ADT-CNC9620
CNC9620 Lathe Series Control System
Programming Manual

ADTECH (Shenzhen) Technology Co., Ltd.
Add: 5/F, Building 27-29, Tianxia IC Industrial Park, Yiyuan Road,
Nanshan District, Shenzhen, 518052, China
Tel: 0755-26722719 Fax: 0755-26722718
E-mail: tech@adtechcn.com http://www.adtechcn.com

Copyright Statement

Version information

Item No.	Initial upload	Version number	Total No. of Pages	Prepared by	Typeset by
	Month/Day/Year	A0101			

Proofreading record			
Date	Rev./Pages	Result	ACK

Note: The table of above version is updated only for the version that has been changed in the Manual

1. ADTECH (Shenzhen) Technology Co., Ltd. has made a strict and careful emendation and verification of this Manual, but no guarantee is provided to ensure that it is completely free from any errors and negligence.
2. Adtech is committed to constantly improving product functionalities and service qualities. Therefore, it reserves the right to change any products and software programs described in this Manual as well as its content, without prior notice.

Precautions and Explanations

- ※ **Transport and storage:**
- Do not stack product package more than six layers;
- Do not climb, stand on or place heavy stuff on the product package;
- Do not pull the cable still connecting with machine to move product.
- Forbid impact and scratch on the panel and display. Prevent the product package from humidity, sun exposure, and rain.

- ※ **Open-box inspection:**
- Open the package to confirm the product to be purchased by you.
- Check damages situation after transportation;
- Confirm the integrity of parts comparing with the parts list or damages situation. Contact our company promptly for discrepant models, shortage accessories, or transport damages.

- ※ **Wiring**
- Ensure the persons involved into wiring and inspecting are specialized staff;
- Guarantee the product is grounded with less than 4Ω grounding resistance. Do not use neutral line (N) to substitute earth wire;
- Ensure grounding to be correct and solid, in order to avoid product failures or unexpected consequences;
- Connect the surge absorption diodes to the product in the required direction, otherwise, the product will be damaged. Ensure the power switch is OFF before inserting or removing plug, or disassembling chassis.

- ※**Overhauling**
- Ensure the power is OFF before overhauling or components replacement;
- Make sure to check failures after short circuit or overloading, and then restart the machine after troubleshooting. Do not allow to frequently connect and disconnect the power, and at least one minute interval between power-on and power-off.

Table of Contents

1. Programming Basics

1.1. Overview of CNC Machining

The CNC machine tool is composed of a Computer Numerical Controller system (CNC), a servo motor (or a stepper motor) drive unit, and a machine tool (including the headstock, feed transmission mechanism, workbench, tool apron, electrical control cabinet, etc.). After the part program edited by the user is processed by the CNC, it sends motion and control instructions. The motion instructions drive the feed motion of the machine tool via the motor drive unit. The spindle start-stop, tool selection, cooling, lubrication and other controls are available by control instructions. In this way, the cutting of the parts are available through the relative movement of both tool and the workpiece.

CNC programming is the process of writing the parts program based on the programming instructions specific to the CNC system by including the information such as the outline dimensions, machining process, technological parameters, tool parameters and so on. CNC machining is a process in which the CNC system controls the machine tool to complete the parts machining in accordance with the requirements of the part program. The working principles of CNC machine tools and process flow of CNC machining are shown in figure below.

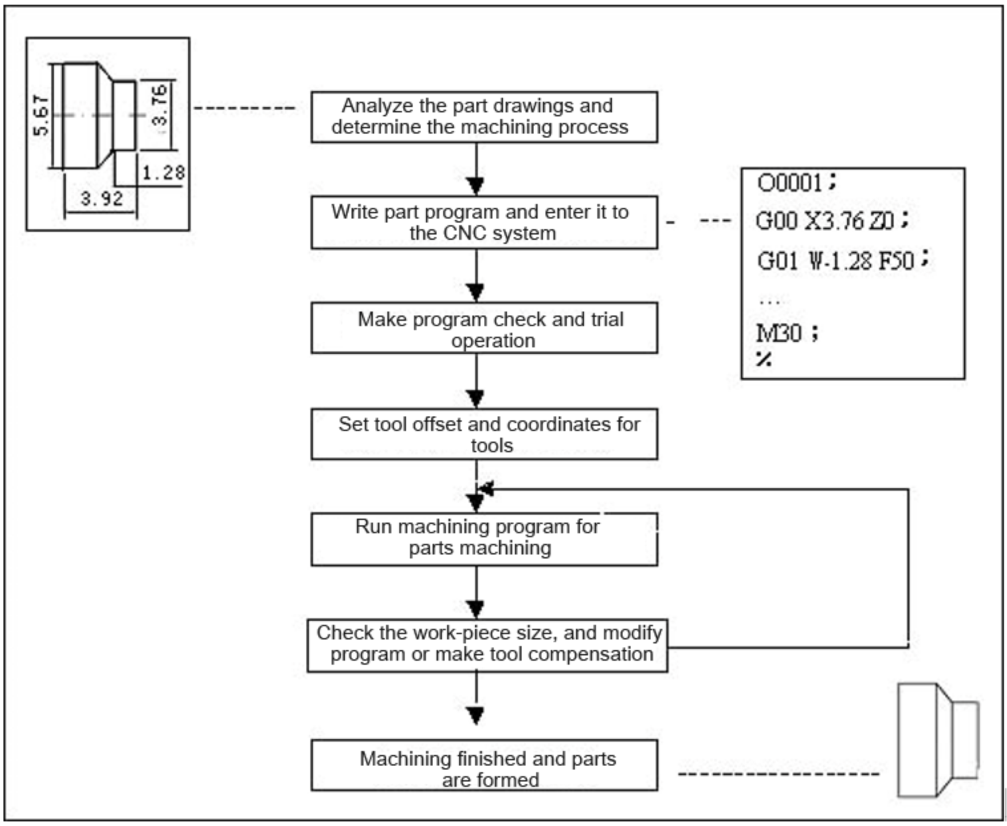


Fig.1-1 Process flow diagram of CNC machining

1.2. Definition of coordinates

Fig. 1-2-1 is a schematic diagram of a CNC lathe

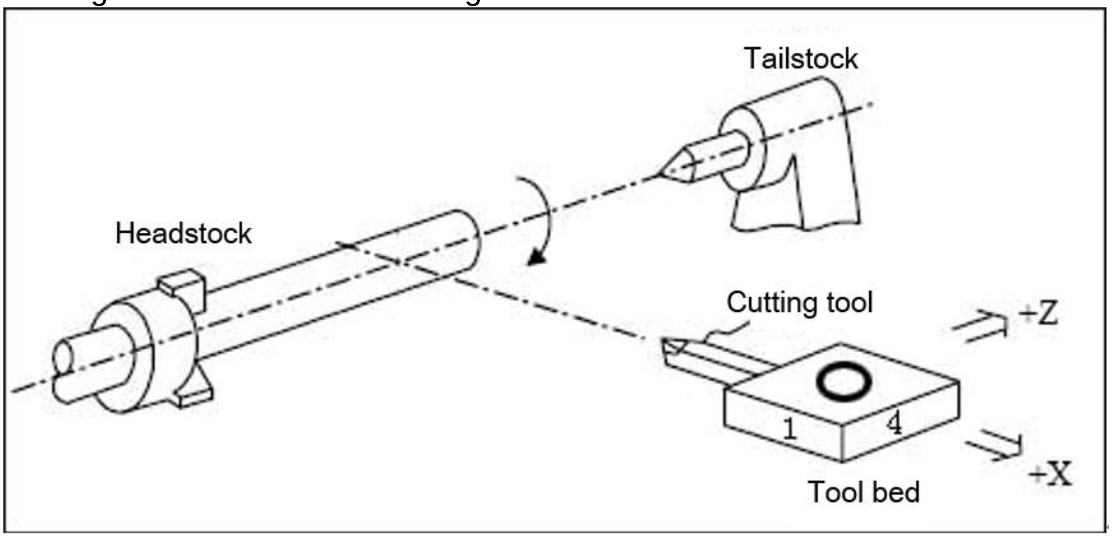


Fig. 1-2-1

The system applies a rectangular coordinate system composed of X-axis and Z-axis, of which the X-axis is perpendicular to the spindle, the Z-axis is parallel to the spindle direction, the direction approaching the workpiece is negative, while that leaving the workpiece is positive.

The CNC lathe is, based on the relative position of the tool apron and the spindle of machine tool, provided with a front tool apron and a rear tool apron. The same programming instructions have different movement trajectories in these two tool aprons. The system can be used as the CNC lathe for front tool apron and the rear tool apron. As can be seen from the figure below, the X-direction of coordinate systems of the front and back tool aprons are exactly opposite, while the Z-direction is in the same direction. In the diagrams and examples below, the application of programming is explained using the coordinate system of front tool apron.

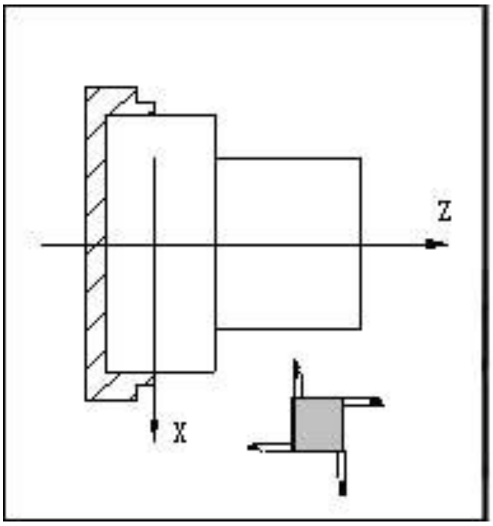


Fig. 1-2-2 Coordinate system of front tool apron

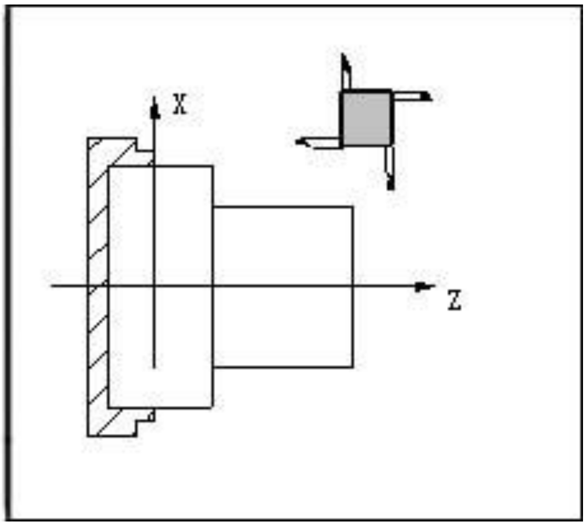


Fig. 1-2-3 Coordinate system of rear tool apron

1.3. Coordinate system and mechanical zero of machine tool

The coordinate system of machine tool is the reference coordinate system used by the CNC for coordinate calculation. It is a coordinate system specific to machine tool. The origin of the coordinate system is called the mechanical reference point or the mechanical zero.

The mechanical zero is determined by the zero switch or zero return switch that is mounted on the machine tool. In general, the zero switch or zero return switch is mounted at the maximum stroke in the positive direction of the X-axis and Z-axis. Perform the mechanical zero return operation. After returning to the mechanical zero, the system sets the current coordinates of machine tool to zero, and establishes a machine tool coordinate system by taking the current position as the origin of coordinates.

Note: If there's no zero switch on the lathe, the mechanical zero return operation is unavailable.

1.4. Work-piece coordinate system and program zero

The work-piece coordinate system is a rectangular coordinate system established on the part drawing to facilitate the programming. It is also known as a floating coordinate system. After the part is clamped to the machine tool, establish the work-piece coordinates in the system by setting the absolute coordinates of the current tool position using the G50 command against the relative position of the tool and the workpiece. The current tool position is called the program zero. Usually, the Z-axis of the work-piece coordinate system coincides with the spindle, and the X-axis is located at the head or tail of the part. Once the work-piece coordinate system is established, it remains valid until it is replaced by a new work-piece coordinate system.

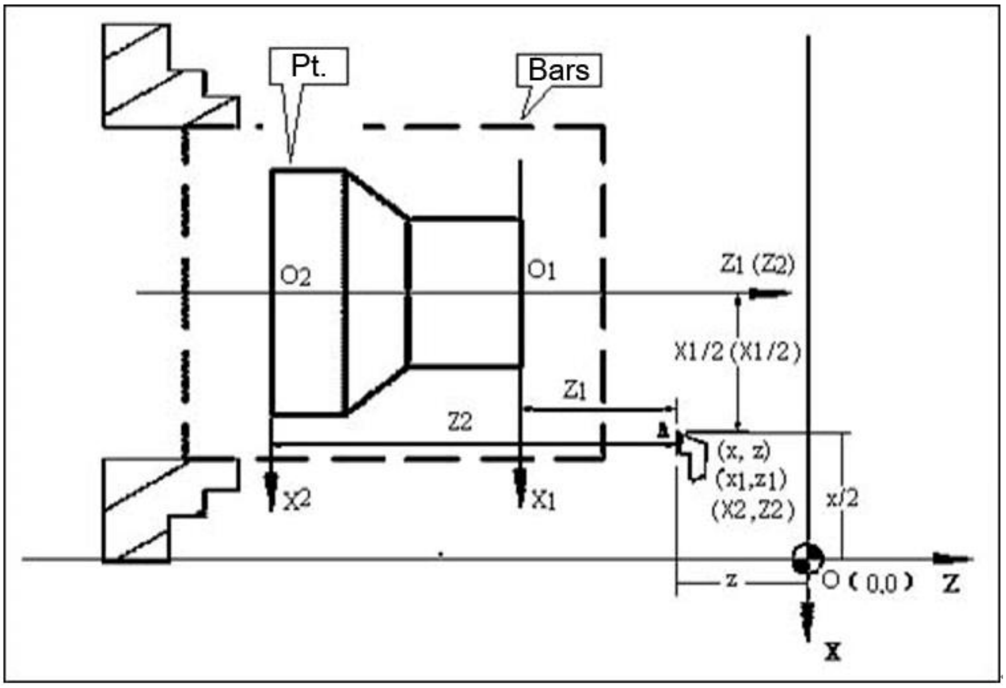


Fig. 1-4

In the figure, XOZ is the coordinate system of machine tool, X1O1Z1 is the work-piece coordinate system of the X-axis at the head of the workpiece, X2O2Z2 is the work-piece coordinate system of the X-axis at the tail of the workpiece, O is the mechanical zero, and A is the tool nose. The coordinates of A in the above three coordinate systems are shown as follows:

The coordinate of Point A in the machine tool coordinate system is (X, Z);

The coordinate of Point A in the X1O1Z1 coordinate system is (X1, Z1);

The coordinate of Point A in the X2O2Z2 coordinate system is (X2, Z2).

Interpolation

Interpolation refers to a two-dimensional (plane) or three-dimensional (spatial) contour formed based on the mathematical relationship determined by the trajectories of the simultaneous movement of two or more axes.

Interpolation is also known as the contour control. The motion axis controlled during interpolation is called a compounding axis. The movement amount, direction, and speed of the compounding axis are controlled simultaneously during the motion process so as to form the required synthetic motion trajectory.

It only controls the end motion point of one or more axes but does not control the trajectory during the motion process. This type of motion control is called point to point control.

The X-axis and Z-axis of this system are compounding axes, which belong to the 2-axis CNC system. The system is provided with the linear, circular and thread interpolations.

Linear interpolation: The synthetic motion trajectory of the X-axis and Z-axis is a straight line from the start point to the end point.

Circular interpolation: The synthetic motion trajectory of the X-axis and Z-axis is an arc from the start point to the end point, of which the radius is specified by R, or the center of the circle is specified by I and K.

Thread interpolation: X-axis, Z-axis, or two-axis motion and rotation of the spindle are interpolated. The F instruction value gives the pitch of the thread. Pitch refers to the amount of movement (unsigned) of the axis (X-axis or Z-axis) that moves a long distance during thread cutting process when the spindle makes one revolution. This system is designed to machine the metric straight threads, taper threads, and end threads. The machine tool can only perform thread cutting by mounting a spindle encoder. No other operation is permitted if the spindle encoder is not installed, and the system cannot receive the encoder signal caused by threading. (1000 wire-level and above encoder is recommended by the system.)

1.5. Programming of absolute coordinate and relative coordinate

There're two ways to specify the end position of the trajectory during programming:

The first type: the position of the end point of the trajectory is represented by the absolute coordinates, which is known as the absolute coordinate programming (the instruction address uses X, Z).

The second type: The position of the end point of the trajectory is represented by the coordinate difference between the end point and the starting point. It is known as the relative coordinate programming (instruction address uses U, W). When the relative coordinate is of a negative value, it indicates that it runs in negative direction along the axis; when it is of a positive value, it indicates that it runs in positive direction along the axis.

This system allows to represent an axis in the position of the end point of the trajectory in the same program segment using the absolute coordinates, and the other axis is represented using the relative coordinates. This type of programming is called hybrid programming.

Example: A → B linear interpolation. (Fig. 1-5)

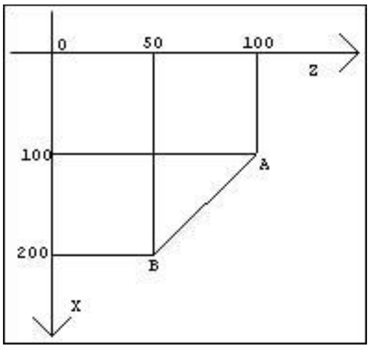


Fig. 1-2-5

Absolute coordinate programming: G01 X200 Z50;

Relative coordinate programming: G01 U100 W-50;

Hybrid coordinate programming: G01 X200 W-50; or G01 U100 Z50.

1.6. Conversion between imperial and metric systems*

Whether the input unit is of imperial or metric, choose it using G code (G20, G21).

System of units	G code	Minimum unit
Imperial unit	G20	0.0001 inch
Metric	G21	0.0001 mm

To conversion between imperial and metric systems, the separate segment instruction is used for code G before the program and setting the coordinate system. The system of units for each of the following values changes according to the code G for the conversion of imperial or metric system.

(1) F represents the value of feed speed instruction.
(2) Position-dependent instruction value.
(3) The compensation.
(4) The value of one skip on the scale of MPG
(5) The amount of movement of one step
(6) Partial value of the parameters

Notes:

1. When the system is powered on, the G code for conversion between the imperial and metric systems remains the same as before the power is off.
2. Do not change G20, G21 in the midway of the programming
3. When the mechanical unit system differs from the input unit system, the maximum error is 0.5 of the minimum movement unit. This error does not accumulated.
4. For the conversion between the imperial input (G20) and the metric input (21), the offset must be set to be consistent with the new setting of the input unit.

1.7. Composition of program

To complete the automatic machining of the part, the user needs to write a part program (a "program") according to the instruction format issued by the CNC system. The CNC system executes the program to finish the controls of the machine's feed motion, spindle start and stop, tool selection, cooling, lubrication, etc., to allow the machining of parts.

Examples:

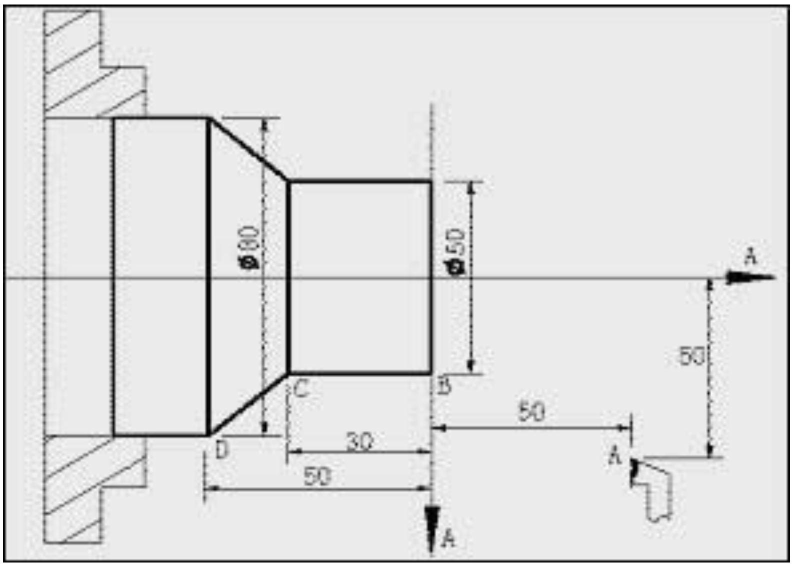


Fig.1-7

O0001 ; (Program name)
G0 X100 Z50; (Quick positioning to Point A)
M12; (Clamping the work-piece)
T0101; (Change to No.1 tool to execute, and No.1 tool offsets)
M3 S600; (Start the spindle and set its speed to 600 rpm)
M8 (Turn on the coolant)
G1 X50 Z0 F600; (Approach Point B at a speed of 600 mm/min)
W-30 F200; (Cut from Point B to Point C)
X80 W-20 F150; (Cut from Point C to Point D)
G0 X100 Z50; (Quickly return to Point A)
T0100; (Cancel the tool offset)
M5 S0; (Stop the spindle)
M9; (Turn off the coolant)
M13; (Release the work-piece)
M30; (When the program is ended, turn off spindle and coolant)
%

After executing the above program, the tool will act in the trajectory of A → B → C → D → A.

1.8. General structure of the program

The program is composed of several lines of segments beginning with "OXXXX" (program name) and ending with "%". The segment is composed of several instruction words starting with the segment number (can be omitted) and ending with the carriage return "CR" and linefeed "LF". The general structure of the program is shown in Fig.1-3-2:

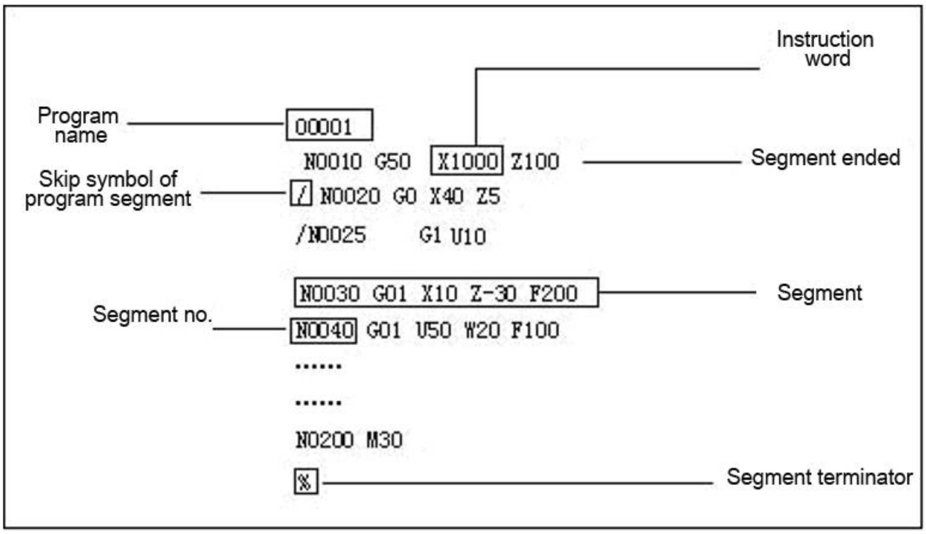
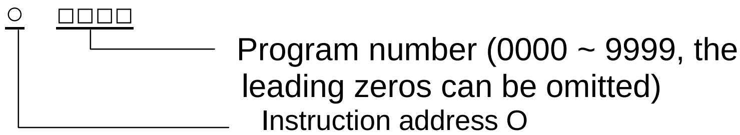


Fig. 1-8 General structure of the program

1) Program name

To identify and distinguish each program, there's a program name consisting of an instruction address O and four digits followed by at the beginning of each program. The system can store up to 9,999 programs. The program names of each program shall not be repeated.



2) Instruction word

The instruction word is the basic instruction unit used to instruct the CNC system to have the controls done. It consists of an English letter (the "instruction address") and the subsequent value (the "instruction value", which is a signed or unsigned number). The instruction address specifies the meaning of subsequent instruction values. In case of different combinations of instruction words, the same instruction address may have different meanings.

X 1000 X -1000

Instruction address Instruction value Instruction address Instruction value

Instruction word Instruction word

3) Skip symbol of segment, segment number and segment

The program contains several program segments. The program is executed in units of program segments. Usually, the next program segment can be executed only when a segment is executed. The segments are separated by the characters ";" or "*", which are indicated by ";" in this Manual. A segment starts with a segment number, ends with ";" or "*" and it may consist of several instruction words.

For example, the segment can be preceded by a "/" symbol, which is known as a skip symbol

4) If the skip function is enabled when the program runs automatically, the program will not execute this segment of program when the program runs into it, and the program will execute the next segment of program. If the skip function is not enabled, this segment of the program will continue to execute. To choose the skip function, you need to enter the auxiliary screen under the main screen of system, and press the soft button. This function will not be saved in case of power off. The system is initialized to "not enable the skip function".

/ N0100 G0 X200 Z300 ;

Terminator of program segment

Segment no.

Skip symbol of program segment

5) Segment no.

N0000 ~ N9999, and the leading zero can be omitted. The segment number may be omitted, while the target segment for program call and jump are indispensable. The segment numbers can be arranged in arbitrary order. The segment numbers in the back does not have to be greater than the previous ones. For easy access, the line numbers are generally sorted by a certain increment value. When editing manually, you can decide whether or not to insert the line number increment automatically by using No.47 comprehensive parameters, which is initialized to 0, i.e. the line number will not be automatically inserted.

6) Segment terminator

The program starts with the program name and ends with "%". "%" is the terminator of the program file. It is also the communication ending flag and leading flag when the program is communicated.

1.9. Master routines and subroutines

To simplify programming, when the same or similar machining trajectories and control processes need to be used multiple times, the program instructions in this section can be edited as independent programs to be called. Programs that call other programs are called master routines, and programs that are called (ending with M99) are called subroutines. Like the master routines, subroutines also occupy the system's program capacity and storage space. The subroutine must also have its own independent program name. The subroutine can be called by any other master routines, or may run independently. After a subroutine ends, it returns to the master routine to continue execution. The system supports nine-fold nesting of programs, that is, the subroutine may also call the other subroutines. See Fig.1-9 below.

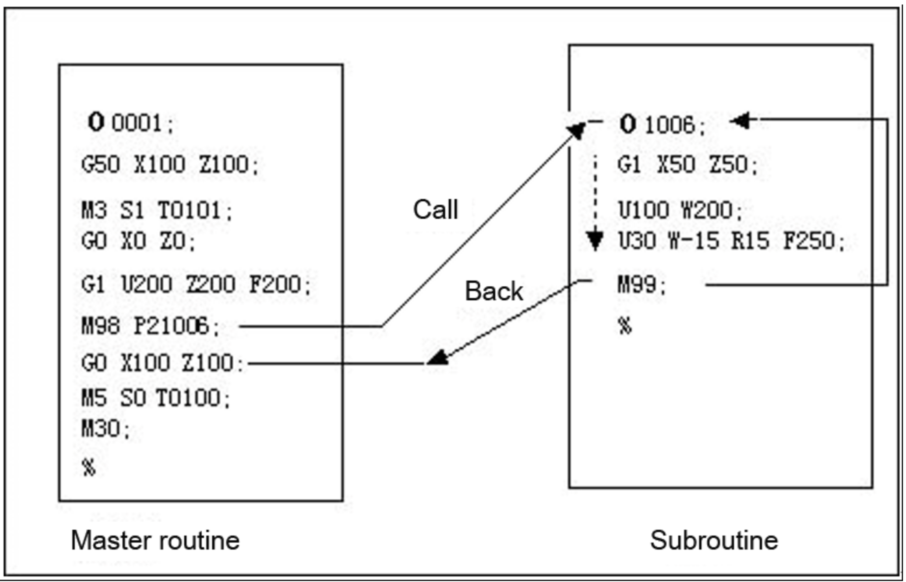
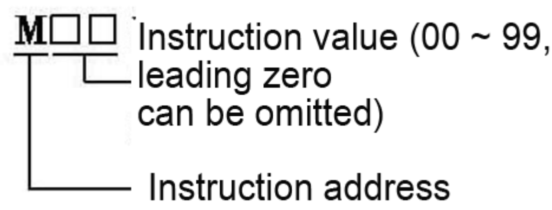


Fig. 1-9

2. M S F T Instruction

2.1. Auxiliary function (M instruction)

The M instruction is composed of the instruction address M and the subsequent 1-2 digits, which are used to control the flow of program execution or output signals to the machine tool.



Only one M instruction is valid in a segment. When two or more M instructions appear in the segment, the last M instruction is valid.
When the M instruction and the instruction word that executes the motion function are in the same segment, the execution sequence can be as follows:

① When the M instruction is M00, M30, M98, and M99, execute the M instruction after the movement is done;
② When the M instruction outputs a signal to the machine tool, the instruction is executed while moving.

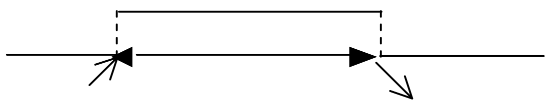
List of M instructions

Instructions	Functions	Remark
M00	Program pause	Status is not maintained
M30	End of program run	
M98	Call a subroutine.	
M99	Return from a subroutine.	
M03	Spindle Forward Rotation	Function interlocking, and the status is maintained
M04	spindle reverses	
*M05	Spindle stops	
M08	Coolant ON	Function interlocking, and the status is maintained
*M09	Coolant OFF	
M10	Tailstock forwards	Function interlocking, and the status is maintained
M11	Tailstock retracts	
M12	Chuck clamped	Function interlocking, and the status is maintained
M13	Chuck loosened	

M32	Lubricant ON	Function interlocking, and the status is maintained
*M33	Lubricant OFF	
M40	Gear setting speed output OFF	
M41	First gear speed output	
M42	Second gear speed output	
M43	Third gear speed output	
M44	Fourth gear speed output	
M88	Detect the signal of the specified input pin	Specify to input active level
M89	Switch control of specified output pins	Specify the output level

Note: The instructions marked with "*" are valid when it is power on.

After the system executes the M instruction that outputs a signal to the machine tool, it delays for a period of time before executing subsequent instruction words or segments. This delay time can be set by the M code waiting time in system parameters.



M code starts execution Delay time

Start to execute the subsequent instruction words or segments

2.1.1. Subroutine calls M98

Format:

M98 P○○○ □□□□

The called subroutine number (0000 ~ 9999). When the number of calls is not entered, the leading 0 of the subroutine number cannot be omitted; when the number of calls is entered, the subroutine number must be 4 digits;

Number of calls (1 - 999). No entering is required when it is called once

Function: After the execution of other instructions in the current segment is done, the system does not execute the next segment, but executes the subroutine specified by P.

The subroutine can be executed up to 999 times.
Subroutines cannot be called under MDI mode.

2.1.2. Return from subroutine and return to master routine M99

Format: M99 P ○ ○ ○ (return from subroutine)

Function: When the called subroutine is finished, it returns to the segment specified by P in the master routine to continue execution. When P is not entered, it returns to the master routine, and calls the subsequent segment of the current subroutine M98 instruction to continue execution. Where M99 is used to end the master routine (that is, the current program is not called and executed by other programs), the current program will be repeatedly executed. M99 instruction is invalid when running under MDI.

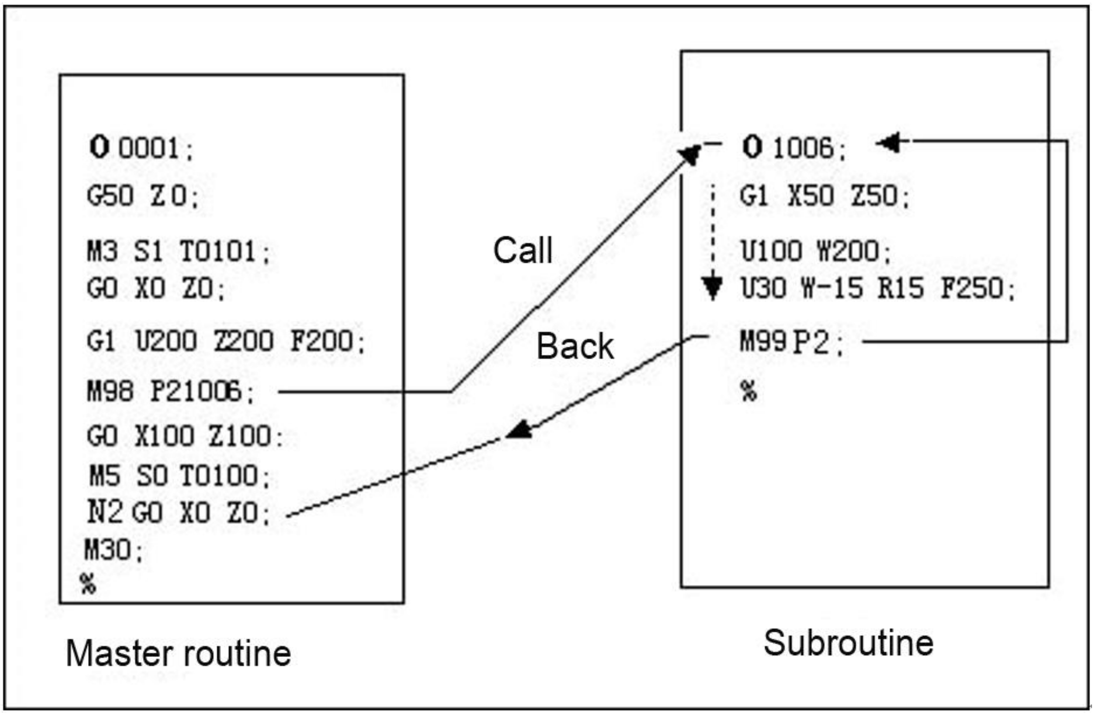


Fig. 2-1-1 Return from subroutine

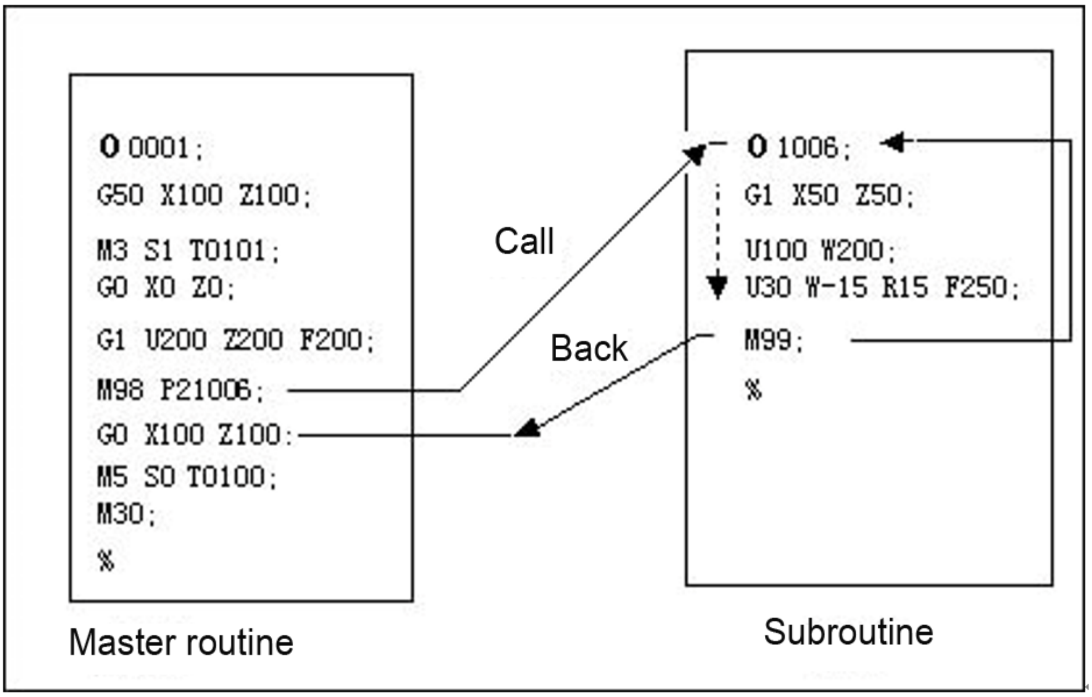


Fig. 2-1-2 Return to master routine

The system can call the nine-fold subroutines, that is, it can call other subroutines in the subroutine (see figure below)

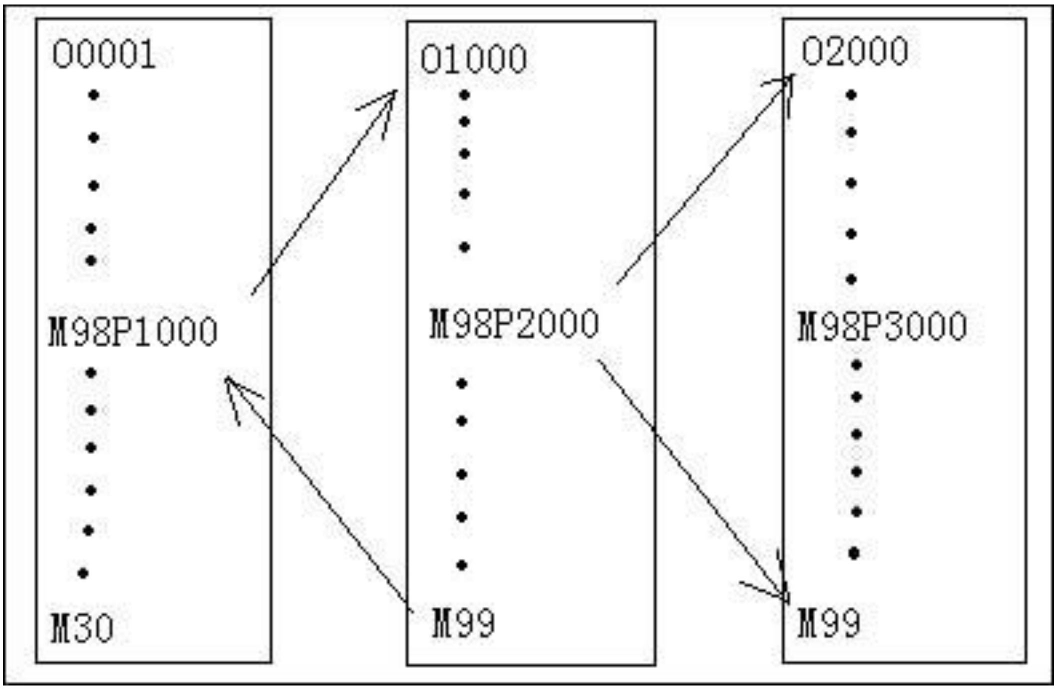


Fig. 2-1-3 Call the routine nesting

2.1.3. M03, M04 and M05 for spindle controls

Function: M03 or M3: spindle forwards;
M04 or M4: spindle reverses;
M05 or M5: spindle stops.

The M05 output is valid after system is powered on. When M05 output is valid, execute M03 or M04, M03 or M04 output is valid and maintained, and cancel M05 output (output is invalid); when M03 or M04 output is valid, execute M05, cancel M03 or M04 output, M05 output is valid and maintained. The interlock between the spindle and the chuck can be selected through the management parameter 022. It is defaulted to MFUNC (L) 1, of which no interlocking is made. If MFUNC (L) 2 is selected, it will be interlocked. User-Def is a parameter in which the user needs to define the M code. The parameter setting will be valid after system restart.

Note: In case of emergency stop to the system, the output of M03 and M04 is cancelled; however, the output of M05 is valid.

2.1.4. M08 and M09 for coolant controls

Function: M08 or M8: cooling pump ON;
M09 or M9: cooling pump OFF.

After the system is powered on, M09 is valid, which means that M08 output is invalid. When M08 is executed, the M08 output is valid and the cooling pump is turned on; when M09 is executed, the M08 output is cancelled and the cooling pump is turned off. The coolant control port is specified by port parameter 075. It is initialized to OUT4.

Note 1: Cancel the output of M08 during emergency stop of the system.

Note 2: There's no corresponding output signal for M09. If M09 is executed, the output of M08 will then be cancelled.

2.1.5. M10 and M11 for tailstock controls

Function: M10: Tailstock forwards.
M11: Tailstock retracts.

After system is powered on, both M11 and M10 have no output. When M10 is executed, M10 output is valid, M11 output is cancelled, and the tailstock forwards. When M11 is executed, M11 output is valid, M10 output is cancelled, and the tailstock retracts. M10 and M11 cannot be valid at the same time.

Note 1: M10, M11 output status remains unchanged during system reset and emergency stop.

2.1.6. M12 and M13 for chuck control

Function: M12: Chuck locked;

M13: Chuck is released.

After system is powered on, there is no output from M12 and M13. If M12 is executed, M12 output is valid and M13 output is cancelled; if M13 is executed, M13 output is valid and M12 output is cancelled. M12 and M13 cannot be valid at the same time. The chuck locking port is OUT8 by default, and the chuck release is OUT9 pin by default. The OUT8 output is valid when the chuck is locked. The OUT9 output is effective when the chuck is released. The external input control port is IN12 pin. For the interlocking between the chuck and the spindle, no interlocking if choosing MFUN (L) 1 using management parameter 022, and interlocking if choosing MFUNC (L) 2. The M12 and M13 are implemented by macro programs, which can be defined by user. After changing the 022 parameter to User-Def, this can be implemented by writing a macro program of M code.

Note 1: During system reset and emergency stop, the output status of M12 and M13 remains unchanged.

Note 2: The chuck can be controlled by an external input signal.

2.1.7. M32 and M33 for lubricant control

Function: M32: Lubrication pump ON;

M33: Lubrication pump OFF.

After system is powered on, M33 is valid, which means that M32 output is invalid. When M32 is executed, the M32 output is valid and the lubrication pump is on; when M33 is executed, the M32 output is cancelled, the lubrication pump is off; the lubrication output port is specified by port parameter 075, which is OUT5 pin by default.

Note 1: M32 output is invalid during system emergency stop;

Note 2: There's no corresponding output signal for M33. When M33 is executed, the output of M32 is cancelled.

2.1.8. Program stop M00

Format: M00 or M0

Function: After the other instructions in the current segment is executed, the program stops running. The next segment is run after pressing the cycle start key.

2.1.9. Program ends and returns to the beginning of the program M30

Format: M30

Function: After the other instructions in the current segment are executed, the automatic operation stops; when the M05 and M09 are executed, the number of machined parts is increased by one. The cursor returns to the beginning of the program.

2.2. Programmable input/output instructions

2.2.1. Programmable input instruction M88

Function: To allow the backup input point being defined by user.

Format: M88 Pxx Lx Qxxxx

P is used to specify the output port number range of 0-23.

L is used to specify the input active level, of which "1" means Active High, and "0" means Active Low.

Q is used to specify the detection time, in milliseconds.

Note 1: If the specified level is not detected within the specified time range of the Q instruction, an alarm indicating "program abnormal to terminate error" will be given.

Note 2: When no Q instruction is specified, it is defaulted as to waiting for input signal forever. The next instruction is not executed only after a valid signal is received.

Note 3: If the specified port is not in the range of 0-23, an alarm indicating "the specified port number error" will be given.

Note 4: If the P instruction is not programmed, the alarm will prompt as “Specified port number error”.

2.2.2. Programmable output instruction M89

Function: To allow the backup input point being defined by user.

Format: M89 Pxx Lx

P is used to specify the output port number range of 0-23.

L is used to specify the output level, of which "0" means Active High, and "1" means Active Low.

Note 1: If the specified port is not within the range of 0-23, an alarm indicating "The specified port number error" will be given.

Note 2: If the P instruction is not programmed, the alarm will prompt as "The specified port number error".

2.3. Spindle function (S instruction)

S instruction is composed of the instruction address S and the subsequent numbers, and is used to control the spindle speed.

Gear control: S _1~16 spindle rotation is controlled by switching 16-gear BCD code. In gear control mode, #061 comprehensive parameter must be 1, and port parameters #070~073 specify the output port of gear position.

Analog control: S _0 ~ max speed. In analog control mode, #061 comprehensive parameter must be 0, and it is required to set the maximum spindle rotation of parameter #20. The controller will output the analog voltage of 0~10V on the spindle port XS8 based on this parameter. If S instruction and the instruction word that executes moving function are in the same segment, motion instruction and S instruction are executed at the same time.

2.3.1. Spindle rotation switching control

Format: S_1 ~ 16.

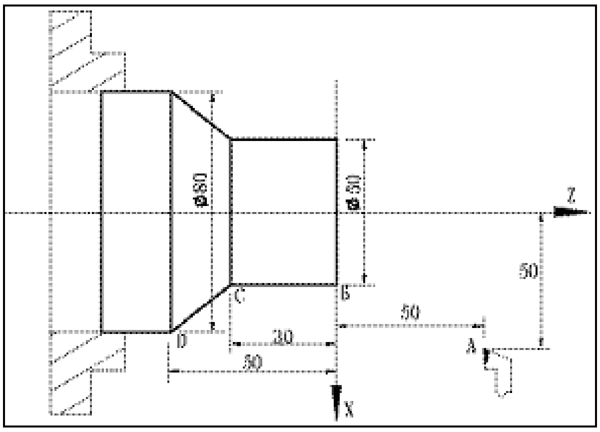
Function: 16-gear BCD coding gear control.

2.3.2. Spindle rotation analog voltage control

Format: M03 (M04) S____

Function: Set the spindle rotation speed, the system outputs 0~10V analog voltage to control spindle servo or inverter, achieve stepless speed change, and the value of S instruction is saved after power off.

Examples:



Program:

O0001;	(Program name)
M3 S300;	(Spindle forward rotation)
G0 X100 Z50;	(Quickly move to point A)
G0 X50 Z0;	(Quickly move to point B)
G1 W-30 F200;	(Cut from Point B to Point C)
X80 W-20 F150;	(Cut from Point C to Point D)
G0 X100 Z50;	(Quickly return to Point A)
M30;	(Program ends, spindle/coolant off)
%	

2.3.3. Spindle override

When the control mode of spindle speed analog voltage is effective, the actual speed of the spindle can be adjusted in 15 levels in real time by using the spindle override adjusting key within the range of 10% to 150% of the instruction speed (change 10% every level). Use the left and right arrow keys to modify the override in the controller's main screen, or modify the spindle override by pressing the knob on the additional panel. When modifying the spindle override with the left and right arrow keys, you can modify only after starting the spindle.

The actual speed upon spindle override adjustment is limited by the maximum speed of the spindle's current gear. The spindle override will not store after power off. The initial override is 100% when powered on.

2.3.4. G96 for constant linear speed control, and G97 for constant speed control *

Format: G96 S___; (S0000-S9999, leading zeros can be omitted).

Function: Constant linear speed control is valid, the cutting linear speed (m/min) is given, and constant speed control is cancelled. G96 is a modal G instruction. If the current modal is G96, G96 can be omitted.

Format: G97 S__; (S0000-S9999, leading zeros can be omitted).

Function: Cancel the constant linear velocity control and constant speed control are effective, and spindle speed (rpm) is given; G97 is a modal G instruction. If the current modal is G97, G97 can be omitted.

When cutting a workpiece on a lathe, the workpiece usually rotates using the spindle as the center-line. The cutting point when tool cutting the workpiece can be regarded as a circular motion around the spindle. The instantaneous rate in the circumferential tangential direction is called the cutting linear velocity (usually referred to as the "linear velocity").

The constant linear velocity control is valid only when the spindle speed analog voltage control is valid. During the constant linear velocity control, the spindle speed changes with the X-axis absolute coordinate of the programmed trajectory (ignoring tool length compensation). When the X-axis absolute coordinate value increases, the spindle speed decreases, and the X-axis absolute coordinate value decreases, the spindle speed increases, the cutting linear velocity then remains at the S instruction value.

Linear velocity= spindle speed * | X | * л / 1000 (m/min)

During the constant linear velocity control, the Z-axis in work-piece coordinate system must coincide with the spindle. Otherwise, the actual linear velocity is inconsistent with the given linear velocity.

Program:

O0001;	(Program name)
G50 S2000;	(Limit the maximum spindle speed at constant linear velocity RPM)
M3 G96 S300;	(Spindle forward, constant linear velocity effective 300 m/min)
G0 X100 Z50;	(Quickly move to point A)
G0 X50 Z0;	(Quickly move to point B)
G1 W-30 F200;	(Cut from Point B to Point C)
X80 W-20 F150;	(Cut from Point C to Point D)

G0 X100 Z50; (Quickly return to Point A)

M30; (Program ends, spindle/coolant off)

2.3.5. Maximum spindle speed limit *

Use the value following G50S to specify the maximum spindle rotation (r/min) of constant linear velocity control

G50 S___;

In constant linear velocity, the spindle rotation is limited to the maximum if it is higher than the value specified in above program.

2.4. Fast moving and feed (G98/G99, F instruction)

Three axis motion control modes are provided in this system: fast moving, cutting feed and manual feed.

2.4.1. Rapid Movement

Fast moving: For a lathe, the X-axis direction and the Z-axis direction move at their respective independent rapid moving rates, which are set by parameters 105 and 107. The movements in the two directions do not constitute a definite straight or a circular arc trajectory. The system can perform fast moving in two modes: G instruction and manual fast. When moving in manual fast mode, the X-axis direction and Z-axis direction cannot move at the same time.

The fast moving rates of the X-axis and Z-axis are set by the system parameters X-axis fast moving rate and Z-axis fast moving rate. The real-time adjustment can be made using the rapid override adjusting key. The actual fast moving override is 25%, 50%, or 100% of the fast moving rate settings.

Fast override power-off memory is available, and the initial override is 100% when power is on.

2.4.2. Cutting feed instruction F

Cutting feed: The system controls the movements in both X and Z axes at the same time, so that the tool's trajectory is consistent with the trajectory (linear, arc) defined by the instruction, and the instantaneous speed in the tangential direction of the trajectory is consistent with the F instruction value. This kind of motion control process is called cutting feed or interpolation. The cutting feed speed is specified by the F instruction word. When the system executes the interpolation instruction (cutting feed), the cutting feed speed given by the F instruction is decomposed into two directions of X-axis and Z-axis according to the programmed trajectory. The system controls the instantaneous speed in both X-axis and the Z-axis direction simultaneously so that the vector resultant velocity of the speeds in both directions is equal to the F instruction value.

Remarks: The feed override key or external override switch on the machine tool panel can be used to adjust the cutting feed rate in real time. The actual cutting feed rate can be adjusted in 16 steps within the range of 0 to 150% of the instruction speed (each step changes by 10 %). Feed override adjustment is invalid for thread cutting.

$$F=\sqrt{f_X^2+f_Z^2}$$

$$f_X=\frac{dx}{dt}$$

$$f_Z=\frac{dz}{dt}$$

F is the vector resultant velocity of the instantaneous speed in both the X-axis and the Z-axis directions;

dx is the instantaneous (dt time period) increment of the X-axis, and fx is the instantaneous speed of the X-axis;

dz is the instantaneous (dt time period) increment of the Z-axis, and fz is the instantaneous speed of the Z-axis.

For example, in Fig. 2-4-1, the coordinates of each point are in the brackets (the diameter in the X direction).

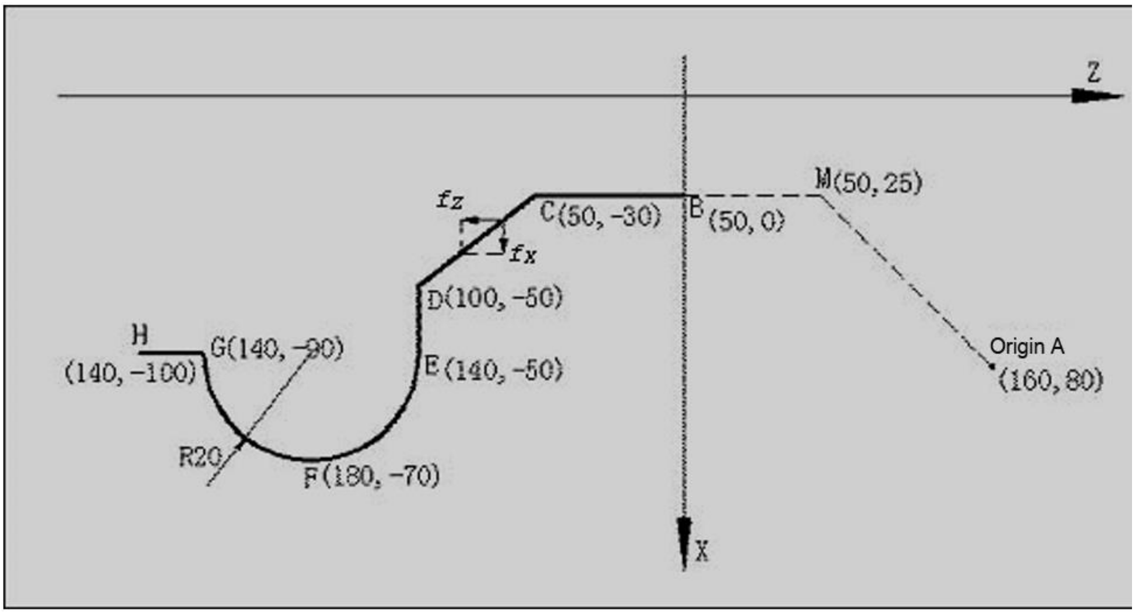


Fig. 2-4-1

The program is as follows:

O0010;	
G00 X160 Z80;	(First move the machine tool to a safe position)
G98;	
G0 X50 Z0;	(Move quickly from point A to point B via point M)
G1 W-30 F250;	(B → C)
X100 W-20;	(C → D)
X140;	(D → E)
G2 W-40 R20;	(EFG circular interpolation)
W-10;	(G → H)
M30;	
%	

2.4.3. G98, G99

Format: G98 Fxxxx; (F0001～F8000, leading zero can be omitted, the feed rate per minute is given, in mm/min)
Function: The cutting feed speed is given in millimeters/minutes; G98 is a modal G instruction. If the current modal is G98, G98 can be omitted.
Format: G99 Fxxxx; (F0.0001～F500, leading zeros can be omitted)
Function: The cutting feed speed is given in mm/rev; G99 is a modal G instruction. If the current modal is G99, G99 can then be omitted. When the system executes G99 Fxxxx, the product of the F instruction value (mm/rev) and the current spindle speed (rev/min) is used as the instruction feed speed, to control the actual cutting feed speed. When the spindle speed changes, the actual cutting feed speed also changes. Set the cutting feed per revolution of the spindle using G99 Fxxxx, then a uniform cutting pattern can be formed on the work-piece surface. When machining in G99 modal, the machine tool must be installed with a spindle encoder, and the number of spindle encoder lines must be set.
G98 and G99 are modal G instructions of the same group, and only one of them can be valid at a time. G98 is the initial G instruction. By default, G98 is valid when the system is powered on.

2.4.4. Manual feed

Manual feed: In the manual mode, the system can move in the positive or negative direction of the X or Z axis at the current manual feed speed. The

X-axis and Z-axis directions cannot move at the same time. The actual manual rate in the X-axis and the Z-axis directions can be adjusted in real time by the manual override adjusting key corresponding to 10% to 150%. The manual speed of each gear is the product of a fixed value set by the system parameters and the manual override, and the manual speed of each axis is set by parameters independently. When the manual rapid feed key is valid, the manual speed uses the rapid feed rate of each axis, and the manual override is invalid at this time.

Manual feed override is not stored in case of power off. The initial override is 100% when powered on.

2.5. Tool compensation (T instruction)

The system allows automatic tool change, in which tools can be changed during machining for the auto tool holders with 4 to 8 tool positions, so that the multiple processes and parts of multiple tools can be realized.

The system is also designed with a tool length compensation. No consideration shall be given to the actual position of the tool when writing the program. It only needs to obtain the position offset data of each tool (the "tool offset") by tool operation before machining. Before using the tool for machining, execute the tool length compensation, that is, offset the system's coordinates based on the tool offset to enable the motion curve of the tool tip consistent with the programmed trajectory. After changing the tool, the tool offset can be modified only by tool setting, with no need to modify the machining program. Where there's a deviation in the machining size due to tool wear, the tool offset can be directly modified on the basis of the deviation in size, so as to eliminate the deviation in machining size.

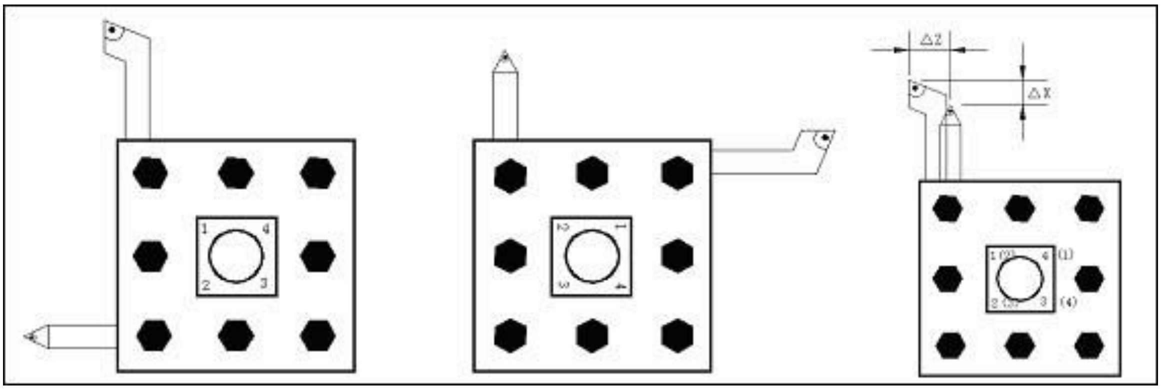


Fig. 2-5-1

Format:

T ○○ ○○

Tool offset number (00-16, leading 0 cannot be omitted)
Target tool number (00-08, leading 0 cannot be omitted)

Function: Automatic tool holders make tool change to the position of the target tool number, and execute the tool length compensation according to the tool offset corresponded to the tool offset number of the instruction. The tool offset number can be the same as or different from the tool number, that is, a tool can correspond to multiple offset numbers. The tool offset corresponded to the tool offset number of 00 is X = 0, Z = 0, and the system is in the state without tool compensation, that is, the coordinate offset of the system is 0 (the coordinate offset is not performed). After executing the tool length compensation, execute T □□ 00, and the system will offset the system coordinates in reverse based on the current tool offset; the system at this time changes from the executed tool length compensation state to the uncompensated state, and the displayed tool offset number is 00. This process is called canceling tool length compensation, referred to as Cancel Tool Compensation.

For example, T0101 means to select Cutter 1 and execute Cutter 1 offset

T0102 means to select Cutter 1 and execute Cutter 2 offset;

T0301 means to select Cutter 3 and execute Cutter 1 offset;

When being powered on, the tool number and offset number displayed by the T instruction are the states before the power failure.

Only one T instruction is valid in a segment. When two or more T instructions appear in a segment, the last T instruction is valid.

When the T instruction is in the same segment as the instruction to execute the movement, the instruction of tool change is executed before the move instruction.

This system is applicable to automatic tool holders with 4--8 tool locations. By modifying the Magazine 2 parameters and modifying parameters 17, 18, 19, and 20 according to the tool holder requirements, the functions such as direct tool location signal input (signal of each tool location is independent), the tool change by holder forward, and in-position reverse and lock are available. The control of the tool holder is realized by the built-in T code macro program. The parameter 023 is set to TFUNC (L) by default. By changing such parameter to User-Def, the user can then define the T code.

In manual mode, pressing the tool change button on the machine tool panel, you can then perform manual tool change.

The tool setting of system is under the coordinate menu, which is divided into tool setting and compensation. When setting the tool, you only need to enter the measured X-axis diameter and Z-axis length after trial cutting on the workpiece. The system will calculate the offset and stored it in the parameters automatically. For example, after using Cutter 1 for trial cutting, in the tool setting screen, enter the trial cutting diameter of X under entry mode, and enter the trial cutting length of Z to the No.1 parameter, and then call T0101. The compensation amount is the offset to the current compensation number.

3. G instruction

3.1. Summary

G instruction consists of the instruction address G and the 1 or 2 digit(s) instruction values followed by it. It is used to specify the actions such as the interpolation mode of the tool in relative to the workpiece, the pause function, and coordinate settings, etc.

G □□

Instruction value (00 ～ 99, and the first digit 0 can be omitted)

Instruction address G

G instruction words are grouped into 00, 01, 02, 03 and 04. Several G instruction words of different groups can be entered in a same segment. If more than two G instruction words of the same group are entered in the same segment, the last G instruction word is valid. Different groups of G instruction words without common parameters (instruction words) can be in a same segment with effect functions simultaneously, and there's no connection with the sequence. The system does not support G instruction words other than this table, otherwise an alarm will be given.

3.1.1. Modal, non-modal and initial states

Modal function refers to a function that once a code is specified in the current segment, it will remain valid until another code of the same group appears in such segment. No specification is required if the next segment continues to use such instruction.

Non-modal Function refers to that a certain code works only in the segment where it is located. If the next segment uses the instruction again, it must be specified again.

The modal G instruction word that is valid without executing its function or state after system power-on is called the initial G instruction word. When the initial G instruction word is executed after power-on, it is not necessary to input the G instruction word. G01 can be the initial state instruction word of this system.

3.1.2. Definitions

In this Manual, following terms can have the meanings given below:

Start point: the position where the current segment is running;

End point: the position after current segment is executed;

X: the absolute coordinate in the X direction of the end point;

U: the difference between the end point and the absolute coordinates of start point in X direction;

Z: the absolute coordinate in the Z direction of the end point;

W: the difference between the end point and the absolute coordinates of start point in Z direction;

F: the cutting feed rate.

List of G instruction words

Instruction words	Group	Function(s)	Remark
G00	01	Rapid Movement	Initial-state G instruction
G01		Linear interpolation	Modal G instruction
G02		Circular interpolation (clockwise)	
G03		Circular interpolation (counterclockwise)	
G32		Thread cutting	
G90		Axial cutting cycle	
G33		Z-axis tapping cycle	
G92		Thread cutting cycle	
G94		Radial cutting cycle	
G04	00	Pause and orientation	Non-modal G instruction
G31		Skip instruction	
G27		Reference point echo check instruction	
G28		Return to mechanical zero	
G29		Return instruction from reference point	
G30		Return to 2nd reference point instruction	
G50		Coordinate system setting	
G70		Finish machining cycle	

G71		Axial rough turning cycle	
G72		Radial rough turning cycle	
G73		Closed cutting cycle	
G74		Axial grooving multiple cycles	
G75		Radial grooving multiple cycles	
G96	02	Constant linear velocity ON	Modal G instruction
G97		Constant linear velocity OFF	Initial-state G instruction
G98	03	Feed per minute	Initial-state G instruction
G99		Feed per revolution	Modal G instruction

3.2. Interpolation

3.2.1. Fast Movement G00

Format: G00 X (U) Z (W);

Function: X-axis and Z-axis move from the start point to the end point simultaneously at the same moving speed. G00 is the initial G instruction.

The two axes move at the independent speeds. The X-axis and Z-axis fast moving speed are set by the parameters 009 and 011. Please ensure that the synthetic trajectory may not be a straight line, and the two axes may not reach the end point at the same time.

The instruction addresses X (U) and Z (W) may be partly or completely omitted. When one of them is omitted, the coordinate values of the start point and the end point of the axis are the same; when both the two are omitted, the end point and the start point are at the same position.

The fast movement speeds of X-axis and Z-axis are set respectively by the fast movement speeds of both the X-axis and the Z-axis. The actual movement speed can be adjusted by pressing the rapid override key on the panel of machine tool.

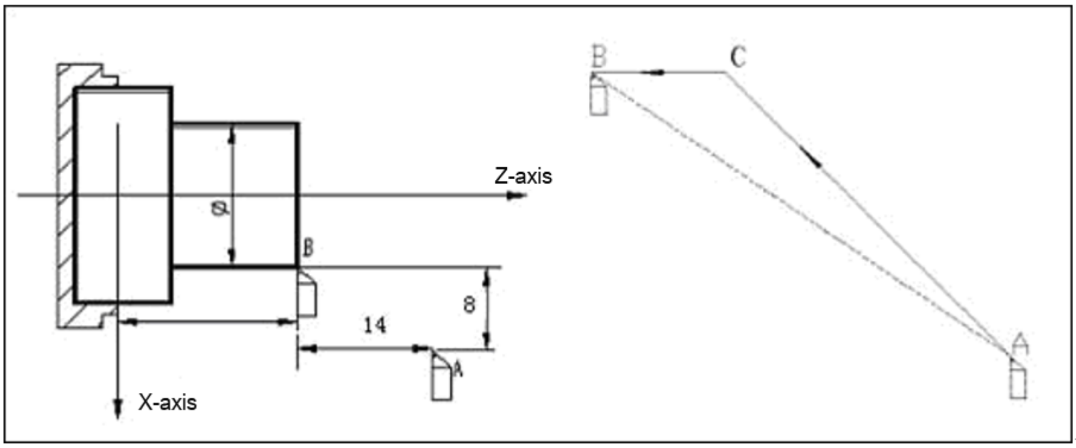
The actual maximum speed that a machine tool can achieve depends on its actual condition as well as the supporting motor. For detailed parameters, refer to the Instruction Manual provided by the manufacturer.

G00 is a modal instruction, which can be omitted when the next segment is the same.

G00 can be abbreviated as G0, which is equivalent to G00.

When the X-axis and Z-axis are instructed to move at the same time, pay special attention to whether the position of the tool is in the safe area. In this way, the tool collision may be avoided.

Example: The tool moves quickly from Point A to Point B.



G0 X20 Z25; (Point A coordinate)
G0 U-8 W-14; (A → B)

3.2.2. Linear interpolation G01

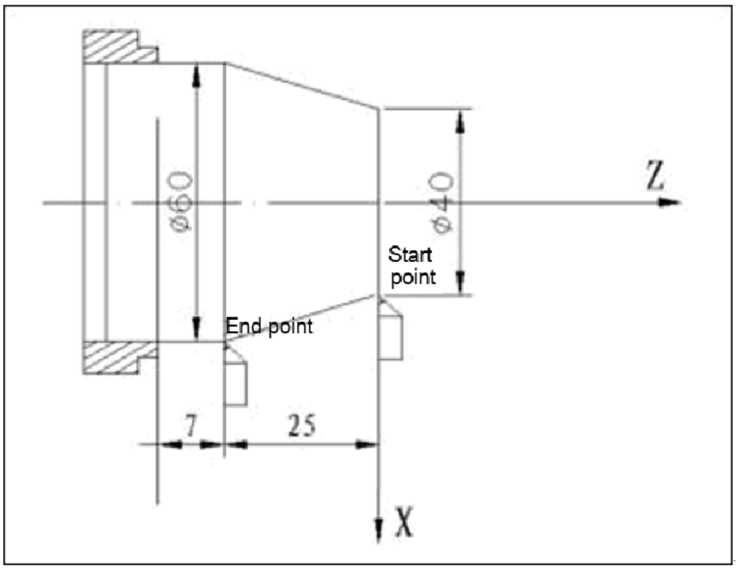
Format: G01 X (U) _ Z (W) _F_;
X (U)/Z (W): The absolute or relative coordinate values of the end point.
F : The cutting feed speed

Function: G01 instruction enables the tools to reach the position of specified point simultaneously along with the connection of the specified point of X (U) and Z (W) at the set speed. The motion trajectory can be a straight line from the start point to the end point. The F instruction value is the vector resultant speed of the instantaneous speed in the X-axis and the Z-axis directions. The actual cutting feed rate is the product of the feed override and the F instruction value. Once the F instruction value is executed, it remains unchanged until a new F instruction value is executed.

Examples:

G01 X60.0 Z-25 F200; (absolute value programming)

G01 U20.0 W-25.0; (relative value programming)



3.2.3. Circular interpolation G03, G02

Format: G03/G02 X (U) _ Z (W) _ R_ (I_ K_) F_;

Function: The motion trajectory is a clockwise/counterclockwise arc running from the start point to the end point. The trajectory is shown in the figure below.

R: The circular arc radius (0 ~ 9999.999mm);

I: The difference between the center of the circle and the X-axis coordinate of start point of the circular arc (-9999.999 ~ 9999.999 mm);

K: The difference between the coordinates of the center of the circle and the Z-axis coordinate of start point of the circular arc (-9999.999 ~ 9999.999 mm);

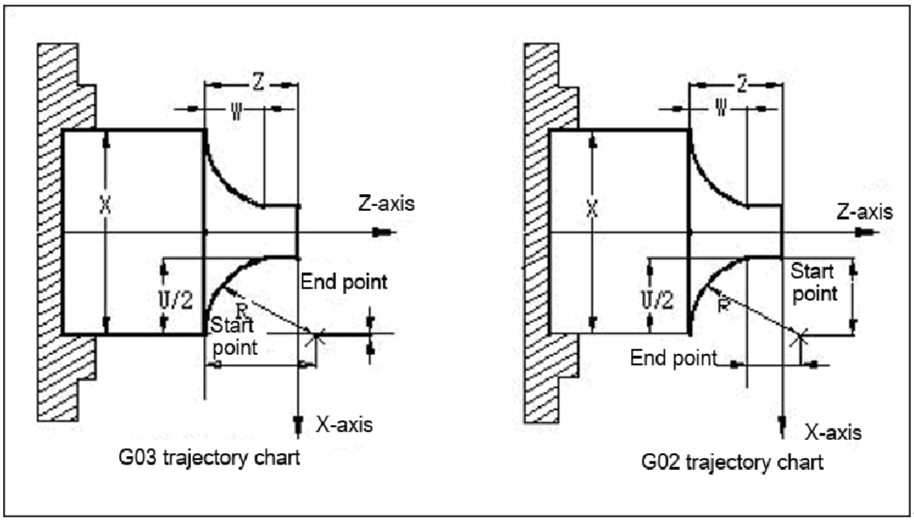


Fig. 3-2-3-1

Example: as shown in Fig. 3-2-3-2

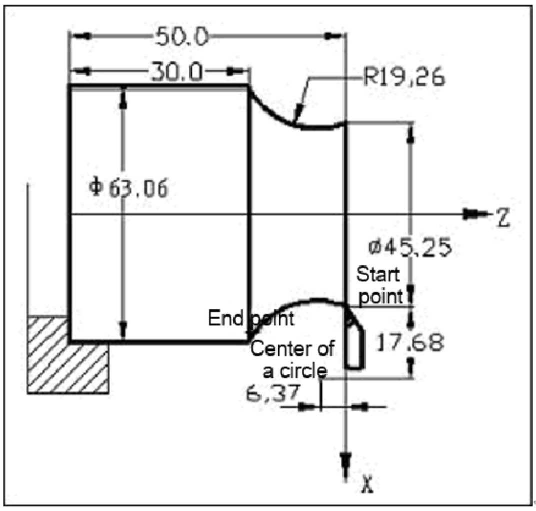


Fig. 3-2-3-2

Program:

G02 X63.06 Z-20.0 R19.26 F300;

Or G02 U17.81 W-20.0 R19.26 F300;

Or G02 X63.06 Z-20.0 I35.36 K-6.37 F300;

Or G02 U17.81 W-20.0 I35.36 K-6.37 F300;

Notes:

- At least one of the instruction addresses I, K, or R must be input in the G02/G03 segments, otherwise the system will produce an alarm; when I, K, and R are input at the same time, R is valid, but I and K are invalid; when no R is input or the value of R equals to zero, an alarm will be given by the system;
- If X (U) and Z (W) are not input, and if executing the G02/G03 using the given radius R, the X-axis and Z-axis will not move. If R is not

input and I or K instruction words are used, the trajectory executing G02/G03 instruction can be a full circle (360°);

- When R instruction is used, it theoretically may be two arcs greater than 180 ° and less than 180°. The arc less than or equal to 180 ° may be valid in this system. (See Fig. 3-4-3 below). If the end point is not on an arc defined by R, the system will give an alarm;

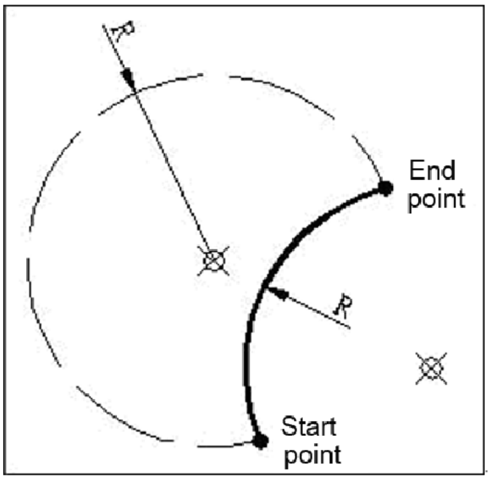


Fig. 3-4-3

- When I and K instruction words are used to define the circle center in the G02/G03 segment, the system will not give an alarm even if the end point is not on the arc. The trajectory of executing the instruction can be: the X-axis and Z-axis move simultaneously along the arc from the start point as per the circle center and arc direction defined by the instruction; the X-axis or Z-axis stops moving when the coordinates of the X-axis or Z-axis are the same as that of the end point, while the other axis (Z- or X-axis) continues to move to the end point. As shown in Fig. 3-4-4

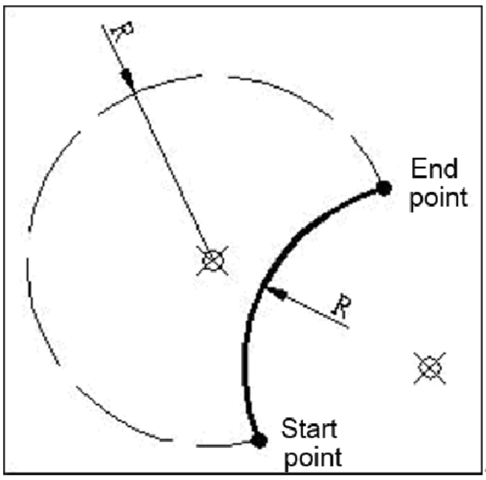


Fig. 3-4-4

3.2.4. Pause Instruction G04

Format: G04 P__;
or G04 X__ ;

Function: Each axis stops to move, but the current G instruction modal and the data and state maintained are not changed; they execute the next segment after a given time delay. The delay time is specified by the instruction word P__ (in ms) and X__ (in sec). G04 is a non-modal G instruction.

G04 P1000 1000 ms of delay = 1 sec

G04 X1 Delay 1 sec

3.2.5. Return to Mechanical Zero G28

Format: G28 X(U)_ Z(W)_;

Function: Reach the intermediate point specified by X (U) and Z (W) at fast moving speed starting from the start point, and then return to the mechanical zero simultaneously. G28 is a non-modal G instruction.

One of or both the instruction addresses X (U) and Z (W) can be omitted. See the table below for details

Instructi on	Function (s)
G28 X(U)_	X-axis returns to mechanical zero, while Z-axis stays at home position
G28 Z(W)_	Z-axis returns to mechanical zero, while X-axis stays at home position
G28	When both axes stay at home position, continue to execute the next segment
G28 X(U) Z(W)	X-axis and Z-axis return to mechanical zero at the same time

X: The absolute coordinate in the X direction of the intermediate point; U: The difference between the absolute coordinate in the X direction of the intermediate point and the start point;

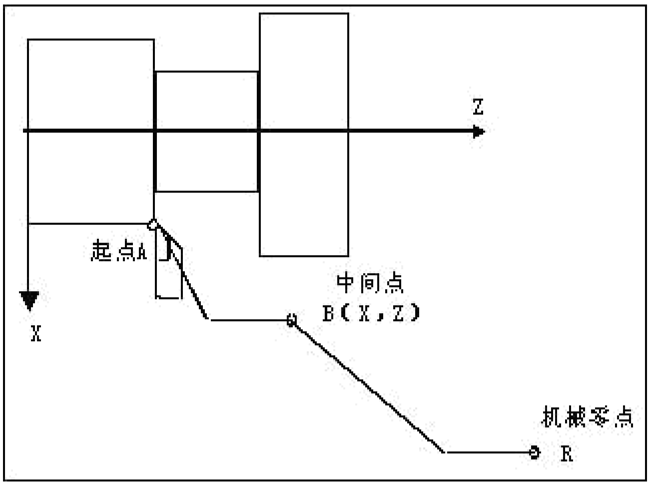
Z: The absolute coordinate of the intermediate point in the Z direction; W: The difference between the absolute coordinate in the Z direction of the intermediate point and the start point.

Action process of instruction (see the figure below):

(1) Move from the start point to the intermediate point (Point A → B) simultaneously with the independent fast moving speed.

(2) After the two axes have reached the intermediate point, move from the intermediate point to the mechanical zero point (Point B → R) simultaneously with the independent fast moving speed.

(3) If U and W are 0, it will directly reach the mechanical zero. For example, in case of G28 U0 W0, the system will directly return to the reference point without passing the intermediate point (Point A → R).



3.3. Thread cutting

3.3.1. Thread Cutting Instruction G32

Format: G32 X (U) __ Z (W) __ F (I) __ P__D__V__;

X (U): End-point coordinate value or incremental value of thread X-axis

Z (W): End-point coordinate value or incremental value of thread Z-axis

F: Metric thread pitch (0.001 ~ 500 mm), which is the movement of the major axis after spindle rotating a circle. The F instruction value remains valid after it is executed until the F instruction word with the given thread pitch is executed again.

I: The threads per inch of thread;

P: Thread cutting acceleration pitch, in mm, unsigned number, is the accelerating length value of the major axis during thread cutting. If the P value is not specified in the programming, the system automatically calls parameter 109 for calculation. When the X-axis is of the major axis, P shall be the radius value.

D: Thread cutting slowing-down pitch, in mm, unsigned number, is the slowing-down length value of the major axis during thread cutting. If the D value is not specified in the programming, the system automatically calls the parameter 110 for calculation. When the X-axis is of the major axis, D shall be the radius value.

V: The tail withdrawal of pitch cutting, in mm, signed number, and the plus-minus sign is related to the retract direction. If the V value is not specified in the programming for the tail withdrawal length value of the minor axis during pitch cutting, the system automatically calls the parameter 111 for calculation. When the X-axis is of the minor axis, V shall be the radius value.

Function: The tool's motion trajectory is a straight line moving from the start point to the end point. The coordinate axis with the larger displacement (radius value for X-axis) from the start point to the end point is called the major axis, and the other axis is called the minor axis. The spindle moves one pitch per revolution during the movement, and the minor axis performs linear interpolation with the major axis. When the tool cuts the workpiece, a helical groove with a constant pitch is formed on the surface of the workpiece, to allow the constant pitch threading. F instruction word is used to calculate the pitch of the metric thread. After executing the G32 instruction, you can process the straight threads, taper threads and end surface threads of the metric pitch:

When the X coordinate values of the start and end points differ (no X or U is input), straight thread cutting is done;

When the Z coordinate values of the start point and the end point are the same (no Z or W is input), end surface thread cutting is done;

If the X and Z coordinate values of the start and end points differ, taper thread cutting is to be done. See Fig. 3-7-1 below

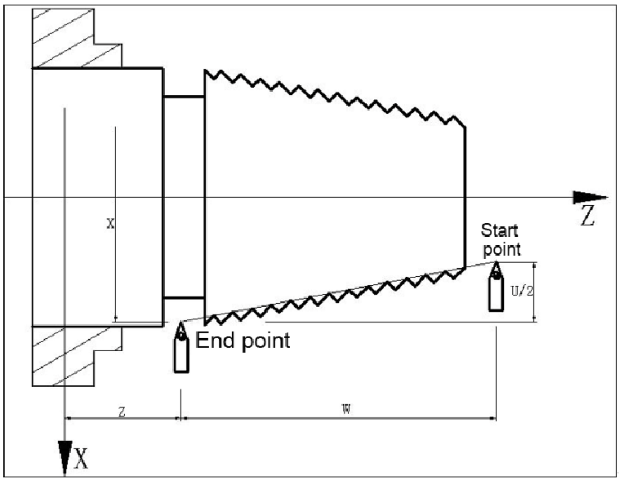


Fig. 3-7-1

The pitch of the thread refers to the amount of displacement upon one revolution of the spindle (the amount of displacement of the X-axis is based on its radius). The method of judging the major and minor axes is shown in Fig. 3-7-2 below:

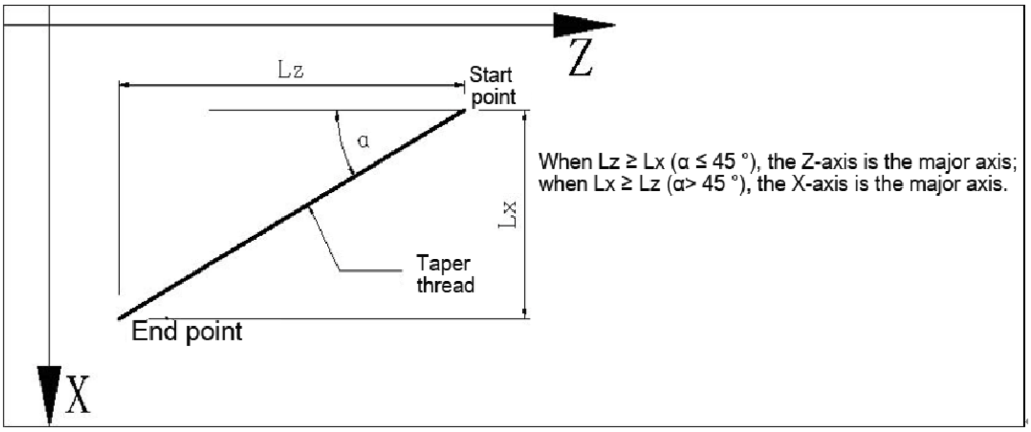


Fig. 3-7-2

Precautions for thread cutting:

- Thread cutting can only be performed when the machine tool is provided with a spindle encoder. The transmission ratio between the spindle encoder and the spindle should be 1: 1, while the number of encoder lines of spindle 013 should be modified. The system is initialized to 0. After receiving the signal from the spindle encoder Z (origin), the system moves the X-axis or Z-axis for thread machining. Therefore, as long as the spindle speed is not changed, the same thread can be machined by roughing and finishing.
- Since there're acceleration and deceleration processes for X-axis and Z-axis at the beginning and end of thread cutting, the pitch error at this time is relatively large. Therefore, in actual thread cutting, there must be an acceleration pitch P before the start point of the major axis, and deceleration pitch D after end point of the major axis (commonly known as "tool escape"), which means that the programmed thread length is longer than the actual thread length. The cutting length of the thread includes P and D. Similarly, there shall be a backlash V at the end position of the minor axis of the thread cutting, that is, the minor axis exits the thread cutting by the tail withdrawal of V at the end position of the thread cutting.

Examples:

Thread pitch: 4 mm. P = 3.5 mm, D = 3.5 mm, V = 1; the total cutting depth is 1 mm (single side), and cut in two times.

G00 X28 Z3; (Cut into 0.5mm for the first time)

G32 X51 W-77 F4.0 P3.5 D3.5 V1; (The first cutting of taper thread)

G00 X55; (Tool withdrawal)

W77; (Back to the start point in Z direction)

X27; (Advance 0.5mm in the 2nd time

G32 X50 W-77 F4.0; (Second cutting of taper thread)

G00 X55; (Tool withdrawal)

W77; (Back to the start point in Z direction)

Fig. 3-7-3:

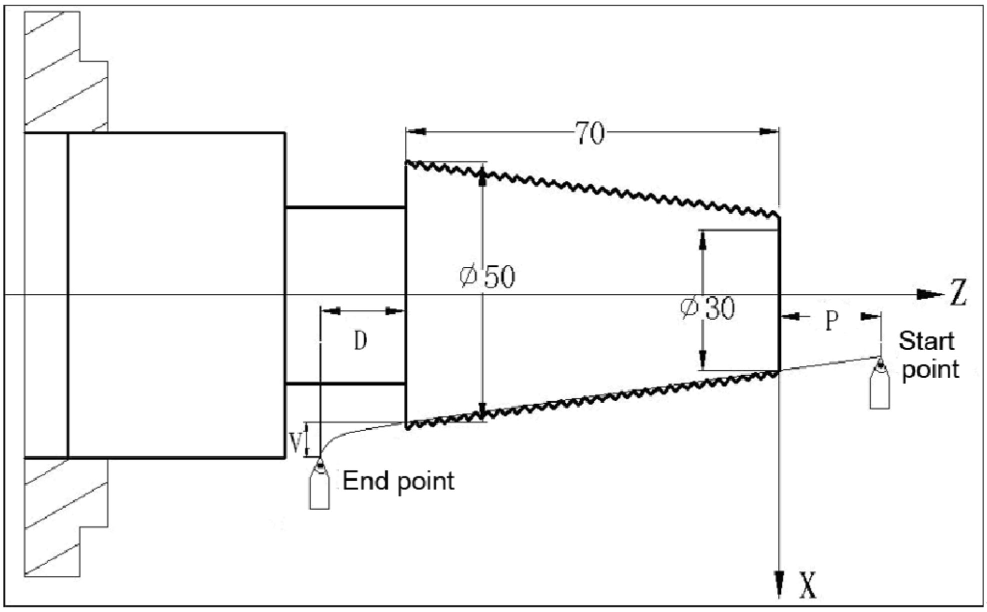


Fig. 3-7-3

- Where the start point, end point and thread pitch of G32 are determined, the movement speed of X-axis and Z-axis during thread cutting is determined by the spindle speed, which is independent of the cutting feed rate override. The spindle override control is effective during thread cutting. When the spindle speed changes, the pitch error may increase arising from the acceleration and deceleration of the X-axis and Z-axis. Therefore, be sure not to adjust the spindle speed during thread cutting, and stop the spindle (otherwise the tool and workpiece may be damaged).
- During thread cutting, the increment of the major axis should be greater than the value of (P + D), and the actual thread length of the major axis should be greater than the pitch value F.
- Thread cutting stops immediately during system reset, emergency stop or drive alarm.

3.3.2. Z-axis Tapping Cycle G33

Format: G33 Z (W) __ F__;

Function: The tool moves from the start point to the end point, and then return to the start point from the end point. Z-axis moves a pitch for each revolution of spindle during movement, which is always consistent with the pitch of the tap. In this way, a helical cutting is formed in the inner hole of the workpiece, and the thread machining of the inner hole can be done in one cut.

Instruction description: G33 is a modal G instruction;

Z (W): When Z or W is not input, the Z coordinate values of the start and end points are the same, and no thread cutting will be made;

F: Metric thread pitch, ranging from 0.001 to 500 mm;

Cyclic process:

① Z-axis feeding and tapping (the spindle must be turned on before G33 instruction);

② After reaching the end point of the Z-axis coordinate specified by programming, M05 outputs signal;

③ After the spindle is detected to be fully stopped;

④ Spindle reverses the signal output (opposite to the original spindle rotation direction);

⑤ Z-axis retracts to the start point;

⑥ The spindle restores to the direction before G33.

Program:

```
O0011;
G00 Z90 X0Z0 M03; Start the spindle
G33 -Z50 F1.5; Tapping cycle
G00 X60 Z10; Continue machining
M30
```

Note 1: Before tapping, the rotating direction of spindle should be determined as per the rotation direction of the tap. Upon tapping, the spindle will return to its original direction.

Note 2: This instruction is of rigid tapping. After the spindle stop signal is valid, the spindle will still stop rotating after being given a certain deceleration time. At this time, the Z-axis will still feed with the rotation of the spindle until the spindle is fully stopped. Therefore, the position of the bottom hole of the thread during machining should be slightly deeper than the actual required position, and the specific length beyond is determined by the spindle speed as well as the spindle braking device during tapping.

Note 3: The Z-axis movement speed during tap cutting is determined by the spindle speed and pitch, but not the cutting feed rate override.

Note 4: During single-segment operation or execution of feed hold, the system displays "Pause", but the tapping cycle continues until tapping is finished and returned to the start point.

Note 5: The tap cutting will decelerate and stop during system reset, emergency stop or alarm drive.

3.4. Work-piece Coordinate System Setting G50

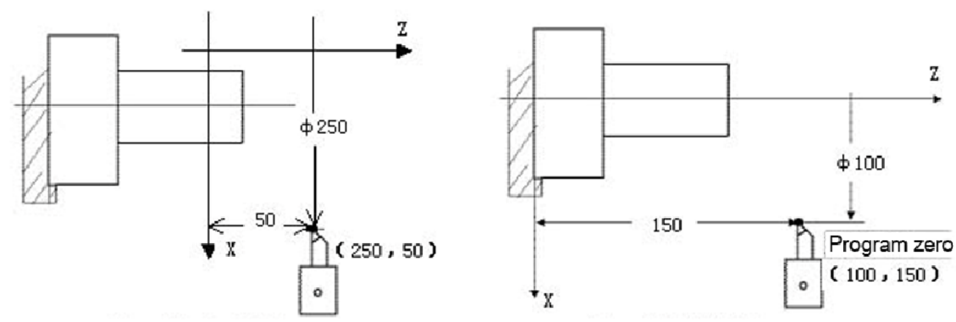
Format: G50 X_ Z_;

X: The new absolute X-axis coordinates at the current position;

Z: The new absolute Z-axis coordinate at the current position;

Function: Set the absolute coordinates of the current position, and establish the work-piece coordinate system (also the "floating coordinate system") in the system by setting the absolute coordinates of the current position. After executing this instruction, the system uses the current position as the program zero. After the work-piece coordinate system is established, the absolute coordinate programming enters the coordinate values according to this coordinate system, until G50 is executed again to create new work-piece coordinate system. G50 is a non-modal G instruction.

X or Z is not input in the G50 instruction. The coordinate axis that is not input is set according to the current absolute coordinate value. When neither X nor Z is input, the current coordinate value is not changed. So long as G50 is executed, the current position is set to the program zero.



Before setting the coordinate system with G50 After setting the coordinate system with G50

As shown in the above figure, after the instruction segment “G50 X100 Z150” is executed, the work-piece coordinate system will be established and the point X100,Z150) considered the program origin.

Note: In the state of tool length offset, if G50 is executed to establish the coordinate system, the absolute coordinates shown in the system are set values modified in accordance with the current tool deviation, while the program origin represents the position determined by G50 in the work-piece coordinate system. When the tool returns to program origin in this state, the end of resetting is the position of program origin after the tool length offset is cancelled.

Examples:

Current tool compensation state	Displayed coordinates after G50 X20 Z20 is executed		Tool compensation value 01
T0100	X: 20	Z: 20	X: 12
T0101	X: 32	Z: 43	Z: 23

3.5. Fixed cycle

To simplify the programming, this system provides a G instruction with which the single-time machining cycle, including fast positioning, linear/thread cutting and fast returning to start point, can be completed by only one program segment.

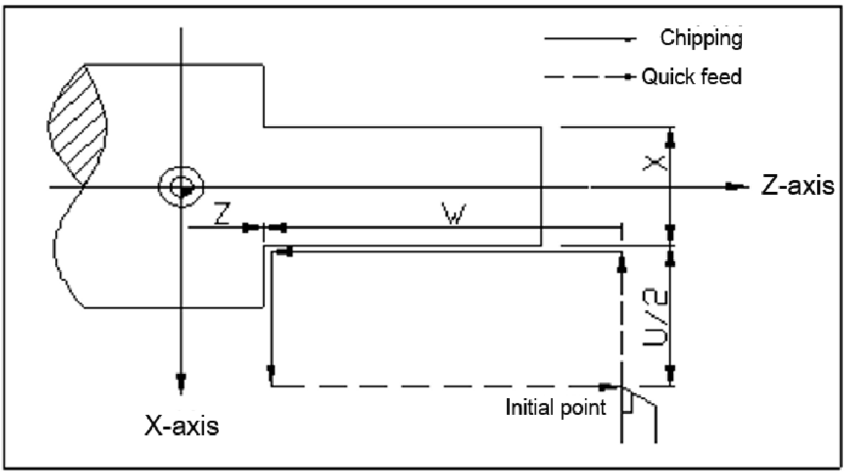
G90: Axial cutting cycle
G94: Radial cutting cycle
G92: Thread cutting cycle

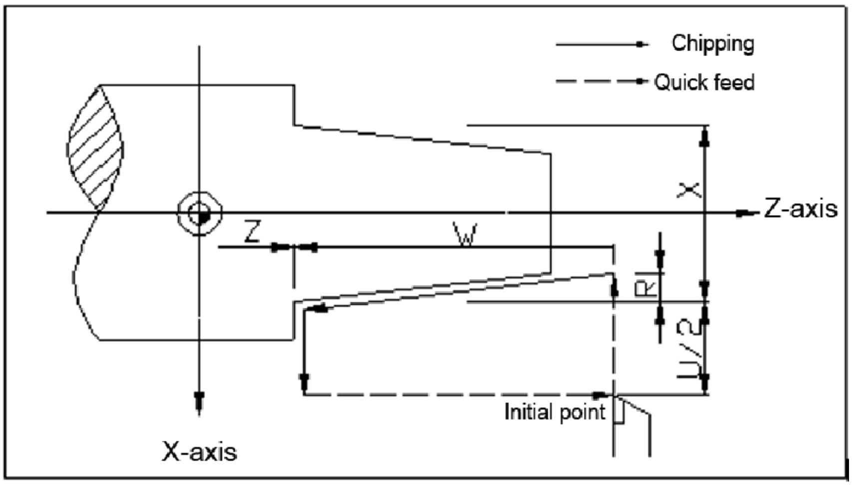
3.5.1. Axial cutting cycle (G90)

Format: G90 X/U_Z/W_R_F_;

X/U: Coordinate of end point for cutting on axis X;
Z/W: Coordinate of end point for cutting on axis Z;
F: Cutting speed

R: Conic surface’s inclination. This refers to the difference between the radial coordinate of the start point for cutting and that of end point (radius). When R is inconsistent with U, it should be as ｜R｜ ≦ ｜U / 2｜ (diameter programming) and ｜R｜ ≦ ｜U / 2｜ (radius programming). If R is not defined, the system will be engaged in the machining for a surface of a column, as shown the figure below:



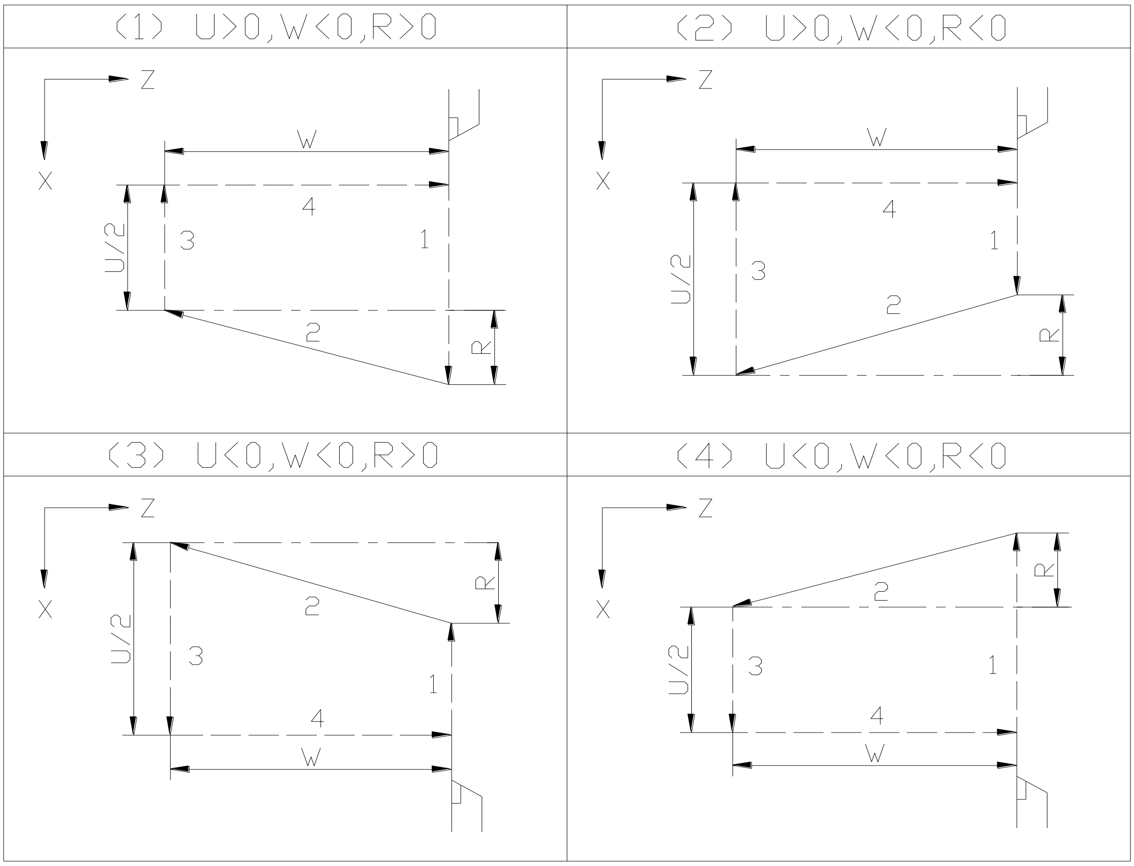


Execution process:

1) X-axis fast positions from start point to the start point of cutting (G0);
2) Linear interpolation (G1) from start point of cutting to end point;
3) The radial coordinates by which X-axis retracts to start point of the cycle in an interpolation way (G1)
4) Z-axis in fast positioning (G0) returns to start point and the cycle ends.

Notes:

1) G90 serves as a mode instruction;
2) In single segment operation, the system will stop at the end point of each program segment, and pause state and operation resumption become effective in the motion.
3) U, W and R show the relative positions of the end point and start point of cutting. Four combined loci are available for G90 depending on the symbols.



For diameter programming, when symbols of R and U are inconsistent with each other, if ｜R｜>｜U/2｜, the system will alarm as “0792-G90, G92 and G94 program error”, meaning R value has exceeded the permitted range.

4) At the time of first cutting, if U is 0, the system will default the current value as the coordinates of the end point. W can’t be 0. Otherwise, the system will alarm
5) Before the execution of G90, the operator should define the position of start point. Otherwise, the system will take the current point as the start point.
6) In MDI mode, G90 is effective. And it also performs the function of modal instruction.

See the example in Fig. 3-5-1:

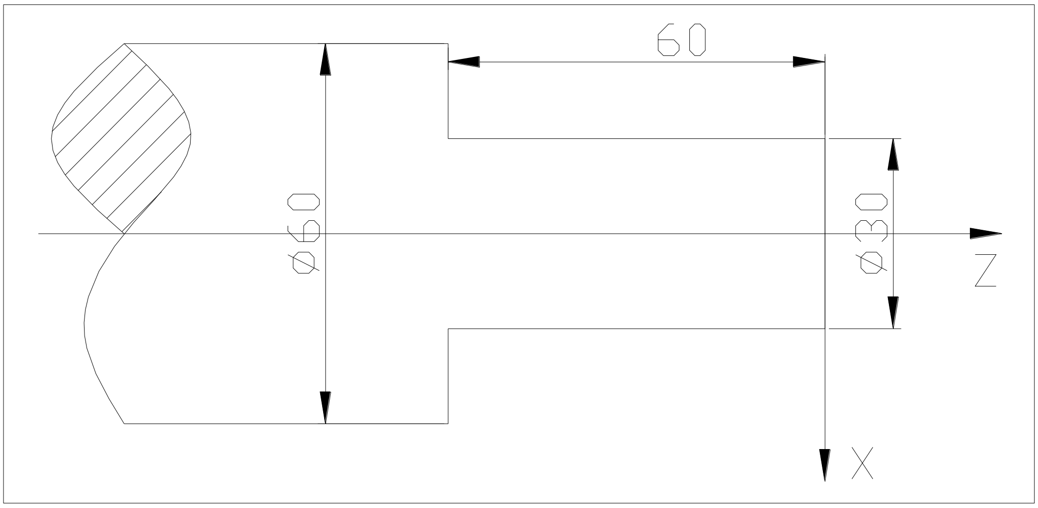


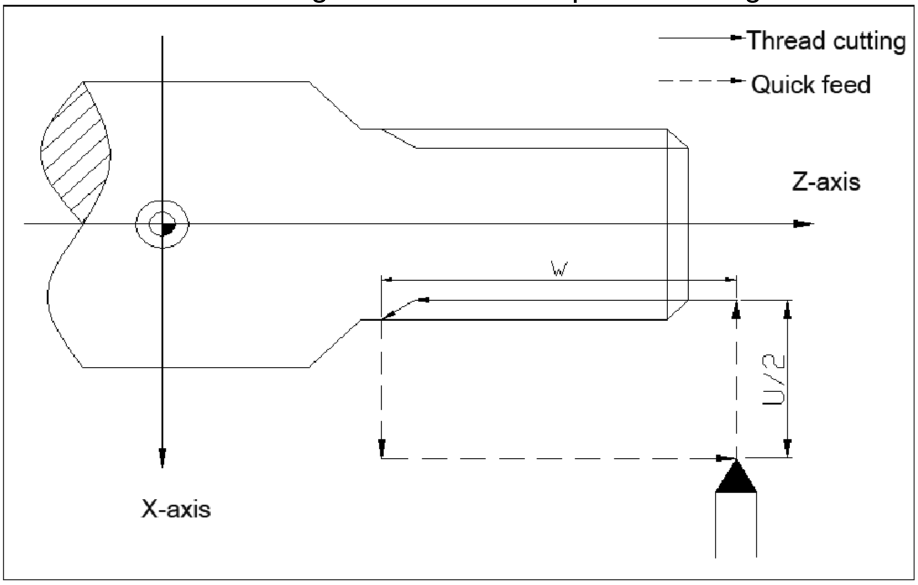
Fig. 3-5-1

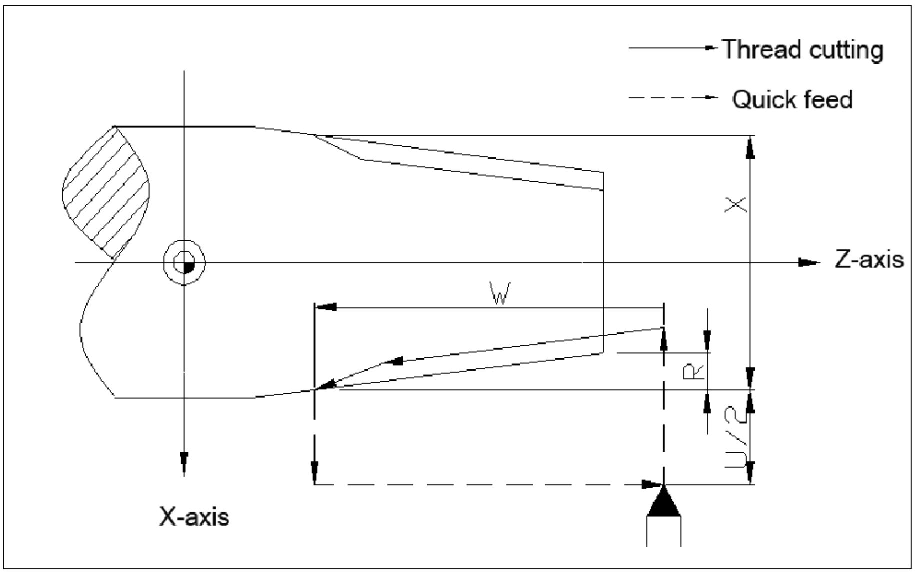
```
O9001
M03 S500T0101
G00 X70 Z2
G90X56Z-60F500
X52
X48
X44
X40
X36
X32
X30
M30
%
```

3.5.2. Thread Cutting Cycle G92

Format: G92 X/U_Z/W_F_ P__ D__ V__; (metric straight thread cutting cycle)
G92 X/U Z/W_I_ P__ D__ V__; (imperial straight thread cutting cycle)
G92 X/U_Z/W_R_ F_ P__ D__ V__; (metric taper thread cutting cycle)
G92 X/U_Z/W_R_ I_ P__ D__ V__; (imperial taper thread cutting cycle)

X/U: Coordinate of the thread's end point on X-axis;
Z/W: Coordinate of the thread's end point on Z-axis;
F: Imperial thread pitch, in mm;
I: The number of teeth per inch for inch screw thread thread; in Teeth/inch
R: Thread's inclination. This refers to the difference between the radial coordinate of the start point for cutting and that of end point (radius). When R is inconsistent with U, it should be as｜R｜ ≦ ｜U/2｜ (diameter programming) and ｜R｜ ≦ ｜U｜ (radius programming). If R is not defined, the system will be engaged in the machining for straight thread
P, D, V: same to G32 instruction function, but the value of V in G92 is unsigned; the system retreats the tool according to the direction of the start point automatically, and G32 needs to check the retreating direction of the specified V sign



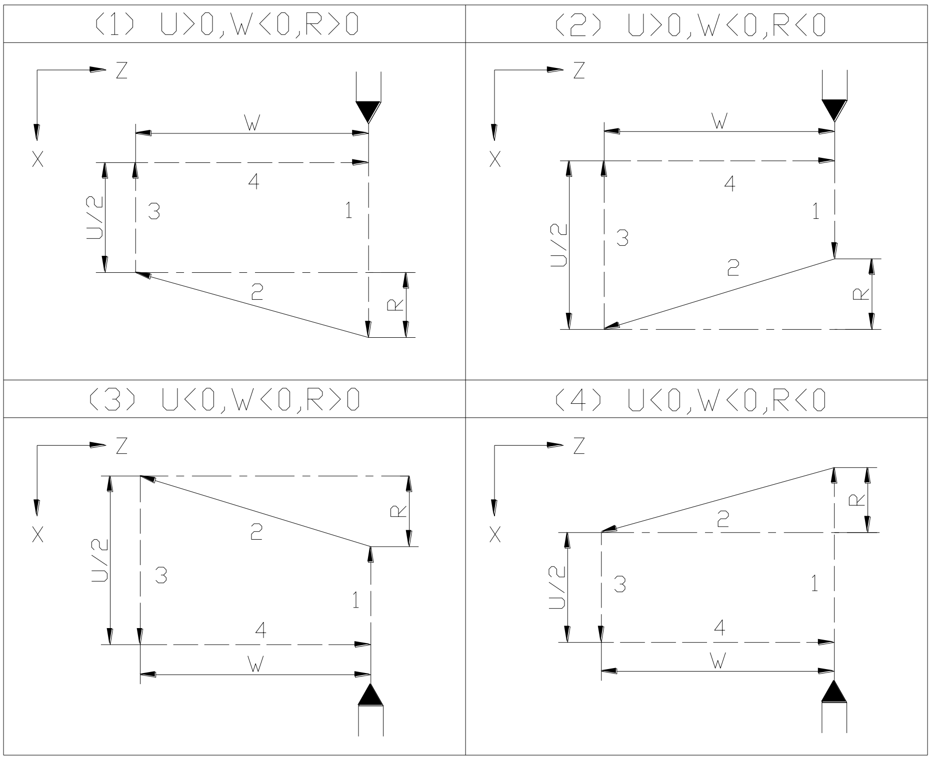


Execution process:

1) X-axis locates (G0) to cutting start quickly from cycle start;
2) Interpolate to thread end from cutting start;
3) X-axis radially gets back to radial coordinate position of cycle start in quick positioning (G0) mode;
4) Z-axis quickly locates (G0) and returns to the start point, and cycle ends.

Function description:

1) G92 instruction function is similar to G32 function, and only cycle cutting function is increased.
2) U, W and R show the relative positions of the end point and start point of cutting. G92 has four track combinations depending on the sign.



3) During first tool cutting, if U is 0, current value is the end coordinates by default; W can't be 0, or else the system alarms.
4) Spindle ratio control is invalid during thread cutting
5) If the converted feeding speed every minute exceeds the maximum feeding speed, the system alarms
6) G92 instruction can process one thread in several times of feeding, but the processing of two continuous threads can't be realized, nor process end thread. G92 instruction thread pitch has the same definition as G32, and the pitch is the displacement of the long axis that the spindle rotates one circle (X-axis displacement depends on the radius).

Precautions for G92 thread cutting:

- The machine tool must be installed with spindle encoder for thread cutting; the transmission ration of spindle encoder and the spindle is 1:1; the spindle encoder outputs A/B differential signal and Z signal (conversion signal) with 90° phase difference. When cutting thread, the system starts thread processing after receiving spindle encoder Z signal. Therefore, the processing of same thread can be finished through roughing and finishing of several connected G92 segments as long as the spindle rotation isn't changed.

- Since X-axis and Z-axis have acceleration and deceleration process in the beginning and at the end of thread cutting, the pitch error is large at this moment. The thread back function of G92 function can be used to process the thread without retreating groove.

- Under the condition that the start point, end point and thread pitch of G92 are fixed, the moving speed of X-axis and Z axis during thread cutting depends on the spindle rotation, and doesn't relate to cutting feed rate override. The spindle override control is effective during thread cutting. When the spindle speed changes, the pitch error may increase arising from the acceleration and deceleration of the X-axis and Z-axis. Therefore, be sure not to adjust the spindle speed during thread cutting, and stop the spindle (otherwise the tool and workpiece may be damaged).

- During thread cutting, and after feeding maintenance operation is executed, thread cutting doesn't stop.

- Thread cutting is invalid when single segment is running, and the running pauses after the first non-thread cutting in the thread cutting cycle is executed. Thread cutting stops immediately during system reset, emergency stop or drive alarm.

See the example in Fig. 3-10-7:

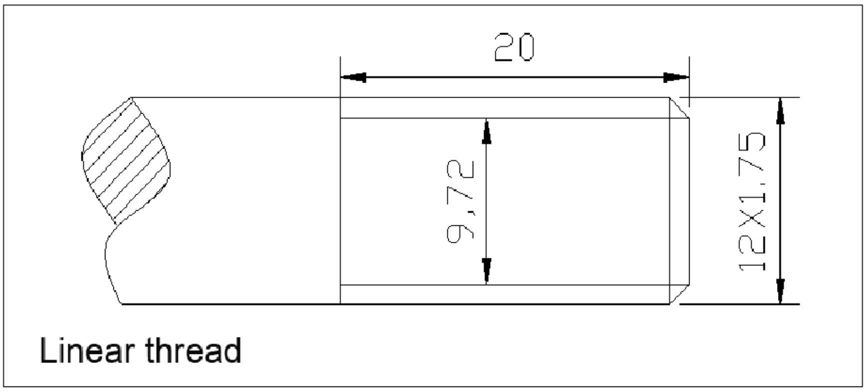


Fig. 3-10-7

Program: O3201

```
M03S500
G00X9.72Z2;
G32W-20 F1.75
G00X20
Z20
M30
```

3.5.3. Radial Cutting Cycle G94

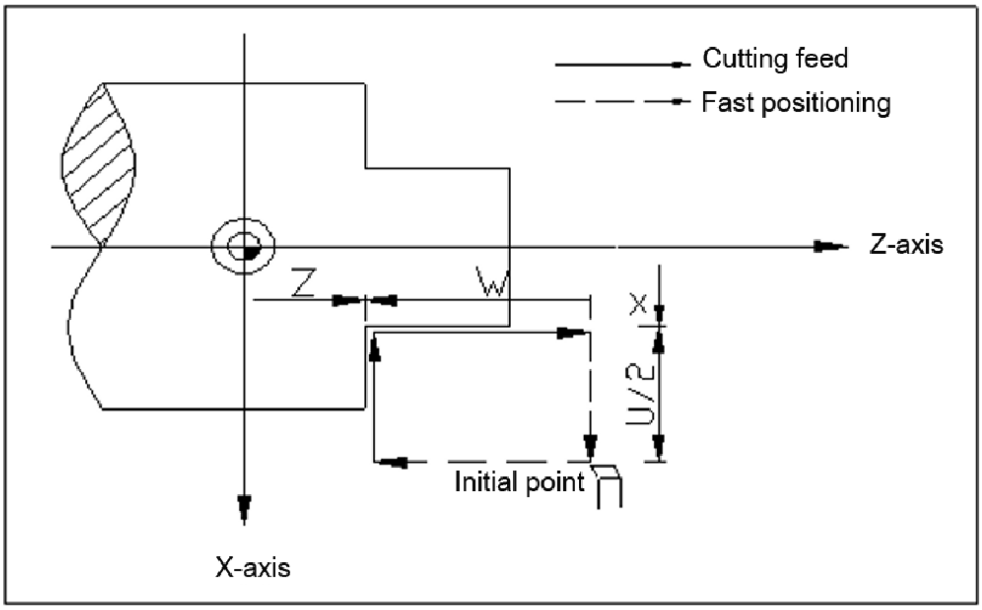
Format: G94 X/U Z / W R_ F_;

X/U: Coordinate of end point for cutting on axis X;

Z/W: Coordinate of end point for cutting on axis Z;

F: Cutting speed

R: Conic surface's inclination. This refers to the difference between the radial coordinate of the start point for cutting and that of end point. When R is inconsistent with W, it should be as ｜R｜≦｜W｜. If R is not defined, the system will be engaged in the machining for a straight end surface

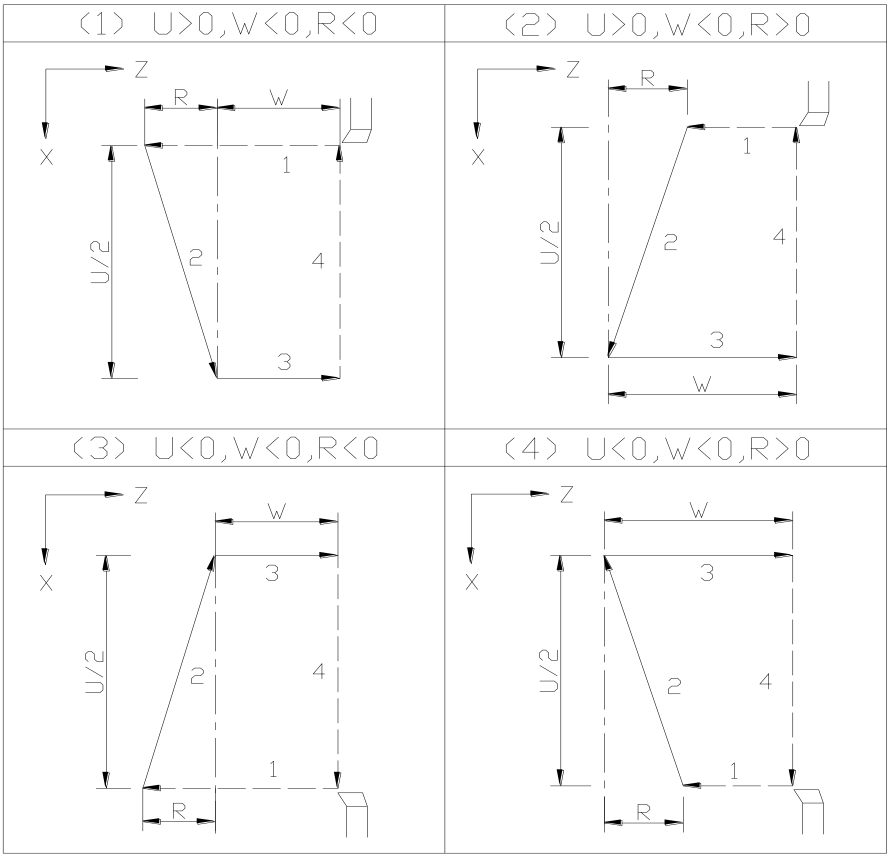


Execution process:

1) X-axis locates (G0) to cutting start quickly from cycle start;
2) Linear interpolation (G1) from start point of cutting to end point;
3) Z-axis axially gets back to axial coordinate position of cycle start in linear interpolation (G1) mode
4) Z-axis quickly locates (G0) and returns to the start point, and cycle ends.

Description:

1) G94 serves as a modal instruction.
2) In single segment operation, the system will stop at the end point of each program segment, and pause state and operation resumption become effective in the motion.
3) U, W and R show the relative positions of the end point and start point of cutting. Four combined trajectories are available for G94 depending on the symbols.



If R and W do not have same symbol and |R| > |W|, 0792-G90, G92, G94 program error alarms, and the value of R exceeds allowed range

4) At the time of first cutting, if W is 0, the system will default the current value as the coordinates of the end point. W can't be 0. Otherwise, the system will alarm
5) Before the execution of G94, the operator should define the position of start point. Otherwise, the system will take the current point as the start point.
6) In MDI mode, G94 is effective. And it also performs the function of modal instruction.

3.5.4. Precautions on fixed cycle instruction

1) In the fixed cycle instruction, once X(U), Z(W) and R are executed, the instruction values of X(U), Z(W) and R are always valid before new fixed cycle instruction to re-specify X(U), Z(W) and R. If G00, G01, G02, G03 and G32 are executed, the instruction value for retaining X(U), Z(W) and R will be deleted;
2) When the program segment next to G90, G92 or G94 program segment is of instruction characters that do not cause motion, if this segment is executed, the action of G90, G92 or G94 program segment will be performed again. To avoid such a situation, you must use other G instructions to cancel the cyclic operation after the fixed cycle.

   (For example)

```
M3; (start the spindle)
…
G90 X200.0 Z10.0 F2000;
X205.0 (repeat the execution of G90 once)
X206.0 Z20.0 (repeat the execution of G90 once)
…
```

3) If the fixed cycle instruction shares the segment with M, S and T instructions, the fixed cycle instruction can be executed simultaneously with them. But if the fixed cycle (because of G00 and G01) is cancelled after M, S and T instructions, the fixed cycle instruction must be executed again, like the example below:

   (For example)

```
N003 T0101;
…
N010 G90 X20.0 Z10.0 F2000;
N011 G00 T0202;
N012 G90 X20.5 Z10.0;
```

3.6. Multi-cycle instructions

G code	Functions	Format
G70	Finishing cycle G70	G70 P(ns) Q(nf);
G71	Axial roughing cycle G71	G71 U(Δd) R(e) F___ S_ T___ ; G71 P(ns) Q(nf) U(Δu) W(Δw);
G72	Radial roughing cycle G72	G72 W(Δd) R(e) F__ S _T__; G72 P(ns) Q(nf) U(Δu) W(Δw);
G73	Closed cutting cycle G73	G73 U(Δi) W (Δk) R (d) F_ S_ T_ ;G73 P(ns) Q(nf) U(Δu) W(Δw) ;
G74	Axial grooving multi-cycle G74	G74 R(e) ; G74 X(U)_ Z(W)_ P(Δi)Q(Δk) R (Δd) F_ ;
G75	Radial grooving multi-cycle G75	G75 R(e) ; G75 X(U)_ Z(W)_ P(Δi)_ Q(Δk) R(Δd) F_ ;
G76	Multi-thread cutting cycle G76	G76 P(m)(r)(a) Q(△dmin) R(d); G76 X(U)_ Z(W)_ R(i) P(k)_ Q(△d) F(I) ;

3.6.1. Axial roughing cycle G71

Standard instruction format:

Format: G71 U (△ d) R (e) F_ S_ T_

G71 P(ns) Q(nf) U(△u) W(△w)

U(△d): the single tool feed (radius) of X-axis during roughing, unsigned

R (e): the single tool retreat (radius) of X-axis during roughing, unsigned

P (ns): start segment line of finishing trajectory

Q(nf): end segment line of finishing trajectory

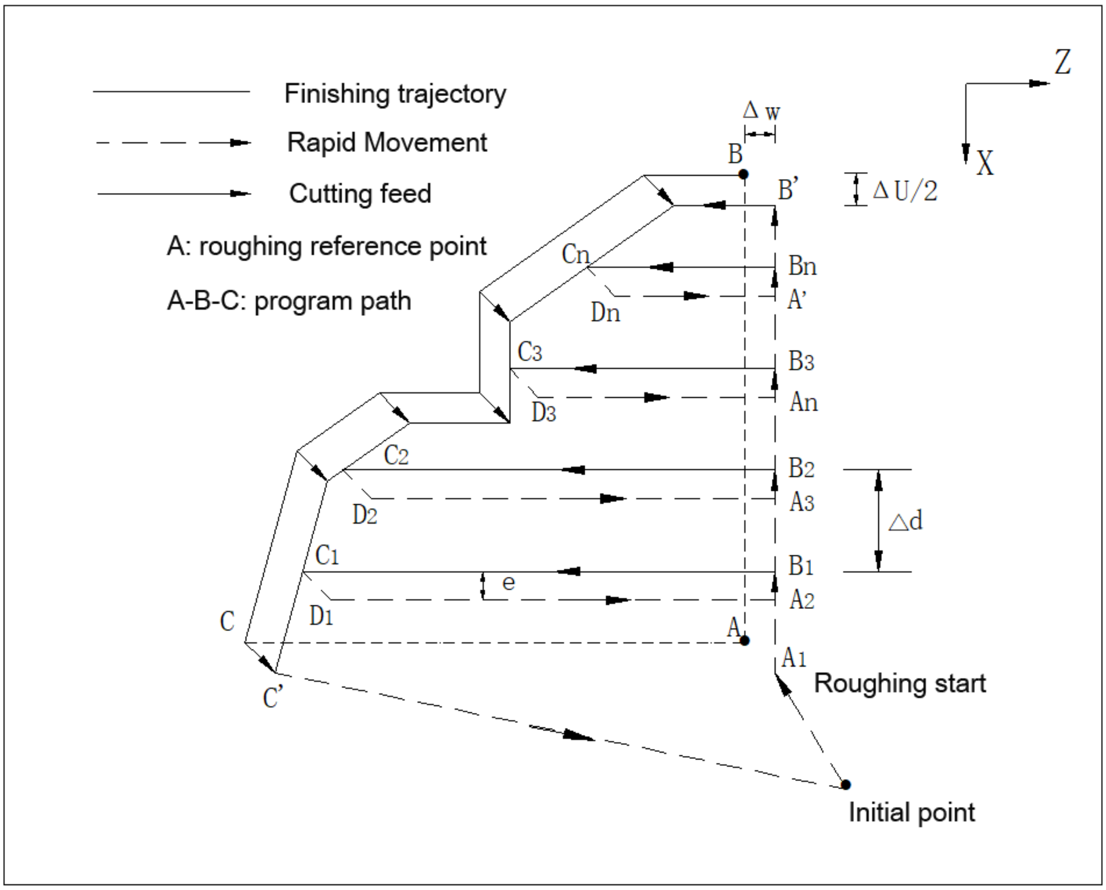
U(△u): reserved finishing amount of X-axis, unsigned

W(△w): reserved finishing amount of Z-axis, unsigned

F: feeding speed during roughing

S: spindle rotation during roughing

T: tool No., tool offset No.



Execution process:

① Initial point → A1: fast moving;

② First roughing cycle: A1 → B1 → C1 → D1 → A2
A1 → B1, C1 → D1 → A2: fast moving
B1 → C1: linear interpolation

③ Second roughing cycle: A2 → B2 → C2 → D2 → A3
A2 → B2, C2 → D2 → A3: fast moving
B2 → C2: linear interpolation

④ The nth roughing cycle: An → Bn → Cn → Dn → A'
An → Bn, Cn → Dn → A': fast moving
Bn → Cn: linear interpolation

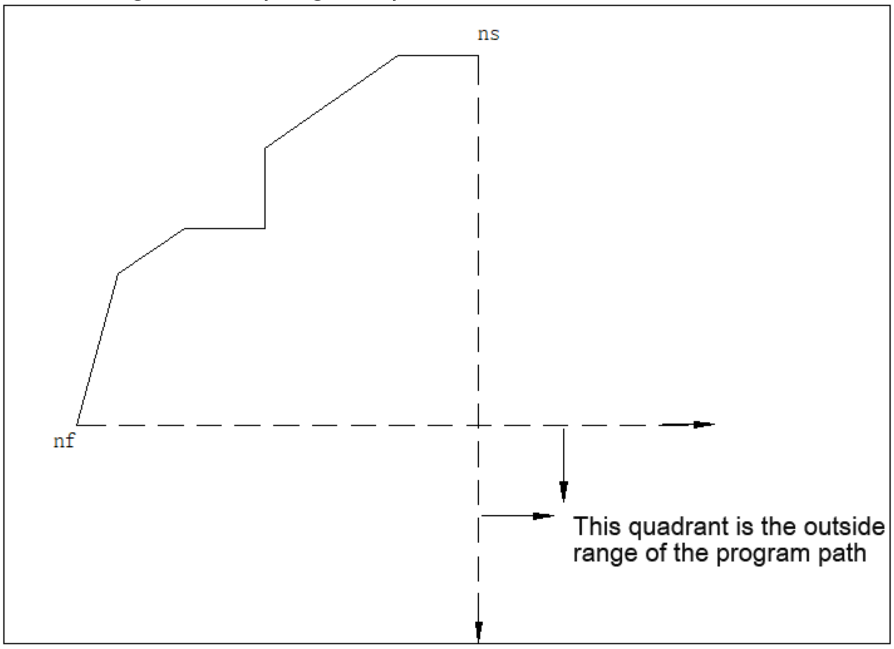
⑤ The last roughing cycle: A' → B' → C' → initial point
A' → B': fast moving
B' → C': interpolation depends on the mode specified by ns → nf segment. Interpolation speed specifies F value or the F value in ns → nf segment according to G71 segment, and uses current default F value if not specified.
C' → Initial point: fast moving. G71 roughing cycle ends.

Function description:

1) Program segment must be compiled after G71 program segment. The system won't execute the program segment between G71 and ns.
2) The F value in G71 program segment is effective when G71 rough machining cycle is executed. If F value is not defined in G71 program

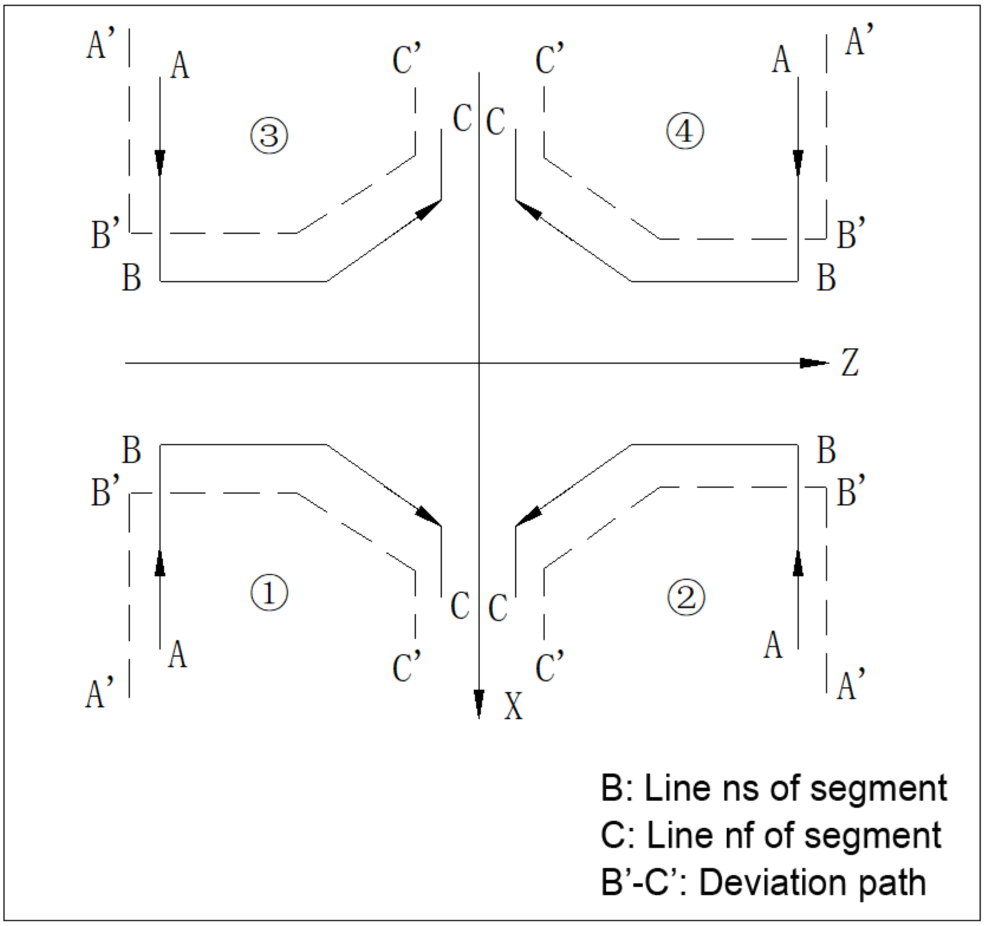
segment, the default F value will be used.

3) Pause and reset are valid in the motion process.
4) If the combined cycle instruction should be used several times in the same program, ns~nf can't have same segment No., or else the ns~nf profile may have error when the next combined cycle instruction is executed.
5) If the last tool feed is smaller than U(△d), the system will execute the last roughing cycle directly
6) Tool retreating path: after last roughing, the tool retreats to the initial point from point nf.
7) Before executing G71, the tool must specify the position of the initial point first, which must be out of the program path limit range, or else the system alarms (0769--G71, G72, G73 initial point position error). If the position of the initial point is not specified, the current point is the initial point.
8) The outside range of the program path is defined as below



9) The offset direction of roughing depends on program path automatically (as in the figure below, the processing path program is A → B → C)

Prerequisite for judgment	Result
X-axis of C (nf segment) > X axis of B (ns segment); Z-axis of C > Z-axis of B	①
X-axis of C (nf segment) > X axis of B (ns segment); Z-axis of C < Z-axis of B	②
X-axis of C (nf segment) < X-axis of B (ns segment); Z-axis of C > Z-axis of B	③
X-axis of C (nf segment) < X-axis of B (ns segment); Z-axis of C < Z-axis of B	④
X-axis of C (nf segment) = X-axis of B (ns segment) or Z-axis of C (nf segment) = Z-axis of B (ns segment)	Generate error alarm 0768



Related parameters and precautions:

1) If Δd is 0 or larger than total cutting amount, the system alarms. The tool retreating angle is 45°, and do not retreat if R (e) is 0.
2) If △u is 0, X-axis doesn't reserve for finishing; if △u is larger than △d, the system alarms because it can't calculate the first roughing end.
3) If △w is 0, Z-axis doesn't reserve for finishing.
4) In ns-nf segment, the size of X-axis and Z-axis must change one-way (constantly increase or decrease), or else the system alarms.
5) ns-nf segment only contains the following instructions: G00, G01, G02, G03, G04; can't include the following M functions: M30, M98, M99, or else the system alarms.
6) ns segment only can be G00 or G01 instruction.

G71 program processing example:

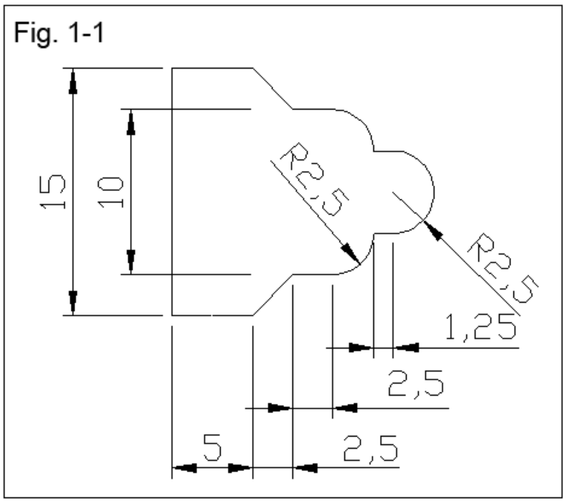


Fig. 3-11-3

```
O7101
M03 S500T0101
G00 X25Z10
M03S550F3000
G71U0.5R0.5
G71P10Q20U0.2W0.2
N10G0X0Z0F300
G2X5Z-2.5R2.5F200
G1W-1.25F300
G2X10W-2.5R2.5F200
G1W-2.5F300
N20X15W-2.5
G70P10Q20
G0X20
Z0
M30
%
```

3.6.2. Radial roughing cycle G72

Format: G72 W (△ d) R (e) F_ S_ T_

G72 P(ns) Q(nf) U(△u) W(△w)

W(△d): the single tool feed of Z-axis during roughing (unsigned, unit: mm)

R(e): the single tool retreating of Z-axis during roughing (unsigned, unit: mm)

P (ns): start segment line of finishing trajectory

Q(nf): end segment line of finishing trajectory

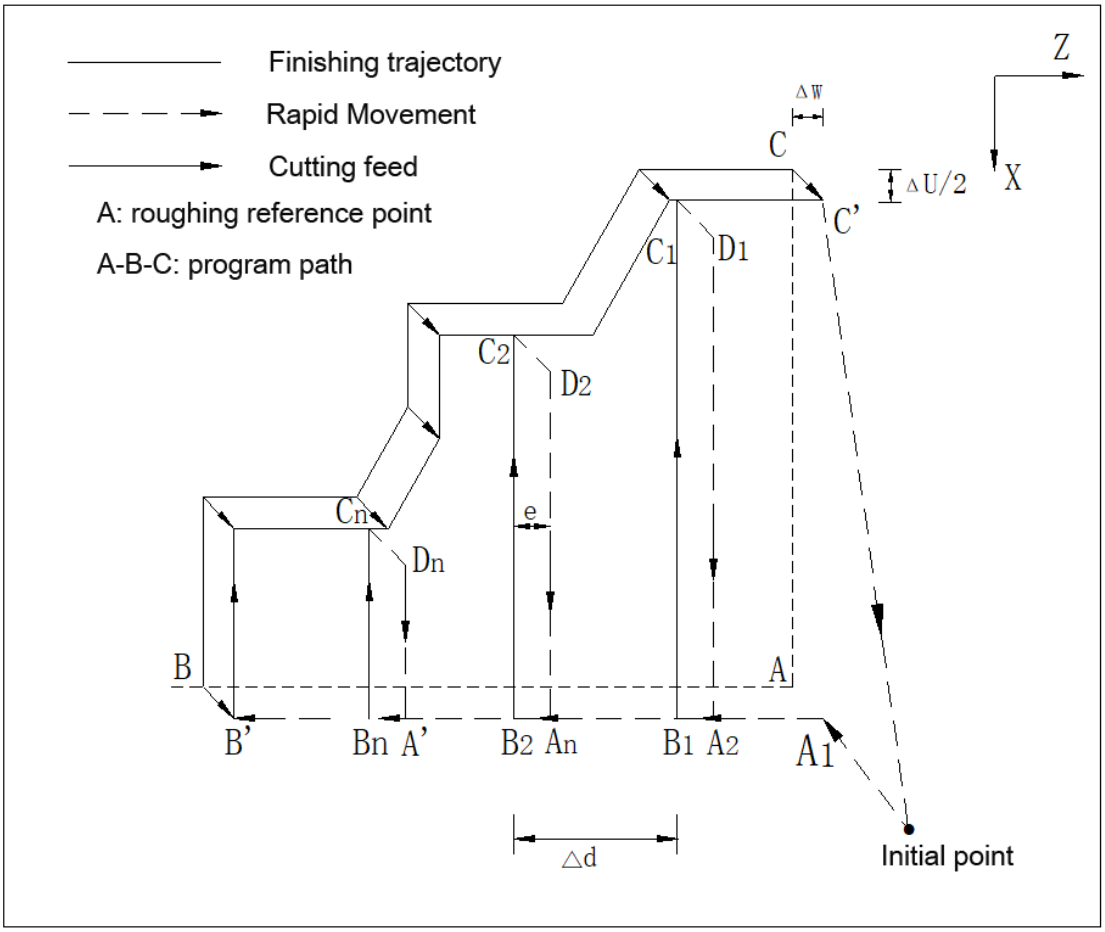
U(△u): reserved finishing amount of X-axis, unsigned

W(△w): reserved finishing amount of Z-axis, unsigned

F: Cutting feeding speed during roughing

S: spindle rotation

T: tool No., tool offset No.



Execution process:

① Initial point → A1: fast moving;

② First roughing cycle: A1 → B1 → C1 → D1 → A2

A1 → B1, C1 → D1 → A2: fast moving

B1 → C1: linear interpolation

③ Second roughing cycle: A2 → B2 → C2 → D2 → An

A2 → B2, C2 → D2 → An: fast moving

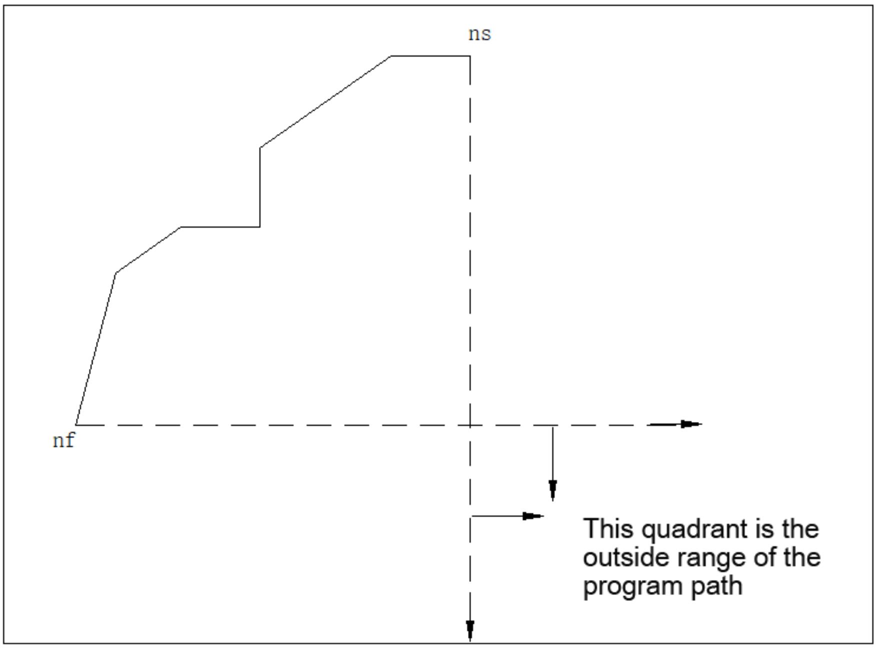
B2 → C2: linear interpolation

④ The n^{th} roughing cycle: An → Bn → Cn → Dn → A'

An → Bn, Cn → Dn → A’: fast moving
Bn → Cn: linear interpolation
⑤ The last roughing cycle: A’ → B’ → C’ → initial point
A’ → B’: fast moving
B’ → C’: interpolation depends on the mode specified by ns → nf segment. Interpolation speed specifies F value or the F value in ns-nf segment according to G72 segment, and uses current default F value if not specified.
C’ → Initial point: fast moving. G72 roughing cycle ends.

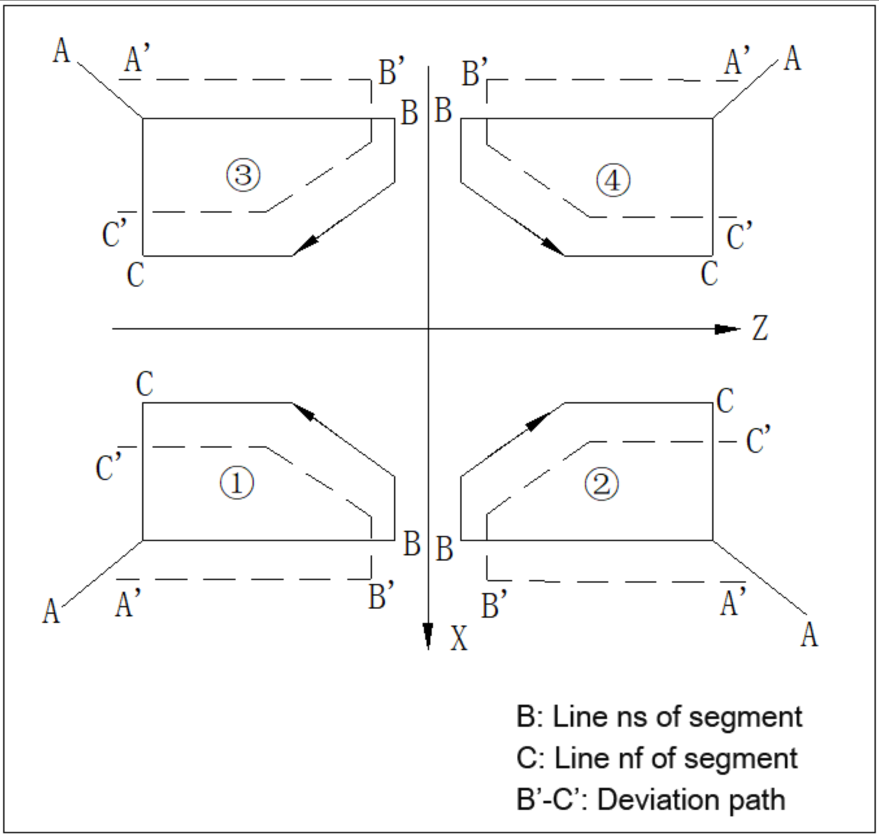
Function description:

1) ns-nf segment must follow G72 segment. The system won’t execute the segment written between G72 segment and ns segment
2) The F value in G72 segment is valid when executing G72 roughing cycle; if the F value is not specified in G72 segment, execute in current default value.
3) Pause and reset are valid in the motion process
4) If the combined cycle instruction should be used several times in the same program, ns~nf can’t have same segment No., or else the ns~nf profile may have error when the next combined cycle instruction is executed.
5) If the last tool feed is smaller than U(△d), the system will execute the last roughing cycle directly
6) Tool retreating path: after last roughing, the tool directly retreats to the initial point from point nf.
7) Before executing G72, the tool must specify the position of the initial point first, which must be out of the program path limit range, or else the system alarms. If the position of the initial point is not specified, the current point is the initial point.
8) The outside range of the program path is defined as below



9) The offset direction of roughing depends on program path automatically (the processing path program is A → B → C)

Prerequisite for judgment	Result
X-axis of C (nf segment) > X axis of B (ns segment); Z-axis of C > Z-axis of B	④
X-axis of C (nf segment) > X axis of B (ns segment); Z-axis of C < Z-axis of B	③
X-axis of C (nf segment) < X-axis of B (ns segment); Z-axis of C > Z-axis of B	②
X-axis of C (nf segment) < X-axis of B (ns segment); Z-axis of C < Z-axis of B	①
X-axis of C (nf segment) = X-axis of B (ns segment) or Z-axis of C (nf segment) = Z-axis of B (ns segment)	Generate error alarm



Related parameters and precautions:

1) If Δd is 0 or larger than total cutting amount, the system alarms.
2) The tool retreating angle is 45°, and do not retreat if R (e) is 0.
3) If △u is 0, X axis does not reserve for finishing
4) If △w is 0, Z axis does not reserve for finishing; if △w is larger than △d, the system alarms.
5) In ns-nf segment, the size of X-axis and Z-axis must change one-way (constantly increase or decrease), or else the system alarms.
6) ns-nf segment only contains the following instructions: G00, G01, G02, G03, G04; can't include the following M functions: M30, M98, M99, or else the system alarms.
7) ns segment only can be G00 or G01 instruction.

Example of machining with G72

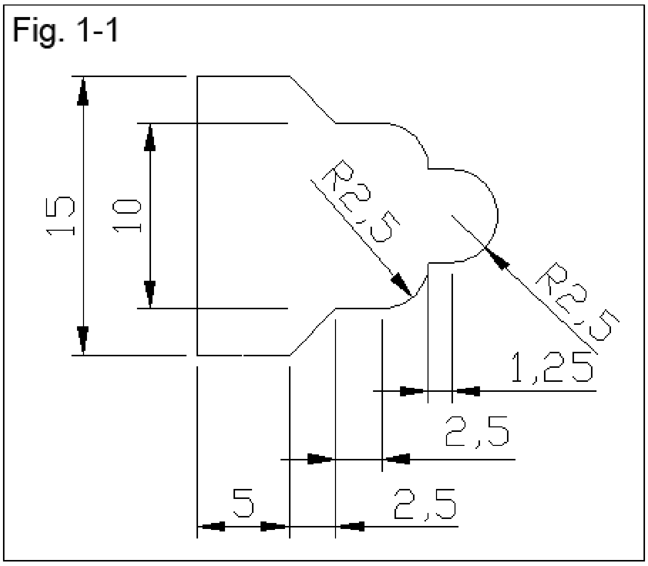


Fig. 3-11-6

O7201
M03S500T0101
G00X20Z5
M03S500F3000
G72W0.5R-0.4
G72Q20U0.2W0.2
N10G1X17Z-11.25F300
X15
W5
G1X10W2.5F200
G1W2.5F300
G02X5W2.5R2.5F200
G1W1.25F300
N20G2X0W2.5R2.5F200
G70P10Q20
G0X20
Z5
M30
%

3.6.3. Closed cutting cycle G73

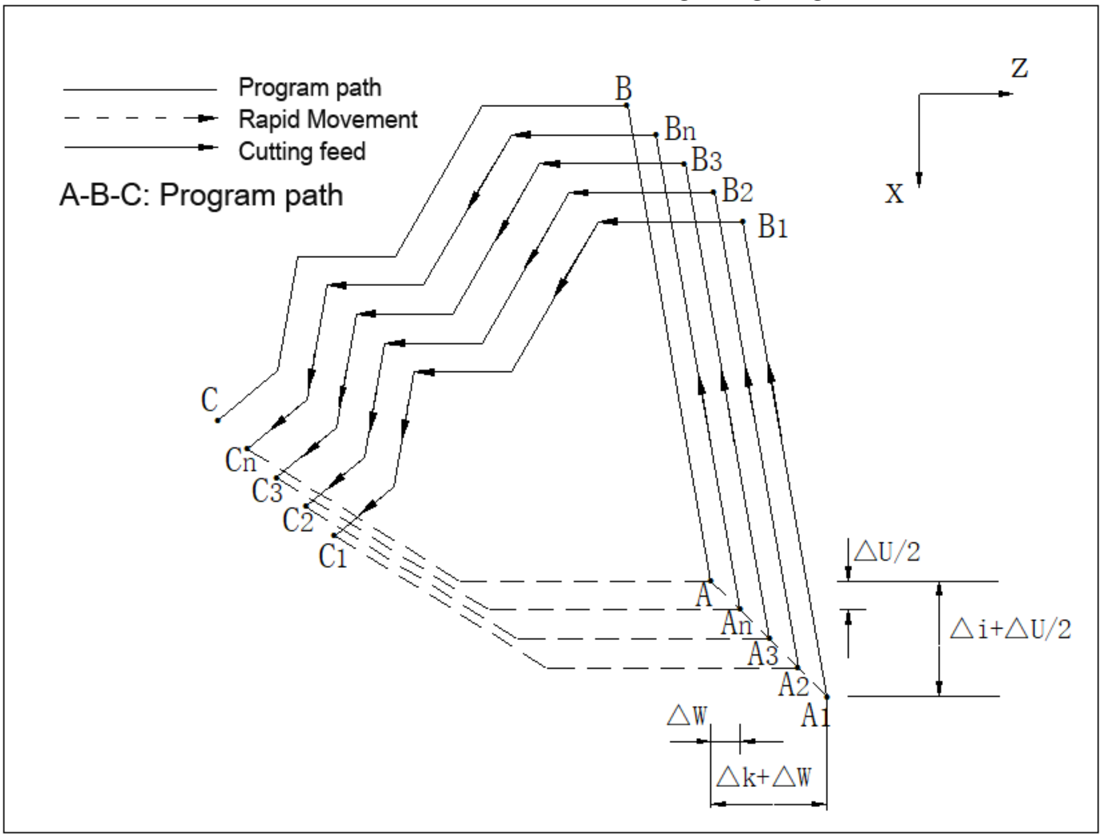
Format:

G73 U(△i) W(△k) R(d) F_ S_ T_

G73 P(ns) Q(nf) U(△u) W(△w)

U(△i): Reserved for roughing on X-axis. Unsigned, negative sign

is invalid (radius)
W(△k): Reserved for roughing on Z-axis. Unsigned, negative sign is invalid
R(d): Roughing cutting times. Unsigned integer, decimal fraction is invalid (unit: time)
P (ns): start segment line of finishing trajectory.
Q(nf): end segment line of finishing trajectory.
U(△u): Reserved finishing amount of X-axis. Unsigned, negative sign is invalid (diameter).
W(△w): Reserved finishing amount of Z-axis. Unsigned, negative sign is invalid
F: the cutting feeding speed during roughing
S: spindle rotation during roughing
T: tool No., tool offset No. during roughing



Execution process:

① A→A1: fast moving;
② First roughing cycle: A1→B1→C1→A2:
A1→B1→C1: Interpolation mode is defined by program segment ns→nf. The interpolation speed is subject to F value defined by G73 program segment. If not defined, the default F value will be used.
C1→A2: fast moving
Second roughing cycle: A2→B2→C2→A3
A2→B2→C2: Interpolation mode is defined by program segment ns→nf. The interpolation speed is subject to F value defined by G73 program segment. If not defined, the default F

value will be used.

C2→A3: fast moving

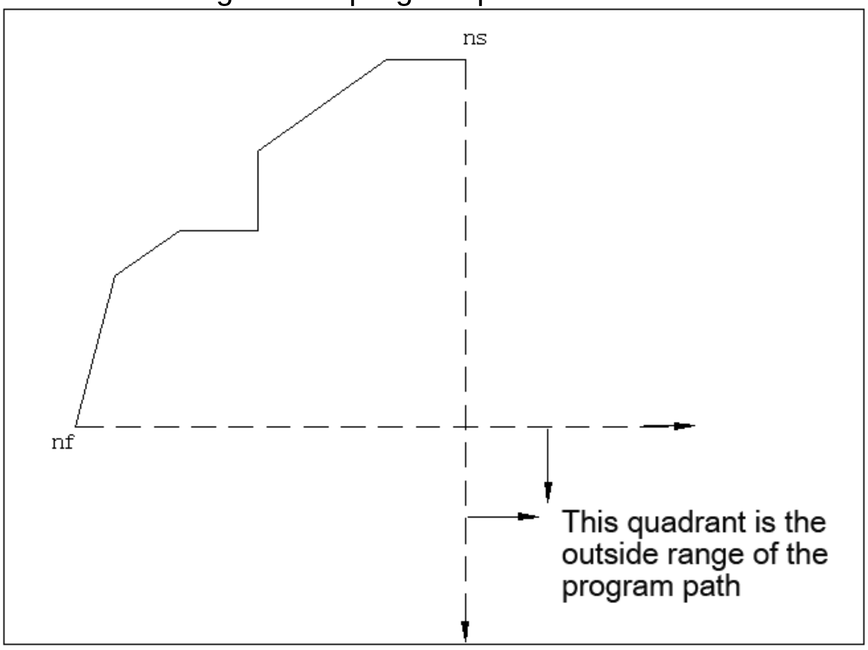
The n^th roughing cycle: An→Bn→Cn→A

An→Bn→Cn: Interpolation mode is defined by program segment ns→nf. The interpolation speed is subject to F value defined by G73 program segment. If not defined, the default F value will be used.

Cn→A: fast moving. Roughing cycle ends.

Function description:

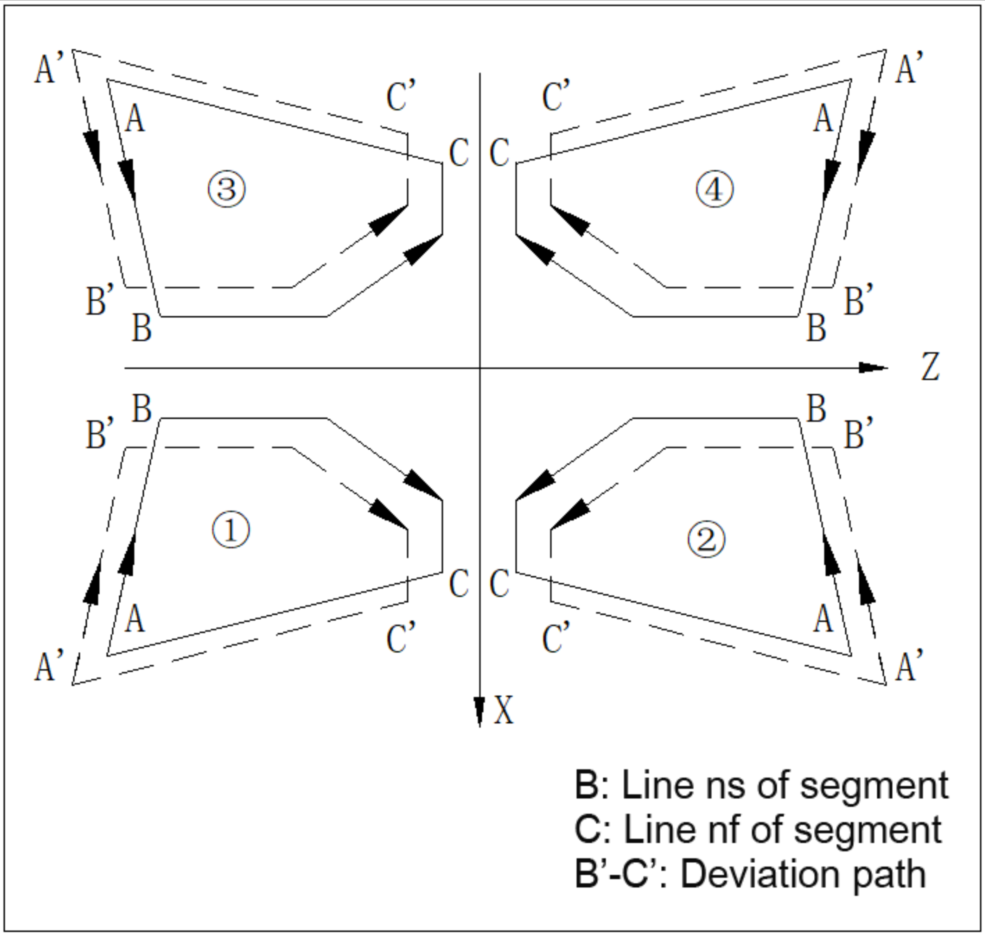
1) The ns-nf segment must follow G73 segment. The system will not execute the segment written between G73 segment and ns segment
2) The F value in G73 segment is valid during G73 roughing cycle. If F value is not defined in G73 program segment, the default F value will be used. The program segment ns-nf is effective only when G70 is executed.
3) In tool tip radius offset, if G73 is executed, the offset will be valid. However, in G73 cycle, the start point of the tool is the position where the tool tip radius offset is temporarily cancelled
4) In single segment operation, the system will stop at the end point of each program segment, and pause state and operation resumption become effective in the motion.
5) If the combined cycle instruction should be used several times in the same program, ns~nf can't have same segment No., or else the ns~nf profile may have error when the next combined cycle instruction is executed.
6) Before the execution of G73, the start point of the tool must be defined. It must be at the outer side of the limit of the program path. Otherwise, the system will alarm. If the position of the initial point is not specified, the current point is the initial point.
7) The outside range of the program path is defined as below



8) The offset direction of roughing depends on ns~nf program path: (the

processing path program is A→B→C)

Prerequisite for judgment	Result
X-axis of C (nf segment) > X axis of B (ns segment); Z-axis of C > Z-axis of B	①
X-axis of C (nf segment) > X axis of B (ns segment); Z-axis of C < Z-axis of B	②
X-axis of C (nf segment) < X-axis of B (ns segment); Z-axis of C > Z-axis of B	③
X-axis of C (nf segment) < X-axis of B (ns segment); Z-axis of C < Z-axis of B	④
X-axis of C (nf segment) = X-axis of B (ns segment) or Z-axis of C (nf segment) = Z-axis of B (ns segment)	Generate error alarm



Related parameters and precautions:

1) △i can be set with parameter (116-G73 X axis margin), or modify parameter setting with program instruction; if △i is smaller than △u, the system alarms.
2) △k can be set with parameter (117-G73 Z axis margin), or modify parameter setting with program instruction; if △k is smaller than △w, the system alarms.

3) d can be set with parameter (120-G73 cycle times), or modify parameter setting with program instruction; if d is non-integer larger than 1, the system only calculates the integer, and if d is smaller than 1, the system calculates as 1
4) The system alarms if ns and nf are not defined.
5) △u can be set with parameter (118-G73 X-axis finishing margin), or modify parameter setting with program instruction; if △u is 0, X-axis does not reserve for finishing
6) △w can be set with parameter (119-G73 Z-axis finishing margin), or modify parameter setting with program instruction; if △w is 0, Z-axis does not reserve for finishing.
7) In ns-nf segment, the size of X axis and Z-axis must change one-way (constantly increase or decrease), or else the system alarms.
8) ns-nf segment only contains the following instructions: G00, G01, G02, G03, G04; can't include the following M functions: M00, M30, M98, M99, or otherwise the system alarms.
9) ns segment only can be G00 or G01 instruction, or otherwise the system alarms.
10) In MDI mode, instruction G73 cannot be executed, otherwise the system alarms.

Example of machining with G73

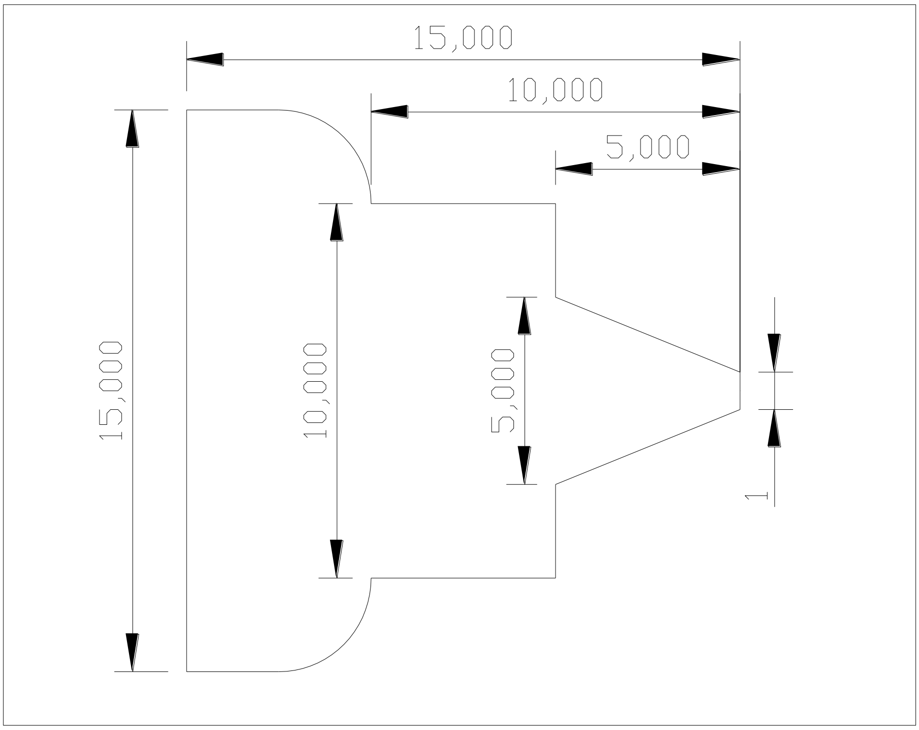


Fig. 3-11-8

O7301

M03S500T0101

G00X25Z5

G73U5W5R5F500

G73P10Q20U0.2W0.2

```
N10G0X1Z-0.5R0.5F300
X5Z-5
X10
Z-10
G03X15W-2.5R2.5
G1Z-15
N20G2X20W-2.5R2.5F200
G70P10Q20
G0X20
Z5
M30
%
```

3.6.4. Finishing cycle G70

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

P(ns): the number of the first segment of the finishing trajectory;

Q(nf): the number of the last segment of the finishing trajectory;

Instruction function: the tool starts finishing from the start position along the work-piece finishing track specified by ns~nf segment. After G71, G72 or G73 roughing, finish with G70 instruction, and complete the cutting of finishing margin in one time. After G70 cycle, the tool returns to the start point and executes the next segment after G70.

G70 instruction track depends on the programming track of ns~nf segment. The relationship of ns and nf in G70~g73 segment is as follow:

```
 ........
 G71/G72/G73 ......;
 N(ns) ......
........                  ⎫
      · F_                ⎪
      · S_                ⎬   Finishing path segment
        T_                ⎪
      .                   ⎪
      .                   ⎭
N(nf) ... ... ...
G70 P(ns)   Q(nf)) ;
```

Notes:

- F, S and T instructions in ns~nf segment are valid when executing G70;
- While executing G70 instruction, it is possible to stop automatic

running and move manually; however, when G70 cycle is executed again, it is required to return to the position before manual movement. If not, it will continue the execution and following running track will be dislocated.

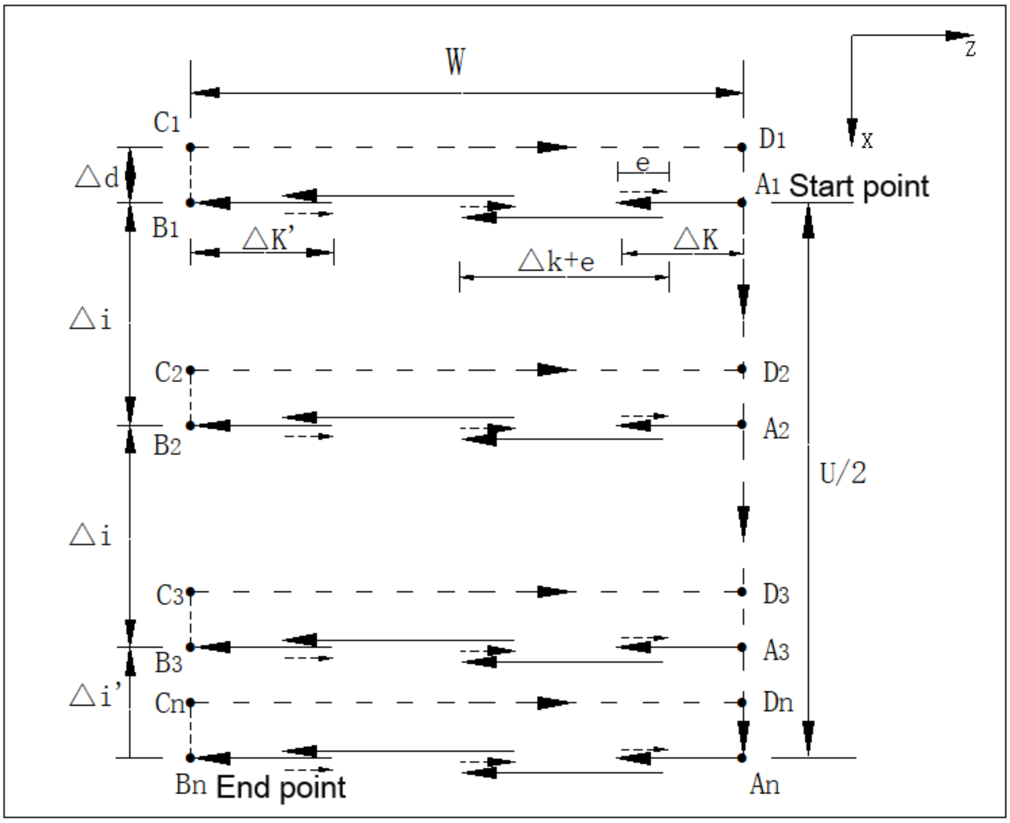
- In segment ns~nf, the following instructions are unavailable:
  Other G instructions except G04 (Pause) in group 00;
  Other G instructions except G00, G01, G02 and G03 in group 01;
  Subroutine call instruction (e.g. M98/M99)

3.6.5. Axial grooving multi-cycle G74

Format: G74 R (e);

G74 X/U _ Z/W_ P(Δi) Q(Δk) R(Δd) F_ ;

R(e): axial (Z-axis) tool retreating, unsigned
X/U: groove end coordinates (X is absolute coordinates, and U is the increment from current coordinates to point coordinates)
Z/W: groove end coordinates (Z is absolute coordinates and W is the increment from current coordinates to point coordinates)
P(Δi): single radial (X-axis) offset of the tool, radius, unsigned. Offset direction is same to the radial direction of the end coordinates
Q(Δk): single axial (Z axis) feeding of the tool, unsigned. Feeding direction is same to the axial direction of the end coordinates
R(Δd): radial (X axis) tool retreating after cutting to the groove bottom, unsigned. The retreating direction is opposite to the radial offset direction.
F: axial grooving cycle feeding speed



Execution process:

① First grooving cycle: A1 → B1 → C1 → D1 → A2

A1 → B1: axially feed to point B1 in pecking style; B1 → C1 → D1 → A2: fast moving (the cycle B1 → C1 does not have tool retreating action)

② Second grooving cycle: A2 → B2 → C2 → D2 → A3

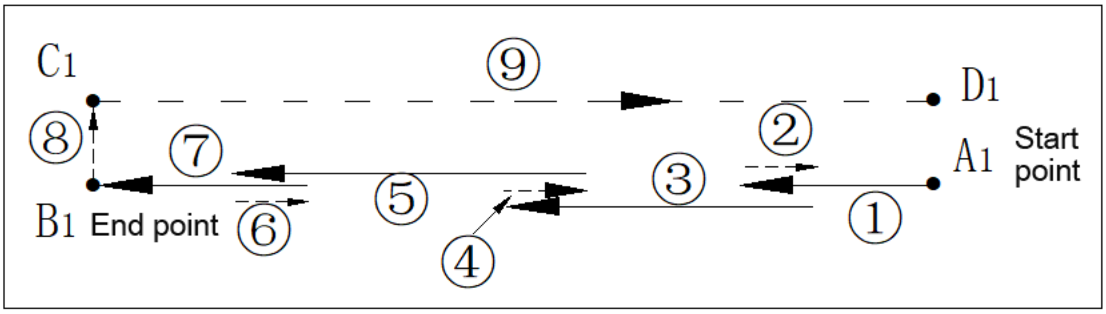
A2 → B2: axially feed to point B2 in pecking style; B2 → C2 → D2 → A3: fast moving.

③ The nth grooving cycle: An → Bn → Cn → Dn → A1

An → Bn: axially feed to point Bn in pecking style; Bn → Cn → Dn → A1: fast moving. G74 grooving cycle ends

Function description:

1) The position of the cutting start (A1) should be defined before grooving cycle, or else the system takes current point as the cutting start.
2) In the first grooving cycle, the tool is not retreated in radial direction when cut to the groove bottom (B1)
3) If X/U is not defined or movement is 0, the system considers P(Δi) as 0 no matter whether P(Δi) is defined. In this case, only Z-axis acts and cuts to groove end, and the tool is not retreated in radial direction.
4) Pecking feeding cycle



Related parameters and precautions:

1) If e is 0, the system does not have axial tool retreating action.
2) The system alarms if Z/W is not defined or the movement is 0.
3) The system alarms if X/U is specified but P(Δi) is larger than X axis movement (U/2). If P(Δi) is specified to 0, it can be used for axial cycle drilling.
4) If Δk is 0 or larger than entire groove depth (W), the system reports error.
5) If Δd is 0, the tool isn't retreated in radial direction; if Δd is larger than Δi, the system reports error.

Example of machining with G74: see Fig. 3-11-10

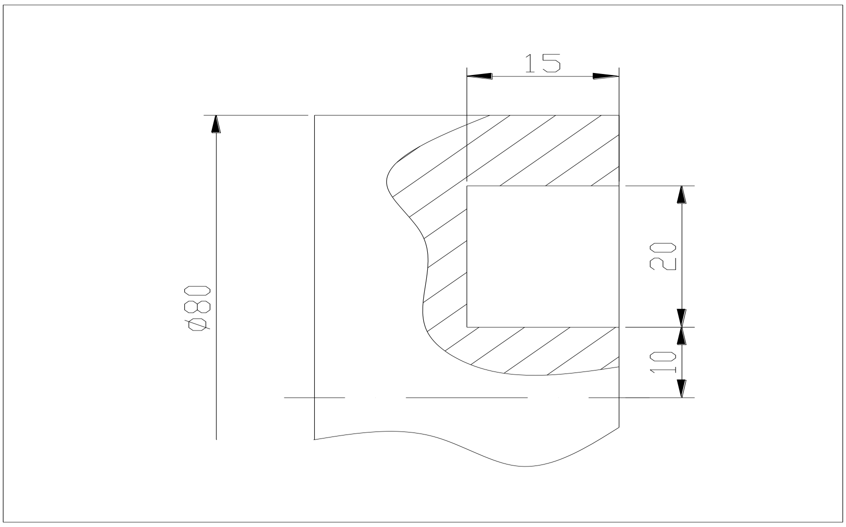


Fig. 3-11-10

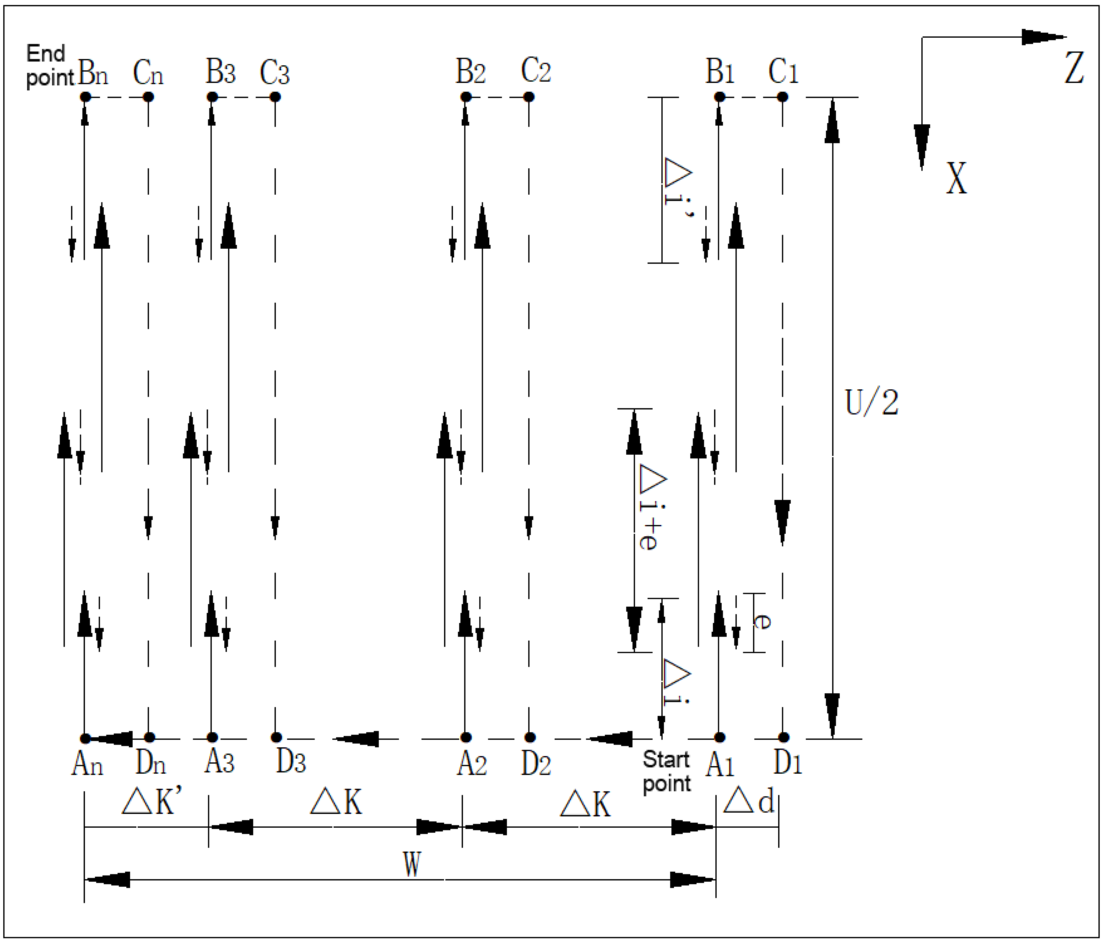
```
O7401
G00X100Z20
M03S800T0101
 G01X60Z2
G74R3
G74X20Z-15P2Q5R1F500
G00X100Z20
 M30
 %
```

3.6.6. Radial grooving multi-cycle G75

Format: G75 R (e);

G75 X/U _ Z/W_ P(Δi) Q(Δk) R(Δd) F_ ;

R(e): radial (X axis) tool retreating, radius, unsigned
X/U: groove end coordinates (X is absolute coordinates, and U is the increment from current coordinates to point coordinates)
Z/W: groove end coordinates (Z is absolute coordinates and W is the increment from current coordinates to point coordinates)
P(Δi): single radial tool feeding, unsigned, radius. The feeding direction is same to the radial direction of the end coordinates.
Q(Δk): axial offset of the tool, unsigned. Offset direction is same to the axial direction of the end coordinates.
R(Δd): axial (Z) tool retreating after cutting to the groove end, unsigned. The retreating direction is opposite to the axial offset direction of the tool.
F: axial grooving cycle feeding speed



Execution process:

④ First grooving cycle: A1 → B1 → C1 → D1 → A2
A1 → B1: radially feed to point B1 in pecking style; B1 → C1 → D1 → A2: fast moving (no tool retreating form the present cycle B1 → C1)

⑤ Second grooving cycle: A2 → B2 → C2 → D2 → A3

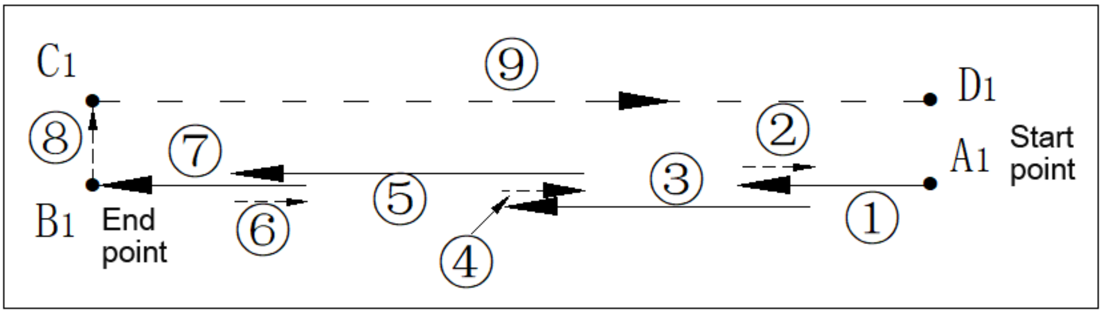
A2 → B2: radially feed to point B2 in pecking style; B2 → C2 → D2 → A3: fast moving.

⑥ The n[th] grooving cycle: An → Bn → Cn → Dn → A1

An → Bn: radially feed to point Bn in pecking style; Bn → Cn → Dn → A1: fast moving. G75 grooving cycle ends

Function description:

5) The position of the cutting start (A1) should be defined before grooving cycle, or else the system takes current point as the cutting start.
6) In the first grooving cycle, the tool is not retreated in axial direction when cut to the groove bottom (B1)
7) If Z/W is not defined or movement is 0, the system considers Q(Δk) as 0 no matter whether Q(Δk) is specified. In this case, only X axis acts and cuts to groove end, and the tool is not retreated in axial direction.
8) Pecking feeding cycle



Related parameters and precautions:

1) If e is 0, the system does not have radial tool retreating action.
2) The system reports error if X/U is not defined or the movement is 0.
3) If Δi is 0 or larger than entire groove depth (U/2), the system reports error.
4) The system reports error if Z/W is defined but Δk is larger than z axis movement (W). If Δk is 0, it can be used for radial cycle drilling
5) If Δd is 0, the tool is not retreated in axial direction; if Δd is larger than Δk, the system reports error.

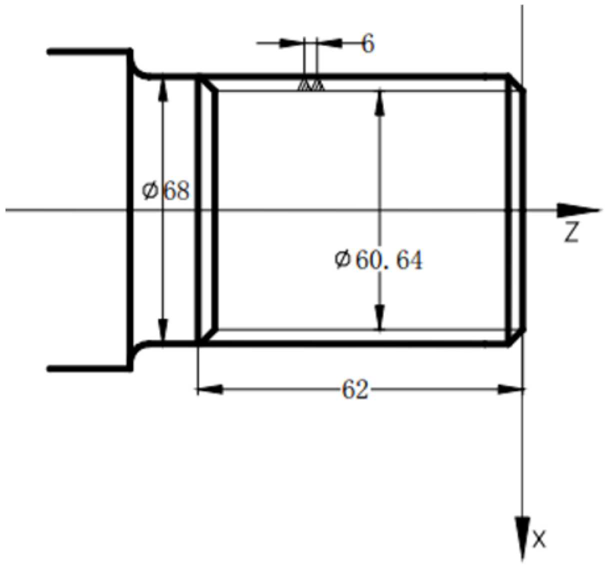
3. 6. 7. Multi-thread cutting cycle G76

Format: G76 P (m) (r) (a) Q (△ dmin) R (d)
G76 X(U) Z(W) R(i) P(k) Q(△d) F(I)

X ------- The absolute coordinate of the X-axis of the thread end point (unit: mm)
U ------- The difference between the absolute X-axis coordinates of the thread end and start points (unit: mm)

Z ------- The absolute coordinate value of the Z-axis of the thread end point (unit: mm)
W ------- The difference between the absolute Z-axis coordinates of the thread end and start points (unit: mm)
P (m) ---- The number of times of thread finishing ranging from 00 to 99, (unit: times);
P (r) ---- Tailback length of thread ranging from 00 to 99, (unit: 0.1mm);
P (a) ---- The angle between the adjacent two threads, ranging from 00 to 99, unit: degree (°).
Q (△ dmin)-The range of the minimum cutting amount during roughing is 0-9999.999 (unit: mm, unsigned). When (√n-√n-1) * ∆d <∆dmin, ∆dmin is regarded as the cutting amount of this roughing, that is, the current thread depth is √n-1 * ∆d + ∆dmin). ∆dmin is set to avoid the too small of roughing amount and too many roughing times caused by the cutting amount deceleration in thread roughing.
R (d) ---- Cutting amount of thread finishing with the value range of 0-99.99 (unit: mm, unsigned, radius)
R (d) must be less than P (k);
R (i) ----Thread taper, which refers to the difference between the absolute X-axis coordinates of the start point and end point of the thread, ranging from -9999.999 to 9999.999 (unit: mm, radius). If R (i) is not entered, the system will process it as R (i) = 0 (straight thread).
P (k) ----height of thread; the total cutting depth of the thread, ranging from 0 to 9999.999 (unit: mm, radius). If P (k) is not input, the system alarms;
Q (△ d)-the first thread cutting depth ranging from 1 to 9999.999, (unit: mm, radius, unsigned); if ∆d is not input, the system alarms, and Q (∆d) must be <P (k);
F ------- Metric thread lead ranging from 0 to 500, (unit: mm);
I ------- thread lead Unit: Threads per Inch

Example: The following (external thread) thread is M68 * 6;



Enlarged view of cutting point

Procedures

O7401	
G50 X100 Z50	(Set the work-piece coordinate system)
M3 S300	(Define the spindle speed)
G00 X80 Z10	(Quickly move to the start point of machining)
G76 P020560 Q0.15 R0.1	(The number of repetitions of finishing is 2, the chamfering width is 0.5mm, the cutter angle is 60 °, the minimum cutting depth is 0.15mm, and the finishing margin is 0.1)
G76 X60.64 Z-62 P3.680 Q1.8 F6	(The height of thread is 3.68, and the first thread cutting depth is 1.8)
G00 X100 Z50	(Return to the origin of the program)
M30	(End of program)
%	

4. CNC Process Knowledge

1. What is the main content of CNC programming?

A: CNC programming mainly includes: analyzing part drawing, determining the process and the process route, calculating the coordinates of tool trajectory, writing processing program, entering program to the CNC system, program verification and test cutting of first piece.

2. What is the objective of CNC processing analysis? And the content?

A: To process parts on the CNC machine tool, it is required to analyze and treat the process according to part drawing, create CNC process, and then prepare processing program. The entire process is automatic. The content includes machine tool cutting amount, work step arrangement, feeding path, processing margin, tool size and model, etc.

3. What is tool setting point? What is the influence of tool setting point selection to programming?

A: Tool setting point is the start point of the tool relative to work-piece motion during CNC processing. This start point is also the program start of programming. If the tool setting point is selected properly, it is convenient for mathematical processing and programming; easy to align on the machine tool; convenient for checking and reduce processing error during processing.

4. What are machine tool coordinate system and work-piece coordinate system? And their differences?

A: Machine tool coordinate system is also called mechanical coordinate system, which is the feeding coordinate system of the moving parts of the machine tool, and the coordinate axes and direction are standard. The coordinate origin is set by the manufacturer, and is called machine tool (or part) origin. Work-piece coordinate system is also called as programming coordinate system, which is used for programming.

5. Please simply describe the tool position, tool change point and work-piece coordinate origin.

A: Tool position point is the reference point to confirm the tool position. If the CNC machine tool with multi-tool processing needs to change tool during processing, it is also required to set a tool change point during programming. Tool change point is the reference point to change tool position. The positions of tool change points shouldn't interfere with each other. The origin of work-piece coordinate system is also work-piecezero or programming zero; the position is set by the programmer at the design and process reference position of the workpiece for size calculation.

6. What is the effect of tool compensation and what are compensation instructions?
A: Tool compensation usually includes length compensation and radius compensation. Tool length compensation can be tool length compensation or position compensation. With tool radius compensation: execute roughing and finishing with the tool of same program and of same size; program with part profile directly to avoid calculating tool center trajectory; if the diameter is changed due to tool abrasion or change, it doesn’t need to modify the program, and only needs to enter the value of tool radius change in tool parameter setting state; with tool compensation function, use same program to process the internal and external profiles of same nominal size.

7. What is the separation between roughing and finishing? And the advantages?
A: When determining the part processing flow, the roughing and finishing should be executed separately, execute the finishing after the roughing of every surface, and do not execute both roughing and finishing on the machine tool, which is called separation of roughing and finishing. This processing allows using the machine tool reasonably, making the distortion and error produced in roughing be modified in finishing, to improve processing precision; in addition, it also finds crack, pore and other defects in advance so as to stop the processing in time.

8. What is the advantage of installing workpiece with clamps?
A: Since the positioning components of the clamp and the relative position of the tool and machine tool motion can be adjusted in advance, using clamp when processing parts is fast and convenient, and has high repeat precision, which can ensure the processing requirement of the workpiece.

9. What is the advantage of selecting refined reference according to the principle of unified reference?
A: The refined reference selected in this principle can be used for the processing of multi-surface and multi-process, reduce the error due to reference change and improve the processing precision. In addition, it also can reduce the types of clamp and the workload in clamp design.

10. What is the principle to confirm the direction of clamping force?
A: (1) The direction of clamping force does not destroy the accuracy of workpiece positioning.

(2) The direction of clamping force should minimize the force.

(3) The direction of clamping force should minimize the workpiece distortion.

11. What are the factors that cause spindle rotation error?

A: Concentricity between over-bearing holes of the bearings, verticality between shell hole positioning face and the axis, bearing clearance, roundness of the bearing raceway, size and shape error of the roller, beating of locking nut end, etc.

12. What are the bad results if the beating of the spindle shoulder bearing surface exceeds the tolerance?

A: In this case, the holes bored through spindle feeding and reference surface tilt. In addition, if the cutter wears due to spindle feeding is uneven and the wear of some blades speeds up, tool use is not economical.

13. How the materials difficult to process are measured?

A: The materials difficult to process are measured in quality of processed surfaces, chip formation and difficulty of discharge. If one of above three aspects is poor obviously, the material is difficult to process.

14. What are the cutting characteristics of materials difficult to process?

A: In the following five aspects:

1) Since the materials difficult to process have low thermal conductivity and high thermal strength, the cutting temperature is high;
2) The chip distortion has large factor, and the hardening is serious;
3) The materials usually have high strength and thermal strength, and thus the cutting force is large;
4) Cutter wear soon, and thus the durability is low;
5) Chip rolling, breaking and discharging are difficult.

15. What improvement measures should be taken to cut the materials difficult to process?

A: The following improvement measures should be taken:

1) Select appropriate tool material with good cutting performance;
2) Select reasonable tool geometric parameters;
3) Use proper cutting liquid;
4) Select reasonable cutting amount.

16. What are the advantages of using indexable carbide blade?

A: It has the following advantages:

1) Since the blade isn't welded and doesn't require grinding during usage, it will avoid internal stress and crack due to welding and grinding and improve the tool durability.
2) The tool is used for a long time, and it will avoid internal stress and crack due to welding and grinding and improve the tool durability.
3) The tool can be used for a long time, and therefore it will not only reduce the tool material, but also reduce the labor and equipment required to manufacture and grind the tool.
4) If the tool is blunt, you can just rotate the blade to continue using, and thus shorten the time in tool change, tool setting, etc.
5) It is convenient to recycle when the blade is blunt, reduce tool material consumption and reduce the cost.

17. What is size reference?
A: The start size is the size reference.

18. How to select reference while drawing?
A: The design reference, process reference and measurement reference should be unified, so as to reduce error and convenient for measurement and checking.

19. What is metal cutting process?
A: While cutting, under the impact of cutting blade and tool face, the metal being cut slips and becomes chips.

20. What are the influences of cutting amount on cutting force?
A: (1) When cutting depth ap and feeding amount f increase, cutting force increases; it is known from the cutting force calculation formula FZ=150apf0.75 that if the cutting depth increases by one time, main cutting force FZ increases by one time; if the feeding amount increases by one time, the main cutting force FZ increases by 70% (chip thickness increases and chip distortion decreases due to the increase of feeding amount, and therefore the cutting force must be lower).
(2) When cutting plastic metal, the cutting force usually decreases along with the increase of cutting speed; when cutting brittle metal, cutting speed doesn't have much influence on cutting force.

21. What are the materials of tool cutter?
A: At present, the tool materials include metal and non-metal materials: metal materials include carbon tool steel, alloy tool steel and high-speed carbide. Non-metallic materials include synthetic diamond and cubic boron nitride and

ceramics, in which the hard red performance of carbon tool steel and alloy tool steel is low (about 200 to 400°C), and are rarely used to make tools.

22. What are the influences of cutting amount on cutting temperature?
A: If cutting speed increases by one time, the cutting temperature increases about 30%-40%; if the feeding amount increases by one time, the cutting temperature only increases 15%-20%; if the cutting depth increases by one time, the cutting temperature only increases 5%-8%.

23. What is process dimension chain?
A: The interrelated dimensions are arranged to a size closure map in a certain order of end to end, and it is called a dimension chain; the dimension chain applied in the processing is known as the process dimension chain.

24. What is positioning error?
A: The position error of the processing surface relative to process reference due to work-piece positioning is called positioning error.
25. What are the eight aspects that may cause errors in processing?
A: Processing errors include principle error, clamping error, machine tool error, fixture accuracy error, processing system deformation error, error in residual stress of the workpiece, tool error and measurement error.

26. What is the objective of using clamp in machinery manufacture?
A: In machinery manufacturing, the objective of using clamp is to ensure product quality, improve labor productivity, solve the special difficulty in lathe processing, expand the processing range of the machine tool, and reduce the technical requirement on workers.
27. What spindles are used if the workpiece is positioned with internal hole?
A: The commonly used mandrels are cylindrical mandrel, small taper spindle, cone mandrel, screw spindle, and splined spindle.

28. What are the effects of positioning device and clamping device?
A: The effect of the positioning device is to confirm the position of the workpiece in the clamp, making the workpiece in the proper position relative to tool and cutting motion during processing. The effect of the clamp device is to clamp the workpiece, ensuring that the given position of the workpiece in the clamp won’t change during processing.

29. What is repeat positioning? What is partial positioning?
A: The positioning points are more than the degrees of freedom that should be limited, indicating that certain positioning points limit same degree of freedom,

which is called repeat positioning.

30. What are machine tool errors? What are their main influences on the quality of workpiece?

A: Machine tool errors include:

1) The error between machine tool spindle and bearing due to manufacture and wear. It has bad influences on roundness, flatness and surface roughness of the workpiece.
2) The error due to machine tool rail wear, which will cause error in the straightness of cylinder;
3) Machine tool transmission error: it destroys proper motion relationship and cause pitch error.
4) Machine tool installation position error, e.g. parallel error of guide rail and spindle installation, which will cause taper error in processed cylinder.

31. What are geometric accuracy and work precision of the lathe?

A: Geometric accuracy of the machine tool refers to the geometric shape accuracy of the basic parts of the machine tool, geometric accuracy of relative position, and geometric accuracy of the relative motion. The work precision of the machine tool is the precision of the machine tool in motion state and under the effect of the cutting force. The precision of the machine tool in work state is reflected on the precision of the processed parts.

32. What are the conditions of ribbon cutting?

A: When process plastic metal materials, the cutting speed is high, cutting thickness is low, cutting angle is large, the slide amount of the chips is low and hasn't destroyed the materials, and thus form ribbon chips.

33. What is the important meaning of process analysis?

A: Appropriate process analysis has important meaning to ensure processing quality, improve productivity, reduce production cost, reduce labor strength and make reasonable process regulations.

34. What are design reference and process reference (including assembly reference, positioning reference, measurement reference and procedure reference)?

A: (1) Design reference is the reference on the part drawing to determine the positions of other points, lines and surfaces.

(2) Process reference is the reference of the parts used in processing and assembly. According to the purpose, it is divided into assembly

reference, measurement reference, positioning reference and procedure reference.

1) Assembly reference is used to confirm the position of the part in component and product during assembly.

2) The measurement reference is used to measure the size and position of processed surface.

3) The positioning reference is used to make the workpiece occupy the proper position in machine tool or clamp during processing.

4) Procedure reference is used to confirm the size, shape and position precision of the processed surface on the procedure drawing.

35. What is the reason of positioning error? How to calculate?

A: When a batch of workpieces are processed in clamp, the reasons for errors in processing size might be:

1) The maximum displacement of same batch of workpieces along the direction of processing size caused by the size and geometric shape error of the positioning reference and the clearance between the positioning reference and positioning components is called positioning reference displacement error, which is expressed in NY.

2) The displacement of the size of same batch workpieces relative to procedure reference caused by non-coincidence between procedure reference and positioning reference is called reference non-coincidence error, which is expressed in ÑB.

The sum of above errors is positioning error, resulting in the formula ÑD=ÑY +ÑB. The positioning reference displacement error and reference non-coincidence error that cause positioning error also include many elements respectively in calculation. For example, in addition to the clearance in NY, the geometric shape and position error of clamp positioning component (verticality, coincidence degree) also have influence on reference displacement.