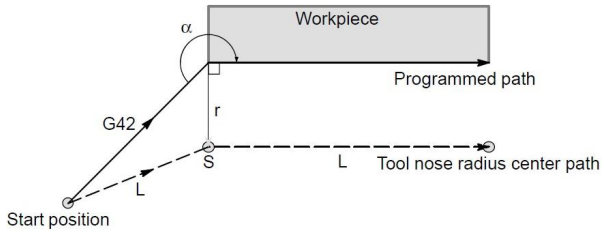


2) Tool Movement in Start-up

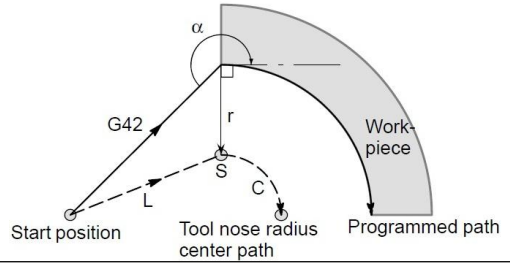
When the offset cancel mode is changed to offset mode, the tool moves as illustrated below (start-up):

(a) Tool movement around an inner side of a corner (α≥180º)

Linear ⟶ Linear

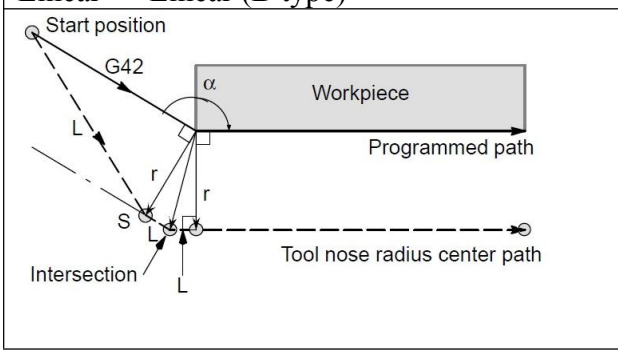


Linear ⟶ Circular

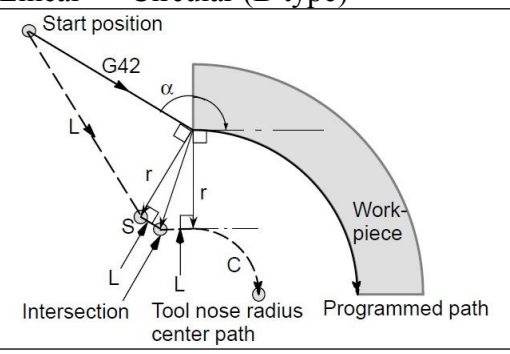


(b) Tool movement around the outside of a corner at obtuse angle (90≤α＜180)

Linear ⟶ Linear (B type)

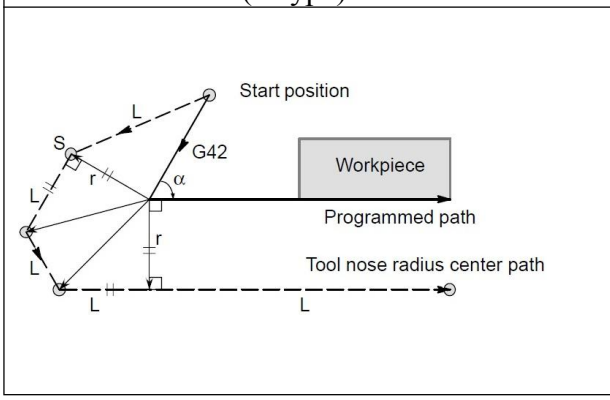


Linear ⟶ Circular (B type)

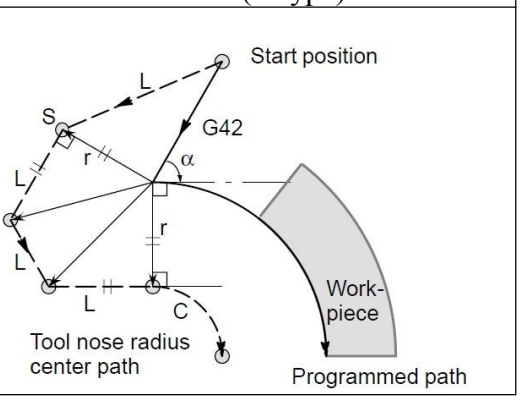


(c)Tool movement around the outside of an acute angle (α＜90º)

Linear ⟶ Linear (B type)



Linear ⟶ Circular (B type)



d)Tool movement around the outside linear→linear at an acute angle less than 1 degree (α ＜1º)

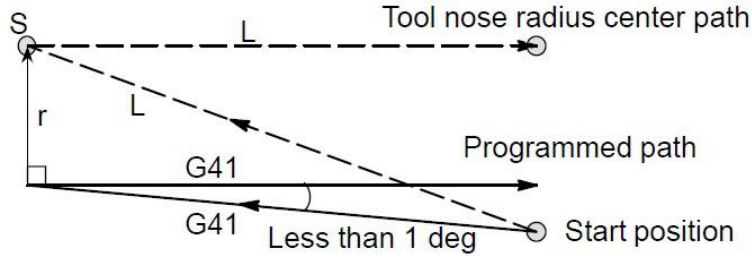
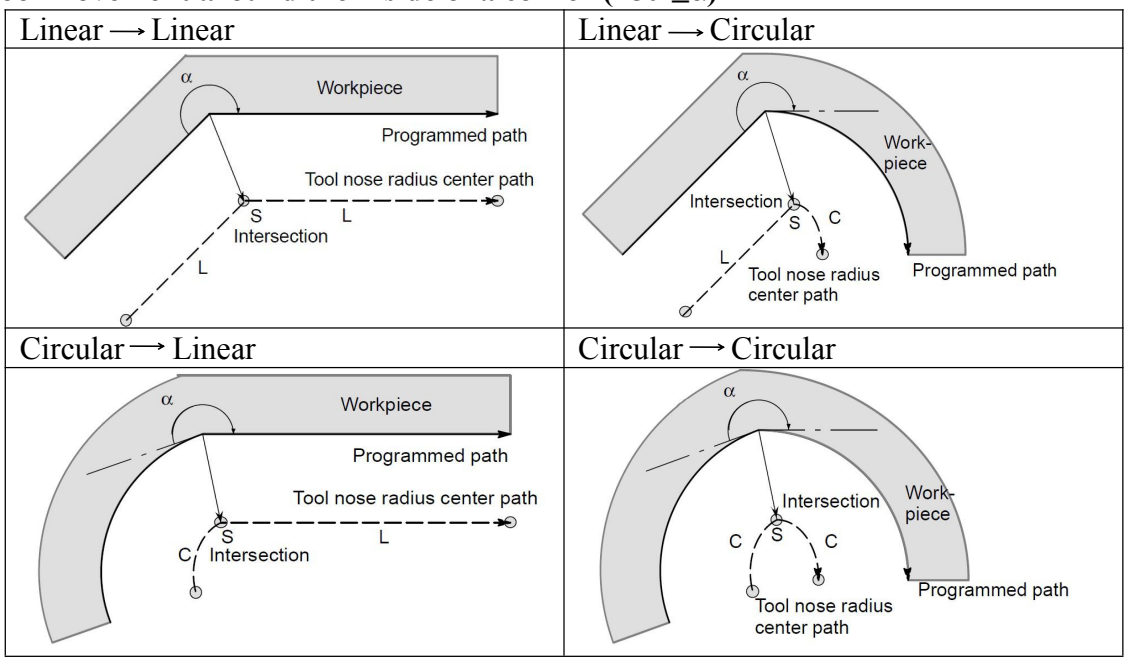


Fig3.26.1 Tool Movement at angle less than 1º

3) Tool Movement in Offset Mode

In the offset mode, the tool moves as illustrated below:

(a)Tool movement around the inside of a corner (180º≤α)



(b)Tool movement around the inside (α<1º) with an abnormally long vector, linear → linear

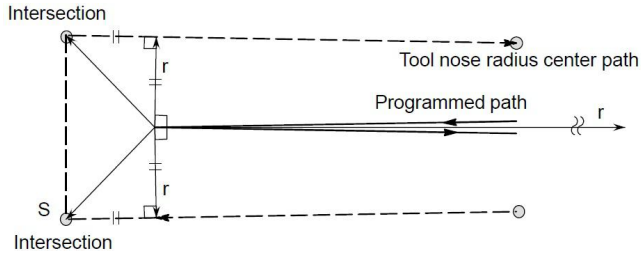
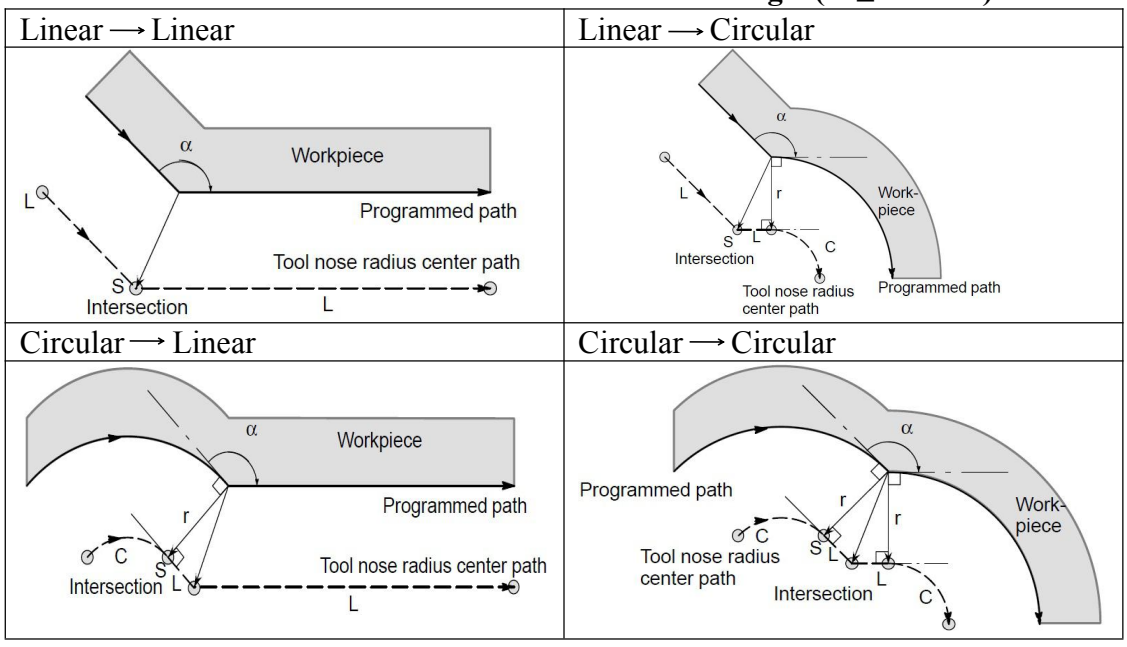


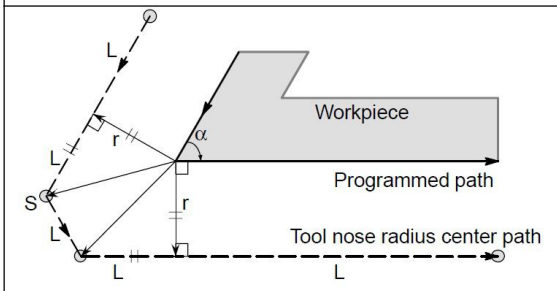
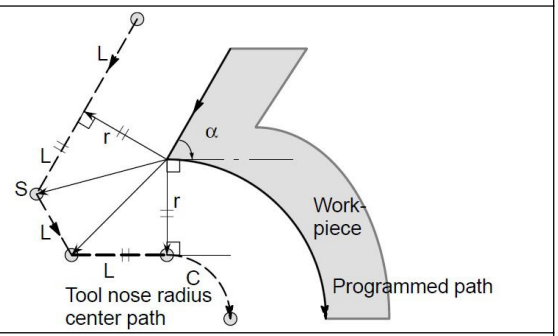
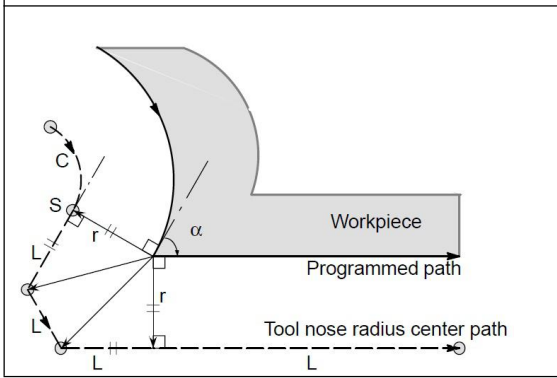
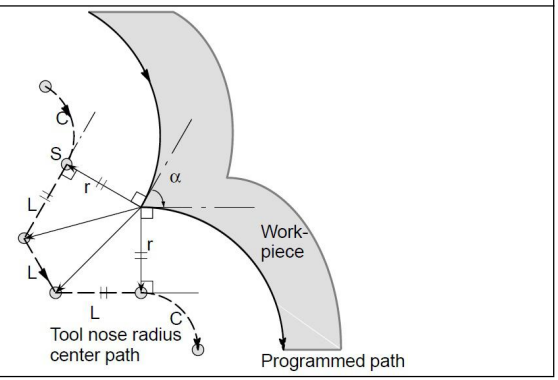
Fig3.26.2 Tool Movement at angle less than 1º

Also in case of arc to straight line, straight line to arc and arc to arc, the reader should infer in the same procedure.

(c)Tool movement around the outside corner at an obtuse angle (90≤α＜180º)

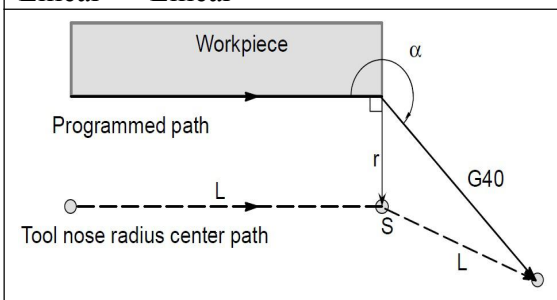
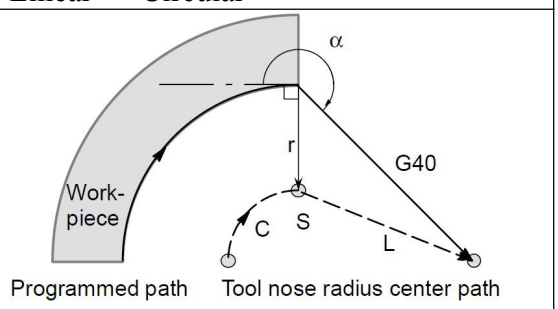


(d)Tool movement around the outside corner at an acute angle (α＜90º)

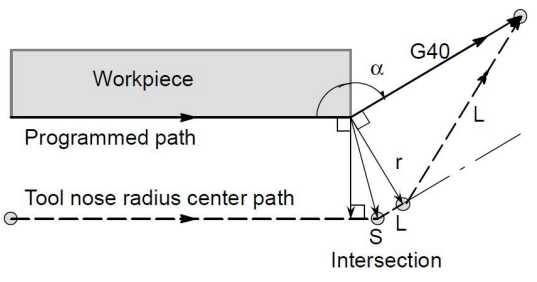
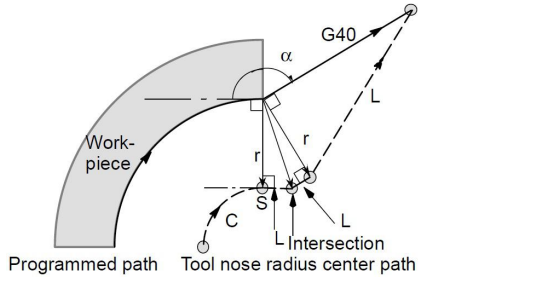
Linear ⟶ Linear	Linear ⟶ Circular
	
Circular ⟶ Linear	**Circular ⟶ Circular**
	

4) Tool Movement in Offset Mode Cancel

(a) Tool movement around an inside corner (180º≤α)

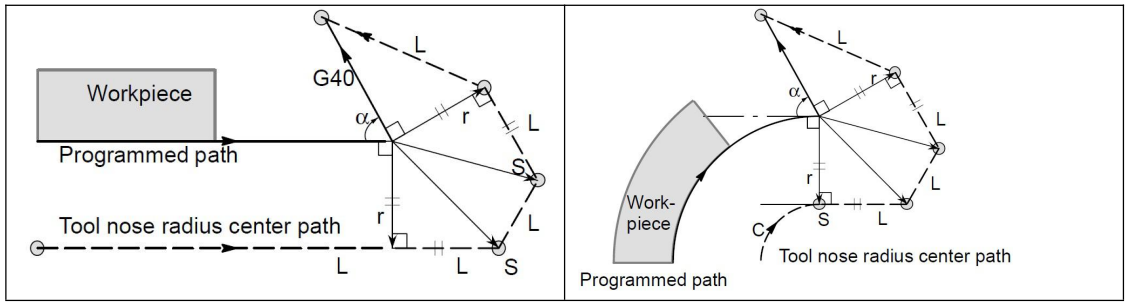
Linear ⟶ Linear	Linear ⟶ Circular
	

(b) Tool Movement around an outside corner at obtuse angle (90≤α＜180º)

Linear ⟶ Linear(B type)	Linear ⟶ Circular(B type)
	

(c) Tool Movement around an outside corner at acute angle (α＜90º)

Linear ⟶ Linear(B type)	Circular ⟶ Linear(B type)



(d) Tool movement around the outside linear→linear at an acute angle less than 1 degree(α＜1º)

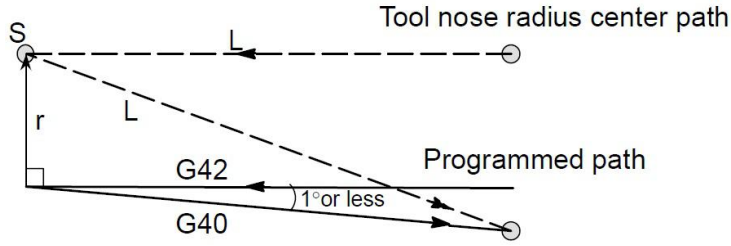


Fig3.26.3 Tool Movement at angle less than 1

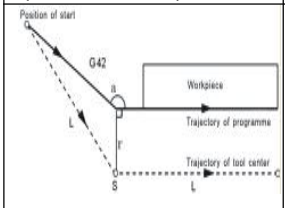
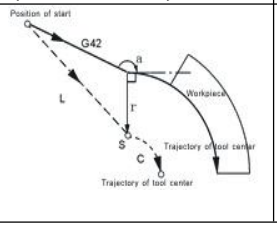
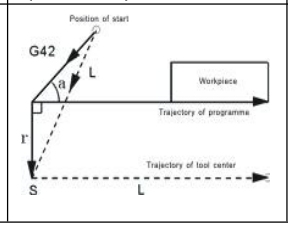
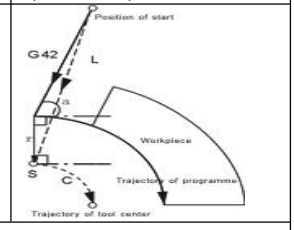
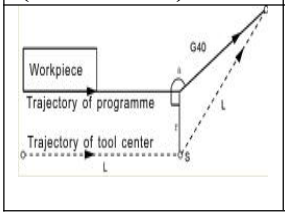
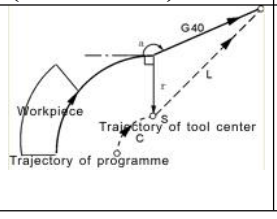
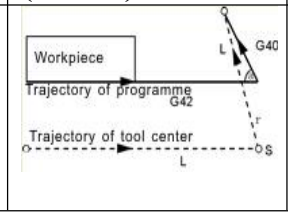
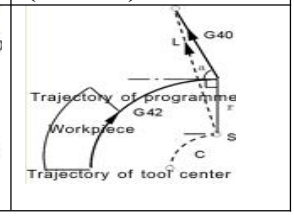
5) Change in the offset direction in the offset mode

The offset direction is decided by G codes (G41 and G42) for tool nose radius and the sign of tool nose radius compensation value as follows.

G code \ Sign of offset value	+	-
G41	Left side offset	Right side offset
G42	Right side offset	Left side offset

The offset direction can be changed in the offset mode. If the offset direction is changed in a block, a vector is generated at the intersection of the tool nose radius center path of that block and the tool nose radius center path of a preceding block. However, the change is not available in the start-up block and the block following it.

6) A Type of Tool Nose Compensation

Tool Movement in Start-up			
(90≤α＜180)	**(90≤α＜180)**	**(α＜90º)**	**(α＜90º)**
			
Tool Movement in Offset Mode Cancel			
(90≤α＜180)	**(90≤α＜180)**	**(α＜90º)**	**(α＜90º)**
			

P41 in Speed is set for A type , B type and other type for Tool Nose Radius Compensation of Offset C.

3.27 Automatical beveling (I) and smoothing(R)

Format:

Automatical Beveling

G01(G00) X(U)_ I_ ;
G01(G00) Z(W)_ } **Automatical Beveling**

G01(G00) Z(W)_ I_;
G01(G00) X(U)_ } **Automatical Beveling**

Automatical Smoothing

G01(G00) X(U)_ R_ ;
G01(G00) Z(W)_ } **Automatical Smoothing**

G01(G00) Z(W)_ R_;
G01(G00) X(U)_ } **Automatical Smoothing**

Note:

1.The address of I and R are specified with radius model. The running distance of this line and the next line must be greater than the length of beveling or radius of smoothing, otherwise the system will decrease the length of beveling or radius of smoothing to minimal running distance of this line and the next line automatically.

2. The two adjacent lines must be 90 degrees.

For example:

```
0 G54 G0 X-50 Y-50 Z20
N1 M03 S500
N2 G01 G42 D01 X0 Y0 F200
N3 G01 Z-5
N4 X100 I4                    ;Beveling4x4
N5 Y40 R6                     ;SmoothingR6
N6 X47 R5                     ;SmoothingR5
N7 Y70 I3                     ;Beveling3x3
N8 X15
N9 X0 Y40
N10 Y0
N11 G0 X-50 Y-50 G40
N12 Z50
N13 M30
```

3.28 3D Space Arc Interpolation G06

When user don't know position of circle center & radius, but know coordinate position of 3 points on arc, now user can use G06 code to processing arc, and direction is decided by middle point between starting point & end point.

Format: G06 X_ Y_ Z_ I_ J_ K_ F_

G06: Modal command

I: Increment Coordiante Value of Middle point relative to starting point in X direction Radius Designation, with direction;

J: Increment Coordiante Value of Middle point relative to starting point in Y direction With direction

K: Increment Coordiante Value of Middle point relative to starting point in Z direction With direction

F: Cutting speed

Note:

1. Middle point is any position point except starting point & end point.

2. System will alarm when three points are at one line.

3. When I,J,K are omitted, default value is I=0, J=0, K=0. But they cannot be omitted all at same time,otherwise system will alarm.

4. The meanings of I,J,K are similar to I,J,K of G02/G03.

5. G06 cannot be used for processing total round.

6. Compute of G06 command is very large, it only can work smoothly on modbus system, at normal system, it would work not smoothly.

Example:

```
G0 X10 Y28 Z10
G06 X30 Y98 Z10 I5 J-6 K-5 F100
      X130 Y198 Z120 I55 J-86 K-65
G0 X0 Z0
M02
```

3.29 Macro program instruction(G65/G66/G67)

3.29.1 Non-Mode Macro Command G65

Format: G65 P_ L_ A_ B_ C_ .....

Non-mode macro command G65 only work at current line , which is different to mode macro command(G66),which always work until macro cancel command(G67)

P_ : Specify name of macro program, E.g: P6000 , name of specified macro program is 6000 .

L_: Set times of call macro program

<A_B_C_... ...> : Argument , which is used for transfer data to macro varible(#**) , Transfering table is as following

Argument	Variable	Argument	Variable	Argument	Variable
A	#0	I	#7	T	#14
B	#1	J	#8	U	#15
C	#2	K	#9	V	#16
D	#3	M	#10	W	#17
E	#4	Q	#11	K	#18
F	#5	R	#12	Y	#19
H	#6	S	#13	Z	#20

Waraning:

1. Macro variables #100-#155&#190-#201 was occupied by system, user cannot use.

2. User cannot use G70,G71,G72,G73,G92,G76 etc loop command on Macro program.

Note: the address G, L, N, Q, P can't be used as user-defined variables.

Example:

Main program:9000	Macro program:8000
G00 X0 Z0 G65 P8000 L1 A5 B6 G0 X0 Z0 M30	N1 #2=#0+#1 N2 IF （#2 EQ 10）GOTO 4 N3 GOO X#2 N4 G00 Z#1 N5 M99 ;Return

3.29.2 Mode Macro Command G66/G67

G66 is mode macro command , G67 is cancel mode macro command

Format: G66 P_ L_ A_ B_ C_ .....

G67

G66 Mode macro command,which always call macro program until macro cancel command(G67)

P_ : Specify name of macro program, E.g: P7000 , name of specified macro program is 7000 .

L_: Set times of call macro program

<A_B_C_... ...> : Argument , which is used for transfer data to macro varible(#**) , the transfering table is same as above table.

Example:

```
Main Program : 4000
G00 X0 Z0
G66 P6000 L2 A5 B6
A8 B1
A9 B10
G67
M30
Macro Program: 6000
N1 #2=#0+#1
N2 IF (#2 EQ 10) GOTO 4
N3 G00 X#2
N4 G00 Z#1
N5 M99              ; Return
```

3.29.3 Macro Program Instruction

3.29.3.1 Input Instruction: WAT

Waiting for the input port X valid or invalid instruction

Format: WAT+ (-) X

Attention: "+" to means the input is effective;

"-" means the input is invalid;

"X" means the input port X00-X55; see the I/O diagnosis;

3.29.3.2 Output Instruction: OUT

Set the output port Y is valid or invalid instruction

Format: OUT +(-)Y

Attention: "+" means the output is effective;

"-" means the output is invalid;

"Y" means the output port Y00-Y31; see the I/O diagnosis;

3.29.3.3 Assignment Instruction: =

Explanation: used for assignment of a variable

Eg.: #251=890.34 #450=#123

And also it could be mathematical expression , eg.: #440=#234+#470

3.29.3.4 Unconditional Jump: GOTO n

"GOTO n" is the command that for jump to the program line that is specified by sequence number (N**) unconditionally. n is the sequence number.

E.g.: GOTO 5 ; // Jump to N5 program line.

Note: when specified program line , n , is beyond sequence number of N1-N99999, cnc system will hint error.

n , program line,could be macro variable (#**)

E.g.: GOTO #100

3.29.3.5 Conditional Jump

1) IF (Conditional express) GOTO n

If condition is met, execute GOTO n ,jump to N** program line; if the condition isnot met, execute the next segment.

Example: N1 IF(#200 EQ 1) GOTO 20

N10 G00 X0

N20 G00 Z0

Explanation: If #200 is equal to 1, system will execute GOTO 20 , jump to N20 , and execute "G00 Z0", if #200 isn't equal to 1, system don't execute operation of "GOTO 20" ,and will execute next segments , "G00 X0",and then execute "G00 Z0".

2) IF (Conditional express) THEN <A Expression>

<B opertional segment>

If condition is met, system execute A expression , and then execute B operational segment ; if condition is not meet, execute the next segment , B operation.

Example: #101=0

N1 IF(#100 EQ 1) THEN #101=1

N2 IF(#101 EQ 1) GOTO 4

N3 G00 X100

N4 G00 Z100

Explanation: If #100 is equal to 1, system will execute “#100=1”, and then judge #101 is equal to 1 , jump to N4 & “execute G00 Z100” ; if #100 isn’t equal to 1, system will judge #101 also isn’t equal to 1 directly , and execute “G00 X100” & “G00 Z100”.

NOTE: 1.<A expression> normally is assignment statement.

2. <A expression> after THEN must exist, otherwise system will hint grammatical errors.

Prolongation:

3) IF(conditional express)

<A operational command>

ELSE

<B operational command>

ENDIF

4) IF(conditional express)

<A operational command>

ELIF

<B operational command>

ENDIF

3.29.3.6 Loop Command

Format: (Conditions Initialization)

WHILE (conditional expression) DO n

<A operational segments>

[Alter condition of loop]

END n

<B operational segments>

When conditions are met during WHILE cycle command, execute the operational segments between DO n and END n . Otherwise,when condition isnot met, jump to the program line after END n ,also execute B operational segments.

We can nest for loops by placing one loop within another.

Note: 1.There must have operational codes that are for change condition at operational segments ,which is

between Do n & END n. Otherwise system will enter endless loop.

2.Nesting of macro program loop statements of SZGH CNC system is 3 pcs of loops at most . Also n only could be 1 , 2 , 3 .

3.n of "DO n" & "END n" must keep same.

Example: #100=2 #150=5 #200=25

```
WHILE （#100 LT 3）DO 1
 G00 X100
 WHILE （#150 EQ 5）DO 2
 G00 Y100
 WHILE （#200 GE 20）DO 3
 G00 Z100
 #200=#200-2
 END 3
 #150=#150-1
 END 2
 #100=#100-1
 END 1
```

3.29.4 Operators' meaning

Operator	Sign	Ex.	Operator	Sign	Ex.	Operator	Sign	Ex.
EQ	=	equal	GT	>	greater	LT	<	Less
NE	≠	unequal	GE	≥	G&E	LE	≤	L&E

3.29.5 Arithmetic & Logic Operation

Table:

Function	Format	Attention
Definition	#i = #j	
Addition Subtraction Multiplication Division	#i = #j + #k ; #i = #j - #k ; #i = #j * #k ; #i = #j / #k ;	
Sin Asin Cos Acos Tan Atan	#i = SIN(#j) ; #i = ASIN(#j); #i = COS(#j) ; #i = ACOS(#j); #i = TAN(#j); #i = ATAN(#j);	90.5 degrees means 90 degrees & 30 points
Square root Absolute value Rounding off Round down Round up	#i = SQRT(#j); #i = ABS(#j) ; #i= ROUND(#j); #i = FIX(#j); #i = FUP(#j);	

Natural logarithm Exponential function	#i = LN(#j); #i = EXP(#j);	
Or Exclusive or And	#i = #j OR #k ; #i = #j XOR #k ; #i = #j AND #k ;	Executing with binary system

3.29.6 Local Variable

#0--#20 : local variables only can be used to store data in macro program, such as a result of operation, when power is off, the local variables are initialized to the empty. The argument assignment to the local variable when calling the macro program.

3.29.7 Global Variable

#21--#600 : Their meanings are the same in different macro program.

When power is off, the variable #21--#100 is initialized to zero, the variable #101--#600 data is saved not to loss even if the power is off.

3.29.8 System Variable

#1000-- : the system variables are used to change various data when reading the running CNC. For example, the current position and the compensation of tool.

Special Attention: macro variables #100--#155 and #190--#202 have been used by the system, users can not use.

3.29.9 System Parameter Variable

#1001--#1099 : Value of X-axis length compensation for T1--T99(Unit: um)

#1101--#1199 : Value of D1 radius compensation for T1--T99(Unit: um)

#1201--#1299 : Value of Y(C)-axis length compensation for T1--T99(Unit: um)

#1301--#1399 : Value of D2 radius compensation for T1--T99(Unit: um)

#1401--#1499 : Value of Z-axis length compensation for T1--T99(Unit: um)

#1501--#1599 : Value of D3 radius compensation for T1--T99(Unit: um)

#1601--#1699 : Value of A-axis length compensation for T1--T99(Unit: um)

#1701--#1799 : Value of D4 radius compensation for T1--T99(Unit: um)

3.29.10 I/O variable

#1800: X00-X07 (D0-D7) ; input resistor

#1801: X08-X15 (D0-D7) ; input resistor

#1802: X16-X23 (D0-D7) ; input resistor

#1802: X16-X23 (D0-D7) ; input resistor

#1803: X24-X31 (D0-D70 ; input resistor

#1804: X32-X39 (D0-D7) ; input resistor

#1805: X40-X47 (D0-D7) ; input resistor

#1806: X60-X67 (D0-D7) ; input resistor

#1807: X74-X81 (D0-D7) ; Alarm of driver/Spindle

#1808: Y00-Y15 (D0-D15) ; output resistor

#1809: Y16-Y31 (D0-D15) ; output resistor

#1810: Y32-Y47 (D0-D15) ; output resistor

Waraning:

1. Macro variables #100-#155&#190-#201 was occupied by system, user cannot use.

2. User cannot use G70,G71,G72,G73,G92,G76 etc loop command on Macro program.

Note: the address G, L, N, Q, P can't be used as user-defined variables.

Chapter 4 M INSTRCUTIONS

4.1 M Function (Auxiliary Function)

M instruction consists of instruction address M and its following 1~2 bit digits, used for controlling the flow of executed program or outputting M instructions to PLC.

M □□

└→ Instruction value(the leading zero can be omitted)

└→ Instruction address

When address M followed by a number is specified, a code signal and strobe signal are transmitted. These signals are used for turning on/off the power to the machine.

In general, only one M code is valid in a block but up to three M codes can be specified in a block (although some machines may not allow that). The correspondence between M codes and functions is up to the machine tool builder.

All M codes are processed in the machine except for M97 M98, M99,M codes for calling a subprogram , and M codes for calling a custom macro. Refer to the appropriate manual issued by the machine tool builder.

The following M codes have special meanings.

M00, M01, M02, M30, M97, M98, M99 must not be specified together with another M code.

Some M codes other than M00, M01, M02, M30, M97, M98 and M99 cannot be specified together with other M codes; each of those M codes must be specified in a single block.

4.1.1 Program Stop(M00)

Automatic operation is stopped after a block containing M00 is executed.

When the program is stopped, all existing modal information remains unchanged. The automatic operation can be restarted by actuating the cycle operation. This differs with the machine tool builder.

4.1.2 Optional Stop (M01)

Similarly to M00, automatic operation is stopped after a block containing M01 is executed. This code is only effective and program stop when input point M22(PIN5 of CN10 Plug) is valid

4.1.3 End of Program (M02,M30)

This indicates the end of the main program.Automatic operation is stopped and the CNC unit is reset.

4.1.4 Cycle of Program (M20)

Run program cycle , cycle time is set by P18 in User parameter.

4.1.5 Account of Workpiece(M87)

Number of workpieces will add one automatically as P10=0 in Other parameter.

4.1.6 Uncondtional Jump (M97)

M97 P_ , jump to specfied block number that specified by P.P4 said entrance line number with 4 field numbers specified program transfer main program.

Example: M97 P0120 , when excutive this code, CNC will jump to “N0120” block and run.

4.2 Subprogram Configuration

There are two program types, main program and subprogram. Normally, the CNC operates

according to the main program. However, when a command calling a subprogram is encountered in the main program, control is passed to the subprogram. When a command specifying a return to the main program is encountered in a subprogram, control is returned to the main program.

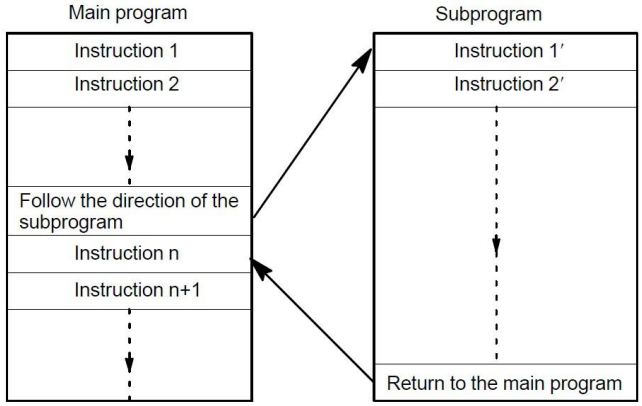


Fig4.2.1 Main program and subprogram

If a program contains a fixed sequence or frequently repeated pattern, such a sequence or pattern can be stored as a subprogram in memory to simplify the program.

A subprogram can be called from the main program.

A called subprogram can also call another subprogram.

4.2.1 Calling of Subprogram (M98)

This code is used to call a subprogram.The code and strobe signals are not sent.

M98 P_ L_ ;

P_: specify address & name of subprogram. Eg.: Psub/1390; sub is a folder.

Subprogram can be hidden files that don't display in program district,First character of these program must be a "HIDEFILE".

Example: "HIDEFILE01", the subprogram in the program area is not displayed, user can use these type commands to call subprogram.

M98 PHIDEFILE01

or **M98 P*01**

If user want to call subprogram in USB-disk, Format is " P[_" or " P]_".

Example: **M98 P[A1234** ; Call A1234 subprogram in USB-disk;

L_: number of times the subprogram is called repeatedly.When no repetition data is specified, the subprogram is called just once.

M98 instruction can be omitted. **Format: PP_** .

Example: **PP[FFDE ;** call "FFDE" subprogram in USB-disk;

Note: 1. There must has a blank before "L_" in this system;

2. Subprogram must be an independent program.

4.2.2 End of Subprogram (M99)

This code indicates the end of a subprogram.M99 execution returns control to the main program.No code or strobe signal is sent.

1) M99 in main program is same to M02;

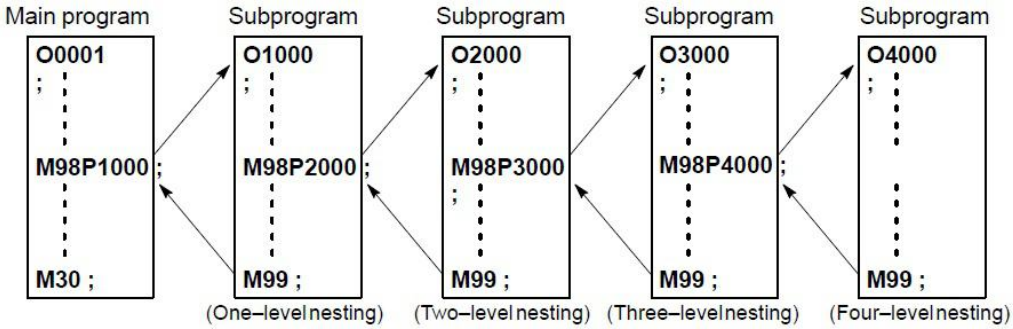
2) M99 with P in main programis same to M97 ;

3) M99 in subprogram return to next block of M98;

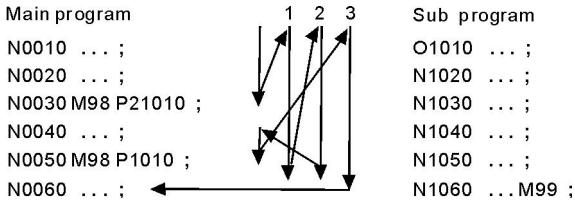
4) M99 with P in the subroutine return to specified block in main program.

When the main program calls a subprogram, it is regarded as a one–level subprogram call.

Thus, subprogram calls can be nested up to four levels as shown below.



Eg.:Execution sequence of subprograms called from a main program.A subprogram can call another subprogram in the same way as a main program calls a subprogram.



A single call command can repeatedly call a subprogram up to 9999 times.

4.3 Standard PLC M Command List

4.3.1 M Output Command List

Function	Code	Introduction	Statement
Spindle	M03	Spindle on CW	Functions interlocked and states reserved
	M04	Spindle on CCW	
	M05	Spindle Stop	
Coolant	M08	Coolant ON	Functions interlocked and states reserved
	M09	Coolnat OFF	
Chuck	M10	Chuck Clamping	Functions interlocked and states reserved
	M11	Chuck Unclamping	
Tailstock	M79	Tailstock Forward	Functions interlocked and states reserved
	M78	Tailstock Backward	
Lubricate	M32	Lubrication ON	Functions interlocked and states reserved
	M33	Lubrication OFF	
Huff	M59	Huff ON	Functions interlocked and states reserved
	M58	Huff OFF	
1	M61	User-define Output 1 ON	Functions interlocked and states reserved
	M60	User-define Output 1 Off	
2	M63	User-define Output 2 ON	Functions interlocked and states reserved
	M62	User-define Output 2 Off	
3	M65	User-define Output 3 ON	Functions interlocked and states reserved
	M64	User-define Output 3 Off	
4	M67	User-define Output 4 ON	Functions interlocked and states reserved
	M66	User-define Output 4 Off	
5	M69	User-define Output 5 ON	Functions interlocked and states reserved
	M68	User-define Output 5 Off	
6	M71	User-define Output 6 ON	Functions interlocked and states reserved
	M70	User-define Output 6 Off	

7	M73	User-define Output 7 ON	Functions interlocked and states reserved
	M72	User-define Output 7 Off	
8	M75	User-define Output 8 ON	Functions interlocked and states reserved
	M74	User-define Output 8 Off	

M75/M74: When CNC system is configurated with C axis , which is used for switch the control mode (Position control Mode & Analog Speed Mode) of servo spindle. When M75 is valid, which is set servo spindle to position control mode,when M03/M04 is valid, turn off M75.

M71/M70: When P20 in Other parameter set to 1, M11 output M71.

M73/M72: When P21 in Other parameter set to 1, M78 output M73.

Note: All M output commands, output 0V effective level.

4.3.1.1 Spindle Control (M03/M04/M05)

M03 is for control CW of spindle , M04 is for control CCW of spindle , M05 is for stop spindle.

Input Point

M03	PIN19_CN3 Plug	CW of Spindle
M04	PIN7_CN3 Plug	CCW of Spindle
M05	PIN20_CN3 Plug	STOP of Spindle

Wiring Diagram

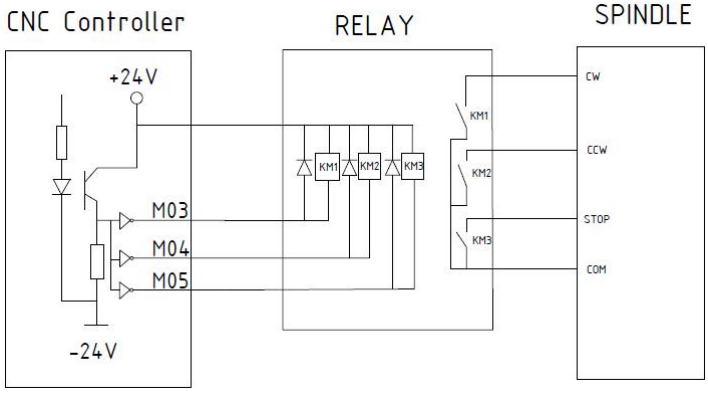


Fig4.3.1 Wiring Diagram for Spindle

According to this wiring diagram, it will consist a control circuit with +24V when system output M03/M04/M05 signal, coil of relay works and NO type switches will be ON , and control related function of spindle.

Note:1. Effective level of all output points is 0V.

2.When the relays and others inductance load, must connected with the reverse diode to absorb the reverse current so as not to damage the system, if use the electromagnetic contactor, then plus resistive and capacitive spark circuit.

Parameter Set

System output dual analog for spindle for control speed of two spindles,2nd analog output 1st analog,which set by D11_P9 in Other parameter.

In Axis parameter

P7 : Set the braking time of spindle, also the hold time of output M05, Unit:10ms. The time less , the braking faster.

P8 : Set the braking signal is long signal 1 or short signal 0.

P9: To set system whether checking spindle feedback signal of spindle position, also the feedback signal is spindle encoder signal. To set the parameter value 1 means check; 0 means not to

check.

P10: To set feedback pulse numbers of spindle encoder turn a round, the value: Line number of encoder * 4.

P11: Whether turn on the spindle or not when shifting [1 means on, 0 means off]

P51: The speed of motor when spindle shifting (unit: 1/100rpm)

P52: The direction when spindle shifting (0 means positive, 1means negative)

P53: The stopping time when spindle shifting (unit: 10ms)

P54: Turning time of low speed when spindle shifting (unit:10ms)

P55: Stopping delay time of spindle (Unit:10ms)

In Speed parameter:

P8: To set the speed of spindle at manual condition. Unit: rpm.

P36: To set the max speed of spindle, also the speed of corresponding 10V.

Attention: when spindle system is with gears,this is the speed of first gear.

P37: To set the max speed of spindle(second gear), that's the turning speed of corresponding 10V instruction voltage.Unit: rpm.

P38: To set the max speed of spindle (Third gear), that's the turning speed of corresponding 10V instruction voltage. Unit: rpm.

P39: To set the max speed of spindle (Fourth gear), that's the turning speed of corresponding 10V instruction voltage. Unit: rpm.

P40: To set the highest speed of second spindle, also the speed of corresponding 10V. Unit: r/min

In Other parameter:

P13: To set whether spindle and chuck is interlocking or not: 0 means they are separately; 1 means the spindle only start turning when chuck on. The thumbstall can't be use when the spindle is turning.

Setting parameter is related with the configuration of lathe and user's service condition, but consider for safe, suggest setting 1,also interlocking.

4.3.1.2 Spindle Gear Shifting(M41/M42/M43/M44)

Remark	PIN	Function	Command
S01	PIN10_CN3 Plug	Output 1st gear of spindle	M41
S02	PIN23_CN3 Plug	Output 2nd gear of spindle	M42
S03	PIN11_CN3 Plug	Output 3rd gear of spindle	M43
S04	PIN24_CN3 Plug	Output 4th gear of spindle	M44

M41/M42/M43/M44 output S01/S02/S03/S04 for shifting gear of spindle , and adjust analog voltage to adjust speed of spindle.

P36 in Speed parameter is set for max speed of 1st class spindle;

P37 in Speed parameter is set for max speed of 2nd class spindle;

P38 in Speed parameter is set for max speed of 3rd class spindle;

P39 in Speed parameter is set for max speed of 4th class spindle;

Note:Functions interlocked and states reserved

4.3.1.3 Coolant(M08/M09)

M08:Turn on coolant

M09: Turn off coolant

Remark	PIN	Function
M08	PIN8_CN3 Plug	Turn On/Off coolant

4.3.1.4 Lubricate(M32/M33)

M32: Turn on lubrication

M33: Turn off lubrication

Remark	PIN	Function
M32	PIN9_CN3 Plug	Turn On/Off Lubrication

In Other parameter,

P4 controls the function of lubricate automatically.

P6 is set the interval time of lubrication (Unit: s);

P5 set the time of lubrication ,also holding time of output M32(Unit: 10ms).

4.3.1.5 Chuck(M10/M11)

M10/M11 are for control clamping/unclamping of chuck.

Remark	PIN	Function
M10	PIN21_CN3 Plug	Control Chuck clamping
M71	PIN9_CN10 Plug	Control chuck unclamping(Spare)
M12	PIN11_CN10 Plug	Detect position of clamping_chuck(spare)
M14	PIN24_CN10 Plug	Detect position of unclamping_chuck(Loose)(spare)
M16	PIN12_CN10 Plug	Input point for switch to control chuck(spare)

Chuck of this system control is related with parameter as follows:

In Other parameter:

P2: Type of chuck, (Inner: Chuck to center when M10; Outer: Chuck opening outward when M10). 1 means outer, 0 means inner.

P13: Interlock between Chuck & Rotation_Spindle.0: No interlock, 1: yes.

P15: Detect position of clamping/unclamping of chuck,0:No detect, 1: yes.

M12: input point for detecting position of clamping of chuck;

M14: input point for detecting position of unclamping of chuck.

P20: Type of controlling singal for chuck,0: one output controlling signal for chuck; 1: two output controlling signals for chuck.

M10:output point for controlling clamping of chuck , controlling code is M10;

M71:output point for controlling unclamping of chuck,controlling code is M11.

P22: External Switch(Foot switch) for chuck, reciprocating mode, price one time clamping chuck; press twice time, unclamping chuck. 0: without switch ; 1: with external switch for control chuck; input signal is M16.

P24: Holding time of output M10/M71 of chuck, unit: s. 0: mode type*.

4.3.1.6 Tailstock(M79/M78)

M79: Tailstock forward

M78: Tailstock backward

Remark	PIN	Function
M79	PIN22_CN3 Plug	Tailstock Forward/Backward
M73	PIN22_CN10 Plug	Control chuck unclamping(Spare)
M18	PIN11_CN10 Plug	Detect position of forward_Tailstock(spare)
M28	PIN24_CN10 Plug	Detect position of backward_Tailstock(spare)
M14	PIN24_CN10 Plug	Input point for switch to control Tailstock(spare)

Parameters set for tailstock,in Other parameter:

P16: Detect Position of Forward/Backward of Tailstock; 0: no detect,1:Yes;

M18: input point for detecting position of forward of tailstock;

M28: input point for detecting position of backward of tailstock.

P21: Type of controlling singal for tailstock,0: one output controlling signal for tailstock; 1: two output controlling signals for tailstock.

M79:output point for controlling forward of tailstock , controlling code is M79;

M73:output point for controlling backward of tailstock,controlling code is M78.

P23: External Switch(Foot switch) for tailstock, reciprocating mode, price one time, tailstock forward; press twice time, tailstock backward. 0: without switch ; 1: with external switch for control tailstock; input signal is M14.

P25: Holding time of output M79/M73 of tailstock, unit: s. 0: mode type*.

Note: 1. When user-defined signals M71/M70 , M73/M72 are used for output signal of spindle chuck and Tailstock, it can't be used for other functions.

2. M12,M14,M1,M18,M28 are multi-function codes, only one function when using.

3. *mode type means that once valid,always valid until cancelling/resetting code.

4.3.1.7 Condition Output of Machine Tool(M65/M67/M69)

M65/M67/M69 be used for output condition of machine tool, set by parameters

Remark	PIN	Function
M65	PIN20_CN3 Plug	Output Alarm condition of machine tool
M67	PIN8_CN3 Plug	Output Running condition of machine tool
M69	PIN21_CN3 Plug	Output Pausing condition of machine tool

Parameter sets for condition output ofmachine tool. In Other parameter:

P28: M65/M69 are as outputs for run/halt condition of machine tool,0:no,1:yes.

P29: M67 is as output for alarm condition of machine tool, 0: no, 1:yes.

Note:1.When M65,M67,M69 are used for output condition of machine tool, and then they can't be used for other functions.

2.Valid level of all output points is 0V. When fixed with inductance load, must connected with the reverse diode to protect inner circuit of cnc system.

4.3.2 M Input Command List

No.	Code	Function Introduction	Statement
1	M12	Check M12 is valid	These codes can be used for conditional wait or conditional jump.
	M13	Check M12 is invalid	
2	M14	Check M14 is valid	
	M15	Check M14 is invalid	
3	M16	Check M16 is valid	
	M17	Check M16 is invalid	
4	M18	Check M18 is valid	
	M19	Check M18 is invalid	
5	M28	Check M28 is valid	
	M29	Check M28 is invalid	
6	M22	Check M22 is valid	
	M23	Check M22 is invalid	
7	M24	Check M24 is valid	
	M25	Check M24 is invalid	

Note: All M input commands, input effective level is 0V.

There are two kinds of special application about M input commands,as following,

a) Conditional Wait

Example: M12

When Input point M12 is valid, program goes on run following blocks, if M12 is invalid, system will wait if M12 is valid.

b) Conditional Jump

Example: M14 P0120

When the program running to this block and the system detecting if M14 input signal is valid. When M14 is valid, program will jump to 120th line of program (also N0120 block), otherwise , system will execute the next block.

4.4 Analog Speed of Spindle(S , SS)

SZGH CNC system support dual analog outputs for SP_speed.

Speed of 1st spindle is set by "S**"; Speed of 2nd spindle is set by "SS**".

P36 in Speed parameter is set for max speed of 1st spindle; P40 in Speed parameter is set for max speed of 2nd spindle.

Note: D11_P9=1 in Other parameter is set for output 1st analog voltage to dual analog output(+10V of CN3&CN10) at same time, without function of "SS**".

There are two gears control ways for 1st spindle as following,

(1) Four gears spindle speed electrical control, output four bits code of step speed change, M41-M44 control instructions correspondly output S01-S04 code, with fixed speed. P50/P51/P52/P53/P54 in Axis parameter are set for mode of shifting.

(2) Four gears+Variable speed, M41-M44 instruction control, correspond the output S01-S04 code. P42/P43/P44/P45 in Speed parameter are set for maximum speed of corresponding gear, P50/P51/P52/P53/P54 in Axis parameter are set for mode of shifting.

Variable speed,range is 0-99999, output 0-10V variable-frequency voltage.

Note: Output 10V is corresponding to max speed of spindle.

4.5 T Tool Function Command

Format	Function
Tab	a: Exchanging Tool Number b: Tool compensate number

Note: 1. a=0 ; means that don't exchange tool;

2. b=0; means that don't make tool compensation , display machine coordinate;

3.Range: a= 0~99 ; b=0~99.

Eg1: T0204 ; change to No.2 tool , and make No.4 tool compensation.

T0300 ; change to No.3 tool , and don't make tool compensation.

T0004 ; don't exchange tool , only make No.4 tool compensation.

When machine tool is with electrical turret, each station can fix serval tools,adopt different tool compensation to exchange tools.

Eg2: 4 station turret, there are 2 tools in 3rd station (use No.3 & No.5 tool compensation)

```
N0000   T0101      ; change to No.1 tool, and make no.1 tool compensation
N0001   G0   X30 Z500
N0002   T0303      ; change to 1st tool in 3rd station
N0003   G00   X50
N0004   T0505      ; change to 2nd tool in 3rd station
N0005   M02
```

4.6 User-defined macro instruction(G120-G160,M880-M889)

Every user-defined G code is corresponding to a macro program ProgramGxxx, the M code is corresponding to a macro program of ProgramUser0 --ProgramUser9, the user cannot programme the macro program in NC system, must edit the macro code in the computer, and then copy into the system.

For example, defines the G152 function: the arc model porous drilling cycle. (must copy the macro program ProgramG152 into system).

Format:G152 Xx Yy Zz Rr Ii Aa Bb Hh Ff;

X: The X coordinate with absolute value or incremental value of center to specify.

Y: The Y coordinate with absolute value or incremental value of center to specify.

Z: Hole depth

R: Approaching fast to the point coordinate

F: Cutting feed speed

I: Radius

A: The angle of the first hole

B: Incremental angle specify(CW when negative)

Macro program ProgramG152 as follows:

```
#80=#0
#81=#1
#82=#2
#83=#3
#84=#4
#85=#5
#86=#6
#87=#7
#88=#8
#89=#9
#90=#10
#91=#11
#92=#12
#93=#13
#94=#14
#95=#15
#96=#16
#97=#17
#98=#18
#99=#19
#100=#20
#30=#4003
#31=#4014
G90
IF[#30 EQ 90] GOTO 1
G53
```

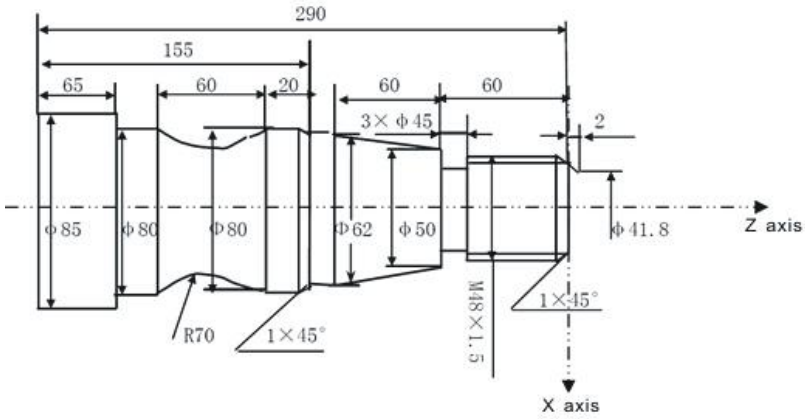
```
#98=#5001+#98
#99=#5002+#99
N1 WHILE[#86 GT 0] DO 1
#35=#98+#87*COS[#80]
#36=#99+#87*SIN[#80]
G81X#35Y#36Z#100R#92F#85
#80=#80+#81
#86=#86-1
END 1
G#30 G#31 G80
M99
```

4.7 Synthetic instance for programming

In the actual programming, must according to the drawing and processing requirement to select the correct install folder mode and suitable tool, and combined with the actual working performance of lathe to select the right cutting allowance, for example:

Example 1：The tool is:

T01 cylindrical cutting tool; T02 cutting groove, tool width 3m; T03 thread tool with 60 degree angle



Program as follows:

N10 M03 S1000; Start spindle

N20 T0101; Choose the first tool and execute the first redeem

N30 G00 X41.8 Z2 M08; Move fast to the cutting point，cutting fluid is on

N40 G01 X48 Z-1 F100; Chamber

N50 Z-60; Fine machining for thread

N60 X50; Tool is backing

N70 X62 W－60; Fine machining in cone

N80 W－15; Fine machining in Φ62mm ex-circle

N90 X78; Tool is backing

N100 X80 W-1; Chamber

N110 W-19; Fine machining in Φ80mm ex-circle

N120 G02 X80 W-60 R70; Fine machining in arc（I63.25 K－30）

N130 G01 Z-225; Fine machining in Φ80mm ex-circle

N140 X85; Tool is backing
N150 Z-290; Fine machining in Φ85mm ex-circle
N160 X90 M09; Tool is backing，cutting fluid is off
N170 G00 X150 Z50; Move fast to the point of changing tool
N180 T0202; Change tool and set the No.2 redeem
N190 M03 S800; Change speed of spindle
N200 G00 X51 Z-60 M08; Move fast to the processing point，use the left point of tool to redeem
N210 G01 X45 F90; Cutting Φ45mm groove
N220 G00 X51; Tool is backing
N230 X150 Z50 M09; Return to the point of backing tool，cutting fluid is off
N240 T0303; Change tool and set the redeem
N250 M03 S1500; Change speed of spindle
N260 G00 X62 Z6 M08; Move fast to the processing point，cutting fluid is on
N270 G92 X47.54 Z－58 F1.5; Cutting thread is cycle
N280 X46.94;
N290 X46.54;
N300 X46.38;
N310 G00 X150 Z50 M09; Return to the point of start cutting, cutting fluid is off
N320 T0300; Cancel redeem
N330 M05; Stop spindle
N340 M30; Program is over