CNC Controller User Manual

TAC2000 Series

This user manual describes all items concerning the operation of the system in detail as much as possible. However, it is impractical to give particular descriptions of all unnecessary and/or unavailable operations of the system due to the manual content limit, specific operations and other causes.Therefore, the operations not specified herein shall be considered impossible or unallowable.

FOREWORD

Dear user,

We are really grateful for your patronage and purchase of this **TAC2000 CNC system** made by TOMATECH (SHENZHEN) TECHNOLOGY CO.,LTD.

The user manual describes the programming, operation, installation and connection of this **TAC2000 system**. Please read it carefully before operation in order to get the safe and effective working.

Warning

This system can only be operated by authorized and qualified personnel as improper operations may cause accidents.

Please carefully read this user manual before use!

Note: The power supply installed on (in) the cabinet is exclusive to TOMATECH CNC systems.

The power supply form is forbidden to be used for other purposes. Otherwise, there may be extreme danger!

This user manualshall be kept by final user.

Notes

■ Delivery and storage

1. Packing box over 6 layers in pile is unallowed.
2. Never climb the packing box, neither stand on it, nor place heavy objects on it.
3. Do not move or drag the product by the cables connected with it.
4. Forbid collision or scratch to the panel and displayer.
5. Packing box should be protected from damping, insolation and raining.

■ Open packing box to check

1. Ensure things in packing box are the required ones.
2. Ensure the product is not damaged in delivery.
3. Ensure the parts in packing box are in accordance to the order.
4. Contact us in time if the product type is inconsistent with the order, there is shortof accessories, or product damage in delivery.

■ Connection

1. Only qualified persons can connect the system or check the connection.
2. The system must be earthed, its resistance must be less than 4 Ω and the ground wire cannot be replaced by zero wire.
3. Connection must be correct and firm to avoid the product tobe damaged or other unexpected result.
4. Connect with surge diode in the specified direction to avoid the damage to the system.
5. Switch off power supply before pulling out plug or opening electric cabinet.

■ Troubleshooting

1. Switch off power supply before troubleshooting or changing components.
2. Troubleshoot and then startup the system when there is short circuit or overload.
3. Do not switch on or off it frequently and an interval is 1 minute at least after the system is powered on again.

Announcement!

This manual describes various items as much as possible. However, operations allowable or unallowable cann't be explained one by one due to so many possibilities that may involve with, so the contents that are not specially stated in this manual shall be considered to be unavailable.

Warning!

Please read this user manual and a manual from machine builder completely before installation, programming and operation; do operate the system and machine according to user manuals, otherwise it may damage the system, machine, workpiece and even injure the operator.

Cautions!

Functions, technical indexes described in this user manual are only for the system. Actual functions and technical performance of machine tool with this CNC system are determined by machine builder's design, so refer to its user manual.

The system is employed with integrated machine control panel and the keys on machine control panel are defined by PLC program. Functions of keys in this user manual are for standard PLC program. Please notice it!

Refer to user manual from machine manufacturer about functions and meanings of keys on machine control panel.

All specification and designs are subject to change without further notice.

Volume Ⅰ Programming

Technical Specification, Product Type, Command and Program Format

Volume Ⅱ Operation

TAC2000 CNC Operation Use

Volume Ⅲ Installation and Connection

TAC2000 CNC Installation, Connection and Setting

Appendix

CNC Ladder Function Allocation, Alarm Message Table

Safety Responsibility

Manufacturer's safety responsibility

——The manufacturer should be responsible for the cleared or the controlled safety in the design and the structure of the CNC system and the accessories.

——The manufacturer should be responsible for the CNC system and the accessories.

——The manufacturer should be responsible for the message and the suggestion for the user.

User's safety responsibility

——The user should study and train the system safety operation, master the safety operation content.

——The user should be responsible for the danger caused by increasing, changing or modifying the CNC system, the accessories by itself.

——The user should be responsible for the danger because of the mistaken operation, regulation, maintenance, installation and storage.

CONTENTS

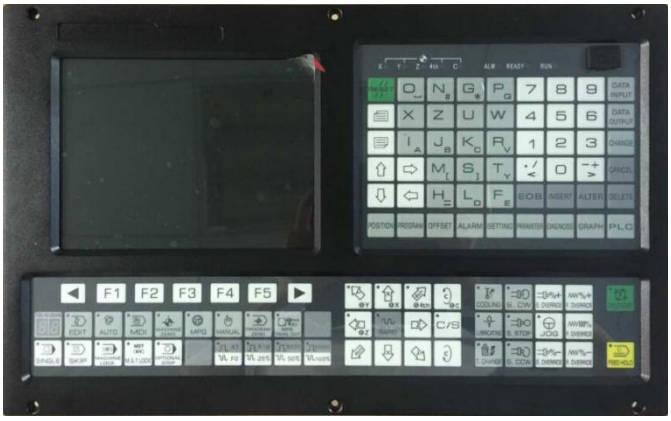
Volume Ⅰ
Programming

CHAPTER 1 PROGRAMMING

1.1 TAC2000 introduction

1.1.1 Product introduction

TAC2000 can control 5 feed axes(including C axis), 2 analog spindles, 1ms high-speed interpolation, 0.1μm control precision, which can obviously improve the machining efficiency, precision and surface quality.



TAC2000

X, Z, Y, 4th, 5th ; axis name and axis type of Y, 4th, 5th can be defined 1ms interpolation period, control precision 1μm, 0.1μm

Max. speed 60m/min（up to 24m/min in 0.1μm）

Adapting to the servo spindle to realize the spindle continuously positioning, rigid tapping, and the rigid thread machining

Built-in multi PLC programs, and the PLC program currently running can be selected Statement macro command programming, macro program call with parameter

Metric/inch programming, automatic tool setting, automatic chamfer, tool life management function

Chinese, English display can be selected by parameters.

USB interface, U disc file operation, system configuration and software 2-channel 0V ~ 10V analog voltage output, two-spindle control

1-channel MPG input, MPG function 36 input signals and 36 output signals

Appearance installation dimension, and command system are compatible with TAC2000

1.1.2 Technical specification

Controllable axes

Controllable axes: 5（X, Z, Y , 4th,5th） Link axes：4

Feed axis function

Least input unit: 0.001mm（0.0001inch）and 0.0001mm（0.00001inch） Least command unit：0.001mm（0.0001inch）and 0.0001mm（0.00001inch） Position command range: ±99999999× least command unit

Rapid traverse speed：max. speed 60m/min in 0.001mm command unit Rapid override: F0, 25%, 50%, 100%

Feedrate override: 0 ~ 150% 16 grades to tune

Interpolation mode: linear interpolation, arc interpolation(three-point arc interpolation), thread interpolation and rigid tapping

Automatic chamfer function

Thread function

General thread(following spindle)/rigid thread

Single/multi metric, inch straight thread, taper thread, end face thread, constant pitch thread and variable pitch thread

Thread run-out length, angle, speed characteristics can be set Thread pitch: 0.01mm ~ 500mm or 0.06 tooth/inch ~ 2540 tooth/inch

Acceleration/deceleration function

Cutting feed: front acceleration/deceleration linear, front acceleration/deceleration S back acceleration/deceleration linear,back acceleration/deceleration exponent Rapid traverse: linear,S type

Thread cutting: linear, exponential

Initial speed, termination speed, time of acceleration/deceleration can be set by parameters.

Spindle function

2-channel 0V ~ 10V analog voltage output, two-spindle control

1-channel spindle encoder feedback, spindle encoder line can be set（100p/r ~ 5000p/r）

Transmission ratio between encoder and spindle:（1 ~ 255）:（1 ~ 255）

Spindle speed: it is set by S or PLC, and speed range: 0r/min ~ 9999r/min Spindle override: 50% ~ 120% 8 grades tune

Spindle constant surface speed control

Rigid tapping

Tool function

Tool length compensation

Tool nose radius compensation（C） Tool wear compensation

Tool life management

Tool setting mode: fixed-point tool setting, trial-cut tool setting, reference point return tool setting, automatic tool setting

Tool offset execution mode: modifying coordinate mode, tool traverse mode

Precision compensation

Backlash compensation

Memory pitch error compensation

PLC function

Two-level PLC program，up to 5000 steps，the 1st program refresh period 8ms PLC program communication download

PLC warning and PLC alarm

Many PLC programs（up to 20PCS）, the PLC program currently running can be selected

Basic I/O：18 input signals /18 output signals

Man-machine interface

8.0″ wide screen LCD，resolution: 800X600

Chinese, English display

Planar tool path display

Real-time clock

Operation management

Operation mode: edit, auto, MDI, machine zero return, MPG/single, manual, program zero return

Multi-level operation privilege management

Alarm record

Program edit

Program capacity: 56MB ， 400 programs （ including subprograms and macro programs）

Edit function: program/block word search, modification, deletion,copying,pasting Program format: ISO command, statement macro command programming, relative coordinate, absolute coordinate and compound coordinate programming Program call: macro program call with parameter, 4-level program built-in

Communication function

RS232：two-way transmitting part programs and parameters, PLC program, system software serial upgrade

USB：U file operation, U file directly machining, PLC program, system software U upgrade

Safety function

Emergency stop Hardware travel limit Software travel check Data backup and recovery

G command table

Table 1-1

G code	Group	F	Whether high-speed and high-precision mode is valid	Function
*G0	01	G00 X_Y_Z_	T	Positioning (rapid traverse)
G01		G X_Y_Z_F_	T	Linear interpolation (cutting feed)
G02		G02 G03 X_Y_ R_ I F_;	T	Circular interpolation CW (clockwise)
G03				Circular interpolation CCW (counter clockwise)
G04	00	G04 P_ or G04 X_	F	Dwell, exact stop
G10		G10 L_N_P_R_	F	Programmable data input
*G11		G11	F	Programmable data input cancel
*G1	16	G12 X_Y_Z_ I_J_K_	F	Stored stroke detection ON
G13		G13	F	Stored stroke detection OFF
*G1	11	G15	F	Polar coordinate Command cancel
G16		G16	F	Polar coordinate Command
*G1 G18 G19	02	Written in blocks, used for circular interpolation and tool radius compensation	F	XY plane selection ZX plane selection YZ plane selection
G20	06	Must be specified in a single block	F	Input in inch
**G21				Input in metric
G22	09	G22 X_Y_Z_R_I_L_W_Q_V_D_F_ K	F	CCW inner circular groove rough milling
G23		G23 X_Y_Z_R_I_L_W_Q_V_D_F_K	F	CW inner circular groove rough milling
G24		G24 X_Y_Z_R_I_J_D_F_K_	F	CCW fine milling cycle within a circle
G25		G X_Y_Z_R_I_J_D_F_K_	F	CW fine milling cycle within a circle
G26		G26 X_Y_Z_R_I_J_D_F_K_	F	CCW outer circle finishing cycle

G27	00	G27	X_Y_Z_	T	Reference point return detection
G28		G28		T	Reference point return
G29				T	Return from reference point
G30		G30Pn		T	2nd, 3rd and 4th reference point return
G31		G31		F	Skip function
G32	09	G32 X_Y_Z_R_I_J_D_F_K_		F	CW outer circle finishing cycle
G33		G33X_Y_Z_R_I_J_L_W_Q_V_U_D_F_K		F	CCW rectangular groove rough milling
G34		G34X Y_Z_R_I_J_L_W_Q_V_U_D_F_K		F	CW rectangular groove rough milling
G35		G35 X_Y_Z_R_I_J_L_ U_D_F_K_		F	CCW rectangular groove rough milling cycle
G36		G36 X_Y_Z_R_I_J_L_ U_D_F_K_		F	CW rectangular groove rough milling cycle
G37		G37 X_Y_Z_R_I_J_L_ U_D_F_K_		F	CCW rectangular outside groove finishing cycle
G38		G38 X_Y_Z_R_I_J_L_ U_D_F_K_		F	CW rectangular outside groove finishing cycle
G39		G39		F	Corner offset circular interpolation
*G4	07	G17	G40 G41 G42	D_X_Y		Tool radius compensation cancel
G41				D_X_Z_	T	Left-hand tool radius compensation
G42				D_Y_Z_	T	Right-hand tool radius compensation
G43	08	G43		H_Z_	T	Tool length compensation in positive direction
G44					T	Tool length compensation in negative direction
*G49					T	Tool length compensation cancel
*G5	12	G50			T	Scaling cancel
G51		G51 X_ Y_ Z_ P_			T	Scaling
G53	00	Written in a program			T	Machine coordinate system selection
*G5	05	Written in a block, usually placed at the program beginning			T	Workpiece coordinate system 1
G55						Workpiece coordinate system 2
G56						Workpiece coordinate system 3
G57						Workpiece coordinate system 4
G58						Workpiece coordinate system 5
G59						Workpiece coordinate system 6

G60	00/01	G60 X_ Y_ Z_	T	Unidirection l positioning
G61	14	G61	T	Exact stop mode
G62			T	Automatic corner override
G63		G63	T	Tapping mode
*G64		G64	T	Cutting mode
G65	00	G65 H_P# i Q# j R# k	T	Macro program Command
G68	13	G68 X_ Y_ R_	T	Coordinate rotation
*G6			T	Coordinate rotation cancel
G73	09	G73 X_Y_Z_R_Q_F_;	F	Peck drilling cycle
G74		G74 X_Y_Z_R_P_F_;	F	Left-hand tapping cycle
G76		G76 X_Y_Z_Q_R_P_F_K_;	F	Fine boring cycle
*G80		Written in a block with other programs	F	Canned cycle cancel
G81		G81 X_Y_Z_R_F_;	F	Drilling cycle (spot drilling cycle)
G82		G82 X_Y_Z_R_P_F_;	F	Drilling cycle (counter boring cycle)
G83		G83 X_Y_Z_R_Q_F_;	F	Peck drilling cycle
G84		G84 X_Y_Z_R_P_F_;	F	Right-hand tapping cycle
G85		G85 X_Y_Z_R_F_;	F	Boring cycle
G86		G86 X_Y_Z_R_F_;	F	Boring cycle
G87		G87 X_Y_Z_R_Q_P_F_;	F	Back boring cycle
G88		G88 X_Y_Z_R_P_F_;	F	Boring cycle
G89		G89 X_Y_Z_R_P_F_;	F	Boring cycle
*G9	03	ritten into blocks	T	Absolute programming
G91				Incrementa programming
G92	00	G92 X_Y_Z_	T	Floating coordinate system setting
*G9	04	G94	T	Feed per minute
G95		G95	T	Feed per revolution
G96	15	G96S_	T	Constant surface speed control （cutting speed）
*G97		G97S_	T	Constant surface speed control cancel（cutting speed）
*G9	10	ritten into blocks	T	Return to initial plane in canned cycle
G99				Return to point R plane in canned cycle

1.1.3 Environment and conditions

TAC2000 storage delivery, working environment as follows:

Table 1-2

Item	Working conditions	Storage delivery conditions
Ambient temperature	0℃ ~ 45℃	-40℃ ~ +70℃
Ambient humidity	≤90%(no freezing)	≤95%(40℃)
Atmosphere pressure	86 kPa ~ 106 kPa	86 kPa ~ 106 kPa
Altitude	≤1000m	≤1000m

1.1.4 Power supply

TAC2000 can normally run in the following AC input power supply. Voltage: within(0.85 ~ 1.1)×rated AC input voltage (AC 220V); Frequency: 49Hz ~ 51Hz continuously changing

1.1.5 Guard

TAC2000 guard level is not less than IP20.

1.2 Program run

1.2.1 Sequence of program run

Running the current open program must be in Auto mode. TAC2000 cannot open two or more programs at the same, and runs only program any time. When the first block is open, the cursor is located in the heading of the first block and can be moved in Edit mode. In the run stop state in Auto mode, the program starts to run by the cycle start signal (is pressed or external cycle start signal)from a block pointed by current cursor, usually blocks are executed one by one according to their programming sequence, the program stops running till executing M02 or M30. The cursor moves along with program running and is located at the heading of the current block. Sequence and state of program running are changed in the followings:

- The program stops running after pressing or emergent stopbutton;
- The program stops running when the system or PLC alarms;
- The program runs and single block stops (the program run stops after the current block runs completely) in Edit, MDI mode, and then a block pointed by the current cursor starts running after the system switches into Auto mode, is pressed or external cycle start signal is switched on;
- The program stops running in Manual(Jog), Handwheel (MPG), Single Block, Program Reference Point Return, Machine Reference Point Return mode and it continuously runs from current position after the system is switched into Auto mode and is pressed or the external cycle start signal is switched on;
- The program pauses after pressing or the external cycle start signal is switched off, and it continuously runs from current position after pressing or the external cycle start signal is switched on;
- When Single Block is ON, the program pauses after every block is executed completely, and then it continuously runs from the next block after is pressed or the external cycle start signal is switched on;
- Block with "/" in the front of it is not executed when the block skipping switch is ON;
- The system skips to the target block to run after executing G65;
- Please see Section Three G Commands about execution sequence of G70~73;
- Call corresponding subprograms or macro program to run when executing M98 ; The system returns to main program to call the next block when executing M99(if M99 specifies a target block number, the system returns to it to run) after the subprograms or macro programs run completely;
- The system return to the first block to run and the current program is executed repetitively when M99 is executed in a main program.

1.2.2 Execution sequence of word

There are many words (G, X, Z, F, R, M, S, T and so on) and most of M, S, T is transmitted to PLC by NC explaining and others are directly executed by NC. M98, M99, S word used to specify the spindle speed r/min, m/min is directly executed by NC.

NC firstly executes G and then M commands when G codes and M00, M01, M02 and M30 are in the same block.

NC firstly executes G and then M commands(without transmitting M signal to PLC) when G codes and M98, M99 are in the same block.

When G codes and M, S, T executed by PLC are in the same block, PLC defines M, S, T and G to be executed simultaneously, or execute M, S ,T after G codes. Please see User Manual of machine manufacturer for execution sequence of commands.

Execution sequence of G, M, S, T in the same block defined by TAC2000 standard PLC program is as follows:

M3, M4, M8, M10, M12, M32, M41, M42, M43, M44, S□□, T□□□□ and G codes are executed simultaneously;

M5, M9, M11, M13, M33 after G codes are executed;

M00, M01, M02, M30 after other commands of current block are executed.

CHAPTER 2 MSTF COMMAND

2.1 M (miscellaneous function)

M command consists of command address M and its following 1 ~ 2 or 4 bit digits, used for controlling the flow of executed program or outputting M commands to PLC.

M □□□□

Command value(00~99, 9000~9999, the leading zero can be omitted)

Command address

M98, M99 is executed by NC separately and NC does not output M commands to PLC.

M02, M03 are for ending of programs defined by NC, and NC outputs M commands to PLC which can control spindle OFF, cooling OFF and so on.

M98, M99 are for calling programs, M02, M30 are for ending of program which are not changed by PLC. Other M commands output to PLC and their function are defined by PLC.

Please refer to User Manual from machine manufacturer.

There is only one M command in one block, otherwise the system alarms. Table

2-1 M commands to control program execution

Commands	Functions
M02	End of program
M30	End of program
M98	Call subprograms
M99	Return from a subprogram; it is executed repeatedly when the program ends in M99(the current program is not called by other programs)

2.1.1 End of program M02

Command format: M02 or M2

Command function: In Auto mode, after other commands of current block are executed, the automatic run stops, the amount of workpiece is added 1,the tool nose radius compensation is cancelled and the cursor return to the start of program (whether return to the start of program or not is defined by parameters).

2.1.2 End of program run M30

Command format: M30

Command function: In Auto mode, after other commands of current block are executed in M30, the automatic run stops, the amount of workpiece is added 1, the tool nose radius compensation is cancelled and the cursor returns to the start of program (whether the cursor return to the start of program or not is defined by parameters).

If No.005 Bit 4 is set to 0, the cursor does not return to the beginning of program, and the cursor returns immediately after the program is executed completely when No.005 Bit 4 is set to 1.

2.1.3 Subprogram call M98

Command format:

M98 P○○○○□□□□

Called subprogram number（ 0000 ~ 9999 ）. The leading zero of subprogram number can be omitted when the calling times is not input; the subprogram number must be with 4 digits when the calling times is input..

Call times: 1 ~ 9999. The calling times cannot be input when it is 1.

Command function: In Auto mode, after other commands are executed in M98, CNC calls subprograms specified by P, and subprograms are executed 9999 times at most. M98 is invalid in MDI mode.

2.1.4 Return from subprogram M99

Command format: M99 P○○○○

Executed block after returning to the main program is 0000 ~ 9999，and its leading zero can be omitted.

Command function: After other commands of current block in the subprogram are executed, the system returns to the main program and continues to execute next block specified by P, and calls a block following M98 of current subprogram when P is not input. The current program is executed repeatedly when M99 is defined to end of program (namely, the current program is executed without calling other programs). M99 is invalid in MDI mode.

Subprogram calls can be nested up to four levels as shown in Fig. 2-3.

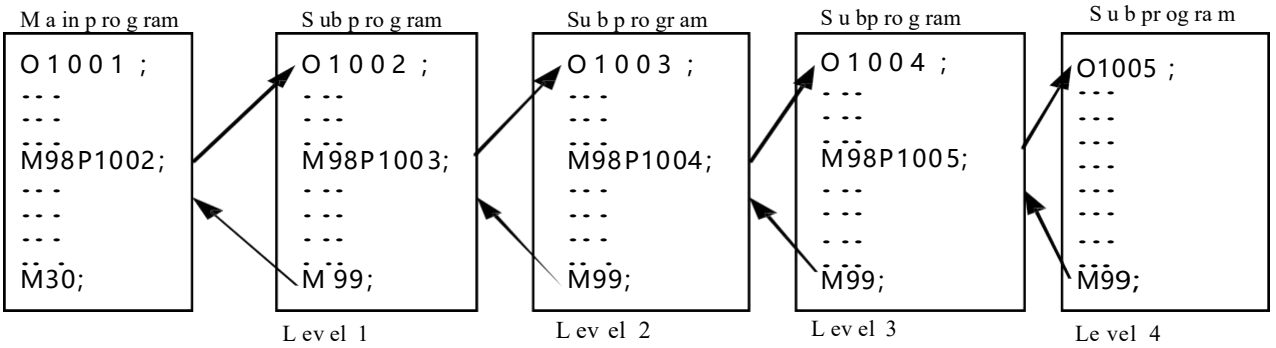


Fig. 2-3 Subprogram nesting

2.1.5 M commands defined by standard PLC ladder diagram

Other M commands are defined by PLC except for the above-mentioned ones(M02, M30, M98, M99, M9000 ~ M9999). The following M commands are defined by standard PLC, and TAC2000 Milling Machine CNC system is used for controlling machine tool. Refer to commands of machine manufacturer about functions, significations, control time sequence and logic of M commands.

M commands defined by standard PLC ladder diagram.

Table 2-2 M commands

Command	Function	Remark
M00	Program pause	
M01	Program optional stop	
M03	Spindle clockwise (CW)	Functions interlocked and states reserved
M04	Spindle counterclockwise (CCW)	
*M05	Spindle stop	
M08	Cooling ON	Functions interlocked and states reserved
*M09	Cooling OFF	
M16	Spindle tool release	Functions interlocked and states reserved
M17	Spindle tool clamp	
M18	Spindle orientation cancle	Functions interlocked and states reserved
M19	Spindle orientation	
M28	Rigid taping cancel	Functions interlocked and states reserved
M29	Rigid taping	
M32	Lubrication ON	Functions interlocked and states reserved
* M33	Lubrication OFF	
*M41, M42, M43, M44	Spindle automatic gear shifting	Functions interlocked and states reserved

Note: Commands with "*" defined by standard PLC is valid when power on.

2.1.6 Program stop M00

Command format: M00 or M0

Command function: After M00 is executed, the program stops and the system displays "*Pause*", and then the program continuously runs after the cycle start key is pressed.

2.1.7 Program optional stop M01

Command format: M01 or M1

Command function: in AUTO, MDI mode, it is valid. Press 

and its indicator lights and the system enters the optional stop state, at the moment, the program stops run and the system displays "PAUSE" after M01 is executed, after the cycle start key is pressed, the program continuously runs. When the program optional stop switch is not open, the program does not pause even if M01 runs.

2.1.8 Spindle CW, CCW and stop control M03, M04, M05

Command format: M03 or M3
M04 or M4;
M05 or M5.

Command function: M03: Spindle CW rotation;
M04: Spindle CCW rotation;
M05: Spindle stop.

Note: Refer to time sequence of output defined by standard PLC ladder in VOLUME III INSTALLATION & CONNECTION.

2.1.9 Cooling control M08, M09

Command format: M08 or M8;
M09 or M9;

Command function: M08: Cooling ON;
M09: Cooling OFF.

Note: Refer to time sequence and logic of M08, M09 defined by standard PLC ladder in VOLUME III INSTALLATION & CONNECTION.

2.1.10 Tool control M16, M17

Command format: M16;
M17;

Command function: M16: Spindle tool release;
M11: Spindle tool clamp.

Note: Refer to time sequence and logic of M10, M11 defined by standard PLC ladder in VOLUME III INSTALLATION & CONNECTION

2.1.11 Spindle orientation M18, M19

Command format: M18;
M19;

Command function: M18: Spindle orientation cancle;
M19: Spindle orientation.

Note: Refer to time sequence and logic of M12, M13 defined by standard PLC ladder in VOLUME III INSTALLATION & CONNECTION.

2.1.12 Rigid taping M28, M29

Command format：M28；
M29；

Command function：M28：Rigid taping cancle;
M29：Rigid taping .

2.1.13 Lubrication M32, M33

Command format： M32；
M33；

Command function：M32：Lubrication ON
M33：Lubrication OFF

Note: Refer to time sequence and logic of M32, M33 defined by standard PLC ladder in VOLUME III INSTALLATION & CONNECTION.

2.1.14 Spindle automatic gear change M41, M42, M43, M44

Command format：M4n；(n=1, 2, 3, 4)

Command function：When the system executes M4n, the spindle changes to gear n.

Note: Refer to time sequence and logic of M41, M42, M43, M44 defined by standard PLC ladder in VOLUME III INSTALLATION&CONNECTION.

2.2 Spindle function

S command is used for controlling spindle speed and this TAC2000 has two modes to control it:

Spindle speed switching value control: S□□(2 digits command value)is executed by PLC, and PLC outputs switching value signal to machine tool to change spindle speed with grades.

Spindle speed analog voltage control: S□□□□(4 digits command value)specifies actual speed of spindle and NC outputs 0~10V analog voltage signal to spindle servo or converter to realize stepless spindle speed.

2.2.1 Spindle speed switching value control

Spindle speed is controlled by switching value when No.001 BIT4 is set to 0. There is only one S command in a block, otherwise the system alarms.

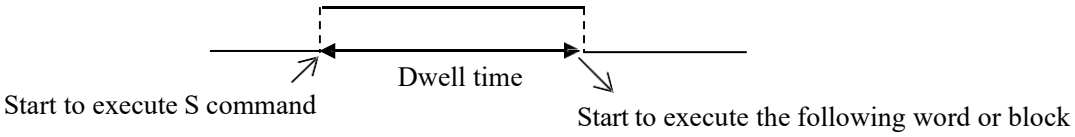
Their executing sequence is defined by PLC when S command and word for moving function are in the same block. Please refer to *User Manual* from machine manufacturer.

When spindle speed is controlled by switching value, TAC2000 Milling CNC system is used for machine tool and the time sequence and logic of executing S command is according to *User Manual* from machine manufacturer. Refer to S command defined by standard PLC of TAC2000 as follows:

Command format: S□□

└── 00 ~ 04(the leading zero can be omitted): No.1 ~ No.4 gear of spindle speed is controlled by switching value.

In spindle speed switching value control mode, after S signal transmits to PLC, the system dwells time defined by No.081, then return FIN signal, and the dwell time is called runtime of Scommand.



S01, S02, S03, S04 output are reserved when resetting CNC.

S1 ~ S4 output are invalid when CNC is switched on. The corresponding S signal output is valid and reserved, and others are cancelled at the same time when executing one of S01, S02, S03, S04. When executing S00, S1 ~ S4 output are cancelled and only one of S1 ~ S4 is valid at the same time.

2.2.2 Spindle speed analog voltage control

Spindle speed is controlled by analog voltage when No.001 BIT4 is set to 1. Command format: S OOOO

└── 0000 ~ 9999 (the leading zero can be omitted.):Spindle speed analog voltage control

Command function: The spindle speed is defined, and the system outputs 0 ~ 10V analog voltage to control spindle servo or converter to realize the stepless timing. S command value is not reserved, and it is 0 after the system is switched on.

When the spindle speed analog voltage control is valid, there are 2 methods to input the spindle speed: the spindle fixed speed is defined by S command(r/min), and is invariant without changing S command value, which is called constant speed control(G97 modal); other is the tangent speed of tool relative to the outer circle of workpiece defined by S command, which is called constant surface speed control (G96 modal), and the spindle speed is changed along with the absolute coordinates value of X absolute coordinates in programming path when cutting feed is executed in the constant surface speed.

Please refer to Section 2.2.3.

The system can execute 4 gears spindle speed. Count the analog voltage value corresponding to the specified speed according to setting value(corresponding to No.210 ~ No.213) of max. spindle speed (analog voltage is 10V)of current gear, and then output to spindle servo or converter to ensure that the spindle actual speed and the requirement are the same.

After the system is switched on, the analog output voltage is 0V. The analog output voltage is reserved (except that the system is in cutting feed in the surface speed control mode and the absolute value of X absolute coordinates is changed) after S command is executed. The analog output voltage is 0V after S0 is executed. The analog output voltage is reserved when the system resets and emergently stops.

2.2.3 Spindle override

When the spindle speed analog voltage control is valid, the spindle actual speed can be tuned real time by the spindle override and is limited by max spindle speed of current gear after the spindle override is tuned, and it also limited by limited values of max. and min. spindle speed in constant surface speed control mode.

The system supplies 8 steps for spindle override (50% ~ 120% increment of 10%). The actual steps and tune of spindle override are defined by PLC ladder and introductions from machine manufacturer should be referred when using it. Refer to the following functions of TAC2000 standard PLC ladder.

The spindle actual speed specified by TAC2000 standard PLC ladder can be tuned real time by the spindle override tune key at 8 steps in 50% ~ 120% and it is not reserved when the spindle override is switched off. Refer to the operations of spindle override in **VOLUME** II **OPERATION**.

2.3 Spindle function

By specifying a numerical value (up to 8 digits) following address T, the tools on the machine can be selected.

Only one T code can be specified in a block by principle. However, if no alarm occurs when a block contains two or more instructions of the same group via setting, the last T code takes effect. Refer to the manual provided by the tool machine builder for the digits after address T and the corresponding machine operation of T code.

When a movement instruction and a T code are specified in the same block, the instructions are executed simultaneously.

When the T code and tool change instruction are in the same block, the T code is executed before tool change instruction. If they are not in the same block, M06 executes the T code specified by the last program.

Such as the program below：

```
O00010；
N10 T2M6；          Spindle tool number is T2
N20 M6T3；          Spindle tool number is T3
N30 T4；            Spindle tool number is
N40 M6；            T3 Spindle tool number
N50 T5；            is T4 Spindle tool
N60 M30             number is T4
%
```

After the tool change, the spindle tool number is T4.

CHAPTER 3 G COMMANDS

3.1 Commands

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

G □□

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

Command address G

G words are divided into 9 groups (00, 01, 02, 03, 05、 06、 07、 16、 21). Except that commands in the group 01 and 00 are not in the same block, G words in the different groups can be input to the same block and the last one is valid when two or more G words in the same group are input. The words in the different groups without the same parameter (word) can be in the same block and their functions are valid without sequence at the same time. The system alarms when G words do not belong to Table 3-1 or they are optional functions without being supplied.

Table 3-1-2 G command list

G code	Group	Format	Whether high-speed and high-precision mode is valid	Function
*G00	01	G00 X_Y_Z_	T	Positioning (rapid traverse)
G01		G01 X_Y_Z_F_	T	Linear interpolation (cutting feed)
G02		G02 G03 X_Y_ R_ I_J_ F_;	T	Circular interpolation CW (clockwise)
G03			T	Circular interpolation CCW (counter clockwise)
G04	00	G04 P_ or G04 X_	F	Dwell, exact stop
G10		G10 L_N_P_R_	F	Programmable data input
*G11		G11	F	Programmable data input cancel
*G12	16	G12 X_Y_Z_ I_J_K_	F	Stored stroke detection ON
G13		G13	F	Stored stroke detection OFF

*G15	11	G15		F	Polar coordinate Command cancel
G16		G16		F	Polar coordinate Command
*G17 G18 G19	02	Written in blocks, used for circular interpolation and tool radius compensation		F	XY plane selection ZX plane selection YZ plane selection
G20	06	Must be specified in a single block		F	Input in inch
*G21					Input in metric
G22	09	G22 X_Y_Z_R_I_L_W_Q_V_D_F_ K		F	CCW inner circular groove rough milling
G23		G23 X_Y_Z_R_I_L_W_Q_V_D_F_K		F	CW inner circular groove rough milling
G24		G24 X_Y_Z_R_I_J_D_F_K_		F	CCW fine milling cycle within a circle
G25		G25 X_Y_Z_R_I_J_D_F_K_		F	CW fine milling cycle within a circle
G26		G26 X_Y_Z_R_I_J_D_F_K_		F	CCW outer circle finishing cycle
G27	00	G27	X_Y_Z_	T	Reference point return detection
G28		G28		T	Reference point return
G29		G29		T	Return from reference point
G30		G30Pn		T	2nd, 3rd and 4th reference point return
G31		G31		F	Skip function
G32	09	G32 X_Y_Z_R_I_J_D_F_K_		F	CW outer circle finishing cycle
G33		G33X_Y_Z_R_I_J_L_W_Q_V_U_D_F_K		F	CCW rectangular groove rough milling
G34		G34X Y_Z_R_I_J_L_W_Q_V_U_D_F_K		F	CW rectangular groove rough milling
G35		G35 X_Y_Z_R_I_J_L_ U_D_F_K_		F	CCW rectangular groove rough milling cycle
G36		G36 X_Y_Z_R_I_J_L_ U_D_F_K_		F	CW rectangular groove rough milling cycle
G37		G37 X_Y_Z_R_I_J_L_ U_D_F_K_		F	CCW rectangular outside groove finishing cycle
G38		G38 X_Y_Z_R_I_J_L_ U_D_F_K_		F	CW rectangular outside groove finishing cycle
G39	00	G39		F	Corner offset circular interpolation

G code	Group	Format			Whether high-speed and high-precision mode is valid	Function
*G40	07	G17	G40 G41 G42	D_X_Y_	T	Tool radius compensation cancel
G41		G18		D_X_Z_	T	Left-hand tool radius compensation
G42		G19		D_Y_Z_	T	Right-hand tool radius compensation
G43	08	G43		H_Z_	T	Tool length compensation in positive direction
G44		G44			T	Tool length compensation in negative direction
*G49		G49			T	Tool length compensation cancel
*G50	12	G50			T	Scaling cancel
G51		G51 X_ Y_ Z_ P_			T	Scaling
G53	00	Written in a program			T	Machine coordinate system selection
*G54	05	Written in a block, usually placed at the program beginning			T	Workpiece coordinate system 1
G55						Workpiece coordinate system 2
G56						Workpiece coordinate system 3
G57						Workpiece coordinate system 4
G58						Workpiece coordinate system 5
G59						Workpiece coordinate system 6
G60	00/01	G60 X_ Y_ Z_			T	Unidirectional positioning
G61	14	G61			T	Exact stop mode
G62		G62			T	Automatic corner override
G63		G63			T	Tapping mode
*G64		G64			T	Cutting mode
G65	00	G65 H_P# i Q# j R# k			T	Macro program Command
G68	13	G68 X_ Y_ R_			T	Coordinate rotation
*G69		G69			T	Coordinate rotation cancel
G73	09	G73 X_Y_Z_R_Q_F_;			F	Peck drilling cycle
G74		G74 X_Y_Z_R_P_F_;			F	Left-hand tapping cycle
G76		G76 X_Y_Z_Q_R_P_F_K_;			F	Fine boring cycle
*G80		Written in a block with other programs			F	Canned cycle cancel

G code	Group	Format	Whether high-speed and high-precision mode is valid	Function
G81		G81 X_Y_Z_R_F_;	F	Drilling cycle (spot drilling cycle)
G82		G82 X_Y_Z_R_P_F_;	F	Drilling cycle (counter boring cycle)
G83		G83 X_Y_Z_R_Q_F_;	F	Peck drilling cycle
G84		G84 X_Y_Z_R_P_F_;	F	Right-hand tapping cycle
G85		G85 X_Y_Z_R_F_;	F	Boring cycle
G86		G86 X_Y_Z_R_F_;	F	Boring cycle
G87		G87 X_Y_Z_R_Q_P_F_;	F	Back boring cycle
G88		G88 X_Y_Z_R_P_F_;	F	Boring cycle
G89		G89 X_Y_Z_R_P_F_;	F	Boring cycle
*G90	03	Written into blocks	T	Absolute programming
G91				Incremental programming
G92	00	G92 X_Y_Z_	T	Floating coordinate system setting
*G94	04	G94	T	Feed per minute
G95		G95	T	Feed per revolution
G96	15	G96S_	T	Constant surface speed control （cutting speed）
*G97		G97S_	T	Constant surface speed control cancel（cutting speed）
*G98	10	Written into blocks	T	Return to initial plane in canned cycle
G99				Return to point R plane in canned cycle

Note 1: If modal Commands and non-modal Commands are in the same block, the non-modal commands take precedence. At the same time, the corresponding modes are changed according to the other modal Commands in the same block, but not executed.

Note 2: For the G code with sign *, when the power is switched on, the system is in the state of this G code (some G codes are determined by bit parameter NO:31#0~7).

Note 3: The G codes of group 00 are all non-modal G codes except G10, G11, G92.

Note 4: An alarm occurs if G codes not listed in this table are used or G codes that cannot be selected are specified.

Note 5: G codes from different groups can be specified in a block, but 2 or more G codes from the same group can not be specified in a block by principle. If no alarm occurs when two or more G codes in the same group are in a block after parameter setting, the latter G code functions.

Note 6: If a G code of group 01 is in the same block with a G code of group 09, the G code of group 01 prevails. In canned cycle mode, if G codes from 01 group are specified, the canned cycle will be cancelled automatically and the system turns into G80 state.

Note 7: G codes are represented by group numbers respectively based on their types. Whether the G codes of each group are cleared after reset or emergency stop is determined by bit parameter NO:46#1~7 and NO:37#0~7.

Note 8: If the rotation scaling Command and the Command of group 01 or that of group 09 share the same block, the rotation scaling Command will be taken, and the modes of group 01 or group 09 are changed. If the rotation scaling Command and the Command of group 00 share the same block, an alarm occurs.

3.2 Simple G Codes

3.2.1 Rapid Positioning G00

Code format： **G00 X_Y_Z_**

Function: G00 command. The tool moves to the position in the workpiece system specified with the absolute or an incremental command at a rapid traverse speed. The bit parameter **NO:14#0**. sets to select one of the following two tool paths (Fig. 3-2-1-1).

1. Linear interpolation positioning: The tool path is the same as linear interpolation (G01). The tool is positioned within the shortest time at a speed not more than the rapid traverse speed of each axis. 1.
2. Nonlinear interpolation positioning: The tool is positioned at the rapid traverse speed of each axis respectively. The tool path is usually not straight.

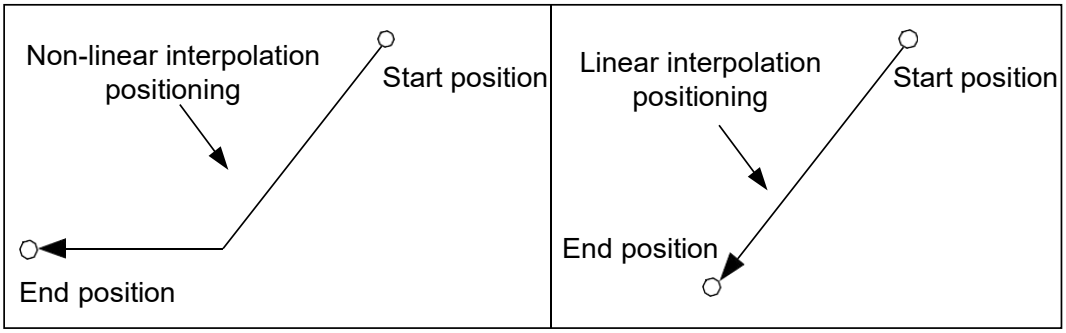


Fig. 3-2-1-1

Explanation:

1. After G00 is executed, the system changes the current tool move mode for G00 mode. Whether the default mode is G00 (parameter value is 0) or G01 (parameter value is 1) after power-on is set by bit parameter **No.048#0.**
2. With no positioning parameter specified, the tool does not move and the system only changes the mode of the current tool movement for G00.
3. G00 is the same as G0.
4. The G0 speed of axes X, Y, Z and 4th is set by data parameters **P90~P92**.

Limitations:

The rapid traverse speed is set by parameter. The speed F specified in the G0 Command is the cutting speed of the following machining blocks.

Example:

G0 X0 Y10 F800； Feeding at the speed set by system parameter G1 X20

Y50； Using the feedrate of F800

The rapid positioning speed is adjusted by the keys F0%，25%，50%，100% on the operation panel (see fig. **3-2-1-2**). The speed to which F0 corresponds is set by data parameter **P85** and it is common to all axes.

Fig. 3-2-1-2 Keys for rapid feedrate override

Note: Note the position of the worktable and workpiece to prevent tool collision.

3.2.2 Linear Interpolation G01

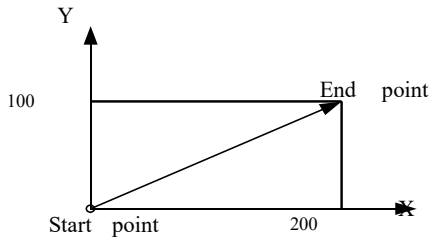
Code format: **G01 X_ Y_ Z_ F_**

Function: The tool moves to the specified position along a straight line at the feedrate (mm/min) specified by parameter.

Explanation:

1. X_ Y_ Z_ are the coordinates of the end point. Since they are related to the coordinate system, please see sections 3.3.1～3.3.3.
2. The feedrate specified by F keeps effective till a new F value is specified. The feedrate specified by F code is calculated by an interpolation along a straight line. If F code is not specified in a program, the default F value at system Power On is used (see data parameter P83 for details).

Program example (Fig. 3-2-2-1)



G01 X200 Y100 F200 ;

Note: feedrate of each axis is as follows:
G01 Xα Yβ ZγFf;
In this program:

Feedrate of X axis $F_X = \frac{\alpha}{L} \times f$

Feedrate of Y axis: $F_Y = \frac{\beta}{L} \times f$

Feedrate of Z axis: $F_Z = \frac{\gamma}{L} \times f$

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

Fig. 3-2-2-1

Note:

1. **All code parameters are positioning parameters except for F code. The upper limit of feedrate F is set by data parameter P86. If the actual cutting feedrate (after using feedrate override) exceeds the upper limit, it is clamped to the upper limit (unit: mm/min). The lower limit of the feedrate F is set by data parameter P87. If the actual cutting feedrate (after using feedrate override) exceeds the lower limit, it is clamped to the lower limit (unit: mm/min).**
2. **The tool does not move when no positioning parameter is specified behind G01, and the**

system only changes the mode of the current tool movement mode for G01. By altering the system bit parameter NO:48#0, the system default mode at power-on can be set to G00 (value is 0) or G01 (value is 1).

3.2.3 Circular (Helical) Interpolation G02/G03

A. Circular interpolation G02/G03

Prescriptions for G02 and G03:

- Arc rotation direction（G02，G03）
- Circular interpolation plane（G17, G18, G19）
- Circle center coordinate or radius, which thus leads to two Command formats: Circle center coordinate I, J ,K or radius R programming.

Only the three points above are all determined, could the interpolation operation be done in coordinate system.

The circular interpolation can be done by the following Commands to make the tool move along an arc, as is shown below:

Arc in XY plane

$$G17 \begin{matrix} G02 \\ G03 \end{matrix} X\_Y\_ \begin{matrix} R\_ \\ I\_J\_ \end{matrix} F\_;$$

Arc in ZX plane

$$G18 \begin{matrix} G02 \\ G03 \end{matrix} X\_Z\_ \begin{matrix} R\_ \\ I\_K\_ \end{matrix} F\_;$$

Arc in YZ plane

$$G19 \begin{matrix} G02 \\ G03 \end{matrix} Y\_Z\_ \begin{matrix} R\_ \\ J\_K\_ \end{matrix} F\_;$$

Table 3-2-3-1

Item	Content	Command	Meaning
1	Plane specification	G17	Arc specification on XY plane
		G18	Arc specification on ZX plane
		G19	Arc specification on YZ plane
2	Rotation direction	G02	CW rotation
		G03	CCW rotation
3	End point position G90 mode	Two axes of X,Y and Z axes	End point coordinate in workpiece coordinate system
	End point position G91 mode	Two axes of X,Y and Z axes	Coordinate of end point relative to start point
4	Distance from start point to circle center	Two axes of I,J and K axes	Coordinate of circle center relative to start point

	Arc radius	R	Arc radius
5	Feedrate	F	Arc tangential speed

CW and CCW on XY plane (ZX plane or YZ plane) refer to the directions viewed in the positive-to-negative direction of the Z axis (Y axis or X axis) in the right-hand Cartesian coordinate system, as is shown in Fig. 3-2-3-1.

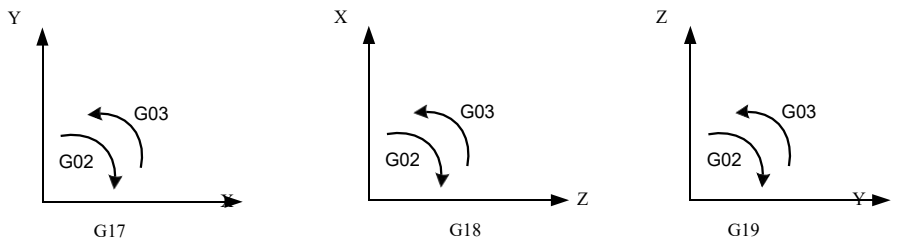


Fig. 3-2-3-1

The default plane mode at power-on can be set by bit parameters NO:48#1 and #2.

The end point of an arc can be specified by parameter words X, Y and Z. It is expressed as absolute values in G90, and incremental values in G91. The incremental values are the coordinates of the end point relative to the start point. The arc center is specified by parameter words I, J, K, corresponding to X, Y, Z respectively. Either in absolute mode G90, or in incremental mode G91, parameter values of I, J, K are the coordinates of the circle center relative to the arc start point (for simplicity, the circle center coordinates with the start point taken as the origin temporarily). They are the incremental values with signs. See Fig. 3-2-3-2.

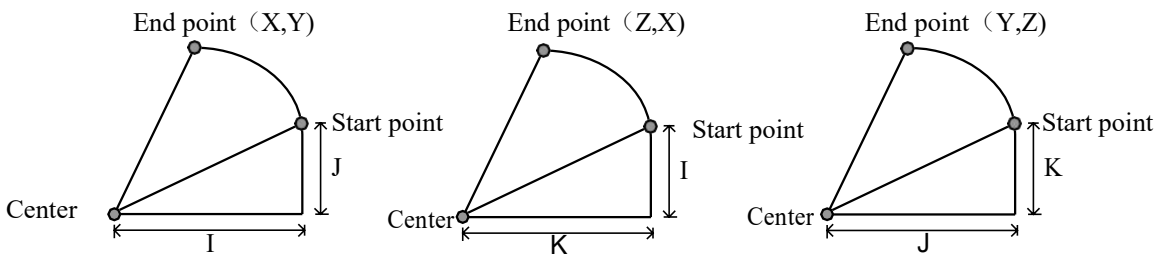


Fig. 3-2-3-2

I, J, K are assigned with a sign according to the direction of the circle center relative to the start point. The circle center can also be specified by radius R besides I, J and K.

G02 X_ Y_ R_ ;

G03 X_ Y_ R_ ;

1. Two arcs can be drawn as follows; one arc is more than 180°, and the other one is less than 180°. For the arc more than 180°, its radius is specified by a negative value.

   (Example: Fig. **3-2-3-3)** ① When arc is less than 180°,

   G91 G02 X60 Y20 R50 F300 ;

   ② When arc is more than 180°, G91

   G02 X60 Y20 R-50 F300 ;

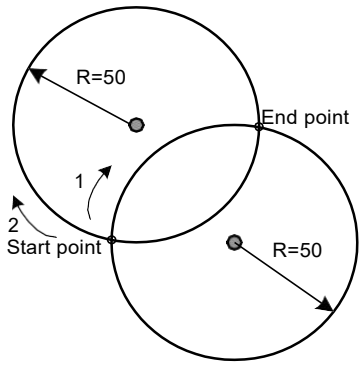


Fig. 3-2-3-3

2. The arc equal to 180° can be programmed either by I, J and K, or by R. Example:

G90 G0 X0 Y0;G2 X20 I10 F100;

Equal to G90 G0 X0 Y0;G2 X20 R10 F100 Or

G90 G0 X0 Y0;G2 X20 R-10 F100

Note: For the arc of 180°, the arc path is not affected whether the value of R is positive or negative.

3. For the arc equal to 360°, only I, J and K can be used for programming.

(Program example):

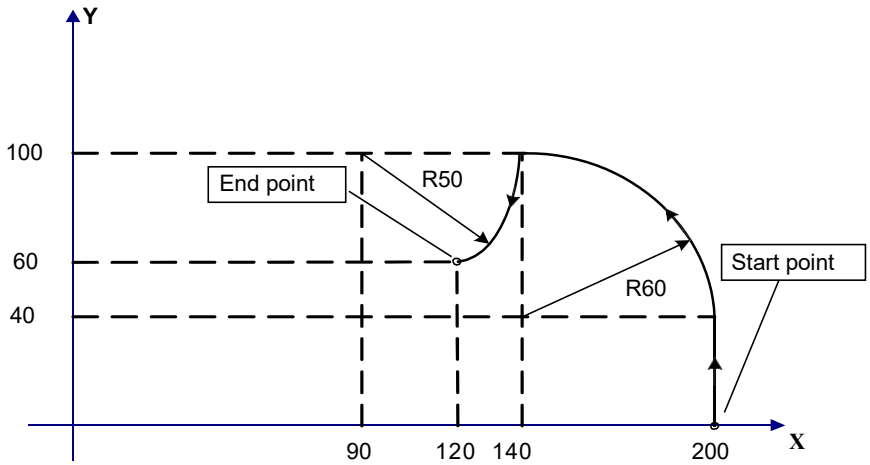


Fig. 3-2-3-4

The tool path programming for Fig. 4-2-3-4 is as follows: 1.

Absolute programming

G90 G0 X200 Y40 Z0； G3

X140 Y100 R60 F300； G2

X120 Y60 R50；

Or

G0 X200 Y40 Z0；

G90 G3 X140 Y100 I-60 F300； G2

X120 Y60 I-50；

2. Incremental programming G0

G90 X200 Y40 Z0；

G91 G3 X-60 Y60 R60
F3000； G2 X-20 Y-40
Or R50；

G0 G90 X200 Y40 Z0；
G91 G3 X-60 Y60 I-60
Restrictions: F300； G2 X-20 Y-40 I-
50；

1. If addresses I, J, K and R are specified simultaneously in a program, the arc specified by R takes precedence, and others are ignored.
2. If neither arc radius parameter nor the parameter from the start point to the circle center is specified, an alarm is issued in the system.
3. A full circle can only be interpolated by parameters I, J, K from start point to circle center rather than parameter R.
4. Pay attention to the setting for selecting the coordinate plane when the circular interpolation is being done.
5. If X, Y, Z are all omitted (i.e., the start point and the final point coincides), and R is specified (e.g. G02R50), the tool does not move.

B. Helical interpolation

Command format：G02/G03

Arc on XY plane

G17 { G02 / G03 } X_p_ Y_p_ Z_p. { I_J_ / R_ } F

Arc on ZX plane

G18 { G02 / G03 } X_p_ Y_p_ Z_p. { I_K_ / R_ } F

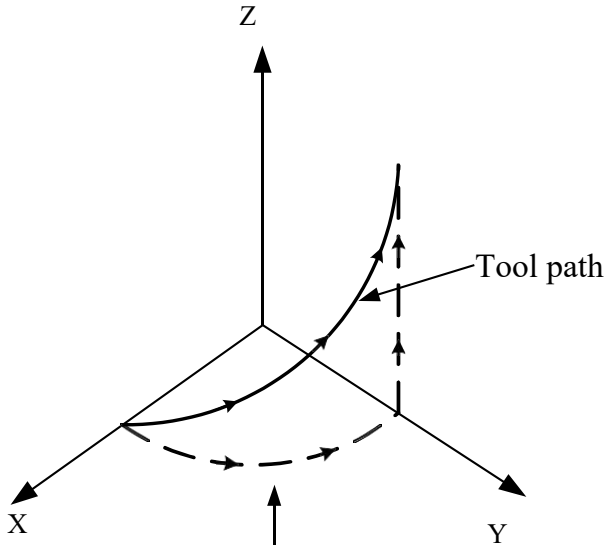
Arc on YZ plane

G19 { G02 / G03 } X_p_ Y_p_ Z_p. { J_K_ / R_ } F

Fig. 3-2-3-5

Function: It is used to move the tool to a specified position from the current position at a feedrate specified by parameter F in a helical path.

Explanation:



The feedrate along the circumference of two circular interpolated axes is the specified feedrate.

Fig. 3-2-3-6

The first two command parameters are positioning parameters. The parameter words are the names of two axes (X, Y or Z) in the current plane. These two positioning parameters specify the position which the tool is to go to. The parameter word of the third command parameter is a linear axis except the circular interpolation axis, and its value is the helical height. The meanings and restrictions for other command parameters are identical with those of circular interpolation.

If the circle can not be machined according to the specified command parameter, the system will give an error message. After the execution, the system changes the current tool traversing mode for G02/G03 mode.

The feedrate along the circumference of two circular interpolation axes is specified. The specification method is to simply add a moving axis which is not a circular interpolation axis. The feedrate along a circular arc is specified by F command. Thus the feedrate of the linear axis is as follows:

$$F_C = F * \frac{\text{Length of liner axis}}{\text{Length of circular arc}}$$

Determine the feedrate to make the linear axis feedrate not exceed any limit.

Restrictions:

Pay attention to the setting for selecting the coordinate plane when the helical interpolation is being done.

3.2.4 Absolute/incremental programming G90/G91

Command format:: G90/G91

Function: There are 2 commands for axis moving, including the absolute command and the incremental command.

The absolute command is a method of programming by the axis moving end point coordinates. The end position involves the concept of coordinate system, please refer to sections 2.4.1～2.4.4.

The incremental command is a method of programming by the axis relative moving amount.

The incremental value is irrelevant with the coordinate system concerned. It only requires the moving direction and distance of the end point relative to the start point.
The absolute command and the incremental command are G90 and G91 respectively.

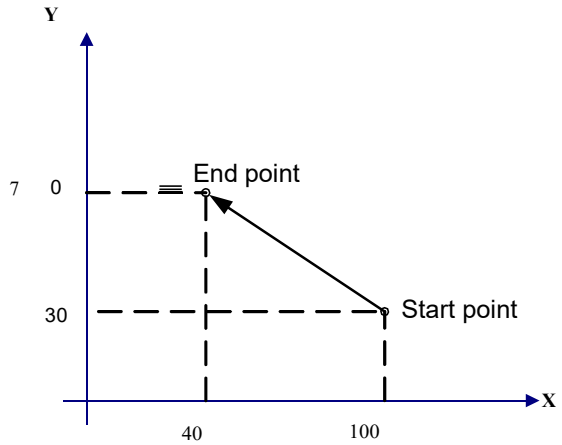


Fig. 3-2-4-1

For the movement from start point to end point in Fig. 4-2-4-1, the programming by using absolute command G90 and incremental command G91 is as follows:

G90 G0 X40 Y70；

Or G91 G0 X－60 Y40 ；

The same action can be performed with the two methods, users thus can choose either one of them as required.

Explanation:

- With no command parameter. It can be written into the block with other commands.
- G90 and G91 are the modal values in the same group, i.e., if G90 is specified, the mode is always G90 (default) till G91 is specified. If G91 specified, the mode is always G91 till G90 specified.

System parameters:

Whether the default positioning parameter is G90 mode **(**parameter is 0**)** or G91 mode (the parameter is 1**)** at Power On is set by bit parameter **N0:31#4.**

3.2.5 Dwell (G04)

Format: G04 X_ or P_

Function: G40 is for dwell operation. It delays the specified time before executing the next block. In cutting mode G64, it is used for exact stop check. The dwell per revolution in Feed per Revolution mode G95 can be specified by bit parameter No.34#0.

Table 3-2-5-1 Value range of dwell time (commanded with X)

Least command increment	Value range	Unit of dwell time
No.5#1=0	0.001～9999.999	S or rev
No.5#1=1	0.0001~9999.9999	

Table 4-2-5-2 Value range of dwell time commanded with P)

Least command increment	Value range	Unit of dwell time

No.5#1=0	1~99999.999	0.001s or rev
No.5#1=1	1~99999.999	0.0001s or rev

Explanation:

1. G04 is non-modal command, which is only effective in the current block.
2. If parameters X and P appear simultaneously, parameter X is effective.
3. An alarm occurs if the values of X and P are negative.
4. Dwell is not executed if neither X nor P is specified.

3.2.6 Single-direction positioning（G60）

Format: G60 X_ Y_ Z_

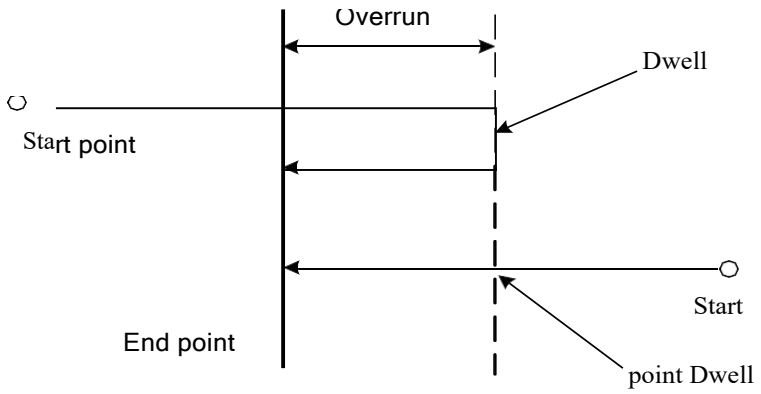


Fig. 3-2-6-1

Function: For accurate positioning without machine backlash, G60 can be used for accurate positioning in a single direction.

Explanation:

G60 is a non-modal G code (it can be set to a modal value by bit parameter **NO:52#2**), which is only effective in a specified block.

Parameters X, Y and Z represent the coordinates of the end point in absolute programming; and the moving distance of the tool in incremental programming. In tool offset mode, the path of single-direction positioning is the one after tool compensation when G60 is used.

The overrun marked in above figure can be set by system parameters P352， P353， P354 , and the dwell time can be set by parameter P334. The positioning direction can be determined by setting positive or negative overrun. Refer to system parameter for details.

Example 1：

G90 G00 X-10 Y10；

G60 X20 Y25； (1)

If the system parameter **P350** = 1, **P351** = -8, **P352** = 5;for statement (1), the tool path is AB→dwell for 1s→BC

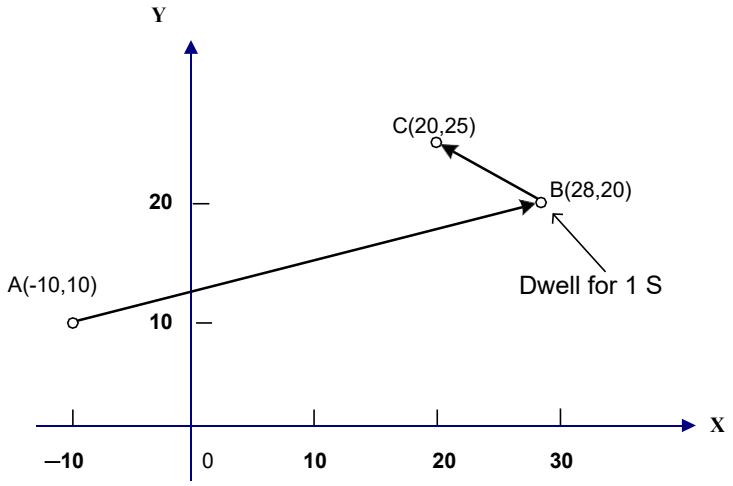


Fig. 3-2-6-2

System parameter:

Table 3-2-6-1

P334	Dwell time of single-direction positioning (unit: s)
P335	Overrun and single-direction positioning direction in X axis（unit:mm）
P336	Overrun and single-direction positioning direction in Y axis（unit:mm）
P337	Overrun and single-direction positioning direction in Z axis（unit:mm）
P338	Overrun and single-direction positioning direction in 4th axis（unit:mm）

Note 1: The signs of parameters P351 ～P354 are for the direction of single-direction positioning, and their values for the overrun.

Note 2: If overrun>0, the positioning direction is positive. Note 3:

If overrun<0, the positioning direction is negative.

Note 4: If overrun=0, no single-direction positioning is available.

3.2.7 On-line modification for system parameters (G10)

Function: It is used to set or modify the values of tool radius, length offset, external zero offset, workpiece zero offset, additional workpiece zero offset, data parameter, bit parameter and so on in a program.

Format:

G10 L50 N_P _R_； Setting or modifying the bit parameter G10 L51 N_ R_； Setting or modifying the data parameter G11； Canceling the parameter input mode

Parameter definition:

N：Parameter number. Sequence number to be modified. P：Parameter bit number. Bit number to be modified.

R：Value. Parameter value after being modified.

The values can also be modified by following codes. Refer to relative sections for details:

G10 L2 P_X_Y_Z_A_B_;	Setting or modifying external zero offset or workpiece zero offset G10
L10 P_R_;	Setting or modifying length offset
G10 L11 P_R_;	Setting or modifying length wear
G10 L12 P_R_;	Setting or modifying radius offset
G10 L13 P_R_;	Setting or modifying radius wear
G10 L20 P_ X_Y_Z_A_B_;	Setting or modifying additional workpiece zero offset

Note 1: In parameter input mode, no NC statement can be specified except annotation statement.
Note 2: G10 must be specified in a separate block or an alarm occurs. Please note that the parameter input mode must be cancelled by G11 after G10 is used.
Note 3: The parameter value modified by G10 must within the range of system parameter, otherwise, an alarm occurs.
Note 4: Modal codes of canned cycle must be cancelled prior to G10 execution, otherwise an alarm occurs.
Note 5: Those parameters which take effect after Power OFF and then On are unavailable to be modified by G10.

3.2.8 Workpiece coordinate system G54～G59

Function: for specifying the current workpiece coordinate system. The workpiece coordinate system is selected by specifying G codes of workpiece coordinate system in a program.

Format: G54～G59

Explanation:

1. With no code parameter.
2. The system itself is capable of setting 6 workpiece coordinate systems, any one of which can be selected by codes G54~G59.

   G54---------- Workpiece coordinate system 1
   G55---------- Workpiece coordinate system 2
   G56---------- Workpiece coordinate system 3
   G57---------- Workpiece coordinate system 4
   G58---------- Workpiece coordinate system 5
   G59---------- Workpiece coordinate system 6
3. At Power On, the system displays the workpiece coordinate codes G54～G59, G92 or additional workpiece coordinate system ever executed before Power Off.
4. When different workpiece coordinate systems are called in a block, the axis to move is positioned to the coordinate of the new coordinate system; for the axis not to move, its coordinate shifts to the corresponding coordinate in the new coordinate system, with its actual position on the machine tool unchanged.

   Example: The corresponding machine tool coordinate for G54 coordinate system origin is (10,10,10)
   The corresponding machine coordinate for G55 coordinate system origin is (30，30， 30)

When the program is executed in order, the absolute coordinates and machine coordinates of the end point l are displayed as follows:

Table 3-2-8-1

Program	Absolute coordinate	Machine coordinate
G0 G54 X50 Y50 Z50	50，50，50	60，60，60
G55 X100 Y100	100，100，30	130，130，60
X120 Z80	120，100，80	150，130，110

5. The offset value of external workpiece zero or the one of workpiece zero can be modified by G10, which is shown as follows:

Using code G10 L2 Pp X_Y_Z_

P=0 ： External workpiece zero offset value (reference offset amount).

P=1 to 6： Workpiece zero offset values of workpiece coordinate systems 1 to 6.

X_Y_Z_ ： For absolute code（G90）, it is workpiece zero offset of each axis.

For incremental code（G91）, it is the offset to be added to the set workpiece zero of each axis (the result of addition is the new workpiece zero offset).

Using G10, each workpiece coordinate can be changed respectively.

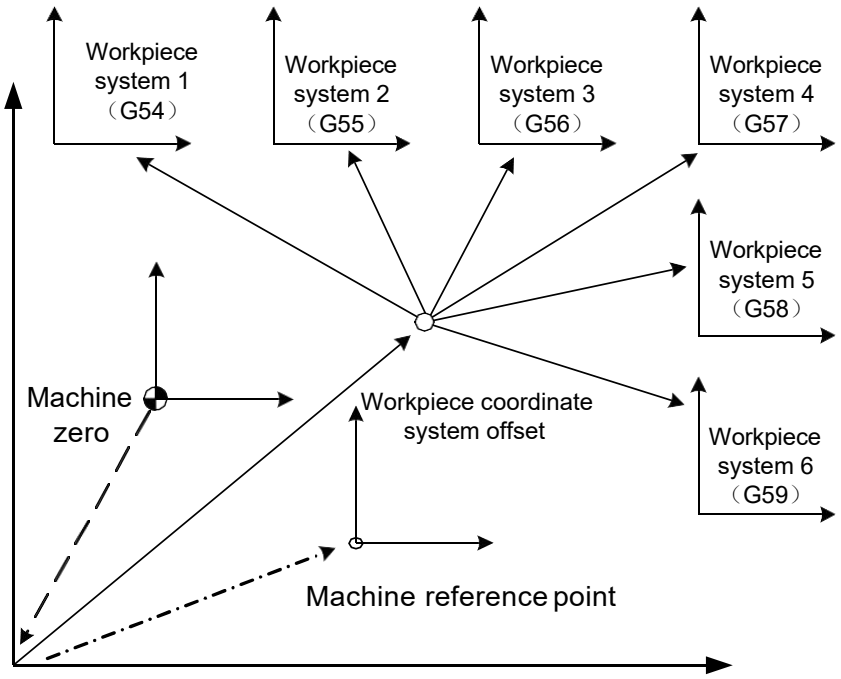


Fig. 3-2-8-1

As shown in Fig. **3-2-8-1**, after power-on, the machine returns to machine zero by manual zero return. The machine coordinate system is set up by the machine zero, which thus generates the machine reference point and determines the workpiece coordinate system. The corresponding values of offset data parameter **P260～P264** in workpiece coordinate system are the integral offset of the 6 workpiece coordinate systems. The origins of these workpiece coordinate systems can be specified by inputting the coordinate offset in MDI mode or by setting data parameters **P265～ P294**. These 6 workpiece coordinate systems are set up by the distances from machine zero to their respective coordinate system origins.

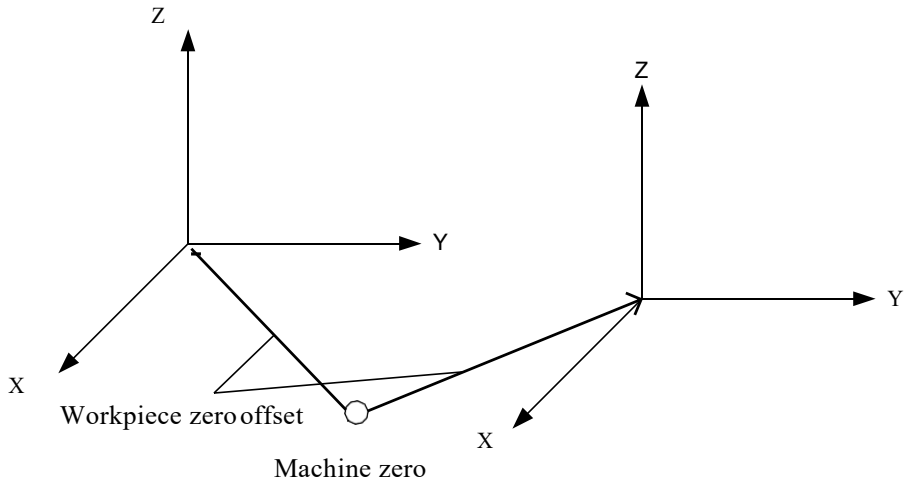


Fig. 3-2-8-2

Example: N10 G55 G90 G00 X100 Y20； N20

G56 X80.5 Z25.5；

In the above example, when block N10 is executed, the tool traverses rapidly to the position in workpiece coordinate system G55 (X=100， Y=20). When block N20 is executed , the tool traverses rapidly to the position in workpiece coordinate system G56, and the absolute coordinates shifts to the coordinates (X=80.5， Z=25.5) in workpiece coordinate system G55 automatically.

3.2.9 Additional workpiece coordinate system

Another 50 additional workpiece coordinate systems can be used besides the 6 workpiece coordinate systems (G54 to G59).

Format: G54 Pn

Pn：A code to specify the additional coordinate system with a range of 1~50.

The setting and restrictions of the additional workpiece coordinate system are the same as those of workpiece coordinate systems G54~G59.

G10 can be used to set the offset value of the workpiece zero in the additional workpiece system, as shown below:

Command：G10 L20 Pn X_Y_Z_；

n=1 to 50 ： Code of additional workpiece coordinate system

X_Y_Z_ ： For setting axis address and offset value for workpiece zero offset. For absolute code (G90), the specified value is the new offsetvalue.
For incremental code (G91), the specified value is added to the current offset value to produce a new offset value.

By G10 code, each workpiece coordinate system can be changed respectively.

3.2.10 Selecting machine coordinate system G53

Format：G53 X_ Y_ Z_

Function: To rapidly position the tool to the corresponding coordinates in the machine coordinate system.

Explanations:

1. While G53 is used in the program, the code coordinates behind it should be the ones in the

machine coordinate system and the machine will rapidly position to the specified location.

2. G53 is a non-modal code, which is only effective in the current block. It does not affect the coordinate system defined before.

Restrictions:

Select machine coordinate system G53

When the position on the machine is specified, the tool traverses to the position rapidly. G53 used for selecting the machine coordinate system is a non-modal code, i.e., it is effective only in the block specifying the machine coordinate system. Absolute value G90 should be specified for G53. If G53 is specified in incremental mode (G91), the code G91 will be ignored (i.e., G53 is still in G90 mode without changing G91 mode). The tool can be specified to move to a special position on the machine, e.g. using G53 to write a moving program to move the tool to the tool changing position.

Note: When G53 is specified, the tool radius compensation and tool length offset are cancelled temporarily. They will resume in the next compensation axis block buffered.

3.2.11 Floating coordinate system G92

Format: G92 X_ Y_ Z_

Function: for setting the floating workpiece coordinate system. The current tool absolute coordinate values in the new floating workpiece coordinate system are specified by 3 code parameters. This code does not cause the movement axis to move.

Explanation:

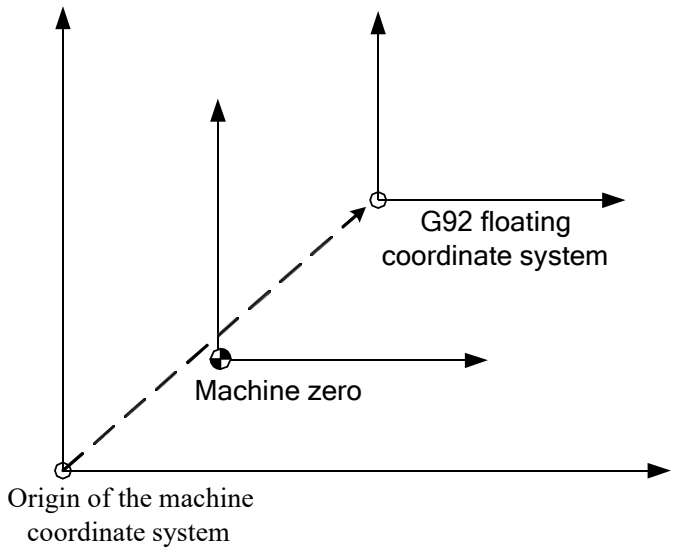


Fig. 3 -2-11-1

1. As shown in Fig. 3-2-11-1, the corresponding origin of the G92 floating coordinate system is the value in machine coordinate system, which is not related to the workpiece coordinate system. G92 setting is effective in the following conditions:
   1) Before the workpiece coordinate system is called
   2) Before the machine zero return

   The G92 floating coordinate system is often used for the alignment for temporary workpiece machining. It is usually specified at the beginning of the program or in MDI mode before the program auto run.

2. There are two methods to determine the floating coordinate system:

1）Determining the coordinate system with tool nose:

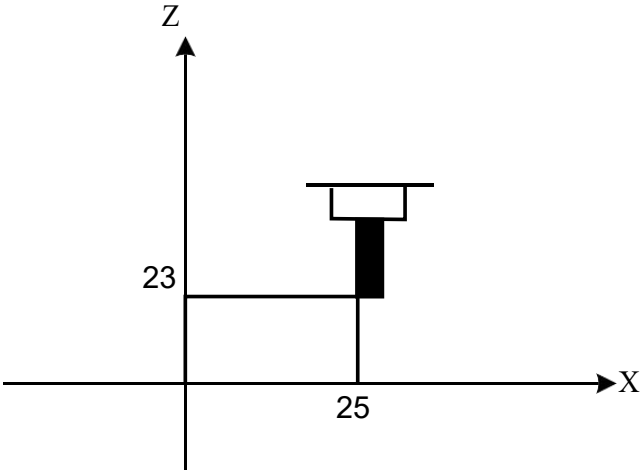


Fig. 3-2-11-2

As shown in Fig. 3-2-11-2, G92 X25 Z23, the tool nose position is taken as point（X25， Z23） in the floating coordinate system.

2）Taking a fixed point on the tool holder as the reference point of the coordinate system:

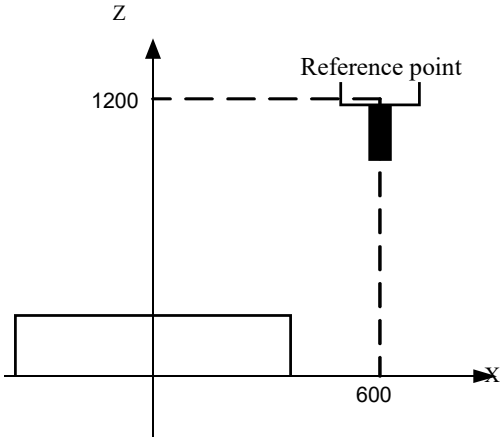


Fig. 3-2-11-3

As Fig. 3-2-11-3 shows, specify the workpiece coordinate system by code "G92 X600 Y1200" (taking a certain reference point on the tool holder as the tool start point). Taking a reference point on the tool holder as the start point, if the tool moves by the absolute value code in the program, the specified position to which the reference point is moved must adds the tool length compensation, the value of which is the difference between reference point and tool nose.

Note 1: If G92 is used to set the coordinate system in the tool offset, the coordinate system for tool length compensation is the one set by G92 before the tool offset is added.

Note 2: For tool radius compensation, the tool offset should be cancelled before G92 is used.

3.2.12 Plane selection G17/G18/G19

Format: G17/G18/G19

Function: Select planes for circular interpolation, tool radius compensation, drilling or boring with G17/G18/G19.

Explanation: It has no code parameter. G17 is the default plane at Power On. The default plane at Power On can also be determined by bit parameters **N0:31#1, and #2.** The relation between code and plane is as follows:

G17---------- XY plane
G18---------- ZX plane
G19---------- YZ plane

The plane keeps unchanged in the block in which G17，G18 or G19 is not specified.

Example：G18 X_ Z_； ZX plane

G0 X_ Y_； Plane remains unchanged (ZX plane)

In addition, the movement code is irrelevant to the plane selection. For example, in the following code, Y is not on the ZX plane, and its movement is irrelevant to the ZX plane. G18Y_；

Note: Only the canned cycle in G17 plane is supported at present. For criterion or astringency, it is strongly recommended that the plane be clearly specified in corresponding blocks when programming, especially in the case that a system is used by different operators. In this way, accidents or abnormality caused by program errors can be avoided.

3.2.13 Polar coordinate start/cancel G16/G15

Format: G16/G15

Function:

G16 specifies start of the positioning parameter's polar coordinate mode. G15 specifies cancel of the positioning parameter's polar coordinate mode.

Explanation:

No command parameters.

By setting G16, the coordinate value can be input with polar coordinate radius and angle. The positive direction of the angle is the counterclockwise direction of the 1st axis in the selected plane, and the negative direction is the clockwise direction. Both the radius and angle can use either absolute code or incremental code (G90 or G91).

After G16 appears, the 1st axis of the positioning parameter of the tool movement code is the polar radius in the polar coordinate system, and the 2nd axis is the polar angle in the polar coordinate system.

G15 can cancel the polar coordinate mode and thus return the coordinate value to the rectangular coordinate mode.

Specifying polar coordinate origin:

1. In G90 absolute mode, when G16 is specified, the zero point of the workpiece coordinate system is set as the origin of the polar coordinate system.

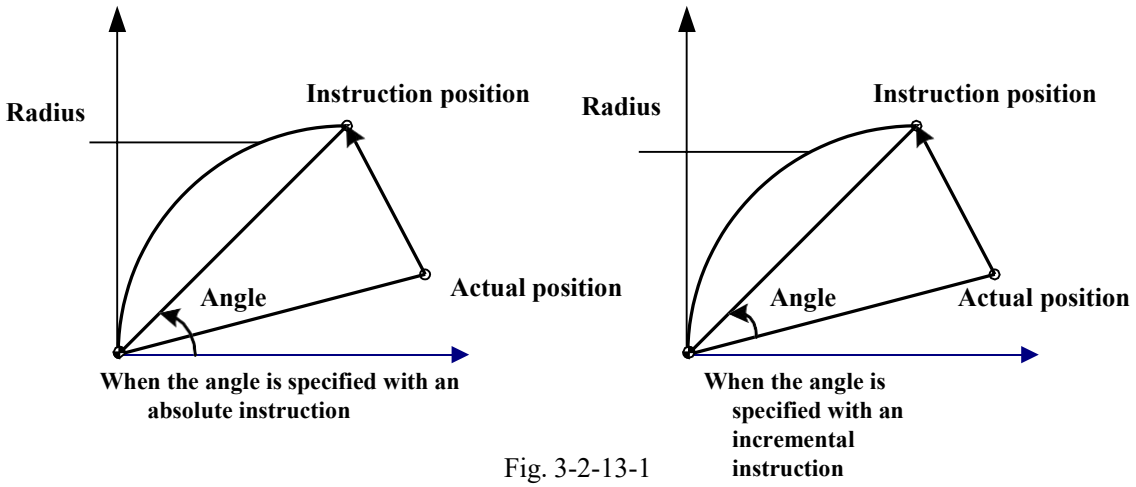


Fig. 3-2-13-1

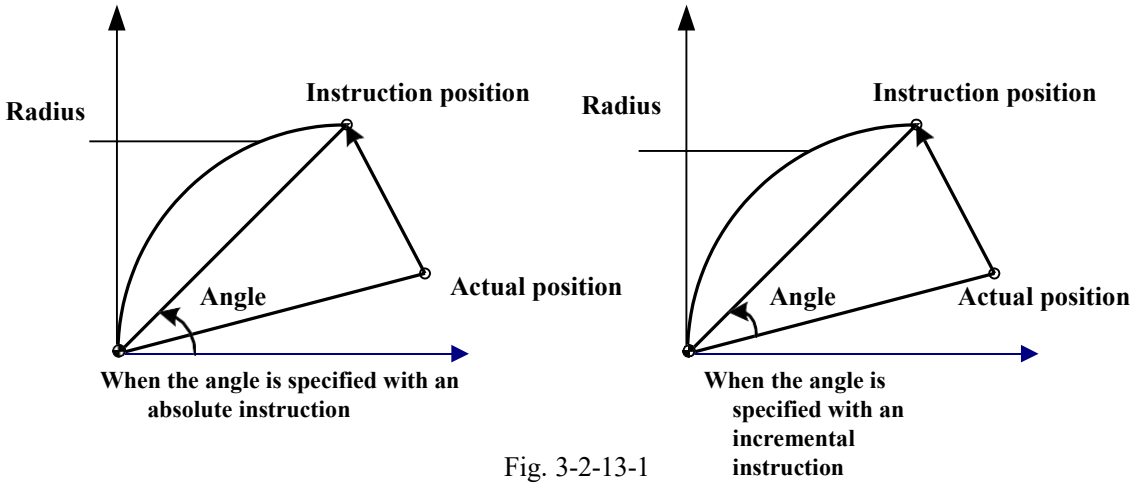


Fig. 3-2-13-1

2. In G91 absolute mode, when G16 is specified, the current point is set as the origin of the polar coordinate system.

Example: bole hole circle (the zero point of the workpiece coordinate system is set as the origin of the polar coordinate system, and X—Y plane is selected)

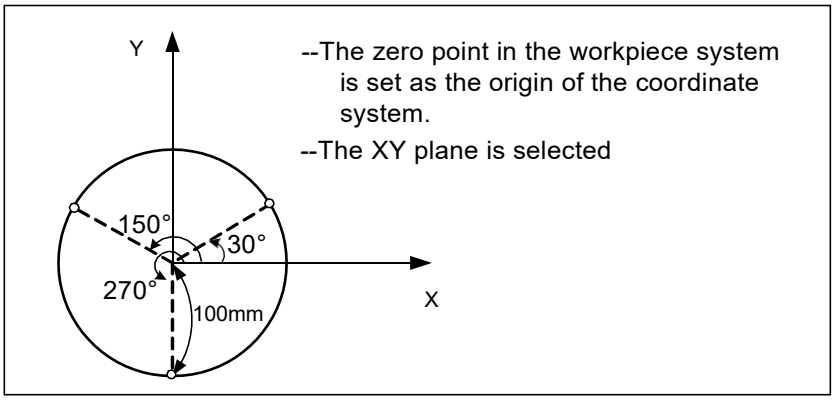


Fig. 3-2-13-2

Specifying angles and a radius with absolute value

G17 G90 G16；Specifying the polar coordinate code and selecting XY plane, setting the zero point of the workpiece coordinate system as the origin of the polar coordinate system.

G81 X100 Y30 Z-20 R -5 F200；Specifying a distance of 100mm and an angle of 30° Y150；
Specifying a distance of 100mm and an angle of150°

Y270； Specifying a distance of 100mm and an angle of270°

G15 G80； Cancelling the polar coordinate code

Specifying angles with incremental value and a polar radius with absolute value

G17 G90 G16；Specifying the polar coordinate code and selecting XY plane, setting the zero point of the workpiece coordinate system as the origin of the polar coordinate system.

G81 X100 Y30 Z-20 R -5 F200； Specifying a distance of 100mm and an angle of 30°. G91

Y120； Specifying a distance of 100mm and an angle of150°.

Y120； Specifying a distance of 100mm and an angle of270°.

G15 G80； Cancelling the polar coordinate code

Moreover, when programming by polar coordinate system, the current coordinate plane setting should be considered. The polar coordinate plane is related to the current coordinate plane. E.g. in

G91 mode, if the current coordinate plane is specified by G17, the components of X axis and Y axis of the current tool position are taken as the origin. If the current coordinate plane is specified by G18, the components of Z axis and X axis of the current tool position are taken as the origin.

If the positioning parameter of the first hole cycle code behind G16 is not specified, the system takes the current tool position as the default positioning parameter of the hole cycle. At present, the first canned cycle code behind the polar coordinate must be complete, or the tool movement is incorrect.

The positioning words of the positioning parameters of the tool movement codes behind G16, except for the hole cycle, are relevant to the actual plane selection mode. After the polar coordinate is cancelled with G15, if there is a movement code following it, the default current tool position is the start point of this movement code.

3.2.14 Scaling in a plane G51/G50

Format:

G51 X_ Y_ Z_ P_ （X.Y.Z: absolute code for the scaling center coordinates, P: each axis is scaled up or down at the same rate of magnification

… Scaled machining blocks

G50 Scaling cancelled

Or G51 X_ Y_Z_ I_ J_ K_ （Each axis is scaled up and down at different rates（I, J, K）of magnification

… Scaled machining blocks

G50 Scaling cancelled

Function:

G51 scales up and down the programmed figure in the same or different rate taking a specified position as its center. It is suggested that the G51 be specified in a separate block (or unexpected results may occur, resulting in workpiece damage and personal injury) and cancelled with G50.

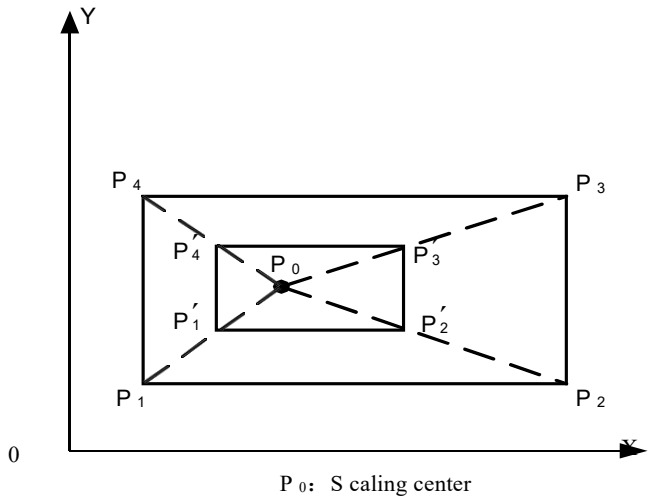


Fig. 3-2-14-1 Scaling up and down (P1P2P3P4→ P1’P2’P3’P4 ')

Explanation:

1. Scaling center: G51 can be specified with three positioning parameters X_Y_Z_, all of which are optional parameters. These positioning parameters are for specifying the scaling center of G51. If they are not specified, the system assumes the tool current position as the scaling center. Whether the current positioning mode is in absolute or incremental mode, the scaling center is always specified with the

absolute positioning mode. Moreover, the parameters of code G51 are also expressed with rectangular coordinate system in polar coordinate G16 mode.

Example: G17 G91 G54 G0 X10 Y10；

G51 X40 Y40 P2； Though in incremental mode, the scaling center is still the absolute coordinates（40,40）in G54 coordinate system

G1 Y90； Parameter Y is still in incremental mode.

2. Scaling: Either in G90 mode or G91 mode, the rate of magnification is always expressed with absolute mode.

   The rate of magnification can be set either in parameters or in programs. Data parameters P331~P333 correspond to the magnifications of X, Y and Z respectively. If there is no scaling code specified, the setting value of data parameter P330 is used for scaling.

   If the parameter values of parameter P or I, J and K are negative, the mirror image is applied for the corresponding axis.

3. Scaling setting: The effectiveness of scaling is set by parameter **No:60#5,** The effectiveness of the X axis scaling is set by bit parameter NO:47#3, the effectiveness of the Y axis scaling is set by bit parameter NO:47#4, the effectiveness of the Z axis scaling is set by bit parameter NO:47#5, and the scaling rate of each axis is set by bit parameter NO:47#6 (0: instructed with P, 1: instructed with I, J, K.).

4. Scaling cancel: After the scaling followed by a movement code is cancelled by G50, the current tool position is regarded as the start point of this movement code by default.

5. In scaling mode, G codes for reference point return （G27～G30 etc.）and coordinate system specification（G52～G59 , G92 etc.）can not be specified. They should be specified after the scaling is cancelled.

6. Even if different magnifications are specified for circular interpolation and each axis, the tool will not trace an ellipse.

   When the magnification for each axis is different and the circular interpolation is programmed with radius R, the interpolation figure is shown in fig. **3-2-14-2** (in the example below, the magnification for X axis is 2, for Y axis is 1)**.**

G90 G0 X0 Y100; G51
X0 Y0 Z0 I2 J1;
G02 X100 Y0 R100 F500;
Above instructions are equivalent to the following ones:
G90 G0 X0 Y100;
G02 X200 Y0 R200 F500;
The magnification of radius R depends on I or J,
whichever is larger.

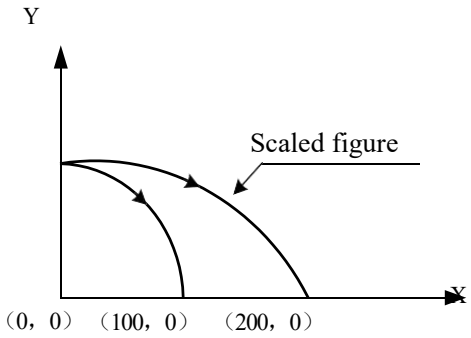
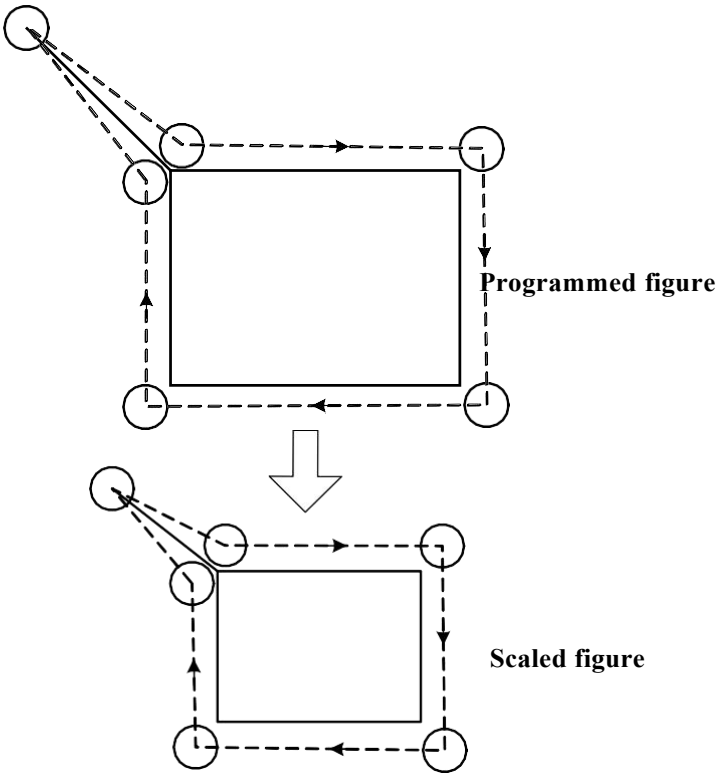


Fig. 3-2-14-2 Scaling for circular interpolation 1

When the magnifications of the axes are different and the circular interpolation is programmed with I, J and K, an alarm is given if the arc does not exist.

7. Scaling has no effect on the tool offset value, see Fig. **3-2-14-3**.



Tool radius compensation value is not scaled

Fig. 3-2-14-3 Scaling for tool radius compensation

Example of a mirror image program:

Main program:

```
G00 G90;  M98
P9000;
G51 X50.0 Y50.0 I-1 J1;
M98 P9000;
G51 X50.0 Y50.0 I-1 J-1;
M98 P9000;
G51 X50.0 Y50.0 I1 J-1;
M98 P9000;
G50;
M30;
```

Subprogram:

```
O9000;
G00 G90 X60.0 Y60.0;
G01 X100.0 F100;  G01
Y100;
G01 X60.0 Y60.0;
M99;
```

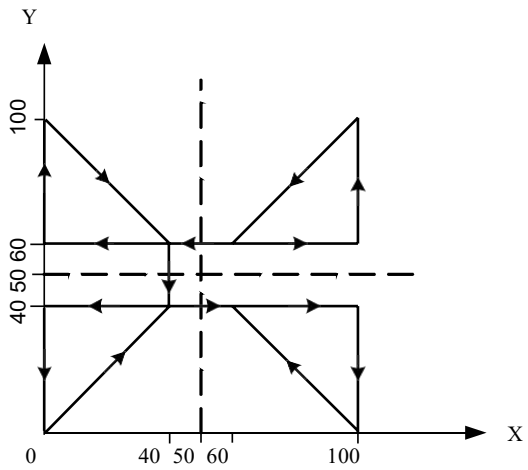


Fig. 3-2-14-4

Restrictions:

1. When the canned cycle is executed in scaling mode, the system only scales up or down the hole positioning data rather than point R, value Q, point Z at hole bottom and dwell time P at hole bottom.
2. In MANUAL mode, the traverse distance cannot be increased or decreased by scaling.

Note 1: The position displays the coordinate values after scaling.

Note 2: The results are as follows when a mirror image is applied to one axis of a specified plane: 1）Circular code….....Direction of rotation is reversed

2）Tool radius compensation C….....Direction of offset is reversed

3）Coordinate system rotation….....Rotation angle is reversed

4）

3.2.15 Coordinate system rotation G68/G69

For the workpiece which consists of many figures with the same shapes, users can program

using the coordinate rotation function, i.e., write a subprogram to the figure unit, and then call the subprogram using rotation function.

Command format: **G17 G68 X_ Y_ R_;**

Or G18 G68 X_ Z_ R_; Or

G19 G68 Y_ Z_R_ ;

G69;

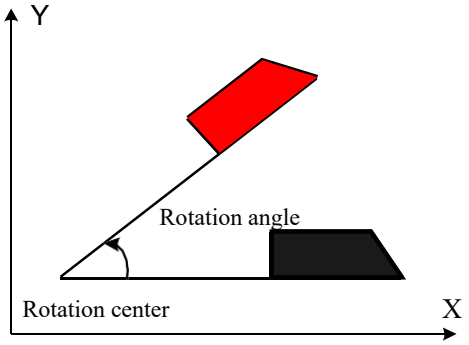


Fig. 3-2-15-1

Function: G68 rotates the programmed shape in a plane taking a specified center as its origin. G69 is used for cancelling the coordinate system rotation.

Explanation:

1. G68 has two positioning parameters, both of which are optional ones. They are used for specifying the rotation center. If the rotation center is not specified, the system assumes the current tool position as the rotation center. The positioning parameters are relative to the current coordinate plane, e.g., X and Y for G17; X and Z for G18; Y and Z forG19.
2. When the current positioning mode is the absolute mode, the system assumes the specified point as the rotation center. When the positioning mode is the relative mode, the system specifies the current point as the rotation center. G68 can also use an code parameter R, of which the value is the rotation angle, with degree as its unit. A positive value of R indicates the counterclock rotation. When there is no rotation angle code in the coordinate rotation, the rotation angle to be used is set by data parameter **P329.**
3. In G91 mode, the system takes the current tool position as the rotation center; the rotation angle by increment is set by bit parameter NO: 47#0 (rotation angle of coordinate system, 0: by absolute code; 1: by G90/91 code).
4. When programming, please note that no plane selection is allowed when the system is in rotation mode, otherwise an alarm occurs.
5. In coordinate system rotation mode, G codes for reference point return （G27～G30 etc.） and coordinate system specification（G52～G59 , G92 etc.）cannot be specified. They should be specified after the scaling is cancelled if needed.
6. After coordinate system rotation, perform operations such as the tool radius compensation, tool length compensation, tool offset and other compensation.
7. If the coordinate system rotation is performed in scaling mode (G51), the rotation center coordinate values will be scaled rather than the rotation angle. When a movement code is given, the scaling will be executed first, then the coordinate system rotation.

Example 1：Rotation：

G92 X-50 Y-50 G69 G17 ;

G68 X-50Y-50 R60;

G90 G01 X0 Y0 F200；
G91 X100；
G02 Y100 R100； G3 X-100 I-50 J-50； G01 Y-100；
G69；
M30；

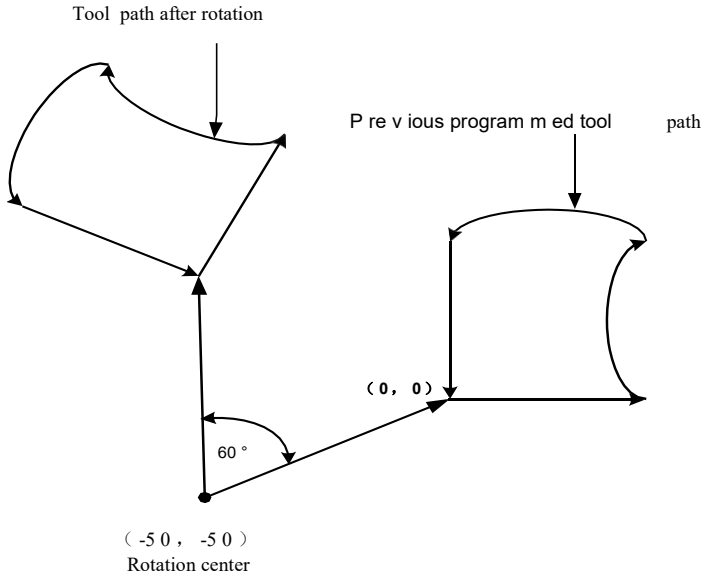


Fig. 3-2-15-2

Example 2: Scaling and rotation G51

X300 Y150 P0.5； G68 X200
Y100 R45； G01 G90 X400
Y100； G91 Y100；
X-200 ；
Y-100 ；
X200；
G69 G50 ；
M30；

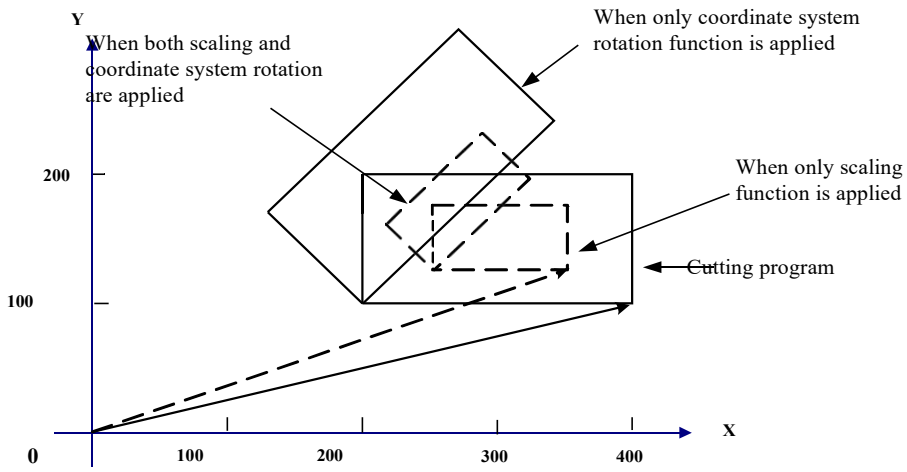


Fig. 3-2-15-3

Example 3: Repetitive use of G68

By program (main program) G92
X0 Y0 Z20 G69 G17；

M3 S1000； G0

Z2；

G42 D01； (tool offset setting)

M98 P2100(P02100)； (subprogram call) M98

P2200L7； （call 7 times） G40；

G0 G90 Z20；

X0Y0； M30；

subprogram 2200

O2200

G91

G68 X0 Y0 R45.0； (relative rotation angle)

G90；

M98 P2100； (subprogram O2200 calls subprogram O2100)

M99；

subprogram 2100

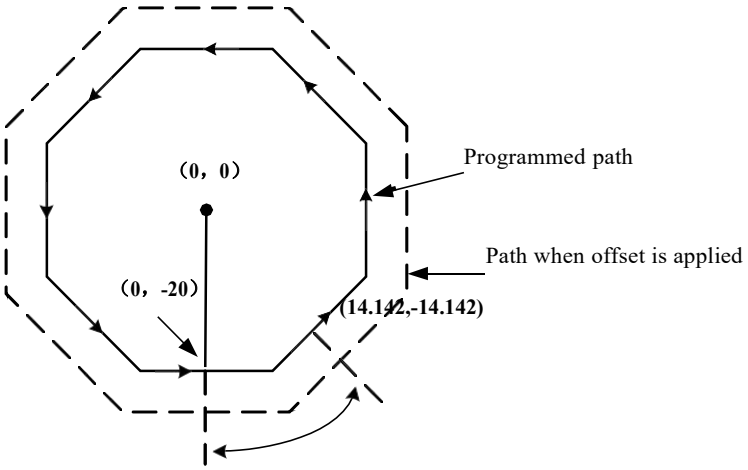
O2100 G90 G0 X0 Y-20； (right-hand tool compensation setup) G01Z－2

F200；

X8.284；

X14.142 Y-14.142；

M99；



(8.284,-20)

Subprogram

Fig. 3-2-15-4

3.2.16 Skip function G31

Command format: **G31 X_Y_Z_**

Function: Linear interpolation can be specified after G31 in the same way as after G01. During the execution of this code, if an external skip signal is input, the execution of the code is

interrupted and the next block is executed. When the machining end point is not programmed, but it is specified using a signal from the machine, use the skip function. For example, use it for grinding. The function is used for measuring the dimension of a workpiece as well.

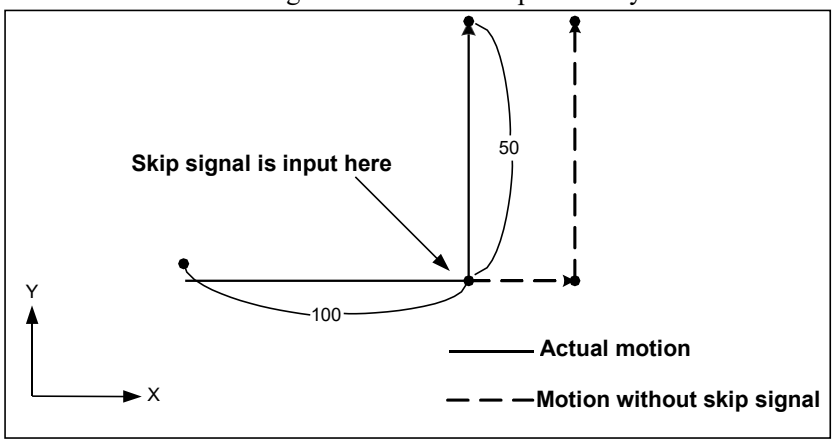
Explanation:

1. G31 is a non-modal G code only effective in the block in which it is specified.
2. When tool radius compensation is being executed, if G31 is specified, an alarm will occur. Therefore, the tool radius compensation should be cancelled before G31.

Example:

The block after G31 is a single axis movement specified by incremental values, as Fig. 3-2-16-1 shows :

Fig. 3-2-16-1 The next block is the single-axis movement specified by incremental values



The next block after G31 is a single-axis movement specified by absolute values, as shown in fig. **3-2-16-2:**

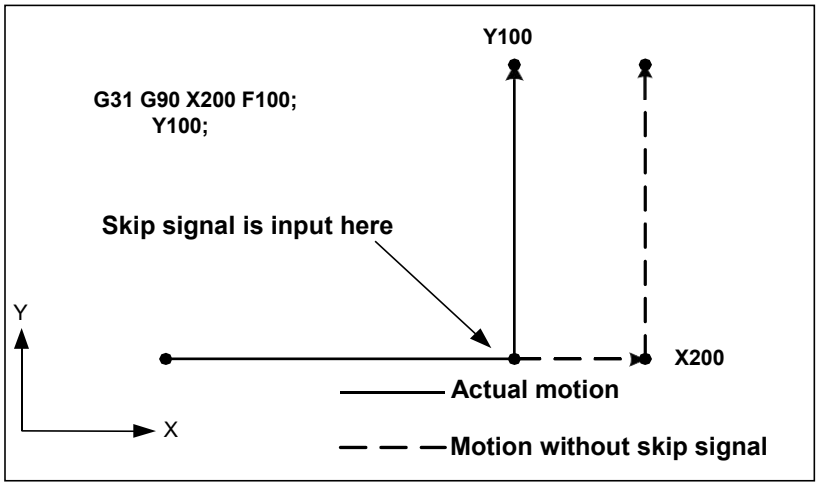


Fig. 3-2-16-2 The next block is a single-axis movement specified by absolute values

The next block after G31 is two-axis movement specified by absolute values, as shown in fig. **3-2-16-3**:

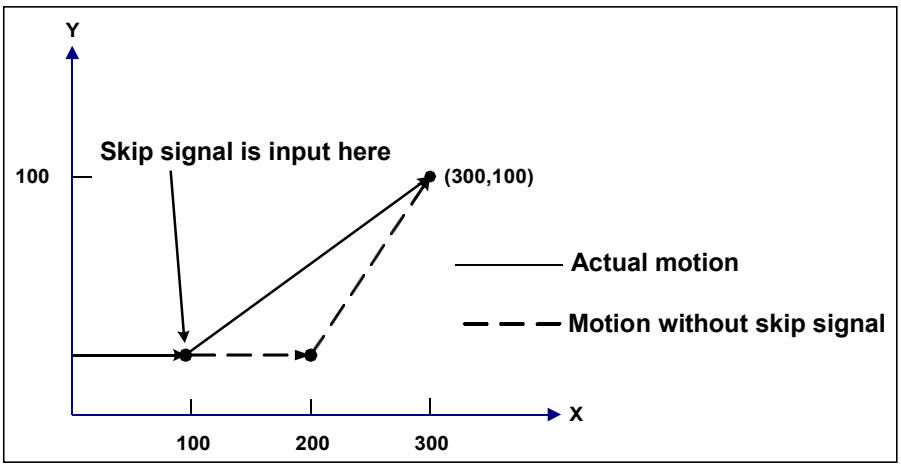


Fig. 3-2-16-3 The next block is two-axis movement specified by absolute value

Note: The setting can be done by bit parameter NO:02#6 [skip signal SKIP, (0:1, 1:0)].

3.2.17 Inch/metric conversion G20/G21

Format: **G20**: inch input

G21: metric input

Function: They are used for the inch/metric input conversion in a program.

Explanation:

After inch/metric conversion, the units of the following values are changed: Inch/Metric. Feedrate specified by F code, position code, workpiece zero offset value, tool compensation value, scale unit of MPG and movement distance in incremental feeding.

The G code status at power-on is the same as that held before power off.

Note:

1. When the inch input is converted to metric input or vice versa, the tool compensation value must be preset according to the least input incremental unit.
2. After inch input is converted to metric input or vice versa, for the first G28, the operation from the intermediate point is the same as that of manual reference point return.
3. When the least input incremental unit is different from the least code incremental unit, the maximum error is half of the least code unit and this error is not accumulated.
4. Program inch/metric input can be set by bit parameter N0:00#2.
5. Program inch/metric output can be set by bit parameter N0:03#0.
6. G20 or G21 must be specified in a separate block.

3.2.18 Optional angle chamfering/corner rounding

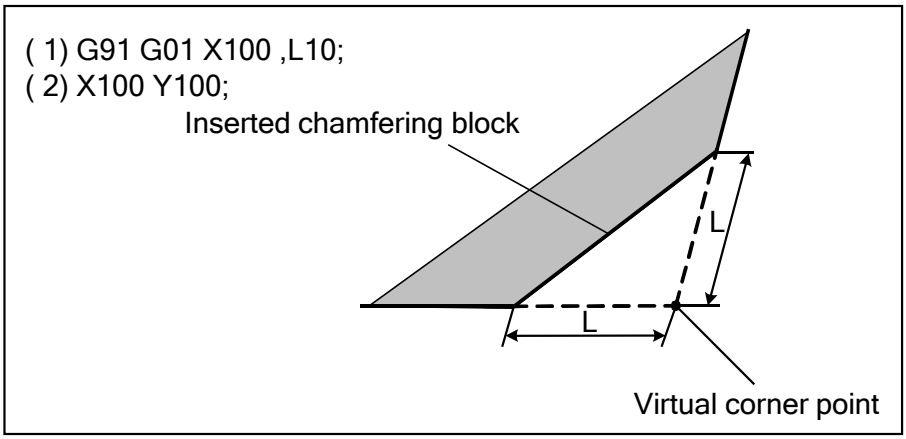
Format: **L_**: Chamfering

R_: Corner rounding

Function: When the codes above are added to the end of the block specifying linear interpolation (G01) or circular interpolation(G02,G03), a chamfering or corner rounding is added automatically outside the corner during machining. Blocks specifying chamfering or corner rounding arc can be specified consecutively.

Explanation:

1. Chamfering: after L, specify the distance from the virtual corner point to the start and the end points of the corner. The virtual corner point is the corner point that exists if chamfer is not performed.



As the following figure shows:

Fig. 3-2-18-1

2. Corner R: after R, specify the radius for the corner rounding, as shown below:

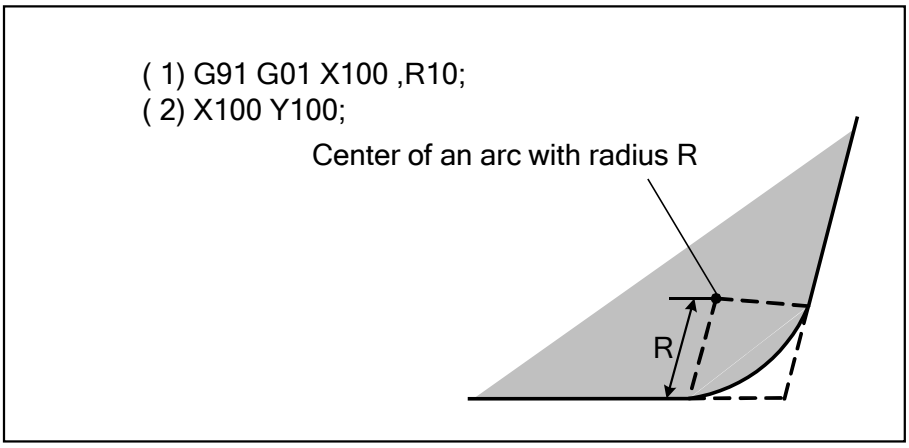


Fig. 3-2-18-2

Restrictions:

1. Chamfering and corner rounding can only be performed in a specified plane, and these functions cannot be performed for parallel axes.
2. If the inserted chamfering or corner rounding block causes the tool to go beyond the original interpolation move range, an alarm is issued.
3. Corner rounding cannot be specified in a threading block.
4. When the values of chamfering and corner rounding are negative, their absolute values are used in the system.

3.3 Reference point G code

The reference point is a fixed point on the machine tool to which the tool can easily be moved by the reference point return function.

There are 3 codes for the reference point, as is shown in Fig. **3-3-1**. The tool can be automatically moved to the reference point via an intermediate point along a specified axis by G28; or be moved automatically from the reference point to a specified point via an intermediate point along a specified axis by G29.

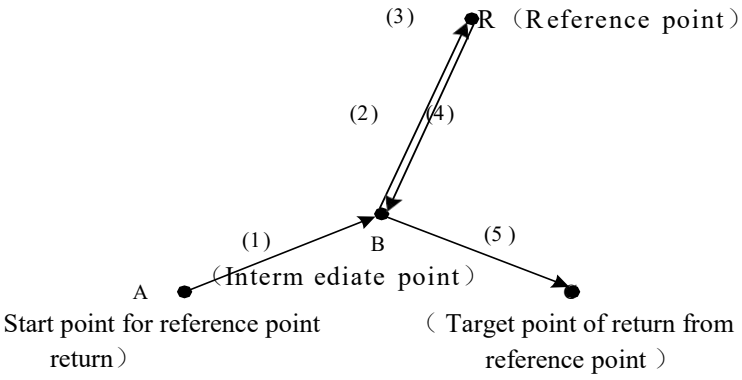


Fig. 3-3-1

3.3.1 Reference point return G28

Format: G28 X_ Y_ Z_

Function： G28 is for the operation of returning to the reference point (a specific point on the machine tool) via intermediate point.

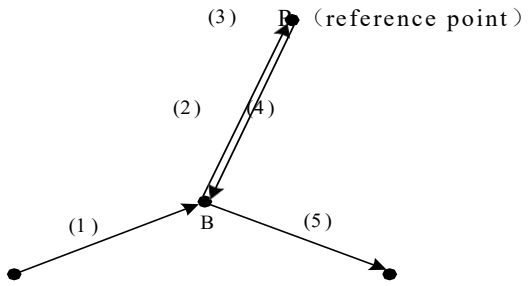
Explanation:

Intermediate point:

An intermediate point is specified by an code parameter in G28. It can be expressed by absolute or incremental codes. During the execution of this block, the coordinate values of the intermediate point of the axis specified are stored for the use of G29 code (returning from the reference point).

Note:

The coordinate values of the intermediate point are stored in the CNC system. Only the axis coordinate values specified by G28 are stored each time, for the other axes not specified by G28, the coordinate values specified by G28 before are used. If the current default intermediate point of the system is unknown when G28 is used, it is recommended that each axis be specified with one. Please take a consideration according to block N5 in the following example.



A C

Fig. 3-3-1-1

1. The action of block G28 can be divided as follows: (refer to Fig.3-3-1-1):
   (1) Positioning to the intermediate point of the specified axis from the current position (point A→point B) at a traverse speed.
   (2) Positioning to the reference point from the intermediate point (point B →point R) at a traverse speed.

2. G28 is a non-modal code which is effective only in the current block.
3. Single-axis reference point return and multi-axis reference point return are available. The intermediate point coordinates are saved by the system when the workpiece coordinate system is changed.

Example:

N1 G90 G54 X0 Y10;

N2 G28 X40 ; Set the intermediate point of X axis to X40 in G54 workpiece coordinate system, and return to reference point via point（40,10）, i.e. X axis returns to the reference point alone.

N3 G29 X30 ; Return to point (30，10) via point（40,10）from reference point, i.e. X axis returns to the target point alone.

N4 G01 X20;

N5 G28 Y60 ; intermediate point is (X40，Y60).

N6 G55; After the workpiece coordinate system is changed, the intermediate point is changed into the point (40,60) in the workpiece coordinate system set by G55 from the point (40, 60) in the workpiece coordinate system set by G54.

N7 G29 X60 Y20; Return to point (60, 20) via the intermediate point (40，60) in G55 workpiece coordinate system from the reference point.

The G28 code can automatically cancel the tool compensation, but this code is only used in automatic tool change mode (i.e. changing the tool at the reference point after reference point return). Therefore, the tool radius compensation and tool length compensation, in principle, should be cancelled before the use of this code. See data parameters P45～P48 for the 1st reference point setting.

3.3.2 2nd, 3rd, 4th reference point return G30

There are 4 reference points in machine coordinate system. In a system without an absolute-position detector, the 2nd, 3rd, 4th reference point return functions can be used only after the auto reference point return（G28） or manual reference point return is performed.

Format:

G30 P2 X_ Y_ Z_; 2nd reference point return (P2 can be omitted)

G30 P3 X_ Y_ Z_; 3rd reference point return

G30 P4 X_ Y_ Z_; 4th reference point return

Function: G30 performs the operation of returning to the specified reference point via the intermediate point specified by G30.

Explanation:

1. X_ Y_ Z_; Code for specifying the intermediate point (absolute/ incremental)
2. The setting and restrictions of code G30 are the same as those of code G28. See data parameter P50～63 for the 2nd, 3rd, 4th reference point setting.
3. The G30 code can also be used together with G29 code (return from the reference point), of which the setting and restrictions are identical with those of G28 code.
4.

3.3.3 Automatic return from reference point G29

Format：G29 X_ Y_ Z_

Function: G29 performs the operation of returning to the specified point via the intermediate point specified by G28 or G29 from the reference point (or the current point).

Explanation:

1. The action of block G29 can be divided as follows: (refer to Fig.3-3-1-1):
   (1) Positioning to the intermediate point (point R→point B) specified by G28 or G30 from the reference point at a traverse speed.
   (2) Positioning to a specified point from the intermediate point (point B →point C) at a traverse speed.
2. G29 is a non-modal code which is only effective in the current block. In general, the code return from Reference Point should be specified immediately after code G28 or G29.
3. The optional parameters X, Y and Z in G29 code are used for specifying the target point (i.e. point C in Fig. 3-3-1-1) of the return from the reference point, all of which can be expressed by absolute or incremental code. The code specifies the incremental value departed from the intermediate point in incremental programming. If the value is not specified for an axis, it means the axis has no movement relative to the intermediate point. The G29 code followed by only one axis means the single axis return with no action performed to other axes.

Example:

G90 G0 X10 Y10；

G91 G28 X20 Y20； Reference point return via the intermediate point (30，30).

G29 X30； Return to (60，30) from the reference point via the intermediate point (30 ， 30). Note that the component in X axis should be 60 in incremental programming.

The values of the intermediate point specified by G29 are assigned by G28 or G30. See the explanation of code G28 for the definition, specification and system default of the intermediate point.

3.3.4 Reference Point Return Check G27

Format：G27 X_ Y_ Z_

Function：G27 performs the reference point return check, with the reference point specified by X_ Y_ Z_ .

Explanation:

1. G27 code, the tool at the rapid traverse speed. If the tool reaches the reference point, the indicator for reference point return lights up. However, if the position the tool reaches is not the reference point, an alarm is issued.
2. In machine lock mode, even if G27 is specified and the tool has automatically returned to the reference point, the indicator for return completion does not light up.
3. In the offset mode, the position to be reached by the tool specified with G27 code is the position obtained after the offset is added. Therefore, if the position with the offset added to it is not the reference point, the indicator does not light up, and an alarm is issued. Usually the tool offset should be cancelled before the use of G27 code.
4. The coordinate position of X, Y and Z specified by G27 is the position in the machine coordinate system.

3.4 Canned cycle G code

The canned cycle uses a single block containing G functions to achieve the machining action which needs to be done with multiple blocks to simply the programming, making the programming easier for programmers (in this system only the canned cycle in G17 plane is available).

General process of canned cycle:

A canned cycle consists of a sequence of 6 operations, as shown in Fig. **3-4-1.**

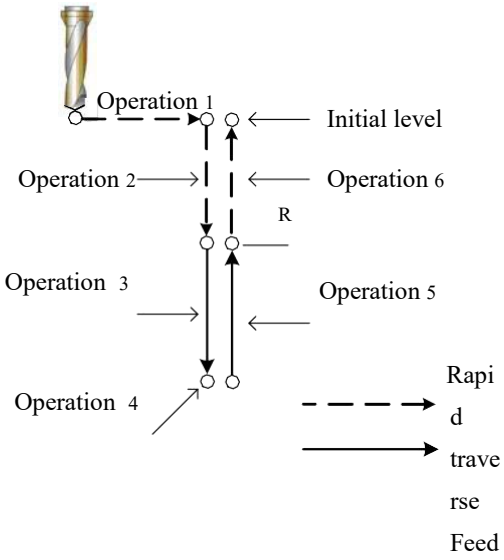


Fig. 3-4-1

Operation 1: Positioning of axes X and Y (another axis can be included)

Operation 2: Rapid traverse to point R level

Operation 3: Hole machining

Operation 4: Operation at the bottom of a hole

Operation 5: Retraction to point R level Operation 6:

Rapid traverse to the initial point

Positioning is performed in XY plane, and hole machining is performed along Z axis. It is defined that a canned cycle operation is determined by 3 types, which are specified by G codes respectively.

1) Data type

G90 absolute mode; G91 incremental mode

2) Return point plane

G98 initial level; G99 point R level

3) Hole machining type

G73, G74, G76, G81～G89

Initial point Z level and point R level

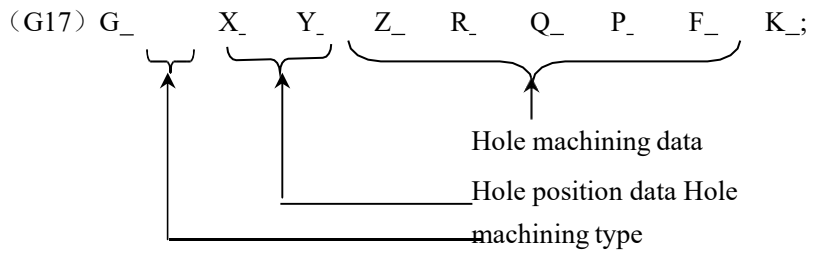
Initial point Z level: It is the absolute position where the tool is located in Z axis before the canned cycle.

Point R level: It is also called safety level. It is the position in Z axis which is generally located a certain distance above the workpiece surface to prevent the tool from colliding with the workpiece and ensure an enough distance for deceleration when the rapid traverse is switched to cutting feed in canned cycle.

G73/G74 /G76/G81 ～ G89 specifies all the data of canned cycle (hole position data, hole machining data and number of repeats) into a single block.

Z, R: If either of hole bottom parameter Z and R is missing when the first hole drilling is executed, the system only changes the mode, with no Z axis action executed.

The format of hole machining is as follows:



The meanings of hole position data and hole machining data are shown in Table **3-4-1:**

Table 3-4-1

Designation	Parameter word	Explanation
Hole machining	G	Refer to table **3-4-3**, and note the restrictions above.
Hole position data	X，Y	The hole position is specified by either absolute value or incremental value and the control is identical to that of G00 positioning.
Hole machining data	Z	As Fig. 3-4-2 shows, the distance from point R level to the hole bottom is specified by incremental values, or the hole bottom coordinates are specified by absolute values. As shown in fig. 3-4-1, the feedrate is the speed specified by F in operation 3, while it is the rapid traverse speed or the speed specified F code in operation 5 depending on different hole machining types.
	R	In Fig. 3-4-2, the distance from the initial level to point R level is specified by incremental value, or point R level coordinates are specified by absolute values. The feedrates, shown in fig. 3-4-1, are both rapid traverse in operations 2 and 6.
	Q	It is used to specify the cut-in value each time in G73 or G83, or the parallel movement value (incremental value) in G76 or G87.
	P	It is used to specify the dwell time at the hole bottom. The canned cycle code can be followed by a parameter P_ , which specifies the dwell time after the tool reaches the Z plane with unit of ms. The min. value of the parameter can be set by number parameter P281, and the max. value by data parameter P282.
	F	It is used for specifying the cutting feedrate.
	K	The number of repeats is specified in K_, which is only effective in the block in which it is specified. If it is omitted, the default is 1 time. The maximum drilling times are 99999. When the value is negative, its absolute value is executed. When the value is 0, only the mode is changed, with no drilling operation executed.

Restrictions:

The canned cycle G codes are modal ones, which remain effective till they are cancelled by a G code for cancelling it.

G80 and G codes in group 01 are used for cancelling the canned cycle.

Once the hole machining data in canned cycle is specified, it is retained till the cycle is

cancelled. All the required hole machining data should be specified at the beginning of the canned cycle, and only the updated data needs to be specified in the subsequent canned cycle.

Note 1: The feedrate specified by F remains effective even if the canned cycle is cancelled. Note 2: The scaling for Z axis (cutting axis direction) is invalid in the canned cycle.
Note 3: In single block mode, the canned cycle uses the 3-stage machining type, i.e. positioning→R level→initial level.
Note 4: In the canned cycle, when the system bit parameter NO:36#1 is 1, if reset or emergency stop is performed, both the hole machining data and hole position data will be cleared. Examples for data remaining and data clearing above are shown in the following table:

Table 3-4-2

Sequence	Data designation	Explanation
①	G00X_M3;	
②	G81X_Y_Z_R_F_;	Specify values for Z, R and F in the beginning.
③	Y_;	G81，Z-R-F- can all be omitted since the hole machining mode and data are the same as those specified in ②. Drill the hole for the length Y once by G81.
④	G82X_P_;	Move only in X axis direction relative to the position of hole ③. Perform hole machining by G82 using the hole machining data Z, R and F specified in ② and P in ④.
⑤	G80X_ Y_	Hole machining is not performed. Cancel all the hole machining data.
⑥	G85X_Z_R_P_;	Since all the data are cancelled in ⑤, Z and R need to be re-specified. F is identical with that in ② , so it can be omitted. P is not required in this block and it is saved.
⑦	X_Z_;	It is the hole machining identical with that in ⑥ except for Z value. And there is movement only in X axis at the hole position.
⑧	G89X_Y_;	Perform G89 hole machining using Z specified in ⑦, R and P in ⑥, F in ② as the machining data.
⑨	G01X_Y_;	Cancel hole machining mode and clear hole machining data.

A. Absolute code and incremental code in canned cycle G90/G91

The change of G90/G91 along drilling axis is shown as Fig. 3-4-2. (Usually it is programmed by G90. if it is programmed by G91, Z and R are processed according to the specified signs + and -).

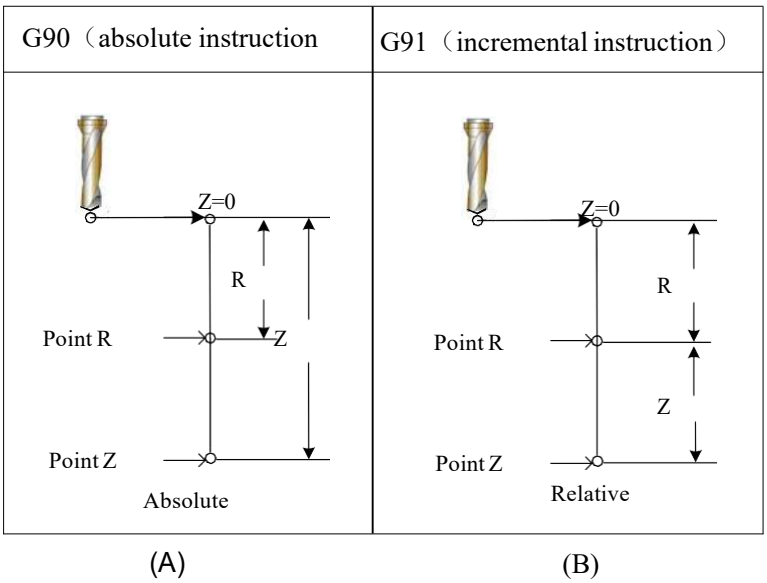


Fig. 3-4-2

B. Return to initial level in canned cycle G98/G99

After the tool reaches the bottom of a hole, it may return to the point R level or the initial level. These operations can be specified by G98 and G99.

Generally, G99 is used for the 1st drilling operation and G98 for the last drilling operation. The initial level does not change even if the drilling is performed in G99 mode. The following figure illustrates the operations of G98 and G99.

G98 is the system default mode.

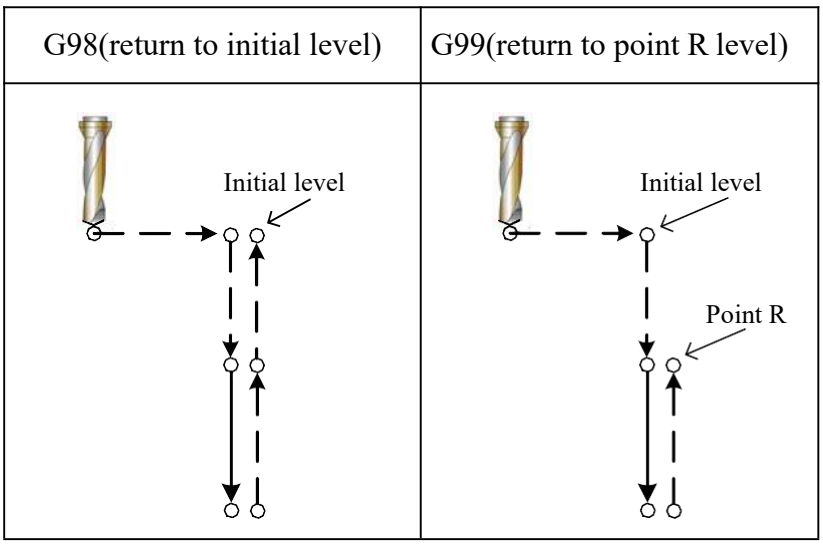


Table 3-4-3

The following symbols are used for the canned cycle illustration:

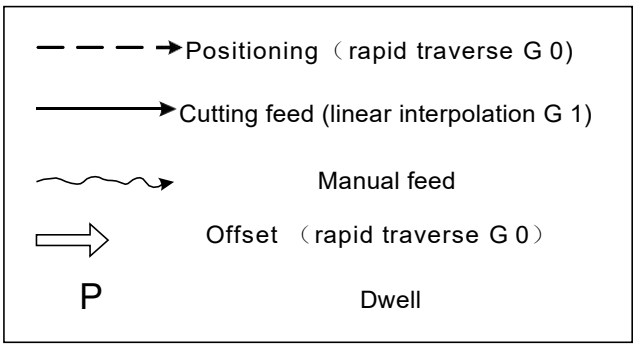


Fig. 3-4-4

Canned cycle comparison table（G22～G89）

Table 3-4-3

G	Driling (-Z)	hole bottom	Retraction (+Z)	Application
G73	Intermittent feed		Rapid traverse	High-speed deep hole drilling cycle
G74	Cutting feed	Dwell→spindle CCW	Rapid traverse	Counter tapping cycle
G76	Cutting feed	Spindle orientation stop	Rapid traverse	Fine boring
G80				取消固定循环
G81	Cutting feed		Rapid traverse	Drilling, spot drilling
G82	Cutting feed	Stop	Rapid traverse	Drilling, counter boring
G83	Intermittent feed		Rapid traverse	Deep holde drilling cycle
G84	Cutting feed	Stop < spindle CCW	Cutting feed	Taping
G85	Cutting feed		Cutting feed	Boring
G86	Cutting feed	Spindle stop	Rapid traverse	Boring
G87	Cutting feed	Spindle CW	Rapid traverse	Boring
G88	Cutting feed	Stop < spindle stop	Manual < spindle CW	Boring
G89	Cutting feed	Dwell	Cutting feed	Boring

Restrictions:

Tool radius offset (D) is ignored during the canned cycle positioning.

4.4.1 High-speed peck drilling cycle G73

Format: G73 X_Y_Z_R_Q_F_K_

Function: The cycle is specially set for the high-speed peck drilling. It performs intermittent cutting feed to the bottom of a hole while removing chips from the hole. The operation illustration is shown as Fig. **3-4-1-1.**

Explanation:

X_Y_:Hole positioning data;

Z_: In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom;

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R;

Q_: Cut depth of each cutting feed; F_:

Cutting feedrate;

K_ : Number of repeats.

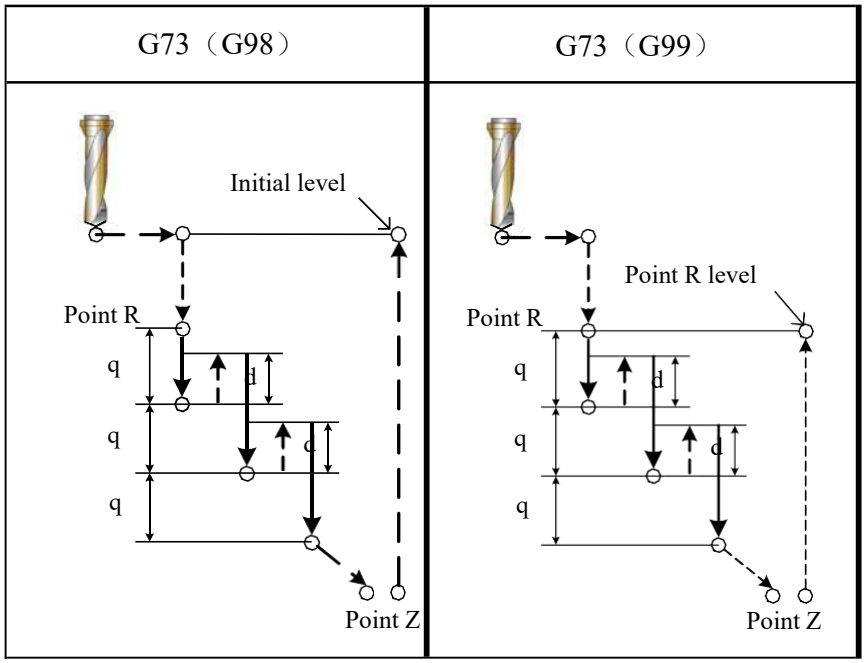


Fig. 3-4-1-1

Z, R： If either of hole bottom parameter Z and R is missing when the first drilling is being executed, the system only changes the mode, with no Z axis action executed.

Q： If parameter Q is specified, the intermittent feed shown in the figure above is performed. Here, the system retracts the tool by the retraction d (Fig.3-4-1-1) specified by data parameter **p270**, and the tool performs rapid retraction for distance d intermittently each feeding.

If G73 and an M code are specified in the same block, the M code is executed at the time of the 1st hole positioning operation, then the system proceeds to the next drilling operation.

If the number of repeats K is specified, M code is only executed for the first hole, not for the other holes.

Note 1: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

Note 2: When the bit parameter NO:43# 1=0, no alarm will be issued if there is no cut-in value specified in the peck drilling (G73， G83). At this moment, if the code parameter Q is not specified or it is 0, the system performs the hole positioning in XY plane, but does not perform the drilling operation. When the bit parameter NO:43#1=1, an alarm will be issued if no cut-in value is specified in the peck drilling (G73， G83), i.e., an alarm "0045:Address Q not found or set to 0（G73/G83）" occurs when the code parameter Q is not specified or it is 0. If the Q value is negative, the system takes its absolute value to perform intermittent feed.

Note 3: Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with a canned cycle code, the offset is added or cancelled when the tool is positioned to point R; If the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Limitation: G codes in 01 group (G00 to G03), G60 modal G code (bit parameter NO: 48#0 is set to 1) and G73 cannot be specified in the same block, otherwise G73 will be cancelled.

Tool offset: The tool radius offset is ignored during the canned cycle positioning.

Example:

M3 S1500; The spindle starts to rotate

G90 G99 G73 X0 Y0 Z-15 R-10 Q5 F120; Positioning, drill hole 1, then return to point R level. Y-50;

Positioning, drill hole 2, then return to point R level

Y-80;	Positioning, drill hole 3, then return to point R level
X10;	Positioning, drill hole 4, then return to point R level
Y10;	Positioning, drill hole 5, then return to point R level
G98 Y75;	Positioning, drill hole 6, then return to initial level G80;
G28 G91 X0 Y0 Z0;	Return to reference point
M5;	Spindle stops
M30;	

Note: In the example above, the chip removal operation is still performed though Q is omitted during the machining for the holes 2 to 6.

4.4.2 Drilling cycle, spot drilling cycle G81

Format: G81 X_ Y_ Z_ R_ F_ K_

Function: This cycle is used for normal drilling. Cutting feed is performed to the bottom of the hole, and then the tool is retracted from the bottom in rapid traverse.

Explanation:

X_Y_: Hole positioning data

Z_:In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming, it specifies the absolute coordinates of the hole bottom.

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R level.

F_: Cutting feedrate

K_: Number of repeats (if necessary)

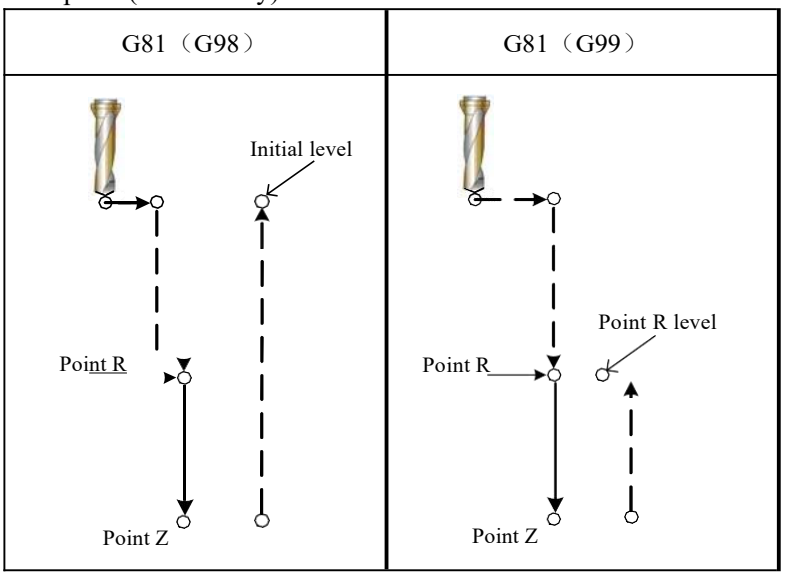


Fig. 3-4-2-1

Z, R: If either of hole bottom parameter Z and R is missing when the first drilling is executed, the system only changes the mode, with no Z axis action executed.

After positioning along X axis and Y axis, rapid traverse is performed to point R. Drilling from point R to point Z is performed, the tool is then retracted in the rapid traverse.

Miscellaneous function M codes are used to rotate the spindle before G81 is specified.

When G81 and an M code are specified in the same block, the M code is executed at the time of the first hole positioning, the system then proceeds to the next drillingoperation.

When the number of repeats K is specified, the M code is only performed for the first hole. For the other holes, it is not performed.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with a canned cycle instructioin, the offset is added or cancelled at the time of positioning to point R level; when the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Example:

M3 S2000	Spindle starts to rotate
G90 G99 G81 X300 Y-250 Z-150 R-10 F120;	Positioning, drill hole 1, then return to point R level Y-550.;
	Positioning, drilling hole 2, then return to point R level
Y-750.;	Positioning, drilling hole 3, then return to point R level
X1000.;	Positioning, drill hole 4, then return to point R level
Y-550.;	Positioning, drill hole 5, then return to point R level
G98 Y-750.;	Positioning, drill hole 6, then return to initial level G80;
G28 G91 X0 Y0 Z0 ;	Return to reference point
M5;	Spindle stops
M30;	

Limitation: When G81 is used, G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G81 cannot be specified in the same block, otherwise, G81 is replaced by other G codes in group 01.

Tool offset: The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

4.4.3 Drilling cycle, counterboring cycle G82

Format: **G82 X_ Y_ Z_ R_ P_ F_ K_;**

Function: This cycle is used for normal drilling. Cutting feed is performed to the bottom of the hole. At the bottom, a dwell is performed, and the tool is then retracted from the bottom of the hole in rapid traverse.

Explanation:

X_Y_: Hole positioning data;

Z_: In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom;

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R;

F_: Cutting feedrate;

P_: The minimum dwell time at the hole bottom; K_:

Number of repeats.

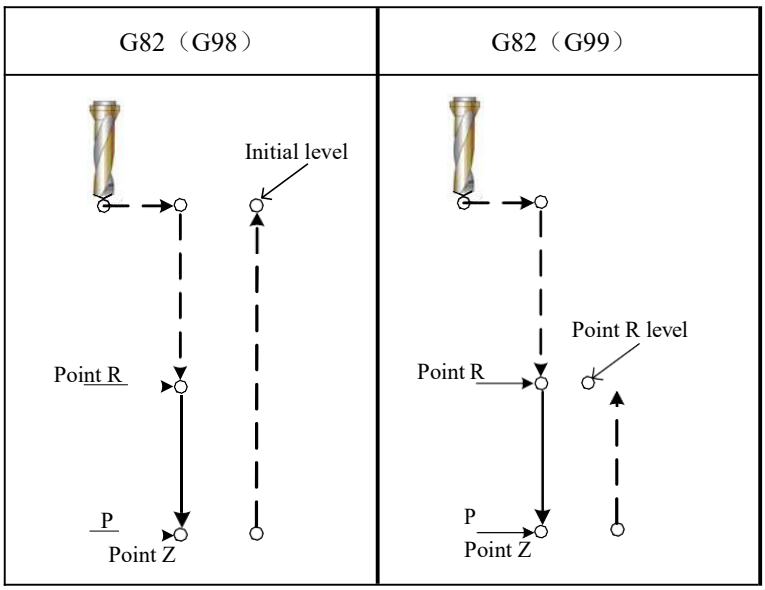


Fig. 3-4-3-1

After positioning along axes X and Y, rapid traverse is performed to point R, and drilling is then performed from point R to point Z. When the tool reaches the bottom of the hole, a dwell is performed and the tool is then retracted in rapid traverse.

Miscellaneous function M codes are used to rotate the spindle before G82 is specified.

When G82 and an M code are specified in the same block, the M code is executed at the time of the first hole positioning, and the system then proceeds to the next drilling operation.

When the number of repeats K is specified, the M code is only executed for the first hole. It is not executed for the other holes.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with the canned cycle code, the offset is added or cancelled at the time of positioning to point R level; when the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

P is a modal code, with its min. value set by data parameter P281 and its max. value by P282. If P value is less than the value set by P281, the min. value takes effect; if P value is more than the value set by P282, the max. value takes effect. P cannot be stored as modal data f it is specified in a block that does not perform drilling.

Example:

M3 S2000 Spindle starts to rotate

G90 G99 G82 X300 Y-250 Z-150 R-100 P1000 F120; Positioning, drill hole 1, dwell for 1s at the hole bottom, then return to point R Y-550; Positioning, drill hole 2, dwell for 1s at the hole bottom, then return to pointR

Y-750; Positioning, drill hole 3, dwell for 1s at the hole bottom, then return to point R X1000.;

Positioning, drill hole 4, dwell for 1s at the hole bottom, then return to point R Y-550;

Positioning, drill hole 5, dwell for 1s at the hole bottom, then return to pointR

G98 Y-750; Positioning, drill hole 6, dwell for 1s at the hole bottom, then return to initial level G80; Cancel the canned cycle

G28 G91 X0 Y0 Z0 ; Return to the reference point

M5; Spindle stops

M30;

Limitation: G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G82 cannot be specified in the same block, otherwise G82 is replaced by other G codes in group 1.

Tool offset:The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

4.4.4 Drilling Cycle with Chip Removal G83

Format: G83 X_ Y_ Z_ R_ Q_ F_ K_

Function: It is used for peck drilling. It performs intermittent cutting feed to the bottom of the hole while removing the chips from the hole.

Explanation:

X_Y_:Hole positioning data;

Z_:In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom;

R_:In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R;

Q_:Cut depth for each cutting feed;

F_:Cutting feedrate;

K_:Repetitive number.

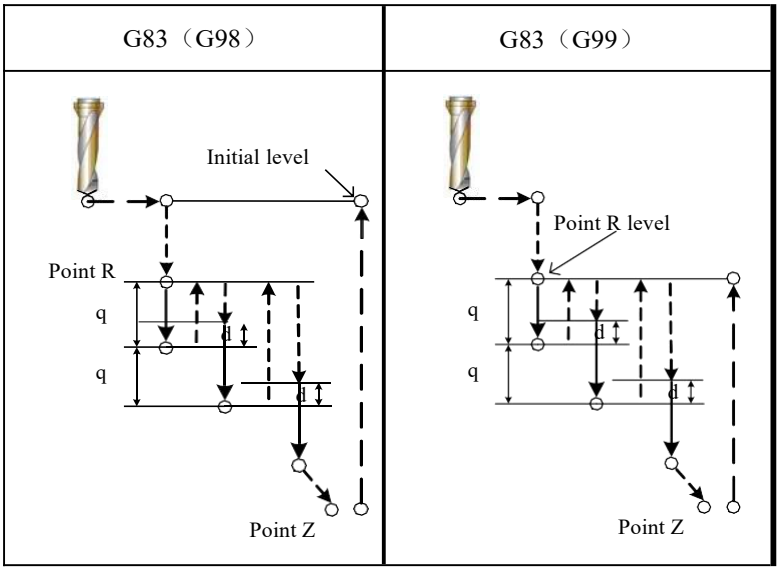


Fig. 3-4-4-1

Q: It specifies the cut depth for each cutting feed, which must be specified as an incremental value. In the second and the subsequent cutting feed, the tool rapidly traverses to the position which has a distance d to the end position of the last drilling and then performs the cutting feed again. d is set by parameter P271, as is shown in Fig. 3-4-4-1.

Specify a positive value for Q, and a negative one will be processed as its absolute value. Specify Q in a drilling block.

If it is specified in the block containing no drilling, it is stored as modal data. Miscellaneous function M codes are used to rotate the spindle before G83 is specified.

When G83 and an M code are specified in the same block, the M code is executed at the time of the first hole positioning, and the system then proceeds to the next drilling operation.

When the number of repeats K is specified, the M code is only executed for the first hole, but not for the other holes.

Note 1: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

Note 2: When the bit parameter NO:43# 1=0, no alarm will be issued if there is no cut-in value specified in the peck drilling (G73， G83). At this moment, if the code parameter Q is not specified or it is 0, the system performs the hole positioning in XY plane, but it does not perform the drilling operation. When the bit parameter NO:43#1=1, an alarm will be issued if no cut-in value is specified in the peck drilling (G73， G83), i.e. an alarm "0045:Address Q not found or set to 0（G73/G83）" occurs when the code parameter Q is not specified or it is 0. If the Q value is negative, the system uses its absolute value to perform intermittent feeding.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with the canned cycle code, the offset is added or cancelled at the time of positioning to point R level; when the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Example:

M3 S2000;	Spindle starts to rotate
G90 G99 G83 X300 Y-250 Z-150 R-100 Q15 F120;	Positioning, drill hole 1, then return to point R
Y-550;	Positioning, drill hole 2, then return to point R
Y-750;	Positioning, drill hole 3, then return to point R
X1000;	Positioning, drill hole 4, then return to point R
Y-550;	Positioning, drill hole 5, then return to point R
G98 Y-750;	Positioning, drill hole 6, then return to initial level G80;
G28 G91 X0 Y0 Z0 ;	Return to the reference point
M5;	Spindle stops
M30;	

Limitation: G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G83 cannot be specified in the same block, is replaced by other G codes in group 1.

Tool offset: The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

4.4.5 Tapping Cycle G74 (or G84)

Format:: G74/G84 X_ Y_ Z_ R_ P_ F_K_

Function: in the tapping cycle, when the tapping axis reaches the hole bottom, the execution pauses, and then the spindle rotates reversely to retract the tapping axis. (G74 is a left-handed tapping cycle and G84 is right-handed rotation tapping) .

Explanation:

X_Y_： Hole positioning data;

Z_： In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom;

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R;

P_： dwell time at the hole bottom;

F_： Cutting federate in tapping;

K_： Repetitive number（specified if necessary）

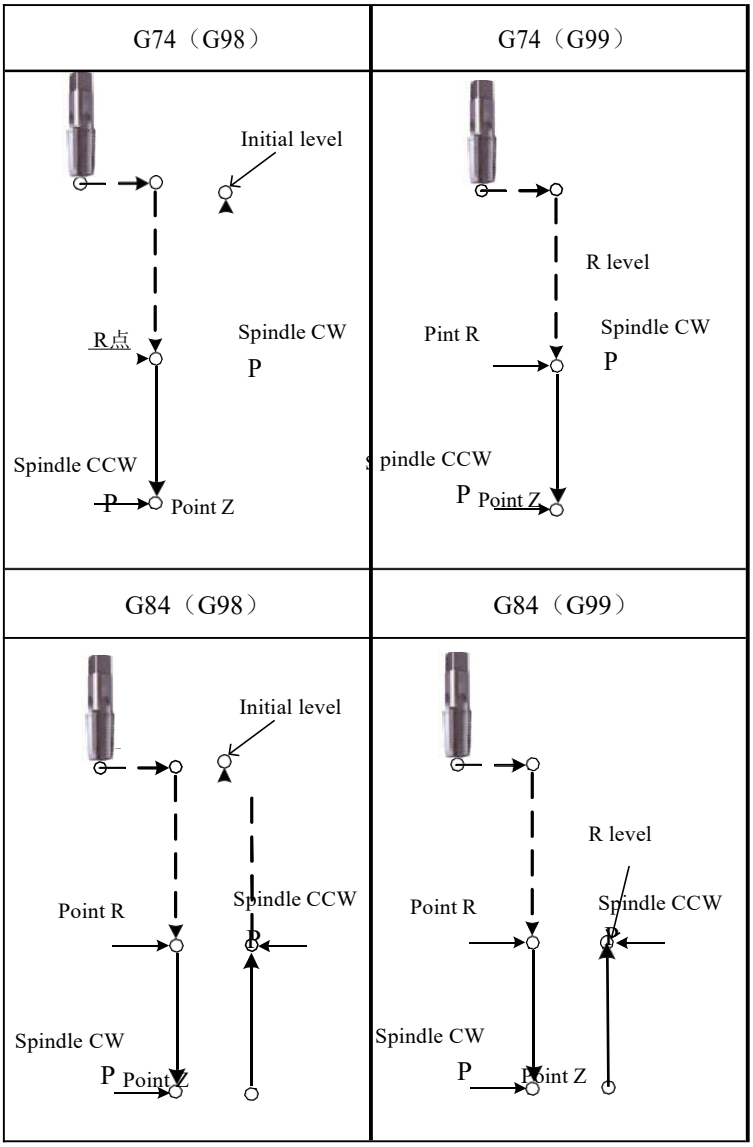


Fig. 3-4-5-1

When G74 is commanded, taping is performed by rotating the spindle clockwise (when G84 is commanded, the spindle rotates counterclockwise). When the bottom of the hole is reached, the spindle is rotated in the reverse direction for retraction. This operation creates threads.

Example:

G94	The spindle starts to rotate;
M29 S1000 ;	The spindle orientates and its speed is specified;
G43 / G44 H10 ;	Call the tool length compensation;
G90 G99 G74 / G84 X100 Y110 Z -50 R5 P3000 F100;	Position, tap hole 1, and return to point R; Y150; Position, tap hole 2, and return to point R;
G91 X50 K5;	X100 , Y150 as a reference point, along X axis; 50mm is the increment unit, execute 5 times tapping;

G98 Y-750;	Position, tap hole 8, and return to the initial point;
G80;	Cancle the tapping cycle;
G28 G91 X0 Y0 Z0 ;	Return to the reference point;
M30;	End of program;

Tool length compensation: when the tool length compensation G43, G44 or G49 is in the same block with a fixed cycle command, and the tool positions to point R, and simultaneously executes the offset or cancels the offset; when the tool compensation instruction G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Thread lead: in feed per minute, relationship between the thread lead and feedrate, spindle speed: Feedrate F=screw pitch×spindle speed S

Example：machining the thread hole M12×1.5, can select a parameter;

S500=500 r /min; F=1.5×500=750mm/min;

When a multi-head thread is machined, multiplying the number of head can gain the F value.

In feed per rev, the thread lead is equal to the feederate.

Example:

Feed per minute mode:		Feed per rev mode:	
Spindle speed 1000r/min;		Spindle speed 1000r/min;	
Thread lead 1.0mm;		Thread lead 1.0mm;	
So Z axis feedrate= 1000*1=1000mm/min;	feed per minute mode	So Z axis feedrate =thread lead =1mm / r; G94 G95	feed per rev mode G00
X120 Y100;	position	G00 X120 Y100;	position
M29 S1000 ;	specify rigid mode	M29 S1000 ;	specify rigid mode G84 Z-
100 R-20 F1000;	right-hand rigid tapping	G84 Z-100 R-20 F1;	right-hand rigid tapping
G80	cancel tapping cycle	G80	cancel tapping cycle
G28 G91 X0 Y0 Z0	return to reference point	G28 G91 X0 Y0 Z0	return to reference point
M30	end of program	M30	end of program

Limitation:

G codes: in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) when G74/G84 is used, otherwise, G74/G84 is replaced by other G codes in group1.

M codes: before G74/G84 is specified, using the miscellaneous function M code makes the spindle rotate. When the spindle rotation is not specified, the system automatically count the current spindle command speed on the R plane, and then the spindle is regulated to clockwise rotation(74)/counterclockwise (G84).

when G74/G84 and an M code are specified in the same block, the M code is executed while the 1st hole positioning operation, then the system proceeds to the next drilling operation.

If number of repeats K is specified, the M code is only executed for the 1st hole.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

S instruction: when the commanded spindle speed exceeds the max. spindle speed during tapping

(P257: the spindle upper speed in the course of tapping cycle), an alarm occurs; the gear of the max. spindle speed during the rigid tapping is determined by P294~P296.

F instruction: when the specified F value exceeds the cutting feedrate's upper value (P96 sets the upper value), the system takes the upper value as the reference.

P instruction： P is a modal code, the least value is set by P281, the max. value is set by P282. P value is less than the least value, and the system runs with the least value; when it is more than the max. value, the system run with the max. value.

Axis switch: must cancel the fixed cycle before switching the tapping axis. No. 206 alarm occurs when the tapping axis is changed in the rigid tapping mode.

Override: during tapping, the feedrate and spindle speed override are defaulted into 100%, and the machine does not stop during the feed hold key being pressed till the return operation is completed.

Tool radius compensation: in the fixed cycle command, the command function does not need executing the tool radius compensation, so, the tool radius compensation is ignored.

Program restart: the program restart function is invalid in tapping cycle.

Tool radius offset: The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

4.4.6 Fine boring cycle G76

Format: G76 X_Y_Z_Q_R_P_F_K_

Function: This cycle is used for boring a hole precisely.。

When the tool reaches the hole bottom, the spindle stops and the tool is moved away from the machined surface of the workpiece and retracted.

Prevent the retraction trail from affecting the machined surface smoothness and avoid the tool damage in the operation.

Explanation:

X_Y_: Hole positioning data

Z_: In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom.

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R level.

Q_: Offset at the hole bottom

P_: Dwell time at the hole bottom F_:

Cutting feedrate.

K_: Reptitive number of fine boring

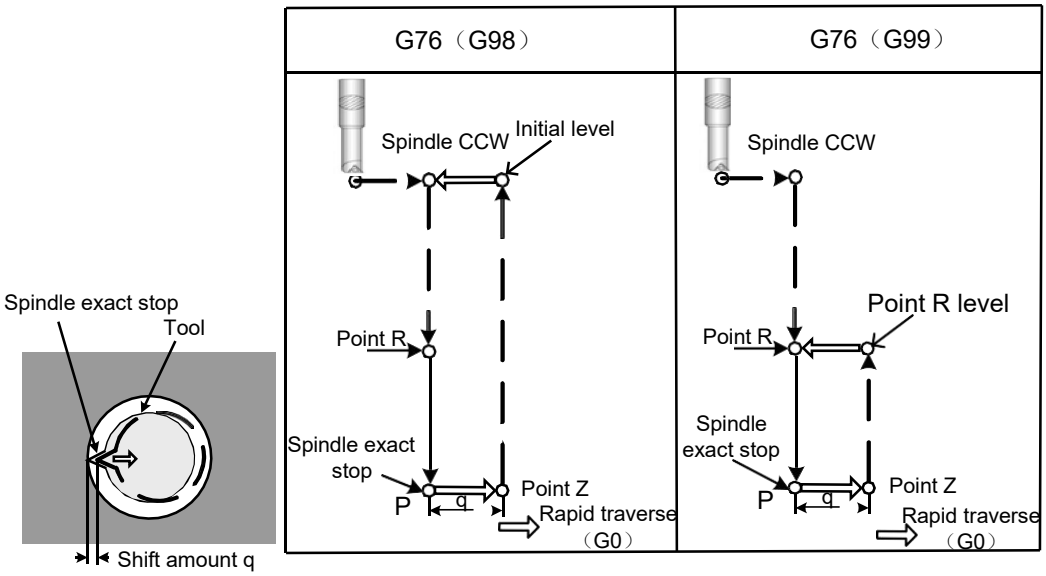


Fig. 3-4-6-1

When the tool reaches the bottom of the hole, the spindle stops at a fixed rotation position and the tool is moved in the direction opposite to the tool nose for retraction. This ensures that the machined surface is not damaged and enables precise and efficient boring. The retraction distance is specified by the parameter Q, and the retraction axis and direction are specified by bit parameter NO.42#4 and NO.42#5 respectively. The value of Q must be positive. If it is a negative value, the negative sign is ignored. The hole bottom shift amount of Q is a modal value saved in canned cycle which must be specified carefully because it is also used as the cutting depth for G73 and G83.

Before specifying G76, use a miscellaneous function (M code) to rotate the spindle.

If G76 and an M code are specified in the same block, the M code is executed at the time of the 1st hole positioning operation, then the system proceeds to the next boring operation.

If the number of repeats K is specified, the M code is only executed for the 1st hole, for the other holes, the M code is not executed.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with the canned cycle code, the offset is added or cancelled at the time of positioning to point R level; If the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Axis switching: The canned cycle must be canceled before the drilling axis is changed. Boring: In a block that does not contain X , Y , Z, or other axes, boring is not performed.

Example:

M3 S500; Spindle starts to rotate

G90 G99 G76 X300 Y-250 Z-150 R-100 Q5 P1000 F120;Positioning, bore hole 1, then return to point R; Orient at the bottom of the hole, then shift by 5mm; Stop at the bottom of the hole for 1s

Y-550; Positioning, bore hole 2, then return to point R

Y-750; Positioning, bore hole 3, then return to point R

X1000; Positioning, bore hole 4, then return to point R

Y-550; Positioning, bore hole 5, then return to point R G98 Y-750; Positioning, bore hole 6, then return to initial level G80

G28 G91 X0 Y0 Z0; Return to the reference point

M5; Spindle stops

Limitation: G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G76 cannot be specified in the same block, otherwise G76 is replaced by other G codes in group 1.

Tool offset:The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

Note: In the instruction, the tool infeed axis and the tool infeed direction are fixed, and the tool infeed direction is not influenced by G68 coordinate system rotation.

4.4.7 Boring cycle G85

Format: G85 X_ Y_ Z_ R_ F_ K_ Function:

This cycle is used for boring a hole. **Explanation:**

X_Y_: Hole positioning data
Z_: In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom.
R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R.
F_: Cutting feedrate.
K_: Repetitive number

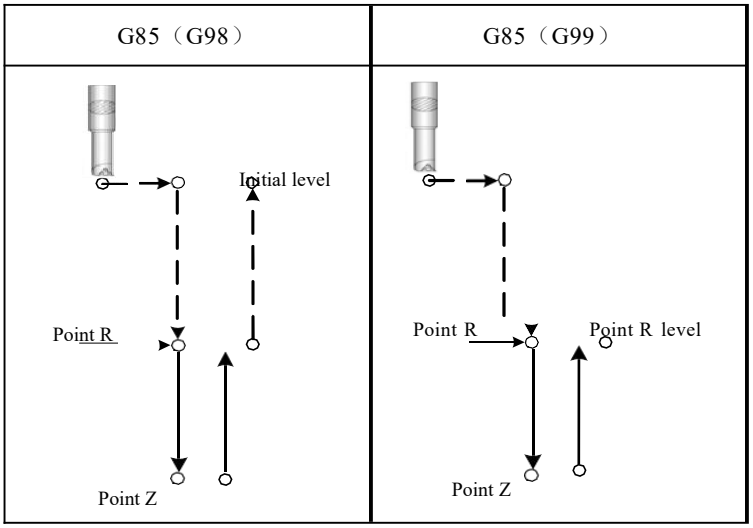


Fig. 3-4-7-1

After positioning along X and Y axes, rapid traverse is performed to point R, and boring is performed from point R to point Z. As the tool reaches the hole bottom, cutting feed is performed to return to point R level.

Use a miscellaneous function (M code) to rotate the spindle before specifying G85.

If G85 and an M code are specified in the same block, the M code is executed at the time of the 1st hole positioning operation, then the system proceeds to the next boring operation.

If the number of repeats K is specified, the M code is only executed for the 1st hole.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with the canned cycle code, the offset is added or canceled at the time of positioning

to point R level; If the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Axis switching: The canned cycle must be canceled before the drilling axis is changed.

Boring: Boring is not performed in a block which does not contain X, Y, Z or otheraxes.

Example：

M3 S100 ;	The spindle starts to rotate
G90 G99 G85 X300 Y-250 Z-150 R-120 F120;	Positioning, bore hole 1, then return to point R
Y-550;	Positioning, bore hole 2, then return to point R
R Y-750;	Positioning, bore hole 3, then return to point R
X1000;	Positioning, bore hole 4, then return to point R
Y-550;	Positioning, bore hole 5, then return to point R
G98 Y-750;	Positioning, bore hole 6, then return to initial level G80;
G28 G91 X0 Y0 Z0 ;	Return to the reference point
M5;	Spindle stops
M30;	

Limitation: G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G85 cannot be specified in a same block, otherwise G85 is replaced by other G codes in group 1.

Tool offset: The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

4.4.8 Boring cycle G86

Format： **G86 X_ Y_ Z_ R_ F_ K_;**

Function: This cycle code is used to perform a boring cycle(the dwell operation is not required when the tool is at the bottom of hole).

Explanation：

X_Y_: Hole positioning data;
Z_: In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom;
R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R;
F_: Cutting feedrate;
K_: Repetitive number.

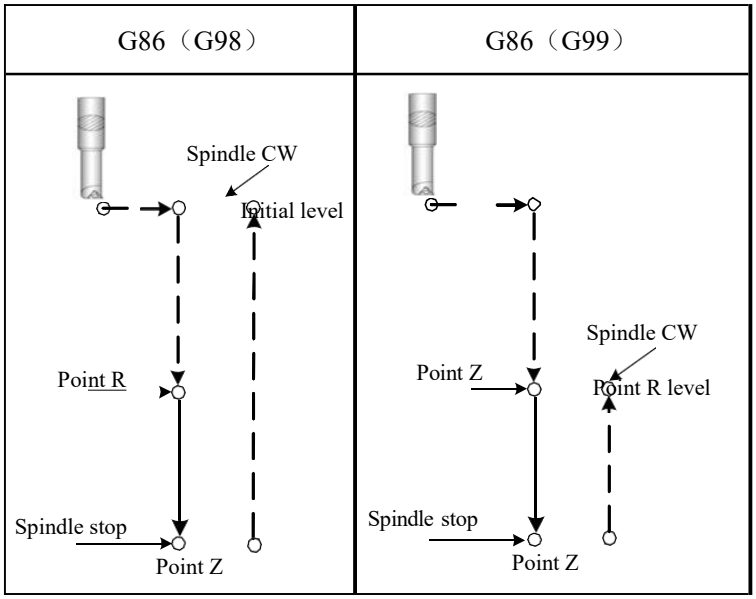


Fig.3-4-8-1

After positioning along X and Y axes, rapid traverse is performed to point R. And boring is performed from point R to point Z. When the spindle stops at the bottom of the hole, the tool is retracted in rapid traverse.

Before specifying G86, use a miscellaneous function (M code) to rotate the spindle.

If G86 and an M code are specified in the same block, the M code is executed at the time of the 1st hole positioning operation, then the system proceeds to the next boring operation.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with the canned cycle code, the offset is added or cancelled at the time of positioning to point R level; If the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Axis switching: The canned cycle must be cancelled before the drilling axis is changed. Boring:

Boring is not performed in a block which does not contain X, Y,Z or other axes.

Example:

M3 S2000; Spindle starts to rotate

G90 G99 G86 X300 Y-250 Z-150 R-100 F120 Positioning, bore hole 1, then return to Point R Y-550;

Positioning, bore hole 2, then return to Point R

Y-750; Positioning, bore hole 3, then return to Point R

X1000; Positioning, bore hole 4, then return to Point R

Y-550; Positioning, bore hole 5, then return to Point R G98 Y-

750; Positioning, bore hole 6, then return to initial level G80;

G28 G91 X0 Y0 Z0 ; Return to the reference point M5;

Spindle stops

M30;

Limitation: G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G86 cannot be specified in the same block, otherwise G86 is replaced by other G codes in group 1.

Tool offset: The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

4.4.9 Boring cycle, back boring cycle G87

Format: G87 X_Y_Z_R_Q_P_ F_;

Function: This cycle performs accurate boring.

Explanation:

X_Y_: Hole positioning data

Z_: In incremental programming it specifies the distance from point R level to point Z level; in absolute programming it specifies the absolute coordinates of the point Z level.

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R level (hole bottom)

Q_: Shift amount at the bottom of the hole P_:

Minimum dwell time at the hole bottom F_:

Cutting feedrate

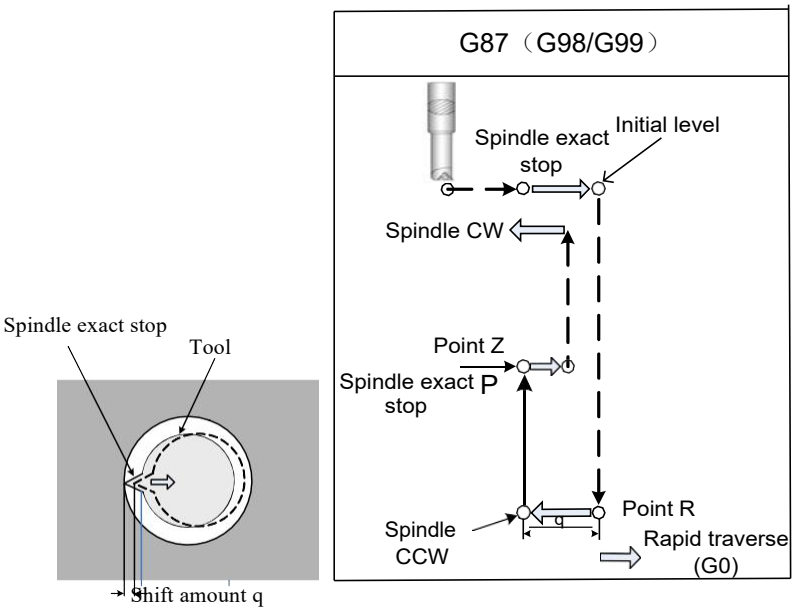


Fig. 3-4-9-1

After positioning along X and Y axes, the tool is stopped after spindle orientation. Then the tool is moved in the direction opposite to the tool nose, and positioning (rapid traverse) is performed to the hole bottom point R. The tool is then shifted in the direction of the tool nose and the spindle is rotated counterclockwise. Boring is performed in the positive direction along Z axis until point Z is reached. At point Z, the spindle is stopped at the fixed rotation position after it is oriented again, and the tool is retracted in the direction opposite to the tool nose, then it is returned to the initial level. The tool is then shifted in the direction of the tool nose and the spindle is rotated counterclockwise to proceed to the next block operation.

The parameter Q specifies the retraction distance. The retraction direct on and retraction axis are set by system parameter NO:42#4 and NO:42#5 respectively. Q must be a positive value, if it is

specified with a negative value, the negative sign is ignored. The hole bottom shift amount of Q is a modal value retained in the canned cycle, which must be specified carefully because it is also used as the cutting depth for G73 and G83.

Before specifying G87, use a miscellaneous function (M code) to rotate the spindle.

If G87 and an M code are specified in the same block, the M code is executed at the time of the 1st hole positioning operation, then the system proceeds to the next boring operation.

If number of repeats K is specified, the M code is only executed for the 1st hole.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the block with the canned cycle code, the offset is added or cancelled at the time of positioning to point R level; If the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

The canned cycle can only be executed in G17 plane.

Boring: In a block which contains no X, Y, Z or other additional axes, boring is not performed.

Note: The values of Z and R must be specified when the back boring cycle is being programmed. In general, point Z is located above point R, otherwise an alarm occurs.

Example：

```
M3 S500;                    Spindle starts to rotate
G90 G99 G87 X300. Y-250. Z-120. R-150. Q5. P1000 F120;
(Positioning, bore hole 1, orient at the initial level then shift by 5mm and dwell at point Z for 1s) Y-550;
                            Positioning, bore hole 2, then return to point R level
Y-750;                      Positioning, bore hole 3, then return to point R level
X1000;                      Positioning, bore hole 4, then return to point R level
Y-550;                      Positioning, bore hole 5, then return to point R level
G98 Y-750.;                 Positioning, bore hole 6, then return to initial level G80
G28 G91 X0 Y0 Z0;           Return to the reference point
M5;                         Spindle stops
```

Limitation： G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G87 cannot be specified in the same block, otherwise G87 is replaced by other G codes in group 1.

Tool offset: The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

Note: In the instruction, the tool infeed axis and the tool infeed direction are fixed, and the tool infeed direction is not influenced by G68 coordinate system rotation.

4.4.10 Boring Cycle G88

Format: G88 X_Y_Z_R_ P_F_ Function:

This cycle is use for boring a hole. **Explanation:**

X_Y_:Hole positioning data;

Z_:In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom;

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R;

P_: Dwell time at the bottom of the hole; F_:
Cutting feedrate.

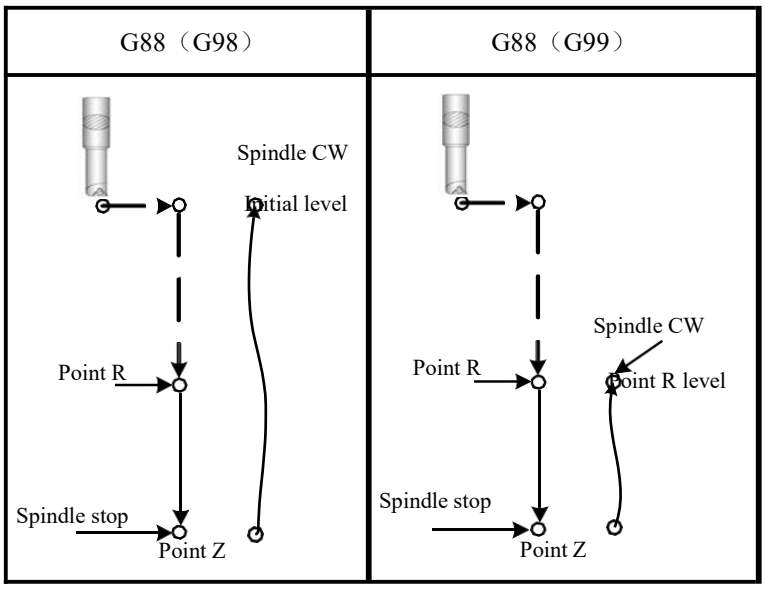


Fig. 3-4-10-1

After positioning along X and Y axes, rapid traverse is performed to point R. Boring is performed from point R to point Z. When boring is completed, a dwell is performed then the spindle is stopped. The tool is manually retracted from point Z at the hole bottom to point R (in G99) or the initial level (in G98) and the spindle is rotated CCW.

Before specifying G88, use a miscellaneous function (M code) to rotate the spindle.

If G88 and an M code are specified in the same block, the M code is executed at the time of the 1st hole positioning operation, then the system proceeds to the next drilling operation.

If the number of repeats K is specified, the M code is only executed for the 1st hole.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

P is a modal code, with its min. value set by data parameter P281 and max. value by P282. If P value is less than the value set by P281, the min. value takes effect; if P value is more than the value set by P282, the max. value takes effect. P cannot be stored as modal data if it is specified in a block that does not perform drilling.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with the canned cycle code, the offset is added or cancelled at the time of positioning to point R level; If the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Axis switching: Before the boring axis is changed, the canned cycle must be cancelled. Boring: In a block which contains no X, Y, Z or other additional axes, boring is not performed.

Example:

M3 S2000	Spindle starts to rotate
G90 G99 G88 X300. Y-250. Z-150. R-100. P1000 F120.	Positioning, bore hole 1, then return to point R
Y-550;	Positioning, bore hole 2, then return to point R
Y-750;	Positioning, bore hole 3, then return to point R
X1000;	Positioning, bore hole 4, then return to point R
Y-550;	Positioning, bore hole 5, then return to point R
G98 Y-750;	Positioning, bore hole 6, then return to initial level G80
G28 G91 X0 Y0 Z0;	Return to the reference point
M5;	Spindle stops

Limitation: G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1)

and G88 cannot be specified in the same block, otherwise G88 is replaced by other G codes in group 1.

Tool offset: The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

4.4.11 Boring cycle G89

Format: G89 X_ Y_ Z_ R_ P_ F_ K_

Function: This cycle is used for boring a hole.

Explanation:

X_Y_: Hole positioning data;

Z_: In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom;

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R;

P_: Minimum dwell time at the bottom of the hole; F_: Cutting feedrate;

K_: Repetitive number .

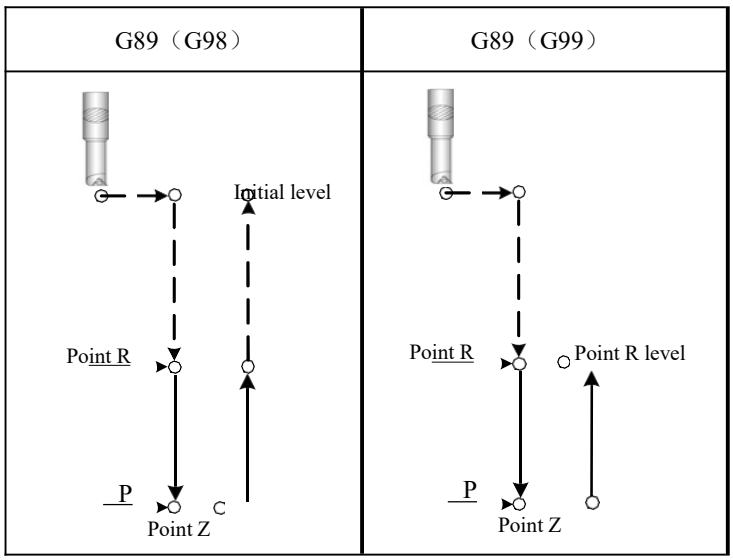


Fig.3-4-11-1

This cycle is almost the same as G85. The difference is that this cycle performs a dwell at the hole bottom.

Before specifying G89, use a miscellaneous function (M code) to rotate the spindle.

If G89 and an M code are specified in the same block, the M code is executed while the 1st hole positioning operation, then the system proceeds to the next drilling operation.

If number of repeats K is specified, the M code is only executed for the 1st hole.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

P is a modal code, with its min. value set by data parameter P281 and max. value by P282. If P value is less than the value set by P281, the min. value takes effect; if P value is more than the value set by P282, the max. value takes effect. P cannot be stored as modal data if it is specified in a block that does not perform drilling.

Tool length compensation: If the tool length compensation code G43, G44 or G49 is specified in the same block with the canned cycle code, the offset is added or cancelled at the time of positioning to point R level; If the tool compensation code G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Axis switching: Before the boring axis is changed, the canned cycle must be cancelled.

Boring: In a block that does not contain X, Y, Z, R or any additional axes, boring is not performed.

Example:

M3 S100 Spindle starts to rotate G90
G99 G89 X300. Y-250. Z-150. R-120. P1000 F120.
Positioning, bore hole 1, return to point R level, then stop at the hole bottom for 1s Y-550;
Positioning, bore hole 2, then return to point R level
Y-750; Positioning, bore hole 3, then return to point R level
X1000; Positioning, bore hole 4, then return to point R level
Y-550; Positioning, bore hole 5, then return to point R level
G98 Y-750; Positioning, bore hole 6, then return to initial level G80;
G28 G91 X0 Y0 Z0; Return to the reference point M5;
Spindle stops
M30;

Limitation: G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G89 cannot be specified in the same block, otherwise G89 is replaced by other G codes in group 1.

Tool offset: The tool radius offset is ignored at the time of the canned cycle positioning because the instruction function does not execute the tool radius compensation.

4.4.12 Canned cycle cancel G80

Format: **G80**

Function: It is used for cancelling the canned cycle.

Explanation:

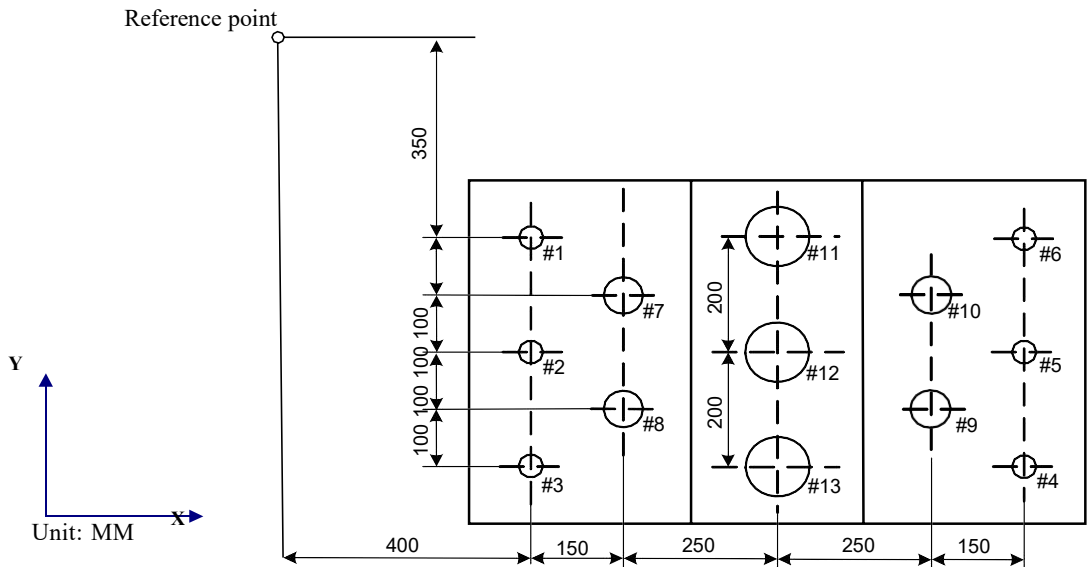
All the canned cycles are cancelled to perform normal operation. Point R, point Z are also cancelled, and the other drilling and boring data is cleared as well.

Example:

M3 S100; Spindle starts to rotate G90
G99 G88 X300 Y-250 Z-150 R-120 F120;
Positioning, bore hole 1, then return to point R
Y-550; Positioning, bore hole 2, then return to point R
Y-750; Positioning, bore hole 3, then return to point R
X1000; Positioning, bore hole 4, then return to point R
Y-550; Positioning, bore hole 5, then return to point R
G98 Y-750; Positioning, bore hole 6, then return to the initial level G80;
G28 G91 X0 Y0 Z0; Return to the reference point and cancel the canned cycle M5;

Example:

Explanation for the usage of the canned cycle using the tool length compensation:



 # 1～ 6... drilling of a Φ10 hole # 7～ 10... drilling of a Φ20 hole #11～13.. boring of a Φ95 hole

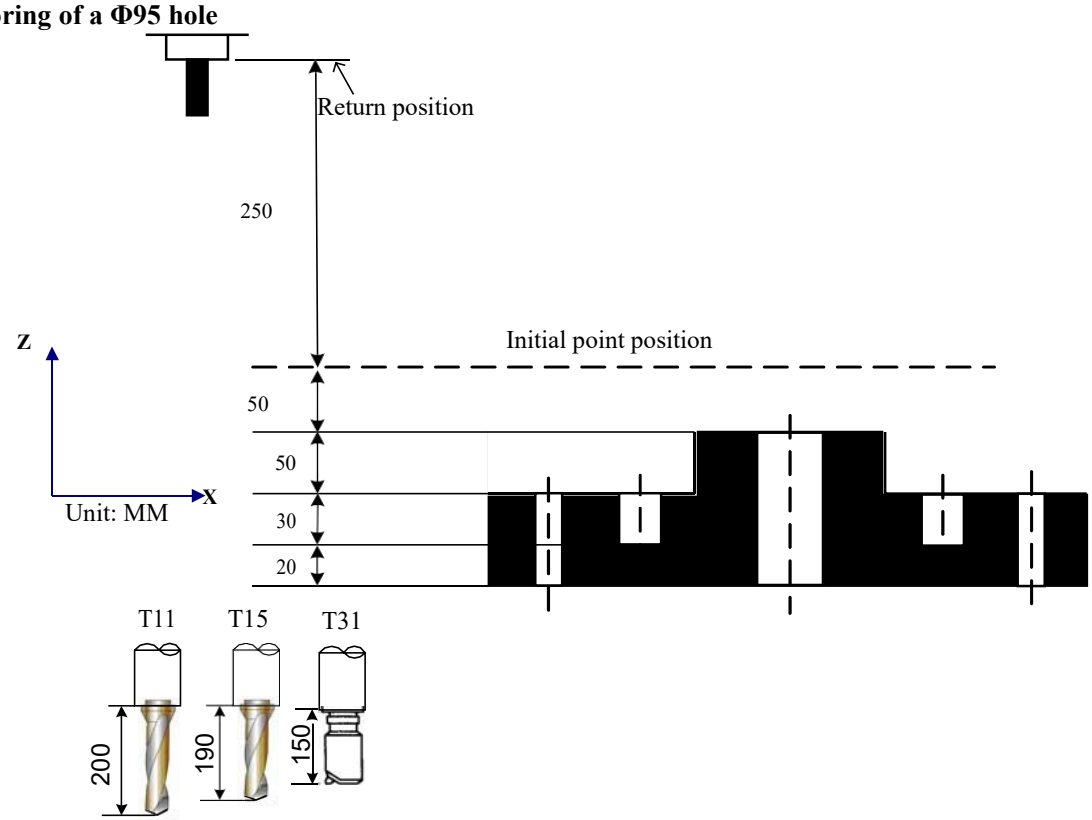


Fig. 3-4-12-1

The values of offset No.11, No. 15 and No. 31 are set to 200, 190 and 150 respectively. The program is as follows:

N001 G92 X0 Y0 Z0 ；	The coordinate system is set at reference point. N002
G90 G00 Z250 T11 M6 ；	Tool change.
N003 G43 Z0 H11 ；	Tool length compensation at the initial point.
N004 S300 M3 ；	Spindle start.
N005 G99 G81 X400 Y-350 ； Z-153 R-97 F120 ；	Positioning, then hole #1 drilling.

N006 Y-550 ；	Positioning, then hole #2 drilling and point R level return.
N007 G98 Y-750 ；	Positioning, then hole #3 drilling and initial level return.
N008 G99 X1200 ；	Positioning, then hole #4 drilling and point R level return.
N009 Y-550 ；	Positioning, then hole #5 drilling and point R level return.
N010 G98 Y-350 ；	Positioning, then hole #6 drilling and initial level return.
N011 G00 X0 Y0 M5 ；	Reference point return, then spindle stop.
N012 G49 Z250 T15 M6 ；	Tool length compensation cancel, then tool change.
N013 G43 Z0 H15 ；	Initial level, tool length compensation.
N014 S200 M3 ；	Spindle start.
N015 G99 G82 X550 Y-450 ； Z-130 R-97 P30 F70 ；	Positioning, then hole #7 drilling and point R level return.
N016 G98 Y-650 ；	Positioning, then hole #8 drilling and initial level return.
N017 G99 X1050 ；	Positioning, then hole #9 drilling and point R level return.
N018 G98 Y-450 ；	Positioning, then hole #10 drilling and initial level return.
N019 G00 X0 Y0 M5 ；	Reference point return, spindle stop.
N020 G49 Z250 T31 M6 ；	Tool length compensation cancel, tool change. N021
G43 Z0 H31 ；	Initial level, tool length compensation.
N022 S100 M3 ；	Spindle start.
N023 G85 G99 X800 Y-350 ； Z-153 R47 F50 ； N024	Positioning, then hole #11 drilling and point R level return.
G91 Y-200 ； Y-200 ；	Positioning, then holes #12 and #13 drilling and point R level return.
N025 G00 G90 X0 Y0 M5 ；	Reference point return, spindle stop. N026
G49 Z0 ；	Tool length compensation cancel.
N027 M30 ；	Program sto.

4.5 Rigid Tapping G Code

4.5.1 Left-Hand Tapping Cycle G74

Format：G74 X_Y_Z_R_P_F_K_

Function: The spindle is rotated in the reverse direction when the bottom of the hole is reached in this tapping cycle.

Explanation:

X_Y_：Hole positioning data

Z_：In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom

R_：In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R.

P_：Minimum dwell time at the hole bottom. The absolute value is used if it is a negative one. F_：Cutting feedrate.

K_：Repetitive number.（specify it if neccessary）

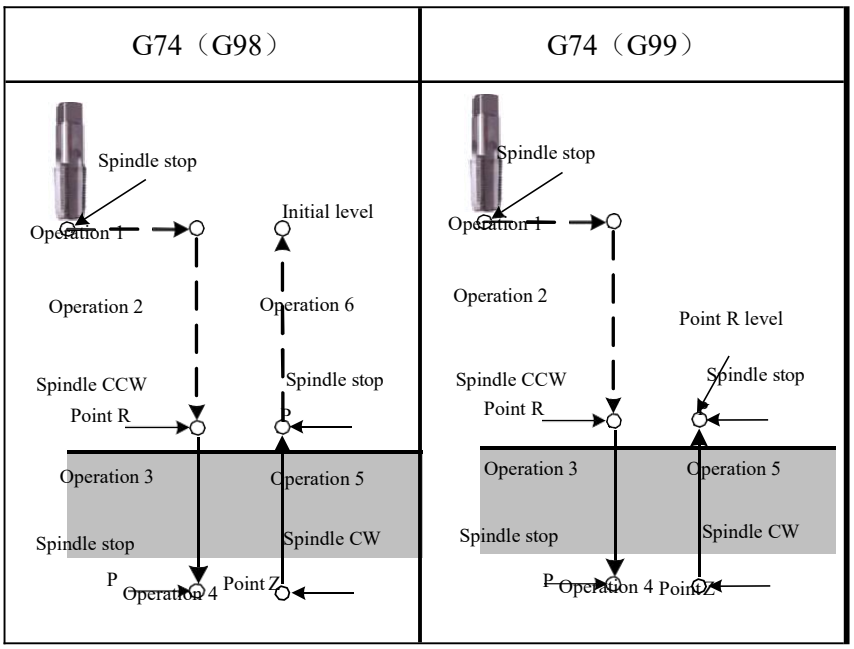


Fig. 3-5-1-1

After positioning along X and Y axes, rapid traverse is performed along Z axis to point R level. The spindle is rotated CW for tapping from point R level to Z level by G74 instruction. When tapping is completed, the spindle is stopped and a dwell is performed. The spindle is then rotated in the reverse direction, the tool is retracted to point R level, then the spindle is stopped. Rapid traverse is then performed to initial level. When the tapping is being performed, the feedrate override and the spindle override are assumed to be 100%.

Rigid mode: in position mode (NO:46#1 is set to 1，K parameter NO:7#7 to 1), before the tapping code, specifying M29 S***** can specify the rigid mode.

Tool length compensation: If the tool length compensation instruction G43, G44 or G49 is specified in the same block with the canned cycle instruction, the offset is added or cancelled at the time of positioning to point R level; when the tool compensation instruction G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Thread lead: in feed per minute, relationship between the thread lead and feedrate, spindle spindle: Feedrate speed F=screw taper pitch×spindle speed S

Example: machining the thread hole M12×1.5 on a workpiece can select the parameters;

S500=500 r /min；　F=1.5×500=750mm/min；

When a multi-head thread is machined, it multiplies the number of head to get the F value. In feed per rev, the thread lead is equal to the feedate.

Example:

Feed per minute mode

Spindle speed 1000r/min;

Thread lead 1.0mm;

So, Z axis' feedrate= 1000*1=1000mm/min;

	feed per minute mode
X120 Y100;	position
M29 S1000 ;	specify rigid mode
Z-100 R-20 F1000;	left-hand rigid tapping
	cancel tapping cycle
G28 G91 X0 Y0 Z0	return to the reference point
	end of program

Feed per rev mode:

Spindle speed 1000r/min;

Thread lead 1.0mm;

So, Z axis' feedrate =thread lead1=1mm / r; G94

G95	feed per rev mode G00
G00 X120 Y100;	position
M29 S1000 ;	specify rigid mode G74
G74 Z-100 R-20 F1;	left-hand rigid tapping G80
G80	cancel tapping cycle
G28 G91 X0 Y0 Z0	return to the reference point M30
M30	end of program

Limitation:

G code: G codes in 01 group (G00 to G03, G60 modal G code (bit parameter NO: 48#0 is set to 1) and G74 cannot be specified in the same block, otherwise G74 is replaced by other codes in group 01.

M code: Before G74 is specified, using the miscellaneous function M code makes the spindle rotate. When the spindle rotation is not specified, the system automatically count the current spindle command speed on the R plane, and then the spindle is regulated to clockwise rotation.

When G74 and an M code are specified in the same block, the M code is executed while the 1st hole positioning operation, then the system proceeds to the next drilling operation.

If number of repeats K is specified, the M code is only executed for the 1st hole.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

S instruction: when the commanded spindle speed exceeds the max. spindle speed during tapping (P257: the spindle upper speed in the course of tapping cycle), an alarm occurs; the gear of the max. spindle speed during the rigid tapping is determined byP294~P296.

F instruction: when the specified F value exceeds the cutting feedrate's upper value (P96 sets the upper value), the system takes the upper value as the reference.

P instruction: P is a modal code, the least value is set by P281, the max. value is set by P282. P value is less than the least value, and the system runs with the least value; when it is more than the max. value, the system run with the max. value.

Axis switch: must cancel the fixed cycle before switching the tapping axis. No. 206 alarm occurs when the tapping axis is changed in the rigid tapping mode.

Override: during tapping, the feedrate and spindle speed override are defaulted into 100%, and the machine does not stop during the feed hold key being pressed till the return operation is completed.

Tool radius compensation: in the fixed cycle command, the command function does not need executing the tool radius compensation, so, the tool radiuscompensation is ignored.

Program restart: It is invalid during the rigid taping.

Note: when the flexible tapping, rigid tapping or deep-hole rigid tapping is executed, using G97 cancels the constant surface cutting feedrate, otherwise, teeth disorder or broken screw taper exists.

4.5.2 Right-Hand Tapping Cycle G84

Format: **G84 X_Y_Z_R_P_F_K_**

Function: In rigid taping, the spindle motor is controlled as if it were a servo motor, which is used for high-speed and high-precision taping. It keeps the start positions of the taping unchanged if point R is not changed. Even if taping is performed repeatedly in a position, the threads will not be broken.

Explanation:

X_Y_: Hole positioning data;

Z_: In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom;

R_: In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R;

P_: Dwell time at the bottom of the hole, with its absolute value used if it is negative; F_: Cutting feedrate;

K_: Number of repeats(specify it if necessary).

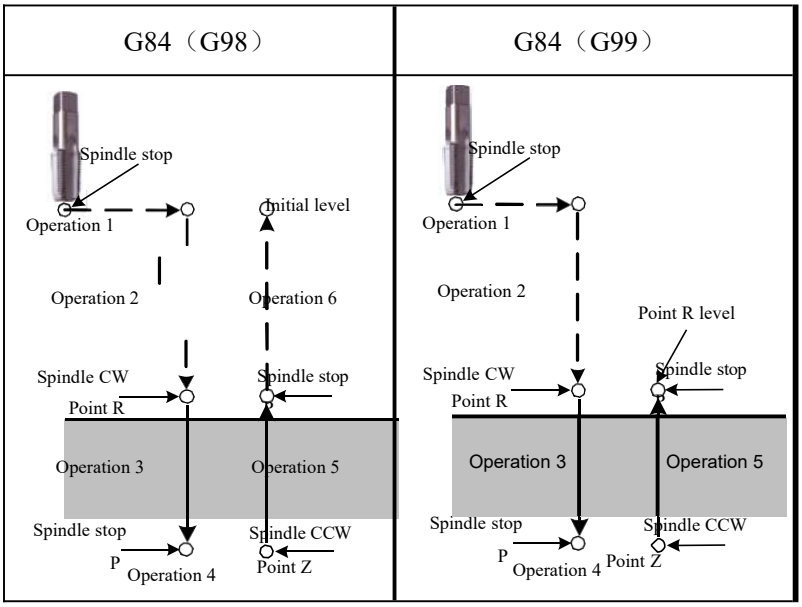


Fig. 3-5-2-1

After positioning along X and Y axes, rapid traverse is performed to point R level along Z axis. The spindle is rotated CCW for tapping from point R level to Z level by G84 instruction. When tapping is completed, the spindle is stopped and a dwell is performed. The spindle is then rotated in the reverse direction, the tool is retracted to point R level, then the spindle is stopped. Rapid traverse to initial level is then performed. When taping is being performed, the feedrate override and spindle override are assumed to be 100%.

Rigid mode: in position mode (NO:46#1 is set to 1， K parameter NO:7#7 to 1), before the tapping code, specifying M29 S***** can specify the rigid mode.

Tool length compensation: If the tool length compensation instruction G43, G44 or G49 is specified

in the same block with the canned cycle instruction, the offset is added or cancelled at the time of positioning to point R level; If the tool compensation instruction G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Thread lead: In feed per minute, relationship between the thread lead and feedrate, spindle spindle: Feedrate speed F=screw taper pitch×spindle speed S

Example: machining the thread hole M12×1.5 on a workpiece can select the parameters;

S500=500 r /min; F=1.5×500=750mm/min;

When a multi-head thread is machined, it multiplies the number of head to get the F value.

In feed per rev, the thread lead is equal to the feedate.

Example:

Feed per minute mode

Spindle speed 1000r/min;

Thread lead 1.0mm;

So, Z axis' feedrate= 1000*1=1000mm/min; feed per minute mode

X120 Y100; position

M29 S1000 ; specify rigid mode

Z-100 R-20 F1000; right-hand rigid tapping

cancel tapping cycle

G28 G91 X0 Y0 Z0 return to the reference point

end of program

Feed per rev mode:

Spindle speed 1000r/min;

Thread lead 1.0mm;

So, Z axis' feedrate =thread lead1=1mm / r; G94

G95 feed per rev mode G00

G00 X120 Y100; position

M29 S1000 ; specify rigid mode G74

G74 Z-100 R-20 F1; right-hand rigid tapping G80

G80 cancel tapping cycle

G28 G91 X0 Y0 Z0 return to the reference point M30

M30 end of program

Limitation:

G code: When G84 is used, G codes in 01 group (G00 to G03), G60 modal G code (bit parameter NO: 48#0 is set to 1) and G84 cannot be specified in the same block, otherwise G84 is replaced by other codes in group 1.

M code: before G84 is specified, using the miscellaneous function M code makes the spindle rotate. When the spindle rotation is not specified, the system automatically count the current spindle command speed on the R plane, and then the spindle is regulated to /counterclockwise.

when G84 and an M code are specified in the same block, the M code is executed while the 1st hole positioning operation, then the system proceeds to the next drilling operation.

If number of repeats K is specified, the M code is only executed for the 1st hole.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

S instruction: when the commanded spindle speed exceeds the max. spindle speed during tapping (P257: the spindle upper speed in the course of tapping cycle), an alarm occurs; the gear of the max. spindle speed during the rigid tapping is determined by P294~P296.

F instruction: when the specified F value exceeds the cutting feedrate's upper value (P96 sets the

upper value), the system takes the upper value as the reference.

P instruction： P is a modal code, the least value is set by P281, the max. value is set by P282. P value is less than the least value, and the system runs with the least value; when it is more than the max. value, the system run with the max. value.

Axis switch: must cancel the fixed cycle before switching the tapping axis. No. 206 alarm occurs when the tapping axis is changed in the rigid tapping mode.

Override: during tapping, the feedrate and spindle speed override are defaulted into 100%, and the machine does not stop during the feed hold key being pressed till the return operation is completed.

Tool radius compensation: in the fixed cycle command, the command function does not need executing the tool radius compensation, so, the tool radius compensation is ignored.

Program restart: It is invalid during the rigid taping.

Note： when the flexible tapping, rigid tapping or deep-hole rigid tapping is executed, using G97 cancels the constant surface cutting feedrate, otherwise, teeth disorder or broken screw taper exists.

4.5.3 Peck Rigid Taping (Chip Removal) Cycle

Command format：G84（or G74） X_Y_Z_R_P_Q_F_K_

Function: In peck rigid taping, cutting is performed several times until the bottom of the hole is reached.

X_Y_： Hole positioning data

Z_： In incremental programming it specifies the distance from point R level to the bottom of the hole; in absolute programming it specifies the absolute coordinates of the hole bottom.

R_： In incremental programming it specifies the distance from the initial level to point R level; in absolute programming it specifies the absolute coordinates of point R.

P_： Minimum dwell time at the bottom of the hole or at point R when a return is made. Its absolute value is used if it is negative.

Q_： Cut depth for each cutting feed F_：

Cutting feedrate.

V_： Retraction distance d. when it is not specified, it is set by P284; K_：

Number of repeats （specify it if necessary）

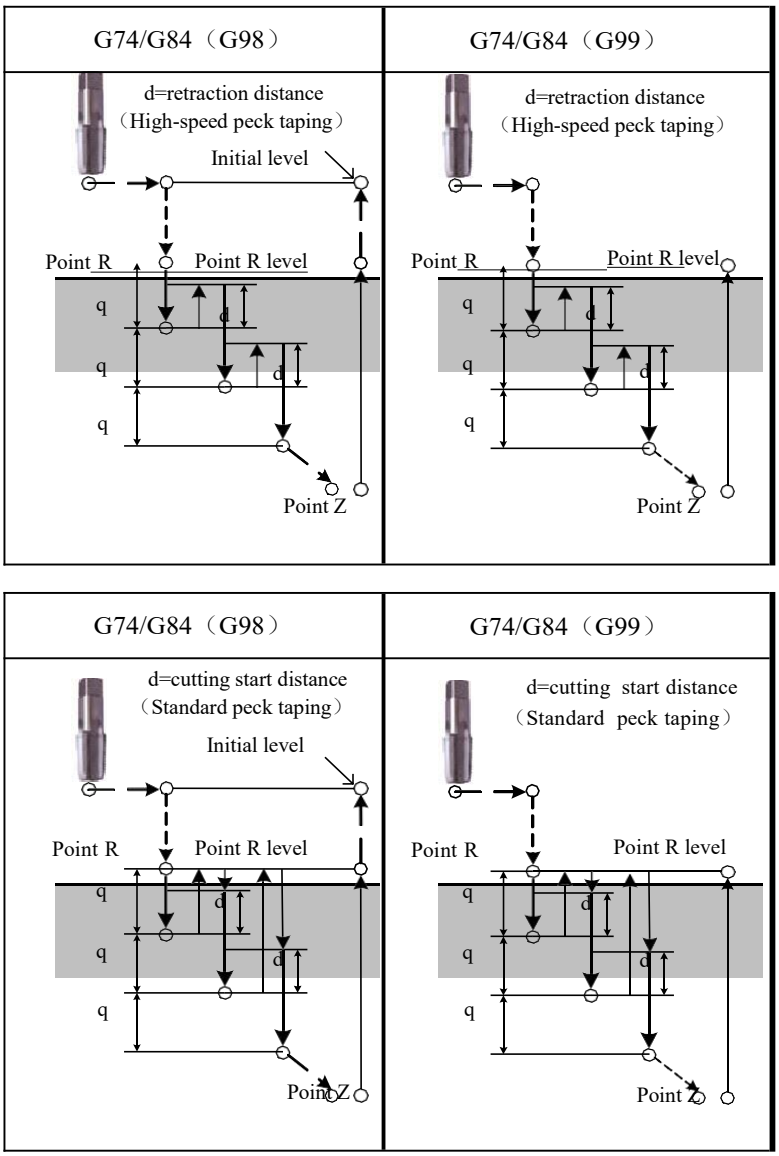


Fig. 3-5-3-1

Table 3-5-3-1

Deep-hole tapping cycle	Parameter setting	Used tapping mode
Deep-hole flexible tapping cycle	NO:46#1=0 and NO:K007#7=0	NO:44#5=1: high-speed deep-hole tapping cycle; NO:44#5=0: standard deep-hole tapping cycle.
Deep-hole rigid tapping cycle	NO:46#1=1 and NO:K007#7=1	NO:44#5=1: high-speed deep-hold tapping cycle; NO:44#5=0: standard deep-hole tapping cycle.

There are two types of peck rigid tapping cycles: high-speed peck tapping cycle and standard peck tapping cycle, both of which are set by bit parameter NO: 46#1.

Deep-hole flexible tapping cycle:

When NO:46#1=0 and NO:K007#7=0, it is a deep-hole flexble tapping cycle, which is divided into high-speed deep-hole tapping cycle and standard deep-hole tapping cycle set by NO:44#5.

High-speed deep-hole tapping cycle:

When NO:44#5=1， it is a high-speed deep-hole tapping cycle: the tool moves along X axis and Y axis to position, and executes rapid feed to point R, perform cutting from point R to the tool infeed depth Q (depth every cutting feed), then, the tool retracts the distance d (it is specified by the fixed cycle parameter V and set by P284 without being specified). No:44#4 sets whether the override is valid when the rigid tapping retraction is done, No: 45#3 specifies the retraction speed override, No:45#2 sets whether to use the same time constant when the rigid tapping tool infeed/retraction is performed, No:45#4 sets whether the federate override selection signal and override cancellation signals are valid during the rigid tapping. When the tool reaches point Z, the spindle stops and retreats reversely.

Standard deep-hole (flexible) tapping cycle:

When NO:44#5=1，it is a standard deep-hole tapping cycle: the tool moves along X axis and Y axis to position, and executes rapid feed to point R, perform cutting from point R to the tool infeed depth Q (depth every cutting feed), then, the tool returns to point R. No:44#4 sets whether the override is valid when the rigid tapping retraction is done, No: 45#3 specifies the retraction speed override, and performs cutting again with the cutting speed F from point to the end point distance d which is far away from the last cutting (set by P284), No:45#2 sets whether to use the same time constant when the rigid tapping tool infeed/retraction is performed, When the tool reaches point Z, the spindle stops and retreats reversely.

Standard deep-hole(rigid) tapping cycle:

In position mode（NO:46.1 is set to 1，K parameter NO:7.7 is set to 1），specify M29 S***** to be a deep-hole rigid tapping cycle before tapping code, use a standard deep-hole tapping cycle mode, and its setting method is the same that of the flexible standard deep-hole tapping.

Tool length compensation: If the tool length compensation instruction G43, G44 or G49 is specified in the same block with the canned cycle instruction, the offset is added or cancelled at the time of positioning to point R level; If the tool compensation instruction G43, G44 or G49 is specified in a separate block in the canned cycle mode, the system can add or cancel the offset in real time.

Limitation:

G code: when G74/G84 is used, G codes in 01 group (G00 to G03, G60 modal code (NO:48#0 is set to 1), and G84（or G74）cannot be specified in the same block, otherwise G84（or G74） is replaced by other codes in group 1.

M codes: before G74/G84 is specified, using the miscellaneous function M code makes the spindle rotate. When the spindle rotation is not specified, the system automatically count the current spindle command speed on the R plane, and then the spindle is regulated to clockwise rotation(74)/counterclockwise (G84).

when G74/G84 and an M code are specified in the same block, the M code is executed while the 1st hole positioning operation, then the system proceeds to the next drilling operation.

If number of repeats K is specified, the M code is only executed for the 1st hole.

Note: In the current version, M00, M01, M02, M06, M30, M98 and M99 are the M codes executed after the other codes in a block, i.e. these M codes are executed after the execution of the current statement block.

S instruction: when the commanded spindle speed exceeds the max. spindle speed during tapping (P257: the spindle upper speed in the course of tapping cycle), an alarm occurs; the

gear of the max. spindle speed during the rigid tapping is determined by P294~P296.

F instruction: when the specified F value exceeds the cutting feedrate's upper value (P96 sets the upper value), the system takes the upper value as the reference.

P instruction: P is a modal code, the least value is set by P281, the max. value is set by P282. P value is less than the least value, and the system runs with the least value; when it is more than the max. value, the system run with the max. value.

Axis switch: must cancel the fixed cycle before switching the tapping axis. No. 206 alarm occurs when the tapping axis is changed in the rigid tapping mode.

Override: during tapping, the feedrate and spindle speed override are defaulted into 100%, and the machine does not stop during the feed hold key being pressed till the return operation is completed.

Tool radius compensation: in the fixed cycle command, the command function does not need executing the tool radius compensation, so, the tool radius compensation is ignored.

Note: when the flexible tapping, rigid tapping or deep-hole rigid tapping is executed, using G97 cancels the constant surface cutting feedrate, otherwise, teeth disorder or broken screw taper exists.

4.6 Compound Cycle G Code

Comparative table of compound cycle (G22～G38)

Table 3-6-1

G code	Drilling and cutting (-Z direction)	Hole bottom operation	Tool retraction operation (+Z direction)	Use
G22	Cutting feed		Rapid feed	Inner circular groove rough milling (CCW)
G23	Cutting feed		Rapid feed	Inner circular groove rough milling (CW)
G24	Cutting feed		Rapid feed	Fine milling cycle within a full circle(CCW)
G25	Cutting feed		Rapid feed	Fine milling cycle within a full circle(CW)
G26	Cutting feed		Rapid feed	Outer circle finish milling cycle (CCW)
G32	Cutting feed		Rapid feed	Outer circle finish milling cycle (CW)
G33	Cutting feed		Rapid feed	Rectangular groove rough milling(CCW)
G34	Cutting feed		Rapid feed	Rectangular groove rough milling(CW)
G35	Cutting feed		Rapid feed	Inner rectangular groove fine milling cycle(CCW)
G36	Cutting feed		Rapid feed	Inner rectangular groove fine milling cycle(CW)
G37	Cutting feed		Rapid feed	Rectangle outside fine milling cycle(CCW)
G38	Cutting feed		Rapid feed	Rectangle outside fine milling cycle(CW)

Limitation:

During the compound cycle positioning, the tool radius offset (D) will be ignored.

4.6.1 Inner circular groove rough milling G22/G23

Command format:

G22

G98/G99 **X_ Y_ Z_ R_ I_ L_ W_ Q_ V_ D_ F_ K_**

G23

Function: it is used for performing circular interpolations from the circle center by helical line till the programmed figure of the circle groove is machined.

Explanation:

G22: CCW inner circular groove rough milling G23:

CW inner circular groove rough milling X, Y：The

start point in X, Y plane；

Z：Machining depth, which is the absolute position in G90, and the position relative to R level in G91；

R：R reference level, which is the absolute position in G90, and the position relative to the start point of this block in G91;

I：Circular groove radius, which should be greater than the current tool radius；

L：Cut width increment within XY plane, which is less than the tool diameter but more than 0; W：First cutting depth in Z axis direction. It is the distance below the R level, which should be greater than 0 (if the first cutting depth exceeds the groove bottom, then the machining is performed at the groove bottom)；

Q：Cutting depth for each cutting feed；

V：Distance (greater than 0) to the end surface to be machined at rapid tool traverse;

D：Tool compensation number, ranging from 1～256. D0 is 0 by default. The current tool diameter value is obtained by the specified sequence number;

K：Number of repeats.

Cycle process:

(1) Rapid positioning to the position in XY plane;
(2) Rapid down to point R level;
(3) Cut a depth W downward at the cutting speed by helical mode→feed to the circle center;
(4) Mill the circle surface with a radius of I helically outward from the center by an increment of L each time;
(5) Return to R reference level along Z axis;
(6) Axes X and Y rapidly position to the start point;
(7) Down to the position at which the distance to the end surface to be machined is V along Z axis;
(8) Cut a depth (Q+V) downward along Z axis;
(9) Repeat the operations (4)~(8) till the total depth of circle surface is finished;
(10) Return to initial level or point R level depending on G98 or G99.

Command path:

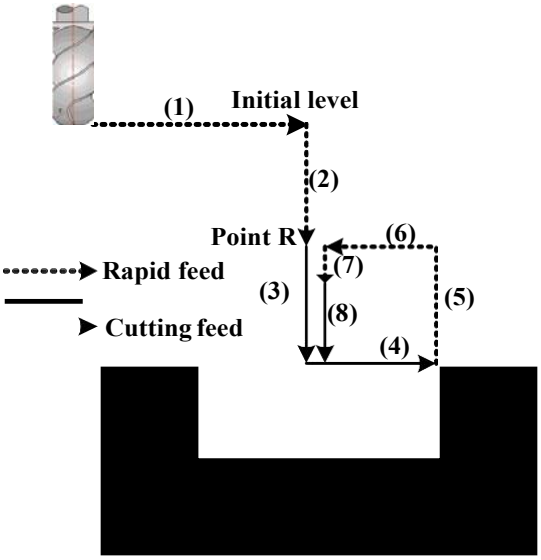


Fig. 3-6-1-1

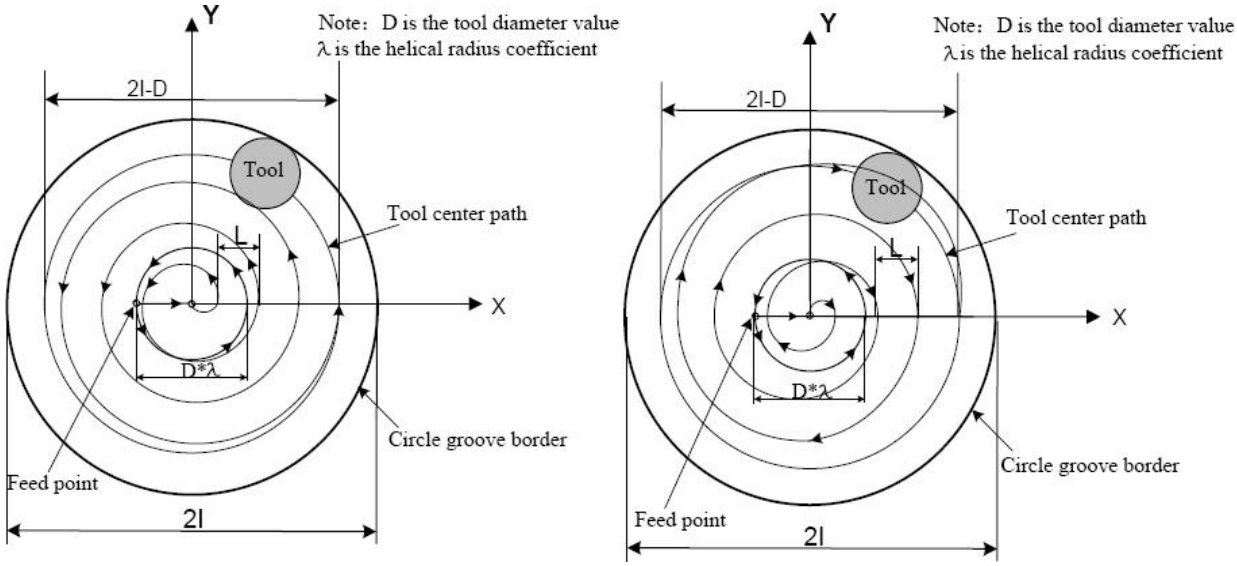


Fig. 3-6-1-2

Note:

1. It is suggested that the NO：12#1 be set to 1 when this code is used.
2. The helical radius coefficient in the groove cycle must be greater than 0. The coefficient is set by data parameter P269.

Example: Rough milling an inner circle groove using the canned cycle code G22, as shown in the figure below:

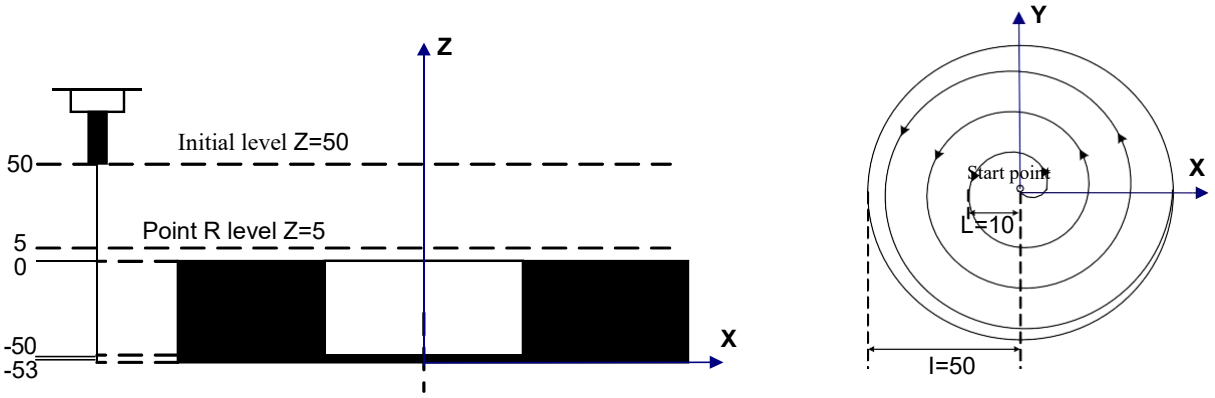


Fig. 3-6-1-3

G90 G00 X50 Y50 Z50; （G00 Rapid positioning）

G99 G22 X25 Y25 Z-50 R5 I50 L10 W20 Q10 V10 D1 F800;（Groove rough milling within a circle）

G80 X50 Y50 Z50; (Canned cycle cancel and return from R level) M30;

Limitation: when G22/G23 is used, G codes in 01 group (G00 to G03, G60 modal code (NO:48#0 is set to 1), otherwise G22/G23 is replaced by other codes in group 1.

Tool radius compensation: in the fixed cycle command, the tool radius compensation is ignored, the system calls the tool radius compensation specified by the program during the tool infeed.

4.6.2 Fine Milling Cycle within a Full Circle G24/G25

Command format:

G98/G99 G24/G25 X_ Y_ Z_ R_ I_ J_ D_ F_ K_

Function: The tool fine mills a full circle within a circle by the specified radius I and the specified direction, and it returns after finishing the fine milling.

Explanation:

G24: CCW fine milling inside a circle

G25: CW fine milling inside a circle

X,Y: The start point position within X, Y plane

Z: Machining depth, which is absolute position in G90 and position relative to R reference level in G91

R: R reference level which is the absolute position in G90 and the position relative to start point of this block in G91

I: Fine milling circle radius, ranging from 0.0001mm~99999.9999mm. Its absolute value is used if it is negative;

J: Distance from fine milling start point to circle center, ranging from 0~99999.9999mm. Its absolute value is used if it is negative;

D: Tool diameter number, ranging from 1~256. D0 is 0 by default. The tool diameter value is obtained by the given number.

K: Number of repeats

Cycle process:

(1) Rapid positioning to a location within XY plane;

(2) Rapid down to point R level;

(3)Feed to the machining start point at hole bottom;

(4)To make circular interpolation by the transition arc 1 from the start point;

(5)To make circular interpolation for the whole circle by inner arc path of finish-milling.
(6)To make circular interpolation by transition arc 4 and return to the start point;
(7)Return to the initial level or R level according to code G98 or G99.

Command path:

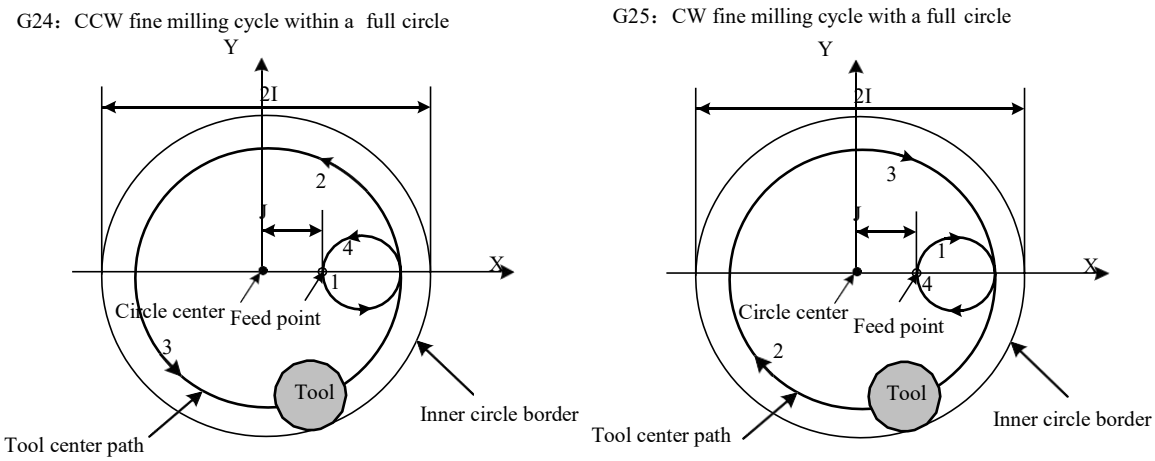


Fig. 3-6-2-1

Note：The NO: 12#1 should be set to 1 when this code is used.

Example: Fine milling a circular groove that has been rough milled as follows by canned cycle code G24:

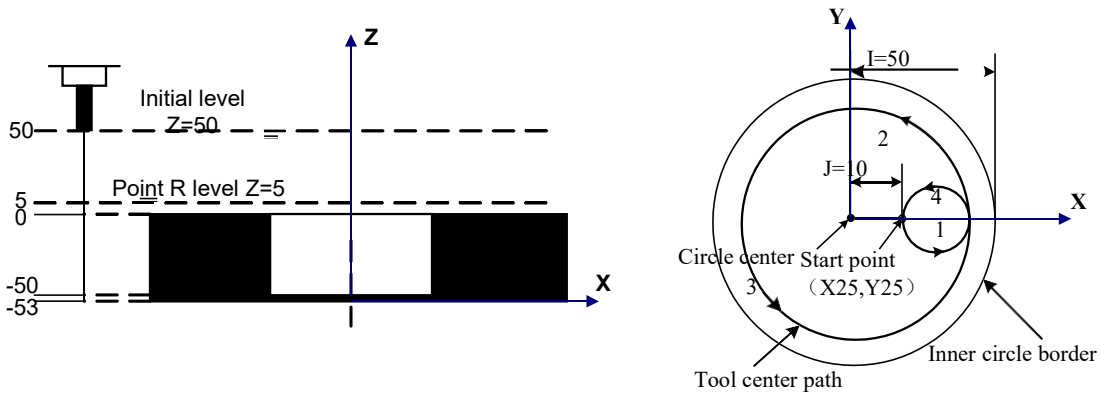


Fig. 3-6-2-2

G90 G00 X50 Y50 Z50； (G00 rapid positioning)
G99 G24 X25 Y25 Z-50 R5 I50 J10 D1 F800； (Canned cycle starts, and goes down to the bottom to perform the inner circle finish milling) G80
X50 Y50 Z50； (To cancel canned cycle and return from R level)
M30；

Limitation: when G24/G25 is used, G codes in 01 group (G00 to G03, G60 modal code (NO:48#0 is set to 1), otherwise G24/G25 is replaced by other codes in group 1.

Tool radius compensation: in the fixed cycle command, the tool radius compensation is ignored, the system calls the tool radius compensation specified by the program during the tool infeed.

4.6.3 Outer Circle Finish Milling Cycle G26/G32

Command format:

G98/G99 G26/G32 X_ Y_ Z_ R_ I_ J_ D_ F_ K_;

Explanation:

G26: CCW outer circle fine milling cycle

G32: CW outer circle fine milling cycle X,Y:

The start point within X, Y plane

Z: Machining depth, which is absolute position in G90 and position relative to R reference level in G91;

R: R reference level, which is absolute position in G90 and position relative to the start point of this block in G91;

I: Fine milling circle radius, ranging from 0.0001mm~99999.9999mm mm. Its absolute value is used if it is a negative one;

J: Distance from the milling start point to the milling circle center, ranging from 0.0001mm~99999.9999mm. Its absolute value is used if it is a negative one;

D: Tool radius number, ranging from 0 ~256, D0 is defaulted for 0. The current tool radius value is obtained by the given number;

K: Number of repeats.

Cycle process:

(1) Rapid positioning to a location within XY plane;
(2) Rapid down to R level;
(3) Feed to the hole bottom;
(4)To make circular interpolation by the transition arc 1 from the start point;
(5)To make circular interpolation for the whole circle by the path of arc2 and arc 3;
(6)To make circular interpolation by transition arc 4 and return to the start point;
(7) Return to the initial level or R level according to code G98 or G99.

Command path:

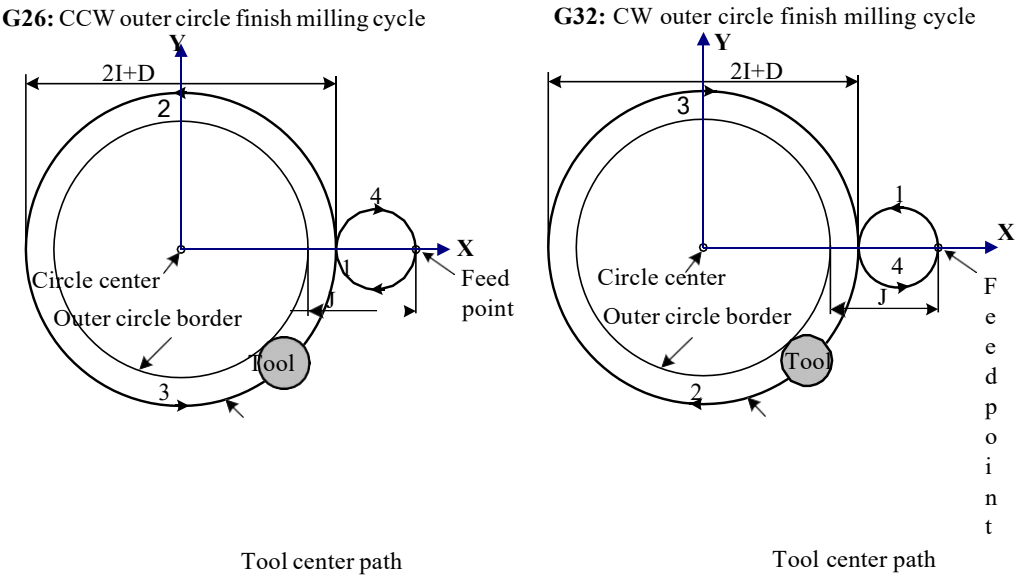


Fig. 3-6-3-1

Explanation:

In outer circle finish milling, the interpolation directions of the transition arc and fine milling arc are different. The interpolation direction in the code means the one of the fine milling.

Example: Fine milling a circular groove that has been rough milled as follows by the canned cycle code G26:

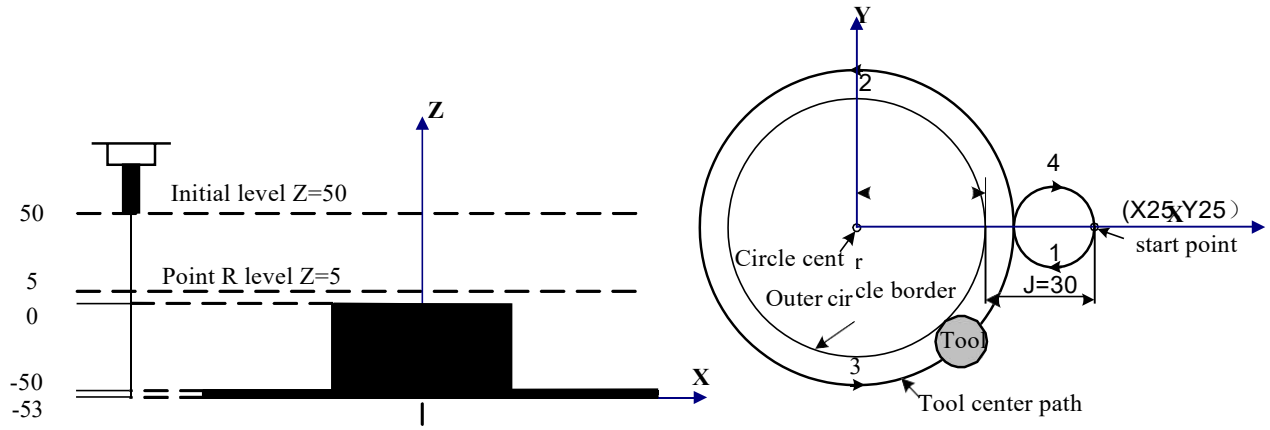


Fig. 3-6-3-2

G90 G00 X50 Y50 Z50; （G00 rapid positioning)

G99 G26 X25 Y25 Z-50 R5 I50 J30 D1 F800; (Canned cycle starts, and goes down to the bottom to perform the outer circle fine milling)

G80 X50 Y50 Z50; （To cancel canned cycle and return from R level）

M30;

Limitation: when G26/G32 is used, G codes in 01 group (G00 to G03, G60 modal code (NO:48#0 is set to 1), otherwise G26/G32 is replaced by other codes in group 1.

Tool radius compensation: in the fixed cycle command, the tool radius compensation is ignored, the system calls the tool radius compensation specified by the program during the tool infeed.

4.6.4 Rectangular Groove Rough Milling G33/G34

Command format:

G33

G98/G99 X_ Y_ Z_ R_ I_ J_ L_ W_ Q_ V_ U_ D_ F_ K_

G34

Function: These codes are used for linear cutting cycle by the specified parameter data from the rectangle center till the programmed rectangular groove is machined.

Explanation:

G33: CCW rectangular groove rough milling

G34: CW rectangular groove rough milling X, Y:

The start point within X, Y plane

Z: Machining depth, which is absolute position in G90 and position relative to R reference plane in G91

R: R reference plane, which is absolute position in G90 and position relative to the start point of this block in G91

I: Rectangular groove width in X axis, which should be greater than { (The setting value of data parameter **P269** * tool radius)＋ tool radius} * 2, and the helical feed radius should be smaller than {(I/2) - tool radius}.

J: Rectangular groove width in Y axis, which should be greater than { (The setting value of data parameter **P269** * tool radius)＋ tool radius} * 2, and helical feed radius should be smaller than {(J/2) - tool radius} .

L: Cutting width increment within a specified plane, which should be less than the tool diameter but greater than 0. Its absolute value is used if it is a negative one.

W：First cut depth in Z axis, which is a downward distance from R level and is greater than 0 (if the first cut exceeds the groove bottom, it will cut at the bottom position). Its absolute value is used if it is a negative one.

Q：Cut depth of each cutting feed

V：Distance to the end surface to be machined in rapid feed, which is greater than 0. Its absolute value is used if it is negative.

U：Corner arc radius. No corner arc transition if it is omitted. The range of U is |U|, which is greater than or equal to D/2, and smaller than I/2 or J/2 whichever is smaller.

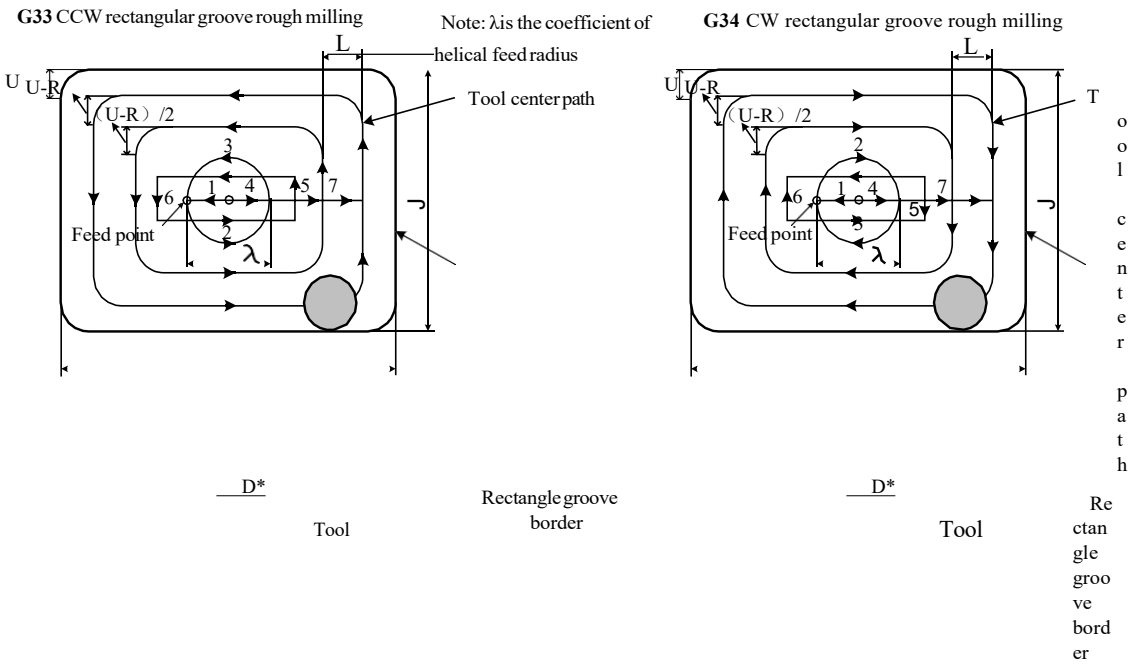
D：Tool diameter number, ranging from 1 ~ 256, D0 is 0 by default. The current tool diameter value is given by the specified number.

K：Number of repeats.

Cycle process

(1) Rapid positioning to the start point of helical feed within XY plane;
(2) Rapid down to R level;
(3) The diameter helical feed W width is obtained by radius compensation value multiplying the parameter N0. 269 value;
(4) Feed to the rectangle center;
(5) To mill a rectangular surface helically by an increment L from center outward each time;
(6) Rapid return to R level along Z axis;
(7) Rapid positioning to star point of the helical feed in XY plane;
(8) Rapid down to a position at which the distance to the end surface is V along Z axis;
(9) Z axis cuts downward for a（Q+V）depth;
(10) Repeat the actions of（4）~（8）till the rectangular surface with the total depth machined;
(11) Return to the initial level or R level according to code G98 or G99.

Command path：



I I

Fig. 3-6-4-1

Note: The NO:12#1 should be set to 1 when this code is used.

Example: Rough milling an inner rectangular groove by the canned cycle code G33, as shown in the following figure:

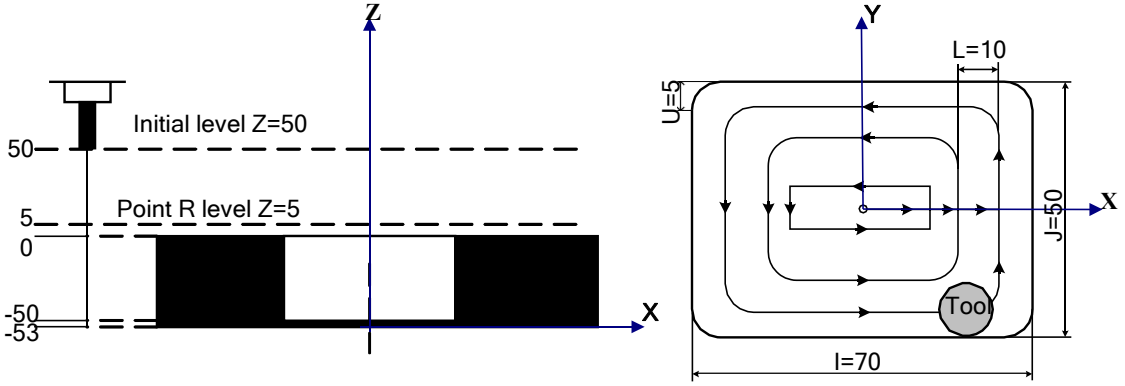


Fig. 3-6-4-2

G90 G00 X50 Y50 Z50; （G00 rapid positioning）

G99 G33 X25 Y25 Z-50 R5 I70 J50 L10 W20 Q10 V10 U5 D1 F800;

（To perform inner rectangular grove rough milling cycle）

G80 X50 Y50 Z50; (To cancel canned cycle and return from R level)

M30;

Limitation: when G23/G34 is used, G codes in 01 group (G00 to G03, G60 modal code (NO:48#0 is set to 1), otherwise G33/G34 is replaced by other codes in group 1.

Tool radius compensation: in the fixed cycle command, the tool radius compensation is ignored, the system calls the tool radius compensation specified by the program during the tool infeed.

4.6.5 Inner Rectangular Groove Fine Milling Cycle G35/G36

Command format:

G35

G98/G99 **X_ Y_ Z_ R_ I_ J_ L_ U_ D_ F_ K_;**

G36

Function: They are used for fine milling within a rectangle by the specified width and direction, and the tool returns after finishing the fine milling.

Explanation:

G35: CCW inner rectangular groove finish milling cycle. G36: CW inner rectangular groove finish milling cycle. X,Y: The start point within X, Y plane;

Z: Machining depth, which is absolute position in G90 and position relative to R reference plane in G91;

R: R reference plane, which is absolute position in G90 and position relative to the start point of this block in G91;

I: Rectangular width in X axis, ranging from tool diameter~99999.9999mm. Its absolute value is used if it is negative;

J: Rectangular width in Y axis, ranging from tool diameter~99999.9999mm. Its absolute value is used if it

is negative;

L： Distance from milling start point to rectangular side in X axis, ranging from tool radius~99999.9999mm. Its absolute value is used if it is negative;

U：Corner arc radius. No corner transition if it is omitted. Alarm is issued if 0＜U＜tool radius;

D：Tool diameter number, ranging from 1 ~ 256, D0 is 0 by default. The current tool diameter value is given by the specified number;

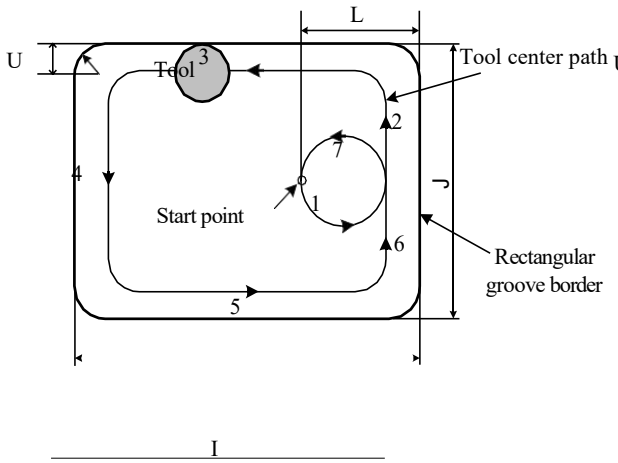
K：Number of repeats.

Cycle process:

(1) Rapid positioning to the start point within XY plane;
(2) Rapid down to R level;
(3) Feed to the hole bottom;
(4) Perform circular interpolation by the path of transition arc 1 from the start point;
(5) Perform linear and circular interpolation by the path 2-3-4-5-6;
(6) perform circular interpolation by the path of transition arc 7 and return to the start point;
(7) Return to the initial level or R level according to G98 or G99.

Command path：

G35: CCW rectangular groove finish milling cycle



G36: CW rectangular groove finish milling cycle

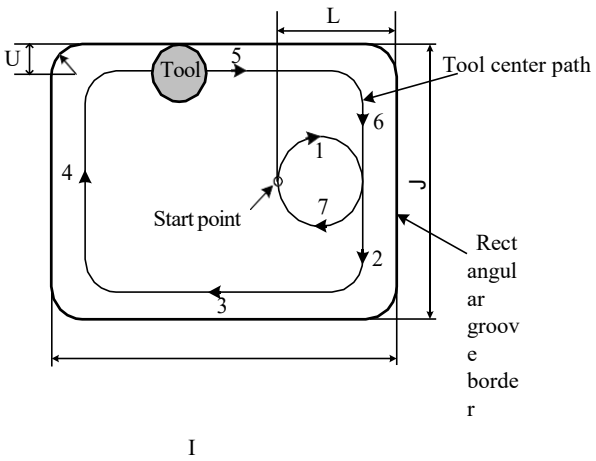
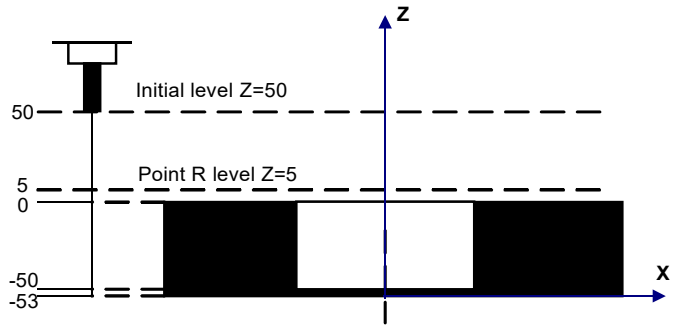


Fig.3-6-5-1

Note: The NO:12#1 should be set to 1 when this code is used.

Example: Fine milling a circular groove that has been rough milled in the figure below by canned cycle G35 code:



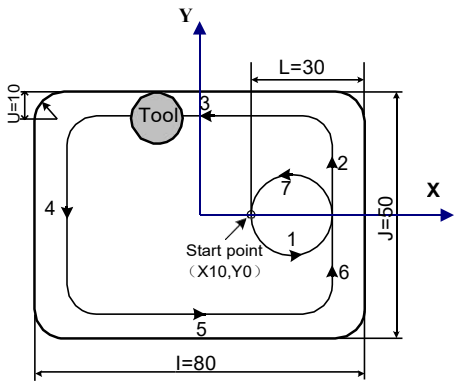


Fig. 3-6-5-2

G90 G00 X50 Y50 Z50； （G00 rapid positioning）

G99 G35 X10 Y0 Z-50 R5 I80 J50 L30 U10 D1 F800；（Performing inner rectangular groove milling at hole bottom in the canned cycle）

G80 X50 Y50 Z50； (Cancelling the canned cycle, and returning from point R level) M30；

Limitation: when G35/G36 is used, G codes in 01 group (G00 to G03, G60 modal code (NO:48#0 is set to 1), otherwise G35/G36 is replaced by other codes in group 1.

Tool radius compensation: in the fixed cycle command, the tool radius compensation is ignored, the system calls the tool radius compensation specified by the program during the tool infeed.

4.6.6 Rectangle Outside Fine Milling Cycle G37/G38

Command format:

G98/G99 G37/G38 X_ Y_ Z_ R_ I_ J_ L_ U_ D_ F_ K_

Function: The tool performs fine milling outside the rectangle by the specified width and direction, and then returns after finishing the fine milling.

Explanation:

G37: CCW fine milling cycle outside a rectangle.

G38: CW fine milling cycle outside a rectangle. X,Y:

The start point within X, Y plane;

Z: Machining depth, which is absolute position in G90 and position relative to R reference plane in G91;

R: R reference plane, which is absolute position in G90 and position relative to the start point of this block in G91;

I: Rectangular width in X axis, ranging from 0 mm ~99999.9999mm. Its absolute value is used if it is negative;

J: Rectangular width in Y axis, ranging from 0 mm ~99999.9999mm. Its absolute value is used if it is negative;

L: Distance from the milling start point to rectangular side in X axis, ranging from 0 mm ~99999.9999mm. Its absolute value is used if it is negative;

U: Corner arc radius. There is no corner transition arc if it is omitted;

D: Tool diameter number, ranging from 1 ~ 256, D0 is 0 by default. The current tool diameter value is given by the specified number;

K: Number of repeats.

Cycle process:

(1) Rapid positioning to the start point within XY plane;
(2) Rapid down to R level;
(3) Feed to the hole bottom;
(4)Perform circular interpolation by the path of transition arc 1 from the start point;
(5)Perform linear and circular interpolation by the path 2-3-4-5-6
(6); Perform circular interpolation by the path of transition arc 7 and return to the start point;
(7) Return to the initial level or R level according to G98 or G99.

Command path:

G37 CCW fine milling cycle outside a rectangle G38 CW fine milling cycle outside a rectangle

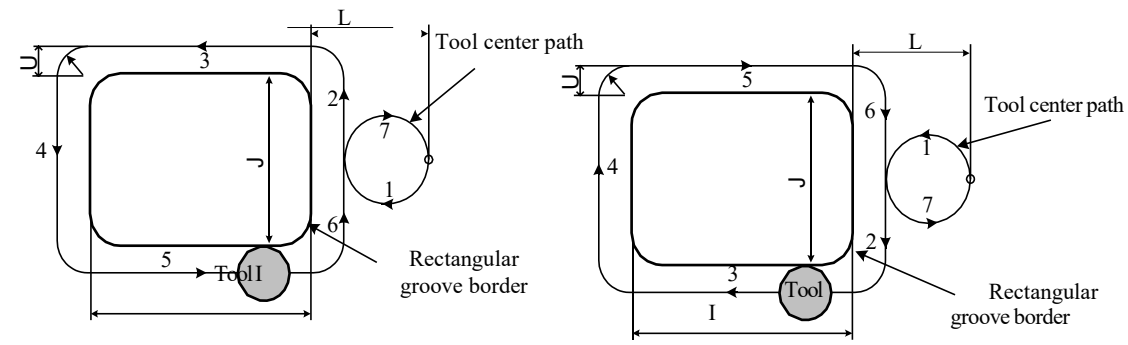


Fig. 3-6-6-1

Explanation:

For the rectangle outside fine milling, if the interpolation directions of the transition arc and fine milling arc are inconsistent, the interpolation direction in the code is the one of the fine milling arc.

Example: Performing fine milling outside a rectangle by the canned cycle code G37.

G90 G00 X50 Y50 Z50; （G00 rapid positioning）

G99 G37 X25 Y25 Z-50 R5 I80 J50 L30 U10 D1 F800;（Performing fine milling outside a rectangle at the hole bottom in the canned cycle）

G80 X50 Y50 Z50; (Cancelling the canned cycle, returning from the Point R level) M30;

Limitation: when G37/G38 is used, G codes in 01 group (G00 to G03, G60 modal code (NO:48#0 is set to 1), otherwise G37/G38 is replaced by other codes in group 1.

Tool radius compensation: in the fixed cycle command, the tool radius compensation is ignored, the system calls the tool radius compensation specified by the program during the tool infeed.

4.7 Tool Compensation G Code

4.7.1 Tool Length Compensation G43, G44, G49

Function:

G43 specifies the positive compensation for tool length.

G44 specifies the negative compensation for tool length. G49 is used to cancel tool length compensation.

Format:

There are 2 modes A/B for tool length offset which are set by bit parameter No：39#0 in this system.

Mode A：

G43
G44 }Z_ H_ ;

Mode B：

G17 G43 Z_H；

G17 G44 Z_H ；

G18 G43 Y_H ；

G18 G44 Y_H ；

G19 G43 X_H ；

G19 G44 X_H；

Tool length offset mode cancel: G49 or H0.

Explanation:

The above codes are used to shift an offset value for the end point of the specified axis. The difference between assumed tool length (usually the 1st tool) and actual tool length used is saved into the offset memory, tools of different length thus can be used to machine the workpiece only by changing the tool length offset values instead of the program.

G43 and G44 specify the different offset directions, and H code specifies the offset number.

1. Offset direction

G43 ： Positive offset (frequently-used)

G44： Negative offset

Either for absolute code or incremental code, when G43 is specified, the offset value (stored in offset memory) specified with the H code is added to the coordinates of the moving end point specified by an code in the program. When G44 is specified, the offset value specified by H code is subtracted from the coordinates of the end position, and the resulting value obtained is taken as the final coordinates of the end position.

G43， G44 are modal G codes, which are effective till another G code belonging to the same group is used.

2. Specification of offset value

The length offset number is specified by H code. The offset value assigned to the offset number is added to or subtracted from the moving code value of Z axis, which obtains the new code value of Z axis. H00～H255 can be specified as the offset number as required.

The range of the offset value is as follows:

Table 3-7-1-1

	Range
Offset value H（input in mm）	-999.999 mm～+999.999mm
Offset value H（input in inch）	-39.3700 inch～+39.3700 inch

The offset value assigned to offset number 00 (H00) is 0, which cannot be set in the system.

Note: When the offset value is changed due to the change of the offset number, the new offset value replaces the old one directly rather than being added to the old compensation value. For example:

H01..... Offset value 20

H02..... Offset value 30

G90 G43 Z100 H01 ； .....Z moves to 120

G90 G43 Z100 H02 ； .....Z moves to 130

3. Sequence of the offset number

Once the length offset mode is set up, the current offset number takes effect at once; if the

offset number is changed, the old offset value will be immediately replaced by the new one. For example:

O×××××;
H01;
G43 Z10; (1) Offset number H01 takes effect G44 Z20
H02; (2) Offset number H02 takes effect H03; (3) Offset number H03 takes effect
G49; (4) Offset is cancelled at the end of the block
M30;

4. Tool length compensation cancel

Specify G49 or H00 to cancel tool length compensation. The tool length compensation is cancelled immediately after they are specified.

Note: 1. After B mode of tool length offset is executed along two or more axes, all the axis offsets are canceled by specifying G49, however, only the axis offset perpendicular to a specified plane is canceled by specifying H00.

2. It is suggested that a moving code of Z axis be added for the set-up and cancel of the tool length offset, otherwise, the length offset will be set up or canceled at the current point. Therefore, please ensure a safe height in the Z axis when using G49 to prevent tool collision and workpiece damage.

5. Example for tool length compensation

(A) Tool length compensation (boring hole # 1, #2, #3)

(B) H01= offset value – 4

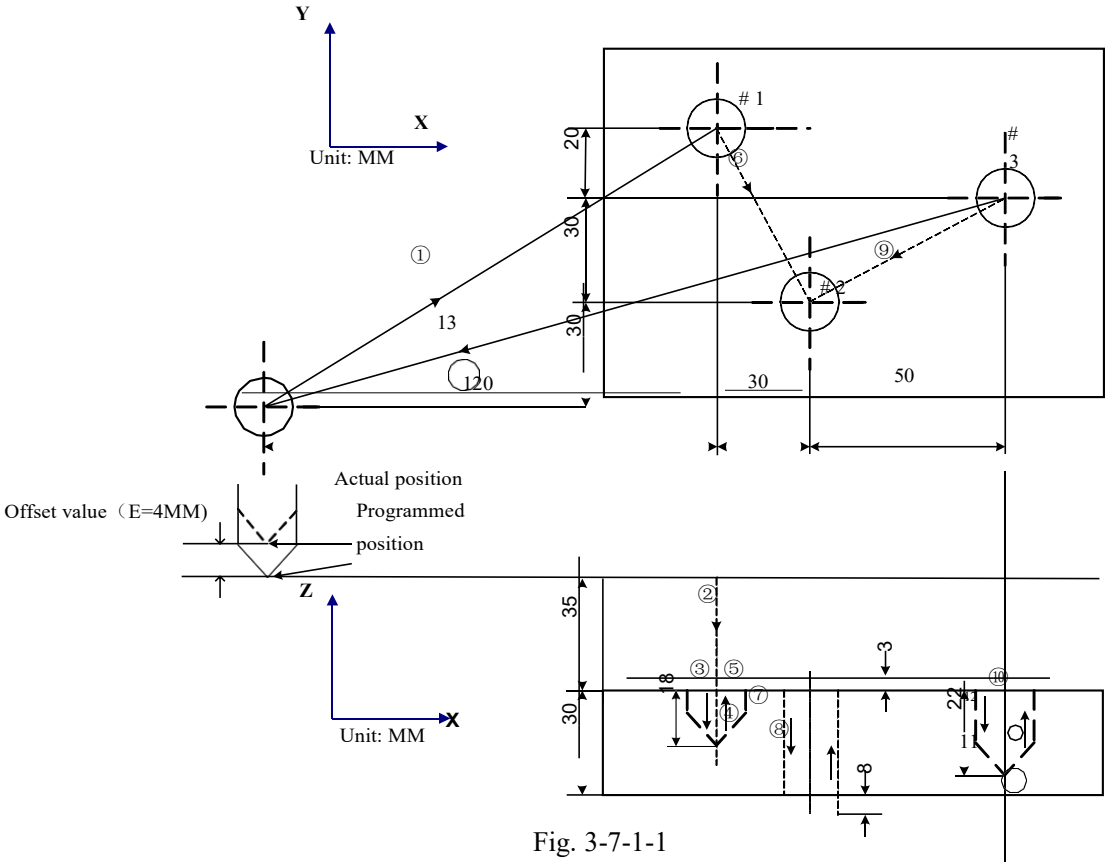


Fig. 3-7-1-1

N1 G91 G00 X120 Y80 ; ..... (1)

N2 G43 Z-32 H01 ； ..... (2)
N3 G01 Z-21 F200 ； .....(3)
N4 G04 P2000 ； ..... (4)
N5 G00 Z21 ； .....(5)
N6 X30 Y-50 ； .....(6)
N7 G01 Z-41 F200 ； .....(7)
N8 G00 Z41 ； .....(8)
N9 X50 Y30 ； .....(9)
N10 G01 Z-25 F100 ； .....(10)
N11 G04 P2000 ； .....(11)
N12 G00 Z57 H00 ； .....(12)
N13 X-200 Y-60 ； .....(13)
N14 M30；

4.7.2 Tool radius compensation G40/G41/G42

Command format:

G41 D_ X_Y_ ;
G42 D_ X_Y_ ;
G40 X_Y_;

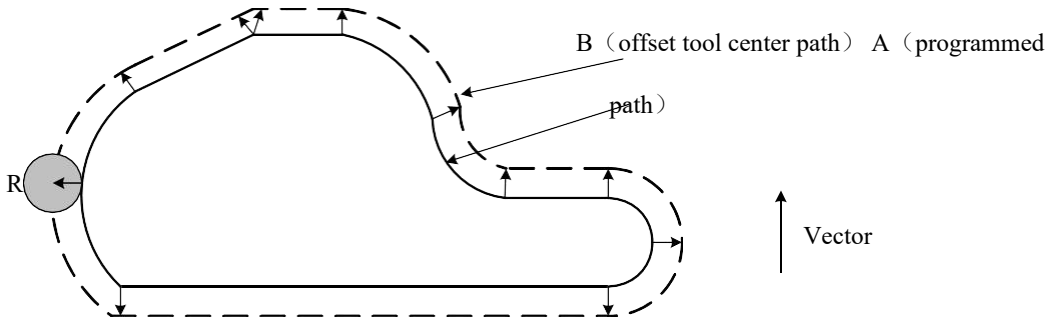
Function:

G41 specifies the left compensation of the tool moving. G42 specifies the right compensation of the tool moving. G40 cancels the tool radius compensation.

Explanation:

1. Tool radius compensation

As the following figure, when using a tool with radius R to cut workpiece A, the tool center path is shown as B, and the distance from path B to path A is R. That the tool is moved by tool radius apart from the workpiece A is called compensation.



Compensation and vector

Fig. 3-7-2-1

Programmers write machining programs using the tool radius compensation mode. During the machining, the tool diameter is measured and input into the CNC memory, then the tool path turns into offset path B.

2. Offset value (D value)

The radius offset number is specified by D code. The offset value corresponding to the offset number is

added to or subtracted from the moving code value in the program, thus obtains the new moving code value. The offset number can be specified by D00～D255 as required. Whether the radius offset value is set by parameter value or radius value is selected by bit parameter **N0：40#7.**

The offset value assigned to the offset number can be saved into the offset memory in advance using LCD/MDI panel.

The range of the offset value is as follows:

Table 3-7-2-1

	Range
Offset value D (input in mm)	-999.999mm～+999.999mm
Offset value D (input in inch)	-39.3700 inch～+39.3700 inch

Note: The default offset value of D00 is 0 that cannot be set or modified by the user.
The change of the offset plane can only be performed after the offset mode is cancelled. If the offset plane is changed without cancelling the offset mode, an alarm will be issued.

3. Plane selection and vector

Compensation calculation is carried out in the plane selected by G17，G18 or G19. This plane is called the offset plane. For example, if XY plane is selected, the compensation and vector calculation are carried out by (X，Y) in the program. The coordinates of the axes not in the offset plane are not affected by compensation.

In simultaneous 3-axis control, only the tool path projected on the offset plane is compensated. The change of the offset plane can only be performed after the compensation is cancelled.

Table 3-7-2-2

G code	Offset plane
G17	X – Y plane
G18	Z - X plane
G19	Y – Z plane

4. G40, G41, G42

The cancellation and execution of the tool radius compensation vector are specified by G40, G41, G42. They are used in combination with G00, G01, G02, G03 to define a mode to determine the value and the direction of the offset vector.

Table 3-7-2-3

G code	Function
G40	Tool radius compensation cancel
G41	Tool radius compensation left
G42	Tool radius compensation right

5. G53, G28 or G30 code in tool radius compensation mode

If G53, G28, or G30 code is specified in tool radius compensation, the offset vector of tool radius offset axis is cancelled after the specified position is reached. (cancelled at the specified

position in G53, cancelled at the reference point in G28,G30), and the other axes except tool radius offset axes are not cancelled. When G53 is in the same block with G41/G42, all the axes cancel their radius compensation when the specified position is reached; when G28 or G30 is in the same block with G41/G42, all the axes cancel their radius compensation after the reference point is reached. The cancelled tool radius compensation vector will be restored in the next buffered block containing a compensation plane.

Note: In offset mode, whether the compensation is temporarily cancelled when G28 or G30 moves to the intermittent point is decided by bit parameter No: 40#2.

Tool radius compensation cancel (G40)

In G00, G01 mode, using the following code G40 X Y ;

Perform the linear motion from the old vector of the start point to the end point:. In G00 mode, rapid traverse is performed to the end point along each axis. By using this code, the system switches from tool compensation mode to tool compensation cancel mode. If G40 is specified without X_ Y_, no operation is performed by the tool.

When it is G40, and X Y_does not exist，the tool does not move.

Tool radius compensation left (G41) 1）

G00, G01

G41 X Y D ; It forms a new vector perpendicular to the direction of (X, Y) at the block end point. The tool is moved from the tip of the old vector to the tip of the new vector at the start point.

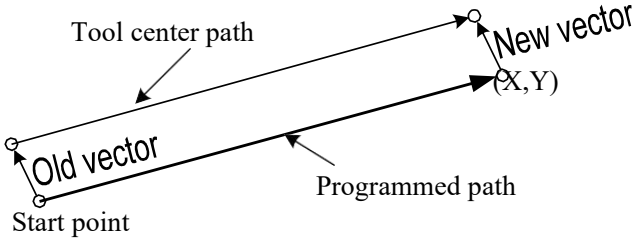


Fig. 3-7-2-2

When the old vector is zero, the tool is switched to tool radius compensation mode from tool offset cancel mode using this code. Here, the offset value is specified by D code.

2） G02, G03

G41……;

……
……

G02 /G03 X Y R ;

According to the program above, the new vector that is located on the line between the circle center and the end point can be created. Viewed from the arc advancing direction, it points to the left (or right). The tool center moves along an arc from the old vector tip to the new vector tip on the precondition that the old vector has been created correctly.

The offset vector points towards or is apart from the arc center from the start point or the end point.

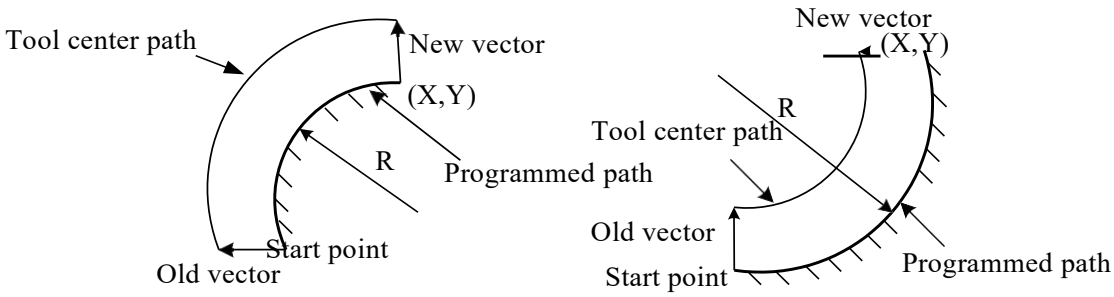


Fig. 3-7-2-3

Tool radius compensation right (G42)

In contrast with G41, G42 specifies the tool to deviate at the right side of the workpiece along the tool advancing direction, i.e. the vector direction obtained in G42 is reverse to the vector direction obtained in G41. Except for the direction, the deviation of G42 is identical with that of G41.

1） G00, G01

G42 X Y D ;

G42 X Y ;

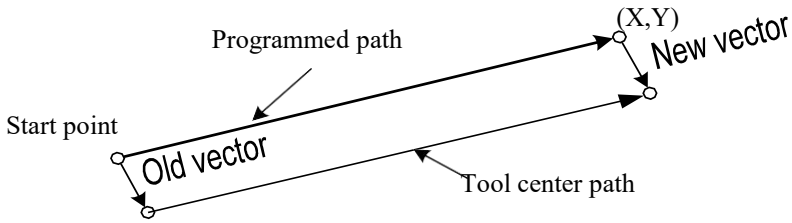


Fig. 3-7-2-4

2） G02, G03

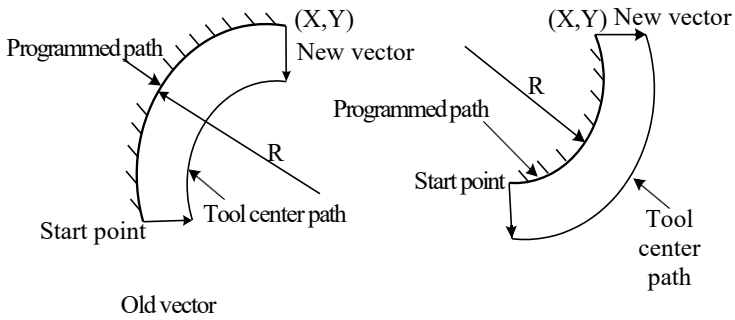


Fig. 3-7-2-5

6. Precautions on offset

(A) Offset number specification

G41， G42 and G40 are modal codes. The offset number can be specified by D code anywhere before the offset cancel mode is switched to the tool radius compensation mode.

(B) Switching from the offset cancel mode to tool radius compensation mode

The moving code must be positioning (G00) or linear interpolation (G01) when the mode is switched from the offset cancel mode to tool radius compensation mode. The circular interpolation (G02, G03) is not permitted.

(C) Switching between tool radius compensation left and tool radius compensation right

In general, the offset direction is changed from the left to the right or vice versus via offset cancel mode, but the direction in positioning (G00) or linear interpolation (G01) can be changed directly regardless of the offset cancel mode, and the tool path is as follows:

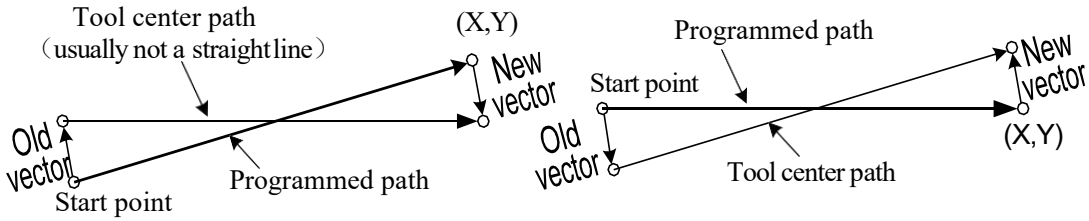


Fig. 3-7-2-6

G1G41 D__X__ Y__;	G42 D_X Y_;
……	……
G1G42 D__X__ Y__;	G41 D X Y_;

(D) Change of offset value

In general, the tool offset value is changed in the offset cancel mode when the tool is changed, but for positioning (G00) and linear interpolation, the value can also be changed in the offset mode. It is shown below:

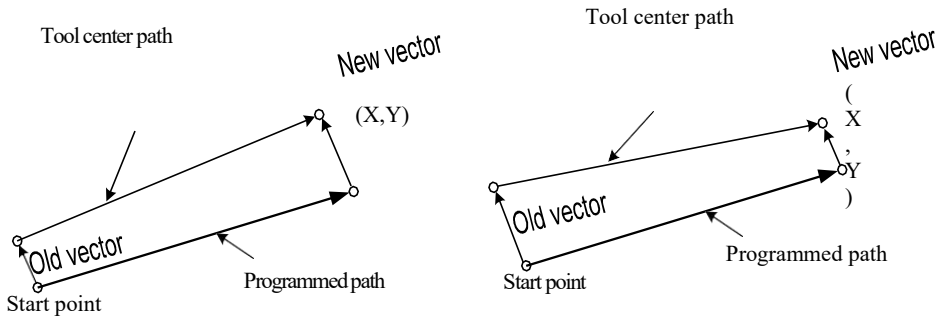


Fig. 3-7-2-7(Change of offset value)

(E) Positive and negative offset value and the tool center path

If the offset value is negative, the workpiece is machined in the same way as G41 and G42 are replaced with each other in the program. Therefore, the outer cutting for workpiece turns into inner cutting, and the inner cutting turns into outer cutting.

As the usual programming shown in the following figure, the offset value is assumed as positive:

When a tool path is programmed as (A), if the offset value is negative, the tool center moves as in (B); when a tool path is programmed as (B), if the offset value is negative, the tool center moves as in (A).

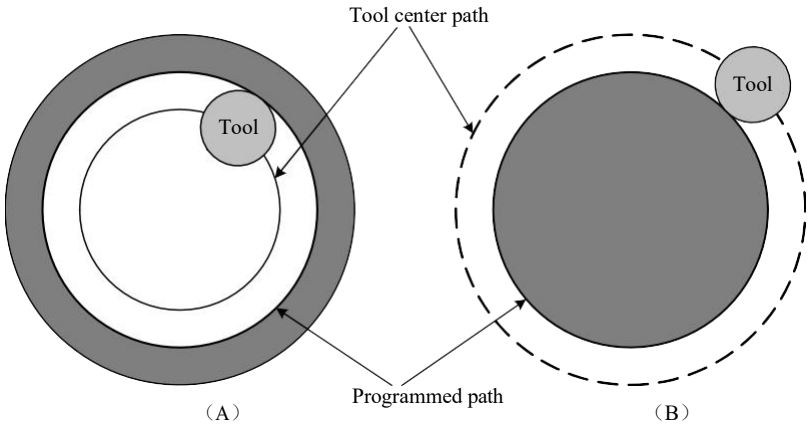


Fig. 3-7-2-8

It is common to see a figure with acute angles (figure with sharp-angle arc interpolation). However, if the offset value is negative, the inner side of the workpiece cannot be machined. When cutting the inner sharp angle at a point, insert an arc with a proper radius there, and then perform cutting after the smooth transition.

The compensation for left or right means the compensation direction is at the left side or right side of the tool moving direction relative to the workpiece (workpiece assumed as unmovable). By G41or G42, the system enters compensation mode, and by G40 the compensation mode is cancelled.

The example for compensation program is as follows:

The block (1), in which the compensation cancel mode is changed for compensation mode by G41 code, is called start. At the end of the block, the tool center is compensated by the tool radius that is vertical to the path of the next block (from P1 to P2). The offset value is specified by D07, i.e. the offset number is set to 7, and G41 specifies the tool path compensation left.

After the offset starts, when the workpiece figure is programmed as P1→P2……P9→P10→P11, the tool path compensation is performed automatically.

Example for tool path compensation program G92

X0 Y0 Z0；

(1) N1 G90 G17 G0 G41 D7 X250 Y550 ；（Offset value must be preset using offset number） (2) N2 G1 Y900 F150 ；

(3) N3 X450 ；

(4) N4 G3 X500 Y1150 R650 ；

(5) N5 G2 X900 R-250 ；

(6) N6 G3 X950 Y900 R650 ；

(7) N7 G1 X1150 ；

(8) N8 Y550 ；

(9) N9 X700 Y650 ； (10) N10 X250 Y550 ； (11) N11 G0 G40 X0 Y0 ；

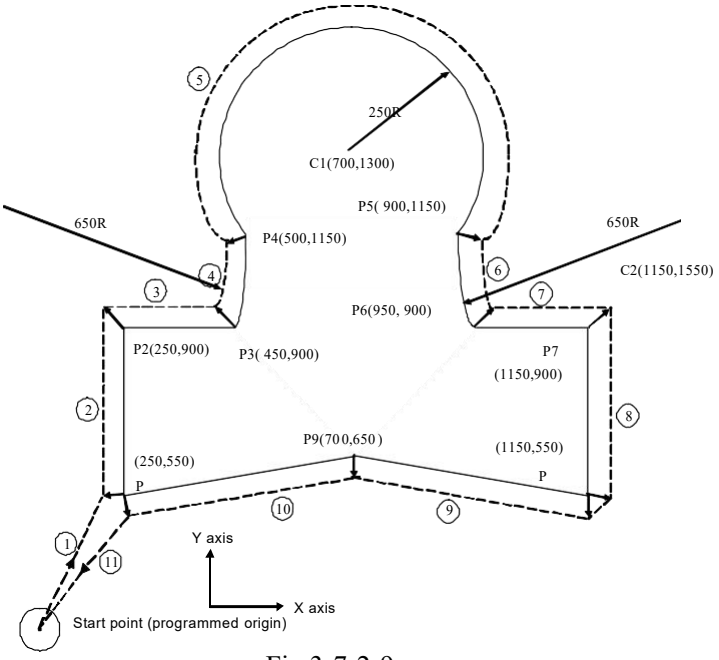


Fig.3-7-2-9

4.7.3 Explanation for Tool Radius Compensation

Conception: Inner side and outer side: when an angle of intersection created by tool paths specified with move codes for two blocks is over 180°, it is called inner side, when the angle is between 0° and 180°, it is called outer side.

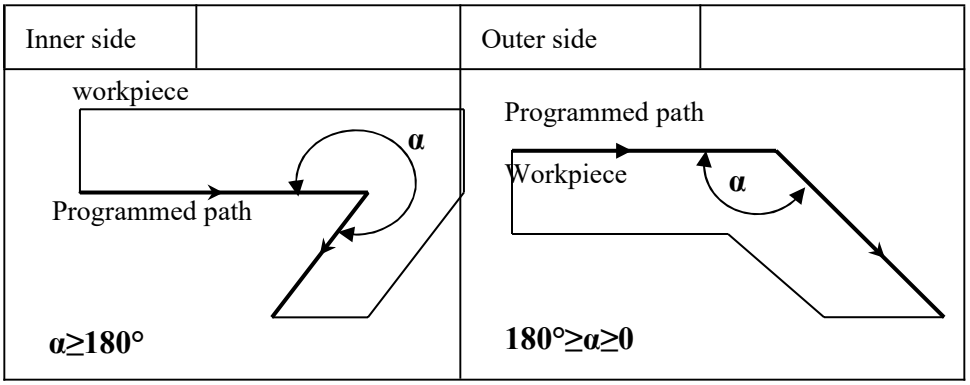


Fig. 3-7-3-1

Symbol meanings:

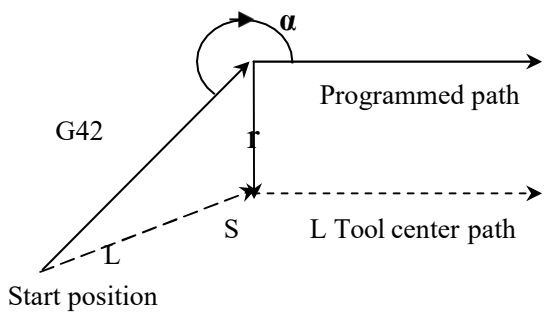
The following symbols are used in subsequent figures:

——S indicates a position at which a single block is executed once.
——SS indicates a position at which a single block is executed twice.
——SSS indicates a position at which a single block is executed three times
——L indicates that the tool moves along a straight line.
——C indicates that the tool moves along an arc.
——r indicates the tool radius compensation value.
——An intersection is a position at which the programmed paths of two blocks intersect with each other after they are shifted by r.
——O indicates the center of the tool.

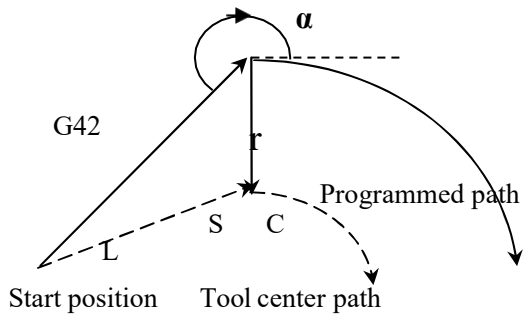
1. Tool movement in start-up When the offset cancel mode is changed to offset mode, the tool moves as illustrated below (start-up):

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

Linear-Linear



Linear-Circular

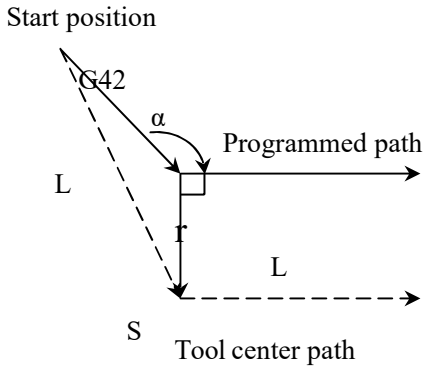


(b) Tool movement around an outer side of a corner at an obtuse angle （180°＞α≥90°）

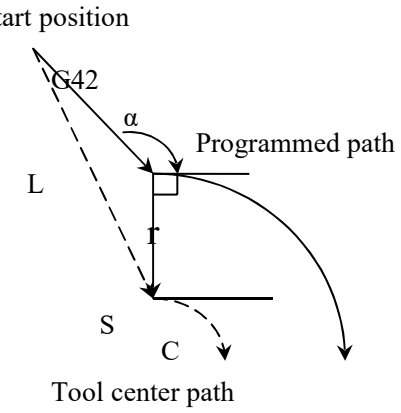
There are 2 tool path types at offset start or cancel: A and B, which are set by bit parameter No： 40#0.

A

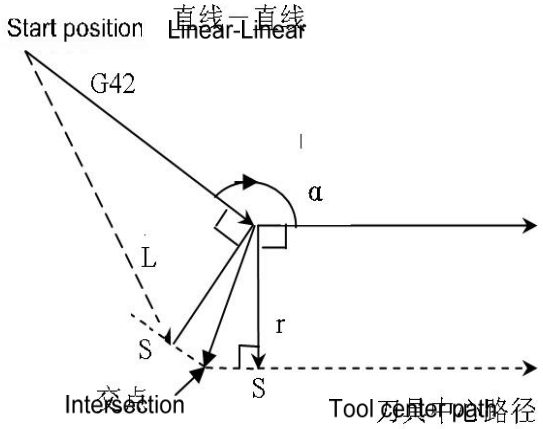
Linear -Linear



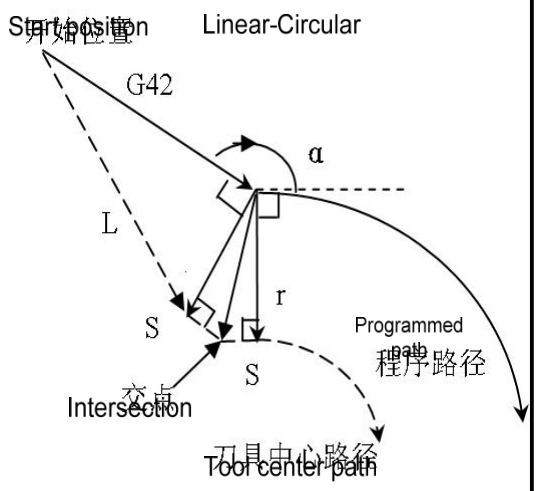
Linear-Circular



B



Note: Intersection is the position where offset paths of two successive blocks intersect.



（C）Tool movement around an outer side of a corner at an acute angle （α＜90°）

There are 2 tool path types at offset start or cancel: A and B, which are set by bit parameter NO:40#0.

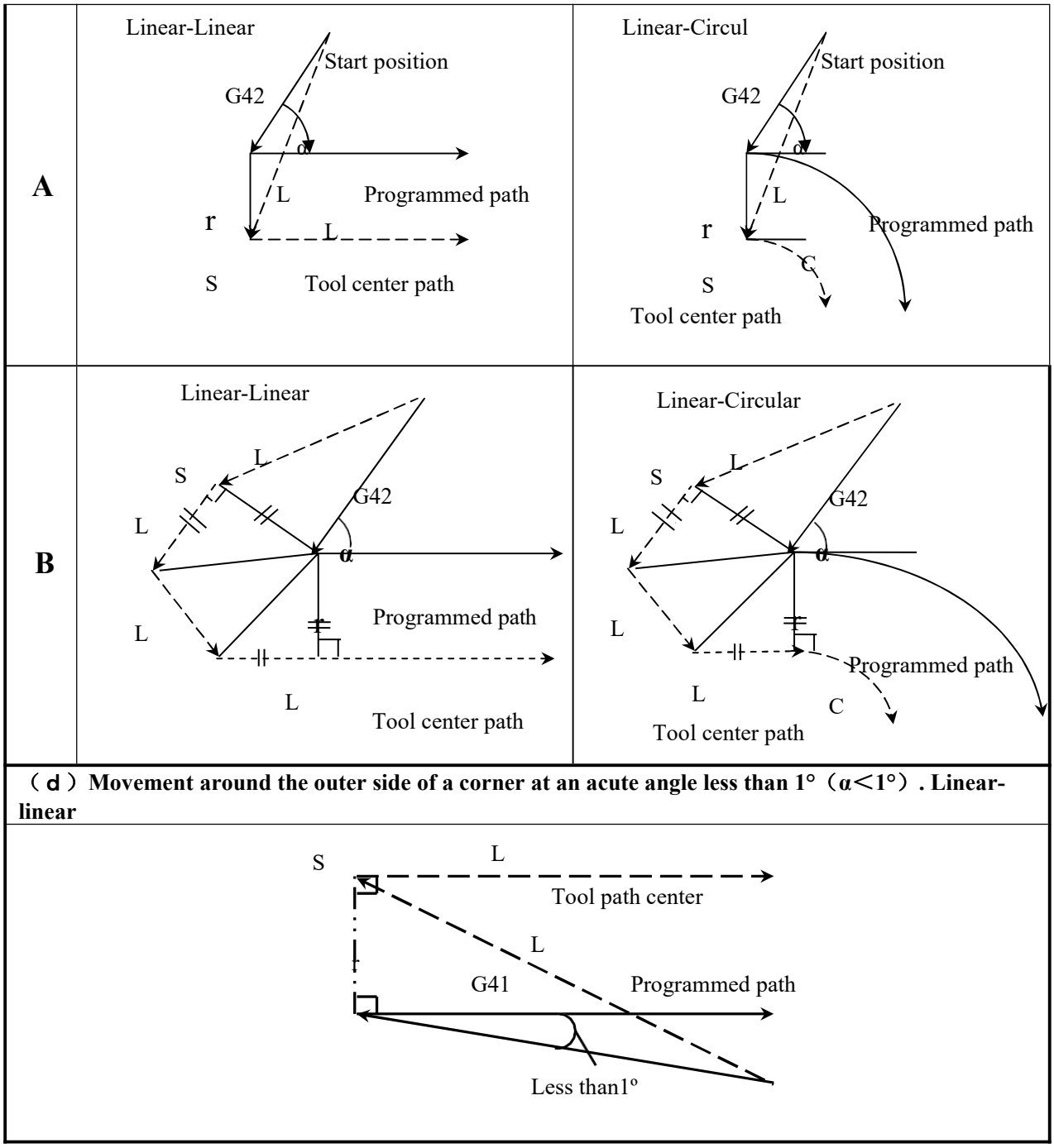


Fig. 3-7-3-2

2. Tool movement in offset mode

An alarm occurs and the tool is stopped if the offset plane is changed when the offset mode is being performed. The tool movement in the offset mode is shown below.

（a）Movement around an inner side of a corner （α≥180°）

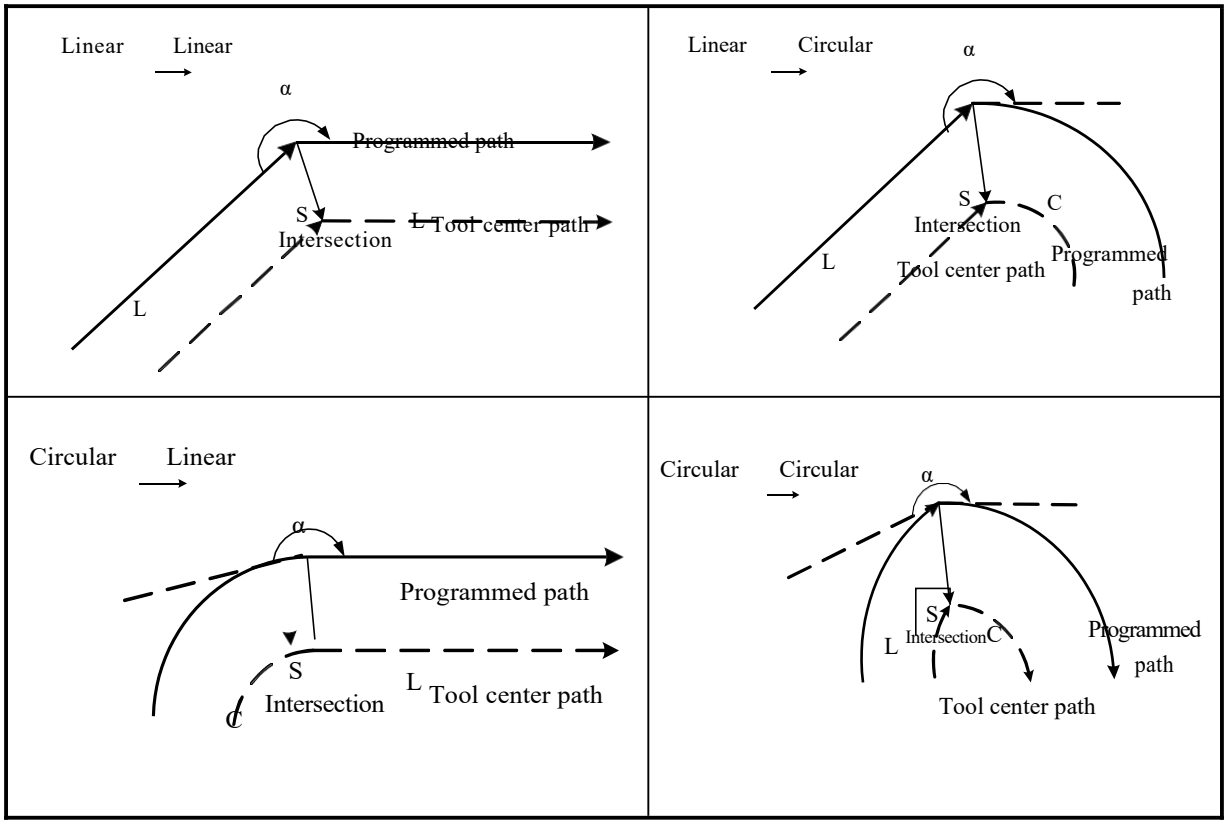


Fig. 4-7-3-3

3. Special cases

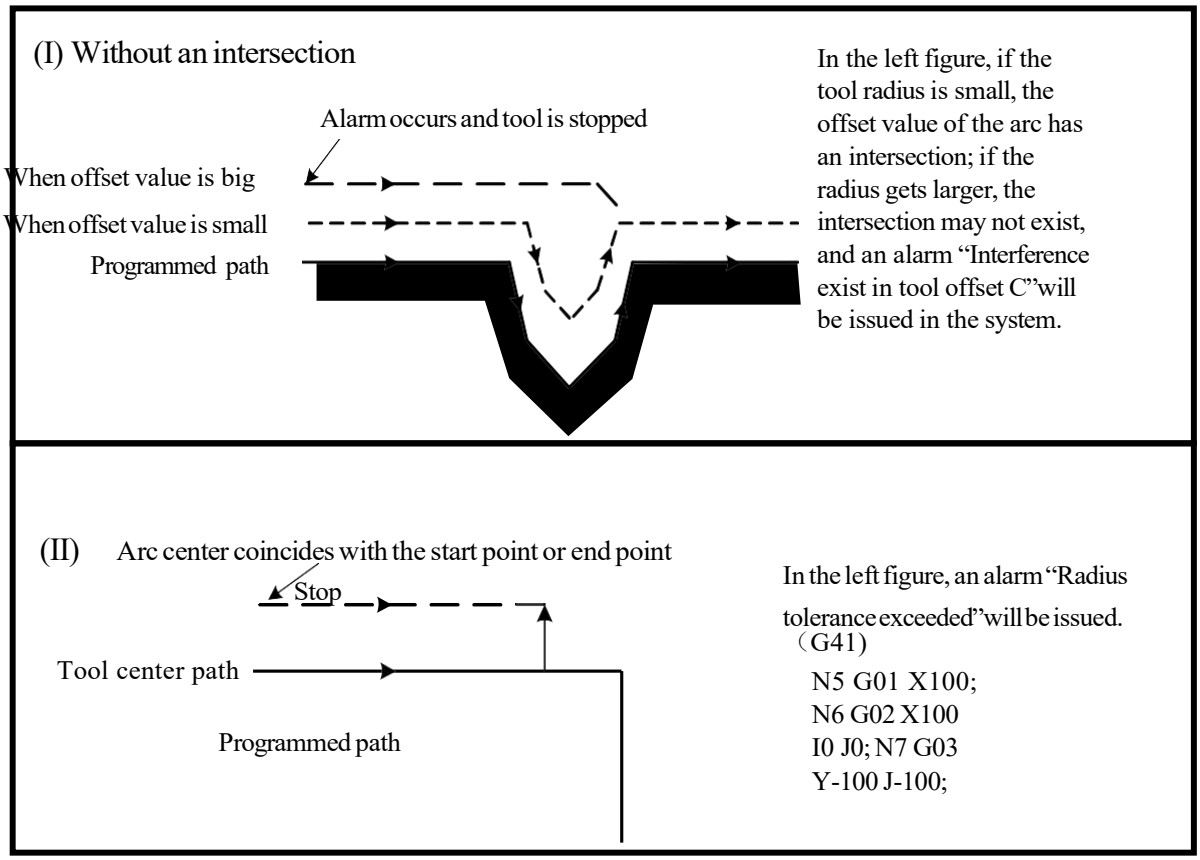


Fig. 3-7-3-4

Tool movement in offset cancel mode

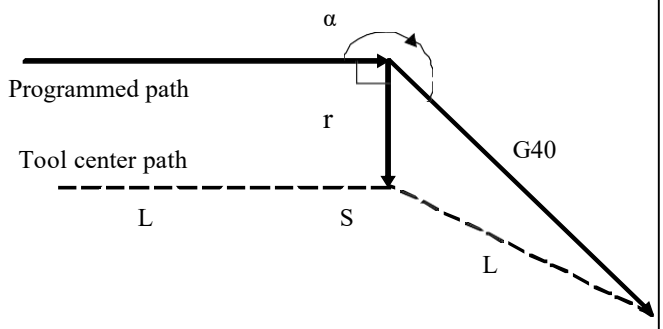
In the offset mode, when a block that satisfies any of the following conditions is performed, the system enters into offset cancel mode. The operation of this block is called the offset cancel. a) G40

b) When the tool radius compensation number is 0.

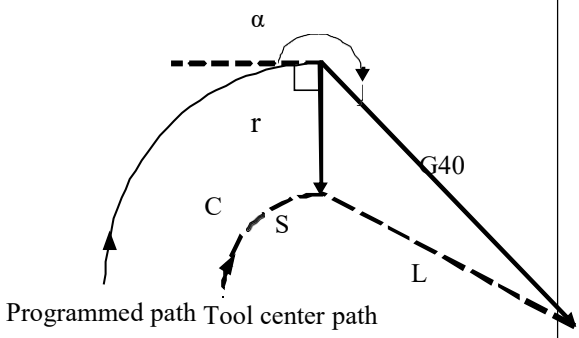
Arc code (G03 or G02) cannot be used for cancellation in offset cancel mode. An alarm is issued and tool is stopped if an arc is specified.

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

Linear→Linear



Circular→Linear

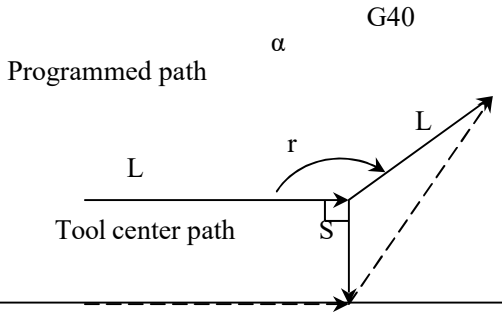


(b) Tool movement around the inner side of a corner (90° α 180°)

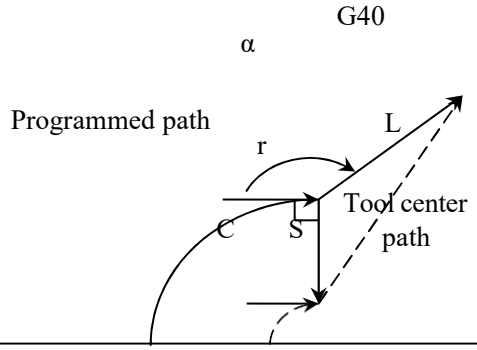
There are 2 tool path types at offset start or cancel: type A and type B, which are set by bit parameter NO: 40#0.

A

Linear—Linear

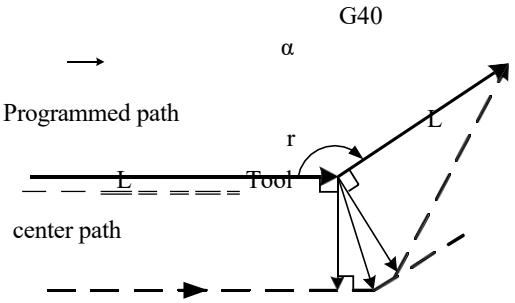


Circular—linear

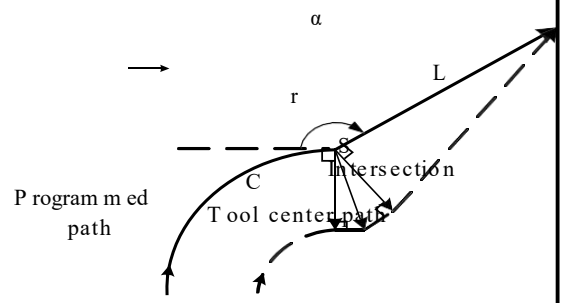


B

Linear Linear



Circular Linear



(c) Tool movement around an outer side of an corner at an acute angle (α 90°)

There are 2 types of tool paths at offset start or cancel: type A and type B, which are set by bit parameter NO: 40#0.

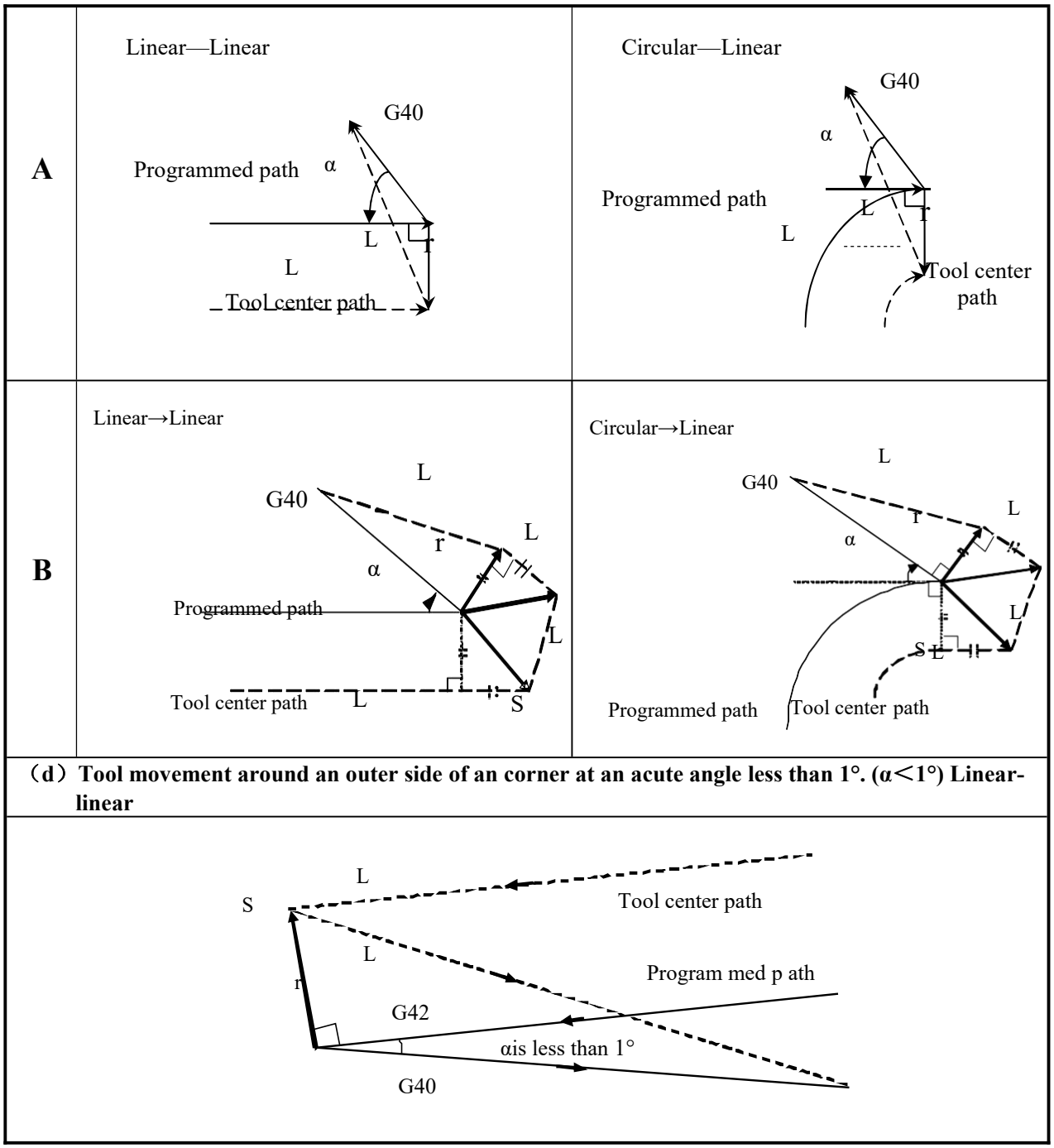


Fig. 3-7-3-5

4. Changing offset direction in offset mode

The offset direction is determined by tool radius compensation G code (G41 and G42). The signs of the offset value are as follows:

Table 3-7-3-1

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

In a special case, the offset direction can be changed in offset mode. However, the direction change is unavailable in the start-up block and the block following it. There is no such concepts as

inner and outer side when the offset direction is changed. The following offset value is assumed to be positive.

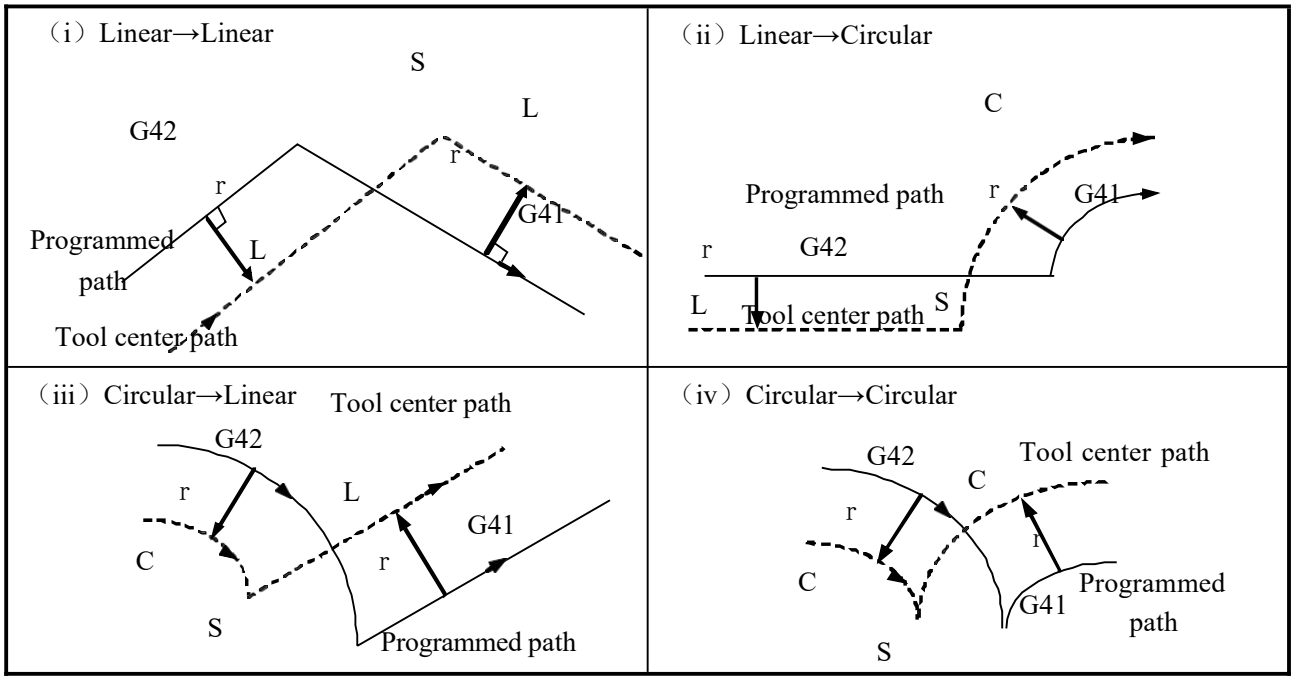


Fig. 3-7-3-6

（v） When the tool compensation is executed normally without an intersection

When changing the offset direction from block A to block B using G41 and G42, if the intersection of the offset path is not required, the vector normal to block B is created at the start point.

(1) Linear Linear

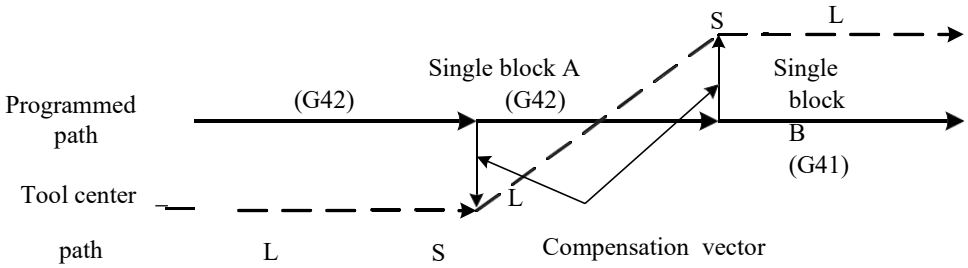


Fig. 3-7-3-7

(2) Linear----- Circular

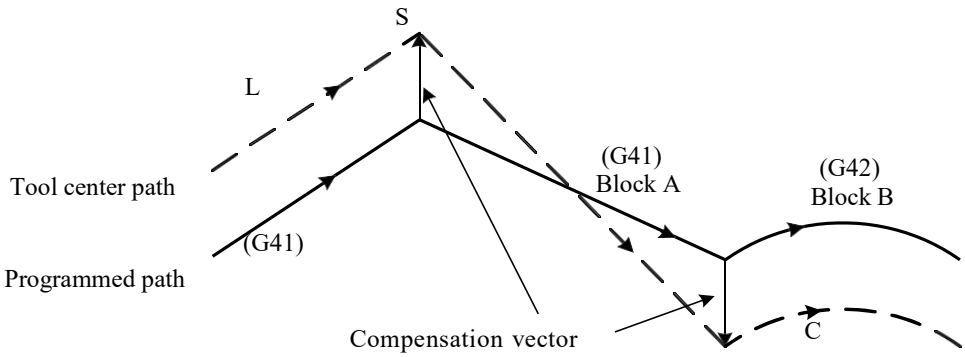


Fig. 3-7-3-8

(3) Circular----- Circular

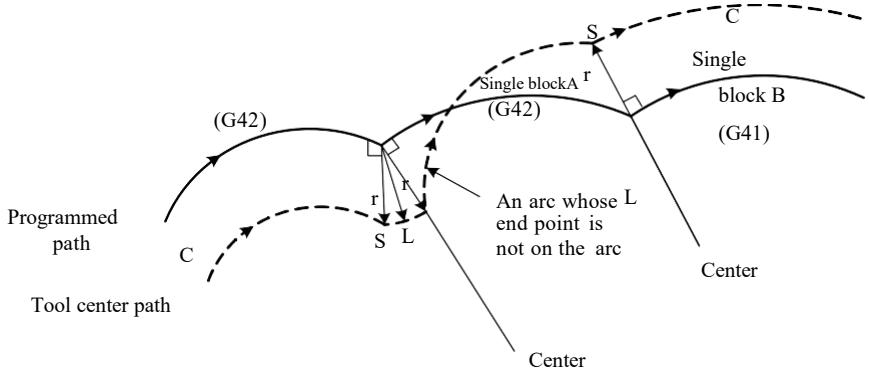
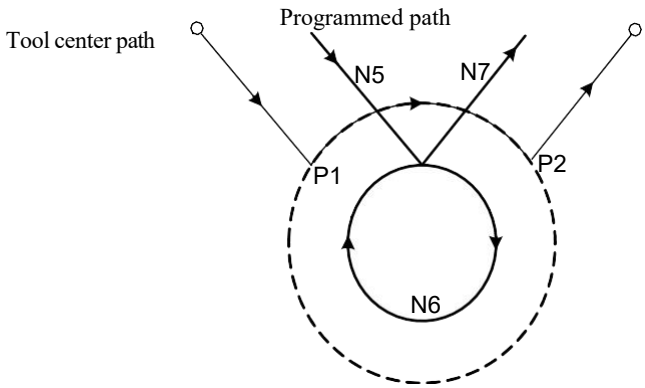


Fig. 3-7-3-9

（vi）Normally there is almost no possibility of generating the situation that the length of the tool center path is larger than the circumference of a circle. However, when G41 and G42 are changed, the following situation may occur:

Circular ----- circular (linear-----circular) An alarm occurs when the tool offset direction is changed, and an alarm “Tool offset cannot be cancelled by arc code” is issued when the tool number is D0.
Linear linear The tool offset direction can be changed.



(G42)
N5 G01 G91 X500 Y-700; N6 G41 G02 J-500;
N7 G42 G01 X500 Y700;
Here, the tool center path is not an arc of a circle, but an arc from P1 to P2. Under some conditions, an alarm may occur because of the interference check.
To move the tool around a full circle, the circle must be specified in segments.

Fig. 3-7-3-10

5. Temporary offset cancel

In offset mode, bit parameter NO：40#2 determines whether the offset is canceled at the intermediate point temporarily when G28, G30 is specified.

Please refer to the description of offset cancel and compensation start for detail information about this operation.

a） G28 automatic reference point return

If G28 is specified in offset mode, the offset is cancelled at the intermediate point and automatically restored after reference point return.

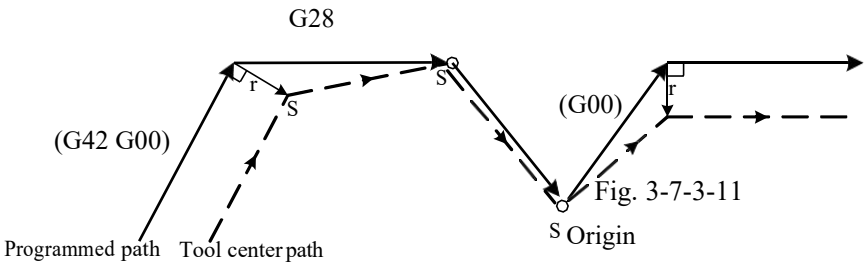


Fig. 3-7-3-11

b） G29 automatic return from reference origin point

If G29 is specified in offset mode, the offset is cancelled at the intermediate position and automatically restored at the next block.

If it is specified immediately after G28:

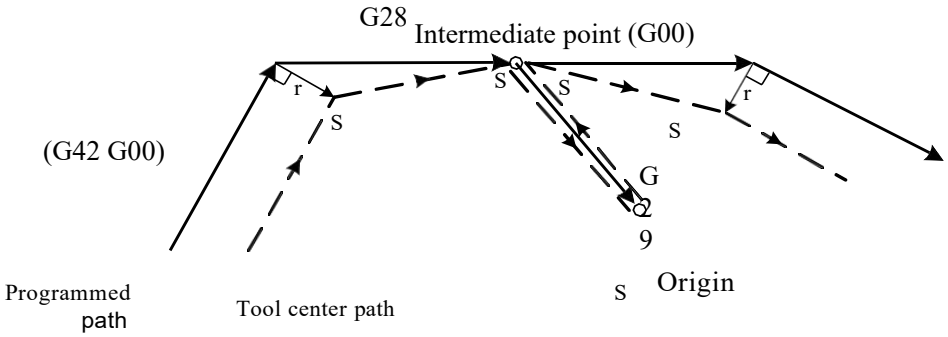


Fig. 3-7-3-12

If it is not specified immediately after G28:

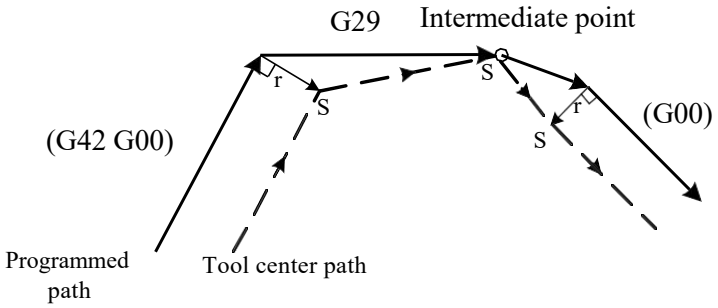


Fig. 3-7-3-13

6. Tool radius compensation G code in offset mode

In offset mode, if the tool radius compensation G code (G41, G42) is specified, a vector can be set to form a right angle to the moving direction in the previous block, which is irrelative to the machining inner or outer side. If this G code is specified in circular codes, the arc will not be correctly generated.

Refer to (5) when the offset direction is changed using tool radius compensation G (G41， G42). Linear---Linear

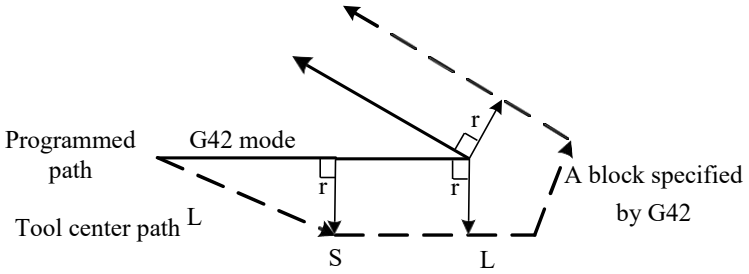


Fig. 3-7-3-14

Circular---Linear

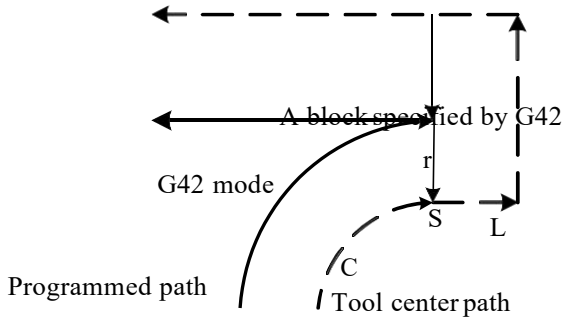


Fig. 3-7-3-15

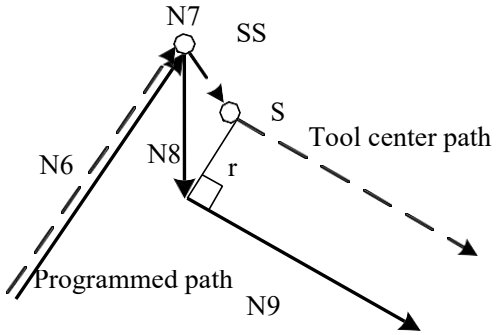
8 A block without tool movement

The following blocks have no tool movement. In these blocks, the tool will not move even if tool radius compensation mode is effective.

(1) M05 ; ..... M code output
(2) S21 ; ..... S code output
(3) G04 X10000; ..... Dwell
(4) (G17) Z100 ; .....Move code not included in offset plane
(5) G90 ; .....G code only
(6) G01 G91 X0; ..... Move distance is zero.

a） Specified at offset start

If the tool movement is not made by the start-up block, it will be done by the next moving code block by the system.

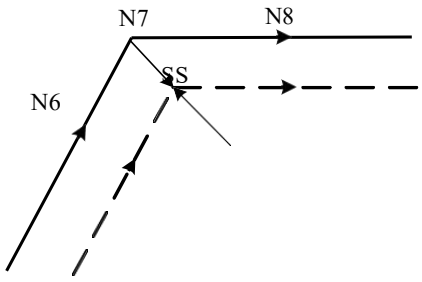



```
G40 G91……
.
.
N6 X100
Y100; N7
G41 X0;
N8 Y-100;
N9 X100 Y-100;
```

Fig. 3-7-3-16

b）Specified in offset mode

If a single block with no tool movement is specified in offset mode, the vector and the tool center path are the same as when the block is not specified. (Refer to item (3) Offset mode).This block is executed at the single block stop position.




```
N6 G91 X100
Y200; N7 G04
X100;
N8 X100;
```

Block N7 is executed here

Programmed path Tool center path

Fig. 3-7-3-17

However, when the block moving amount is 0, the tool movement is the same as that of two or more blocks without moving codes even if only one bloclk is specified.

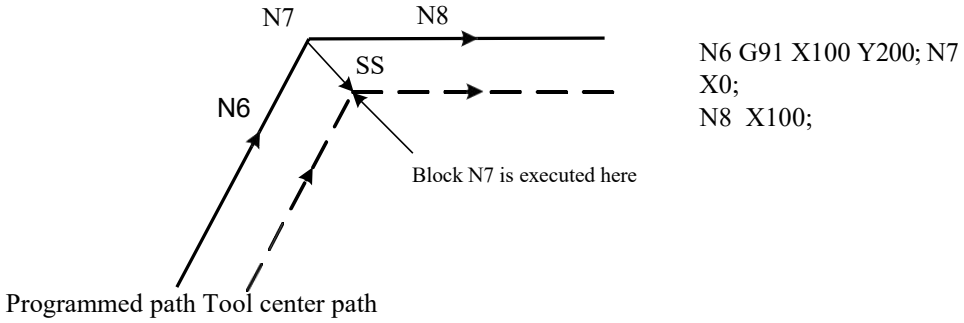


Fig. 3-7-3-18

Note: The blocks above are executed in G1, G41 mode. The path in G0 does not conform to the figure.

c）Specified together with offset cancel

A vector with a length of offset value and with its direction perpendicular to the movement direction of the previous block is formed when the block specified together with offset cancel contains no tool movement. This vector will be canceled in next moving code.

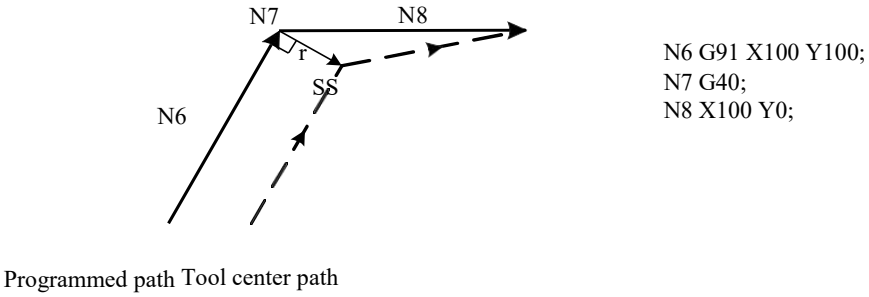


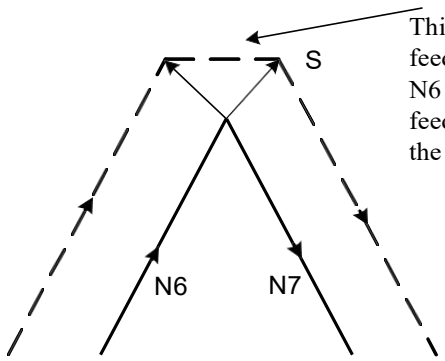
Fig. 3-7-3-19

9. Corner movement

If two or more vectors are formed at the end of the block, the tool traverses linearly from one vector to another. The movement is called corner movement.

If ΔVX≤ΔV limit and ΔVY≤ΔV limit, the latter vector is ignored.

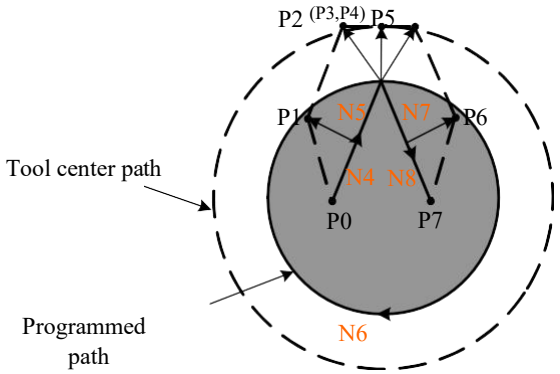
If these vectors do not coincide, then a movement around the corner is created. This movement belongs to the former block.



This move belongs to block N6, t hus, the feedrate is equal to that in block N6. If block N6 is G00 mode, the tool moves at rapid feedrate; if block N6 is G01, G02, G03 mode, the tool moves at cutting feedrate.

Fig. 3-7-3-20

However, if the path of the next block overpasses the semicircle, the function above is not performed. The reason is that:



N4 G41 G91 X150 Y200; N5 X150 Y200;
N6 G02 J-600;
N7 G01 X150 Y-200; N8 G40 X150 Y-200;

Fig. 3-7-3-21

If the vector is not ignored, the tool path is as follows: P0 →P1 →P2 →P3（arc） →P4 →P5 →P6 →P7

If the distance between P2 and P3 is ignored, P3 is ignored. The tool path is as follows: P0 →P1 →P2 →P4 →P6 →P7. The arc cutting of the block N6 is ignored.

10. Interference check

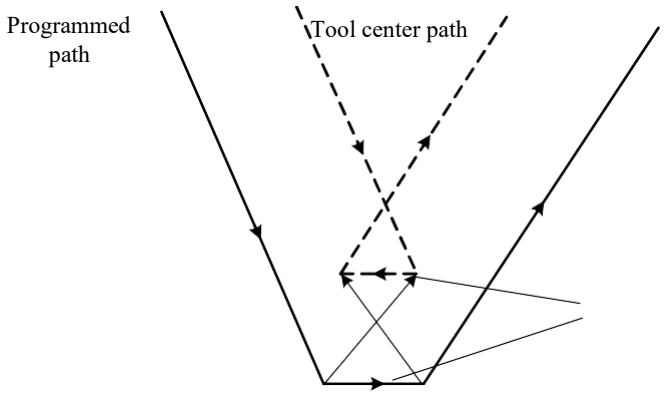
The tool overcutting is called “interference”. The Interference check function checks the tool overcutting in advance. If the interference is detected by grammar check function after the program is loaded, an alarm is issued. Whether the interference check is performed during radius compensation is set by bit parameter NO：41#6.

Basic conditions for interference

（1） The moving distance of the block which establishes tool radius compensation is less than the tool radius.

(2) The direction of the tool path is different from that of the program path. (The included angle between the two paths is from 90°to 270°).

（3）Besides the above conditions, in arc machining, the included angle between the start point and the end point of the tool center path is very different from that between the start point and end point of the program path(above 180°).


Programmed
path
Tool center path

The directions of the two paths are quite different （180°）.

Fig. 3-7-3-22

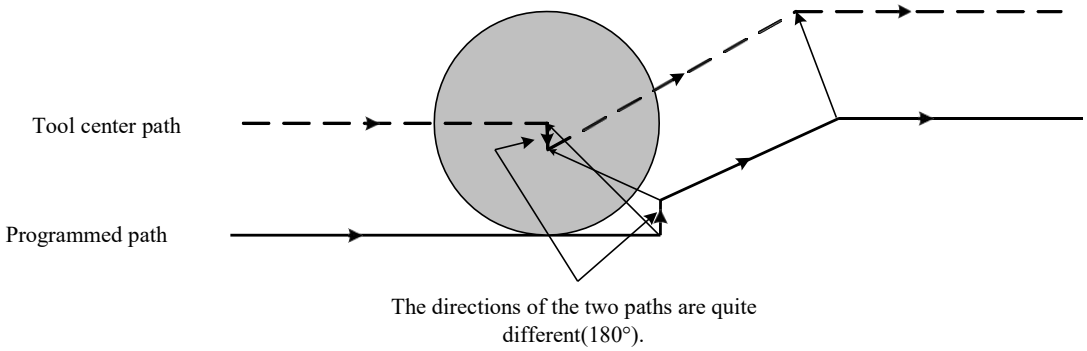


Fig. 3-7-3-23

11. Manual operation

Refer to Manual Operation section in Operation part for the manual operation during the tool radius offset.

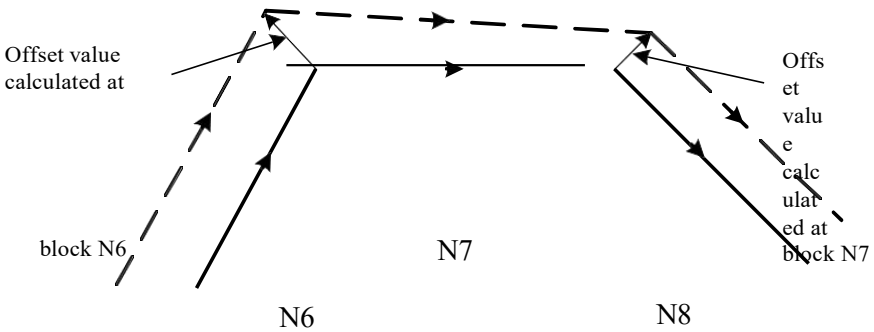
12. Precautions for offset

a） Specifying offset value

The offset value number is specified by D code. Once specified, D code keeps effective till another D code is specified or the offset is cancelled. D code is not only used for specifying the offset value for the tool radius compensation, but also for specifying offset value for tool offset.。

b） Changing offset value

In general, during tool change, the offset value must be changed in offset cancel mode. If it is changed in offset mode, the new offset value is calculated at the end of the block.



Programmed path

Fig. 3-7-3-24

c）Positive/negative offset value and tool center path

If the offset value is negative(－), G41 and G42 are replaced with each other in the program. If the tool center is passing around the outer side of the workpiece, it will pass around the inner side instead, and vice versa.

As shown in the example below: In general, the offset value is programmed to be positive（＋）. When a tool path is programmed as in figure（a）, if the offset value is made for negative（－）, the tool center moves as in（b）,and vice versa. Therefore, the same program permits cutting for male or female shape, and the gap between them can be adjusted by the selection of the offset value.

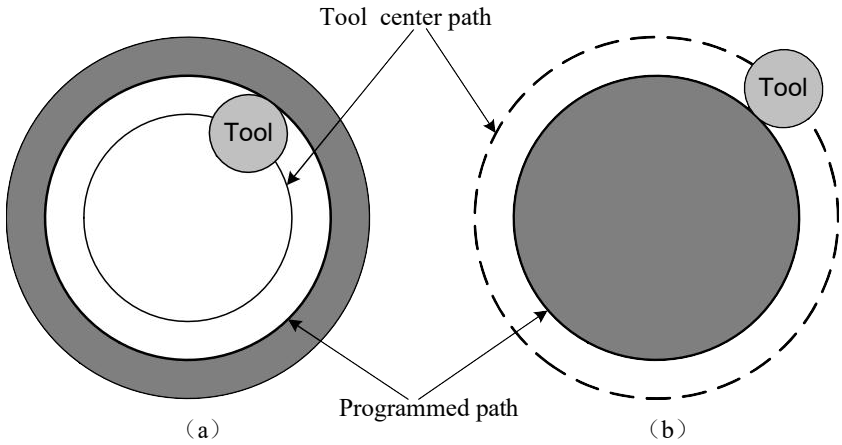
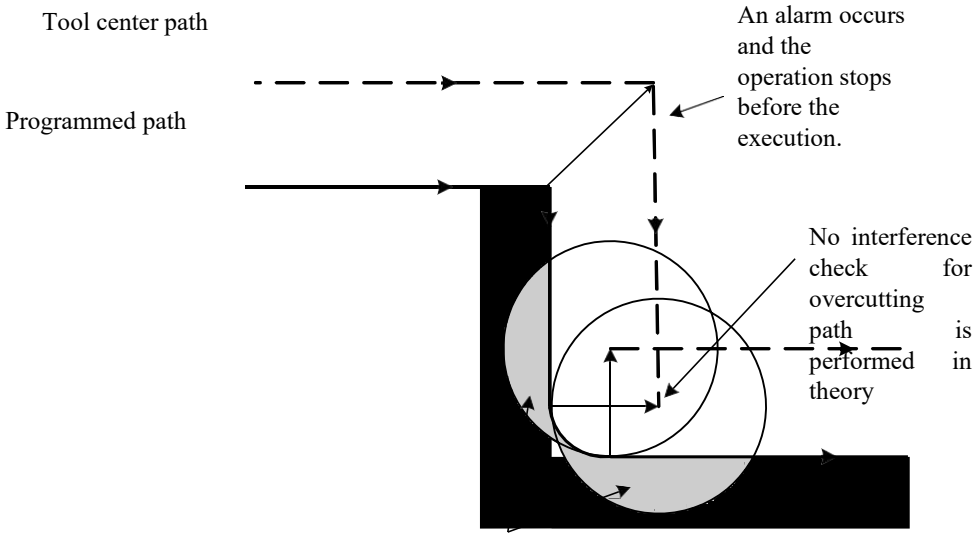


Fig. 3-7-3-25

d）Overcutting by tool radius compensation

(1) Machining an inner side of the corner at a radius smaller than the tool radius

When the radius of a corner is smaller than the tool radius, because the inner offsetting of the tool will result in overcutting, an alarm for interference occurs and the CNC stops before the execution of the program.



An overcutting occurs if the CNC does not stop

Fig.3-7-3-26

（2） When machining a groove smaller than the tool radius

When a groove smaller than the tool radius is machined, since the tool radius offset forces the path of the tool center to move in the reverse direction of the programmed path, the overcutting will occur.

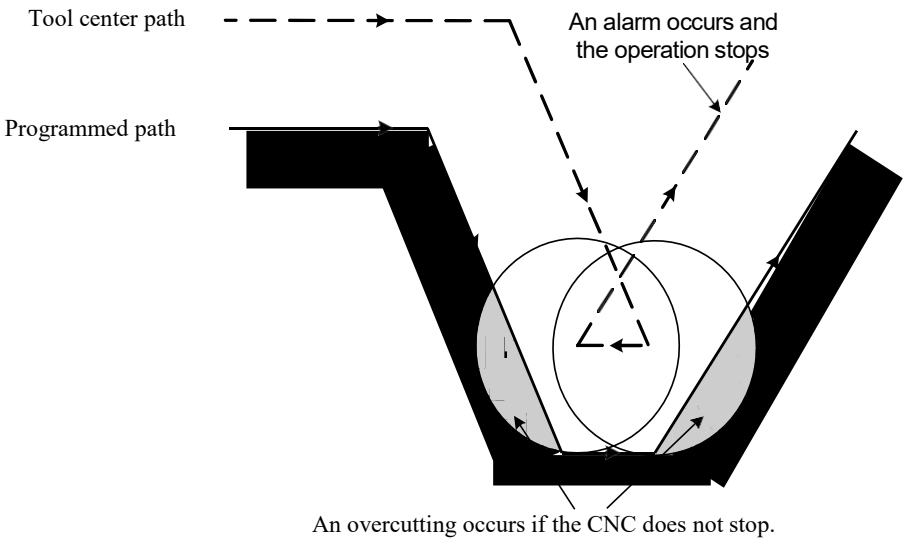


Fig. 3-7-3-27

（3） Machining a step smaller than the tool radius

When the machining of the step is instructed by circular machining in the case of a program containing a step smaller than the tool radius, the tool center path with the common offset becomes reverse to the programmed direction. In this case, the first vector is ignored, and the tool moves linearly to the second vector position. This single block operation is stopped at this point. If the machining is not in the single block mode, the auto run continues. If the step is linear, no alarm will be issued and the tool cuts correctly. However, the uncut part will exist.

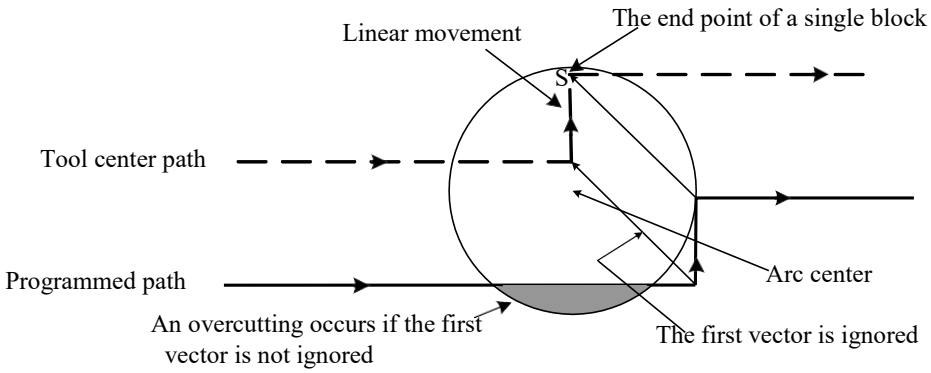


Fig. 3-7-3-28

Starting tool radius compensation and cutting along Z axis

It is usually used such a method that the tool is moved along the Z axis after the tool radius compensation is effected at some distance from the workpiece at the start of the machining. In the case above, if it is desired to divide the motion along the Z axis into rapid feed and cutting feed, follow the procedure below:

If block N3 is divided as follows: N1

```
G91 G00 X500 Y500 H01; N3 Z-250;
N5 G01 Z-50 F1;
N6 Y100 F2;
```

N1 G91 G0 X500 Y500 H01; N3
G01 Z-300 F1;
N6 Y100 F2;

N6 is entered into the buffer storage when N3 is being executed. By the relationship between them the correct offset is performed in the left figure.

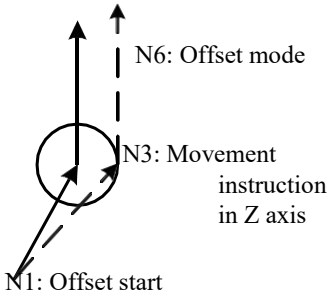


Fig. 3-7-3-29

4.7.4 Corner offset circular interpolation（G39）

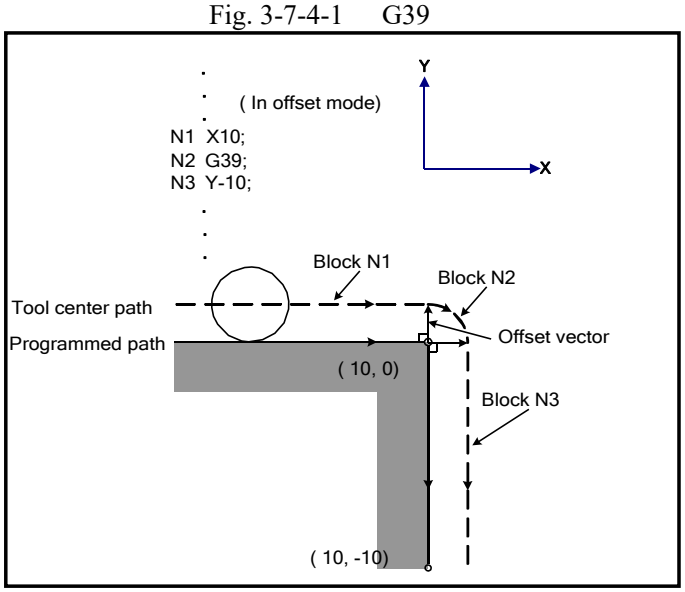
Format: **G39**

Function: By specifying G39 in offset mode during tool radius compensation, corner offset circular interpolation can be specified. The radius of the corner offset equals the offset value. Whether the corner arc is valid or not is determined by bit parameter NO：41#5.

Explanation:

1. When G39 is specified, corner circular interpolation of which the radius equals offset value can be performed.
2. G41 or G42 preceding this code determines whether the arc is CW or CCW. G39 is a non-modal G code.
3. When G39 is programmed, the arc is formed at the corner so that the vector at the end point of the arc is perpendicular to the start point of the next block. It is shown as follows:

Fig. 3-7-4-1 G39



4.7.5 Tool Offset Value and Offset Number Input by Program（G10）

Format: **G10 L10 P_ R_ ;** Geometric offset value of H code **G10 L12 P_ R_ ;** Geometric offset value of D code **G10 L11 P_ R_ ;** Wear offset value of H code
G10 L13 P_ R_ ; Wear offset value of D code
P : Tool offset number
R : Tool offset value in absolute mode（G90）

Value to be added to the value of the specified offset number in incremental mode （G91）(the sum is the tool offset value).

Explanation: The range of tool offset value:

Geometric offset: metric input -999.999mm~+999.999mm；
inch input -99.9998inch~+99.9998inch Wear
offset: metric input -400.000mm~+400.000mm；
inch input -40.0000inch~+40.0000inch

Note : The max. value of the wear offset is restrained by data parameter P267.

4.8 Feed G Code

4.8.1 Feed Mode G64/G61/G63

Format: Exact stop mode **G61**
Taping mode **G63** Cutting
mode **G64**

Function:

Exact stop mode G61: Once specified, this function keeps effective till G62, G63 or G64 is specified. The tool is decelerated for an in-position check at the end point of a block, then next block is executed.

Tapping mode G63: Once specified, this function keeps effective till G61, G62 or G64 is specified. The tool is not decelerated at the end point of a block, but the next block is executed. When G63 is specified, both feedrate override and feed hold are invalid.

Cutting mode G64: Once specified, this function keeps effective till G61, G62 or G63 is specified. The tool is not decelerated at the end point of a block, and the next block is executed.

Explanation:

1. No parameter format.
2. G64 is the system default feed mode, no deceleration is performed at the end point of a block, and next block is executed directly.
3. The purpose of in-position check in exact stop mode is to check whether the servo motor has reached within a specified position range.
4. In exact stop mode, the tool movement paths in cutting mode and tapping mode are different.

See Fig. 3-8-1-1

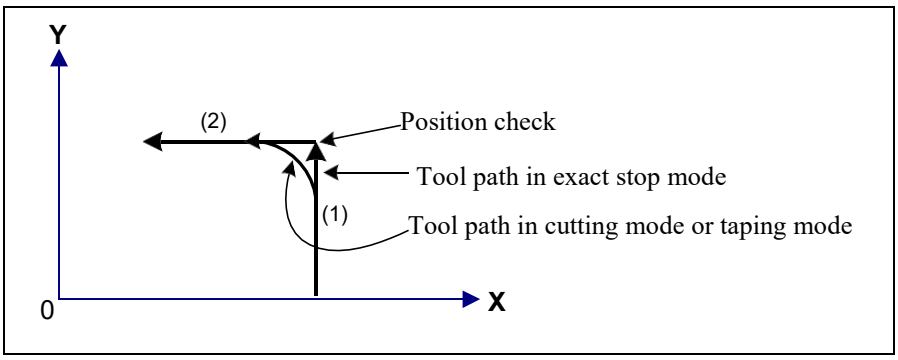


Fig. 3-8-1-1 Tool path from block 1 to block 2

4.8.2 Automatic Override for Inner Corners（G62）

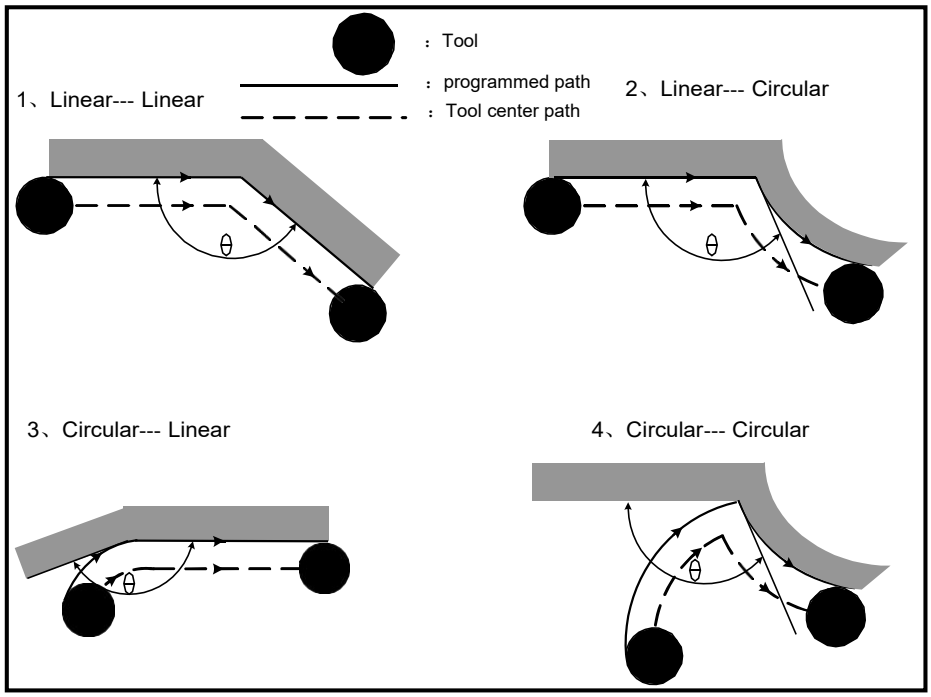
Format： **G62**

Function: Once specified, this function keeps effective till G63, G61 or G64 is specified. When the tool moves along an inner corner during tool radius compensation, override is applied to the cutting feedrate to suppress the amount of cutting per unit time. In this way, a smooth machined surface is produced.

Explanation:

1. When the tool moves along an inner corner and inner arc area during tool radius compensation, it is decelerated automatically to reduce the load on the tool and produce a smooth machined surface.
2. Whether automatic corner override function is valid or not is set by bit parameter NO：16#7; Automatic corner deceleration function is controlled by bit parameter NO： 15#2(0: angle control, 1: speed difference control).
3. When G62 is specified, and the tool path with tool radius compensation applied forms an inner corner, the feedrate is automatically overridden at both ends of the corner. There are four types of inner corners as shown in Fig. 3-6-2-1. In the figure: 2°≤θ≤θp≤178°; θp is set by data parameter P144.

Fig. 3-8-2-1



4. When a corner is determined to be an inner corner, the feedrate is overridden before and after the inner corner. The Ls and Le, where the feedrate is overridden, are distances from points on the tool center path to the corner. As shown in Fig. 3-8-2-2, Ls+Le≤2mm.

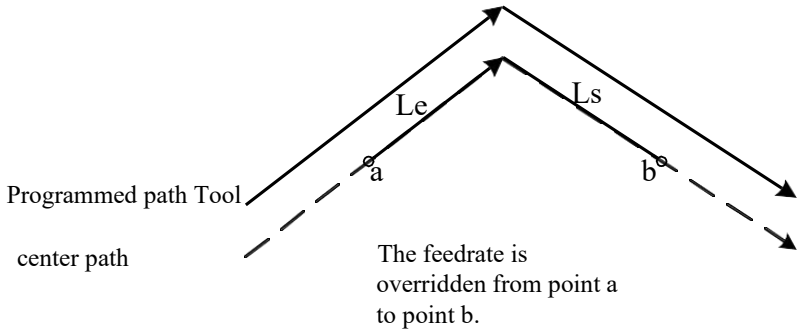


Fig. 3-8-2-2 Straight line to straight line

5. When a programmed path consists of two arcs, the feedrate is overridden if the start and end points are in the same quadrant or in adjacent quadrants, and P145 controls the lowest feedrate of the automatic corner deceleration. (Fig. 3-8-2-3)

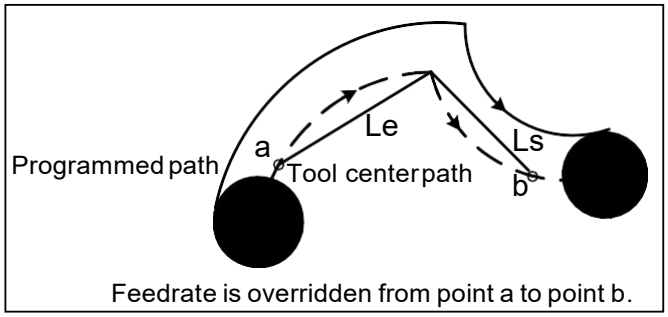


Fig. 3-8-2-3 Arc to arc

6. Regarding a program from straight line to arc or from arc to straight line, the feedrate is overridden from point a to point b and from point c to point d. (Fig. 3-8-2-4)

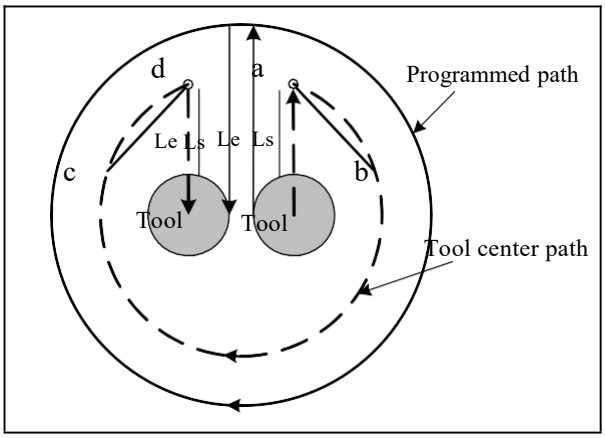


Fig.3-8-2-4 Straight line to straight line, arc to straight line

Restrictions:

1. Override for inner corners is disabled during acceleration/deceleration before interpolation.
2. Override for inner corners is disabled if the corner is preceded by a start-up block or followed by a block including G41 or G42.
3. Override for inner corners is not performed if the offset is zero.

4.9 Macro G Code

4.9.1 Custom Macro

The functions realized by a group of codes can be prestored into memory like a subprogram using an representing code. If the code is written into the program, all these functions can be realized. This group of codes is called custom macro body, and the representing code is called “custom macro code”. Moreover, the custom macro body is also called “macro program” for short, and the custom macro code is also called macro calling code.

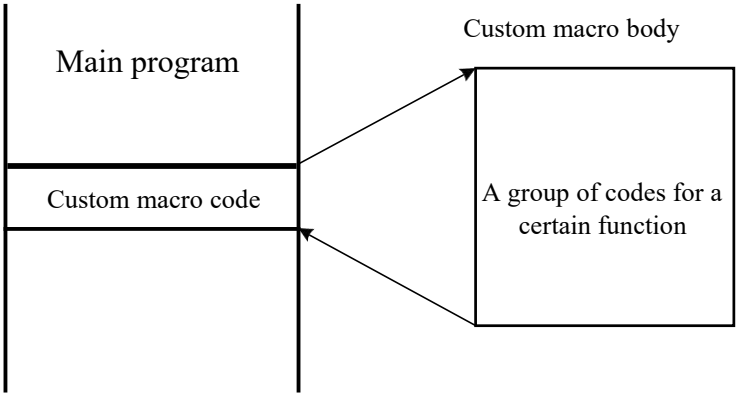


Fig. 3-9-1-1

Variables can be used in custom macro body. Operation can be performed between them and they can be assigned values by macro instructions.

4.9.2 Macro Variables

The common CNC instructions and the variables, operation as well as the transfer instructions can be used in the custom macro body.

The custom macro body begins with a program number and ends with M99.

```
O0066;                    Programnumber
G65H01……;                Operation instruction
G90G00X#101……;            CNCinstructionusingvariables
……
……
……
G65H82……;                 Transfer instruction
……
……
M99;                      Custommacro body ends
```

Fig. 3-9-2-1 （structure of custom macro body）

1. Variable usage

With a variable, the parameter value in custom macro body can be specified. The variable value can be assigned by the main program, or set by LCD/MDI, or be assigned by a computation during the execution of custom macro body.

Multiple variables can be used in custom macro and they are differentiated by their variable numbers.

(1) Variable representation

The variable is expressed by a sign # followed by a variable number, the format of which is as follows:

#i (i = 1， 2， 3， 4 ……)

(example) #5， #109， #1005

(2) Variable reference

The variable can be used to replace the value of a parameter. (Example)

F#103 When #103 = 15, it is the same as F15.

G#130 When #103 =3, it is the same as G3.

Note 1: Variables cannot be referenced by parameter word O and N (program number and sequence number), e.g., O#100 and N#120 are not permitted in programming.

Note 2: Variables exceeding the max. limit of the parameter cannot t be used. When #30 =120, M#30 exceeds the max. limit of the instruction.

Note 3: Display and setting of variable values: The values can be displayed on LCD, or be set by MDI mode.

2. Types of variables

Variables are divided into null variables, local variables, common variables and system variables depending on their different applications and characteristics.

（1） Null variable: #0 (This variable is always null, so no value can be assigned to it.)

（2） Local variables: #1～#50： they can only be used for data storage in a macro, such as the results of operations. When the power is turned off or the program ends (M30 or M02 is executed), they are cleared automatically; whether the local variables are cleared or not after reset is set by bit parameter NO：52#7. When a macro is called, arguments are assigned to local variables.

（3） Common variables: #100～#199， #500～#999： whether common variables #100～#199 are cleared or not after reset is set by bit parameter NO：52#6.

The common variables can be shared among the main program and the custom macros called by the main program. Namely, the variable #I in a custom macro program is the same as those in other macro programs. Therefore, the common variable #I of operation result of a macro program can be used in other macro programs.

The usage of common variables is not specified in this system, users thus can define it freely.

Table 3-9-2-1

Variable number	Variable type	Function
# 100～ # 199	Common variable	They are cleared at power-off, and all are initialized to “null” at power-on
# 500～ # 999		Data is saved in files and it will not be lost even if the power is turned off.

（4） System variables: They are used for reading and writing a variety of CNC data, which are shown as follows:

1) Interface input signal #1000 --- #1015 (read signal input to system from PLC by bit, i.e. G signal) #1032 (read signal input to system from PLC by byte, i.e., G signal)

2) Interface output signal #1100 --- #1115 (write signal output to PLC from the system by bit, i.e. F signal)
#1132 (write signal output to PLC from the system by byte, i.e. F signal)

3) Tool length offset value #1500～#1755 （readable and writable）

4) Tool length wear offset value #1800～#2055 （readable and writable）

5) Tool radius offset value #2100～#2355 （readable and writable）

6) Tool radius wear offset value #2400～#2655 （readable and writable）

7) Alarm #3000
8) User data list #3500～#3755 （read-only，unwritable）
9) Modal message #4000～#4030 （read-only，unwritable）
10) Position message #5001～#5030 （read-only，unwritable）
11) Workpiece zero offset #5201～#5235 （readable and writable）
12) Additional workpiece coordinate system #7001～#7250 （readable and writable）

3. Explanation for system variables

1） Modal message

Table 3-9-2-2

Variable number	Function	Group number
#4000	G10,G11	00
#4001	G00,G01,G02,G03	01
#4002	G17,G18,G19	02
#4003	G90,G91	03
#4004	G94,G95	04
#4005	G54,G55,G56,G57,G58,G59	05
#4006	G20,G21	06
#4007	G40,G41,G42	07
#4008	G43,G44,G49	08
#4009	G22,G23,G24,G25,G26 G32,G33,G34,G35,G36,G37,G38 G73,G74,G76,G80,G81,G82,G83,G84,G85,G86,G87,G88,G89	09
#4010	G98,G99	10
#4011	G15,G16	11
#4012	G50,G51	12
#4013	G68,G69	13
#4014	G61,G62,G63,G64	14
#4015	G96,G97	15
#4016	Reserved	16
#4017	Reserved	17
#4018	Reserved	18
#4019	Reserved	19
#4020	Reserved	20
#4021	Reserved	21
#4022	D	
#4023	H	
#4024	F	
#4025	M	
#4026	S	
#4027	T	
#4028	N	
#4029	O	
#4030	P(the current selected additional workpiece coordinate system)	

Note 1: P code indicates the current selected additional workpiece coordinate system. Note 2: When G#4002 code is being executed, the value obtained in #4002 is 17, 18 or 19. Note 3: The modal message can be read but not written.

2） Current position message

Table 3-9-2-3

Variable number	Position message	Relative coordinate system	Reading operation during moving	Tool offset value
#5001	Block end position of X axis (ABSIO)	Workpiece coordinate system	allowed	Tool nose position not involved （Position instructed by program）
#5002	Block end position of Y axis (ABSIO)			
#5003	Block end position of Z axis (ABSIO)			
#5004	Block end position of 4th axis (ABSIO)			
#5006	Block end position of X axis (ABSMT)	Machine coordinate system	unallowed	Tool reference Position involved （Machine coordinate）
#5007	Block end position of Y axis (ABSMT)			
#5008	Block end position of Z axis (ABSMT)			
#5009	Block end position of 4th axis (ABSMT)			
#5011	Block end position of X axis (ABSOT)	Workpiece coordinate system		
#5012	Block end position of Y axis (ABSOT)			
#5013	Block end position of Z axis (ABSOT)			
#5014	Block end position of 4th axis (ABSOT)			
#5016	Block end position of X axis (ABSKP)		allowed	
#5017	Block end position of Y axis (ABSKP)			
#5018	Block end position of Z axis (ABSKP)			
#5019	Block end position of 4th axis (ABSKP)			
#5021	Tool length offset value of X axis		unallowed	
#5022	Tool length offset value of Y axis			
#5023	Tool length offset value of Z axis			
#5024	Tool length offset value of 4th axis			
#5026	Servo position offset of X axis			
#5027	Servo position offset of Y axis			
#5028	Servo position offset of Z axis			
#5029	Servo position offset of 4th axis			

Note 1: ABSIO： The end point coordinates of the last block in workpiece coordinate system. Note 2: ABSMT：The current machine coordinate system position in machine coordinate system Note 3: ABSOT：The current coordinate position in workpiece coordinate system
Note 4: ABSKP：The effective position of the skip signal of block G31 in workpiece coordinate system.

3）Workpiece zero offset value and additional zero offset value:

Table 3-9-2-4

Variable number	Function
#5201 … #5204	External workpiece zero offset value of 1st axis … External workpiece zero offset value of 4th axis
#5206 … #5209	G54 workpiece zero offset value of 1st axis … G54 workpiece zero offset value of 4th axis
#5211 … #5214	G55 workpiece zero offset value of 1st axis … G55 workpiece zero offset value of 4th axis
#5216 … #5219	G56 workpiece zero offset value of 1st axis … G56 workpiece zero offset value of 4th axis
#5221 … #5224	G57 workpiece zero offset value of 1st axis … G57 workpiece zero offset value of 4th axis
#5226 … #5229	G58 workpiece zero offset value of 1st axis … G58 workpiece zero offset value of 4th axis
#5231 … #5234	G59 workpiece zero offset value of 1st axis … G59 workpiece zero offset value of 4th axis
#7001 … #7004	G54 P1 workpiece zero offset value of 1st axis … G54 P1 workpiece zero offset value of 4th axis
#7006 … #7009	G54 P2 workpiece zero offset value of 1st axis … G54 P2 workpiece zero offset value of 4th axis
#7246 … #7249	G54 P50 workpiece zero offset value of 1st axis … G54 P50 workpiece zero offset value of 4th axis

4. Local variables

The correspondence between address and local variable:

Table 3-9-2-5

Argument address	Local variable No.	Argument address	Local variable No.
A	#1	Q	#17
B	#2	R	#18
C	#3	S	#19
I	#4	T	#20
J	#5	U	#21
K	#6	V	#22
D	#7	W	#23
E	#8	X	#24
F	#9	Y	#25
M	#13	Z	#26

Note 1: The assignment is done by an English letter followed by a numerical value. Except letters G, L, O, N, H and P, all the other 20 letters can assign values for arguments. Each letter from A-B-C-D… to X－Y－Z can assign a value once and the assignment needs not to be performed in alphabetical order. The addresses that assign no values can be omitted.

Note 2: G65 must be specified before any argument is used.

5. Precautions for custom macro body

1) Input by keys
Press key # behind the parameter words G, X, Y, Z, R, I, J, K, F, H, M, S, T, P, Q for inputting "#".

2) Either operation or transfer instruction can be specified in MDI mode.

3) H, P, Q, R of the operation and transfer instructions preceding or behind G65 are all used as parameters for G65.
H02 G65 P#100 Q#101 R#102 ； Correct.
N100 G65 H01 P#100 Q10 ； Correct.

4) The input range of variable cannot exceed valid 15-digit numbers, and operation result cannot exceed 9-digit numbers and manual input range of variable is valid 8-digit numbers.

5) The result of the variable operation can be a decimal fraction with a precision of 0.0001. All operations, except H11 (OR operation), H12 (AND operation), H13 (NOT operation), H23 (ROUNDING operation) with decimal portions neglected in operation, are done without the decimal portions abnegated.
Example:
#100 ＝ 35， #101 ＝ 10， #102 ＝ 5
#110 ＝ #100÷#101 (＝3.5)
#111 ＝ #110×#102 (＝17.5)
#120 ＝ #100×#102 (＝175)
#121 ＝ #120÷#101 (＝17.5)

6) The execution time of operation and transfer instruction differs depending on different conditions. The average time is usually 10ms.

7) When the variable value is not defined, the variable becomes "vacant". The variable #0 is always vacant. It is read instead of being written.

a. Reference
When an undefined variabled is referred, the address itselft is also ignored.
Example：
When the variable #1 value is 0 and the variable #2 value is vacant, execution result of G00X#1 Y#2 is G00X0；

b. Operation
Besides using <Vacant> to assign, <Vacant> is the same with 0 in other conditions.

Table 3-9-2-6

When #1=<vacant>	When #1=0
#2=#1 ↓ #2=<空>	#2=#1 ↓ #2=0
#2=#1*5 ↓ #2=0	#2=#1*5 ↓ #2=0
#2=#1+#1 ↓ #2=0	#2=#1+#1 ↓ #2=0

c. conditional expressions
<Vacant> differs from 0 only for EQ and NE.

Table 3-9-2-7

When #1=<vacant>	When #1=0
#1 EQ #0 ↓ Established	#1 EQ #0 ↓ Not established
#1 NE #0 ↓ Not established	#1 NE #0 ↓ Established
#1 GE #0 ↓ Established	#1 GE #0 ↓ Established
#1 GT #0 ↓ Not established	#1 GT #0 ↓ Not established

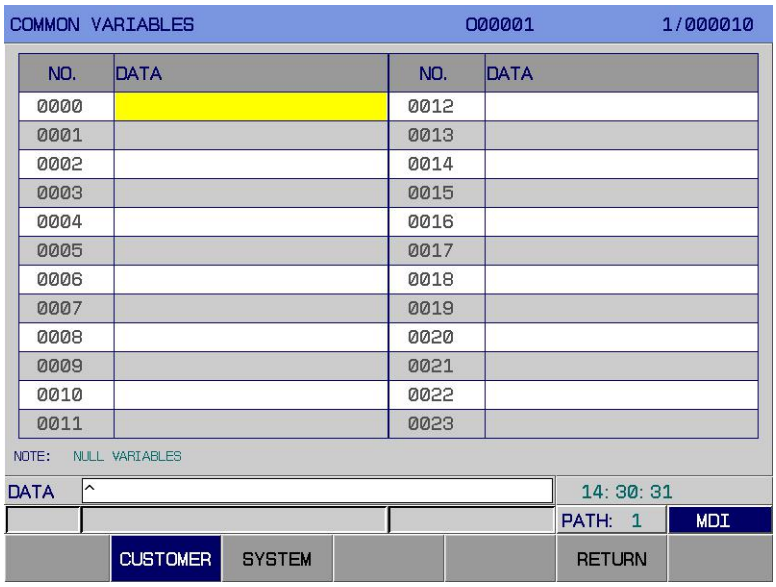


Fig. 3-9-2-2 Whne the variable value is vacant, the variable is null.

4.9.3 Custom Macro Call

When G65 is specified, the custom macro specified by address P is called, and the data is transferred to the custom macro body by arguments.

Format:

G65 P □□□□□L□□□□ < argument specification >;

Calling times

Program number of the custom macro body called

Behind G65 code, P is used to specify custom macro number, L is used to specify custom macro calling times, and the arguments are used to transfer data to custom macro.

If repetition is needed, specify the number of repeats behind L code from 1-9999; if L is omitted, the default time is 1.

If it is specified by arguments, the values will be assigned to the corresponding local variables.

Note 1: If the subprogram number specified by address P is not retrieved, an alarm (PS 078) will be issued. Note 2: No. 90000~99999 subprograms are the system reserved programs, if such subprograms are called, they can be executed, but the cursor will keep staying at block N65 and the program page displays the main program all the time. (The subprogram can be displayed by setting bit parameter No: 27#4)
Note 3: The macro program cannot be called in DNC mode. Note 4:
The macro program call can nest up to 5-level.

4.9.4 Custom Macro Function A

1. Format:

G65 Hm P#i Q#j R#k ;

m: 01～99 indicate functions of operation instruction or transfer instruction。#i:

Variable name for saving the operation result.

#j: Variable name 1 for operation, or a constant which is expressed directly without #. #k:

Variable name 2 for operation, or a constant.

Meaning: #i = #j ○ #k

└──────Operation sign, specified by Hm

(Example) P#100 Q#101 R#102……#100 = #101 ○ #102 ;

P#100 Q#101 R15 ……#100 = #101 ○ 15 ;

P#100 Q-100 R#102……#100 = -100 ○ #102

H code specified by G65 has no effect on the offset selection.

G code	H code	Function	Definition
G65	H01	Value assignment	#i = #j
G65	H02	Addition	#i = #j + #k
G65	H03	Subtraction	#i = #j - #k
G65	H04	Multiplication	#i = #j × #k
G65	H05	Division	#i = #j ÷ #k
G65	H11	Logic addition (OR)	#i = #j OR #k
G65	H12	Logic multiplication (AND)	#i = #j AND #k
G65	H13	Exclusive OR	#i = #j XOR #k
G65	H21	Square root	#i = # j
G65	H22	Absolute value	#i =| #j |
G65	H23	Complement	#i=#j-trunc(#j ÷ #k)×#k
G65	H26	Compound multiplication and division operation	#i = (#i × #j) ÷ #k
G65	H27	Compound square root	#i = # j^2 # k^2
G65	H31	Sine	#i = #j×SIN(#k)
G65	H32	Cosine	#i = #j×COS(#k)
G65	H33	Tangent	#i = #j×TAN(#k)
G65	H34	Arc tangent	#i = ATAN(#j/#k)
G65	H80	Unconditional transfer	GOTO N
G65	H81	Conditional transfer 1	IF #j = #k, GOTO N
G65	H82	Conditional transfer 2	IF #j = #k, GOTO N

G65	H83	Conditional transfer 3	IF #j ＞ #k， GOTO N
G65	H84	Conditional transfer 4	IF #j ＜ #k， GOTO N
G65	H85	Conditional transfer 5	IF #j ＞ #k， GOTO N
G65	H86	Conditional transfer 6	IF #j ＜ #k， GOTO N
G65	H99	Alarm	

Fig. 3-9-4-1

2. Operation code：

1） Variable assignment: # I = # J

G65 H01 P#I Q#J；

（e.g.）G65 H01 P#101 Q1005； （#101 = 1005）

G65 H01 P#101 Q#110； （#101 = #110） G65

H01 P#101 Q-#102； （#101 = -#102）

2） Addition：# I = # J+# K

G65 H02 P#I Q#J R#K；

（e.g.）G65 H02 P#101 Q#102 R15； （#101 = #102+15）

3） Subtraction：# I = # J- #K

G65 H03 P#I Q#J R# K；

（e.g.）G65 H03 P#101 Q#102 R#103； （#101 = #102-#103）

4） Multiplication：# I = # J×# K

G65 H04 P#I Q#J R#K；

（e.g.）G65 H04 P#101 Q#102 R#103； （#101 = #102×#103）

5） Division：# I = # J÷# K

G65 H05 P#I Q#J R#K；

（e.g.）G65 H05 P#101 Q#102 R#103； （#101 = #102÷#103）

6） Logic addition（OR）：# I = # J.OR. # K

G65 H11 P#I Q#J R#K；

（e.g.）G65 H11 P#101 Q#102 R#103； （#101 = #102.OR. #103）

7） Logic multiplication (AND)：# I = # J.AND. # K

G65 H12 P#I Q#J R#K；

（e.g.）G65 H12 P# 101 Q#102 R#103； （#101 = #102.AND.#103）

8） Exclusive OR：# I = # J.XOR. # K

G65 H13 P#I Q#J R#K；

（e.g.）G65 H13 P#101 Q#102 R#103； （#101 = #102.XOR. #103）

9） Square root：# I = $\sqrt{\#j}$

G65 H21 P#I Q#J；

（e.g.）G65 H21 P#101 Q#102 ； （#101= $\sqrt{\#102}$ ）

10） Absolute value：# I = | # J |

G65 H22 P#I Q#J ；

（e.g.）G65 H22 P#101 Q#102 ；（#101 = | #102 |）

11） Complement：# I = # J－TRUNC(#J/#K)×# K，TRUNC：abandon the decimal portion.

G65 H23 P#I Q#J R#K；

（e.g.）G65 H23 P#101 Q#102 R#103；（#101 = #102- TRUNC (#102/#103)×#103）

12） Compound multiplication and division operation：# I =（# I×# J）÷# K

G65 H26 P#I Q#J R# k；

（e.g.）G65 H26 P#101 Q#102 R#103；（#101 =（#101×# 102）÷#103）

13) Compound square root: # I = **G65 H27 P#I** $\sqrt{\#j^2+\#k^2}$ **Q#J R#K;**

(e.g.) G65 H27 P#101 Q#102 R#103; (#101 = #10$\sqrt{2^2+}$ #103^2

14） Sine：# I = # J•SIN（# K）（Unit：°）

G65 H31 P#I Q#J R#K；

（e.g.）G65 H31 P#101 Q#102 R#103；（#101 = #102•SIN（#103））

15） Cosine：# I = # J•COS（# K）（Unit：°）

G65 H32 P#I Q#J R# K；

（e.g.）G65 H32 P#101 Q#102 R#103；（#101 =#102•COS（#103））

16） Tangent：# I = # J•TAN（# K）（Unit：°）

G65 H33 P#I Q#J R# K；

（e.g.）G65 H33 P#101 Q#102 R#103；（#101 = #102•TAN（#103））

17） Arc tangent：# I = ATAN（# J /# K）（Unit：°）

G65 H34 P#I Q#J R# K；

（e.g.）G65 H34 P#101 Q#102 R#103；（#101 =ATAN（#102/#103））

Note 1: The unit of angular variable is degree.
Note 2: If the required Q and R are not specified in operations above, their values are 0 by default. Note 3: trunc: rounding operation, the decimal portion is abandoned.

3. Transfer command

1） Unconditional transfer

G65 H80 Pn； n：Sequence number

（e.g.）G65 H80 P120；（Go to block N120）

2） Conditional transfer 1 #J.EQ.# K (＝)

G65 H81 Pn Q#J R# K； n：Sequence number

(e.g.) G65 H81 P1000 Q#101 R#102；

When # 101 = #102, it goes to block N1000; when #101 ≠ #102, the program is executed in sequence.

3） Conditional transfer 2 #J.NE.# K (≠)

G65 H82 Pn Q#J R# K； n：Sequence number

(e.g.) G65 H82 P1000 Q#101 R#102；

When # 101 ≠ #102, it goes to block N1000; when #101 = #102, the program is executed in sequence.

4） Conditional transfer 3 #J.GT.# K (＞)

G65 H83 Pn Q#J R# K;　　n：Sequence number

(e.g.) G65 H83 P1000 Q#101 R#102;

When #101 ＞ #102, it goes to block N1000; when #101 ≤ #102, the program is executed in sequence.

5） Conditional transfer 4#J.LT.# K (＜)

G65 H84 Pn Q#J R# K;　　n：Sequence number

(e.g.) G65 H84 P1000 Q#101 R#102;

When # 101＜#102, it goes to block N1000; when #101 ≥ #102, the program is executed in sequence.

6） Conditional transfer 5 #J.GE.# K (≥)

G65 H85 Pn Q#J R# K;　　n：Sequence number

(e.g.) G65 H85 P1000 Q#101 R#102;

When # 101 ≥ #102, it goes to block N1000; when #101＜#102, the program is executed in sequence.

7） Conditional transfer 6 #J.LE. # K (≤)

G65 H86 Pn Q#J R# K;　　n：Sequence number

(e.g.) G65 H86 P1000 Q#101 R#102;

When # 101≤ #102, it goes to N1000; when #101＞#102, the program is executed in sequence.

Note: The sequence number can be specified by variables. Such as G65 H81 P#100 Q#101 R#102；if the conditions are satisfied, it goes to the block of which the number is specified by #100.

4. Logic AND, logic OR and logic NOT codes

Example：

G65 H01 P#101 Q3;

G65 H01 P#102 Q5;

G65 H11 P#100 Q#101 Q#102;

The binary expression for 5 is 101, for 3 is 011, and the operation result is #100=7; G65 H12 P#100 Q#101 Q#102;

The binary expression for 5 is 101, for 3 is 011, and the operation result is #100=1.

5. Macro variable alarm

Example：

G65 H99 P1;　　Macro variable 3001 alarm G65

H99 P124;　　Macro variable 3124 alarm

Example for custom macro

1. Bolt hole cycle

To drill N equal-spaced holes on the circumference of the circle whose center is the reference point (X0，Y0) and radius is R, with an intial angle (A).

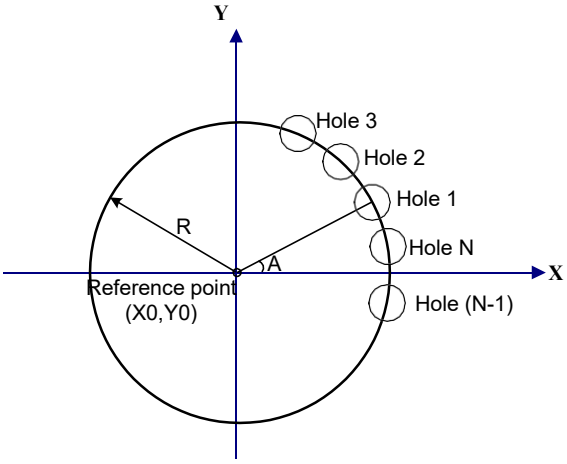


Fig. 3-9-4-2

X0，Y0 is the coordinates of the reference point in bolt hole cycle.

R：Radius, A: Initial angle, N: Number. Parameters above use the following variables: #500： X coordinate value of the reference point (X0)

#501： Y coordinate value of the reference point (Y0)

#502： Radius (R)

#503： Initial angle (A)

#504： N numbers

If N＞0, the rotation is CCW, and the number is N. If N ＜0, the rotation is CW, and the number is N.

The variables below are used for the operation in macro.

#100： For the counting of the hole I machining (I) #101：
The final value of the counting(＝ | N |)(IE)

#102： The angle of hole I (θI)

#103： X coordinate of hole I (Xi)

#104： Y coordinate of hole I (Yi)

The custom macro body can be programmed as follows:

```
O9010;
N100 G65 H01 P#100 Q0;                I=0
G65 H22 P#101 Q#504;                  IE=|N|
N200 G65 H04 P#102 Q#100 R360;
G65 H05 P#102 Q#102 R#504;            θI=A+360°×I/N
G65 H02 P#102 Q#503 R#102;
G65 H32 P#103 Q#502 R#102;            X I=XI+R·COS(θI)
G65 H02 P#103 Q#500 R#103;
G65 H31 P#104 Q#502 R#102;            Y I=Y I+R·SIN(θI)
G65 H02 P#104 Q#501 R#104;
G90 G00 X#103 Y#104;                  Positioning of hole I
G**;                                  Hole machining G code
```

G65 H02 P#100 Q#100 R1； I=I＋1

G65 H84 P200 Q#100 R#101； When I＜IE, go to block N 200, drill IE holes.

M99；

Example for a program calling the above custom macro body is as follows:

O0010；

G65 H01 P#500 Q100； X0=100MM

G65 H01 P#501 Q-200； Y0=-200MM

G65 H01 P#502 Q100； R=100MM

G65 H01 P#503 Q20； A=20°

G65 H01 P#504 Q12； N=12 in CCW rotation

G92 X0 Y0 Z0；

M98 P9010； Calling the custom macro

G80；

X0 Y0；

M30；

4.9.5 Custom Macro Function B

1. Arithmetic and logic operation

The operations listed in the following table can be executed on variables. The expressions on the right of the operation characters can contain constants and/or variables constituted by functions or operation characters. The variables #j and #k in the expression can be replaced by constants. The values of the variables on the left can also be assigned by an expression.

Table 3-9-5-1 Arithmetic and logic operation

Function	Format	Remarks
Definition	#i = #j	
Addition Subtraction Multiplication Division	#i = #j + #k; #i = #j - #k; #i = #j * #k; #i = #j / #k;	
Sine Arcsine Cosine Arc cosine Tangent Arc tangent	#i = SIN[#j]; #i = ASIN[#j]; #i = COS[#j]; #i = ACOS[#j]; #i = TAN[#j]; #i = ATAN[#j] / [#k];	The angle is specified by degree. 90°30’ indicates an angle of 90.5°.
Square root Absolute value Rounding-off Rounding up to an integer Rounding down to an integer Natural logarithm Exponential function	#i = SQRT[#j]; #i = ABS[#j]; #i = ROUND[#j]; #i = FUP[#j]; #i = FIX [#j j]; #i = LN[#j]; #i = EXP[#j];	
OR Exclusive OR AND	#i = #j OR #k; #i = #j XOR #k; #i = #j AND #k;	Logic operation is executed by the binary system.
BCD to BIN Bin to BCD	#i = BIN[#j]; #i = BCD[#j];	Used for switching with PMC signal

Explanation：

（1） **Angle unit**

The angle unit of functions SIN，COS，ASIN，ACOS，TAN and ATAN is degree, e.g., 90°30′ indicates an angle of 90.5°.

（2） **ARCSIN #i = ASIN [#j]**

Ranging from -90° to 90°.
When #j is beyond the range from -1 to 1, an alarm occurs. The constant can replace the variable #j.

（3） **ARCCOS #i = ACOS [#j]**

Ranging from 180° to 0°。
When #j is beyond the range from -1 to 1, an alarm occurs.
Variable #j can be replaced by constants.

（4） **ARCTAN #i = ATAN [#j] / [#k]**

Specify the lengths of two sides, separated by a slash（/）. Ranging from 0° to 360°.
[Example] When #1 = ATAN [-1] / [-1]; is executed, #1=225°.
The constant can replace the variable #j.

（5） **Natural logarithm #i = LN [#j]**

When antilog (# j) is 0 or smaller, an alarm occurs. The constant can replace the variable #j.

（6） **Exponential function #i = EXP [#j]**

When the operation result exceeds 99997.453535 (j is about 11.5129), an overflow occurs and an alarm is issued.
The constant can replace the variable #j.

（7） **ROUND（rounding-off）function**

The round function rounds off at the first decimal place.
Example:
When #1=ROUND[#2]; is executed where #2 holds 1.2345, the value of variable #1 is 1.0.

（8） **Rounding up and down to a integer**

When the value operation is processed by CNC, if the absolute value of the integer produced by an operation on a number is greater than the absolute value of the original number, such an operation is referred to as rounding up to an integer. If the absolute value of the integer produced by an operation on a number is smaller than the absolute value of the original number, such an operation is referred to as rounding down to an integer. Please be careful when handling negative numbers.

Example：
Suppose that #1=1.2，#2=－1.2。
When #3=FUP[#1] is executed, 2.0 is assigned to #3. When #3=FIX[#1] is executed, 1.0 is assigned to #3.

When #3=FUP[#2] is executed, -2.0 is assigned to #3. When #3=FIX[#2] is executed, -1.0 is assigned to #3.

（9） The abbreviations of the arithmetic and logic instructions.

When a function is specified in a program, the first two characters of the function name can be used to specify the function. (See table 3-9-5-1)

Example:

ROUND→RO

FIX→FI

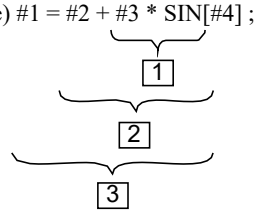
（10） Operation sequence

① Function

②Multiplication and division operation（* / AND）

③Addition and subtraction operation（+ - OR XOR）

Example) #1 = #2 + #3 * SIN[#4] ;



1 , 2 and 3 indicate the operation sequence.

（11） Restrictions

Brackets [,] are used to enclose an expression.

When a divisor of 0 is specified in a division or TAN[90], an alarm occurs.

2. Transfer and loop

1） Transfer and loop

In the program, GOTO statement and IF statement are used to change the control flow. There are three types of transfer and loop operations:

Transfer and cycle
- GOTO statement（unconditional transfer）
- IF statement（conditional transfer： IF…THEN…）
- WHILE statement（When…loop is performed）

2） Unconditional transfer

➢ GOTO statement

Transfer to the block with sequence number n. The sequence number can be specified by an expression.

GOTOn; n: Sequence number（1 to 99999）

Example:

GOTO 1；

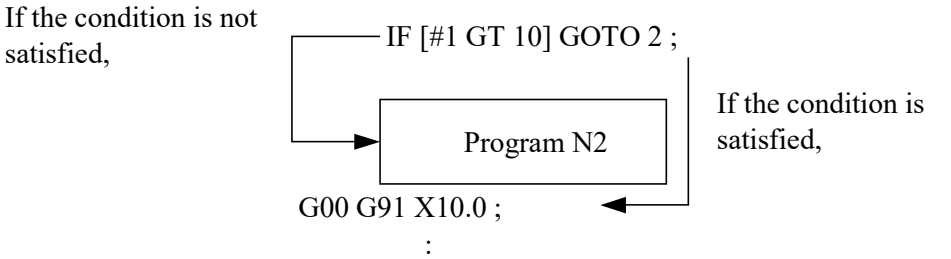
GOTO #10；

3）Conditional transfer (IF statement) [<conditional expression>]

IF[<conditional expression >]GOTO n

If the specified conditional expression is satisfied, the system transfers to the block with sequence number n; if the specified conditional expression is not satisfied, the next block is executed.

If the value of a variable is greater than 10, the system transfers to the block with sequence number N2.



IF[<conditional expression >]THEN

If the conditional expression is satisfied, a predetermined macro statement is executed. Only a single macro statement is executed.

If the values of #1 and #2 are the same，0 is assigned to #3.
IF[#1 EQ #2] THEN #3=0;

Explanation:

- Conditional expression

A conditional expression must include an operator, which is inserted between two variables or between a variable and a constant, and must be enclosed with brackets （[，]）. An expression can replace a variable.

- Operator

Operators each consists of two letters are used to compare two values to determine whether they are equal or one is greater or smaller than the other one.

Table 3-9-5-2 Operators

Operator	Meaning
EQ	Equal to （=）
NE	Not equal to ≠）
GT	Greater than （>）
GE	Greater than or equal to （≥）
LT	Smaller than （<）
LE	Smaller than or equal to （≤）

- Typical program

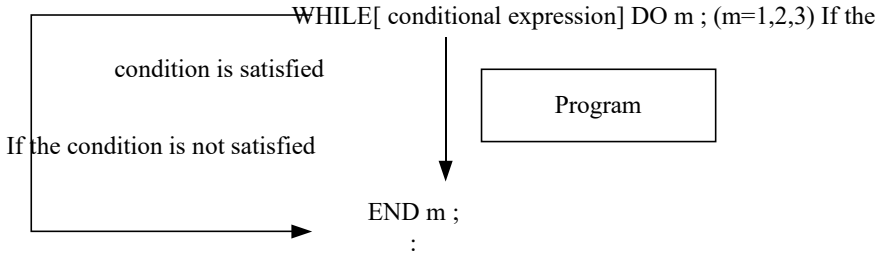
The program below calculates the sum of numerical value 1 to 10.

```
O9500；
#1=0；                    Initial value of the variable to hold the
#2=1；                    sum Initial value of the variable as an
                          addend
     N1 IF[#1 GE 10]GOTO 2；  Transfers to N2 when the addend is greater
                          than or equal to 10
#1=#1+#2；                GOTO 1； N2 M30；
#1=#2+1     ；
```

C	ulation to find the sum
a	The next addend
l	Traverse to N1
c	Program end

4） Loop (WHILE statement)

Specify a conditional expression behind WHILE, when the specified condition is satisfied, the program from DO to END is executed, otherwise, program execution proceeds to the block after END.



When the specified condition is satisfied, the program from DO to END is executed. Otherwise, program execution proceeds to the block after END. This kind of instruction format is applicable to IF statement. A number after DO and a number after END are the identification numbers for specifying the range of execution. The identification numbers are 1, 2 and 3. If numbers other than 1, 2 and 3 are used, an alarm occurs.

Explanation:

- Nesting

The identification numbers (1 to 3) in the loop from D0 to END can be used repeatedly as required. However, when a program includes crossing repetition loop (overlapped DO ranges), an alarm occurs.

1. The identification numbers (1 to 3) can be used as many times as required.

```
WHILE[ … ] DO 1;
    [ Program ]
END 1 ;
:
WHILE[ … ] DO 1;
    [ Processing ]
END 1 ;
:
```

2. The ranges of DO cannot overlap

```
WHILE[ … ] DO 1;
    [ Program ]
WHILE[ …] DO 2;
  : END
1 ;
    [ Program ]
END 2 ;
```

3. DO loops can be nested to 3 levels

```
WHILE[ … ] DO 1;
  :
  WHILE[ … ] DO 2;
    :
    WHILE[ … ] DO 3;
      [ Program ]
    END 3 ;
  END 2 ;
END 1 ;
```

4. The control can be transferred to the outside of a loop.

```
WHILE[ … ] DO 1;
IF[…] GOTO n ;
END 1 ;
Nn
```

5. Transfer cannot enter the loop area.

```
IF […] GOTO n ;
:
WHILE […] DO 1 ;
Nn… ;
END 1 ;
```

Explanation:

- Infinite loop

  When DO is specified without specifying WHILE statement, an infinite loop from DO to END is produced.

- Processing time

  When a transfer to a sequence number in GOTO statement occurs, the sequence number is searched for. Processing in the reverse direction is longer than the one in the forward direction. The processing time can be reduced by using WHILE statement for repetition.

- Undefined variables

In the conditional expression using EQ or NE, <vacant> and zero have different affects. In the other conditional expressions, <vacant> is taken as 0.

- Typical program

The program below calculates the sum of numbers 1 to 10.

```
O0001 ;
#1=0;
#2=1;
WHILE [#2 LE 10] DO 1;
#1=#1+#2;
#2=#2+1;
END 1;
M30;
```

Notes:

- When a macro program is called by G65, and M, S, T, D and F are used for transferring variables, only positive integers can be transferred.
- The line number N code cannot be in the same line with WHILE/DO/END, or the loop is ineffective.
- Loop and skip instructions cannot be used in DNC mode.
- A GOTO statement starts searching at the beginning of the program and skips when the first corresponding line number is retrieved. Try not to use the same N code in one program.
- When the variable number is expressed by a decimal fraction, the system will remove the decimal part with carry ignored.
- The values of local variables are retained before the main program ends. They are common to each subprogram.

Example 2:

O0002;

G0 X50 Z5; (rapid traverse to X50 Z5)

G04 X4; (dwell 4 seconds)

G04 X5; (dwell 5 seconds again, G04 is non-modal and is needed to input again) M30;

Example 3 (the first run after power-on) : O0003;

G98 F500 G01 X100 Z100; (Feedrate per minute 500mm/min in G98)

G92 X50 W-20 F2 ; (F value is a pitch and must be input in thread cutting) G99

G01 U10 F0.01 (Feedrate per revolution in G99 must be input again) G00

X80 Z50 M30;

3.1.3 Related definitions

In the user manual, the definitions of Word are as follows except for the especial explanations: Starting point: position before the current block runs;

End point: position after the current block ends; X:

X absolute coordinates of end point;

U: different value of absolute coordinates between starting point and end point; Z: Z absolute coordinates of end point;

W: different value of absolute coordinates between starting point and end point; F: cutting feedrate.

3.2 Rapid traverse movement G00

Command format: G00 X(U) Z(W) ;

Command function: X, Z rapidly traverses at the respective traverse speed to the end points from their starting point. G00 is initial command as Fig.3-1.

X, Z traverses at the respective traverse speed, the short axis arrives the end point and the length axis continuously moves to the end point and the compound path may be not linear.

Command specification: G00 is initial mode;

X, U, Z, W range: ±99999999× least input increment ;

Can omit one or all command addresses X(U), Z(W). The coordinate values of starting point and end point are the same when omitting one command address; the end point and the starting point are in the same position when all are omitted. X, Z are valid, and U, W are invalid when X, U, Z and W are in the same one block.

Command path:

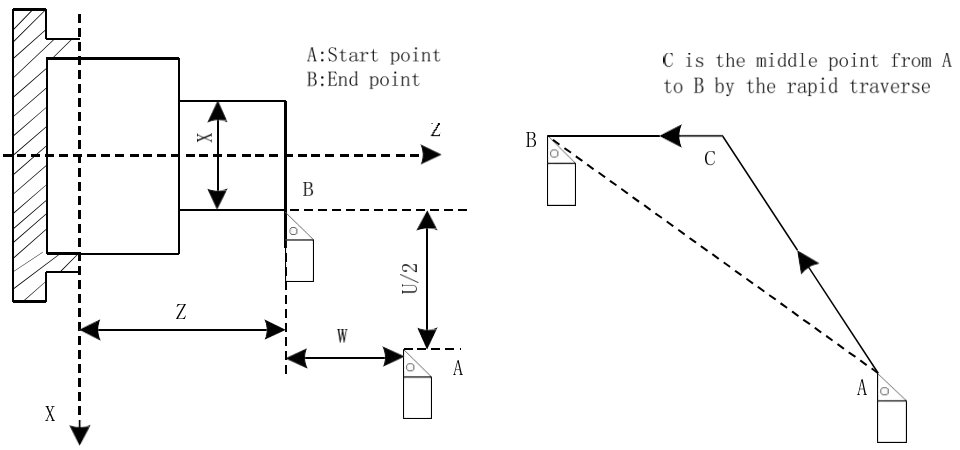


Fig. 3-1

The respective rapid traverse speed of X, Z is defined by the system parameter No.022, No.023, and their traverse speed can changed by rapid override key on the machine control panel.

Example: The tool rapidly traverses to B from A as Fig. 3-2.

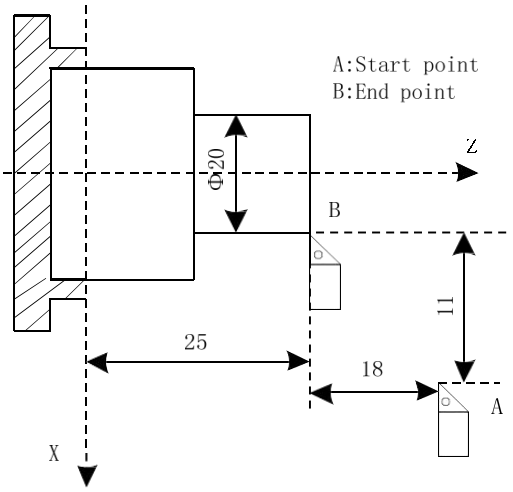


Fig. 3-2

G0 X20 Z25; (absolute programming) G0
U-22 W-18; (incremental programming)
G0 X20 W-18; (compound programming)
G0 U-22 Z25; (compound programming)

3.3 Linear interpolation G01

Command format: G01 X(U) _ Z(W) _ F_;

Command function: The movement path is a straight line from starting point to end point as Fig.3-3.

Command specification: G01 is modal.

Can omit one or all command addresses X (U), Z (W). The coordinate

values of starting point and end point are the same when omitting one command address; the end point and the starting point are in the same position when all are omitted.

F command value is the vector compound speed of X and Z instantaneous speed and the actual cutting feedrate is the product between the feedrate override and F command value.

After F command value is executed, it has been reserved unless the new one is executed. Do not repeat it when the following G commands adopt functions of F word. Its range is referred to Table 1-10.

Note: In G98, F max. value cannot exceed the value set by the data parameter No.027, otherwise, the system alarms.

Command path:

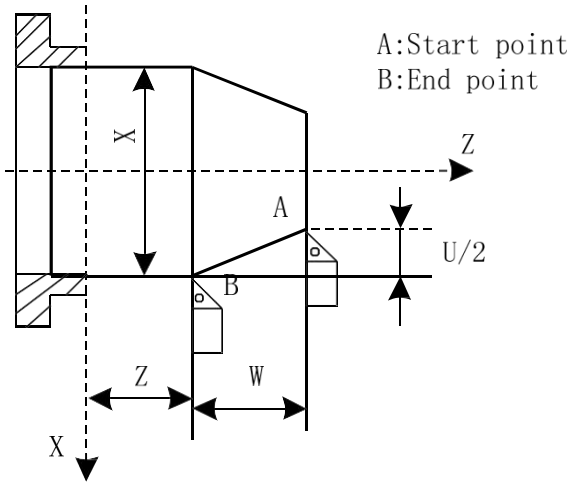


Fig. 3-3

Example: Cutting path from Φ40 to Φ60 as Fig.3-4:

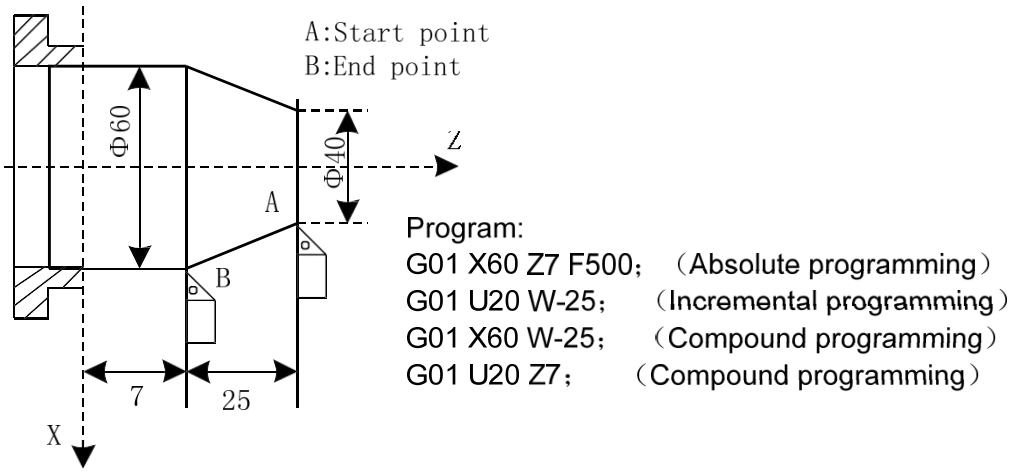


Fig. 3-4

3.4 Circular interpolation G02, G03

Command format:

G02 } X(U)___Z(W) ____ { R___
G03 } I__K___

Command function:

G02 movement path is clockwise (rear tool post coordinate system)/counterclockwise (fronttool post coordinate system) arc from starting point to end point as Fig. 3-5(a).

G03 movement path is counterclockwise (rear tool post coordinate system/clockwise(front tool post coordinate system) arc from starting point to end point as Fig. 3-5(b).

Command path:

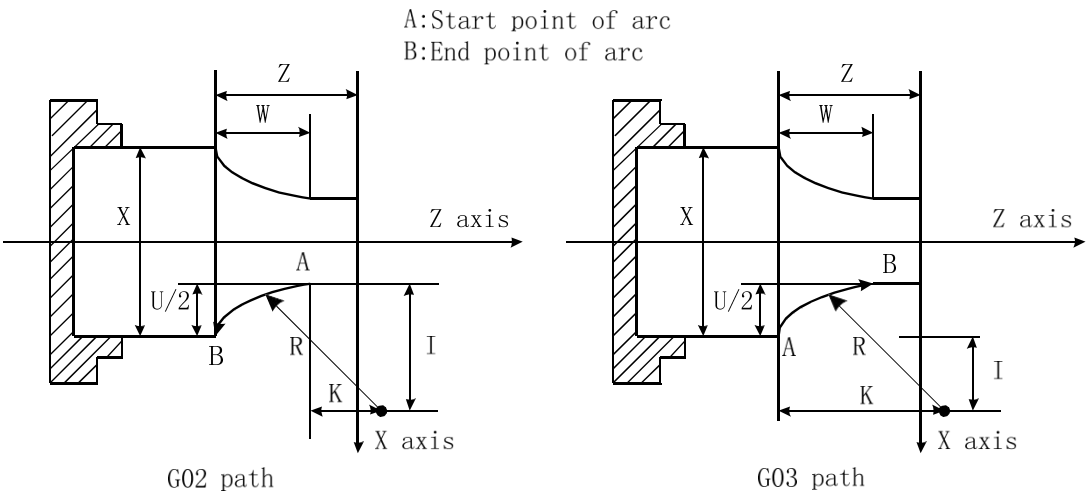


Fig. 3-5 G02 and G03 path

Command specification:

G02, G03 are modal,

R is arc radius, range: ±99999999× least input increment;

I: X difference value between circle center and starting point of arc in radius; K: Z difference value between circle center and starting point of arc;

Center point of arc is specified by address I, K which separately corresponds to X, Z, I, K expresses the vector (it is the increment value) from starting point to center point of arc as the following figure;

I=Coordinates of center point-that of starting point in X direction; K= Coordinates of center point-that of starting point in Z direction;

I, K are with sign symbol. When directions of I, K are the same as those of X, Z, they are positive, otherwise, they are negative.

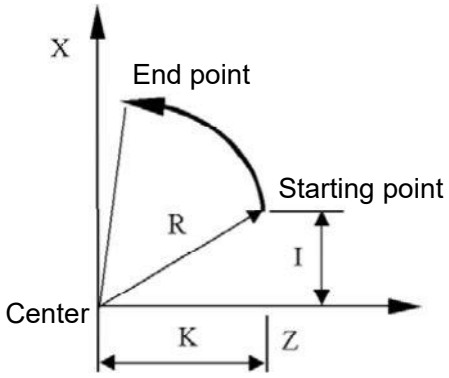
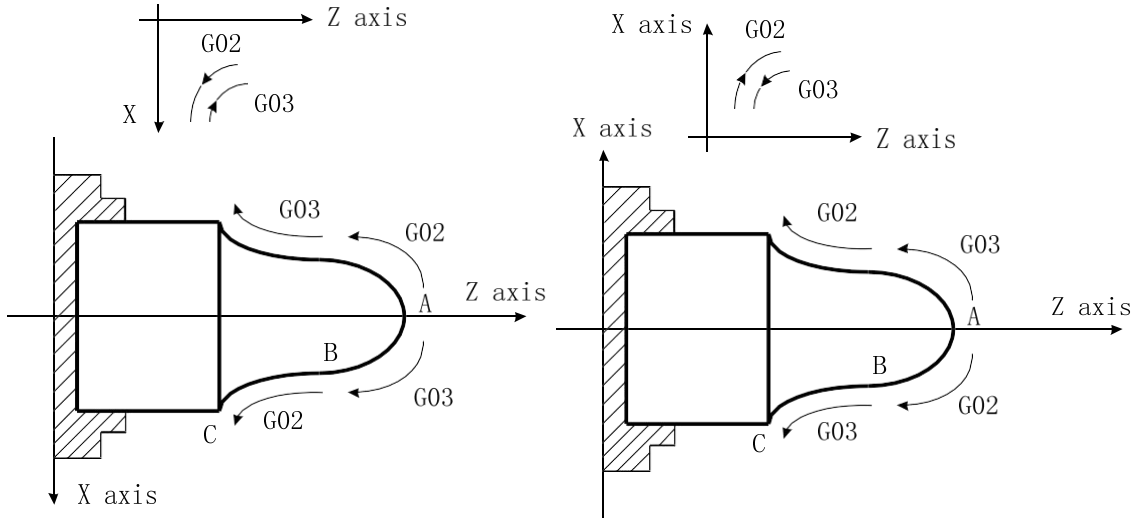


Fig. 3-6

Arc direction: G02/G03 direction (clockwise/counterclockwise) is opposite on the front tool post coordinate system and the rear one as Fig.3-7:



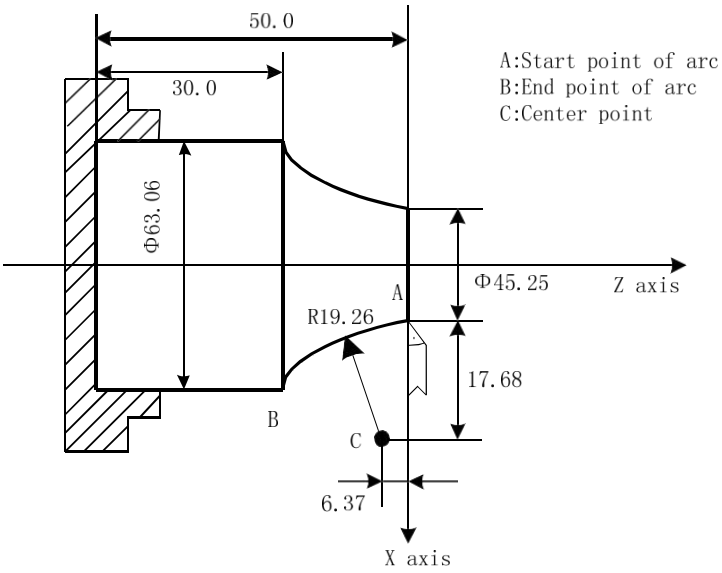
*

Fig. 3-7

Notes:

- When I = 0 or K = 0, they can be omitted; one of I, K or R must be input, otherwise the system alarms.
- R is valid and I, K are invalid when they are input at the same time.
- R value must be equal to or more than half distance from starting point to end point, and the system alarms if the end point is not on the arc defined by R command;
- Omit all or one of X(U), Z(W); coordinates of starting point and end point of this axis are the same when omitting ones, the path is a full circle(360°) in G02/G03 when center point are specified by I,K; the path is 0(0°) when center point is specified by R.
- R should be used for programming. The system executes in R= $\sqrt{I^2+K^2}$ to ensure starting point and end point of arc path are the specified ones in I, K programming.
- When the distance from center point to end point is not equal to R(R = $\sqrt{I^2+K^2}$) in I,K programming, the system automatically adjusts position of center point to ensure starting point and end point of arc path are the specified ones; when the distance from center point to end point is more than 2R, and the system alarms.
- Arc is less than 360° when R is commanded, the arc is more than 180° when R is negative, and it is less than or equal to 180° when R is positive.

Example: Arc cutting path from Φ45.25 to Φ63.06 shown in Fig. 3-8.



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 I17.68 K-6.37 ； or
G02 U17.81 W-20.0 I17.68 K-6.37 F300

Fig. 3-8

Compound programming in G02/G03:

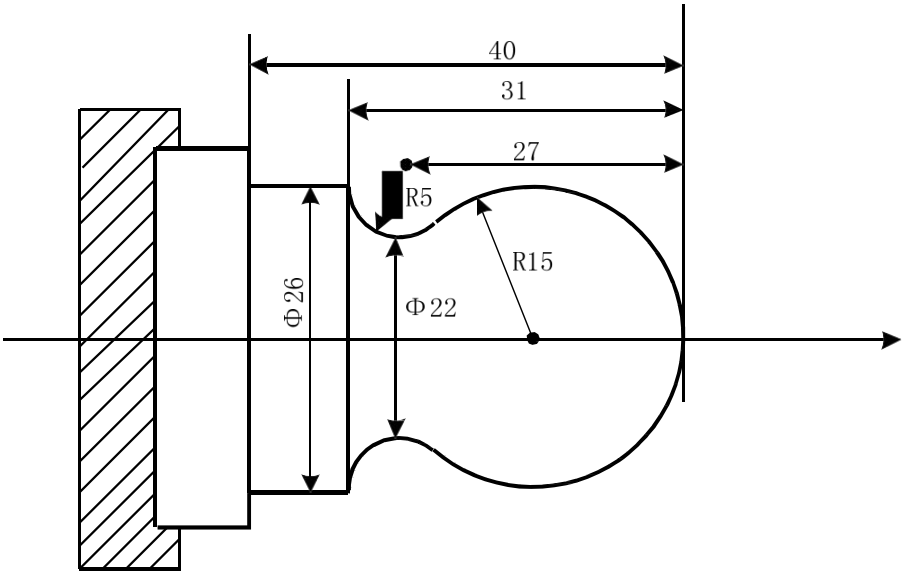


Fig. 3-9 Circular programming example

Program: O0001

N001 G0 X40 Z5;	(Rapidly traverse)
N002 M03 S200;	(Start spindle)
N003 G01 X0 Z0 F900;	(Approach workpiece)
N005 G03 U24 W-24 R15;	(Cut R15 arc)
N006 G02 X26 Z-31 R5;	(Cut R5 arc)
N007 G01 Z-40;	(Cut φ26)
N008 X40 Z5;	(Return to starting point)
N009 M30;	(End of program)

3.5 Plane selection G17 ~ G19

Command format：

G17……XY plane
G18……ZX plane
G19……YZ plane

Command function: use G commands to select the plane of the arc interpolation or the one of the cutter compensation

Command explanation: G17, G18, G19 are modal, and the plane does not change in the block without the command.

Notes:

- Firstly set the basic axis Y when the system selects G17, G19 plane;
- Cannot switch the planes in C tool compensation;
- G71~G76，G90，G92，G94 can be used in G18 plane;
- The plane selection code can be in the same block with G codes in the other groups;
- The movement command is not related to the plane selection;
- Diameter or radius programming: currently, because there is only one bit parameter No 1.2 to select the diameter or the radius programming and is valid to only X axis, Z and Y axis use the only radius programming in G2, G3, and X axis is selected by the parameter;
- The tool nose direction of C tool compensation is 0 in G17, G19.

3.6 Chamfering function

Chamfering function is to insert one straight line or circular between two contours to make the tool smoothly transmit from one contour to another one. TAC2000 uses the linear and circular chamfering functions.

3.6.1 Linear chamfering

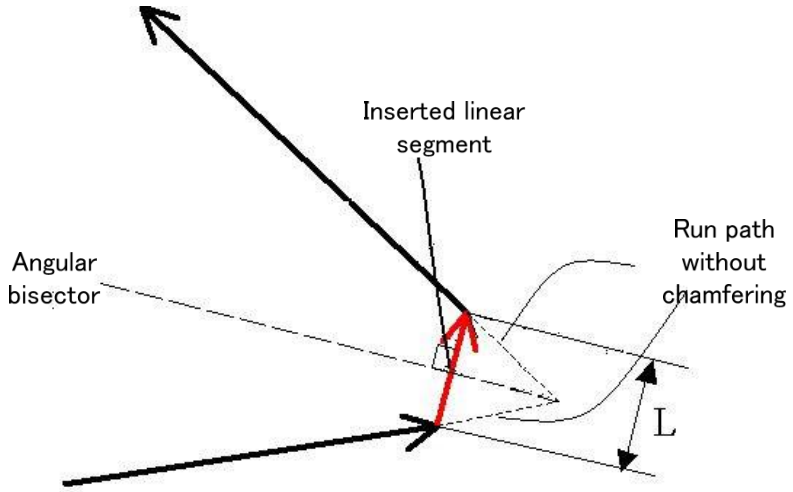
Linear chamfering: insert one straight line in the linear contours, arc contours, linear contour and arc contour. The command address of linear chamfering is L, behind which data is the length of chamfering straight line. The linear chamfering must be used in G01, G02 or G03 command.

A. Linear to linear

Command format: G01 X(U)_ Z(W)_ L_ ;
G01 X(U)_ Z(W)_ ;

Command function: insert one straight line between two linear interpolation blocks

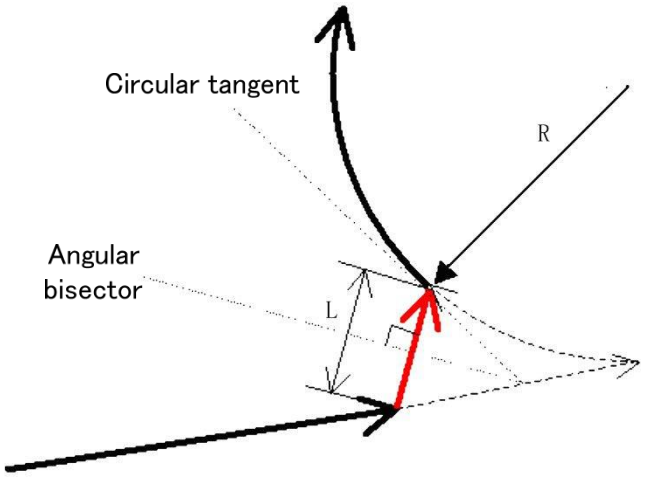
.



B. Linear to circular

Command format: G01 X(U)_ Z(W)_ L_;
G02/G03 X(U)_ Z(W)_ R_;
Or
G01 X(U)_ Z(W)_ L_; G02/G03 X(U)_ Z(W)_ I_ K_;

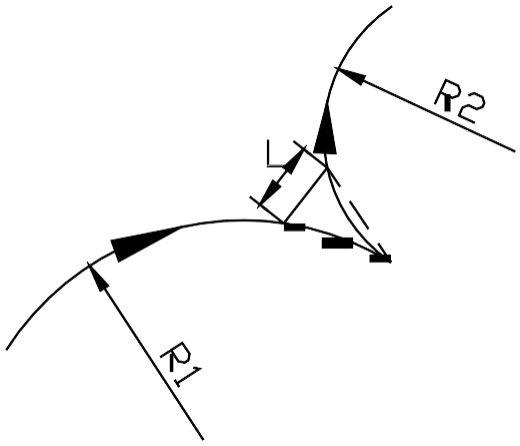
Command function: insert one straight line between the linear and circular interpolation blocks.



C. Circular to circular

Command format: G02/G03 X(U)_ Z(W)_ R_ L_;
G02/G03 X(U)_ Z(W)_ R_;
Or
G02/G03 X(U)_ Z(W)_ I_ K_ L_;
G02/G03 X(U)_ Z(W)_ I_ K_;

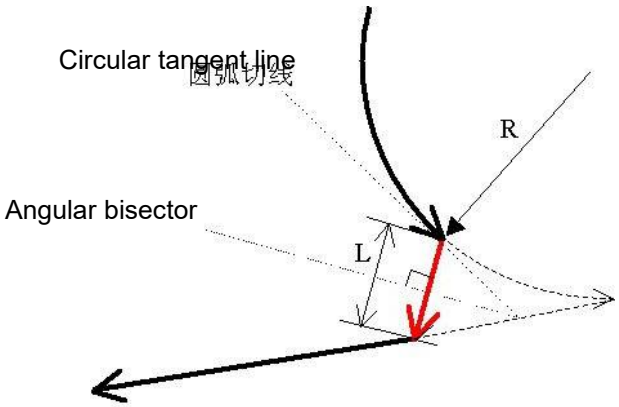
Command function: insert one straight line between two circular interpolation blocks.



D. Circular to linear

Command format: G02/G03 X(U)_ Z(W)_ R_ L_;
G01 X(U)_ Z(W)_;
Or
G02/G03 X(U)_ Z(W)_ I_ K_ L_;
G01 X(U)_ Z(W)_;

Command function: insert one straight line block between circular and linear interpolation block.



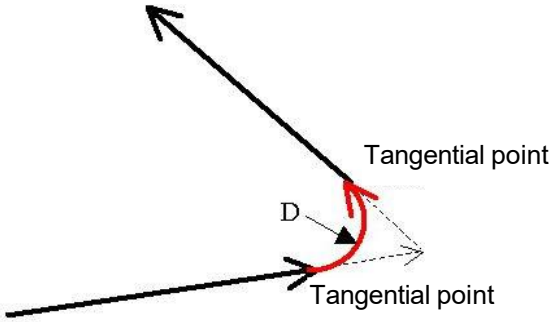
3.6.2 Circular chamfering

Circular chamfering: insert one circular between linear contours, circular contours, linear contour and circular contour, the circular and the contour line are transited by the tangent. The command of circular chamfering is D, and the data behind the command is the radius of chamfering circular. The circular chamfering must be used in G01, G02 or G03.

A. Linear to linear

Command format: G01 X(U)_ Z(W)_ D_;
G01 X(U)_ Z(W)_;

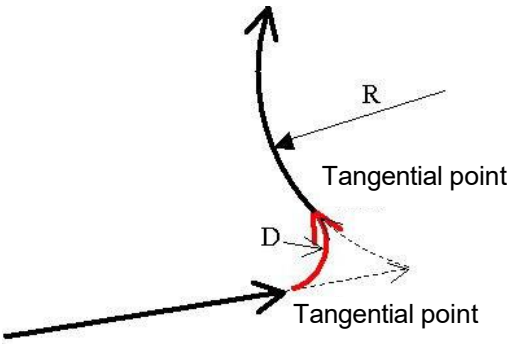
Command function: insert one circular between two straight lines, the inserted circular block and two straight lines are tangent, the radius is the data behind the command address D.



B. Linear to circular

Command format: G01 X(U)_ Z(W)_ D_;
G02/G03 X(U)_ Z(W)_ R_;
or
G01 X(U)_ Z(W)_ D_; G02/G03
X(U)_ Z(W)_ I_ K_;

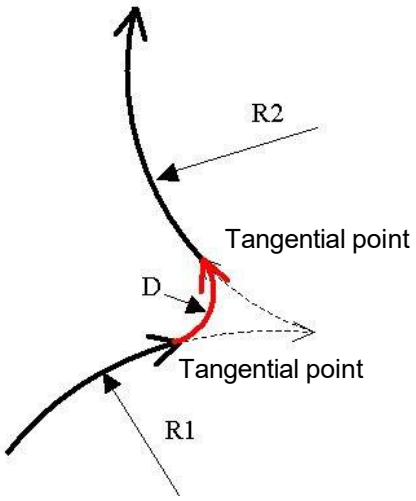
Command function: insert one circular between linear and circular, the inserted circular is tangent to the linear and the circular, and the radius is the data behind the command address D.



C. Circular to circular

Command format: G02/G03 X(U)_ Z(W)_ R_ D_;
G02/G03 X(U)_ Z(W)_ R_;
or
G02/G03 X(U)_ Z(W)_ R_ D_;
G02/G03 X(U)_ Z(W)_ I_ K_;
or
G02/G03 X(U)_ Z(W)_ I_ K_ D_;
G02/G03 X(U)_ Z(W)_ I_ K_;
or
G02/G03 X(U)_ Z(W)_ I_ K_ D_;
G02/G03 X(U)_ Z(W)_ R_;

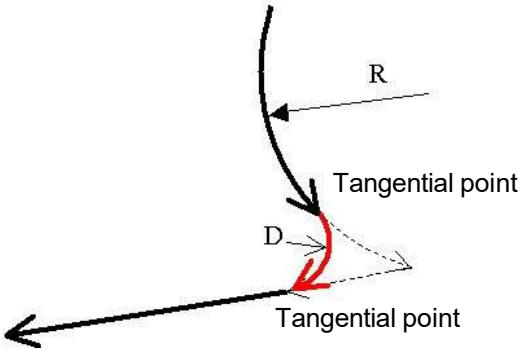
Command function: insert one circular between two circular blocks, the inserted circular is tangent to the two circular blocks, and the radius is the data behind the command address D.



D. Circular to linear

Command format: G02/G03 X(U)_ Z(W)_ R_ D_;
G01 X(U)_ Z(W)_;
Or
G02/G03 X(U)_ Z(W)_ I_ K_ D_;
G01 X(U)_ Z(W)_;

Command function: insert one circular block between the circular and the linear, the inserted circular block is tangent to the circular and the linear, and the radius is the data behind the command address D.



3.6.3 Special cases

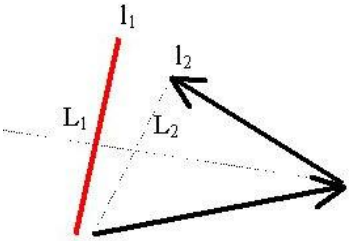
The chamfering function is invalid or alarms as follows:

1) Linear chamfering

A. The chamfering function is invalid when two interpolation straight lines are in the same linear.

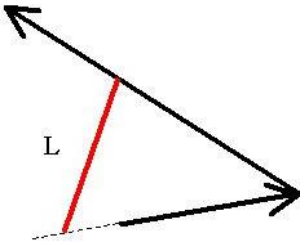
B. CNC alarms when the chamfering linear is too long.

L1 i is the chamfering linear, and the length is L_1; l_2 is the third edge of the triangle which is formed by two interpolation straight lines, the length is L_2, CNC alarms when L_1 is bigger than L_2 as follows:



C. Some linear block is too short

The chamfering linear length is L, CNC alarms when other end of the caculated chamfering linear is not in the interpolation linear(in the extension line of the interpolation linear).

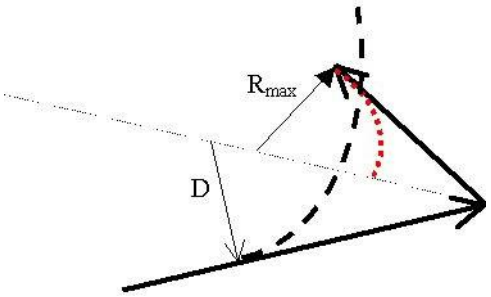


2) Circular chamfering

A. The circular chamfering function is invalid when two interpolation straight lines are in the same block.

B. CNC alarms when the chamfering circular radius is too big.

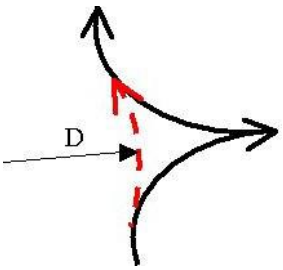
CNC alarms when the chamfering circular radius is D, max. circular radius of the tangential linear lines is R_{max} which is less than D as follows.



C. The circular chamfering function is invalid when the linear and the circular, or the circular and the linear are tangential.

D. The circular chamfering function is invalid when one circular and another one are tangential.

The circular chamfering function is valid when the circular tangency is as follows:



3.7 Dwell G04

Command format: G04 P_; or
G04 X_ ; or G04
U_; or G04;

Command function: each axis stops the motion, the modal of G commands and the reserved data, state are not changed, and execute the next block after dwelling the defined time.

Command specification: G04 is non-modal.
G04 dwell time is defined by the word P__, X__or U___. P range is 0 ~ 99999 （unit: ms）.

X, U range is 0 ~ 9999.999 x least input unit (unit: s)

Notes:

- The system exactly stop a block when P, X, U are not input
- P, X, U can not be in the same block;

3.8 Machine Zero function

3.8.1 Machine 1st reference point G28

Command format: G28 X/U Z/W ;

Command function: the tool rapid traverses to the middle point defined by X/U 、 Z/W from starting point and then return to the machine zero.

Command specifications:

G28 is non-modal.

X, Z: absolute coordinates of middle point;

U,W: Difference value of absolute coordinates between middle point and starting point in Z direction

.

Omit all or one of X/U ,Z/W as follows:

Table 3-4

Command	Function
G28 X/U	X returns to machine zero and Z/Y axis remain in the previous position
G28 Z/W	Z returns to machine zero and X/ Y axis remain in the previous position
G28	in the previous positions and continuously execute the next block
G28 X/U Z/W	X, Z axis return to machine zero simultaneously

Running path(as Fig. 3-12) :

(1) Rapid traverse to middle point of specified axis from current position(A point→B point) ;

(2) Rapid traverse to reference point from the middle point(B point→R point) ;

(3) If the machine is not locked, LED is ON when the machine reference point return is completed.

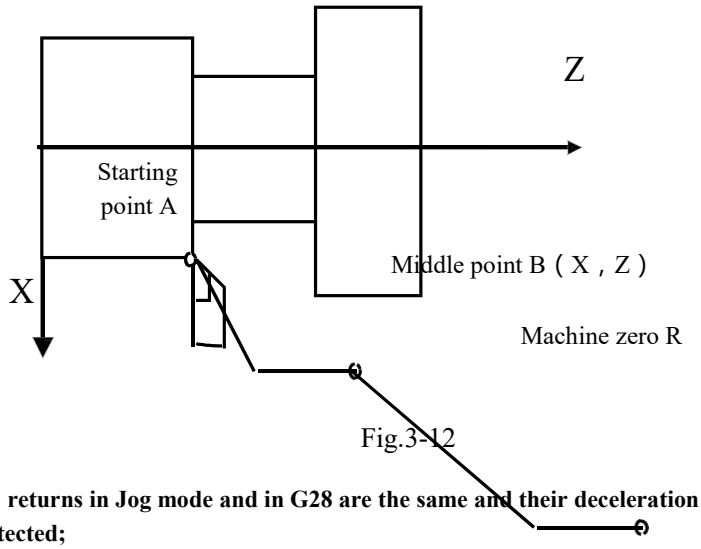


Fig.3-12

Note 1: Machine zero returns in Jog mode and in G28 are the same and their deceleration signals and signals per rev must be detected;

Note 2: X and Z move at the respectively rapid traverse speed from A to B and from B to R, and so the path is not always a straight line;

Note 3: The system cancels the tool length compensation after executing G28 to perform the machine zero return;

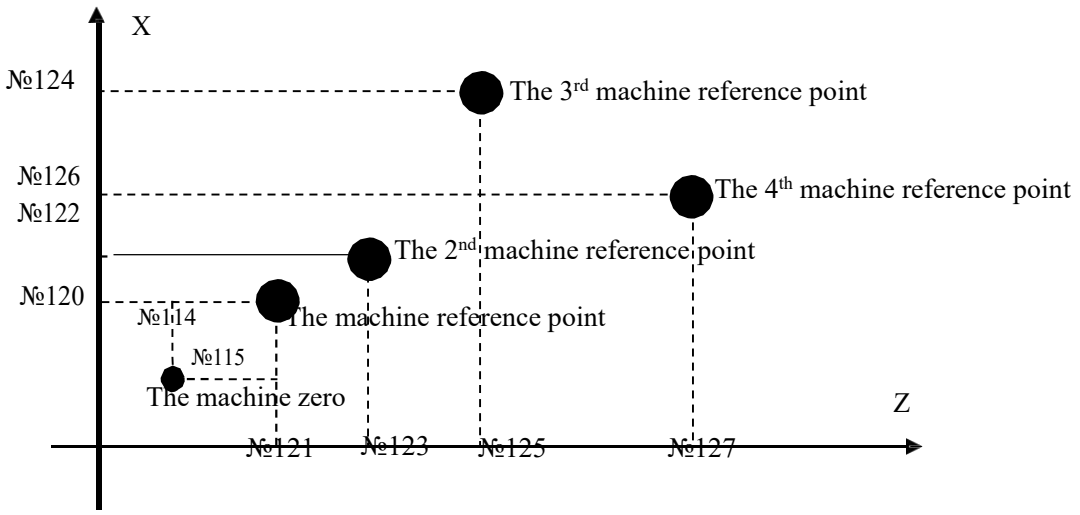
Note 4: Do not execute G28 and machine zero return without the zero switch on the machine.

3.8.2 Machine 2nd, 3rd, 4th reference point G30

Machine zero is fixed point in the machine tool, decided by the zero switch and zero return switch installed on the machine tool.The coordinates of machine reference point are No.120, No.121 setting value.

TAC2000 has machine 2nd, 3rd, 4th reference point functions. Use separately No.122 ~ No.127 to set X, Z machine coordinates of the machine 2nd, 3rd, 4th reference point.

The relationship between the machine zero, machine reference point, machine 2nd, 3rd, 4th reference point is as follows:



Command format: G30

P2 X/U____ Z/W_;

G30 P3 X/U____ Z/W_;

G30 P4 X/U____ Z/W ;

Command function: the tool rapidly traverses with the rapid traverse speed to the middle point specified by X/U , Z/W and then return to machine 2nd, 3rd, 4th reference point

Command specifications: G30 is non-modal.

X: X absolute coordinate of the middle point; U: X relative coordinate of the middle point; Z: Z absolute coordinate of the middle point; W: Z relative coordinate of the middle point; Omit one or all of X/U , Z/W as follows:

Command	Function
G30 Pn X/U ____	X returns to the machine nth reference point, Z axis retains
G30 Pn Z/W ___	Z return to the nth machine reference point, X axis retains
G30	X and Z retain, go on executing the next program block
G30 Pn X/U___ Z/W ___	X and Z return to the machine nth reference point simultaneously

Note 1: n in the above table is 2, 3 or 4;

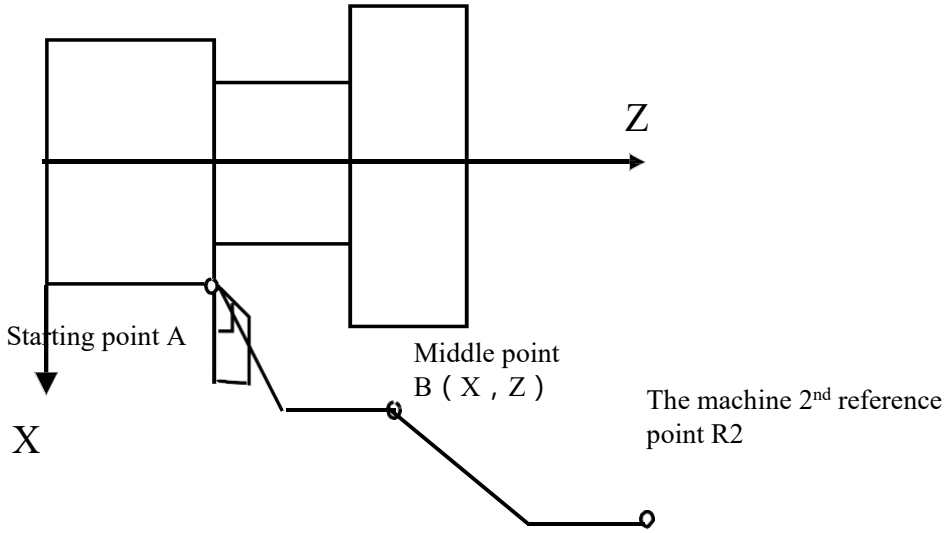
Note 2: Do not check the deceleration, zero signal when you execute the machine 2nd, 3rd, 4th reference point.

Command operations: (taking example of returning to machine 2nd reference point as follows):

(1) Rapidly traverse to the middle position of command axis from the current position (A point →B point);

(2) Traverse from the middle point with the speed set by No.113 to the 2nd reference point set by No.122 and No.123 (B point →R2 point);

(3) When CNC is not in the machine lock state, the completion signal of reference point return ZP21 Bit0, Bit1 is high.



Note 1: Execute the machine 2nd, 3rd, 4th reference point return after you manually execute the machine reference point return or G28 (machine reference point return).

Note 2: A→B and B→R2, two axes separately traverse, and so their trails are linear or not.

Note 3: CNC cancels the tool length compensation after you execute G30 to return 2nd, 3rd, and 4th reference point.

Note 4: Must not execute G30 (machine 2nd, 3rd, 4th reference point return) when the zero switch is not installed on the machine.

Note 5: Do not set the workpiece coordinate system when you execute the 2nd, 3rd, and the machine 4th reference point return.

3.9 Skip interpolation G31

Command format: G31 X/U_ Z/W_ F_;

Command function: in executing the command, when the outside skip signal (X3.5) is input, the system stops the command to execute the next block. The function is used to the dynamic measure (such as milling machine), toolsetting measure and so on of workpiece measure.

Command explanations: non-modal G command (00 group);

Its address format and usage are same that of G01; Cancel the tool nose radius compensation before using it;

Feedrate should not be set to too big to get the precise stop position;

a. following block execution after skip:

1. The next block of G31 is the incremental coordinate programming shown in Fig. 3-13:

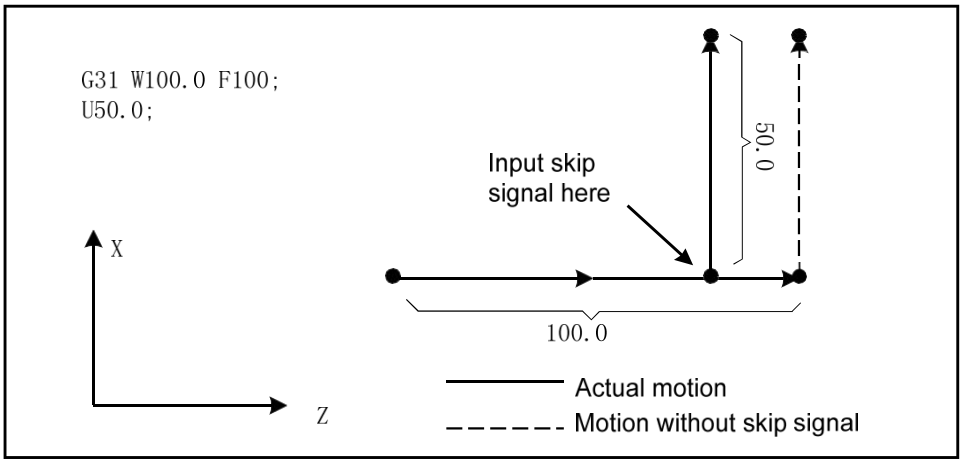


Fig. 3-13

2. The next block of G31 is the absolute coordinate programming of one axis as Fig. 3-14:

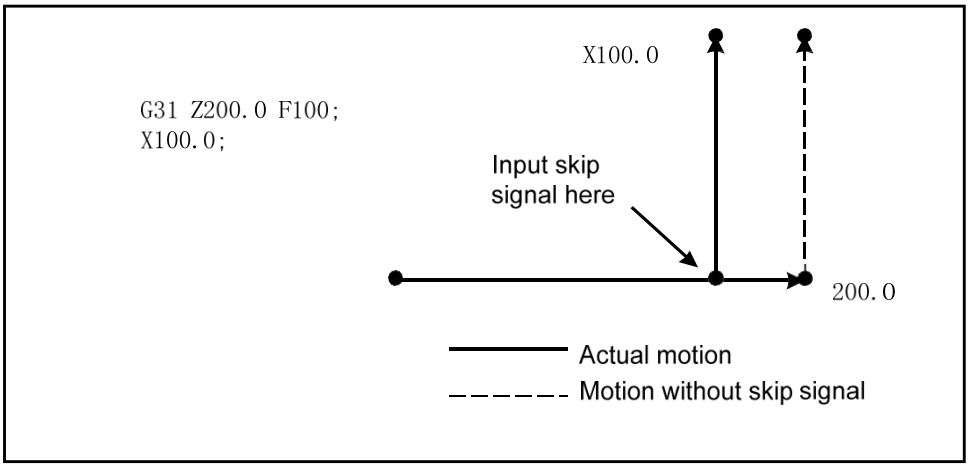


Fig. 3-14

3. The next block of G31 is the absolute coordinate programming of two axes shown in Fig. 3-15: Program:

G31 Z200 F100

G01 X100 Z300

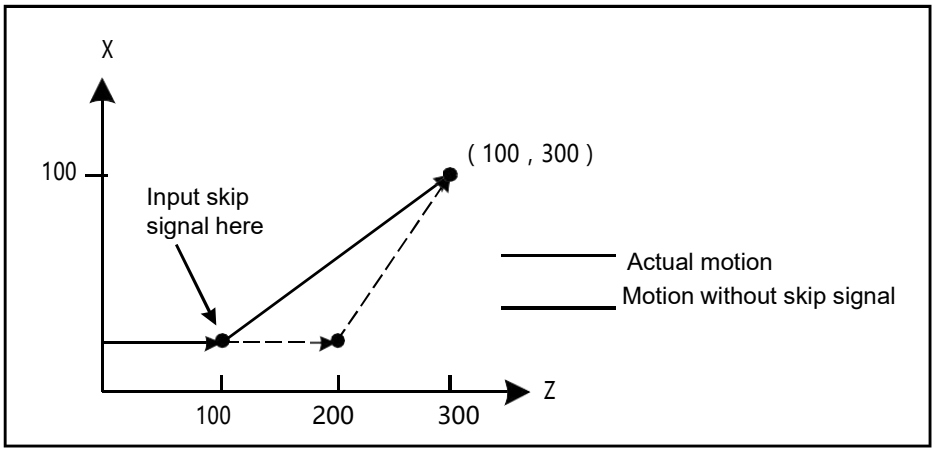


Fig. 3-15

b. Signals related to G31

Skip signal:

SKIP: G6.6

Type: input signal

Function: G6.6 ends the skip cutting. I.e. in a block containing G31, the skip signal becoming the absolute coordinate position of "1" is to be stored in the macro variable (#5011 ~ #5015 separately corresponds to X, Z, Y,4th,5th)

Operation: when the skip signal becomes "1", CNC executes as follows:

When the block is executing G31, CNC stores the current absolute coordinates of each axis. CNC stops G31 to execute the next block, the skip signal detects its state instead of its RISING EDGE. So when the skip signal is "1", it meets the skip conditions.

Note: CNC immediately stops the feed axis (without acceleration/deceleration execution), and G31 feedrate should be as low as possible below 1000 mm/min to get the precise stop position.

3.10 Workpiece coordinate system G50

Command format: G50 X/U Z/W ;

Command function: define the absolute coordinates of current position and create the workpiece coordinates system (called floating coordinates system) by setting the absolute coordinates of current position in the system. After G50 is executed, the system takes the current position as the program zero (program reference point), and the system returns to the point after executing the program zero return. After the workpiece coordinate system is created, input the coordinate values with the coordinate system in the absolute coordinates programming until the next workpiece coordinate system is created again (using G50).

Command specifications:

G50 is non-modal;

X: New absolute coordinates of current position in X direction;

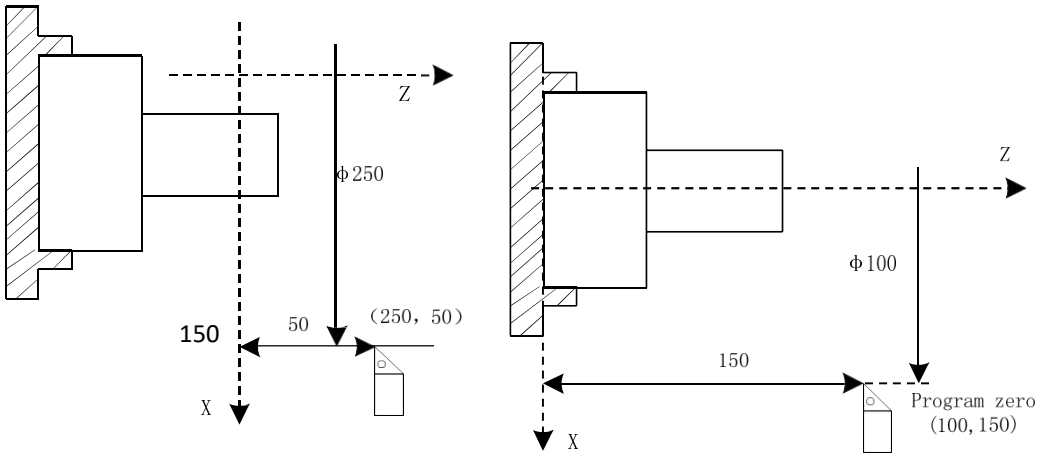
U: Different value between the new absolute coordinates of current position in X direction and the absolute coordinates before executing commands;

Z: New absolute coordinates of current position in Z direction;

W: Different value between the new absolute coordinates of current position in X direction and the absolute coordinates before executing commands;

In G50, when X/U 、 Z/W are not input, the system does not change current coordinates position as program zero; (In G50 SXXXX,not set program zero)

Example:



Before setting coordinate system with G50 **After setting coordinate system with G50**

Fig.3-16

As Fig.3-16, create the above-mentioned workpiece coordinate system and set (X100 Z150) to program zero point after executing “G50 X100 Z150”.

3.11 Workpiece coordinate system G54～G59

Format: G54～G59

Function: It specifies the current workpiece coordinate system. It is used to select workpiece coordinate system by specifying workpiece coordinate system G code in program.

Explanation:

1. No instruction parameter.
2. 6 workpiece coordinate systems can be set in the system, any of which can be selected b by G54~G59 instruction.

G54----------Workpiece coordinate system 1

G55----------Workpiece coordinate system 2

G56----------Workpiece coordinate system 3

G57----------Workpiece coordinate system 4

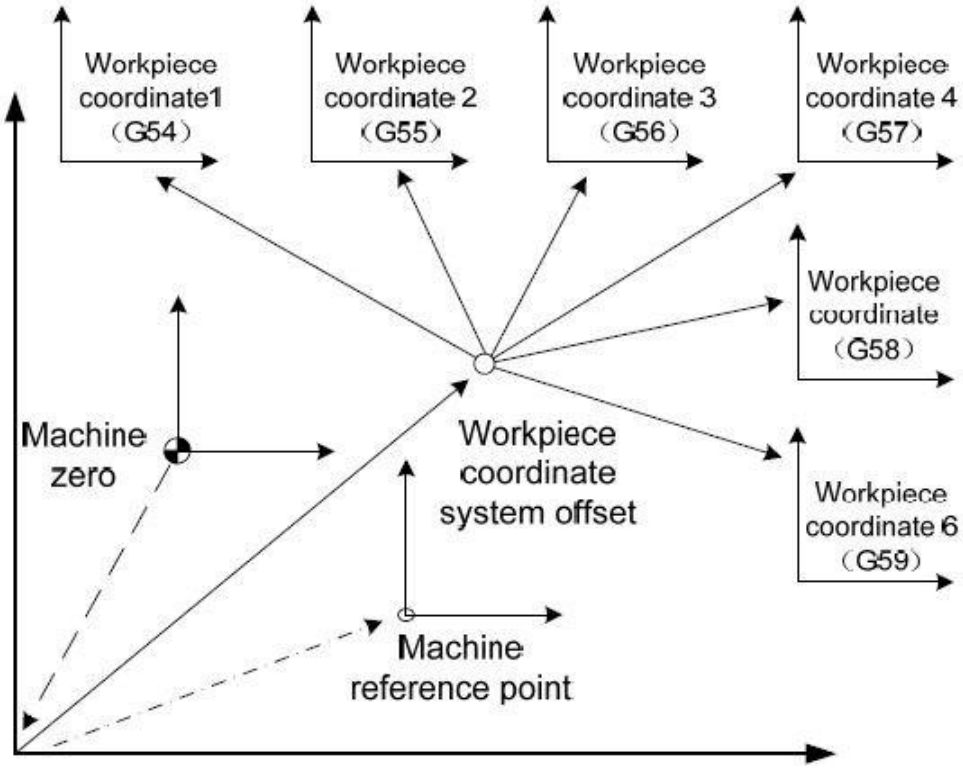
G58----------Workpiece coordinate system 5

G59----------Workpiece coordinate system 6

3. When different workpiece coordinate system is called by block, the axis for move by instruction will be located in the new workpiece coordinate system; for the coordinate of the axis not move, It turns to the corresponding coordinate in the new workpiece coordinate system and the actual machine position doesn't change.

Example: The corresponding machine coordinate for G54 coordinate system origin is （20，20) The corresponding machine coordinate for G55 coordinate system origin is （30，30)

When the program is executed by sequence, the absolute coordinate and the machine coordinate of the end point are shown as follows:



As shown in Fig. 4-2-8-1, after power-on, the machine returns to machine zero by manual zero return. The machine coordinate system is set up by machine zero with the machine reference point generating and workpiece coordinate system to be defined. The corresponding values of offset number parameter P270～ 274 in workpiece coordinate system are the integral offset of the 6 workpiece coordinate system. The 6 workpiece coordinate system origins can be specified by coordinate offset input in MDI mode or set by number parameter P128 ～ 139,P275~P292. These 6 workpiece coordinate systems are set up by the distances from machine zero to each coordinate system origin

Example：

N10 G55 G90 G00 X100 Y20；

N20 G56 X80.5 Z25.5；

For the example above, when N10 block is being executed, it rapidly traverses to a position

（X=100，Y=20）in G55 workpiece coordinate system.

When N20 block is being executed, the absolute coordinate value automatically turns to the coordinate

value （X=80.5，Z=25.5）in G55 workpiece coordinate system for rapid positioning.

3.12 Fixed cycle command

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

G90: axial cutting cycle;

G92: thread cutting cycle;

G94: radial cutting cycle;

G92 will be introduced in section Thread Function.

3.12.1 Axial cutting cycle G90

Command format: G90 X/U_ Z/W_ F_ ; (cylinder cutting)

G90 X/U_ Z/W_ R_ F_ ; (taper cutting)

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

Command specifications:

G90 is modal;

Starting point of cutting: starting position of linear interpolation(cutting feed) End point of cutting: end position of linear interpolation(cutting feed)

X: X absolute coordinates of cutting endpoint

U: different value of X absolute coordinate between end point and starting point of cutting Z: Z absolute coordinates of cutting end point

W: different value of Z absolute coordinate between end point and starting point of cutting

R: different value (radius value) of X absolute coordinates between end point and start point of cutting. When the signs of R is not the same that of U, $R \le |U/2|$; when R = 0 or the input is default, the cylinder cutting is executed as Fig.3-17, otherwise, the cone cutting is executed as Fig. 3-18; unit: mm.

Cycle process:

① X rapidly traverses from starting point to cutting starting point;

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

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

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

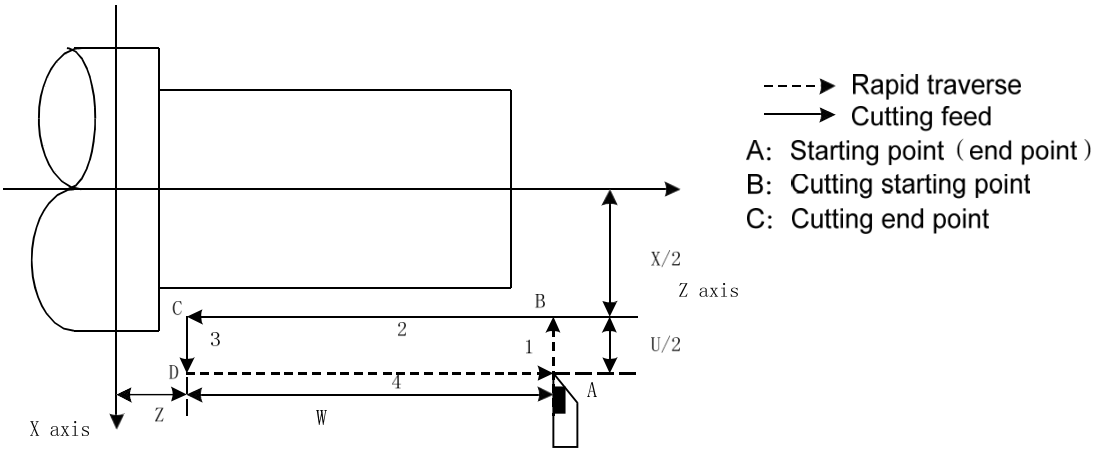


Fig. 3-17

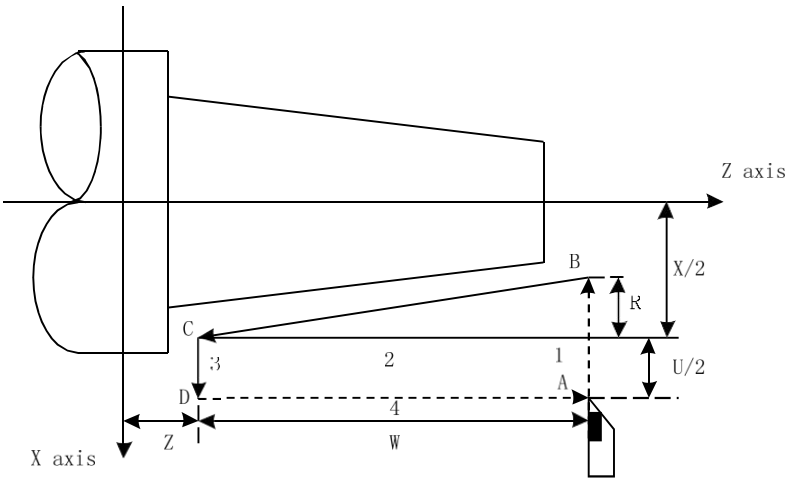
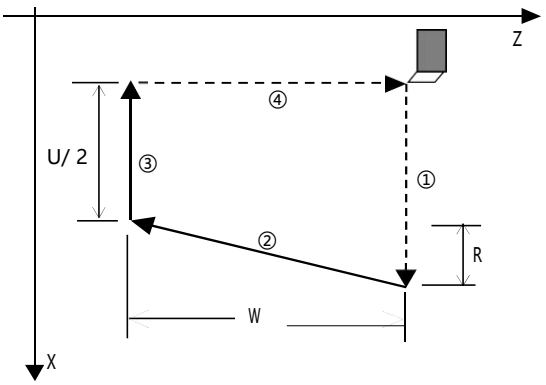


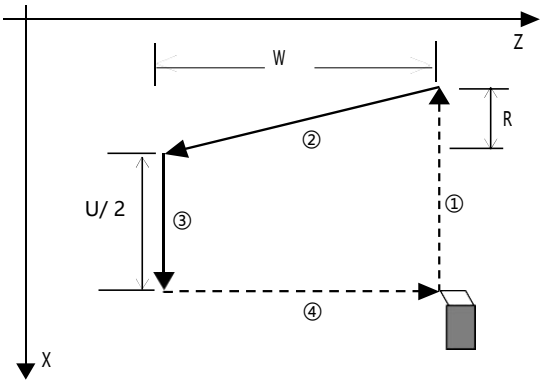
Fig. 3-18

Cutting path: Relative position between cutting end point and starting point with U, W, R, and tool path of U, W, R with different signs are shown in Fig. 3-19:

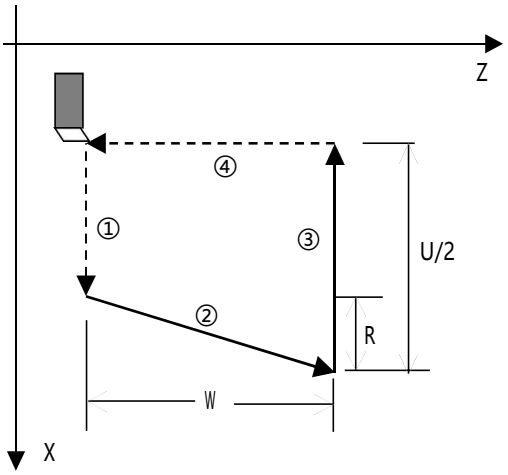
1） U>0，W<0，R>0



2） U<0，W<0，R<0



3） U>0，W>0，R<0，|R|≤|U/2|



4） U<0，W>0，R>0，|R|≤|U/2|

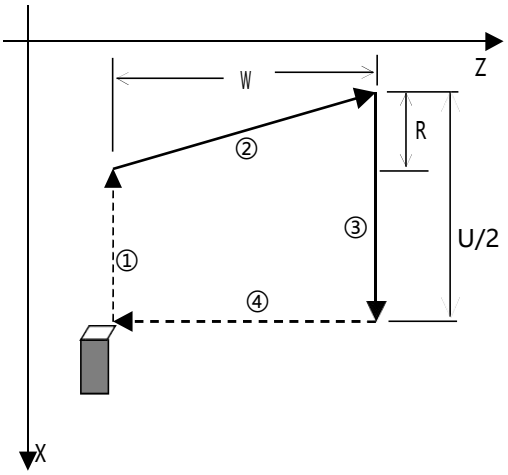


Fig. 3-19

Example: Fig. 3-20 rod Φ125×110

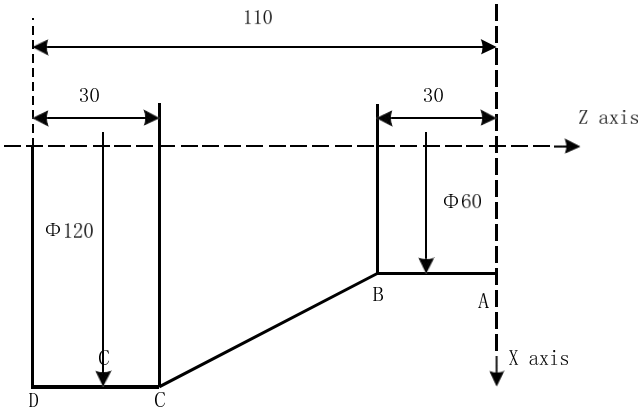


Fig. 3-20

```
Program : O0002;
M3 S300 G0 X130 Z3;
G90 X120 Z-110 F200;          (A→D, cut Φ120)
X110 Z-30;
X100;
X90;
X80;                          ( A→B，6 times cutting cycle Φ60, increment of 10mm )
X70;
X60;
G0 X120 Z-30;
G90 X120 Z-44 R-7.5 F150;     ( B→C，4 times taper cutting )
Z-56 R-15
Z-68 R-22.5
Z-80 R-30
M30;
```

3.12.2 Radial cutting cycle G94

Command format: G94 X/U_____Z/W___F__; (face cutting)

G94 X/U____Z/W___R___F_; (taper face cutting)

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

Command specifications:

G94 is modal;

Starting point of cutting: starting position of linear interpolation (cutting feed). Unit: mm; End point of cutting: end position of linear interpolation (cutting feed). Unit: mm;

X: X absolute coordinate of end point of cutting. Unit: mm;

U: Different value of absolute coordinate from end point to starting point of cutting in X direction .Unit: mm;

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

W: Different value of X absolute coordinate from end point to starting point of cutting, Unit: mm;

R: Different value(R value) of X absolute coordinates from end point to starting point of cutting. When the sign of R is not the same as that of U, R, $|R| \le |W|$.

Radial linear cutting is shown in Fig. 3-21, radial taper cutting is as Fig. 3-22. Unit: mm

Cycle process:

① Z rapidly traverses from starting point to cutting starting point;

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

③ Z executes the tool retraction at the cutting feedrate (opposite direction to the above-mentioned ①), and returns to the position which the absolute coordinates and the starting point are the same;

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

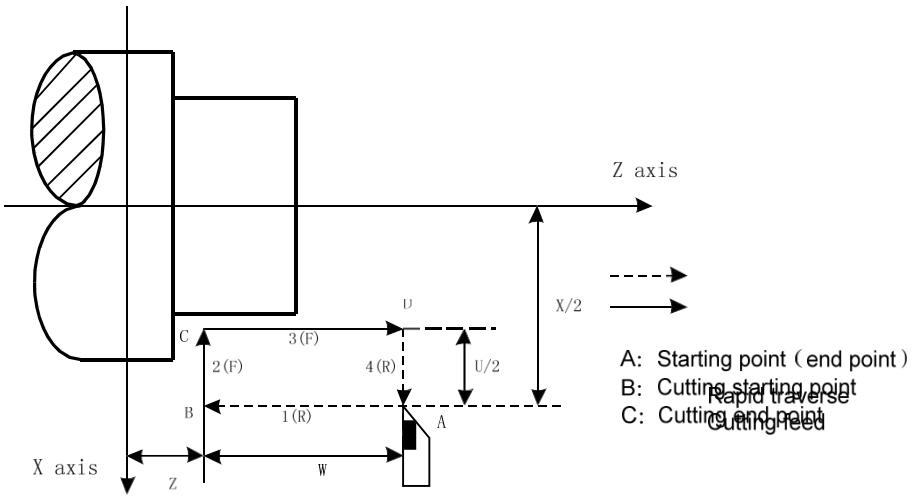


Fig. 3-21

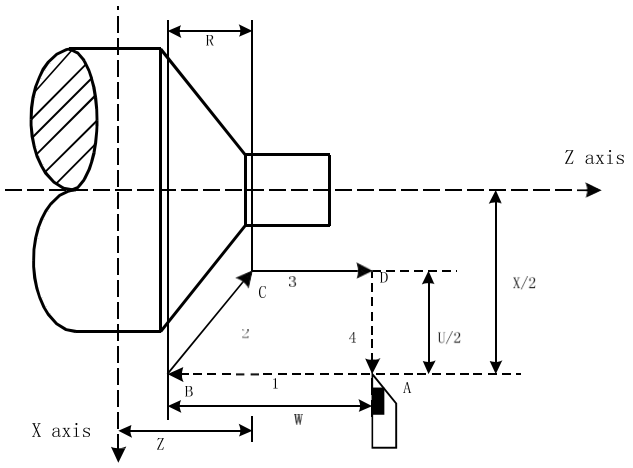
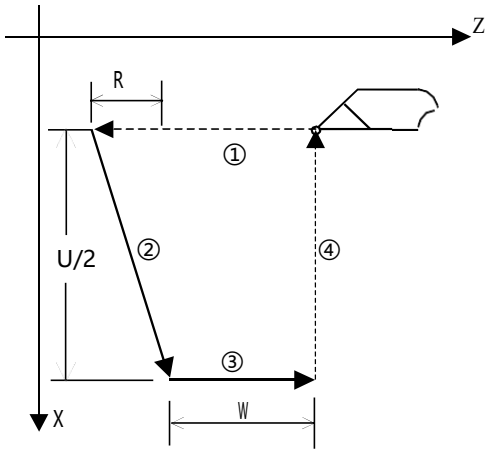


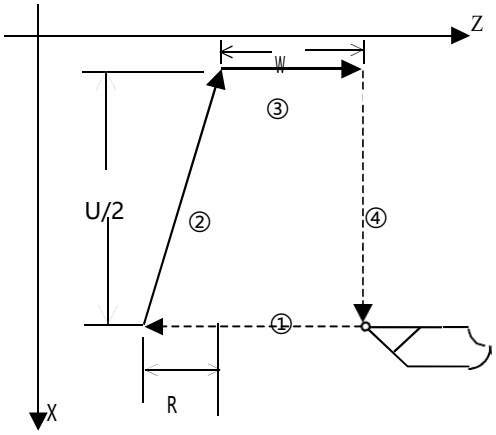
Fig. 3-22

Cutting path: Relative position between cutting end point and starting point with U, W,R is shown in Fig.3-23:

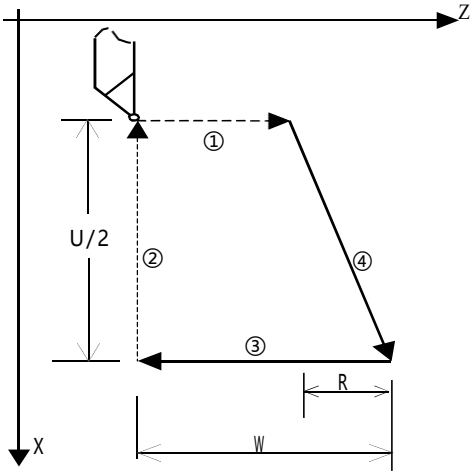
1) U>0 W<0 R<0



2) U<0 W<0 R<0



(3) U>0 W>0 R<0 (|R|≤|W|)



4) U<0 W>0 R<0 (|R|≤|W|)

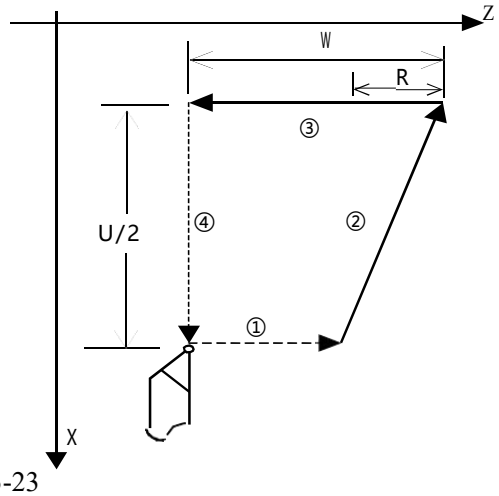


Fig. 3-23

Example: Fig. 3-24, rod Φ125×112

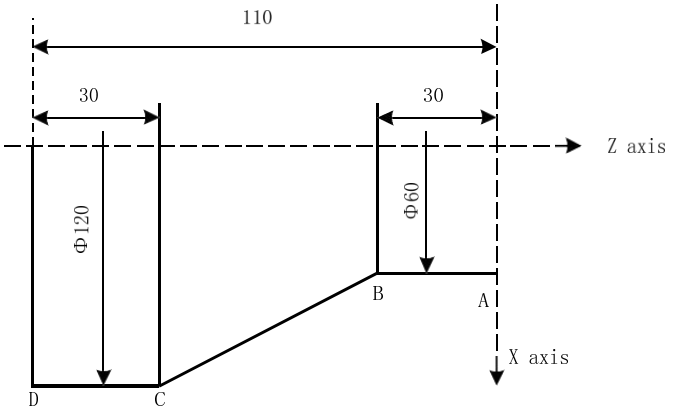


Fig. 3-24

Program: O0003;

```
G00 X130 Z5 M3 S1;
G94 X0 Z0 F200
X120 Z-110 F300;
G00 X120 Z0
G94  X108  Z-30  R-10
X96 R-20
X84   R-30
X72   R-40
X60  R-50;
M30;
```

End face cutting

(Outer cutting Φ120)

(C→B→A , cutting Φ60)

3.12.3 Caution of fixed cycle commands

1) After X(U) , Z(W) , R are executed in the canned cycle command, their command values are valid if X(U) , Z(W) ,R are not redefined by executing a new canned cycle commands. The command values of X(U) ,Z(W) ,R are cleared if non-modal G command(00 Group) except for G04 or G00, G01, G02, G03, G32 is executed.

2) Pause or single block is executed in G90, G94, the single block stops after the tool moves end point of current path.

3.13 Multiple cycle commands

Multiple cycle commands of the system includes axial roughing cycle G71, radial roughing cycle G72, closed cutting cycle G73, finishing cycle G70, axial grooving multiple cycle G74, axial grooving multiple cycle G75 and multiple thread cutting cycle G76. When the system executes these commands, it automatically counts the cutting times and the cutting path according to the programmed path, travels of tool infeed and tool retraction, executes multiple machining cycle (tool infeed → cutting→retract tool→tool infeed), automatically completes the roughing, finishing workpiece and the starting point and the end point of command are the same one.

3.13.1 Axial roughing cycle G71

Command format：G71 U(Δd) R(e) F_____S___T__ ; (1)

G71 P(ns) Q(nf) U(Δu) W(Δw) K0/1 ; (2)

```
N(ns) G0/G1 X(U) . . ;
 . . . . . . . . . ;
  . . . . F ;
  . . . . S ;
  . . . .
N(nf) . . . . . ;
```
(3)

Command function: G71 is divided into three parts:

(1) 1st blocks for defining the travels of tool infeed and retract tool, the cutting feedrate, the spindle speed and the tool function when roughing;

(2) 2nd blocks for defining the block interval, finishing allowance;

(3) 3rd blocks for some continuous finishing path, counting the roughing path without being executed actually when executing G71.

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

Relevant definitions:

Finishing path: The above-mentioned Part 3 of G71(ns ~ nf block)defines the finishing path, and the starting point of finishing path (starting point of ns block)is the same these of starting point and end point of G71, called A point; the first block of finishing path(ns block)is used for X rapid traversing or tool infeed, and the end point of finishing path is called to B point; the end point of finishing path(end point of nf block)is called to C point. The finishing path is A→B→C.

Roughing path: The finishing path is the one after offsetting the finishing allowance(Δu,Δw) and is the path contour formed by executing G71. A, B, C point of finishing path after offset corresponds separately to A', B', C' point of roughing path, and the final continuous cutting path of G71 is B'→C' point.

Δd: It is each travel(unit: mm, radius value) of X tool infeed in roughing, its value: 0.001 （IS_B） /0.0001 （IS_C） ~99.999(unit: mm,radius value) without sign, and the direction of tool infeed is defined by move direction of ns block. The command value Δd is reserved after executing U(Δd) and the value of system parameter No.051 is rewritten to Δd×1000(unit: 0.001 mm) . The value of system parameter No.051 is regarded as the travel of tool infeed when U(Δd) is not input.

e: It is travel(unit: mm, radius value) of X tool retraction in roughing its value: 0~99.999(unit: mm,radius value) without sign, and the direction of tool retraction is opposite to that of tool infeed, the command value e is reserved and the value of system parameter No.052 is rewritten to e×1000(unit: 0.001 mm) after R(e) is executed. The value of system parameter No.052 is regarded as the travel of tool retraction when R(e) is not input.

ns: Block number of the first block of finishing path. nf:

Block number of the last block of finishing path.

Δu: X finishing allowance is ±99999.999× least input increment with sign symbol (diameter). X coordinate offset of roughing path compared to finishing path, i.e. the different value of X absolute coordinates between A'and A. The system defaults Δu=0 when U(Δu) is not input,
i.e. there is no finishing allowance in X direction for roughing cycle.

Δw: Z finishing allowance is ±99999.999× least input increment with sign symbol (diameter). the Z coordinate offset of roughing path compared to finishing path, i.e. the different value of Z absolute coordinate between A' and A. The system defaults Δw=0 when W(Δw) is not input,
i.e. there is no Z finishing allowance for roughing cycle.

K: When K is not input or is not 1, the system does not check the program monotonicity except that the Z value of starting point and end point of the arc or ellipse or parabola or the arc is more than 180 degree; K=1, the system checks the program monotonicity.

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

M, S, T, F: They can be specified in the first G71 or the second ones or program ns ~ nf. M, S, T, F functions of M, S, T, F blocks are invalid in G71, and they are valid in G70 finishing blocks.

Type I：

1）Execution process: (Fig. 3-25)

① X rapidly traverses to A' from A point, X travel is Δu, and Z travel is Δw;

② X moves from A'is Δd(tool infeed), ns block is for tool infeed at rapid traverse speed with G0, is for tool infeed at feedrate F with G71, and its direction of tool infeed is that of A→B point;

③ Z executes the cutting feeds to the roughing path, and its direction is the same that of Z coordinate B→C point;

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

⑤ Z rapidly retracts at rapid traverse speed to the position which is the same that of Z coordinate;

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

executed;

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

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

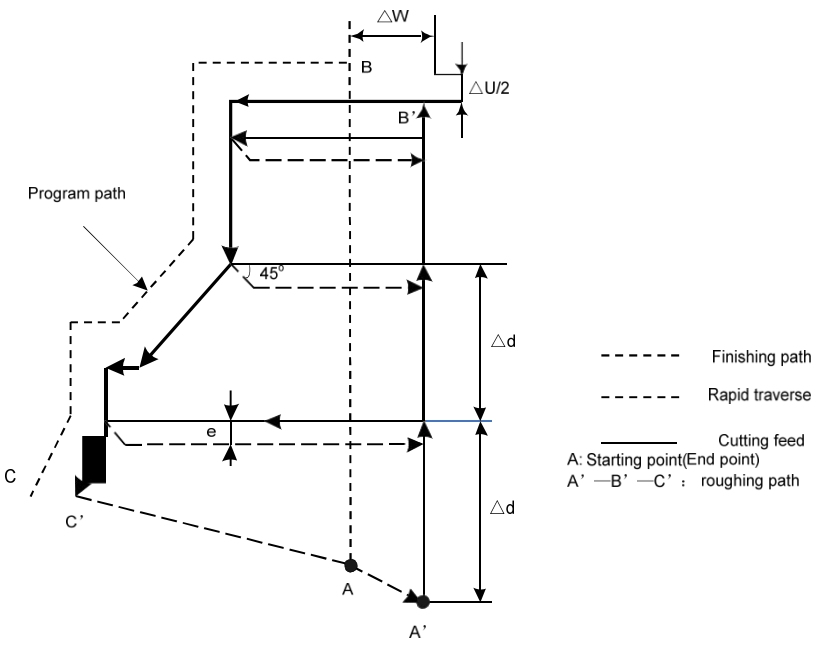


Fig. 3-25 G71 cycle path 2）

Coordinate offset direction with finishing allowance:

Δu, Δw define the coordinate offset and cut-in direction in finishing, and their sign symbol are as follows Fig. 3-26: B→C for finishing path, B'→C' for roughing path and A is the tool start-up point.

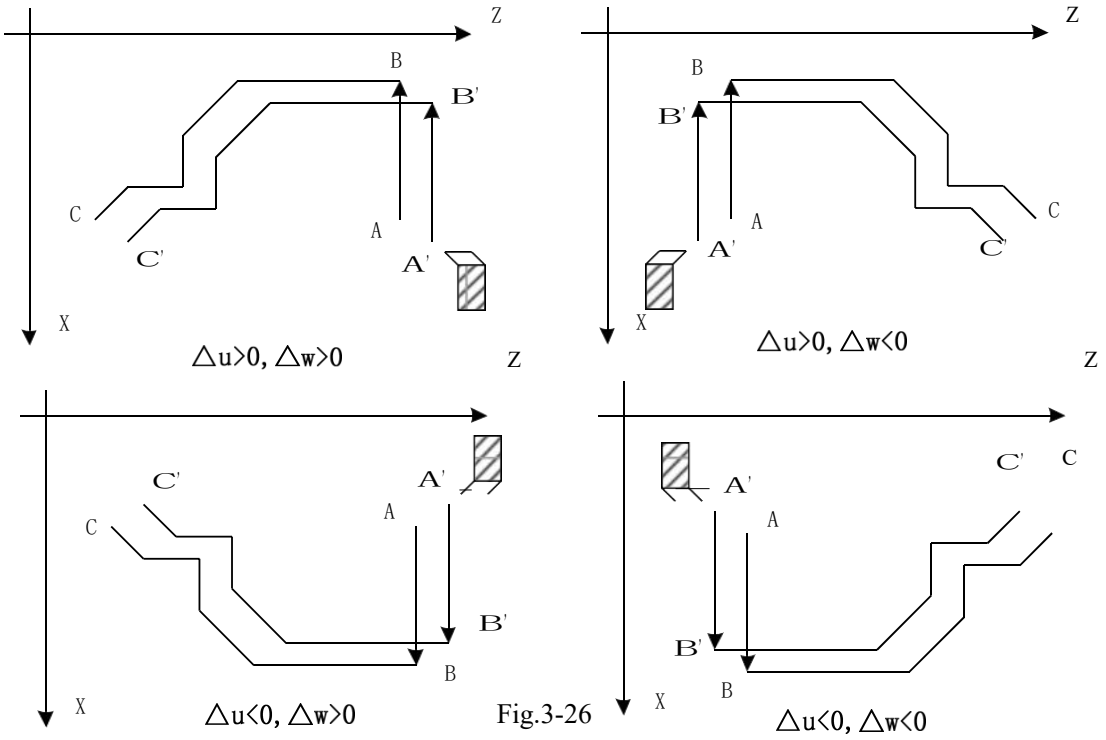


Fig.3-26

Notes：

- ns block is only G00, G01.
- For the finishing path(ns ~ nf block)，Z dimension must be monotonous change(always increasing or decreasing)
- ns ~ nf blocks in programming must be followed G71 blocks.
- ns ~ nf blocks are used for counting the roughing path and the blocks are not executed when G71 is executed. F, S, T commands of ns ~ nf blocks are invalid when G71 is executed, at the moment, F, S, T commands of G71 blocks are valid. F, S, T of ns ~ nf blocks are valid when executing ns ~ nf to command G70 finishing cycle;
- In ns ~ nf blocks, there are only G commands: G00, G01, G02, G03, G04, G96, G97, G98, G99, G40, G41, G42 and the system cannot call subprograms(M98/M99);
- G96, G97, G98, G99, G40, G41, G42 are invalid when G71 is executed, and are valid when G70 is executed;
- When G71 is executed, the system can stop the automatic run and manual traverse
- When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path;
- △d,△u are specified by the same U and different with or without being specified P, Q commands;
- G71 cannot be executed in MDI, otherwise, the system alarms;
- There are no the same block number in ns~nf when compound cycle commands are executed repetitively in one program;
- The tool retraction point should be high or low as possible to avoid crashing the workpiece

. Example：Fig. 3-27（Type I）

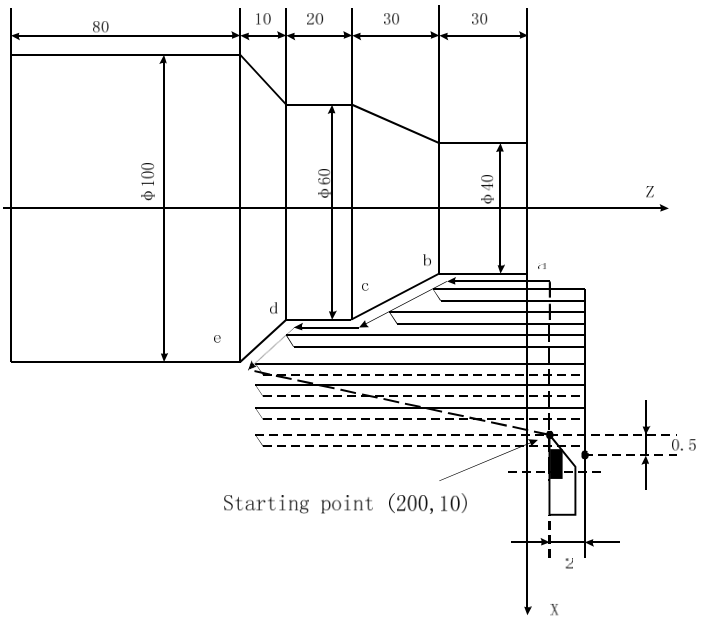


Fig. 3-27

Program: O0004;

G00 X200 Z10 M3 S800; (Spindle clockwise with 800 r/min)

G71 U2 R1 F200; (Cutting depth each time 4mm, tool retraction 2mm [in diameter])

G71 P80 Q120 U0.5 W0.2; (roughing a---e, machining allowance: X, 1mm;Z, 2mm) N80

G00 X40 S1200; (Positioning)

G01 Z-30 F100 ; (a→b)

X60 W-30; (b→c)

W-20; (c→d)

N120 X100 W-10; (d→e)

a→b→c→d→e blocks for finishing path

G70 P80 Q120; (a---e blocks for finishing path)

M30; (End of block)

3.13.2 Radial roughing cycle G72

Command format：G72 W(Δd) R(e) F____S____T__ ; (1)

G72 P(ns) Q(nf) U(Δu) W(Δw) K0/1 (2)

N_(ns) . . . . . ;
. . . . . ;
. . . . F ;
. . . . S ;
. . . . . ; (3)
.
N_(nf) . . . . . ;

Command function: G72 is divided into three parts:

(1) 1st blocks for defining the travels of tool infeed and tool retraction, the cutting speed, the spindle speed and the tool function in roughing;

(2) 2nd blocks for defining the block interval, finishing allowance;

(3) 3rd blocks for some continuous finishing path, counting the roughing path without being executed actually when G72 is executed.

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

Relevant definitions:

Finishing path: the above-mentioned Part (3) of G71(ns ~ nf block)defines the finishing path, and the starting point of finishing path (i.e. starting point of ns block)is the same these of starting point and end point of G72, called A point; the first block of finishing path(ns block)is used for Z rapid traversing or cutting feed, and the end point of finishing path is called to B point; the end point of finishing path(end point of nf block)is called to C point. The finishing path is A→B→C.

Roughing path: The finishing path is the one after offsetting the finishing allowance(Δu, Δw) and is the path contour formed by executing G72. A, B, C point of finishing path after offset corresponds separately to A', B', C' point of roughing path, and the final continuous cutting path of G72 is B'→C' point.

Δd: it is Z cutting in roughing, its value: 0.001~99.999(unit: mm) without sign symbol, and the direction of tool infeed is determined by ns block traverse direction. the specified value Δd

is reserved and the data value is switched to the corresponding value to save to No.051 after W(Δd) is executed. The value of system parameter No.051 is regarded as the tool infeed clearance when R(e) is not input.

e: it is Z tool retraction clearance in roughing, its value: 0~99.999(unit: mm) without sign symbol, and the direction of tool retraction is opposite to that of tool infeed, the specified value e is reserved and the data value is switched to the corresponding value to save to No.052 after R(e) is executed. The value of system parameter No.052 is regarded as the tool retraction clearance when R(e) is not input.

ns: Block number of the first block of finishing path. nf:
Block number of the last block of finishing path.

Δu：it is X finishing allowance in roughing, its range: ±99999999×least input increment(X coordinate offset of roughing contour corresponding to the finishing path, i.e. X absolute coordinate difference between A' and A, in diameter with sign symbol).

Δw：it is Z finishing allowance in roughing, its range: ±99999999×least input increment(Z coordinate offset of roughing contour corresponding to the finishing path, i.e. Z absolute coordinate difference between A' and A, in diameter with sign symbol).

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

M, S, T, F: They can be specified in the first G72 or the second ones or program ns ~ nf. M, S, T, F functions of M, S, T, F blocks are invalid in G72, and they are valid in G70 finishing blocks.

Execution process: Fig. 3-28

①X rapidly traverses to A' from A point, X travel is Δu, and Z travel is Δw;

②X moves from from A' is Δd(tool infeed), ns block is for tool infeed at rapid traverse speed with G0, is for tool infeed at G72 feedrate F in G1, and its direction of tool infeed is that of A→B point;

③X executes the cutting feeds to the roughing path, and its direction is the same that of X coordinate B→C point;

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

⑤X rapidly retracts at rapid traverse speed to the position which is the same that of Z coordinate;

⑥After Z tool infeed (Δd+e)again is executed, the end point of traversing tool is still on the middle point of straight line between A'and B'(the tool does not reach or exceed B'), and after Z executes the tool infeed (Δd+e)again, ③ is executed; after the tool infeed (Δd+e) is executed again, the end point of tool traversing reaches B' point or exceeds the straight line between A'→B' point and Z executes the tool infeed to B' point, and then the next step is executed;

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

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

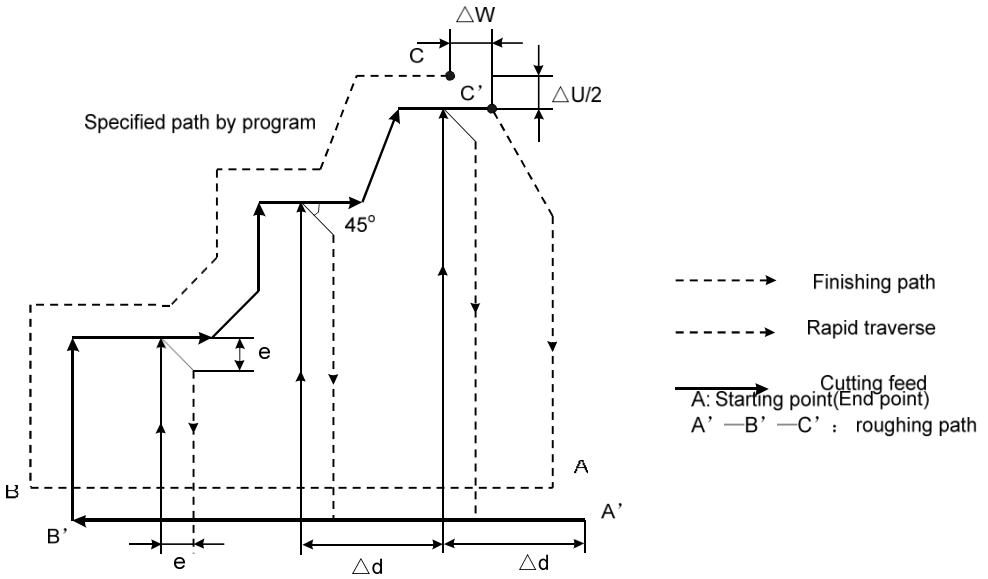


Fig. 3-28

Command specifications:

- ns ~ nf blocks in programming must be followed G72 blocks.
- ns ~ nf blocks are used for counting the roughing path and the blocks are not executed when G72 is executed. F, S, T commands of ns ~ nf blocks are invalid when G72 is executed, at the moment, F, S, T commands of G72 blocks are valid. F, S, T of ns ~ nf blocks are valid when executing ns ~ nf to command G70 finishing cycle;
- There are G00, G01 without the word X(U) in ns block, otherwise the system alarms;
- The dimensions in X, Z direction must be changed monotonously (always increasing or reducing) for the finishing path;
- In ns ~ nf blocks, there are only G commands: G01, G02, G03, G04, G96, G97, G98, G99, G40, G41, G42 and the system cannot call subprograms(M98/M99);
- G96, G97, G98, G99, G40, G41, G42 are invalid when G72 is executed, and are valid when G70 is done;
- When G72 is executed, the system can stop the automatic run and manual traverse
- When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path;
- △d, △w are specified by the same W and different with or without being specified P, Q commands;
- There are no the same block number in ns~nf when compound cycle commands are executed repetitively in one program;
- G72 cannot be executed in MDI, otherwise, the system alarms;
- The tool retraction point should be high or low as possible to avoid crashing the workpiece.

Coordinate offset direction with finishing allowance:

Δu, Δw define the coordinate offset and its direction of cut-in in finishing, and their sign symbol are as follows Fig. 3-29: B→C for finishing path, B'→C' for roughing path and A is the tool start-up point.

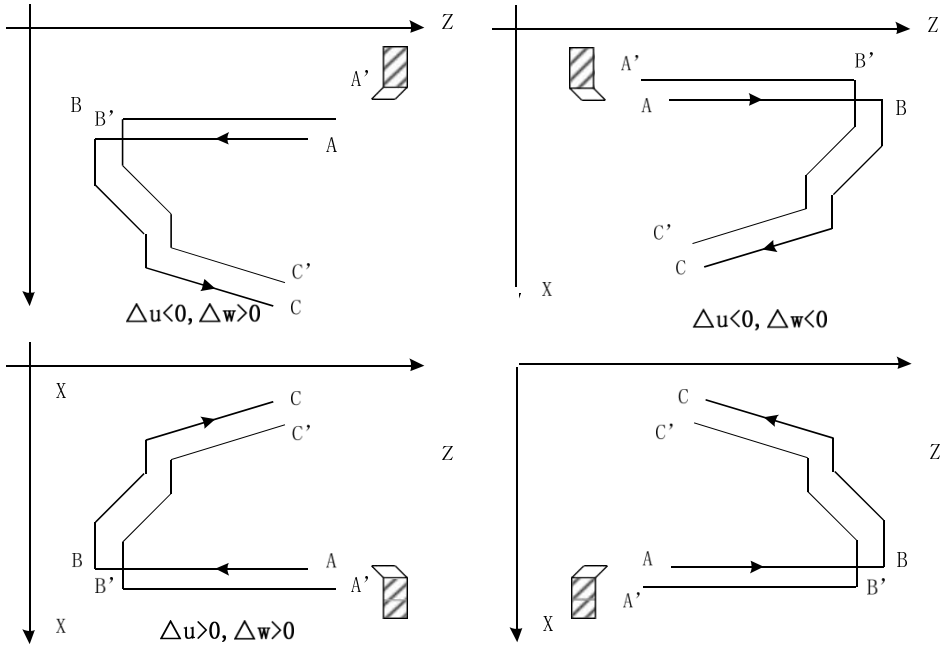


Fig.3-29

Example：Fig. 3-30

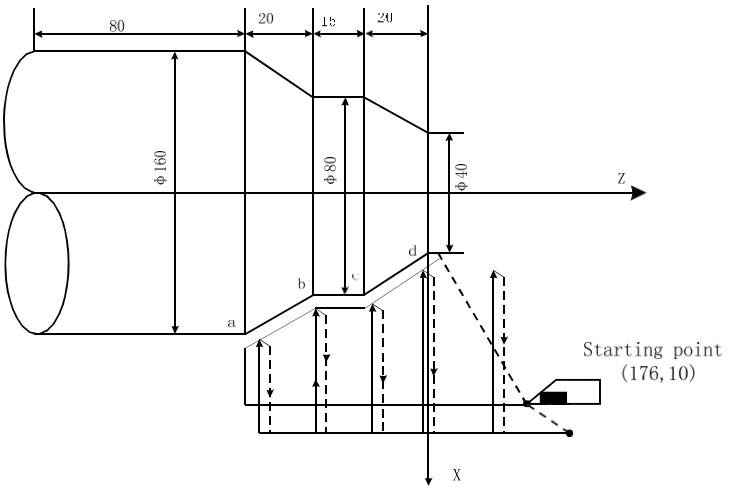


Fig.3-30

Program:

```
O0005;
G00 X176 Z10 M03 S500        (Change No.2 tool and execute its compensation, spindle CW
                               rotation with 500 r/min)
G72 W2.0 R0.5 F300;          (Tool infeed 2mm, tool retraction 0.5mm)
G72 P10 Q20 U0.2 W0.1;       (Roughing a--d,X roughing allowance 0.2mm and Z 0.1mm)
```

```
N10 G00 Z-55 S800 ;        (Rapid traverse)
G01 X160 F120;             (Infeed to a point)
X80 W20;                   (Machining a—b)        Blocks for finishing path
W15;                       (Machining b—c)
N20 X40 W20 ;              (Machining c—d)
G70 P050 Q090 M30;         (Finishing a—d)
```

3.13.3 Closed cutting cycle G73

Command format:

```
G73 U(Δi)  W (Δk)  R (d)  F___S___T__;        (1)
G73 P(ns)  Q(nf)  U(Δu)  W(Δw) ;              (2)
N__(ns)  . . . . . ;
  . . . . . . . ;
  . . . . F;
  . . . . S;                                  (3)
  . . . . . ;
  .
  N__ (nf)  . . . . . ;
```

Command functions: G73 is divided into three parts:

(1) Blocks for defining the travels of tool infeed and tool retraction, the cutting speed, the spindle speed and the tool function when roughing;

(2) Blocks for defining the block interval, finishing allowance;

(3) Blocks for some continuous finishing path, counting the roughing path without being executed actually when executing G73.

According to the finishing allowance, the travel of tool retraction and the cutting times, the system automatically counts the travel of roughing offset, the travel of each tool infeed and the path of roughing, the path of each cutting is the offset travel of finishing path, the cutting path approaches gradually the finishing one, and last cutting path is the finishing one according to the finishing allowance. The starting point and end point of G73 are the same one, and G73 is applied to roughing for the formed rod. G73 is non-modal and its path is shown in Fig.3-77.

Relevant definitions:

Finishing path: The above-mentioned Part 3 of G73 (ns ~ nf block)defines the finishing path, and the starting point of finishing path (start point of ns block)is the same these of starting point and end point of G73, called A point; the end point of the first block of finishing path(ns block)is called B point; the end point of finishing path(end point of nf block) is called C point. The finishing path is A→B→C.

Roughing path: It is one group of offset path of finishing one, and the roughing path times are the same that of cutting. After the coordinates offset, A, B, C of finishing path separately corresponds to A_n, B_n , C_n of roughing path(n is the cutting times, the first cutting path is A_1, B_1, C_1 and the last one is A_d, B_d, C_d). The coordinates offset value of the first cutting compared to finishing path is (Δi×2+Δu, Δw+Δk) (diameter programming) , the coordinates offset value of the last cutting compared to finishing path is(Δu, Δw) , the coordinates offset value of each cutting compared to the previous one is as follows:

Δi: It is X tool retraction clearance in roughing, and its range is ±99999.999× least input increment (radius, with sign symbol) , Δi is equal to X coordinate offset value (radius

value) of A1 point compared to Ad point. The X total cutting travel(radius value) is equal to |Δi| in roughing, and X cutting direction is opposite to the sign of Δi: Δi > 0, the system

executes X negative cutting in roughing. It is reserved after Δi specified value is executed and the data is switched to the corresponding value to save to NO.053. The No.053 value is regarded as X tool retraction clearance in roughing when U(Δi) is not input.

Δk: It is Z tool retraction clearance in roughing, and its range is ±99999.999× least input increment (radius, with sign symbol) , Δk is equal to Z coordinate offset value (radius value) of A1 point compared to Ad point. Z total cutting travel(radius value) is equal to

|Δk| in roughing, and Z cutting direction is opposite to the sign of Δk: Δi > 0, the system executes Z negative cutting in roughing. It is reserved after Δk specified value is executed and the data is switched to the corresponding value to save to NO.054. The No.054 value is regarded as Z tool retraction clearance in roughing when W(Δk) is not input.

d: It is the cutting times 1~9999 (unit: times). R5 means the closed cutting cycle is completed by 5 times cutting. R (d) is reserved after it is executed and NO.055 value is rewritten to d (unit: times). No.055 value is regarded as the cutting times when R(d) is not input. When the cutting times is 1, the system completes the closed cutting cycle based on 2 times cutting.

ns: Block number of the first block of finishing path. nf:

Block number of the last block of finishing path.

Δu: It is X finishing allowance and its range is ±99999.999× least input increment (diameter, with sign symbol) and is the X coordinate offset of roughing path compared to finishing

path, i.e. the different value of X absolute coordinates of A_1 compared to A. Δu > 0,it is the offset of the last X positive roughing path compared to finishing path. The system defaults Δu=0 when U(Δu) is not input, i.e. there is no X finishing allowance for roughing cycle.

Δw: It is Z finishing allowance and its range is ±99999.999× least input increment (diameter, with sign symbol) and is the X coordinate offset of roughing path compared to finishing

path, i.e. the different value of Z absolute coordinates of A_1 compared to A. Δw > 0,it is the offset of the last X positive roughing path compared to finishing path. The system defaults Δw=0 when W(Δw) is not input, i.e. there is no Z finishing allowance for roughing cycle.

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

M, S, T, F: They can be specified in the first G73 or the second ones or program ns ~ nf. M, S, T, F functions of M, S, T, F blocks are invalid in G73, and they are valid in G70 finishing blocks.

Execution process: (Fig. 3-31)

① $A \rightarrow A_1$: Rapid traverse;

② First roughing $A_1 \rightarrow B_1 \rightarrow C_1$:

$A_1 \rightarrow B_1$: Rapid traverse speed in ns block in G0, cutting feedrate specified by G73 in ns block in G1;

$B_1 \rightarrow C_1$: Cutting feed.

③ $C_1 \rightarrow A_2$: Rapid traverse.

④ Second roughing $A_2 \rightarrow B_2 \rightarrow C_2$:

$A_2 \rightarrow B_2$: Rapid traverse speed in ns block in G0, cutting feedrate specified by G73 in ns block in G1;

$B_2 \rightarrow C_2$: Cutting feed.

⑤ $C_2 \rightarrow A_3$: Rapid traverse:

……

No. n times roughing, $A_n \rightarrow B_n \rightarrow C_n$:

$A_n \rightarrow B_n$: ns Rapid traverse speed in ns block in G0, cutting feedrate specified by G73 in ns block in G1;

$B_n \rightarrow C_n$: Cutting feed.

$C_n \rightarrow A_{n+1}$: Rapid traverse;

……

Last roughing, $A_d \rightarrow B_d \rightarrow C_d$:

$A_d \rightarrow B_d$: Rapid traverse speed in ns block in G0, cutting feedrate specified by G73 in ns block in G1;

$B_d \rightarrow C_d$: Cutting feed.

$C_d \rightarrow A$: Rapid traverse to starting point;

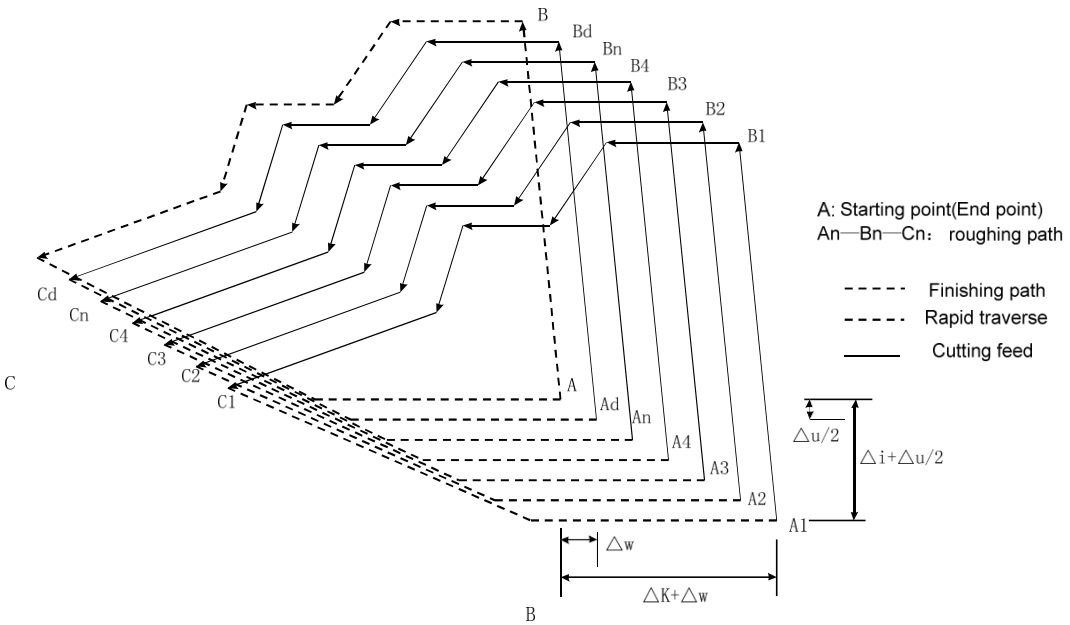


Fig. 3-31 G73 path

Command specifications:

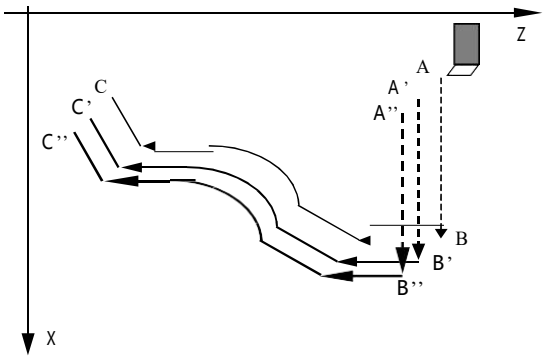
- ns ~ nf blocks in programming must be followed G73 blocks.
- ns ~ nf blocks are used for counting the roughing path and the blocks are not executed when G73 is executed. F, S, T commands of ns ~ nf blocks are invalid when G71 is executed, at the moment, F, S, T commands of G73 blocks are valid. F, S, T of ns ~ nf blocks are valid when executing ns ~ nf to command G70 finishing cycle.
- There are only G00, G01 in ns block.
- I n ns ~ nf blocks, there are only G commands: G00, G01, G02, G03, G04, G96, G97, G98, G99, G40, G41, G42 and the system cannot call subprograms(M98/M99)

- G96, G97, G98, G99, G40, G41, G42 are invalid when G73 is executed, and are valid when G70 is executed.
- When G73 is executed, the system can stop the automatic run and manual traverse
- When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path.
- △i, △u are specified by the same U and Δk, Δw are specified by the same U, and they are different with or without being specified P,Q commands.
- G73 cannot be executed in MDI, otherwise, the system alarms.
- There are no the same block number in ns~nf when compound cycle commands are executed repetitively in one program.
- The tool retraction point should be high or low as possible to avoid crashing the workpiece.

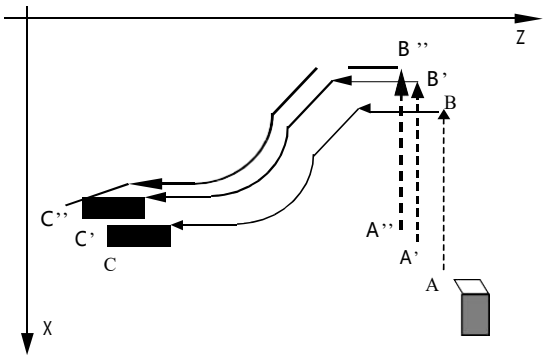
Coordinate offset direction with finishing allowance:

Δi, Δk define the coordinates offset and its direction of roughing; Δu, Δw define the coordinate offset and the cut-in direction in finishing, and their sign symbols are as follows Fig. 3-32: A is tool start-up point, B→C for workpiece contour, B'→C' for roughing contour and B''→C'' for finishing path.

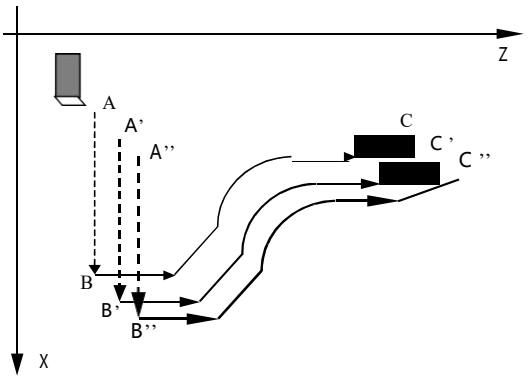
1）Δi<0 Δk>0， △u<0 Δw>0；



2）Δi>0 Δk>0， △u>0 Δw>0；



3）Δi<0 Δk<0， △u<0 Δw<0；



4）Δi>0 Δk<0， △u>0 Δw<0；

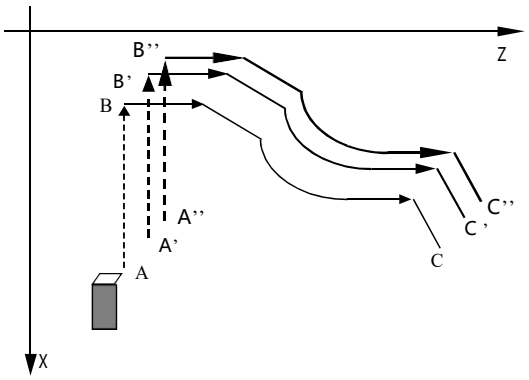


Fig.3-32

Example：Fig. 3-33

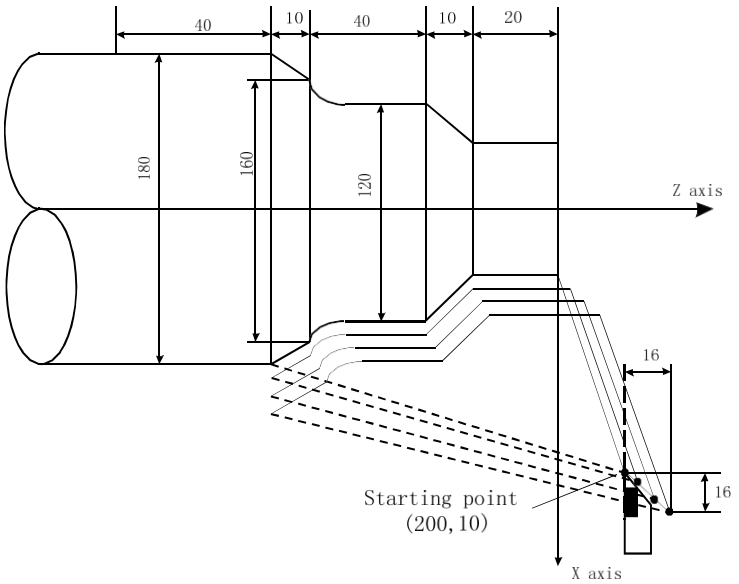


Fig.3-33

```
Program: O0006;
G99 G00 X200 Z10 M03 S500;        (Specify feedrate per rev and position starting point and start
                                    spindle)
G73 U1.0 W1.0 R3 ;                (X tool retraction with 2mm, Z 1mm)
G73 P14 Q19 U0.5 W0.3 F0.3 ;      (X roughing with 0.5 allowance and Z 0.3mm)
N14 G00 X80 W-40 ;
G01 W-20 F0.15 S600 ;
X120 W-10 ;
W-20 ;                            Blocks for finishing
G02 X160 W-20 R20 ; N19
G01 X180 W-10 ;
G70 P14 Q19 M30;                  (Finishing)
```

3.13.4 Finishing cycle G70

Command format: G70 P(ns) Q(nf) ;

Command function: The tool executes the finishing of workpiece from starting point along with the finishing path defined by ns ~ nf blocks. After executing G71, G72 or G73 to roughing, execute G70 to finishing and single cutting of finishing allowance is completed. The tool returns to starting point and execute the next block following G70 block after G70 cycle is completed.

ns: Block number of the first block of finishing path. nf: Block number of the last block of finishing path.

G70 path is defined by programmed one of ns ~ nf blocks. Relationships of relative position of ns, nf block in G70 ~ G73 blocks are as follows:

```
 . . . . . . . .
G71/G72/G73 …… ; N_
(ns) . . . . . .
 . . . . . . . .
        · F
        · S
        .
        .
N_(nf)……
        . . .
 G70 P(ns) Q(nf) ;
        . . .
```

Blocks for finishing path

Command specifications:

- ns ~ nf blocks in programming must be followed G70 blocks.
- F, S, T in ns ~ nf blocks are valid when executing ns ~ nf to command G70 finishing cycle.
- G96, G97, G98, G99, G40, G41, G42 are valid in G70;
- When G70 is executed, the system can stop the automatic run and manual traverse
- When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path.
- G70 cannot be executed in MDI, otherwise, the system alarms.
- There are no the same block number in ns~nf when compound cycle commands are executed repetitively in one program.
- The tool retraction point should be high or low as possible to avoid crashing the workpiece.

3.13.5 Axial grooving multiple cycle G74

Command format: G74 R(e) ;

G74 X/U____Z/W____P(Δi) Q(Δk) R(Δd) F__;

Command function: Axial (X axis) tool infeed cycle compounds radial discontinuous cutting cycle: Tool infeeds from starting point in radial direction(Z), retracts, infeeds again, and again and again, and last tool retracts in axial direction, and retracts to the Z position in radial direction, which is called one radial cutting cycle; tool infeeds in axial direction and execute the next radial cutting cycle; cut to end point of cutting, and then return to starting point (starting point and end point are the same one in G74), which is called one radial grooving compound cycle. Directions of axial tool infeed and radial tool infeed are defined by relative position between end point X/U Z/W and starting point of cutting. G74 is used for machining radial loop groove or column surface by radial discontinuously cutting, breaking stock and stock removal.

Relevant definitions:

Starting point of axial cutting cycle: starting position of axial tool infeed for each axial cutting cycle, defining with A_n(n=1,2,3……), Z coordinate of A_n is

the same that of starting point A, the different value of X coordinate between A_n and A_{n-1} is Δi. The starting point A_1 of the first axial cutting cycle is the same as the starting point A, and the X coordinate of starting point (A_f) of the last axial cutting cycle is the same that of cutting end point.

End point of axial tool infeed: starting position of axial tool infeed for each axial cutting cycle, defining with B_n(n=1,2,3……), Z coordinate of B_n is the same that of cutting end point, X coordinate of B_n is the same that of A_n , and the end point (B_f) of the last axial tool infeed is the same that of cutting end point.

End point of radius tool retraction: end position of radius tool infeed (travel of tool infeed is Δd) after each axial cutting cycle reaches the end point of axial tool infeed, defining with C_n(n=1,2,3……), Z coordinate of C_n is the same that of cutting end point, and the different value of X coordinate between C_n and A_n is Δd;

End point of axial cutting cycle: end position of axial tool retraction from the end point of radius tool retraction, defining with D_n(n=1, 2, 3……), Z coordinate of D_n is the same that of starting point, X coordinate of D_n is the same that of C_n (the different value of X coordinate between it and A_n is Δd);

Cutting end point: it is defined by X/U ___Z/W ___, and is defined with B_f of the last axial tool infeed.

R(e) : it is the tool retraction clearance after each axial(Z) tool infeed, and its range is 0~99.999(unit：mm) without sign symbols. The specified value is reserved validly after R(e) is executed and the data is switched to the corresponding value to save to NO.056. The NO.056 value is regarded as the tool retraction clearance when R(e) is not input.

X: X absolute coordinate value of cutting end point B_f (unit:mm).

U: Different value of X absolute coordinate between cutting end point B_f and starting point. Z: Z absolute coordinate value of cutting end point B_f (unit: mm).

W: Different value of Z absolute coordinates between cutting end point B_f and starting point.

P(Δi) ：radial(X) cutting for each axial cutting cycle , range: 0 < Δi≤ 9999999× least input increment (diameter value), without sign symbol.

Q(Δk)：radial(Z) cutting for each axial cutting cycle, range: 0 < Δk≤ 9999999× least input increment (diameter value), without sign symbol.

R(Δd) : radial (X) tool retraction after cutting to end point of axial cutting,range:0~99999.999× least input increment (diameter value) without sign symbol. The radial (X) tool retraction clearance is 0 when the system defaults the axial cutting end point. The system defaults the tool retraction is executed in positive direction when X/U and P(Δi) are omitted.

Execution process: (Fig. 3-34)

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

② Axial (Z) rapid tool retraction e and its direction is opposite to the feed direction of ①;

③ X executes the cutting feed (Δk+e) again, the end point of cutting feed is still in it between starting point A_n of axial cutting cycle and end point of axial tool infeed, Z executes the cutting feed (Δk+e)again and execute ②; after Z executing the cutting feed (Δk+e)again, the end point of cutting feed is on B_n or is not on it between A_n and

B_n cutting feed to B_n in Z direction and then execute ④;

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

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

⑥Radial(X axial)rapid tool infeed, and it direction is opposite to ④retract tool. If the end point of tool infeed is still on it between A and A_f (starting point of last axial cutting cycle) after X executes the tool infeed (△d+△i) (radius value) , i.e. $D_n \rightarrow A_{n+1}$ and then execute ① (start the next axial cycle); if X end point of tool infeed is not on it between D_n and A_f after tool infeed (△d+△i) (radius value), rapidly traverse to A_f and execute ①to start the first axial cycle;

⑦X rapidly traverse to return to A, and G74 is completed.

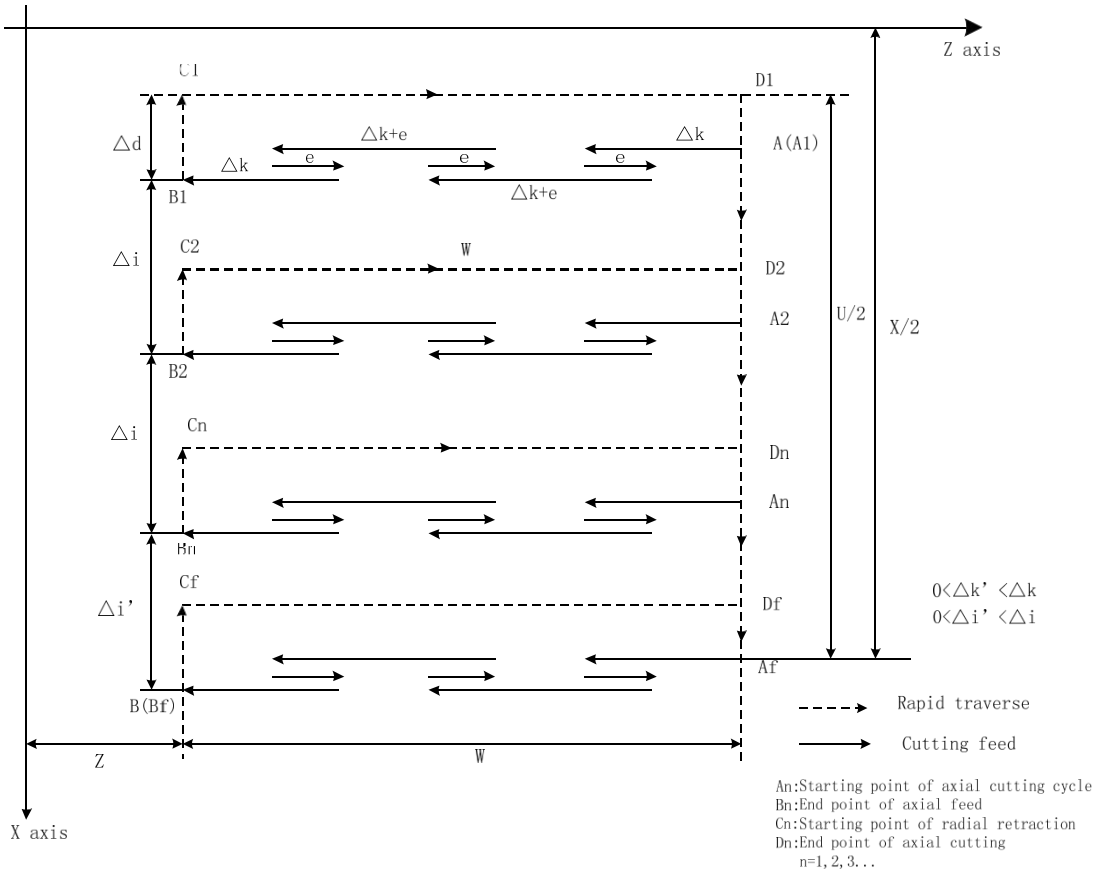


Fig. 3-34 G74 path

Command specifications:

- The cycle movement is executed by Z/W and P(Δk) blocks of G74, and the movement is not executed if only "G74 R(e) ; " block is executed;
- Δd and e are specified by the same address and whether there are Z/W and P(Δk) word or not in blocks to distinguish them;
- When G74 is executed, the system can stop the automatic run and manual traverse

- When the single block is running, programs dwell after each axial cutting cycle is completed.
- R(Δd) must be omitted in blind hole cutting, and so there is no distance of tool retraction when the tool cuts to axial end point of cutting.

Example：Fig. 3-35

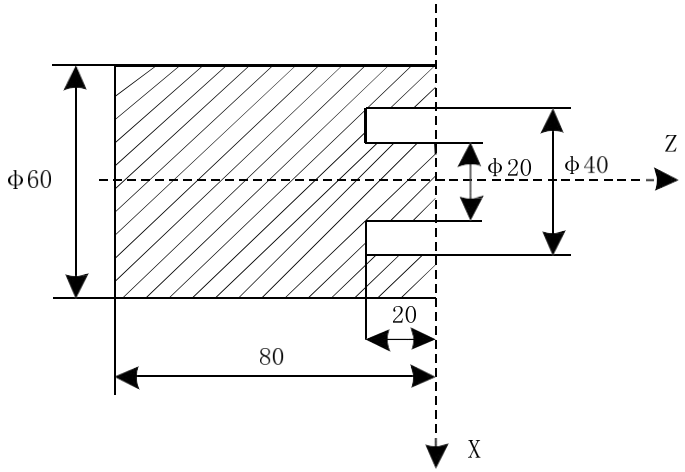


Fig.3-35

Program (suppose that the grooving tool width is 4mm, system least increment is 0.001mm): O0007;
G0 X40 Z5 M3 S500; (Start spindle and position to starting point of machining) G74 R0.5 ; (Machining cycle)
G74 X20 Z60 P3000 Q5000 F50; (Z tool infeed 5mm and tool retraction 0.5mm each time; rapid return to starting point (Z5) after cutting feed to end point (Z-20), X tool infeed 3mm and cycle the above-mentioned steps)
M30; (End of program)

3.13.6 Radial grooving multiple cycle G75

Command format：G75 R(e)；

G75 X/U___Z/W___P(Δi) Q(Δk) R(Δd) F__ ；

Command function: Axial (Z) tool infeed cycle compounds radial discontinuous cutting cycle: Tool infeeds from starting point in radial direction, retracts, infeeds again, and again and again, and last tool retracts in axial direction, and retracts to position in radial direction, which is called one radial cutting cycle; tool infeeds in axial direction and execute the next radial cutting cycle; cut to end point of cutting, and then return to starting point (starting point and end point are the same one in G75), which is called one radial grooving compound cycle. Directions of axial tool infeed and radial tool infeed are defined by relative position between end point X(U) Z(W) and starting point of cutting. G75 is used for machining radial loop groove or column surface by radial discontinuously cutting, breaking stock and stock removal.

Relevant definitions:

Starting point of radial cutting cycle: Starting position of axial tool infeed for each radial cutting cycle, defined by A_n(n=1, 2, 3……), X coordinate of A_n is the same that of starting point A, the different value of X

coordinate between A_n and A_{n-1} is $\underline{\Delta k}$. The starting point A_1 of the first radial cutting cycle is the same as the starting point A, and Z starting point (A_f) of the last axial cutting cycle is the same that of cutting end point.

End point of radial tool infeed: Starting position of radial tool infeed for each radial cutting cycle, defined by B_n (n=1, 2, 3……), X coordinates of B_n is the same that of cutting end point, Z coordinates of B_n is the same that of A_n, and the end point (B_f) of the last radial tool infeed is the same that of cutting end point.

End point of axial tool retraction: End position of axial tool infeed (travel of tool infeed is Δd) after each axial cutting cycle reaches the end point of axial tool infeed, defining with C_n(n=1,2,3……), X coordinate of C_n is the same that of cutting end point, and the different value of Z coordinate between C_n and A_n is Δd;

End point of radial cutting cycle: End position of radial tool retraction from the end point of axial tool retraction, defined by D_n(n=1, 2, 3……), X coordinate of D_n is the same that of starting point, Z coordinates of D_n is the same that of C_n (the different value of Z coordinate between it and A_n is Δd);

Cutting end point: It is defined by X/U ___Z/W ___, and is defined with B_f of the last radial tool infeed.

R(e) : It is the tool retraction clearance after each radial(X) tool infeed, its range is 0~99.999 (unit: mm, radius value) without sign symbols. The specified value is reserved validly after R(e) is executed and the data is switched and saved to No.056. NO.056 value is regarded as the tool retraction clearance when R(e) is not input.

X: X absolute coordinate value of cutting end point B_f (unit:mm).

U: Different value of X absolute coordinate between cutting end point B_f and starting point. Z: Z absolute coordinate value of cutting end point B_f (unit:mm).

W: Different value of Z absolute coordinate between cutting end point B_f and starting point A(unit: mm).

P($\underline{\Delta i}$) : Radial(X) discontinuous tool infeed of each axial cutting cycle, its range: 0 < Δi≤ 9999999 x least input increment without sign.

Q($\underline{\Delta k}$) : Axial(Z) discontinuous tool infeed of each radial cutting cycle, its range: 0 < $\underline{\Delta k}$≤ 9999999× least input increment without sign symbol.

R($\underline{\Delta d}$) : Axial (Z) tool retraction clearance after cutting to end point of radial cutting, its range: 0~99999.999×least input increment without sign symbol.

The system defaults the tool retraction clearance is 0 after the radial cutting end point is completed when R($\underline{\Delta d}$) is omitted.

The system defaults it executes the positive tool retraction when Z/W and Q($\underline{\Delta k}$) are omitted.

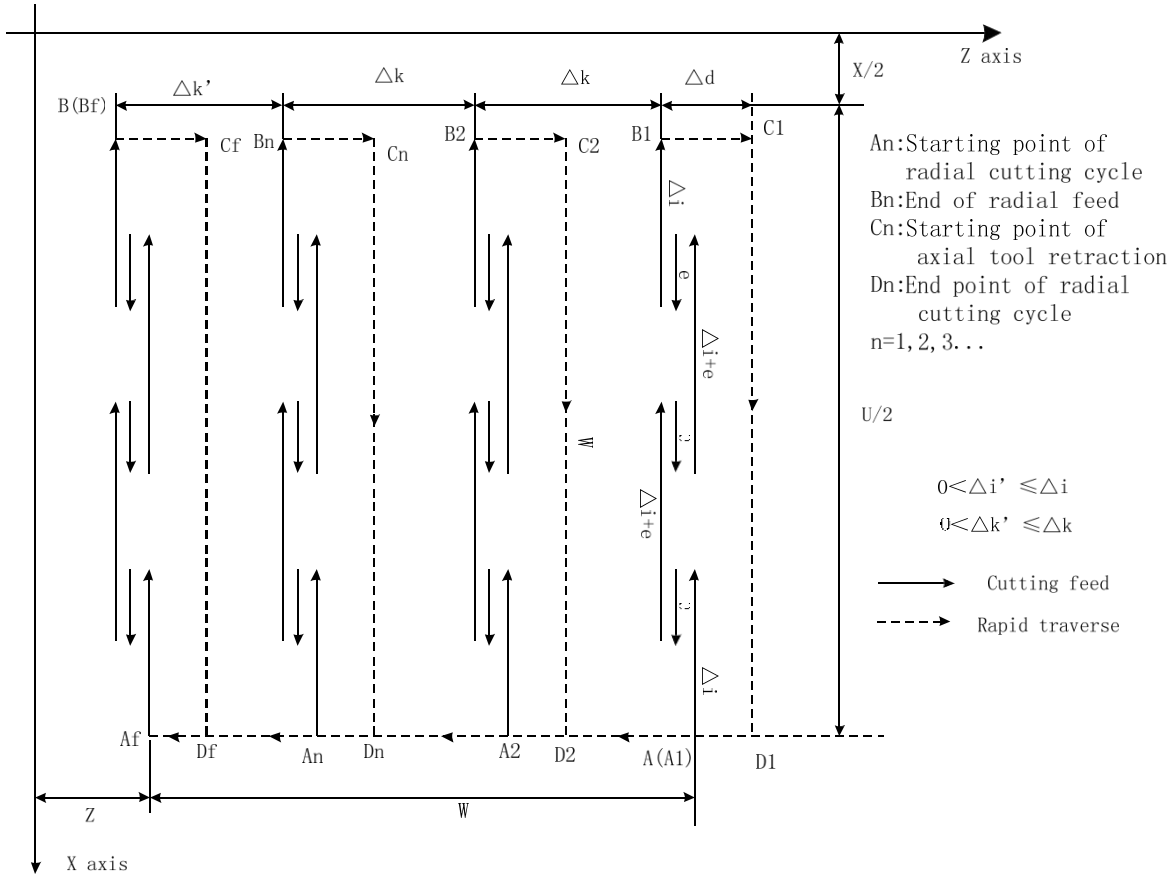


Fig. 3-36 G75 path

Execution process: (Fig. 3-36)

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

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

③ X executes the cutting feed (Δk+e) again, the end point of cutting feed is still in it between starting point A_n of radial cutting cycle and end point of radial tool infeed, X executes the cutting feed (Δi+e) again and executes ②; after X cutting feed (Δi+e) is executed again, the end point of X cutting feed is on B_n or is not on it between A_n and B_n cutting feed to B_n and then execute④;

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

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

⑥Axial(X) rapid tool infeed, and it direction is opposite to ④retract tool. If the end point of tool infeed is still on it between A and A_f (starting point of last radial cutting cycle) after Z tool infeed (△d+△k) (radius value), i.e. $D_n \rightarrow A_{n+1}$ and then execute ① (start the next radial cutting cycle); if the end point of tool infeed is not on it between D_n and A_f after Z tool infeed (△d+△k), rapidly traverse to A_f and execute ①to start the first radial cutting cycle;

⑦Z rapidly traverses to A, and G75 is completed.

Explanation:

- The cycle movement is executed by X/W and P(Δi) blocks of G75, and the movement is not executed if only "G75 R(e) ; " block is executed;
- Δd and e are specified by the same address R and whether there are X(U) and P(Δi) words or not in blocks to distinguish them;
- When G75 is executed, the system can stop the automatic run and manual traverse
- When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path;
- R(Δd) must be omitted in grooving, and so there is no tool retraction clearance whenthe tool cuts to radial cutting end point.

Example：Fig.3-37

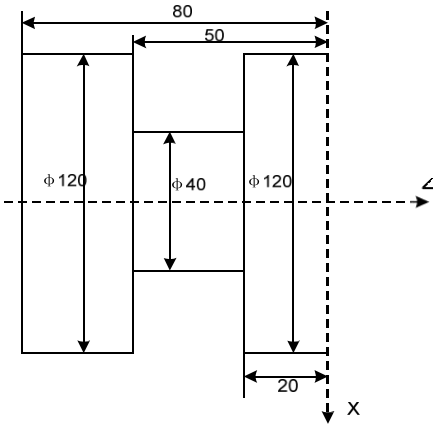


Fig. 3-37 G75 cutting

Program (suppose the grooving tool width is 4mm, the system least increment is 0.001mm): O0008;

G00 X150 Z50 M3 S500;	(Start spindle with 500 r/min)
G0 X125 Z-20;	(Position to starting point of machining)
G75 R0.5 F150;	(Machining cycle)
G75 X40 Z-50 P6000 Q3000;	(X tool infeed 6mm every time, tool retraction 0.5mm, rapid returning to starting point (X125) after infeeding to end point (X40), Z tool infeed 3mm and cycle the above-mentioned steps to continuously run programs)
G0 X150 Z50;	(Return to starting point of machining)
M30;	(End of program)

3.14 Thread cutting commands

TAC2000 CNC system can machine many kinds of thread cutting, including metric/inch single, multi threads, thread with variable lead and tapping cycle. Length and angle of thread run-out can be changed, multiple cycle thread is machined by single sided to protect tool and improve smooth finish of its surface. Thread cutting includes: continuous thread cutting G32, thread cutting with variable lead G34, Z thread cutting G33, Thread cutting cycle G92, Multiple thread cutting cycle G76.

The machine used for thread cutting must be installed with spindle encoder whose pulses are set by No.070m. Drive ratio between spindle and encoder is set by No.110 and No.111. X or Z traverses

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

The system can machine many kinds of thread cutting, such as thread cutting without tool retraction groove. There is a big error in the thread pitch because there are the acceleration and the deceleration at the starting and ending of X and Z thread cutting, and so there is length of thread lead-in and distance of tool retraction at the actual starting and ending of thread cutting.

X, Z traverse speeds are defined by spindle speed instead of cutting feedrate override in thread cutting when the pitch is defined. The spindle override control is valid in thread cutting. When the spindle speed is changed, there is error in pitch caused by X and Z acceleration/deceleration, and so the spindle speed cannot be changed and the spindle cannot be stopped in thread cutting, which will cause tool and workpiece to be damaged.

3.14.1 Thread cutting with constant lead G32

Command format: G32 X/U_ Z/W_ F(I)_ J_ K_ Q_

Command function: The path of tool traversing is a straight line from starting point to end point as Fig.3-84; the longer moving distance from starting point to end point(X in radius value) is called as the long axis and another is called as the short axis. In course of motion, the long axis traverses one lead when the spindle rotates one revolution, and the short axis and the long axis execute the linear interpolation. Form one spiral grooving with variable lead on the surface of workpiece to realize thread cutting with constant lead. Metric pitch and inch pitch are defined respectively by F, I. Metric or inch straight, taper, end face thread and continuous multi-section thread can by machined in G32.

Command specifications:

G32 is modal;

Pitch is defined to moving distance when the spindle rotates one rev(X in radius);

Execute the straight thread cutting when X coordinates of starting point and end point are the same one(not input X or U);

Execute the end face thread cutting when Z coordinates of starting point and end point are the same one(not input Z or W);

Execute the cutting taper thread when X and Z coordinates of starting point and end point are different;

Related definitions:

F: Metric pitch is moving distance of long axis when the spindle rotates one rev: 1 mm ~ 500 mm. After F is executed, it is valid until F with specified pitch is executed again.

I: Teeth per inch. It is ones per inch(25.4 mm) in long axis, and also is circles of spindle rotation when the long axis traverses one inch(25.4 mm) :0.06tooth/inch ~ 25400tooth/inch. After I is executed, it is valid until I with specified pitch is executed again. The metric, inch input both express the teeth per inch thread.

J: Movement in the short axis in thread run-out, negative sign; if the short axis is X, its value is specified with the radius; J value is the modal parameter.

K: Length in the long axis in thread run-out. If the long axis is X, its value is in radius without direction; K is modal parameter.

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

Q rules:

1. Its initial angle is 0° if Q is not specified;
2. For continuous thread cutting, Q specified by its following thread cutting block except for the first block is invalid, namely Q is omitted even if it is specified;
3. Multi threads formed by initial angle is not more than 65535;
4. Q unit : 0.001^0 . Q180000 is input in program if it offsets 180^0 with spindle one-turn; if Q180 or Q180.0, it is 0.18°.

Difference between long axis and short axis is shown in Fig. 3-38

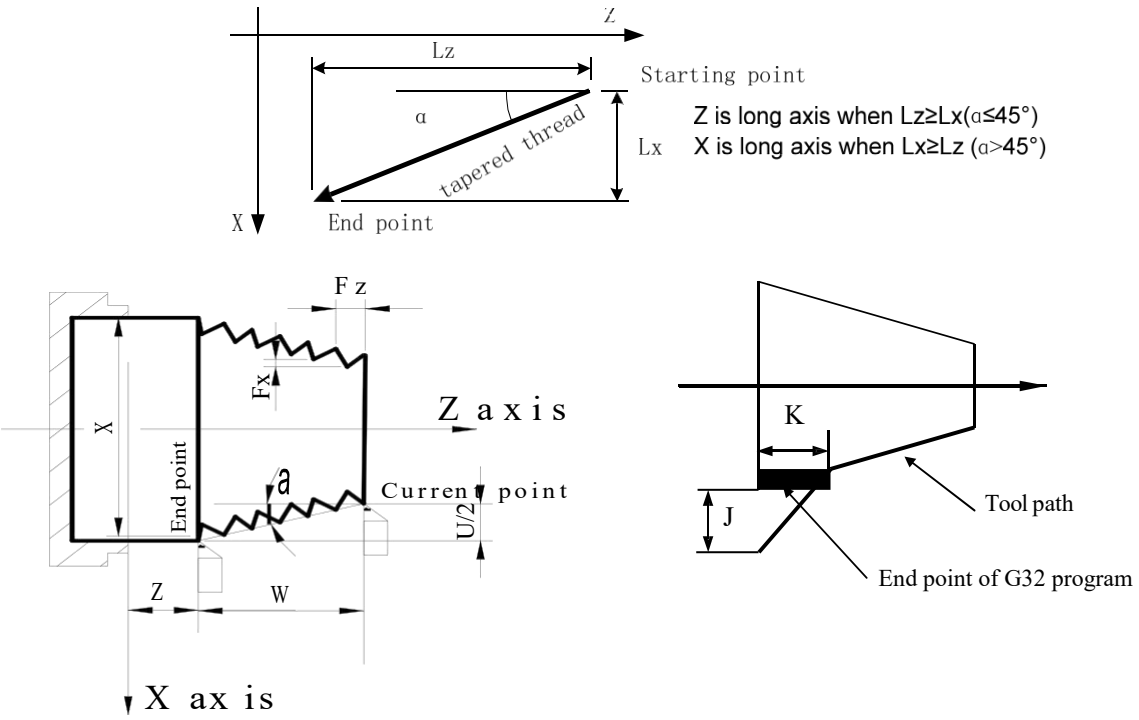


Fig. 3-38 G32 path

Notes:

- There is no thread run-out when J, or J, K are omitted; K=J is the thread run-out value when K is omitted;
- There is no thread run-out when J=0 or J=0, K=0;
- The thread run-out value J=K when J≠0,K=0;
- There is no thread run-out when J=0 or K≠0;
- If the current block is for thread and the next block is the same, the system does not test the spindle encoder signal per rev at starting the next block to execute the direct thread cutting, which function is called as continuous thread machining;
- After the feed hold is executed, the system displays “Pause” and the thread cutting continuously executes not to stop until the current block is executed completely; if the continuous thread cutting is executed, the program run pauses after thread cutting blocks are executed completely;
- In Single block, the program stops run after the current block is executed. The program stops running after all blocks for thread cutting are executed;

- The thread cutting decelerates to stop when the system resets, emergently stop or its drive unit alarms.

Example: Pitch: 2mm. δ1 = 3mm,δ2 = 2mm,total cutting depth 2mm divided into two times cut-in.

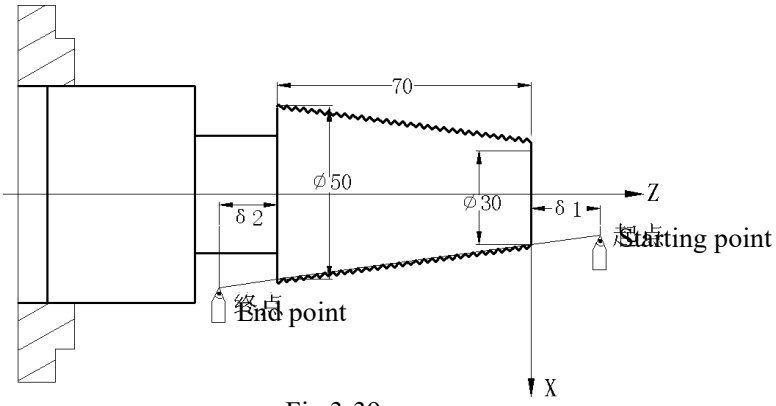


Fig.3-39

Program:

```
O0009;
G00 X28 Z3;          (First cut-in 1mm)
G32 X51 W-75 F2.0;   (First taper cutting)
G00 X55;             (Tool retraction)
W75;                 (Z returns to the starting point)
X27;                 (Second tool infeed 0.5mm)
G32 X50 W-75 F2.0;   (Second taper thread cutting )
G00 X55;             (Tool retraction)
W75 ;                (Z returns to the starting point)
M30;
```

3.14.2 Thread cutting with variable lead G34

Command format：G34 X/U_____Z/W_____F(I)____J_____K____R_____ ;

Command function: The motion path of tool is a straight line from starting point of X, Z to end point specified by the block, the longer moving distance from starting point to end point(X in radius value) is called as the long axis and another is called as the short axis. In course of motion, the long axis traverses one lead when the spindle rotates one rev, the pitch increases or decreases a specified value per rev and one spiral grooving with variable lead on the surface of workpiece to realize thread cutting with variable lead. Tool retraction can be set in thread cutting.

F, I are specified separately to metric, inch pitch. Executing G34 can machine metric or inch straight, taper, end face thread with variable pitch.

Command specifications:

G34 is modal;

Functions of X/U , Z/W , J, K are the same that of G32; F:
specifying lead, and its range: 0 ~ 500 mm;

I: Inch thread of first pitch from starting point: 0.06 tooth/inch ~ 25400 tooth/inch;

R: Increment or decrement of pitch per rev, R=F1- F2, with direction; F1>F2, pitch decreases when R is negative; F1<F2, pitch increases when R is positive (as Fig. 3-87);

R: ±0.001~±500.000 mm/pitch (metric thread);
±0.060~±25400 tooth/inch (inch thread).

The system alarms when R exceeds the above-mentioned range or the pitch exceeds permissive value or is negative owing to R increases or decreases.

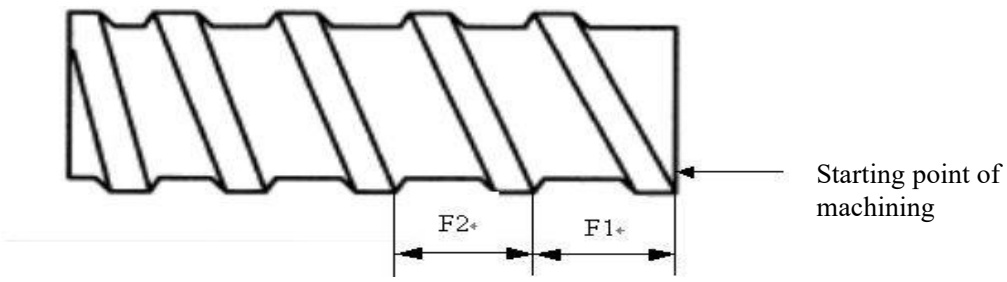
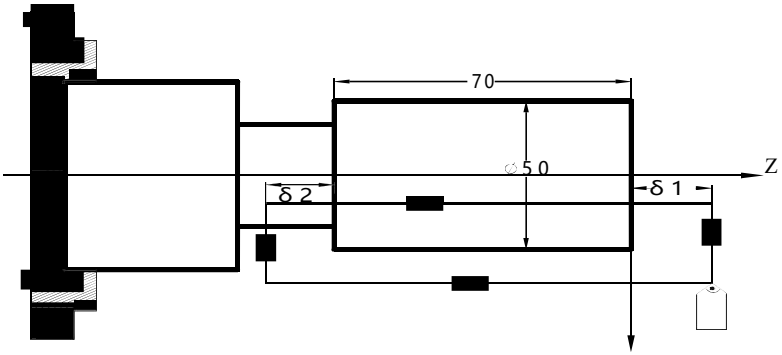


Fig. 3-40 Variable pitch thread

Note: It is the same as that of G32.

Example: First pitch of starting point**:** 4mm, increment 0.2mm per rev of spindle.



Program：O0010； G00 X60 Z4 M03 S500; G00 X48；x

Fig. 3-42 Variable pitch thread machining

G34 W-78 F3.8 J5 K2 R0.2;

N30 M30；

3.14.3 Z thread cutting G33

Command format：G33 Z/W F(I) L ；

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

Command specification: G33 is modal command;

Z/W : When Z or W is not input and starting point and end point of Z axis are the same one, the thread cutting must not be executed;

F: Thread pitch, and its range is referred to Table 1-4;

I: Teeth per inch thread 0.06～25400 teeth/inch; its range is referred to Table 1-4. L:The number ofmulti threads.Its range is 1~99. It is single thread when L is omitted. **Cycle process:**

① Z tool infeed (start spindle before G33 is executed);

② M05 signal outputs after Z reaches the specified end point in programming;

③ Test spindle after completely stopping;

④ Spindle rotation (CW) signal outputs(reverse to the original rotation direction);

⑤ Z executes the tool retracts to starting point;

⑥ M05 signal outputs and the spindle stops;

⑦ Repeat the steps ①～⑤ if multi threads are machined.

Example: Fig. 3-43 thread M10×1.5

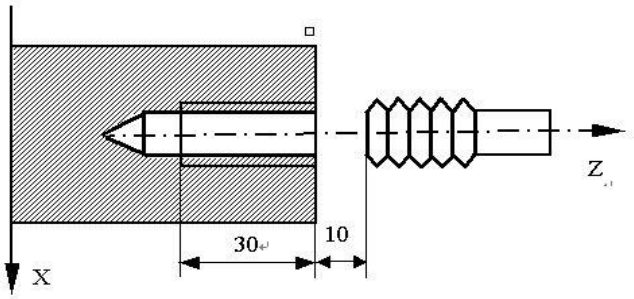


Fig. 3-43

Program:

O0011;

G00 Z90 X0 M03;	Start spindle
G33 Z50 F1.5;	Tap cycle
M03	Start spindle again
G00 X60 Z100;	Machine continuously M30

Note 1: Before tapping, define rotation direction of spindle according to tool rotating. The spindle stops rotation after the tapping is completed and the spindle is started again when machining thread continuously.

Note 2: G33 is for rigid tapping. The spindle decelerates to stop after its stop signal is valid, at the moment, Z executes continuously infeeds along with the spindle rotating, and so the actual cutting bottom hole is deeper than requirement and the length is defined by the spindle speed and its brake in tapping.

Note 3: Z rapid traverse speed in tapping is defined by spindle speed and pitch is not related to cutting feedrate override.

Note 4: In Single block to feed hold, the tapping cycle continuously executes not to stop until the tool returns to starting point when the system displays "Pause".

Note 5: The thread cutting decelerates to stop when the system resets, emergently stop or its driver alarms.

3.14.4 Thread cutting cycle G92

Command format: G92 X/U _ Z/W _ F_ J_ K_ L ; (Metric straight thread cutting cycle) G92 X/U _ Z/W _ I_ J_ K_ L ; (Inch straight thread cutting cycle) G92 X/U _ Z/W _ R_ F_ J_ K_ L ; (Metric taper thread cutting cycle) G92 X/U _ Z/W _ R_ I_ J_ K_ L ; (Metric taper thread cutting cycle)

Command function: Tool infeeds in radial(X) direction and cuts in axial(Z or X, Z) direction from starting point of cutting to realize straight thread, taper thread cutting cycle with constant thread pitch. Thread run-out in G92: at the fixed distance from end point of thread cutting, Z executes thread interpolation and X retracts with exponential or linear acceleration, and X retracts at rapidly traverse speed after Z reaches to end point of cutting as Fig. 3-94.

Command specifications:

G92 is modal;

Starting point of cutting: starting position of thread interpolation; End point of cutting: end position of thread interpolation;

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

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

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

R: Different value(radius value) of X absolute coordinate from end point to starting point of cutting. When the sign of R is not the same that of U, $R \le |U/2|$, unit:mm;

F: Thread lead, its range: $0 < F \le 500$ mm. After F value is executed, it is reserved and can be omitted;

I: Thread teeth per inch, its range: 0.06tooth/inch ~ 25400tooth/inch , it is reserved and it can be omitted not to input after I specified value is executed;

J: Movement in the short axis in thread run-out, its range 0~99999.999× least input increment without direction (automatically define its direction according to starting position of program), and it is modal parameter. If the short axis is X, its value is specified by radius;

K: Movement in the long axis in thread run-out, its range: 0~99999.999× least input increment without direction (automatically define its direction according to starting position of program), and it is modal parameter. If the long axis is X, its value is specified by radius;

L: Multi threads: 1 ~ 99 and it is modal parameter. (The system defaults it is single thread when L is omitted).

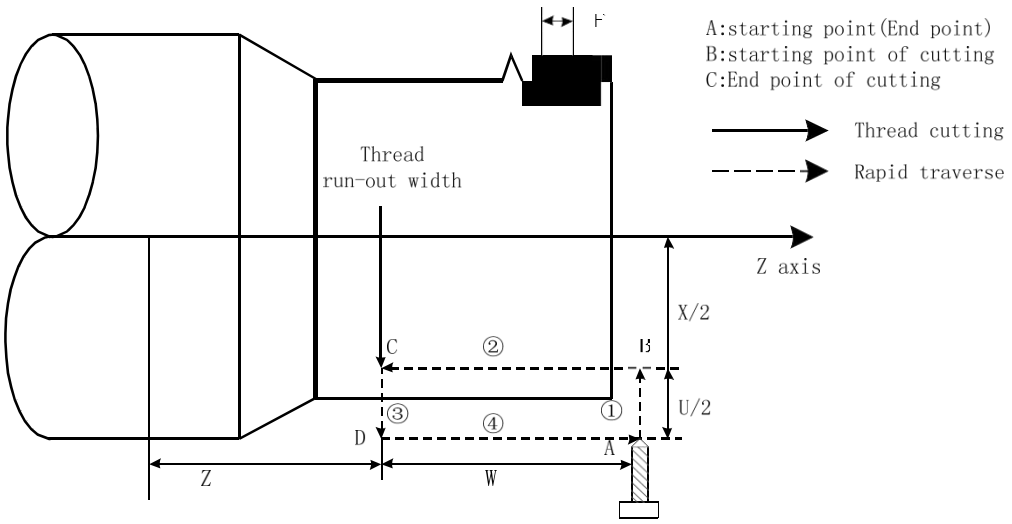


Fig. 3-44

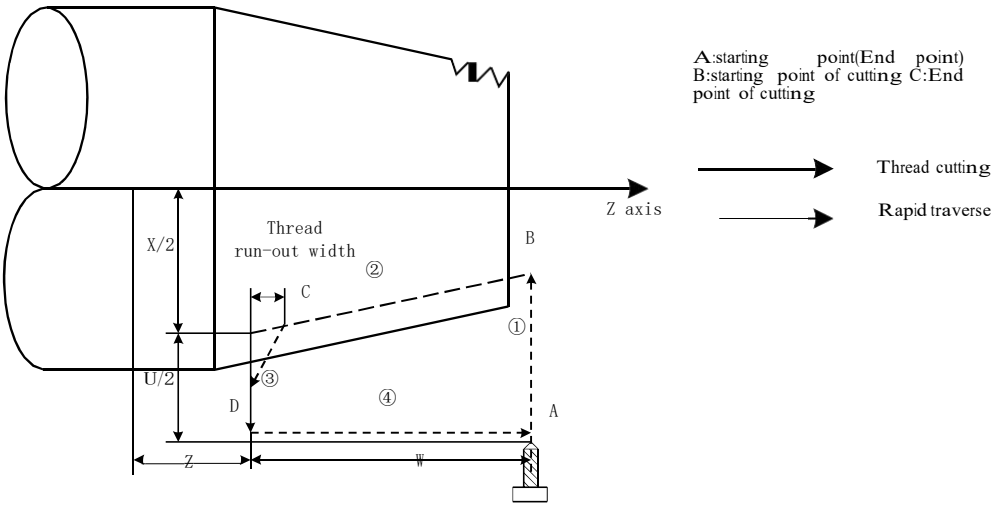


Fig. 3-45

The system can machine one thread with many tool infeed in G92, but cannot do continuous two thread and end face thread. Definition of thread pitch in G92 is the same that of G32, and a pitch is defined that it is a moving distance of long axis(X in radius) when the spindle rotates one rev.

Pitch of taper thread is defined that it is a moving distance of long axis(X in radius). When absolute value of Z coordinate difference between B point and C point is more than that of X (in radius), Z is long axis; and vice versa.

Cycle process: straight thread as Fig.3-44 and taper thread as Fig.3-45.

① X traverses from starting point to cutting starting point;

②Thread interpolates (linear interpolation) from the cutting starting point to cutting end point;

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

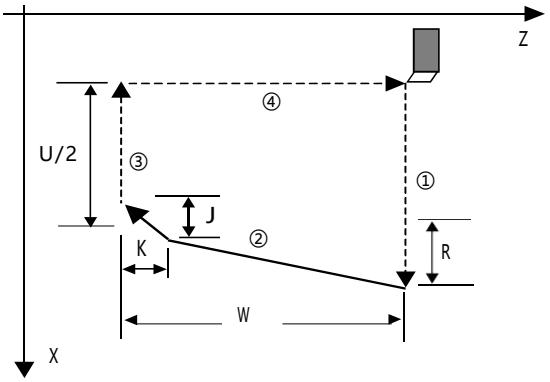
④ Z rapidly traverses to return to the starting point and the cycle is completed.

Notes：

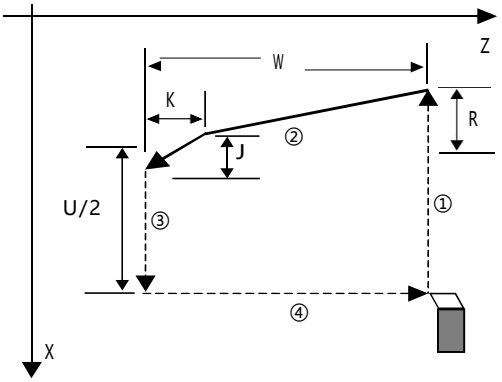
- Length of thread run-out is specified by №019 when J, K are omitted;
- Length of thread run-out is K in the long direction and is specified by №019 when J is omitted；
- Length of thread run-out is J=K when K is omitted;
- There is no thread run-out when J=0 or J=0, K=0;
- Length of thread run-out is J=K when J≠0,K=0;
- There is no thread run-out when J=0,K≠0;
- After executing the feed hold in thread cutting, the system does not stop cutting until the thread cutting is completed with ***Pause*** on screen;
- After executing single block in thread cutting, the program run stops after the system returns to starting point(one thread cutting cycle is completed);
- They are executed as the positive values when J, K negative values are input;
- Thread cutting decelerates to stop when the system resets, emergently stops or its driver alarms.

Command path: relative position between thread cutting end point and starting point with U, W, R and tool path and thread run-out direction with different U, W, R signs below:

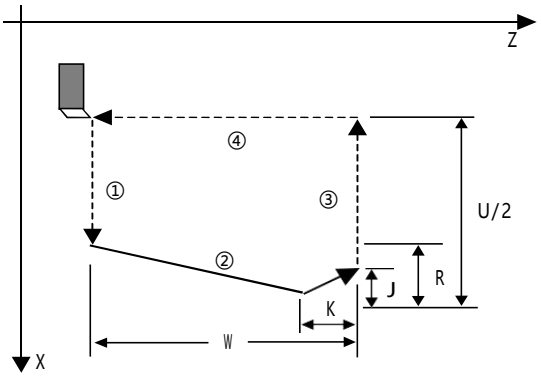
1） U>0，W<0，R>0



2） U<0，W<0，R<0



3） U>0，W>0，R<0，|R|≤|U/2|



4） U<0，W>0，R>0，|R|≤|U/2|

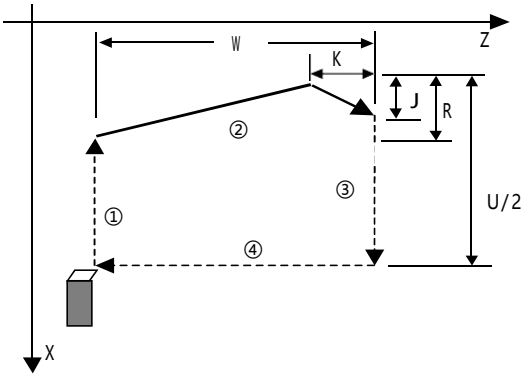


Fig.3-46

Example：Fig.3-47

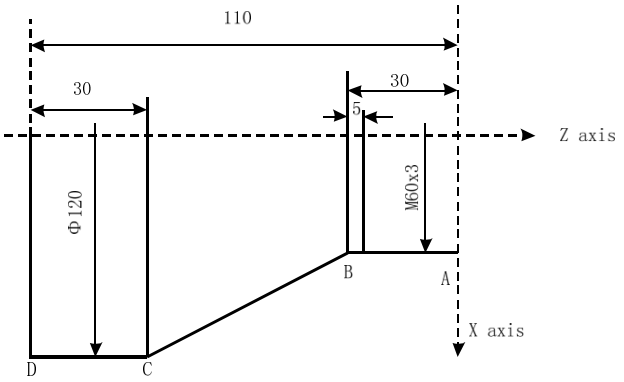


Fig.3-47

Program:

```
O0012;
M3 S300 G0 X150 Z50 T0101;      (Thread tool)
G0 X65 Z5;                      (Rapid traverse)
G92 X58.7 Z-28 F3 J3 K1;        (Machine thread with 4 times cutting, the first tool infeed
                                 1.3mm)
X57.7 ;                          (The second tool infeed 1mm)
X57;                             (The third tool infeed 0.7mm)
X56.9;                          (The fourth tool infeed 0.1mm)
M30;
```

3.14.5 Multiple thread cutting cycle G76

Command format: G76 P(m) (r) (a) Q(△dmin) R(d) ;

G76 X/U____Z/W____R(i) P(k) Q(△d)___F(I) ___;

Command function: Machining thread with specified depth of thread (total cutting depth)is completed by multiple roughing and finishing, if the defined angle of thread is not 0°, thread run-in path of roughing is from its top to bottom, and angle of neighboring thread teeth is the defined angle of thread. G76 can be used for machining the straight and taper thread with thread run-out path, which is contributed to thread cutting with single tool edge to reduce the wear of tool and to improve the precision of machining thread. But G76 cannot be used for machining the face thread. machining path is shown in Fig. 3-98:

Relevant definitions:

Starting point(end point): Position before block runs and behind blocks run, defined by A point; **End point of thread(D point):** End point of thread cutting defined by X/U __ Z/W ___.The tool will not reach the point in cutting if there is the thread run-out path;

Starting point of thread: Its absolute coordinates is the same that of A point and the different value of X absolute coordinates between C and D is i(thread taper with radius value). The tool cannot reach C point in cutting when the defined angle of thread is not 0°;

Reference point of thread cutting depth (B point) : Its absolute coordinates is the same that of A point and the different value of X absolute coordinate between B and C is k(thread taper with radius value).The cutting depth of thread at B point is 0 which is the reference point used for counting each thread cutting depth by the system;

Thread cutting depth: It is the cutting depth for each thread cutting cycle. It is the different value

(radius value, without signs) of X absolute coordinate between B and intersection of reversal extension line for each thread cutting path and straight line BC. The cutting depth for each roughing is $\sqrt{n}\times\triangle d$, n is the current roughing cycle times, △d is the thread cutting depth of first roughing;

Thread cutting amount: Different value between the current thread current depth and the previous one:($\sqrt{n}-\sqrt{n-1}$)×△d;

End point of tool retraction: It is the end position of radial (X) tool retraction after the thread cutting in each thread roughing, finishing cycle is completed, defining with E point;

Thread cut-in point: B_n (n is the cutting cycle times) is the actual thread cutting starting point in each thread roughing cycle and finishing cycle, B_1 is the first thread roughing cutting-in point, B_f is the last thread roughing cut-in point, B_e is the thread finishing cutting-in point. B_n is X, Z replacement formula corresponding to B.

$$tg\frac{\alpha}{2}=\frac{Z\ replacement}{X\ replacement}$$

a：thread angle；

X: X absolute coordinate (unit: mm) of thread end point;

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

Z: Z absolute coordinate (unit: mm) of thread end point;

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

P(m): Times of thread finishing: 00～99 (unit: times). It is valid after m specified value is executed, and the system parameter №057 value is rewritten to m. The value of system parameter №057 is regarded as finishing times when m is not input. In thread finishing, every feed cutting amount is equal to the cutting amount d in thread finishing dividing the finishing times m;

P(r): Width of thread run-out 00～99(unit: 0.1×L,L is the thread pitch). It is valid after r specified value is executed and the system parameter №019 value is rewritten to r. The value of system parameter №019 is the width of thread run-out when r is not input. The thread run-out function can be applied to thread machining without tool retraction groove and the width of thread run-out defined by system parameter №019 is valid for G92, G76;

P(a): Angles at taper of neighboring two tooth，range: 00～99，unit：deg(°). It is valid after a specified value is executed and the system parameter №058 value is rewritten to a. The system parameter №058 value is regarded as angle of thread tooth. The actual angle of thread in defined by tool ones and so a should be the same as the tool angle;

△Q(△dmin): Minimum cutting travel of thread roughing (unit: 0.001mm(IS-B) or 0.0001 mm(IS-C), radius value without sign symbols). When ($\sqrt{n}-\sqrt{n-1}$) ×△d < △dmin, △dmin is regarded as the cutting travel of current roughing, i.e. depth of current thread cutting is ($\sqrt{n-1}$ ×△d+△dmin) . Setting △dmin is to avoid the too small of roughing amount and too many roughing times caused by the cutting amount deceleration in thread roughing. After Q(△dmin) is executed, the specified value △dmin is valid and the system data parameter NO. 059 value is rewritten to △dmin (unit: 0.001). when Q(△dmin) is not input, the system data parameter NO.059 value is taken as the least cutting amount;

R(d): It is the cutting amount in thread finishing, range: 00 ～ 99.999 (unit:mm, radius value without sign symbols), the radius value is equal to X absolute coordinates between cut-in point Be of thread finishing and Bf of thread roughing. After R(d) is executed, the specified value d is reserved and the system parameter №060 value is rewritten to d×1000(unit: 0.001 mm) . The value of system parameter №060 is regarded as the cutting travel of thread finishing when R(d) is not input;

R(i): It is thread taper and is the different value of X absolute coordinate between thread starting point and end point, range: ±99999.999× least input increment (radius value). The system defaults R(i)=0(straight thread) when R(i) is not input;

P(k): Depth of thread tooth, the total cutting depth of thread, range: 1～99999999× least input increment (radius value without sign symbols). The system alarms when P(k) is not input;

Q(△d): Depth of the 1st thread cutting, range: 1～99999999× least input increment (radius value without sign symbols). The system alarms when △d is not input;

F: metric thread lead, its range :0~500mm

I:thread teeth per inch for inch thread,its range:0.06~25400teeth/per inc

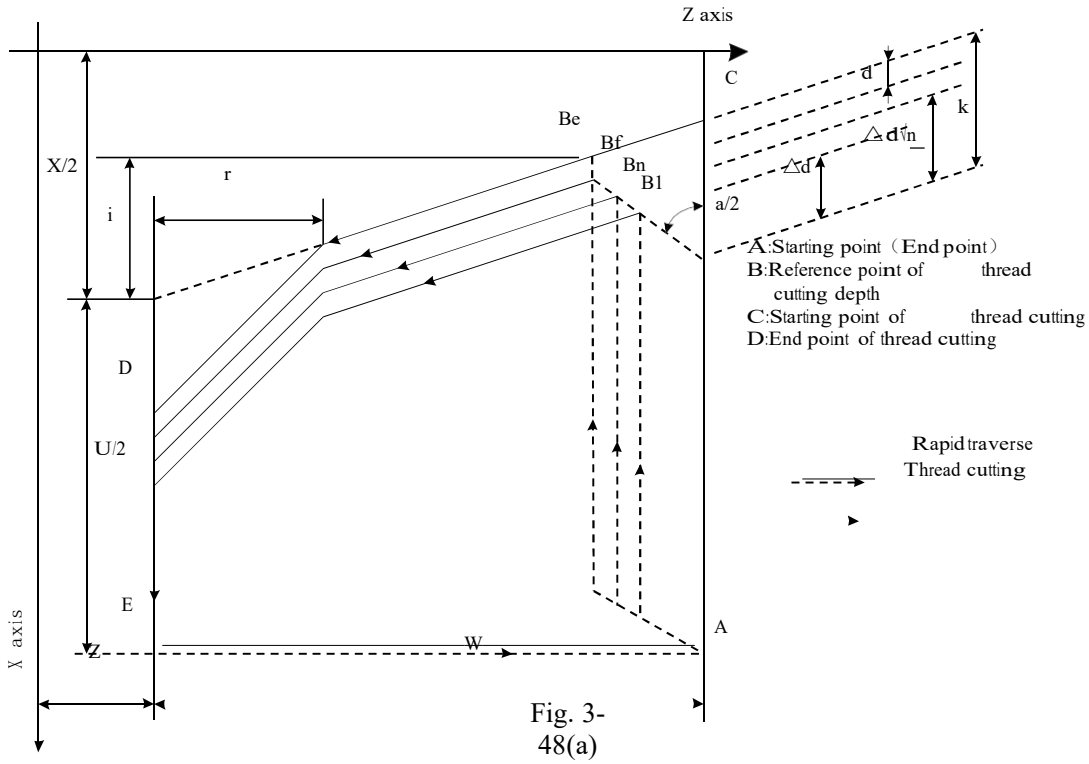


Fig. 3-48(a)

Cut-in method as follows: Fig. 3-48(b):

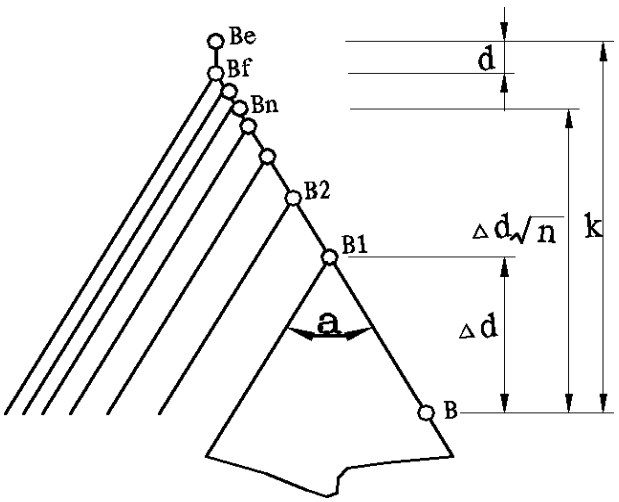


Fig. 3-48(b)

Pitch is defined to moving distance (X radius value) of long axis when the spindle rotates one rev. Z is long when absolute value of coordinate difference between C point and D point in Z direction is more than that of X direction (radius value, be equal to absolute value of i); and vice versa

Execution process:

① The tool rapidly traverses to B_1, and the thread cutting depth is △d. The tool only traverses in X direction when a=0; the tool traverses in X and Z direction and its direction is the same that of A→D when a≠0;

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

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

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

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

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

⑦ X axis rapidly traverses to E point;

⑧ Z axis traverses to A point and the thread roughing cycle is completed to execute the finishing;

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

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

Notes:

- In thread cutting, execute the feed hold, the system displays ***Pause*** after the thread cutting is executed completely, and then the program run pauses;
- Execute single block in thread cutting, the program run stops after returning to starting point(one thread cutting cycle is completed);
- The thread cutting decelerates to stop when the system resets and emergently stop or the driver alarms;
- Omit all or some of G76 P(m) (r) (a) Q(△d_{min}) R(d) . The omitted address runs according to setting value of parameters;
- m, r, a used for one command address P are input one time. Program runs according to setting value of №57, 19, 58 when m, r, a are all omitted; Setting value is a when address P is input with 1 or 2 digits; setting values are r, a when address P is input with 3 or 4 digits;
- The direction of A→C→D→E is defined by signs of U,W , and the direction of C→D is defined by the sign of R(i) . There are four kinds of sign composition of U, W corresponding to four kinds of machining path as Fig. 3-100.

Example: Fig. 3-49, thread M68×6.

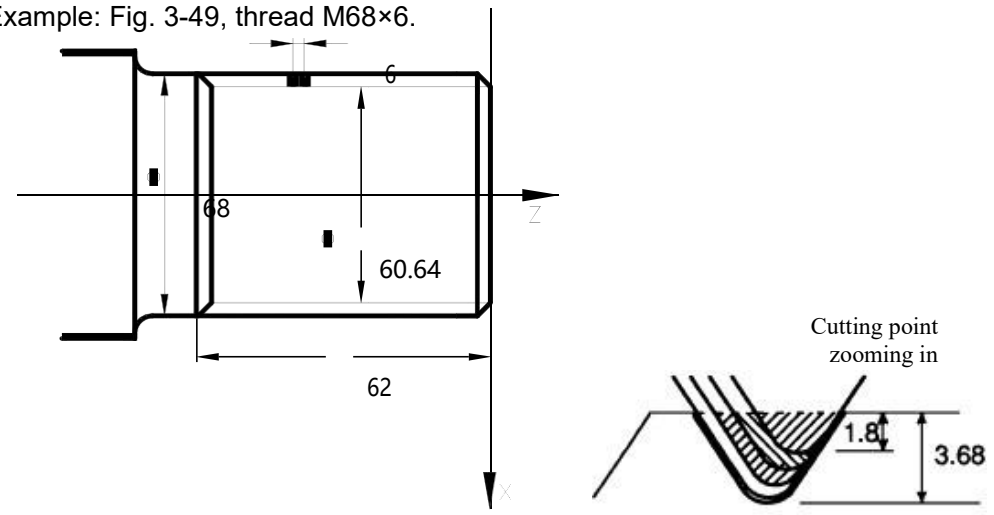


Fig.3-49

Program:

```
O0013;
G50  X100  Z50 M3  S300;          (Set workpiece coordinate system, start spindle
                                  and specify spindle speed)
G00 X80 Z10;                      (Rapid traverse to starting point of machining)
G76 P020560  Q150  R0.1;          (Finishing 2 times, chamfering width 0.5mm, tool
                                  angle 60°, min. cutting depth 0.15, finishing allowance
                                  0.1)
G76 X60.64 Z-62 P3680 Q1800 F6; (Tooth height 3.68, the first cutting depth 1.8) G00
X100 Z50 ;                        (Return to starting point of program)
M30;                              (End of program)
```

3.15 Constant surface speed control G96, constant rotational speed control G97

The detailed is referred to Chapter 2.2.3.

3.16 Feedrate per minute G98, feedrate per rev G99

Command format: G98 F_; (the leading zero can be omitted, feed rate per minute is specified)

Command function: cutting feed rate is specified as mm/min, G98 is the modal G command. G98 cannot be input if the current command is G98 modal.

Command format: G99 F_; (its range is referred to Table 1-4, the leading zero can be omitted)

Command function: Cutting feed rate is specified as mm/min, G99 is the modal G command. G99 input may be omitted if current state is G99. The actual cutting feedrate is gotten by multiplying the F command value (mm/r) to the current spindle speed(r/min). If the spindle speed varies, the actual feedrate changes too. If the spindle cutting feed amount per rev is specified by G99 F_____, the even cutting texture on the surface of

workpiece will be gotten. In G99 state, a spindle encoder should be fixed on the machine tool to machine the workpiece.

G98, G99 are the modal G commands in the same group and only one is valid. G98 is the initial state G command and the system defaults G98 is valid when the system turns on.

Reduction formula of feed between per rev and per min: $F_m = F_r \times S$

F_m: feed per min (mm/min) ;

F_r: feed per rev (mm/r) ; S: spindle speed (r/min) .

After the system turns on, the feedrate is ones set by №076 and F value is reserved after F is executed. The feed rate is 0 after F0 is executed. F value is reserved when the system resets and emergently stops. The feedrate override is reserved when the system is turned off.

Note: In G99 modal, there is the uneven cutting feed rate when the spindle speed is lower than 1 r/min; there is the follow error in the actual cutting feed rate when there is the swing in the spindle speed. To gain the high machining quality, it is recommended that the selected spindle speed should be not lower than min. speed of spindle servo or converter.

Related parameters:

System parameter No.027: the upper limit value of cutting feedrate(they are the same in X, Z direction, diameter/min in X direction);

System parameter No.029: exponential function for time constant of acceleration/deceleration when cutting feed and manual feed;

System parameter No.030: initial (ultimate) speed of acceleration/deceleration in exponential function when cutting feed and manual feed.

3.17 Macro commands

TAC2000 provides the macro command which is similar to the high language, and can realize the variable assignment, and subtract operation, logic decision and conditional jump by user macro command, contributed to compiling part program for special workpiece, reduce the fussy counting and simplify the user program.

3.17.1 MACRO variables

- **Presentation of macro variables**

  Present with "#" + macro variables number.;

  Format: # i(i=100,102,103,……) ;

  Example: #105, #109, #125.

- **Variable Type**

  The variable is divided into four types according to the variable number:

Number NO.	Variable type	Function
#0	Null variable	The variable is null and is not valued.
#1~#50	Local variable	The local variable is used to store data in the macro program, such as result. When the system is turned off, the local variable is initialized to be null. When the macro program is called, the argument values to the local.
#100~#199 #500~#999	Share variable	The share variable has the same meaning in the different macro program. When the system is turned off, the variable #100~#199 is initialized to be null, #500~#999 is saved and is not lost.
#1000~#5235	System variable	System variable

- **Macro variables reference**

1. Macro variables can replace command values

Format: ‹ Address › +"# i" or ‹ Address › +" - # I". It shows the system takes variable value or negative value of variable value as address value.

Example: F#103…when #103=15, its function is the same that of F15;

Z-#110…when #110=250, its function is the same that of Z-250;

Note 1: The address O, G and N cannot refer macro variables. For example, O#100，G#101，N#120 are illegal;

Note 2: If macro variables values exceed the maximum rang of command values, they cannot be used. For example: #130 = 120, M#130 exceeds the maximum command value.

- **Null variable**

When the variable value is not defined, it is null, the variable #0 is always null and only is read instead of writing.

a. Reference

When an undefined variable (null variable) is referred, the address is ignored.

#1=<null>	#1=0
G00 X100 Z#1 is equal to G00 X100	G00 X100 Z#1 is equal to G00 X100 Z0

● **Variable display**

```
MACRO                                                    O0099 N0000
 NO.      DATA      NO.      DATA      NO.      DATA
 _100    123.123    110                120
 _101    ********   111      2.001     121        120
  102               112                122
  103               113                123
  104          0    114      4.002     124
  105               115                125
  106               116                126
  107               117                127
  108               118       1000     128
  109          1    119                129
NO.  100
MDI                                                      S0000 T0101
```

(1) In macro window, the variable being displayed to the null means it is null, i.e. it is not defined.

(2) The share variable (#100~#199, #500~#999) values are displayed in the macro variable window, and is also displayed the window, the data is input directly to value the share variable.

(3) The local variable (#1~#50) and the system variable values are not displayed. Some local variable or system variable value is displayed by assigned with the share variable.

● **System variable**

(1)Interface signal: CNC only executes G and F signals. Whether there are I/O to correspond to it is defined by PLC.

Variable No.	Function
#1000~#1015	Correspond G54.0~G54.7, G55.0~G55.7 signal states
#1032	Correspond G54, G55 signal states
#1100~#1115	Correspond F54.0~G54.7, F55.0~F55.7 signal states
#1132	Correspond F54, F55 signal states
#1133	Correspond F56, F57, F58, F59 signal states

(2) Tool compensation system variable:

Compensation No.	Offset compensation value				Wear compensation value			
	X	Z	Y	radius	X	Z	Y	radius
1	#1500	#1600	#1700	#1800	#1900	#2000	#2100	#2200
…	…	…	…	…	…	…	…	…
32	#1531	#1631	#1731	#1831	#1931	#2031	#2131	#2231

(3) System modal information variable

Variable No.	Function	
#4001	G00, G01, G02, G03, G32, G33, G34, G80, G84, G88, G90, G92, G94	No. 1 group
#4002	G96, G97	No. 2 group
#4003	G98, G99	No. 3 group
#4005	G54,G55,G56,G57,G58,G59	No. 5 group
#4006	G20, G21	No. 6 group
#407	G40,G41,G42	No. 7 group
#4016	G17,G18,G19	No. 16 group
#4120	F command	
#4121	M command	
#4122	Serial No.	
#4123	Program No.	
#4119	S command	
#4120	T command	

(4) system variable of coordinate position information:

Variable No.	Position signal	Coordinate system	Tool compensation value	Read in running
#5001~#5005	End point of block	Workpiece coordinate system	Not including	Possible
#5006~#5010	Current position (Machine coor.)	Machine coordinate system	Including	impossible
#5011~#5015	Current position (Abs. coordinate)	Workpiece coordinate system		

Note: The position listed in the above table separately corresponds orderly to X, Y, Z, 4th, 5th axis. For example: #5001 meanings to be X position information, #5002 meanings to be Y position information, #5003 meanings to be Z position information and #5004 meanings to 4th position information and #5005 meanings to 5th position information.

(5) Workpiece zero offset value and Workpiece coordinate system: Basic offset value:

G54：#5201 ～#5205
G54：#5206 ～#5210
G55：#5211 ～#5215
G56：#5216 ～#5220
G57：#5221 ～#5225
G58：#5226 ～#5230
G59：#5231 ～#5235

Local variable

- The relation of adress and local variable:

Variable adress	Local variable	Variable adress	Local variable	Variable adress	Local variable
A	#1	E	#8	U	#21
B	#2	F	#9	V	#22
C	#3	M	#13	W	#23
I	#4	Q	#17	X	#24
J	#5	R	#18	Y	#25
K	#6	S	#19	Z	#26
D	#7	T	#20		

3.17.2 Operation and jump command G65

Command format:

G65 Hm P# i Q# j R# k;

m: operation or jump command

I: macro variables name for storing values.

j: macro variables name 1 for operation, can be constant. # k:

macro variables name 2 for operation, can be constant.

Command significance: # i = #j O # k

Operation sign specified by Hm

Example: P#100 Q#101 R#102…..#100 = #101 O #102;

P#100 Q#101 R15….#100 = #101 O 15;

P#100 Q-100 R#102…..#100 = -100 O #102;

Note: Macro variable name has no "#" when it is presented directly with constant.

Macro command list

Command format	Functions	Definitions
G65 H01 P#i Q#j	Assignment	# i = # j assign value of j to i
G65 H02 P#i Q#j R#k;	Decimal add operation	# i = # j + # k
G65 H03 P#i Q#j R#k;	Decimal subtract operation	# i = # j - # k
G65 H04 P#i Q#j R#k;	Decimal multiplication operation	# i = # j×# k
G65 H05 P#i Q#j R#k;	Decimal division operation	# i = # j÷# k
G65 H11 P#i Q#j R#k;	Binary addition	# i = # j OR # k
G65 H12 P#i Q#j R#k;	Binary multiplication(operation)	# i = # j AND # k
G65 H13 P#i Q#j R#k;	Binary exclusive or	# i = # j XOR # k
G65 H21 P#i Q#j;	Decimal square root	# i = # j
G65 H22 P#i Q#j;	Decimal absolute value	# i = |# j |
G65 H23 P#i Q#j R#k;	Decimal remainder	Remainder of # i = (#j÷# k)
G65 H24 P#i Q#j;	Decimal into binary	# i = BIN(# j)
G65 H25 P#i Q#j;	Binary into decimal	# i = DEC(# j)
G65 H26 P#i Q#j R#k;	Decimal multiplication/division operation	# i = # i×# j÷# k
G65 H27 P#i Q#j R#k;	Compound square root	# i = $\#j^2+\#k^2$
G65 H31 P#i Q#j R#k;	Sine	# i = # j×sin(# k)
G65 H32 P#i Q#j R#k;	Cosine	# i = # j×cos(# k)
G65 H33 P#i Q#j R#k;	Tangent	# i = # j×tan(# k)
G65 H34 P#i Q#j R#k;	Arc tangent	# i = ATAN(# j / # k)
G65 H80 Pn;	Unconditional jump	Jump to block n
G65 H81 Pn Q#j R#k;	Conditional jump 1	Jump to block n if # j = # k,otherwise the system executes in order
G65 H82 Pn Q#j R#k;	Conditional jump 2	Jump to block n if # j ≠ # k, otherwise the system executes in order
G65 H83 Pn Q#j R#k;	Conditional jump 3	Jump to block n if # j > # k, otherwise the system order executes in

Command format	Functions	Definitions
G65 H84 Pn Q#j R#k;	Conditional jump 4	Jump to block n if # j < # k, otherwise the system executes in order
G65 H85 Pn Q#j R#k;	Conditional jump 5	Jump to block n if # j ≥ # k, otherwise the system executes in order
G65 H86 Pn Q#j R#k;	Conditional jump 6	Jump to block n if # j ≤# k, otherwise the system executes in order
G65 H99 Pn;	P/S alarm	(500+n) alarms

1 Operation commands

1) Assignment of macro variables: # I = # J

G65 H01 P#I Q#J

(Example) G65 H01 P# 101 Q1005; (#101 = 1005) G65 H01 P#101 Q#110; (#101 = #110)
G65 H01 P#101 Q-#102; (#101 = -#102)

2) Decimal add operation: # I = # J+# K

G65 H02 P#I Q#J R#K

(Example) G65 H02 P#101 Q#102 R15; (#101 = #102+15)

3) Decimal subtract operation: # I = # J - # K

G65 H03 P#I Q#J R# K

(Example) G65 H03 P#101 Q#102 R#103; (#101 = #102 - #103)

4) Decimal multiplication operation: # I = # J×# K

G65 H04 P#I Q#J R#K

(Example) G65 H04 P#101 Q#102 R#103; (#101 = #102×#103)

5) Decimal division operation: # I = # J÷# K

G65 H05 P#I Q#J R#K

(Example) G65 H05 P#101 Q#102 R#103; (#101 = #102÷#103)

6) Binary logic add(or) : # I = # J.OR. # K

G65 H11 P#I Q#J R#K

(Example) G65 H11 P#101 Q#102 R#103; (#101 = #102.OR. #103)

7) Binary logic multiply(and) : # I = # J.AND. # K

G65 H12 P#I Q#J R#K

(Example) G65 H12 P# 201 Q#102 R#103; (#101 = #102.AND.#103)

8) Binary executive or: # I = # J.XOR. # K

G65 H13 P#I Q#J R#K

(Example) G65 H13 P#101 Q#102 R#103; (#101 = #102.XOR. #103)

9) Decimal square root: # I = $\sqrt{\#J}$

G65 H21 P#I Q#J

(Example) G65 H21 P#101 Q#102 ; (#101 = $\sqrt{\#102}$)

10) Decimal absolute value: # I = | # J |

G65 H22 P#I Q#J

(Example) G65 H22 P#101 Q#102 ; (#101 = | #102 |)

11) Decimal remainder: # I = # J - TRUNC(#J/#K)×# K,TRUNC: omit decimal fraction

G65 H23 P#I Q#J R#K

(Example) G65 H23 P#101 Q#102 R#103; (#101 = #102- TRUNC (#102/#103)×#103

12) Decimal converting into binary: # I = BIN (# J)

G65 H24 P#I Q#J

(Example) G65 H24 P#101 Q#102 ; (#101 = BIN(#102))

13) Binary converting into decimal: # I = BCD (# J)

G65 H25 P#I Q#J

(Example) G65 H25 P#101 Q#102 ; (#101 = BCD(#102))

14) Decimal multiplication/division operation: # I =(# I×# J) ÷# K

G65 H26 P#I Q#J R# k

(Example) G65 H26 P#101 Q#102 R#103; (#101 =(# 101×# 102) ÷#103)

15) Compound square root: # I = $\sqrt{\#J^2+\#K^2}$

G65 H27 P#I Q#J R#K

(Example) G65 H27 P#101 Q#102 R#103; (#101 = $\sqrt{\quad^2 \quad ^2}$)

16) Sine: # I = # J•SIN(# K) (Unit: ‰)

G65 H31 P#I Q#J R#K

(Example) G65 H31 P#101 Q#102 R#103; (#101 = #102•SIN(#103))

17) Cosine: # I = # J•COS(# K) (Unit: ‰)

G65 H32 P#I Q#J R# k

(Example) G65 H32 P#1Q#102 R#103; (#101 =#102•COS(#103))

18) Tangent: # I = # J•TAM(# K) (Unit: ‰)

G65 H33 P#I Q#J R# K

(Example) G65 H33 P#101 Q#102 R#103; (#101 = #102•TAM(#103))

19) Cosine: # I = ATAN(# J /# K) (Unit: ‰)

G65 H34 P#I Q#J R# k

(Example) G65 H34 P#101 Q#102 R#103; (#101 =ATAN(#102/#103))

2 Jump commands

1) Unconditional jump

G65 H80 Pn; n: Block number

(Example) G65 H80 P120; (jump to N120)

2) Conditional jump 1 #J.EQ.# K (=)

G65 H81 Pn Q#J R# K; n: Block number

(Example) G65 H81 P1000 Q#101 R#102;

The program jumps N1000 when # 101= #102 and executes in order when #101 ≠#102.

3) Conditional jump 2 #J.NE.# K (≠)

G65 H82 Pn Q#J R# K; n: Block number

(Example) G65 H82 P1000 Q#101 R#102;

The program jumps N1000 when # 101 ≠ #102 and executes in order when #101 = #102.

4) Conditional jump 3 #J.GT.# K (>)

G65 H83 Pn Q#J R# K; n: Block number

(Example) G65 H83 P1000 Q#101 R#102;

The program jumps N1000 when # 101 > #202 and executes in order when #101 ≤ #102.

5) Conditional jump 4 #J.LT.# K (< =)

G65 H84 Pn Q#J R# K; n: Block number

(Example) G65 H84 P1000 Q#101 R#102;

The program jumps N1000 when # 101 < #102 and executes in order when #101≥#102.

6) Conditional jump 5 #J.GE.# K (≥)

G65 H85 Pn Q#J R# K; n: Block number

(Example) G65 H85 P1000 Q#101 R#102;

The program jumps N1000 when # 101 ≤ #1 and executes in order when #101 < #102.

7) Conditional jump 6 #J.LE.# K (≤)

G65 H86 Pn Q#J R# K; n: Block number

(Example) G65 H86 P1000 Q#101 R#102;

8) P/S alarm

G65 H99 Pi; i: alarm number +500

(Example) G65 H99 P15; P/S alarm 515.

Note: Block number can be specified by variables. Such as: G65 H81 P#100 Q#101 R#102;
The program jumps to block that its block number is specified by #100.

3.17.3 Program example with macro command

Differences between user macro program call (G65, G66) and subprogram call (M98) are as follows:

1. G65 can specify the argument data and send them to macro program and M98 has no such function.
2. G65 can change the level of local variable and M98 has no such function.
3. G65 only follows N and only P or H follows them.

z **Non-modal call (G65)**

Command format: G65 P_ L_ <argument>_;

Macro program specified by P is called, the argument(data) is send to the user macro program body.

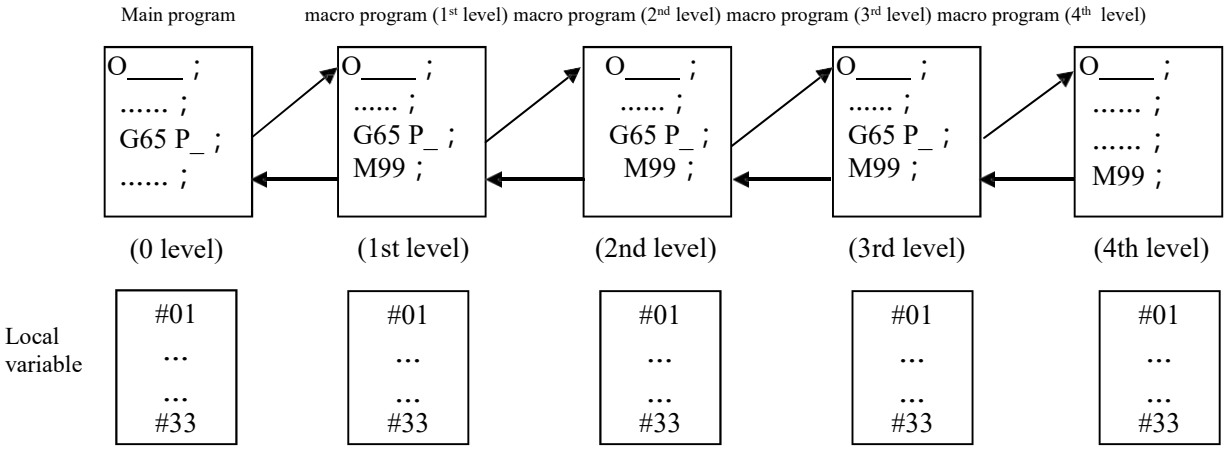
Command explanation:

P —— called macro program number

L —— called times (it is 1 when it is omitted, it can be the repetitive times from 1 to 9999)

<argument> —— data sent to macro program is valued with the corresponding local variable.

Nest call: G65 call has four-level nest.



Specifying argument:

use the letter besides G, L, O, N, P, and each is only specified one time, and the last which is specified many times is valid.

Argument address and corresponding variable No. table in method 1

Address	Variable No.		Address	Variable No.		Address	Variable No.
A	#1		I	#4		T	#20
B	#2		J	#5		U	#21
C	#3		K	#6		V	#22
D	#7		M	#13		W	#23
E	#8		Q	#17		X	#24
F	#9		R	#18		Y	#25
H	#11		S	#19		Z	#26

Note: The addresses which are not needed to specify can be omitted, the corresponding local variable of the omitted address is valued by <null>.

3.18 Metric/Inch Switch

3.18.1 Functional summary

CNC input, output separately has two kinds of unit: metric unit , mm; inch unit, inch. The corresponding state parameter related to metric, inch in TAC2000

CNC: No001＃0(INI)：input incremental unit selection

0：metric input (G21) 1：inch input (G20)

The parameter completely corresponds to G20/G21. i.e. the parameter changes along when G20/G21 is being executed; G20/G21 modal correspondingly changes when the parameter is changed.

No003＃0(OIM)：when the metric/inch input mode is switched, whether the tool compensation value and wear value is automatically switched:

0: do not automatically switch(only move one-bit decimal point) 1：automatically switch

No004＃0(SCW)：metric machine, inch machine selection (least output incrementselection) 0：metric machine output (0.001mm)

1：inch machine output (0.0001inch)

3.18.2 Function command G20/G21

Command format: G20; (inch input)
G21; (mm input)

G command must be in the beginning of the program, and is specified by the single block.

Warning: must not switch G20/G21 in program being executed; the system is turned on again after G20/G21 is executed.

3.18.3 Notes

⑴ **No.001 # 0(INI) input increment unit change**

① . After the input increment unit is changed (inch/metric input), the following unit system is changed: (i.e.: mm<>inch; mm/min<>inch/min):

- F specifies the feedrate (mm/min<>inch/min), thread lead (mm <>inch)
—position command (mm<>inch)
—tool compensation value (mm<>inch)
—MPG graduation unit (mm<>inch)
—movement distance in incremental feed (mm<>inch)
—some data parameters, including NO.45~NO.48, NO.56, NO.59, NO.60, NO.114~ NO.116, NO.120~ NO.131, NO.139, No.140, No.154; the unit is 0.001mm(IS-B) in the metric input system, is 0.0001inch(IS-B) in the inch input system. For example, the same parameter NO.45 setting value is 100m, it means to be 0.1mm in the metric input system (G21), and it means 0.1inch in the inch input system (G20).

② . The machine coordinates will automatically switch after the input increment unit change is switched:

⑵ **No.004 # 0(SCW) output command unit change**

SCW=0: the system minimal command increment uses the metric output (0.001mm) SCW=1: the system minimal command increment uses the inch output (0.0001inch)

Some data parameter meanings will be changed when the output control bit parameter SCW is changed:

① . Speed parameter:
Metric machine: mm/min Inch
machine: 0.1 inch/min
Example: when the speed is set to 3800, the metric machine is 3800 mm/min and the inch machine is 380 inch/min.
Speed parameters: No.22, No.23, No.27, No.28~No.31, No.32, No.33, No.41, No.107, No.113,No.134;

② . Position(length) parameter
metric machine: 0.001 mm
inch machine: 0.0001 inch
When the setting is 100, the metric machine is 0.1mm and the inch machine is 0.01 inch.
Position parameters: No.34, No.35, No.37~No.40, No.45~No.48, No.102~No.104,No.136~No.138 and all pitch error compensation parameter;

Note 1: When the minimal input increment unit and the minimal command unit are different, the maximal error is the half of minimal command unit. The error cannot be accumulated.

Note 2: The current system increment is IS-B in the above explanation.

CHAPTER 4 TOOL NOSE RADIUS COMPENSATION (G41, G42)

4.1 Application

4.1.1 Overview

Part program is compiled generally for one point of tool according to a workpiece contour. The point is generally regarded as the tool nose A point in an imaginary state (there is no imaginary tool nose point in fact and the tool nose radius can be omitted when using the imaginary tool nose point to program) or as the center point of tool nose arc (as Fig. 4-1). Its nose of turning tool is not the imaginary point but one arc owing to the processing and other requirement in the practical machining. There is an error between the actual cutting point and the desired cutting point, which will cause the over- or under-cutting affecting the part precision. So a tool nose radius compensation is needed in machining to improve the part precision.

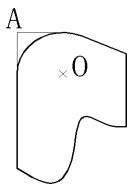


Fig. 4-1 Tool

B tool compensation is defined that a workpiece contour path is offset one tool nose radius, which cause there is excessive cutting at an intersection of two programs because of executing motion path of next after completing the previous block.

To avoid the above-mentioned ones, the system uses C tool compensation method (namely, tool nose radius compensation). The system will read the next block instead of executing it immediately after reading a block in C tool compensation method, and count corresponding motion path according to intersection of blocks. Contour can be compensated precisely because reading two blocks are pretreated as Fig.4-2.

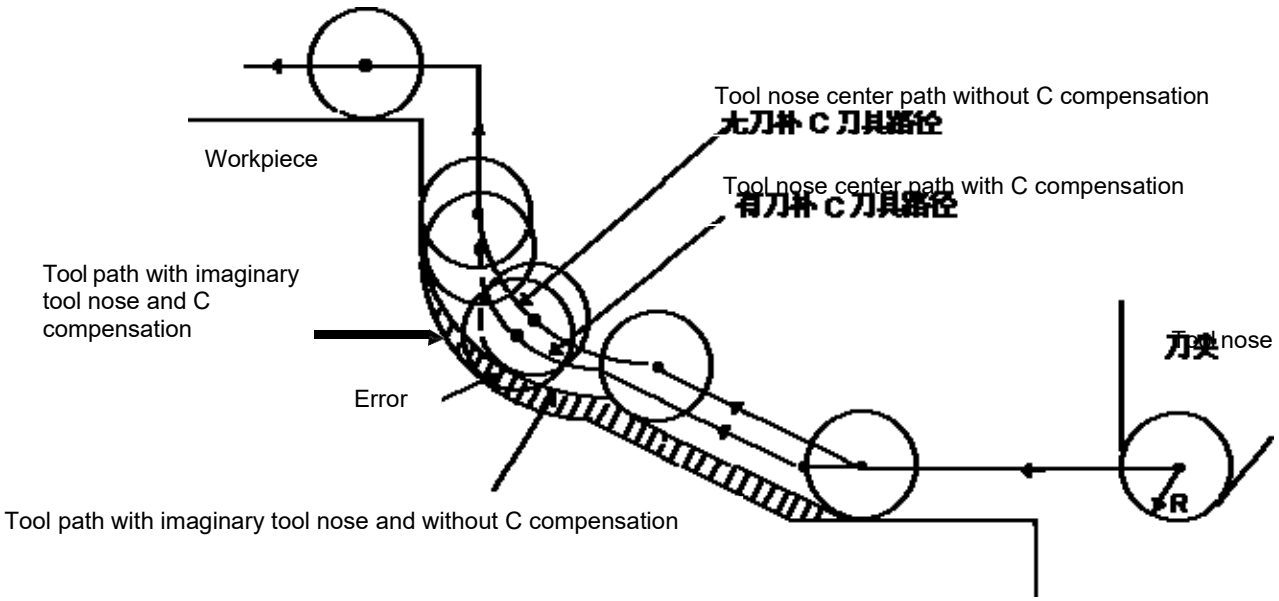


Fig. 4-2 Tool nose center path

4.1.2 Imaginary tool nose direction

Suppose that it is generally difficult to set the tool nose radius center on the initial position as Fig. 4-3; suppose that it is easily set the tool nose on it as Fig. 4-4; The tool nose radius can be omitted in programming. Fig. 4-5 and Fig.4-6 correspond separately to the tool paths of tool nose center programming and imaginary tool nose programming when tool nose radius is executed or not.

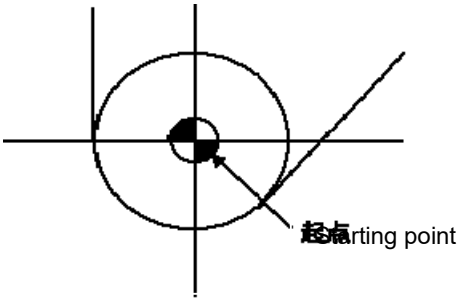


Fig. 4-3 用刀尖中心编程时 Programming with tool nose

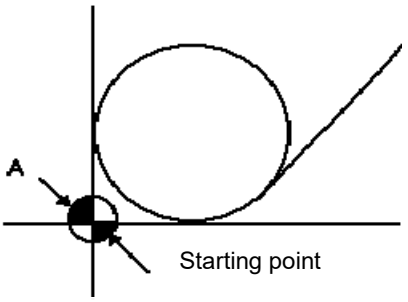


Fig. 4-4 用假想刀尖编程时 Programming with imaginary tool nose

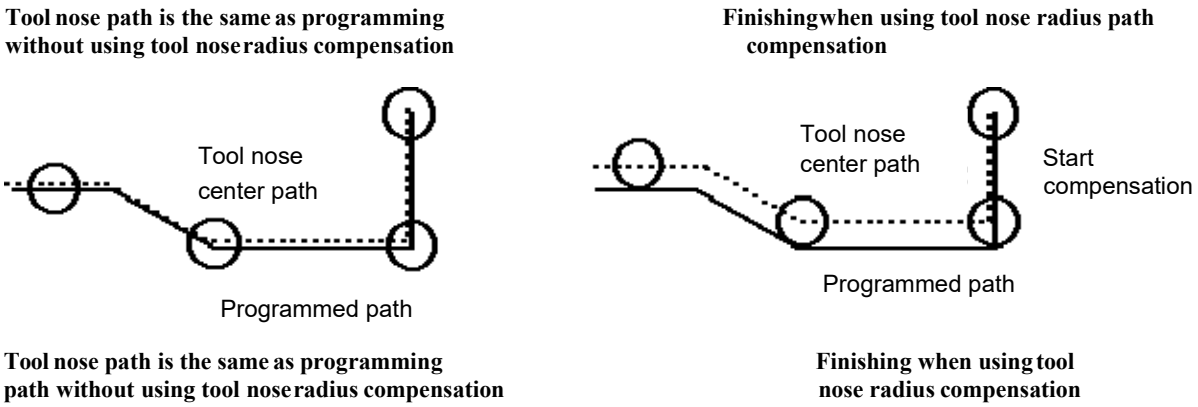


Fig. 4-5 Tool path in tool nose center programming

Fig. 4-6 Tool path in imaginary tool nose programming

The tool is supposed to one point in programming but the actual cutting blade is not one ideal point owing to machining technology. Because the cutting blade is not one point but one circular, machining error is caused which can be deleted by tool nose circular radius compensation. In actual machining, suppose that there are different position relationship between tool nose point and tool nose circular center point, and so it must create correct its direction of imaginary tool nose.

From tool nose center to imaginary tool nose, set imaginary tool nose numbers according to tool direction in cutting. Suppose there are 10 kinds of tool nose setting and 9 directions for position relationship. The tool nose directions are different in different coordinate system (rear tool post

coordinate system and front tool post coordinate system) even if they are the same tool nose direction numbers as the following figures. In figures, it represents relationships between tool nose and starting point, and end point of arrowhead is the imaginary tool nose; T1 ~ T8 in rear tool post coordinate system is as Fig. 4-7; T1 ~ T8 in front tool post coordinate system is as Fig. 4-8. The tool nose center and starting point for T0 and T9 are shown in Fig. 4-9.

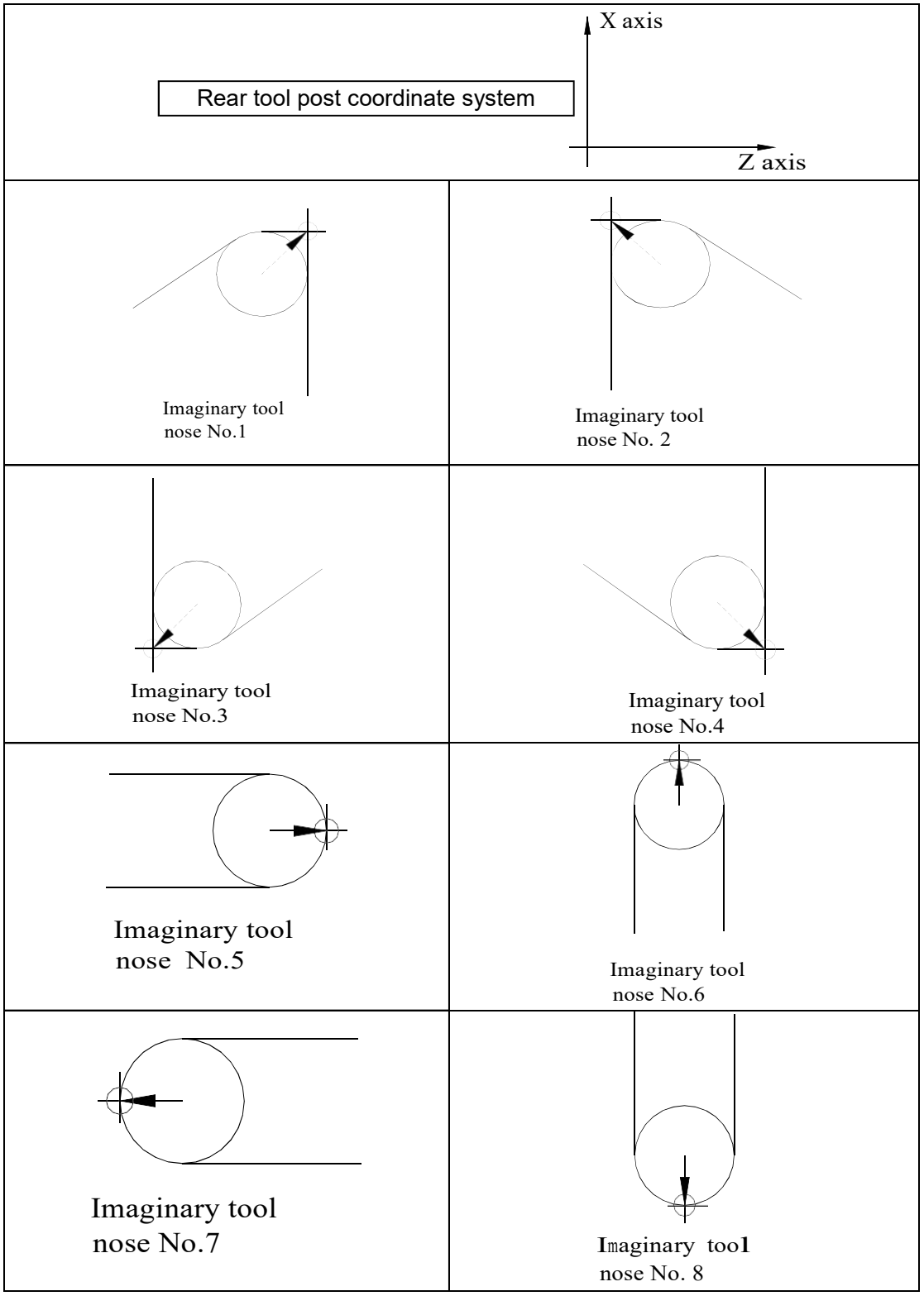


Fig. 4-7 Imaginary tool nose number in rear tool post coordinate system

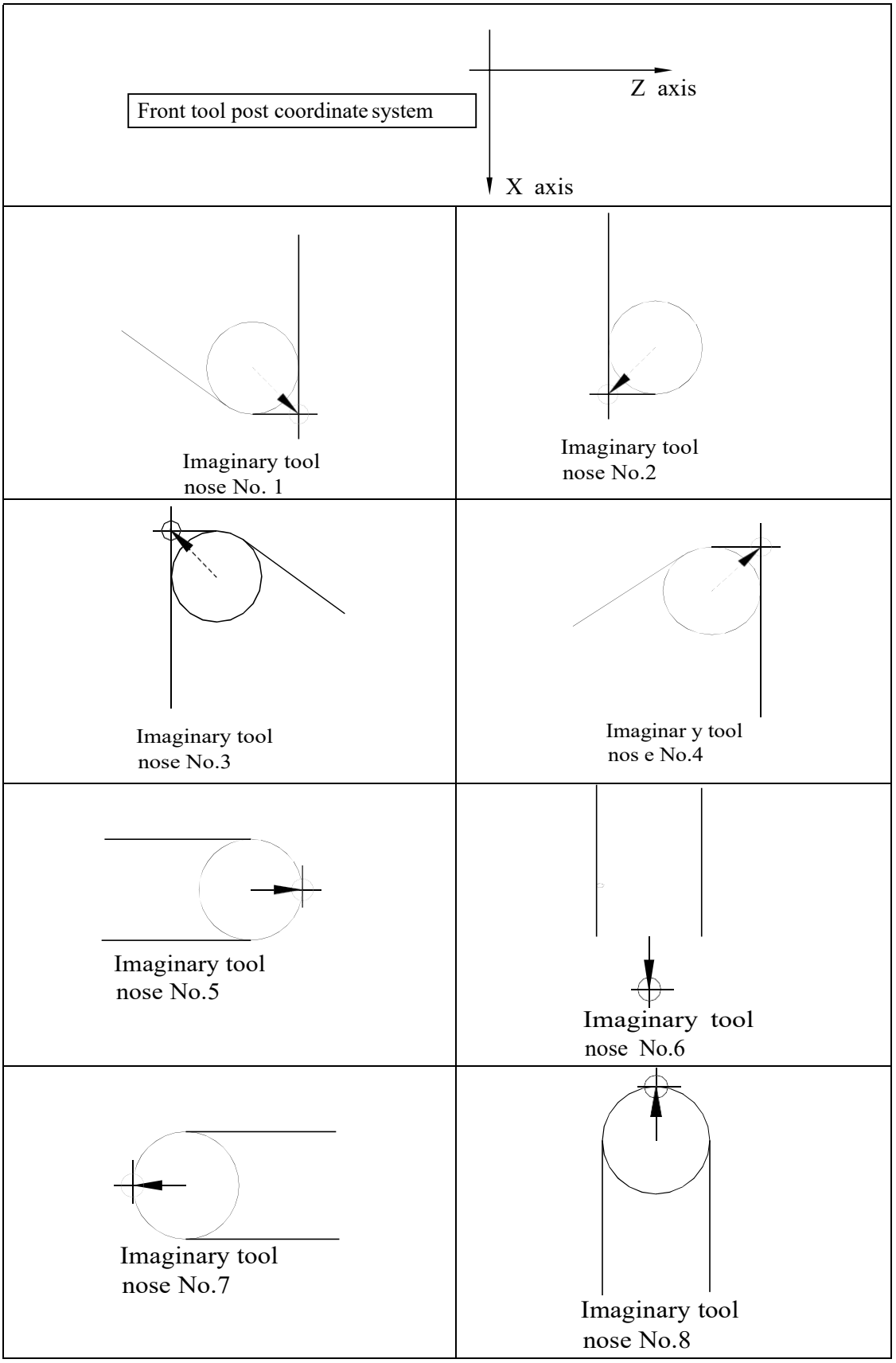


Fig. 4-8 Imaginary tool nose number in front tool post coordinate system

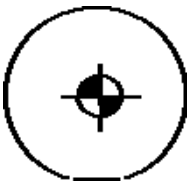


Fig. 4-9 Tool nose center on starting point

4.1.3 Compensation value setting

Preset imaginary tool nose number and tool nose radius value for each tool before executing tool nose radius compensation. Set the tool nose radius compensation value in ***OFFSET*** window (as Fig. 4-1), R is tool nose radius compensation value and T is imaginary tool nose number.

Table 4-1 CNC tool nose radius compensation value display window

number	X	Z	R	T
000	0.000	0.000	0.000	0
001	0.020	0.030	0.020	2
002	1.020	20.123	0.180	3
…	…	…	…	…
032	0.050	0.038	0.300	6

Note: X tool offset value can be specified in diameter or radius, set by No.004 Bit4 ORC, offset value is in radius when ORC=1 and is in diameter when ORC=0.

In toolsetting, the tool nose is also imaginary tool nose point of Tn (n=0~9) when taking Tn(n=0~9) as imaginary tool nose. For the same tool, offset value from standard point to tool nose radius center (imaginary tool nose is T3) is different with that of ones from standard point to imaginary tool nose(imaginary tool nose is T3) when T0 and T3 tool nose points are selected to toolsetting in rear tool post coordinate system, taking tool post center as standard point. It is easier to measure distances from the standard point to the tool nose radius center than from the standard point to the imaginary tool nose, and so set the tool offset value by measuring distance from the standard point to the imaginary tool nose(tool nose direction of T3).

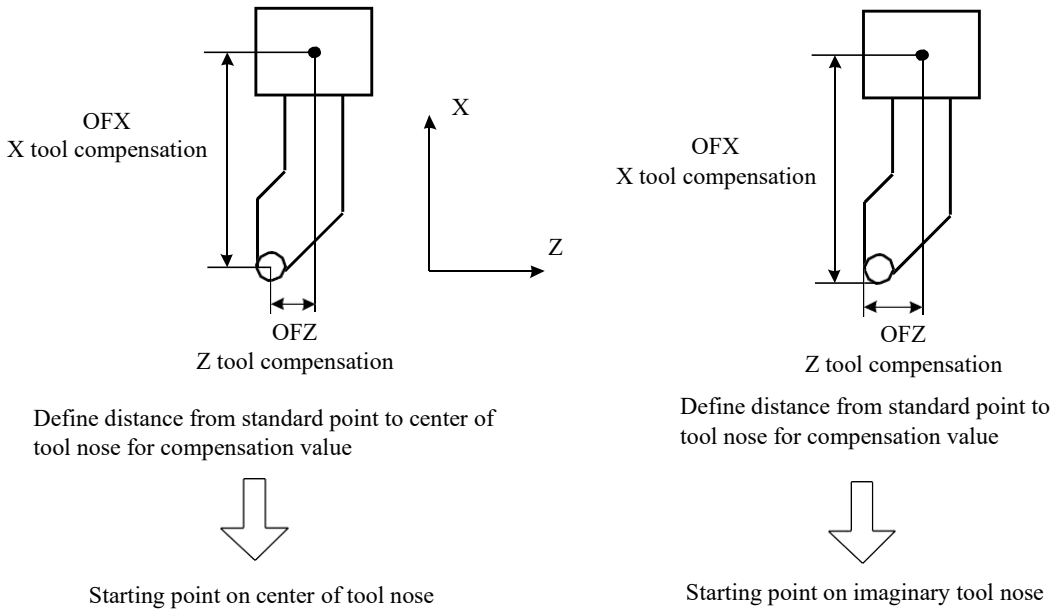


Fig. 4-10 Tool offset value of tool post center as benchmark

4.1.4 Command format

{ G40 / G41 / G42 } { G00 / G01 } X___Z____T____;

Commands	Function specifications	Remark
G40	Cancel the tool nose radius compensation	See Fig.4-11 and 4-12
G41	Tool nose radius left compensation is specified by G41 in rear tool post coordinate system and tool nose radius right compensation is specified by G41 in front tool post coordinate system	
G42	Tool nose radius right compensation is specified by G42 in rear tool post coordinate system and tool nose radius left compensation is specified by G42 in front tool post coordinate system	

4.1.5 Compensation direction

Specify its direction according to relative position between tool nose and workpiece when executing tool nose radius compensation is shown in Fig. 4-11 and Fig.4-12.

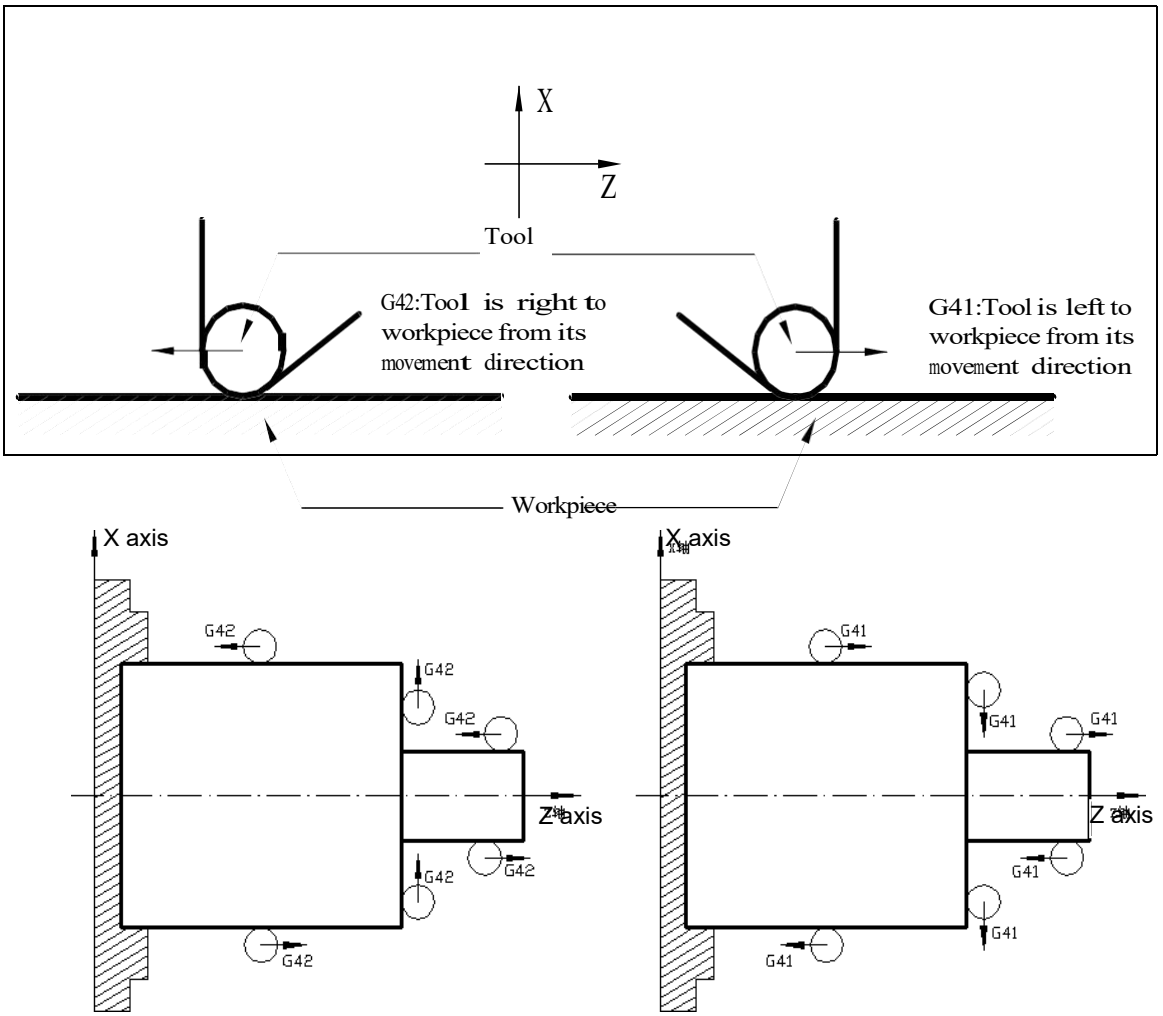


Fig. 4-11 Compensation direction of rear coordinate system

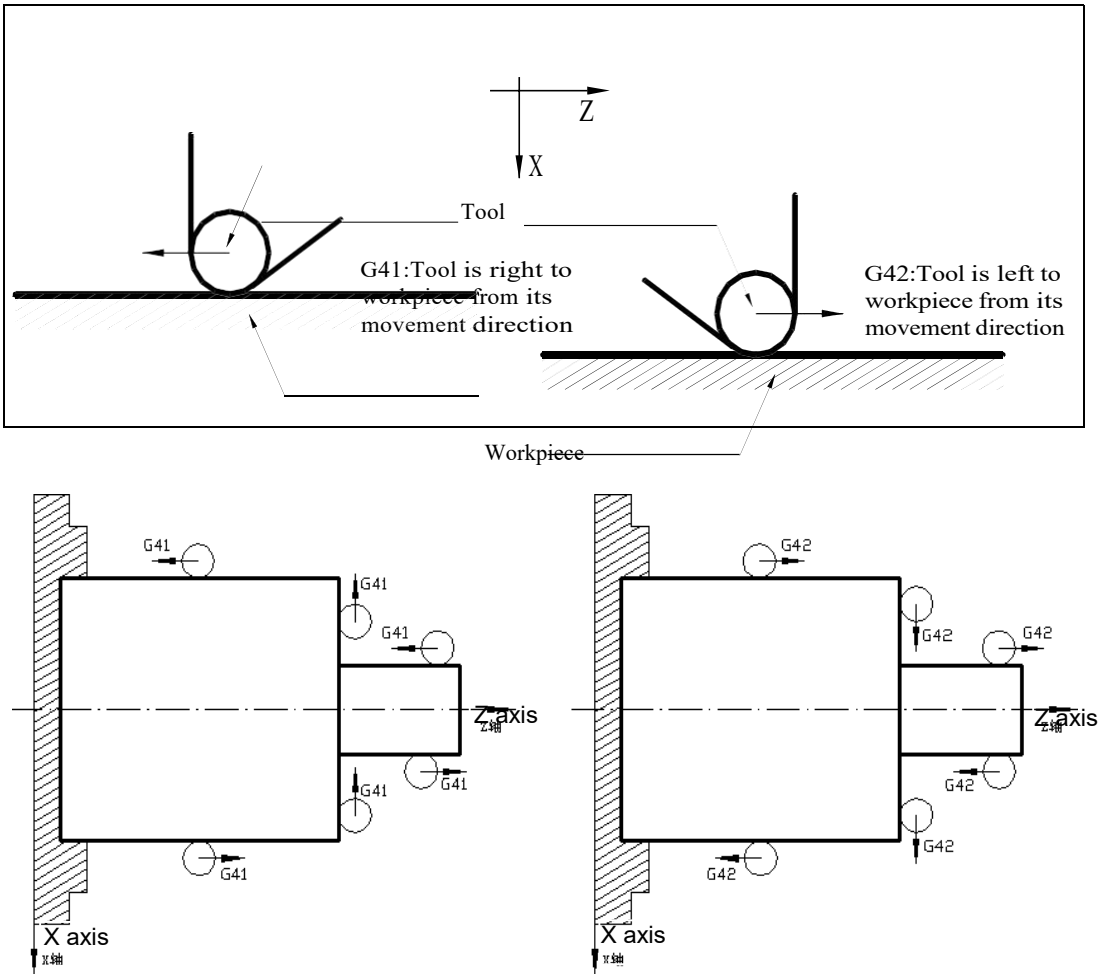


Fig. 4-12 Compensation direction of front coordinate system

4.1.6 Notes

z The system is in tool nose radius compensation mode at initial state, and starts to create tool nose radius compensation offset mode when executing G41 or G42. When the system starts to execute compensation, it pre-read two blocks, and the next block is saved to storage for tool nose radius compensation when executing one of them. The system reads two blocks in ***"Single"*** mode and stops after executing end point of the first block.

z In tool nose radius compensation mode, the tool nose center moves to end point of previous block and is vertical to its path when the system executes two block or more than blocks without motion Command.

z The system cannot create and cancel tool nose radius compensation.

z Tool nose radius R is without negative value, otherwise there is a mistake running path.

z Tool nose radius compensation is created and cancelled in G00 or G01 instead of G02 or G03, otherwise, the system alarms.

z The system cancels the tool nose radius compensation mode when pressing  key.

z G40 must be specified to cancel offset mode before the program is ended, otherwise the tool path offsets one tool nose radius.

z The system executes the tool nose radius compensation in main program and subprogram but

must cancel it before calling subprogram and then create it again in the subprogram.

- The system does not execute the tool nose radius compensation in G71, G72, G73, G74, G75, G76 and cancel it temporarily.
- The system executes the tool nose radius compensation in G90, G94, it offsets one tool nose radius for G41 or G42.

4.1.7 Application

Machine a workpiece in the front tool post coordinate system as Fig. 4-13. Tool number: T0101, tool nose radius R=2, imaginary tool nose number T=3.

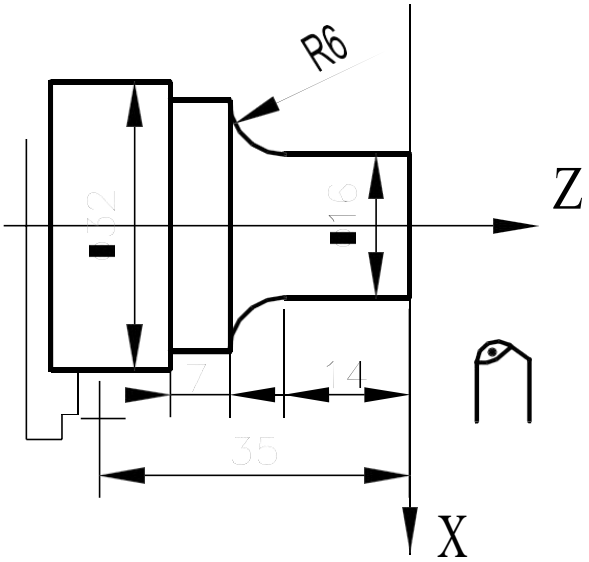


Fig. 4-13

For toolsetting in Offset Cancel mode, after toolsetting, Z axis offsets one tool nose radius and its direction is relative to that of imaginary tool nose and toolsetting point, otherwise the system excessively cuts tool nose radius when it starts to cut.

Set the tool nose radius R and imaginary tool nose direction in "***TOOL OFFSET&WEAR***" window as following:

Table 4-3

No.	X	Z	R	T
001			2.000	3
002	…	…	…	…
…	…	…	…	…
007	…	…	…	…
008	…	…	…	…

Program:

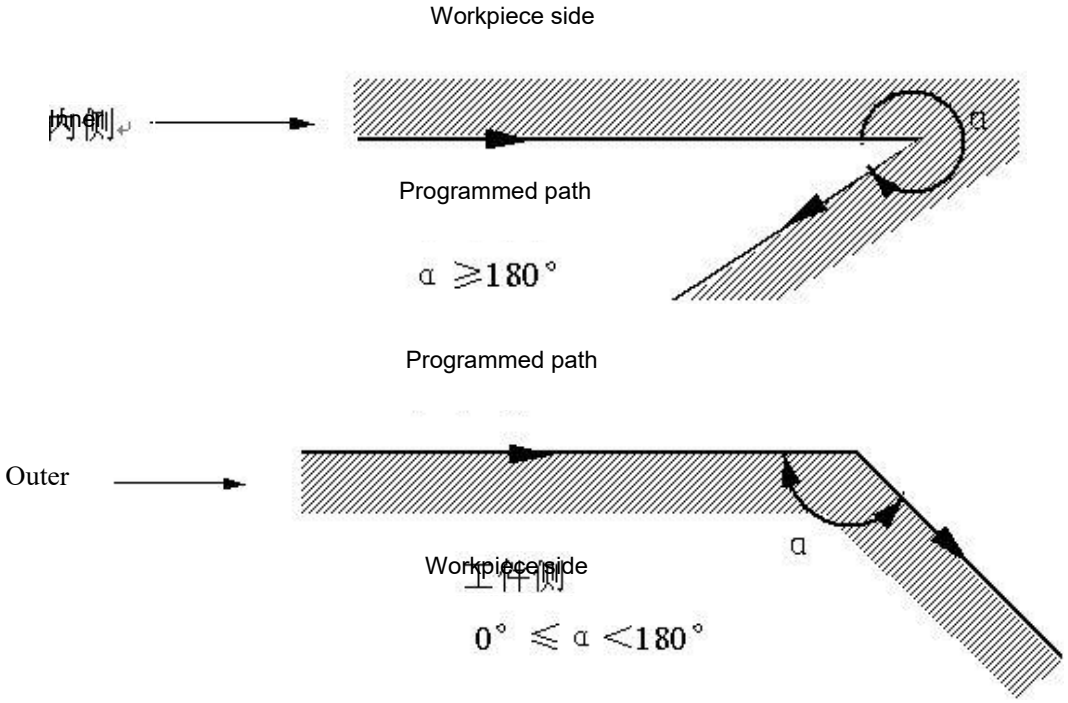
```
G00 X100 Z50 M3  T0101  S600;      (Position, start spindle, tool change and execute
                                    tool compensation)
G42 G00 X0 Z3;                     (Set tool nose radius compensation)
G01 Z0 F300;                       (Start cutting)
X16;
```

```
Z-14 F200;
G02 X28 W-6 R6;
G01 W-7;
X32;
Z-35;
G40 G00 X90 Z40;                    (Cancel tool nose radius compensation)
G00 X100 Z50 T0100;
M30;
```

4.2 Tool nose radius compensation offset path

4.2.1 Inner and outer side

Inside is defined that an angle at intersection of two motion blocks is more than or equal to 180°;
Outside is 0~180°.



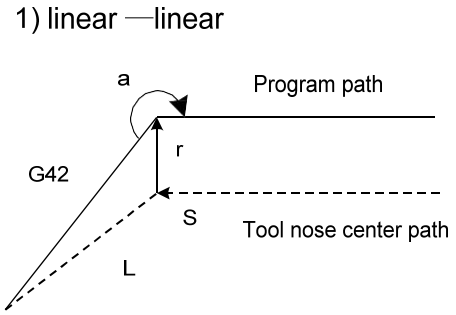
4.2.2 Tool traversing when starting tool

3 steps to execute tool nose radius compensation: tool compensation creation, tool compensation execution and tool compensation canceling.

Tool traverse is called tool compensation creation (starting tool) from offset canceling to G41 or G42 execution.

Note: Meanings of S, L, C in the following figures are as follows:
S——Stop point of single block; L——linear; C——circular.

(a) Tool traversing inside along corner(α≥180°)



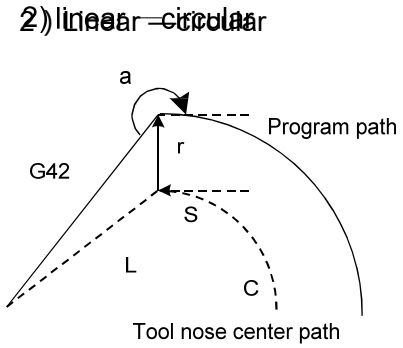
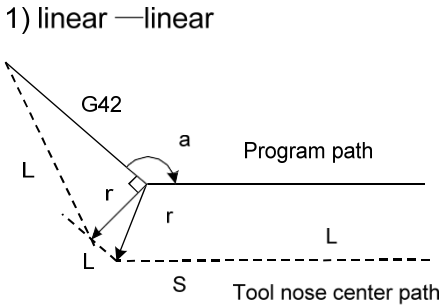


Fig.4-14a Linear —linear(starting tool inside) Fig. 4-14b Linear —circular (starting tool inside)

(b) Tool traversing inside along corner(180° > α≥90°)



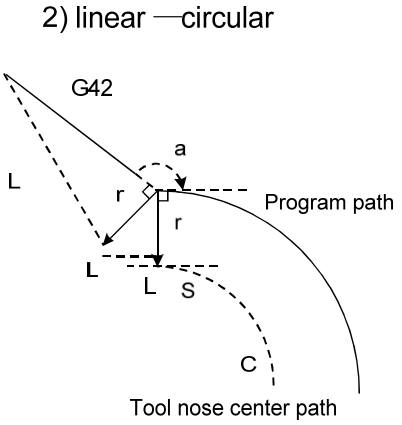
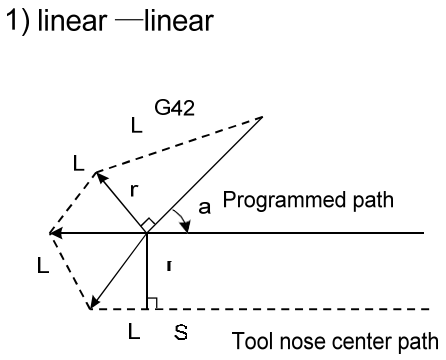


Fig.4-15a Linear —linear(starting tool outside) Fig.4-15b Linear—circular(starting tool outside)

(c) Tool traversing inside along corner (α < 90°)



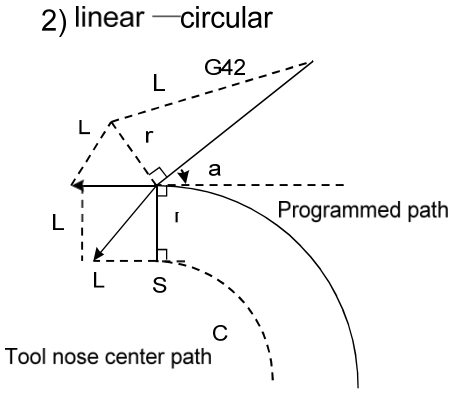


Fig.4-16a Linear —linear(starting tool outside) Fig. 4-16b Linear—circular (starting tool outside)

(d) Tool traversing inside along corner(α≤1°) , linear →linear

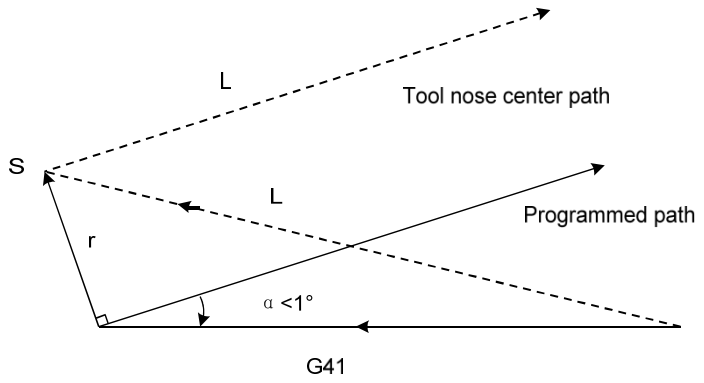


Fig. 4-17 Linear—linear (α < 1°, starting tool outside)

4.2.3 Tool traversing in Offset mode

Offset mode is called to ones after creating tool nose radius compensation and before canceling it.

z **Offset path without changing compensation direction in compensation mode**

(a) Tool traversing inside along corner(α≥180°)

1) linear —linear

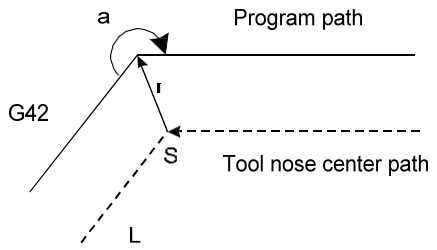


Fig. 4-18a Linear —linear (moving inside)

2) linear —circular

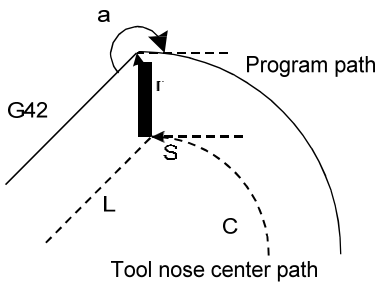


Fig. 4-18b Linear—circular (moving inside)

3) circular—linear

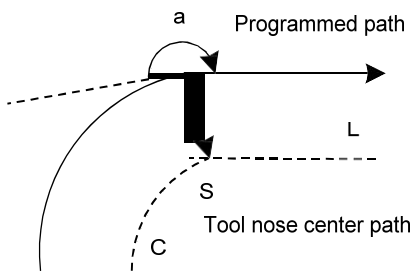


Fig. 4-18c Circular—linear(moving inside)

4) circular —circular

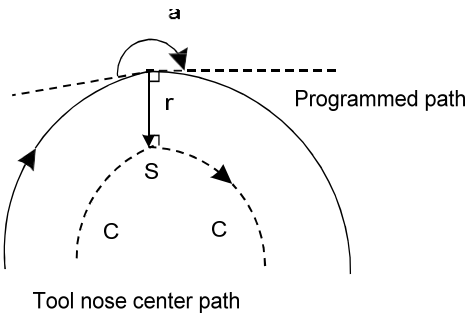


Fig. 4-18d Circular—circular(moving inside)

(b) Tool traversing outside along corner(180° > α≥90°)

1) linear —linear

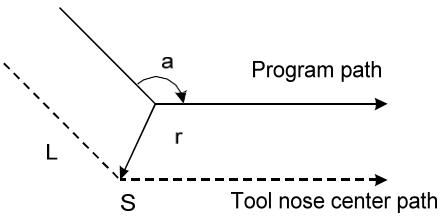


Fig. 4-19a Linear —linear
(180° > α≥90°, obtuse angle, moving outside)

2) linear —circular

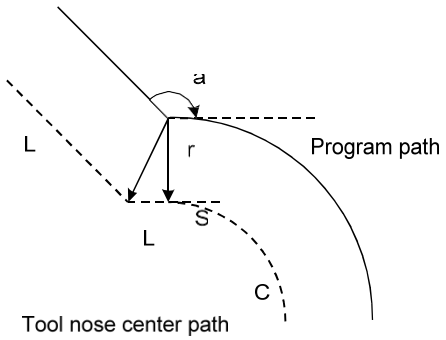


Fig. 4-19b Linear—circular
(180° > α≥90°, obtuse angle, moving outside)

3) circular —linear

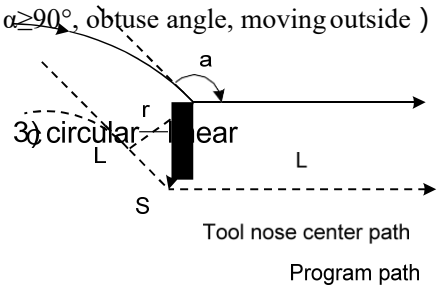


Fig. 4-19c cirular —linear
(180° > α≥90°, obtuse angle, moving outside)

4)circular —circular

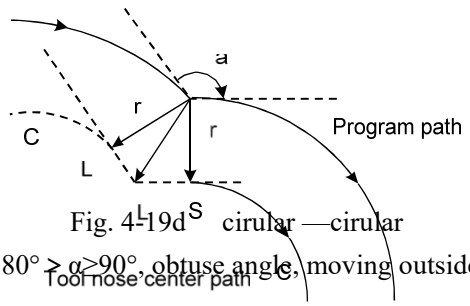


Fig. 4-19d cirular —cirular
(180° > α≥90°, obtuse angle, moving outside)

(c) Tool traversing outside along corner(α < 90°)

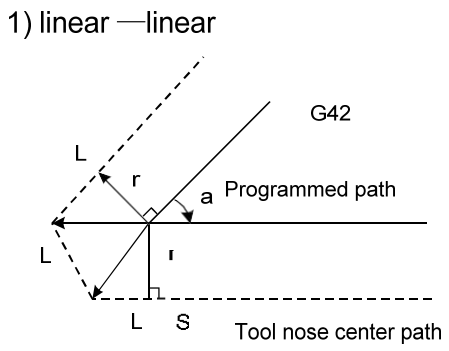


Fig. 4-20a Linear—Linea (moving outside)

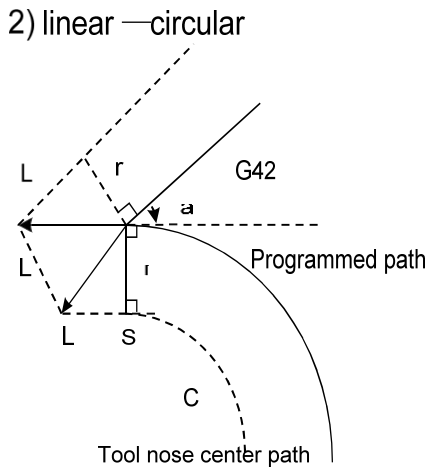


Fig. 4-20b Linear—circular (moving outside)

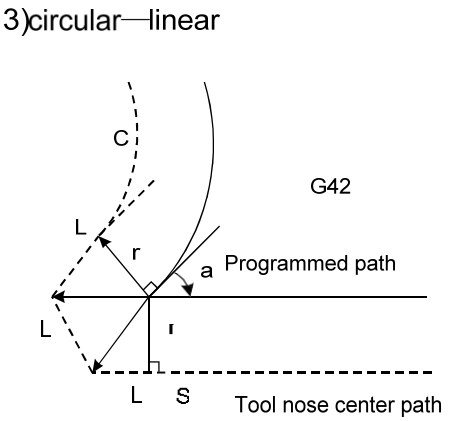


Fig.4-20c Circular—linear(moving outside)

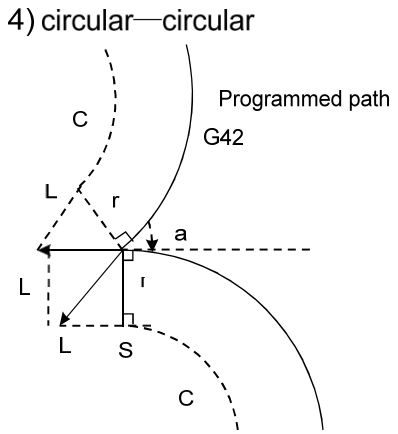


Fig.4-20d Circular—circular(moving outside)

5）Machining inside (α＜1°) and zoom in the compensation vector

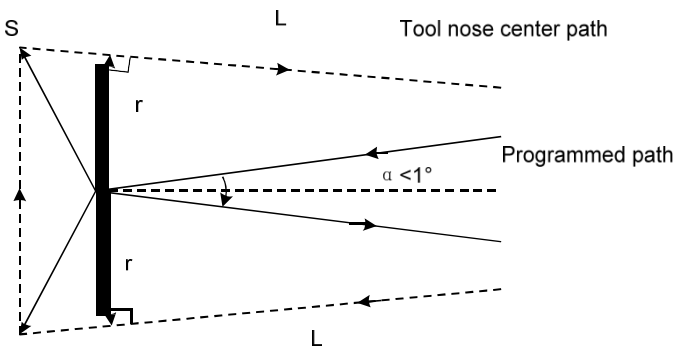


Fig. 4-20e Linear —linear (α < 1°, moving inside)

(d) Special cutting

1) Without intersection

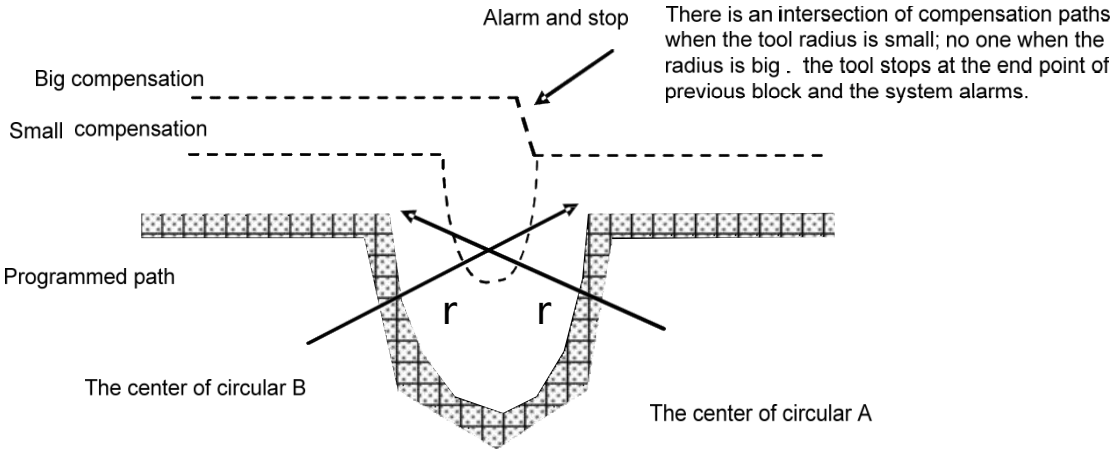


Fig. 4-21 Paths without intersection after offset 2) Center

point and starting point of circular being the same one

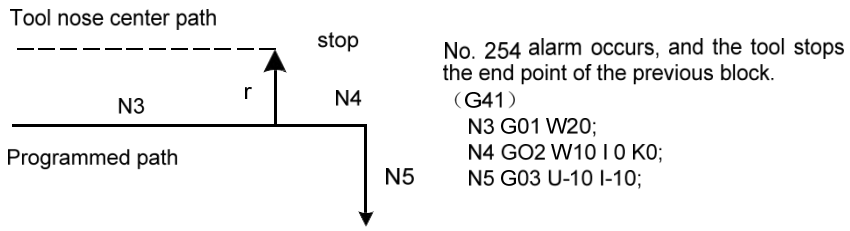


Fig. 4-22 Center point and starting point of circular being the same one

z **Offset path of compensation direction in compensation mode**

The compensation direction of tool nose radius is specified by G41 and G42 and the sign symbol is as follows:

Table 4-3

Comp. sign	Sign symbol of compensation value	
G Command	+	-
G41	Left compensation	Right compensation
G42	Right compensation	Left compensation

The compensation direction can be changed in compensation mode in special cutting, it cannot be changed at starting block and its following one. There is no inside and outside cutting when the system changes the compensation direction. The following compensation value is supposed to be positive.

1)linear—linear

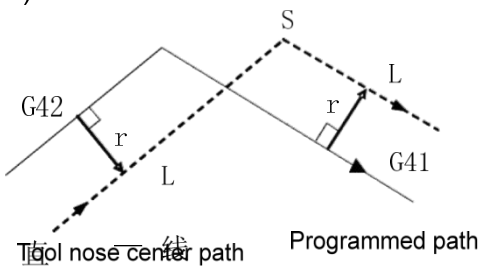


Fig. 4-23 Linear—linear

(changing compensation direction)

2) linear—circular

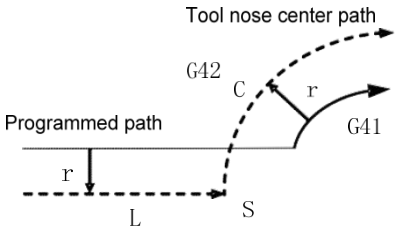


Fig. 4-24 Linear—circular

(changing compensation direction)

3)circular—linear

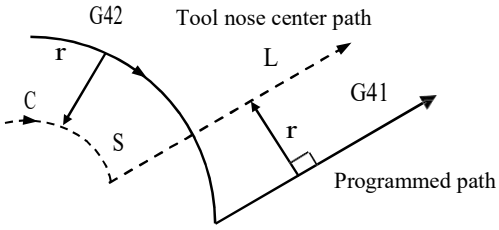


Fig. 4-25 circular—linear

(changing compensation direction)

4) circular—circular

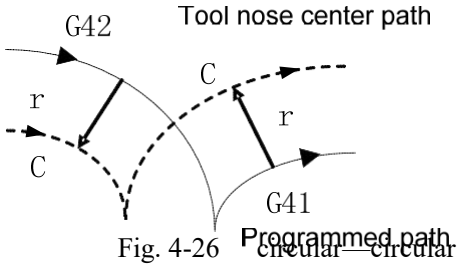


Fig. 4-26 circular—circular

(changing compensation direction)

5) No intersection when compensation is executed normally

When the system executes G41 and G42 to change the offset direction between block A and B, a vector perpendicular to block B is created from its starting point.

i) Linear---- Linear

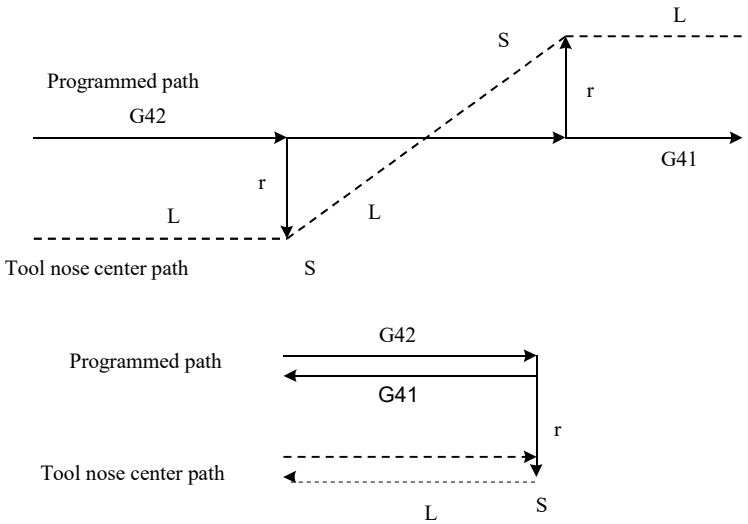


Fig. 4-27a Linear—linear, no intersection (changing compensation direction)

ii) Linear ---circular

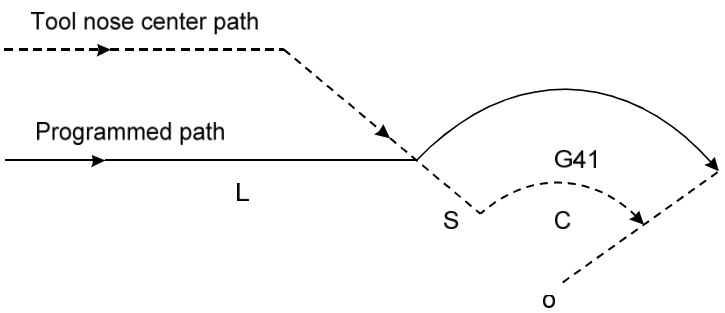


Fig. 4-27b Linear—circular without intersection（changing compensation direction）

iii) Circular----- circular

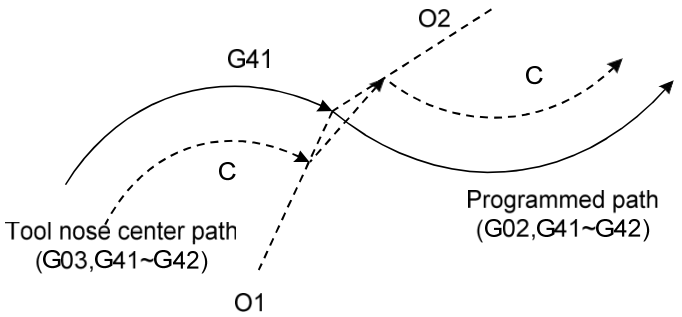


Fig. 4-27c Circular—circular without intersection (changing compensation direction)

4.2.4 Tool traversing in Offset canceling mode

In compensation mode, when the system executes a block with one of the followings, it enters compensation canceling mode, which is defined to compensation canceling of block.

1. Execute G40 in a program;
2. Execute M30.

The system cannot execute G02 and G03 when canceling C tool compensation (tool nose radius compensation), otherwise the system alarms and stops run.

In compensation canceling mode, the system executes the block and ones in the register for tool nose radius compensation. At the moment, the run stops after one block is executed when single block is ON. The system executes the next one but does not read its following one when pressing **CYCLE START** button again.

(a) Tool traversing inside along corner(α≥180°)

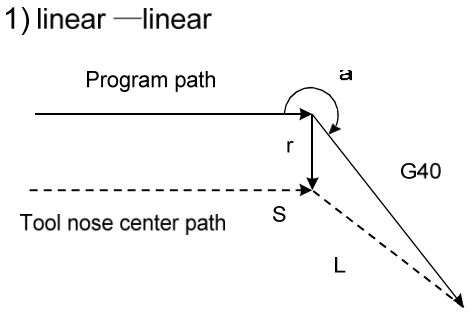


Fig. 4-28a linear-linear
(moving inner and canceling offset)

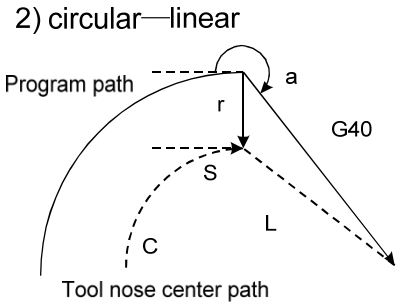


Fig. 4-28b Circular-linear
(moving inner and canceling offset)

(b) Tool traversing outside along corner(180° > α≥90°)

1) linear —linear

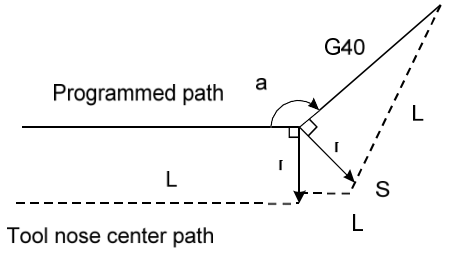


Fig. 4-29a linear—linear (α≥90° moving outside and canceling offset)

2) circular—linear

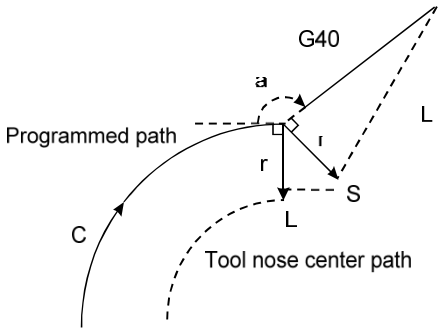


Fig. 4-29b Circular—linear (α≥90°moving outside and canceling offset)

(c) Tool traversing outside alongcorner(α < 90°)

1) linear —linear

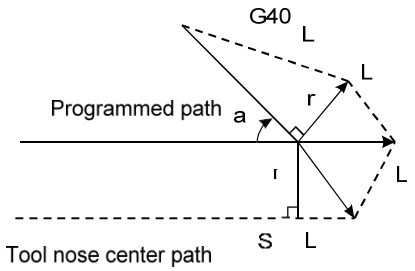


Fig. 4-30a Linear—linear (α < 90°cutting outside and canceling offset)

2) circular —linear

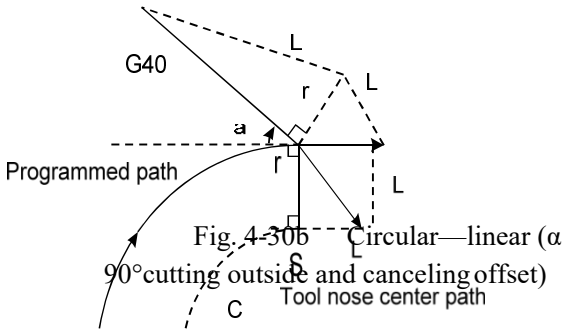


Fig. 4-30b Circular—linear (α < 90°cutting outside and canceling offset)

(d) Tool traversing outside along corner(α < 1°) ; linear→ linear

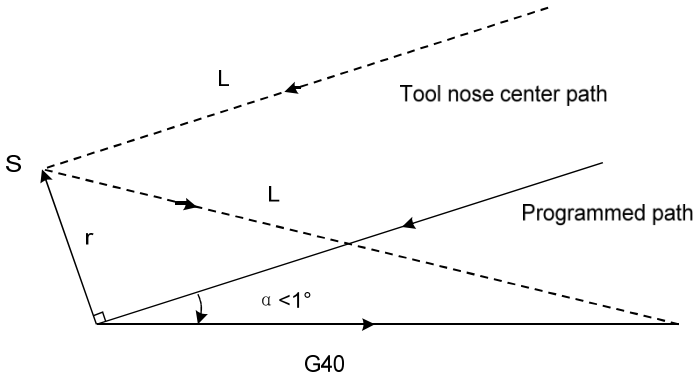


Fig. 4-31 Linear—linear (α < 1°cutting outside and canceling offset)

4.2.5 Tool interference check

"Interference" is defined that the tool cuts workpiece excessively and it can find out excessive cutting in advance, the interference check is executed even if the excessive cutting is not created, but the system cannot find out all tool interferences.

(1) Fundamental conditions

1) The tool path direction is different that of program path (angle is 90°~270°).

2) There is a big difference (α > 180°) for two angles between starting point and end point of tool nose center path, and between starting point and end point of program path.

Example: linear machining

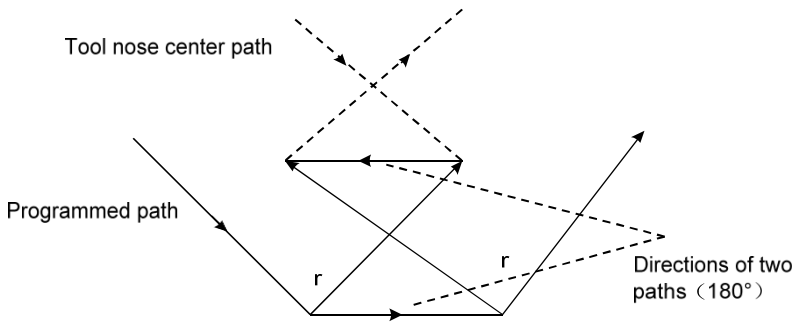


Fig. 4-32a Machining interference (1)

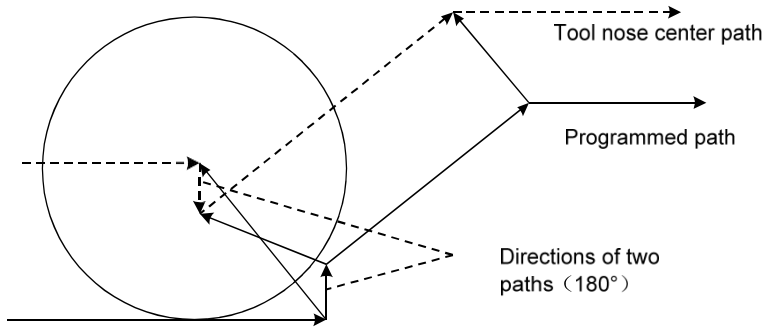


Fig. 4-32b Machining interference (2)

(2) Executing it without actual interference

1) Concave groove less than compensation value

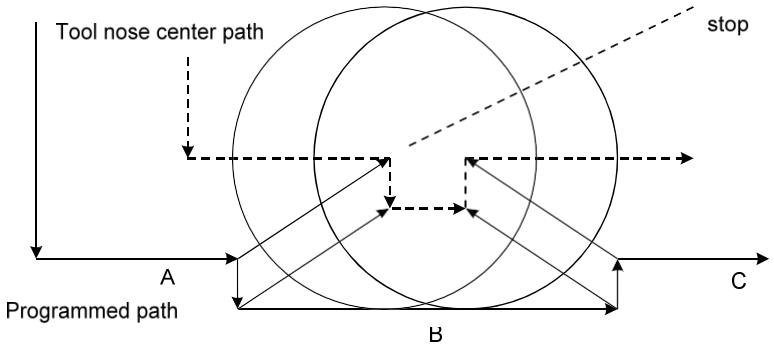


Fig. 4-33 Executing interference (1)

Directions of block B and tool nose radius compensation path are opposite without interference, the tools stops and the system alarms.

2) Concave channel less than compensation value

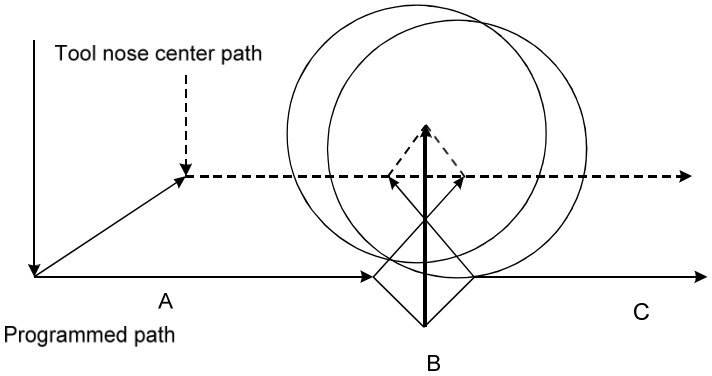


Fig. 4-34 Executing interference (2)

Directions of block B and tool nose radius compensation path are opposite without interference, the tools stops and the system alarms.

4.2.6 Commands for canceling compensation vector temporarily

In compensation mode, the compensation vector is cancelled temporarily in G50, G71~G76 and is automatically resumed after executing the commands. At the moment, the compensation is cancelled temporarily and the tool directly moves from intersection to a point for canceling compensation vector. The tool directly moves again to the intersection after the compensation mode is resumed.

z Setting coordinate system in G50

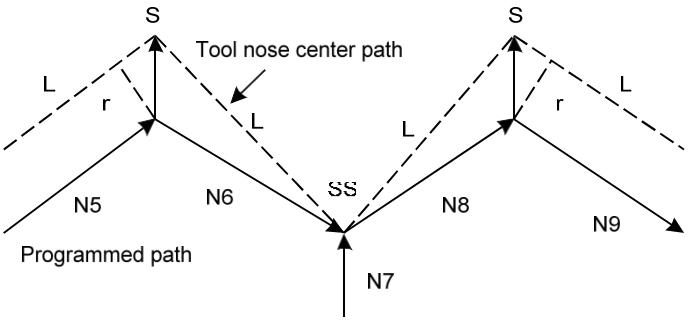


Fig. 4-35 Temporary compensation vector in G50

Note: SS indicates a point at which the tool stops twice in Single mode.

z Reference point automatic return G28

In compensation mode, the compensation is cancelled in a middle point and is automatically resumed after executing the reference point return in G28.

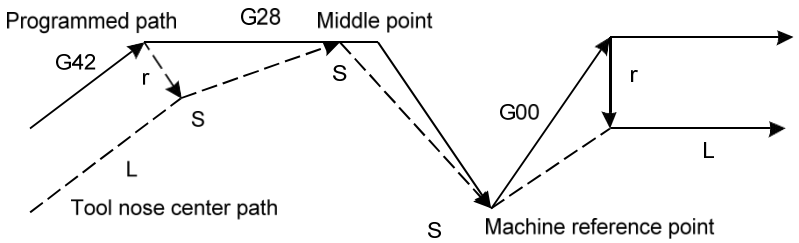


Fig. 4-36 Cancel compensation vector temporarily in G28

z **G71 ~ G75 compound cycle; G76, G92 thread cutting**

When executing G71 ~ G76 , G96 thread cutting, the system does not execute the tool nose radius compensation and cancel it temporarily, and there is G00, G01,in the following blocks, and the system automatically recovers the compensation mode.

z **G32, G33,G34 thread cutting**

They cannot run in the tool nose radius compensation mode, otherwise, No.131 alarm occurs "……CANNOT USED TO C COMPENSATION".

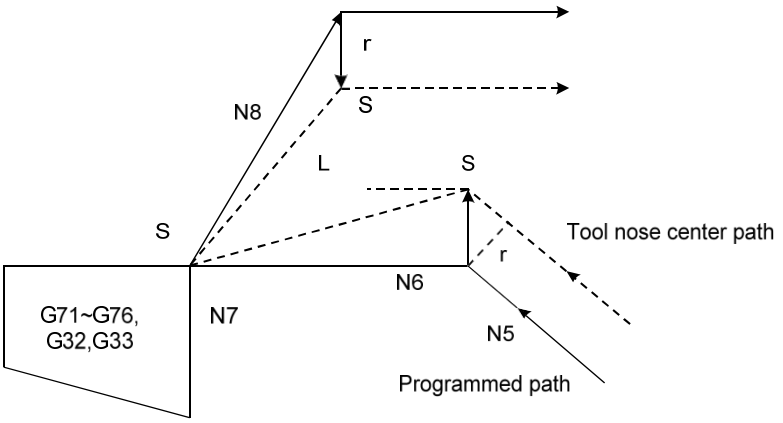


Fig. 4-37 Cancel compensation vector temporarily in G71 ~ G76

Z **G90, G94**

Compensation method of tool nose radius compensation in G90 or G94:

A. Cancel the previous tool nose radius compensation;
B. Create the previous C compensation before cutting, and the path ① in the following figure creates the previous radius compensation mode;
C. The paths ②,③ in the following figure are the radius compensation cutting;
D. The path ④ in the following figure can cancel the radius compensation, and the tool returns to the cycle starting point; there is G00,G01 in the following block, and the CNC automatically recovers the compensation mode.

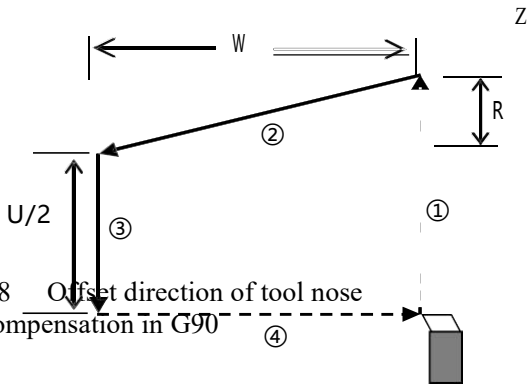


Fig. 4-38 Offset direction of tool nose radius compensation in G90

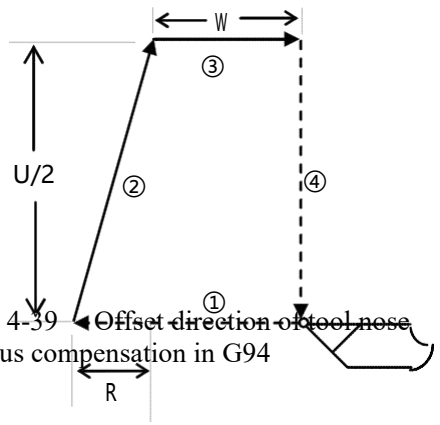


Fig. 4-39 Offset direction of tool nose radius compensation in G94

4.2.7 Particulars

z Inside chamfer machining less than tool nose radius

At the moment, the tool inside offset causes an excessive cutting. The tool stops and the system alarms (P/S41) when starting the previous block or chamfer moving. But the tool stops the end point of previous block when **Single** is ON.

z Machining concave less than tool nose diameter

There is an excessive cutting when the tool nose center path is opposite to program path caused by tool nose radius compensation. At the moment, the tool stops and the system alarms when starting the previous block or chamfer moving.

z Machining sidestep less than tool nose radius

The tool center path can be opposite to program path when the sidestep is less than tool nose radius and is an circular in program. At the moment, the system automatically ignores the first vector and directly moves end point of second vector linearly. The program stops at the end point in single block and otherwise the cycle machining is continuously executed. If the sidestep is a linear, compensation is executed correctly and the system does not alarm (but the not-cutting is still reserved).

z Subprograms in G Commands

The system must be in canceling compensation mode before calling subprograms. After calling subprograms, the offset is executed and the system must be in canceling compensation mode before returning to main programs, otherwise the system alarms.

z Changing compensation value

(a) Change compensation value in canceling tool change mode. New compensation value is valid after tool change when the compensation value is changed in compensation mode.

(b) Compensation value sign symbol and tool nose center path

G41 and G42 are exchanged each other if the compensation value is negative (-). The tool moves along inside when its center moves along outside of workpiece, and vice versa.

Generally, the compensation value is positive (+) in programming. The compensation value is negative (-) when the tool path is as the above-mentioned (a), and vice versa.

Besides, direction of tool nose offset changes when offset value sign symbol is changed, but we suppose the direction of tool nose is not changed. Generally, the offset value sign symbol is not changed.

z End point of programming circular out of circular

The tool stops and the system alarms and displays “End point of circular is not on circular” when the end point of circular is not on circular in programs.

Volume II
Operation

Chapter II Operator Panel

1.1 Panel layout

The TAC2000 machine center has an integrated operator panel. The layout of it is shown as following:

1.1.1 Status indication

X Y Z 4th C	Axis zero return completion indictor
ALM READY RUN	Three color indictor

1.1.2 Edit Keypad

Button	Name	Function
RESET	Reset key	CNC reset, feed and output stop, etc.
X Y& Z: U W	Address keys	Input the address
O N# G* P Q I A J B K C R V M[S] T H= D L F E		Double-address keys, switch between two addresses through pressing the buttons repeatedly
./< −+>	Symbol keys	Three-address key, switch between three addresses through pressing the buttons repeatedly
7 8 9 4 5 6 1 2 3 0	Digit keys	Input with the digits
DATA INPUT	Input key	Confirm the data input, like the parameters and the compensation amount, etc.
DATA OUTPUT	Output key	Start the communication output
CHANGE	Change key	Change between the information and the display
INSERT ALTER DELETE	Edit keys	Inserting, rewriting and deleting the program and the field, etc during editing,

Button	Name	Function
EOB	EOB key	For block end sign input
⇧ ⇨ ⇩ ⇦	Cursor moving keys	For cursor moving control
	Page keys	Page switching in a same interface

1.1.3 Display Menu

Menu key	Remark
POSITION	To enter position interface. There are RELATIVE POS, ABSOLUTE POS, INTEGRATED POS, POS&PRG pages in this interface.
PROGRAM	To enter program interface. There are PRG CONTENT, PRG STATE, PRG LIST, FILE DIRECTORY,4 pages in this interface.
OFFSET	To enter TOOL OFFSET interface. There are TOOL OFFSET, MARRO variables. OFFSET interface displays offset values; MARRO for CNC macro variables.
ALARM	To enter alarm interface. There are CNC, PLC ALARM and ALARM Log pages in this interface.
SETTING	To enter Setting interface. There are SWITCH, PASSWORD SETTING, DATE &TIME, SETTING （G54～G59）, GRAGH SET and TRACK pages in this interface.
PARAMETER	To enter BIT PARAMETER, DATA PARAMETER, PITCH COMP interfaces (switching between each interface by pressing repeatedly).
DIAGNOSE	To enter DIAGNOSIS interface.There are CNC DIAGNOSIS, PLC STATE, PLC VALUE, VERSION MESSAGE interfaces (switching between each interfaces by pressing the key repeatedly). CNC DIAGNOSIS, PLC STATE, PLC VALUE interfaces display CNC internal signal state, PLC addresses, data state message; the VERSION MESSAGE interface displays CNC software, hardware and PLC version No.
PLC	To enter Ladder interface.There are PLC information,PLC ladder,PLC parameter PLC diagnosis interfaces(switching between each interfaces by pressing the key repeatedly).

1.1.4 Machine Panel

The keys function in TAC2000 machine panel is defined by PLC program (ladder), see their function significance in the machine builder's manual.

The functions of the machine panel keys defined by standard PLC program are as follows:

Key	Name	Function explanation	Function mode
FEED HOLD	Feed Hold key	Dwell commanded by program, MDI	Auto mode, DNC, MDI mode
CYCLE START	Cycle Start key	Cycle start commanded by program, MDI	Auto mode, DNC, MDI mode
F. OVERRIDE F. OVERRIDE	Feedrate Override keys	For adjustment of the feedrate	Auto mode, DNC, MDI mode, Edit mode, Machine zero mode, MPG mode, Single Step mode, MANUAL mode
X1 F0 X10 25% X100 50% X1000 100%	Rapid override keys	For adjustment of rapid traverse	Auto mode, DNC, MDI mode, Machine zero mode, MANUAL mode
S. OVERRIDE S. OVERRIDE	Spindle override keys	For spindle speed adjustment (spindle analog control valid)	Auto mode, DNC, MDI mode,edit mode, Machine zero mode, MPG mode, Step mode, MANUAL mode
JOG	JOG key	For spindle Jog ON/OFF	Machine zero mode, MPG mode, Single Step mode, MANUAL mode
LUBRICATING	Lubricating key	For machine lubrication ON/OFF	Machine zero mode, MPG mode, Single Step mode, MANUAL mode
COOLING	Cooling key	For coolant ON/OFF	Auto mode, MDI mode,Edit mode, Machine zero mode, MPG mode Step mode, MANUAL mode
S. CW S. STOP S. CCW	Spindle control keys	Spindle CW Spindle stop Spindle CCW	Machine zero mode, MPGmode, Single Step mode,MANUAL mode
RAPID	Rapid traverse key	For rapid traverse /feedrate switching	Auto mode, DNC,MDI mode, Machine zero mode, MANUAL mode

Key	Name	Function explanation	Function mode
	X axis feed key	For positive/negative moving of X, Y, Z axis in Manual, Step mode	Machine zero mode, Step mode, MANUAL mode
	Y axis feed key		
	Z axis feed key		
	4th axis feed key		
X1 F0 X10 25% X100 50% X1000 100%	MPG/Step increment and Rapid override selection key	Move amount per handwheel scale 0.001/0.01/0.1 mm Move amount per step 0.001/0.01/0.1 mm Rapid override F0, F25%,F50%,F100%	Auto mode, MDI mode, Machine zero mode, MPG mode, Step mode,MANUAL mode
OPTIONAL STOP	Optional stop	Execute M01stop when optional stop is valid	Auto mode, DNC, MDI mode
SINGLE	Single Block key	For switching of block/blocks execution, Single block lamp lights up if Single mode is valid	Auto mode, DNC, MDI mode
SKIP	Block Skip key	For skipping of block headed with“/”sign, if its switch is set for ON, the Block Skip indicator lights up	Auto mode, DNC, MDI mode
MACHINE LOCK	Machine Lock key	If the machine is locked, its lamp lights up, and X, Z axis output is invalid.	Auto mode, DNC, MDI mode, Edit mode, Machine zero mode, MPG mode, Step mode, MANUAL mode
MST M.S.T LOCK	M.S.T. Lock key	If the miscellaneous function is locked, its lamp lights up and M, S, T function output is invalid.	Auto mode, DNC, MDI mode
	Edit mode key	To enter Edit mode	Auto mode, DNC, MDI mode, Machine zero mode, MPG mode, Step mode, MANUAL

Key	Name	Function explanation	Operation mode
EDIT	Edit mode key	To enter Edit mode	Auto, MDI, Machine zero return, Manual, Step, MPG, Program zero return
AUTO	Auto mode key	To enter Auto mode	MDI, Edit, Machine zero return, Manual, Step, MPG, Program zero return
MDI	MDI mode key	To enter MDI mode	Auto, Edit, Machine zero return, Manual, Step, MPG, Program zero return
MACHINE ZERO	Machine zero return mode key	To enter Machine zero return mode	Auto, MDI, Edit, Manual, Step, MPG, Program zero return
MPG	Step/MPG mode key	To enter Step or MPG mode (one mode by parameter)	Auto, MDI, Edit, Machine zero return, Manual, Program zero return
MANUAL	Manual mode key	To enter Manual mode	Auto, MDI, Edit, Machine zero return, Step, MPG, Program zero return
DNC DNC	DNC mode key	To enter DNC mode	Auto, MDI, Edit, Machine zero return, Step, MPG, Manual

1.2 Overview of the Operation Modes

There are 7 modes in TAC2000, which are Edit, Auto, MDI, Machine zero, Step/MPG, Manual, DNC.

- **Edit mode**

  In Auto mode, the operation of part program setup, deletion and alteration can be performed.

- **Auto mode**

  In Auto mode, the program is automatically run.

- **MDI mode**

  In MDI mode, the operation of parameter input, command blocks input and execution can be performed.

- **Machine zero return mode**

  In machine zero return mode, the feeding axis returning to the machine zero can be operated respectively.

- **MPG / Step mode**

  In MPG / Step mode, the moving is performed by an increment selected by CNC system.

- **Manual mode**

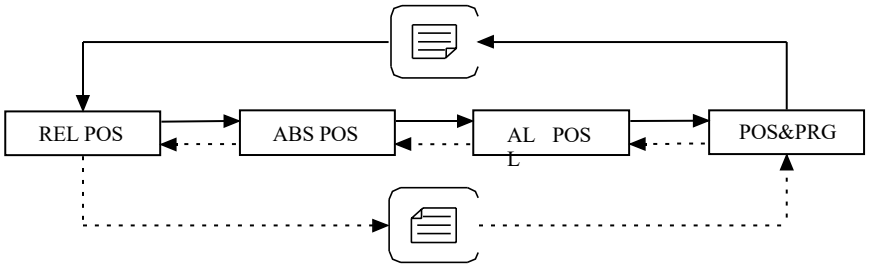
  In Manual mode, the manual feeding, manual rapid rate, the feedrate override adjustment, the rapid override adjustment, the spindle start/stop, the coolant switch, the lubricant switch, the spindle inch and the manual tool change, etc. can be executed.

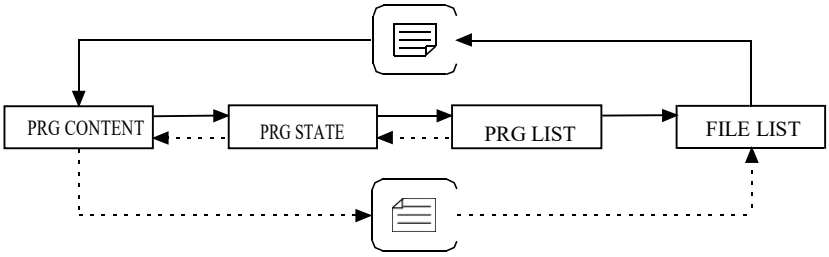
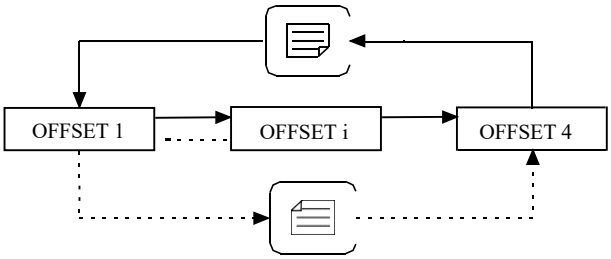
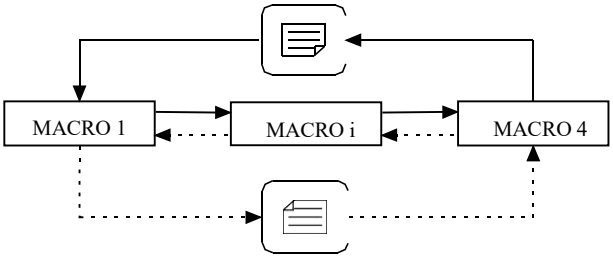
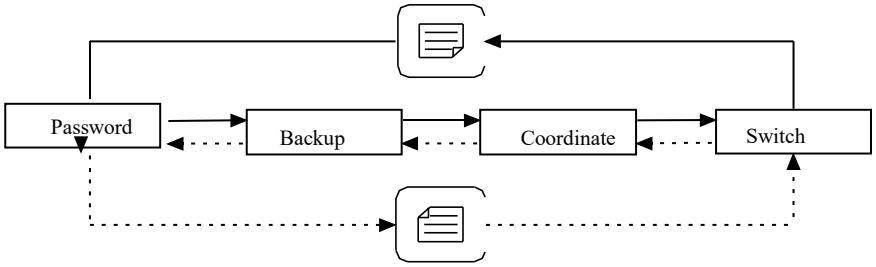
- **DNC mode**

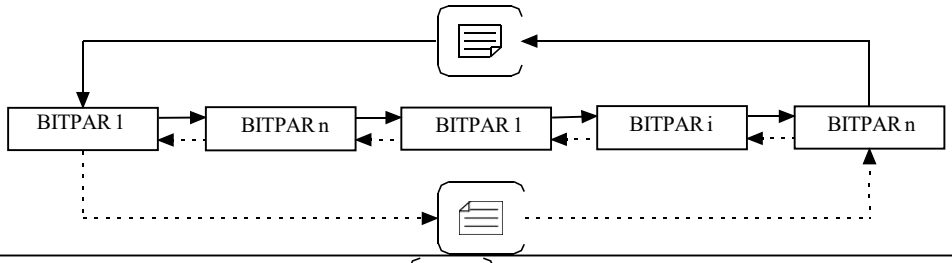
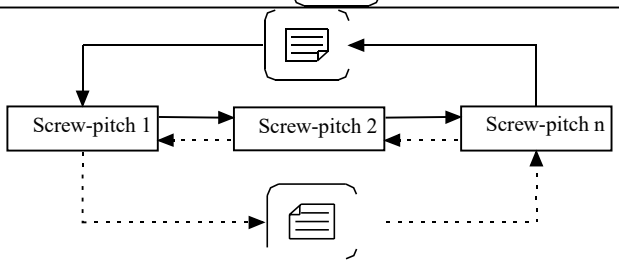
  In DNC mode, the program is run by DNC mode.

1.3 Display Interface

There are 9 interfaces such as Position, Program etc., and there are multiple pages in each interface. Each interface (page) is separated with the operation mode. See the following figures for the display menu, display interface and page layers:

	Position interface	

PROGRAM	Program interface	
OFFSET	Tool offset interface	
		
ALARM	ALARM interface	Alarm information
	ALARM log interface	Alarm log
SETTING		

PARAMETER	Bit parameter	
	Data parameter	
	Screw-pitch parameter	
DIAGNOSE	diagnosis CNC	
	PLC Signal	
	Machine tool Softpanel	Machine Soft panel
	Help	
	Edition Information	Edition Information
PLC	PLC Information	
	PLC Ladder	
	PLC parameter	
	PLC diagnosis	

1.3.1 Position interface

Press POSITION to enter Position interface, which has four interfaces such as ABSOLUTE POS, RELATIVE POS, INTEGRATED POS and POS&PRG, and they can be viewed by or keys.

1) ABSOLUTE POS **display** interface

The X, Y, Z coordinates displayed are the absolute position of the tool in current workpiece coordinate system, as CNC power on, these coordinates are held on and the workpiece coordinate system is specified by G92、G54-G59.

Note It displays "PRG. F" in Auto, MDI mode; "JOG. F" in Machine zero,Manual mode;"HNDL INC"in MPG mode; "STEP INC"in Step mode.

ACT. F: actual speed after feedrate override in a machining.

FED OVRI: an override by feedrate override switch.

G CODE: modal value of 01 group G code and 03 group G code

PART CNT: part number plusing 1 when M30(or M99 in the main program) is executed

CUT TIME: Time counting starts if Auto run starts, time units are hour, minute and second

RAPID OVRI: It displays current rapid rapid override

SPINDLE OVRI: It displays spindle override when BIT4 of NO.001 was set 0

S0000: Feedback spindle speed of spindle encoder, and spindle encoder is a must

Current tool: Tool No. specified by T command in program

Tool offset:H00 tool length compensation of current processing program;

D00 radius compensation of current processing program

The parts counting and the cut time are memorized at power-down and the clearing ways for them are as following:

PART CNT clearing: press CHANGE key then press N# key.

CUT TIME clearing: press CHANGE key then press M[key.

2) RELATIVE POS display page

The X, Y, Z axis coordinates displayed are the current position relative to the relative reference point, and they are held on at CNC power on. They can be cleared at any time. If X, Y, Z axis relative coordinates are cleared, the current position will be the relative reference point.

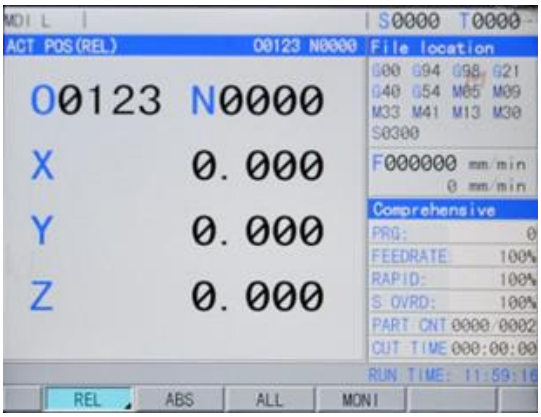
-

The clearing steps of X, Y, Z axis relative coordinates:

In RELATIVE POS page, press and hold [X] key till the “X”in the page blinks, press [CHANGE] key to clear X coordinate;

In RELATIVE POS page, press and hold [Y&] key till the“Y”in the page blinks, press [CHANGE] key to clear Y coordinate;

In RELATIVE POS page, press and hold [Z:] key till the “Z”in the page blinks, press [CHANGE] key to clear Z coordinate;



3) INTEGRATED POS display page

In INTEGRATED POS page, the RELATIVE, ABSOLUTE, MACHINE coordinate, DIST TO GO (only in Auto and MDI mode) are displayed together.

The displayed value of MACHINE coordinate is the current position in the machine coordinate system which is set up according to the machine zero.

DIST TO GO is the difference of the target position by block or MDI command to the current position.

The display page is as following:

-

1.3.2 Program page set

Press PROGRAM to enter Program interface, which has four pages such as PRG CONTENT, PRG STATE, PRG LIST and FILE CONTENT in non-Edit modes, and they can be viewed by pressing PROGRAM key repeatedly or pressing the correspondingsoftware key.

1) Program content page

In the program content page, the content of the current program is displayed. In Edit mode, press and keys to check the program content forward or backward.

2) Program state

In the program content page，Press PROGRAM Key and enter the program statepage

3) Program contentspage

In the program status page ， press PROGRAM key and enter the program contents page 。
At this interface, all of the processing procedures are listed.

The contents of the program Files page

(a) Existing program number：

Shows the number of programs that have been stored in the CNC (including subroutine)

(b) Remaining number of program：

Shows the number of programs that can be stored in CNC

(c) Storage capacity：

The amount of memory that the CNC has been stored in（KB）

（d）Residual memory：

TheremainingcapacityoftheCNCstoragepartprogram(KB) (e)Program contents：

The program number of the part program is displayed in the name of the part program.

(f) Programsize：

The size of the storage space of the CNC program

4) File contentspage

In the program contents page，press CHANGE enter the file contents page。

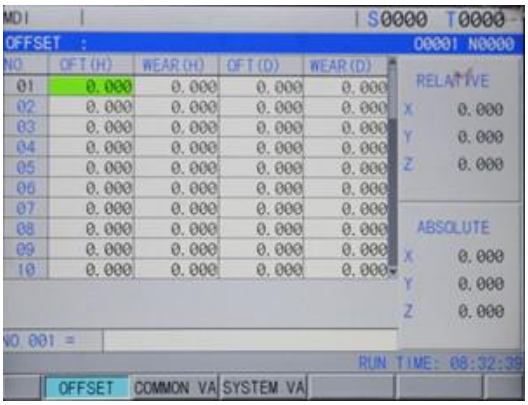
Page display is as follows：

1.3.3 Tool offset, macro variable and tool life management interface

OFFSET is a compound key, press OFFSET key once in other page, it enters the TOOL OFFSET page, press OFFSET key again, it enters the MACRO interface.

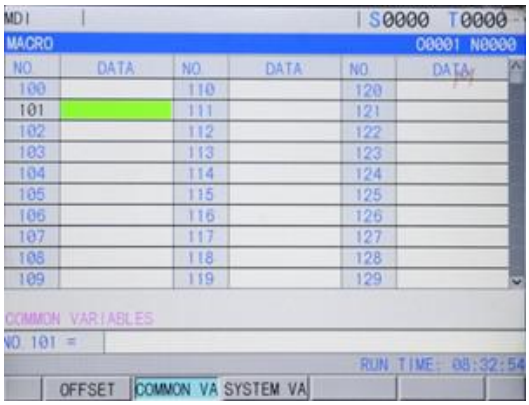
1) OFFSET interface

There are 4 tool offset pages in this interface, and 32 offset No. （No.001～No.032）available for user, which can be shown as following by pressing or keys.



2. COMMON VARIABLE interface

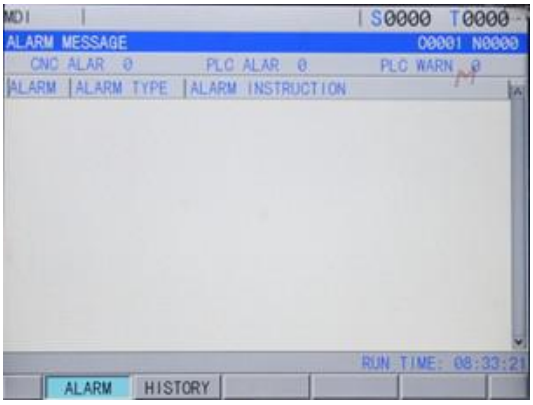
There are 20 windows in this interface, which can be shown by pressing or key. In Macro window there are 600 (No.100～No.199 and No.500～No.999) macro variables which can be specified by macro command or set by keypad.



1.3.4 ALARM interface

1) Alarm:

Press ALARM key to enter alarm interface, which can be viewed by or key, the window is as follows:

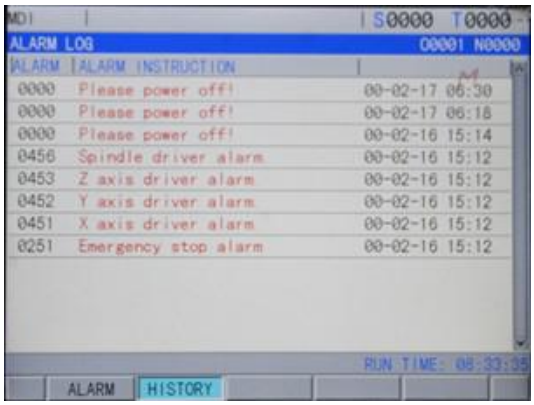


Note: Alarm clearing: It may clear alarms by pressing RESET (it should press RESET and CANCEL keys together to clear No.100 alarm). If the current alarm window is as follows:

2) Alarm log:

Press ALARM key again to enter WARN LOG interface, 200 messages can be viewed by pressing or key.

Sequence of warn log: The latest log message is shown on the forefront of the 1st window, and the others queue in sequence. If the messages are over 200, the last one will be cleared.



Note: Manual clear alarm log: press CANCEL + J B in 2-level password to clear all log message.

1.3.5 Setting interface

SETTING is a compound key, press SETTING key in other window, it enters the SETTING interface, press it again, it enters the GRAPHIC interface. Press SETTING key repeatedly, it switches between SETTING and GRAPHIC interfaces.

1. SETTING interface

There are 5 windows in this interface , which can be viewed by or key. SWITCH SETTING: It is used for the parameter, program, auto sequence No. on-off state.

PARM SWT: when it is turned for ON, the parameters are allowed to be altered; it is turned for OFF, the parameters are forbidden to be altered.

PROG SWT: when it is turned for ON, the programs are allowed to be edited; it is turned for OFF, the programs are forbidden to be edited.

AUTO SEG: when it is turned for ON, the block No. is created automatically; it is turned for OFF, the block No. is not created automatically, but manually if needed.

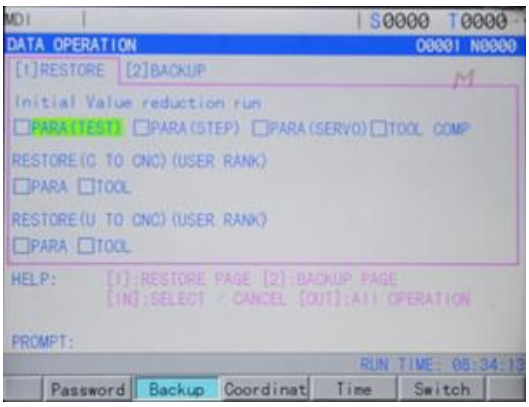
DATA OPERATION: In this window, the CNC data (such as bit parameter, data parameter, screw-pitch parameter, tool offset) can be backup and restored.

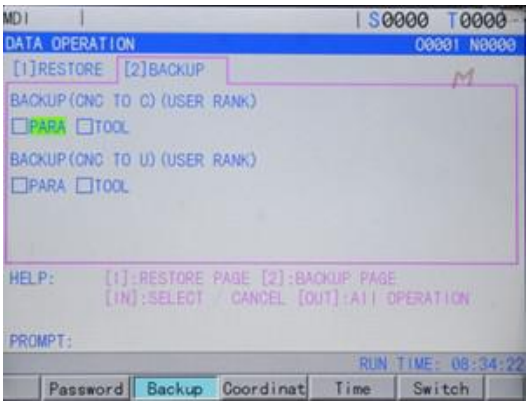
Restore original value:Restore parameters,tool compensation,screw-pitch to system default value. C disk data restore to CNC:Restore data files which are backup to system disk to system

U disk data restore to CNC:Restore data files which are backup to U disk to system CNC data

backup to C disk:Backup current parameters,tool compensation,screw- pitch and ladder to system disk

CNC data backup to U disk :Backup current parameters,tool compensation,screw- pitch and ladder to U disk





PASSWORD SETTING: For user operation level display and setting

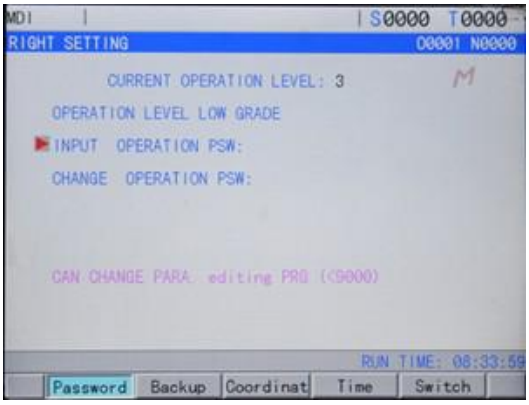
By descending sequence the password of TAC2000 is classified for 4 levels, which are machine builder (2) level, equipment management (3) level, technician (4) level, machining operation (5) level.

Machine builder level: The CNC bit parameter, data parameter, screw-pitch parameter, tool offset data, part program edit(including macro), PLC ladder editing and alteration, ladder upload and download operations are allowed;

Equipment management level: Initial password is 12345, the CNC bit parameter, data parameter, tool offset data, part program edit operations are allowed;

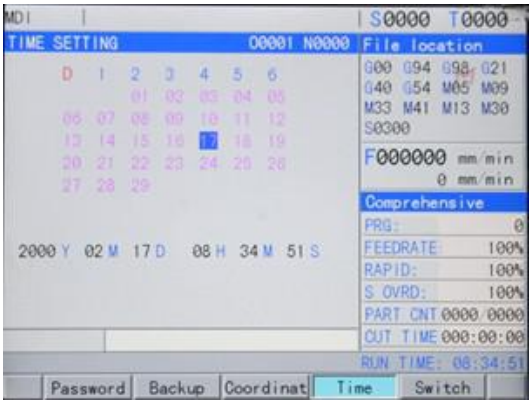
Technician level: Initial password is 1234, tool offset data (for toolsetting), macro variables, part program edit operations are allowed; but the CNC bit parameter, data parameter, screw-pitch parameter operations are forbidden.

Operation level: No password. Only the machine panel operation is allowed, the operations of part program edit and selection, the alteration operations of CNC bit parameter, data parameter, screw-pitch parameter, tool offset data are forbidden.



Time, data: display current time and date.

Press CHANGE to enter the change mode, press ⇦ , ⇨ to change Year/ Month/ Day/ Minute/ Second, press ⇧ , ⇩ to change the value. If the system escapes the mode, CHANGE is pressed again.



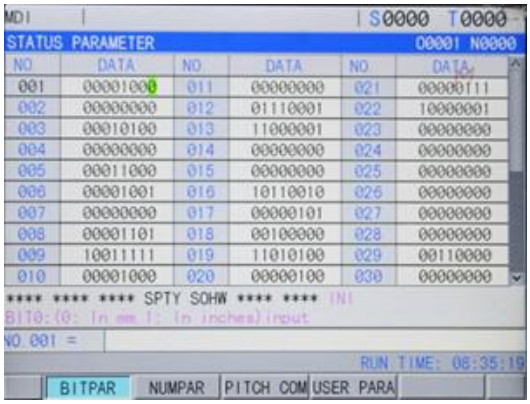
1.3.6 BIT PARAMETER, DATA PARAMETER, SCREW-PITCH COMP interfaces

PARAMETER is a compound key, it enters BIT PARAMETER, DATA PARAMETER and SCREW-PITCH COMP interfaces by pressing this key repeatedly.

1) BIT PARAMETER interface

Press PARAMETER key, it enters BIT PARAMETER interface, there are 60 bit parameters which are displayed by 2 windows in this interface, and they can be viewed or altered by pressing or key to enter the corresponding window. It is shown as follows:

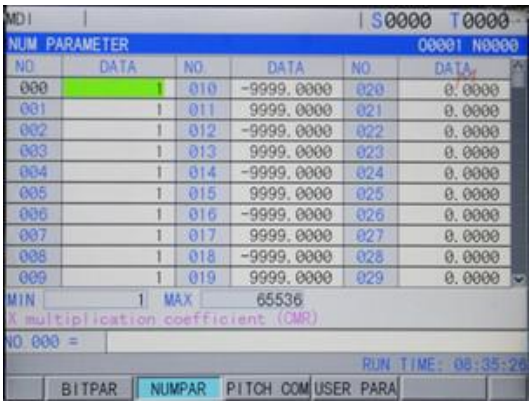
As is shown in this window, there are 2 parameter rows at the window bottom, the 1st row shows the meaning of a bit of a parameter where the cursor locates, the bit to be displayed can be positioned by pressing D or S key. The 2nd row shows the abbreviation of all the bits of a parameter where the cursor locates.



2) DATA PARAMETER interface

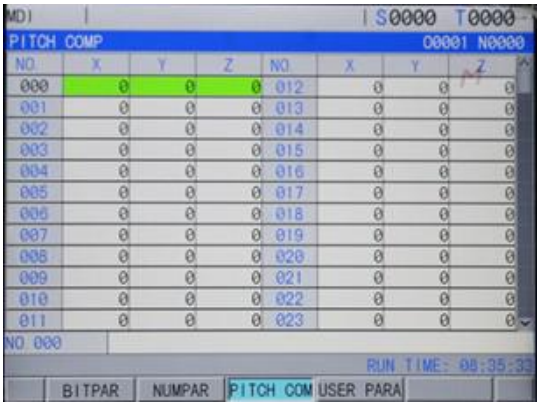
Press PARAMETER key repeatedly (key if in BIT PARAMETER interface), it enters DATA PARAMETER interface, and they can be viewed or altered by pressing or key to enter the corresponding window. It is shownbelow:

As is shown in this window, there is a cue line at the window bottom, it displays the meaning of the parameter where the cursor locates.



3) SCREW-PITCH COMP interface

Press PARAMETER key repeatedly, it enters SCREW-PITCH COMP interface, there are 256 screw-pitch parameters which are displayed by 11 windows in this interface, and they can be viewed by pressing or key.



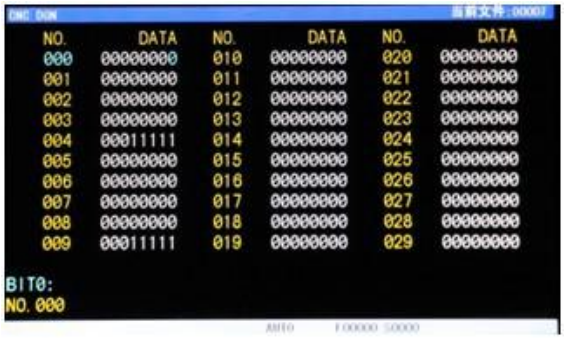
1.3.7 CNC DIAGNOSIS, PLC STATE, MACHINE SOFT PANEL, VERSION MESSAGE,HELP MESSAGE interfaces

DIAGNOSE is a compound key, it enters CNC DIAGNOSIS, PLC STATE, PLC VALUE, TOOL PANEL, VERSION MESSAGE interfaces by pressing this key repeatedly.

1) CNC DIAGNOSIS interface

The input/output signal state between CNC and machine, the transmission signal state between CNC and PLC, PLC internal data and CNC internal state can all be displayed via diagnosis. Press DIAGNOSE key it enters CNC DIAGNOSIS interface, the keypad diagnosis, state diagnosis and miscellaneous function parameters etc. can be shown in this interface, which can be viewed by pressing or key.

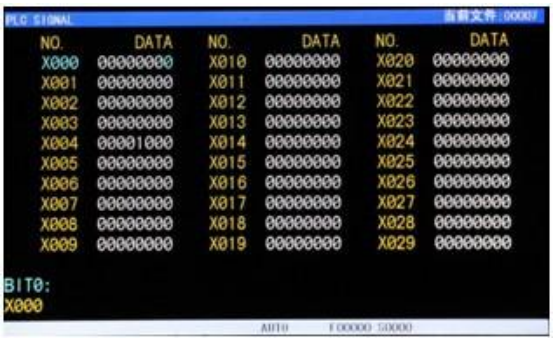
In CNC DIAGNOSIS window, there are two diagnosis rows at the window bottom, the 2nd row shows the meaning of a bit diagnosis No. where the cursor locates, the bit to be displayed can be positioned by pressing S] or D_L key. The 1st row shows the abbreviation of the bit diagnosis number where the cursor locates.



2) PLC SIGNAL interface

In the window of this interface, it orderly displays the state of address X0000~X0063, Y0000~Y0047, F0000~F063, G0000~G063 etc.. And it enters PLC STATE interface by pressing DIAGNOSE key repeatedly. The signal state of PLC addresses can be viewed by pressing or key.

In PLC STATE window, there are 2 rows at the window bottom, the 2nd row shows the meaning of a bit of an address where the cursor locates, the bit to be displayed can be positioned by pressing S] or D_L key. The 1st row shows the abbreviation of the bit address number where the cursor locates.



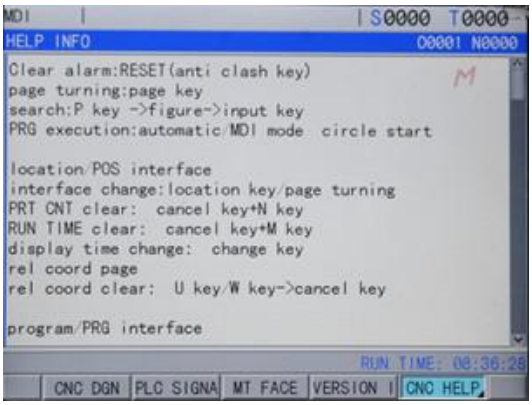
3) MACHINE SOFT PANEL

Press DIAGNOSE key repeatedly, it enters MACHINE SOFT PANEL. In this page,it can control soft keyboard for machine.The page displays as follow:

Note:It can be switched between row and row by pressing change key

4)HELP MESSAGE interface

It enters HELP MESSAGE interface by pressing DIAGNOSE key repeatedly. The operation list, alarm list, G command list ,and macro command message can be shown in this interface. As is shown in the following figure:



5) VERSION MESSAGEinterface

It enters VERSION MESSAGE interface by pressing DIAGNOSE key repeatedly. The software, hardware, and PLC version message can be shown in this interface. As is shown in the following figure:

Chapter IIPower ON/OFF and Safety Protection

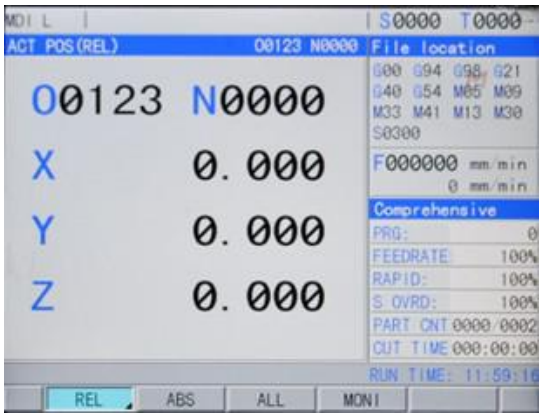
2.1 Power ON

Please confirm before TAC2000 CNC system is powered on:

1) The machine status is normal;
2) The power supply voltage meets the requirements;
3) The connection is correct and firm.

The interface is shown as below after TAC2000 is powered on:

Then, TAC2000 is self-detected and initialized. After the self-detection and initialization, the page of the present position (the relative coordinate) is displayed asbelow:



2.2 Power OFF

Please confirm before power off:

1) The feeding axis of CNC is in stopping status;
2) The miscellaneous function (such as the spindle, the water pump, etc) isoff;
3) Firstly cut off CNC power supply, and then the machine one.

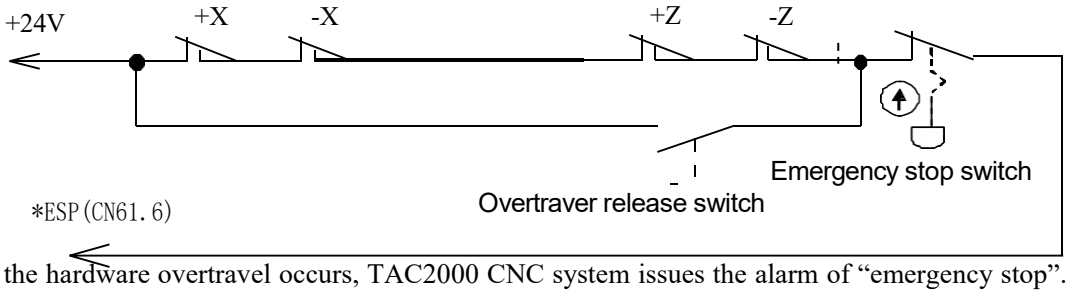
Note: About the operation of switching off the machine power supply, please refer to the user manual of the machine manufacturer.

2.3 Overtravel Protection

The machine must be adopted the overtravel protection measure to avoid the X ,Y and Z axes overtravel causing the machine damage.

2.3.1 Hardware Overtravel Protection

The positive and negative maximum travel positions of the X ,Y and Z axes are installed with the travel limit switch on the machine, and the connection is operated based on the following figure. And then, Bit2(EALM) and Bit3(LALM) of the status parameter No.60 must be set as 0. When the overtravel occurs, the limit switch is operated, and TAC2000 CNC system stops the movement and displays the emergency stop alarm.



When the hardware overtravel occurs, TAC2000 CNC system issues the alarm of “emergency stop”. The method of clearing the alarm of “emergency stop”: Press the overtravel release button without releasing, switch into the alarm information page, and check the alarm information. After clearing the alarm by resetting, the worktable is moved in the opposite direction (for example, move it in the negative direction if it is the positive overtravel; vise versa), and the limit switch is released.

2.3.2 Software Overtravel Protection

The software travel stroke is set by data parameter NO.10~ NO.19, they refer to machine coordinate. Bit1(LZR) of No.22 can be set if the software limit is valid before machine return zero

If the machine position (coordinate) exceeds the setting range, overtravel alarm will occur. The steps to eliminate this alarm is press RESET key to clear the alarm, then moves reversely (for positive overtravel, move out negatively; vice versa)

2.4 Emergency Stop Operation

During machining, due to the user programming, operation and product fault, etc, some unexpected situations may occur, and TAC2000 CNC system must be stopped immediately. In the chapter, it mainly introduces the process for TAC2000 CNC system in the emergency situation; about the machine one, please refer to the relative explanation of the machine manufacturer.

2.4.1 Resetting

When the TAC2000 CNC system is abnormally output and the coordinate axis is moved,

press RESET key to make the TAC2000 CNC system in the resetting status:

1) All axes movement stops;
2) M and S function output is invalid (whether automatically switch off the signals of the spindle rotation, the lubrication and the cooling, etc after pressing RESET key is set by the parameter, the ladder diagram definition);
3) After the automatic running ends, the modal function and status remain.

2.4.2 Emergency Stop

Press the emergency stop button (when the external emergency stop signal is valid) in the dangerous or in the emergency situation during the machine running, and then CNC enters the emergency stop status, then:

The machine movement stops immediately, the output of the spindle rotation and the coolant are all switched off. Release the emergency stop button to release the alarm and CNC enters the resetting status. About the circuit connection method, please refer to the chapter2.3.1.

Note 1: Please confirm the troubleshooting is completed before releasing the emergency stop alarm;

Note 2: Pressing the emergency stop button before power ON/OFF can reduce the electric shock upon the equipment;

Note 3: Please execute the machine zero return after the emergency stop alarm is released, so the correctness of the coordinate position can be guaranteed (if the machine isn't installed with the machine zero, the machine zero return should NOT be operated):

Note 4: Only when Bit 2 of the status parameter NO.60 is set as 0, the external emergency stop is valid.

2.4.3 Feed hold

During the machine tool running, press 

to make the running dwell. Please pay attention to that the function can't make the running stop immediately during the thread cutting and the tapping cycle.

2.4.4 Cut off the Power Supply

During the machining tool running, the machine power supply should be cut off immediately in the dangerous or in the emergency situation to avoid the accident. But please pay attention to that the coordinate displayed on CNC has the big deviation with the actual position after power off, so resetting the tools should be operated.

Chapter III Manual Operation

Note!

The function of the buttons on TAC2000 machine tool panel is defined by PLC program (the ladder diagram), and about the function and meaning of each button, please refer to the manual of the machine tool manufacturer.

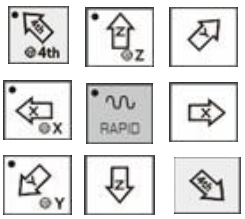
Please pay attention to that the following chapter about the relative function of the operation panel buttons is mainly for TAC2000 standard PLC program!

Press  key to enter the Manual mode. In Manual mode, the manual feed, the spindle control, the override adjusting and the tool change, etc can be operated.

3.1 The Coordinate Axis Movement

In Manual mode, two-axis manual feed and manual rapid traverse can be executed.

3.1.1 Manual Feed

Press  or  of  to make X axis feeding in positive or negative direction, and the axis movement stops after releasing the button; press  or  to make Z axis feeding in positive or negative direction, and the axis movement stops after releasing the button; press  or  to make the Y axis feeding in positive or negative direction, and the Y axis movement stops after releasing the button; press  or  to make the 4th axis feeding in positive or negative direction, and the axis movement stops after releasing the button;

During the manual feeding, press RAPID and the indicator lamp is ON, the system enters the manual rapid traverse status.

3.1.2 Manual Rapid Movement

Press RAPID of 4th Z X RAPID Y Z to make the indicator lamp ON, press X or X to make X axis rapid traverse in positive or negative direction, and the axis movement stops after releasing the button; press Z or Z to make Z axis rapid traverse in positive or negative direction, and the axis movement stops after releasing the button; press Y or Y to make Y axis rapid traverse in positive or negative direction, and the axis movement stops after releasing the button; press 4th or to make the 4th axis rapid traverse in positive or negative direction, and the axis movement stops after releasing the button;

During the manual rapid traverse, press RAPID to make the indicator lamp OFF, the rapid traverse is invalid, and the the feeding is executed in manual feedrate.

Note 1: After power ON, if the reference position isn't returned and the rapid movement switch (the rapid movement button indicator lamp is ON), whether the rapid traverse feedrate is manual feedrate or rapid traverse rate is set by Bit0(ISOT) of TAC2000 the status parameter NO.012.

Note 2: In Edit/MPG mode, RAPID key is invalid.

3.1.3 Speed tune

In Manual mode, MW%+ F. OVERRIDE MW%− F. OVERRIDE or can be pressed to alter the manual feedrate override that has 16 steps. The relation of the override and the feedrate is as follows table if data parameter No.031 is set to 1260:

Feedrate override (%)	Feedrate (mm/min)
0	0
10	126
20	252
30	378
40	504
50	630
60	756

Feedrate override (%)	Feedrate (mm/min)
70	882
80	1008
90	1134
100	1260
110	1386
120	1512
130	1638
140	1764
150	1890

Note : There is about 2% error for the data in the above table.

In the manual rapid traverse, it can press 

or 

key to modify the rapid override, and there are 4 steps of F0, 25%, 50%,100% for the override.(F0 set by data parameter No.085)

3.2 Other Manual Operation

3.2.1 CCW Rotation,CW Rotation,Stop controlling

:In manual mode, press the key,the spindle CW rotate;

:In manual mode, press the key,the spindle start or stop;

: In manual mode, press the key,the spindle decelerate rotate;

3.2.2 Spindle jog

: At the moment, the spindle is in JOG state.

Functional description:Press 

to enter JOG mode, and the spindle JOG function ON/OFF is executed only when the spindle is in the state of stop.

In spindle JOG mode, by pressing 

key, the spindle rotates counterclockwise for jogging;

by pressing 

key, the spindle rotates clockwise for jogging. The jog time and speed are set by data parameter No.108 and No.109 respectively.

When the spindle JOG rotates, 

is pressed to stop the spindle JOG rotation, the spindle brake signal is not output when the JOG rotation stops.

K10.4 is set to 1, the spindle JOG is valid in any mode. In Auto or MDI mode, the spindle is in the JOG rotation state, the program closes the spindle JOG rotation and the JOG function.

Parameter setting:

PLC parameter K104 1/0: the spindle JOG is valid in any mode/Manual, MPG, Zero return mode. Data parameter No.208: rotary speed in spindle JOG.

3.2.3 Cooling control

: In Manual mode, press this key, the cooling is switched on/off.

Parameter setting: PLC parameter K10.1 1/0: the spindle lubricating and cooling output remains/closes in reset.

3.2.4 Lubricating control

Function description:

1. Non-automatic lubricating

DT13 =0: For non-automatic lubricating

When DIT13=0, it is lubricating turn output, by pressing the 

key, the lubricating is output. And the lubricating is cancelled by pressing it again. M32 is for lubricating output, and M33 is for lubricating output cancellation.

When data parameter DIT13＞1, it is timing lubricating output, by pressing the 

key, the lubricating is output. And it is cancelled after a setting time by data parameter No.112; by executing M32, the lubricating is output, and it is cancelled after a setting time by data parameter No.112. If the setting time is not yet up, M33 is executed to cancel the lubricating output.

2. Automatic lubricating

DT13>0: For automatic lubricating, the lubricating time DT13 and lubricating interval time DT53 may be set.

After TAC2000 system is switched on, it is lubricating for a time set by DT13, then the lubricating output stops. After an interval set by DT53, the lubricating is output again, so it cycles by sequence. In the automatic lubricating, M32, M33 codes as well as the 

key are all inactive.

Parameter setting:

PLC parameter: K10.1 1/0: the spindle lubricating/cooling output remains/closes in reset.

PLC parameter:K16.2 1/0: whether the lubricating outputs in power-on when the automatic lubricating is valid.

PLC data: DT53 automatic lubricating interval time (ms) PLC data: DT13: automatic lubricating output time (ms)

DT15: M execution duration(ms)

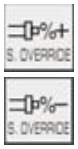
DT13: lubricating start time (0-60000ms)(0:lubricating time is not limited)

3.2.5 Manual tool change

: In Manual mode, by pressing this key, the spindle tools is loosed/clamping

3.2.6 Spindle override

In Manual mode, if the spindle speed is controlled by analog voltage output, the spindle speed may be overrided.

By pressing 

in Spindle Override keys, the spindle speed can be changed by real-time adjusting of the spindle override that has 8 steps of 50％～120％.

Chapter IV MPG/Single Step Operation

In MPG/single step mode, the machine is moved based on the selected incremental value.

NOTE!

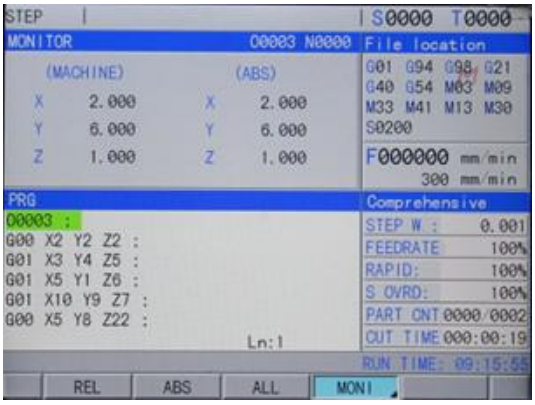
The function of the buttons on TAC2000 machine tool panel is defined by PLCprogram (the ladder diagram), and about the function and meaning of each button, please refer to the manual of the machine tool manufacturer.

Please pay attention to that the following chapter about the relative function of the operation panel buttons is mainly for TAC2000 standard PLC program!

4.1 Single Step Feeding

Set the system parameter No.001 Bit3 to 0, and press mode, it 

key to enter the STEP working displays as follows:



4.1.1 Increment selection

Press X1 F0 X10 25% X100 50% X1000 100% key to select the movement increment, and the movement increment is displayed on the page. When BIT7 (SINC) of PLC status parameter K016 is set as 1, the step length value X1000 100% is invalid; when BIT7 is 0, X1 F0 X10 25% X100 50% X1000 100% all are valid. If X1000 100% key is pressed, the page is shown as below:

4.1.2 Selecting the Movement Direction

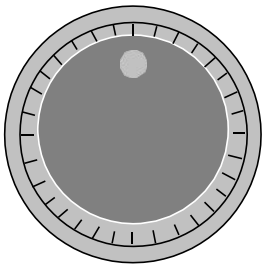
Press X or X once, X axis is fed once based on the single step increment in negative or positive direction; press Z or Z, Z axis is fed once based on the single step increment in negative or positive direction;press in Y or Y, Y axis is fed once based on the single step increment negative or positive direction.

4.2 MPG (manual pulse generator) feeding

Set the BIT3 of the system parameter No.001 to 1, andpress it 

key to enter the MPG mode, displays as follows:

MPG outline is shown as the following figure:



MPG outline figure

4.2.1 Increment selection

Press X1 F0 X10 25% X100 50% X1000 100% key to select the movement increment and it will be displayed on the page. When BIT7（SINC）of PLC K016 parameter is set as 1, X1000 100% step width value is invalid; when BIT7 is 0, X1 X10 X100 X1000 all are valid. If X1000 100% key is pressed, the page is shown as below:

4.2.2 Selecting the Movement Axis and the Direction

In MPG mode, press X, Y, Z, 4th key to select the corresponding axis.

MPG feeding direction is set by MPG rotation direction. Normally, MPG CW rotation is the positive feeding and CCW rotation is the negative feeding. If sometimes MPG CW rotation is negative feeding and CCW rotation is positive feeding, the signals A and B at MPG end can be exchanged or the feeding direction of MPG rotation is selected by BIT0 of parameter No.013.

4.2.3 Other Operation

1） Spindle CCW/CW rotation, stop

:In MPG/single step mode, press the key to make the spindle CW rotation;

: In MPG/single step mode, press the key to make the spindle stop;

: In MPG/single step mode, press the key to make the spindle CCW rotation;

2） Spindle Jog

: at the moment, the spindle is in JOG working mode.

In spindle Jog mode, by pressing 

key, the spindle rotates counterclockwise for jogging; by pressing 

key, the spindle rotates clockwise for jog. The jogging time and speed are set by data parameter No.208 respectively. The concrete is referred to Chapter 3.2.2.

3） Cooling control

Refer to OPERATION, Chapter 3.2.3

4） Lubricating control

Refer to OPERATION, Chapter 3.2.4

5) Manual tool change

: In MPG/Step mode, press it to execute the tool change orderly.

6) Spindle override tune

In MPG/Step mode, if the spindle speed is controlled by analog voltage output, the spindle speed may be overrided.

By pressing 

in Spindle Override keys, the spindle speed can be changed by real-time adjusting of the spindle override that has 8 steps of 50%～120%.

4.2.4 Explanation items

The relation between MPG scale and the machine movement amount is shown as the following list:

	Moving amount of each MPG scale			
MPG increment	0.001	0.01	0.1	1
Specified coordinate value	0.001mm	0.01mm	0.1mm	1mm

（Taking example of the least input increment 0.001mm）

Note 1: MPG increment is relative with the current metric/inch system and the system least input increment. Note 2: The speed of MPG rotation should NOT be higher than 5r/s; otherwise, the scale value may NOT be complied with the movement amount.

Chapter V MDI Operation

In MDI mode, setting the parameters, inputting the single block and executing the single block can be performed.

NOTE!

The function of the buttons on TAC2000 machine tool panel is defined by PLCprogram (the ladder diagram), and about the function and meaning of each button, please refer to the manual of the machine tool manufacturer.

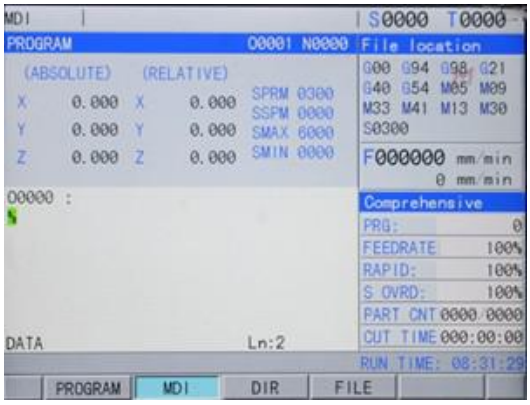
Please pay attention to that the following chapter about the relative function of the operation panel buttons is mainly for TAC2000 standard PLC program!

5.1 MDI the Block

Select MDI mode, enter the program-> MDI program page, input one block G50 X50 Z100, the operation steps are as below:

1) Press MDI key to enter MDI mode;

2) Press PROGRAM key (or or press PROGRAM many times if needed) to enter PRG STATE window:

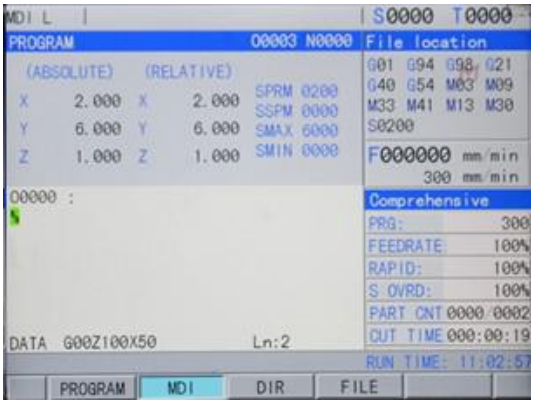


3） Input the address key G* and the digit keys 5 and 0 in order;

4） Input the address key Z: and the digit keys 1, 0 and 0 in order;

5） Input the address key X and the digit keys 5 and 0 in order.

The window is shown as follows after above operations are completed(can input 4 block and display 6 block):



5.2 Executing the Block

After inputting the block, press DATA INPUT key to display the following page:

After inputting the block, press DATA INPUT to confirm and then press CYCLE START to execute the inputted block.

During running, press FEED HOLD , RESET and emergency stop buttons to stop the program running.

Note: The subprogram calling code（M98 P ；etc ）are invalid in MDI mode.

5.3 Setting the Parameters

In MDI mode, enter the parameter interface to rewrite the parameter values; about the details, refer to chapter 10in the section.

5.4 Rewriting the Data

In the PRG STATE window of MDI mode, if there is an error during words inputting, CANCEL is pressed to cancel the display or RESET is pressed to clear all the input, then re-input the correct ones.

5.5 Other Operation

1） Adjusting the spindle override

In MPG/STEP mode, the spindle speed can be adjusted when the spindle speed is controlled by outputting the analog voltage.

The spindle speed can be changed by adjusting the spindle override by pressing which can realize the spindle overrides 50％～120％, total 8-level real-time adjustment.

2） Adjusting the rapid override

The rapid traverse feedrate can be adjusted by pressing X1 F0 X10 25% X100 50% X1000 100% keys and the rapid traverse rate of 4-level real-time adjusting can be realized.

3） Feedrate override is available.

In MDI mode, by pressing F. OVERRIDE, the actual speed real-time adjusting of 0～150％ feedrate by F code can be done by the override that has 16 levels

Chapter VI Editing and Managing the Programs

In Edit mode, the program can be set, opened, rewritten, copied and deleted, and dual-channel communication between CNC and PC can be realized.

To avoid the program is rewritten and deleted by accident, the program switch is set in TAC2000 CNC system. Before editing the program, the program switch must be opened; about setting the program switch, refer to chapter 10.1.1 of the section.

For convenient management, the multi-level user authority management is set in TAC2000 CNC system. The program switch ON/OFF and editing the program can be executed only when the user is with the operation authority above level 4 (level 4 or 3,etc). About the operation for each level authority, refer to chapter 10.1.2.

6.1 Setting the Program

6.1.1 Generating the Block No.

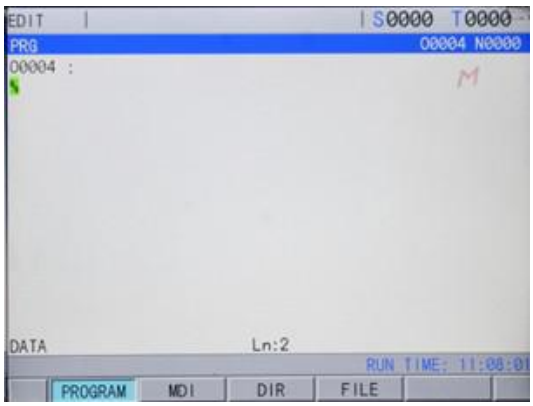
In the program, the block number can be edited and it's OK if it isn't edited. And the program is executed based on the input order. (Except that it is calling.).

On the setting switch page, when "automatic sequence number" is OFF, CNC doesn't generate the block number automatically, while the block number can be input manually during programming.

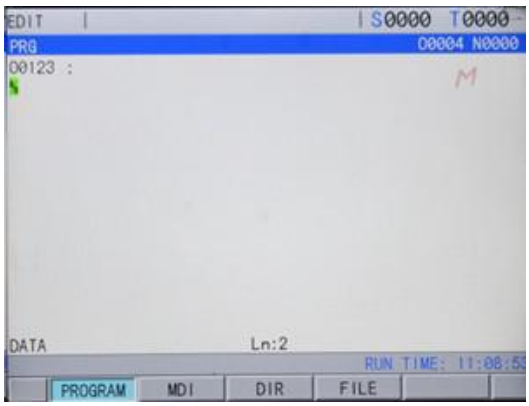
On the setting switch page, when "automatic sequence number" is ON, CNC generates the block number automatically. During editing, press EOB key to automatically generate the block number of the next block, and the incremental value of the block number is set by the data parameter No. 042 of CNC data (about setting the switch of the automatic sequence number, refer to chapter 10.1.1 of the section).

6.1.2 Inputting the Program Content

1. Press EDIT key to enter the Edit mode and press PROGRAM key to enter the program page set.



2. On the program content page, press the address key O, and then input the numerical keys 0, 1, 2, and 3 on the pop-up dianogue box (setting O0001 program is taken as one example, and the leading zero before inputting the program name can be omitted), the display is as below:
3. Press EOB key to set a new program, the display is as below:



4. Input the edited part program one by one, the character will be displayed on the screen immediately as it is input(as for compound key, press this key repeatedly for alternate input), after a block is finished, press EOB key to terminate it.
5. Other blocks input may be finished by step 4 above.

6.1.3 Movement of cursor

1) Press EDIT key to enter the Edit mode, then press PROGRAM key to enter the PRG CONTENT window;

2） Press ⇧ key, the cursor shifts a row upward; if the number of the column where the cursor locates is over the total columns of the previous row, the cursor moves to the previous block end (at ";" sign) after ⇧ key ispressed;

3) Press ⇩ key, the cursor shifts a row downward; if the number of the column where the cursor locates is over the total columns of the next row, the cursor moves to the next block end (at ";" sign) after the ⇩ key ispressed;

4) Press ⇨ key, the cursor shifts a column to the right; if the cursor locates at the row end, it moves to the head of the next block;

5) Press ⇦ key, the cursor shifts a column to the left; if the cursor locates at the row head, it moves to the end of the next block;

6) Press key to window upward

7) Press key to window downward,

6.1.4 Searching No. of character and line

Searching a character: To search for the specified character upward or downward from the cursor current location

The steps of finding is as follows:

1) Press EDIT key to enter Edit mode;

2) Press PROGRAM key to enter the PRG CONTENT window;

3) Input the characters to be searched , the characters over the 10th byte will be ignored.

4) Press ⇧ key (⇧ or ⇩ is determined by the location of the character searched to the character where the cursor locates), it displays as follows:

5) After the finding, the CNC system is still in FIND state, press ⇧ or ⇩ key again, the next character can be found.

Note 1: If the character is not found, the searching character will disappear
Note 2: During the searching, it doesn't search the characters in the called subprogram, and the character in subprogram is searched in subprogram.
Note 3: The system cannot search and scan the character in macro edit mode.

Searching a line:Put the cursor rapidly move to specified line of program The steps of finding is as follows:

1) Press EDIT key to enter Edit mode;

2) Press PROGRAM key to enter the PRG CONTENT window;

3) Input the line No. to be searched

4)Press DATA INPUT key,the cursor will skip to specified line

6.1.5 Inserting a character

Steps:

1) Select the PRG CONTENT window in Edit mode;

2) Press INSERT key to enter the INS mode (the cursor is an underline), the window is as follows:

3) Input the character to be inserted

6.1.6 Deleting a character

Steps:

1) Enter the PRG CONTENT window in Edit mode;

2) Move the cursor to the location where you want to delete, press DELETE key to delete the character where the cursor locates;

6.1.7 Altering a character

Steps:

1) Enter the PRG CONTENT window in Edit mode;

2) Move the cursor to the location where you want to alter, press ALTER key to alter the character instead of the input content

6.1.8 Deleting a single block

Steps:

1) Select the PRG CONTENT window in Edit mode;

2) Move the cursor to the head of the block to be deleted (column 1), press N# key, Andthen press DELETE key to delete a single block

6.1.9 Deleting blocks

Steps:

1) Select the PRG CONTENT window in Edit mode;

2) Move the cursor to head line of a block that you want to delete

3) Input the sequence No. of the last block that you want to delete

4) Press DELETE key, the blocks among cursor and marked address will be deleted

6.1.10 Deleting a segment

Steps:

1) Select the PRG CONTENT window in Edit mode;
2) Move the cursor to the 1st character of a block that you want to delete
3) Input the last character of the block that you want to delete
4) Press DELETE key, the segment among cursor and marked address will be deleted

6.1.11 Copying a single block

Steps:

1) Select the PRG CONTENT window in Edit mode;
2) Move the cursor to head line of a block that you want to copy
3) Press N# keyfirst, then press CHANGE ,copy the block where the cursor located in

6.1.12 Copying blocks

Steps:

1) Select the PRG CONTENT window in Edit mode;
2) Move the cursor to 1st character of a block that you want to copy
3) Input the sequence No. of the last block that you want to copy
4) Press CHANGE key, the blocks among cursor and inputed character will be copyed

6.1.13 Copying a segment

Steps:

1) Select the PRG CONTENT window in Edit mode;
2) Move the cursor to the 1st character of a block that you want to delete
3) Input the last character of the block that you want to delete
4) Press CHANGE key, the segment among cursor and inputed character will be copyed

6.1.14 Pasting a single block

Steps:

1) Select the PRG CONTENT window in Edit mode;
2) Move the cursor to location of program that you want to paste
3) Press DATA OUTPUT key, insert the last copy program content before cursor to finish paste operation

6.2 Deleting program

6.2.1 Deleting a program

Steps:

1) Select the PRG CONTENT window in Edit mode;
2) Press address key O, number key 0, 0, 0, 1 by sequence (by program O0001);
3) Press DELETE key, program O0001 will be deleted.

6.2.2 Deleting all programs

Steps:

1) Select the PRG CONTENT window in Edit mode;
2) Press address key O, symbol key −+ >, number key 9, 9, 9, 9;
3) Press DELETE key, all the programs will be deleted.

6.3 Selecting a program

When there are multiple programs in CNC system, a program can be selected by the following 3 methods:

6.3.1 Search

1) Select Edit or Auto mode;
2) Press PROGRAM key to enter the PRG CONTENT window;
3) Press address key O and key in the program No.;
4) Press ⇩ or EOB key, or press ⇩ key in Auto mode, the searched program will be displayed. If the program does not exist, an alarm will be issued by CNC.

Note: In step 4, if the program does not exist in Edit mode, a new program will be created by CNC system after key is pressed. EOB

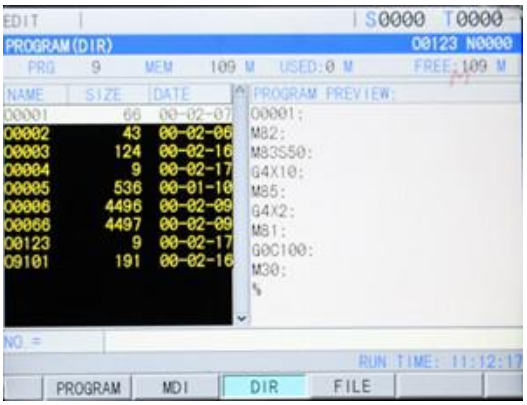
6.3.2 Scanning

1) Select Edit or Auto mode;
2) Press PROGRAM key to enter the Program window;
3) Press address key O;
4) Press ⇩ or ⇧ key to display the next or previous program;
5) Repeat step 3 and 4 to display the saved programs one by one.

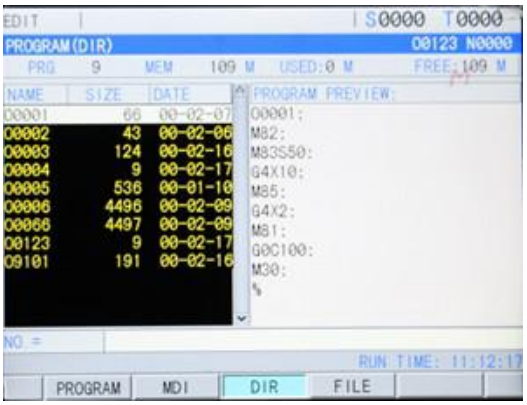
6.3.3 Cursor

1) Select Auto mode **(must be in non-running state)**;

2) Press PROGRAM key to enter the PRG LIST window;



3) Press ⇧, ⇩, key to move the cursor to the program name to be selected (Program content will change as the cursormoves);



4) Press EOB key or DATA INPUT.

6.4 Renaming a program

1) Select the PRG CONTENT window in Edit mode;

2) Press address key O, and key in the new program name;

3) Press ALTER key.

6.5 Copy a program

To save the current program to a location:

1) Select the PRG CONTENT window in Edit mode;

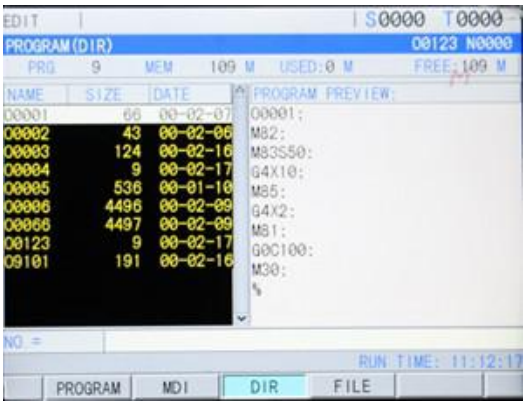
2) Press address key O, and key in the new program No.;

3) Press CHANGE key.

6.6 Program management

6.6.1 Program list

In non-Edit mode, press PROGRAM key to enter the PRG LIST window. In this window, it lists the program names saved in CNC system, and it can display maximum 10 names in a window, if the programs saved exceed 10, itmay press key to display the other program list.



1) Open program

Open specified program : O + No.+ EOB

In Edit mode, it will creat program if program No. is not exit.

2) Delete program

1. In Edit mode, press DELETE to delete program specified by cursor 2.In Edit mode,press O + No. + DELETE or No. + DELETE

6.6.2 Part-Prg number

It shows the total number of the part programs (up to 400) that can be saved in CNC system and the current part programs number that have been saved at present.

6.6.3 Memory size and usedcapacity

They show the total memory capacity (56M) of the CNC and the current capacity that has been occupied.

CHAPTER VII AUTO OPERATION

Note !

The key functions of TAC2000 machine panel are defined by PLC program (ladders), please refer to the materials by the machine builder for their significance.

Please note that the following description for the keys function in this chapter is based on TAC2000 standard PLC program!

7.1 Automatic run

7.1.1 Selection of the program to berun

1. Searching method

1) Select the Edit or Auto mode;

2) Press PROGRAM key to enter the PRG CONTENT window;

3) Press the address key O, and key in the program No.;

4) Press ⇩ or EOB key, the program retrieved will be shown on the screen, if the program doesn't exist, an alarm will beissued.

2. Scanning method

1) Select the Edit or Auto mode;

2) Press PROGRAM key to enter the PRG CONTENT window;

3) Press the address key O

4) Press the ⇩ or ⇧ key to display the next or previousprogram;

5) Repeat the step 3, 4 above to display the saved program one by one.

3. Cursor method

a) Select theAuto mode (in non-run state);

b) Press PROGRAM key to enter the PRG LIST window (press or key ifneeded);

c) Press ⇧, ⇩, ⇦ or ⇨ key to move the cursor to the name of the program to be selected;

d) Press EOB key.

7.1.2 Start of the automatic run

1. Press AUTO key to select the Auto mode;

2. Press CYCLE START key to start the program, and the program automatically runs.

Note: Since the program execution begins from the block where the cursor locates, before pressing the CYCLE START key, make a check whether the cursor is located at the block to be executed.

7.1.3 Stop of the automatic run

***Stop by code (M00)**

1. M00

After the block containing M00 is executed, the auto run is stopped. So the modal function and state are all reserved. Press the CYCLE START key or the external run key, the program execution continues.

2. M01

Press OPTIONAL STOP and the optional stop indicator is ON and the function is valid. After the block with M01 is executed, the system stops the automatic run, the modal function and the state are saved.

Press CYCLE START or the external run key, and the program continuously runs.

*** Stop by a relevant key**

1. In Auto run, by pressing CYCLE STOP key or external dwell key, the machine keeps the following state:

(1) The machine feed slows down to stop;

(2) The modal function and state are reserved;

(3) The program execution continues after pressing the CYCLE START key.

2. Stop by Reset key RESET

(1) All axes movement is stopped.

(2) M, S function output is inactive (the automatic cut-off of signals such as spindle CCW/CW, lubricating, cooling by pressing RESET key can be set by the parameters)

(3) Modal function and state is held on after the auto run.

3. Stop by Emergency stop button

If the external emergency button (external emergency signal active) is pressed under the dangerous or emergent situation during the machine running, the CNC system enters into emergency state, and the machine moving is stopped immediately, all the output (such as spindle rotation, cooling) are all cut off. If the Emergency button is released, the alarm is cancelled and CNC system enters into reset mode.

4. Switching operation mode

When Auto mode is switched to the Machine zero, MPG/Step, Manual, Program zero mode, the current block “dwells” immediately; when the Auto mode is switched to the Edit, MDI mode in Auto

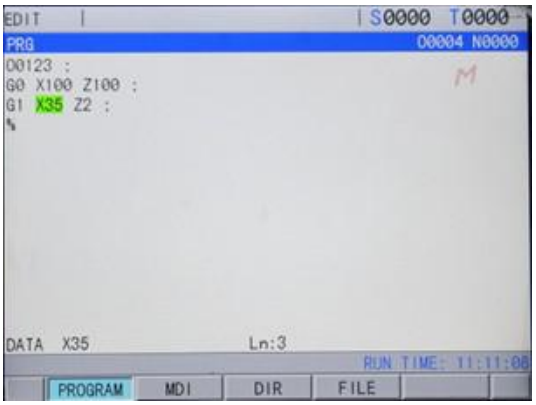
mode, the “dwell” is not displayed till the current block is executed.

Note 1: Ensure that the fault has been resolved before cancelling the emergency alarm.
Note 2: The electric shock to the device may be decreased by pressing the Emergency button before power on and off.
Note 3: The Machine zero return operation should be performed again after the emergency alarm is cancelled to ensure the correctness of the position coordinates (but this operation is forbidden if there is no machine zero in the machine).
Note 4: Only the BIT2 (EALM) of the bit parameter No.215 is set to 0, could the external emergency stop be active.

7.1.4 Automatic run from an arbitrary block

Press EDIT key to enter the Edit mode, press PROGRAM key to enter the Program interface, then press or key to enter the PRG CONTENT window:

1. Move the cursor to the block to be executed (for example, move the cursor to the 3rd row head if it executes from the 3rd row);



2. If the mode (G, M, T, F code) of the current block where the cursor locates is defaulted and inconsistent with the running mode of this block, the corresponding modal function should be executed to continue next step.

3. Press AUTO key to enter the Auto mode, then press CYCLE START key to start the execution.

7.1.5 Adjustment of the feedrate, rapid rate

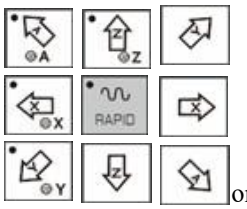
In Auto mode, the running speed can be changed by adjusting the feedrate override, rapid override. It doesn’t need to change the settings of the program and parameter.

*** Adjustment of the feedrate override**

Press F. OVERRIDE , 16-level real time feedrate can be obtained.

Note 1: The actual feedrate value is specified by F in feedrate override adjustment; Note 2: Actual feedrate= value specified by F× feedrate override

*** Adjustment of rapid override**

It can realize the F0, 25 %, 50%, 100 % 4-level real time rapid override adjustment by pressing  or  .

Note 1: The rapid traverse speeds of all axis are set by the CNC parameter No.090~No.094 respectively; Actual rapid traverse rate = value set by parameter × rapid override

Note 2: When the rapid override is F0, the min. rapid traverse rate is set by bit parameter No.085. Note 3: For spinning machine,only after finish returning zero can run program automatically .

7.3 State of running

7.3.1 Single block run

In Auto mode, the methods for turning on single block switch are as follows:

Method 1: Press the  key to make the single block indicator in panel state area to light up, it means that the single block function has been selected;

In Single mode, when the current block execution is finished, the CNC running stops; if next block is to be executed, it needs to press the  key again, then repeat this operation till the whole program is finished.

Note 1: The single block stops at the mid point of G28 code.

Note 2: While the subprogram calling (M98__), or subprogram calling return (M99)is being executed, the single block is inactive. But it is active except for N, O, P addresses in the block that contains M98 or M99 code.

7.3.2 Dry run

Before the program is to be executed automatically, in order to avoid the programming errors, it may select the Dry run mode to check the program.In Auto mode, the methods for turning on the Dry run switch are as follows:

Press the  key to make the Dry run indicator in panel state area to light up, it means that the dry run mode has been selected;

In Dry run mode, the machine feed and miscellaneous functions are both active (as machine lock, MST lock are both OFF), that means the dry run switch has nothing to do with the machine feeding, MST functions, so the feedrate by program is inactive and the CNC system runs by data parameter No.082

7.3.3 Machine lock

In Auto mode, the turning on method of machine lock switch is as follows:

Method 1: Press the 

key to make the Machine Lock indicator  in panel state area to light up, it means that it has entered the machine lock state;

The machine lock and MST lock are usually used together to check the program. While as in the machine lock mode:

1. The machine carriage doesn't move, the "MACHINE" in the INTEGRATED POS window of the Position interface doesn't vary too. The RELATIVE POS and ABSOLUTE POS, DIST TO GO are refreshed continuously, which is the same as that the machine lock switch is OFF.
2. M, S, T commands can be executed normally.

7.3.4 MST lock

In Auto mode, the turning on of MST lock switch is as follows:

Method 1: Press the 

key to make the MST Lock indicator light 

in panel state area to up, it means that it has entered the MST lock state;

The machine carriage moves without the M, S, T code being executed. The machine lock and MST lock are usually used together to check the program.

Note: When the MST lock is active, it takes no effect to the execution of M00, M30, M98, M99.

7.3.5 Block skip

If a block in program is not needed to be executed and not to be deleted, this block skip function can be used. When the block is headed with "/ " sign and Block skip indicator lights up (panel key active or external skip input active), this block is skipped without execution in Auto mode.

In Auto mode, the turning on of Block skip switch is as follows:

Method 1: Press the light 

key to make the Block skip indicator  in panel state area to up;

Note: While the block skip switch is off, the blocks headed with "/" signs are executed normally in Auto mode.

7.4 Other operations

1. In Auto mode, press 

key to switch on/off the cooling;
2. Press any of the 

keys to switch the operation modes;
3. Press the 

key to reset this CNC system.
4. Automatic lubricating operation (Refer to **Volume II Operation, Chapter 3**).

CHAPTER VIII ZERO RETURN OPERATION

Note!

The key functions of this TAC2000 machine panel are defined by PLC program (ladders), please refer to the manuals by the machine builder for their significance.

Please note that the following description for the panel key functions in this chapter is based on the TAC2000 standard PLC program!

8.1 Machine Zero return

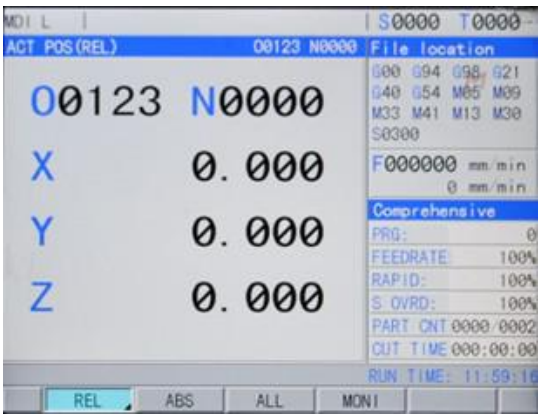
8.2.1 Machine Zero (machine referencepoint)

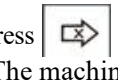
The **machine coordinate system** is a reference coordinate system for CNC coordinate operation. It is an inherent coordinate system of the machine. The origin of the machine coordinate system is called machine zero (or mechanical reference point). It is defined by the zero or zero return switch fixed on the machine. Usually this switch is fixed at the positive stroke point of X or Z axis.

8.1.2 Machine Zero return steps

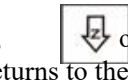
1. Press 

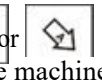
key, it enters the Machine zero mode, the bottom line of the window displays "MACHINE ZERO", as the following figure shows:



2. Press 

, 

, 

or 

key to return to the machine zero of X, Z or Yaxis;

3. The machine axis returns to the machine zero via the deceleration signal, zero signal detection. At the machine zero, the axis stops, and the corresponding machine zero return completion indicator lights up.

Machine zero return completion indicator

Note 1: If there is no machine zero on the machine, machine zero operation is forbidden; Note 2: The machine zero finish indicator is gone out on condition that:

1) The axis is moved out from machine zero;

2)CNC is powered off.

Note 3: After the machine zero operation, the tool length compensation is cancelled by CNC; Note 4: Parameters related to machine zero return are referred to Volume III INSTALLATION and CONNECTION.

Note 5: After the machine zero return is executed, the original workpiece coordinate system is set again

8.2 Other operations in zero return

1. Press 

key, the spindle rotates counterclockwise;
2. Press 

key, the spindle stops;
3. Press 

key, the spindle rotates clockwise;
4. Press 

key, the cooling is switched ON or OFF;
5. Lubricating control(refer to OPERATION, Chapter 3);
6. Press 

key, the tool change is executed;
7. Tune the spindle override;
8. Tune the rapid override;
9. Tune the feedrate override.

CHAPTER IX DATA SETTING, BACKUP and RESTORE

9.1 Data setting

9.1.1 Switch setting

In SWITCH SETTING window, the ON-OFF state of PARM SWT (parameter switch), PROG SWT (program switch), AUTO SEG (auto sequence No.) can be displayed and set, as is shown in following figure:

1. Press SETTING key to enter the Setting interface, then press or key to enter the SWITCH SETTINGwindow;

2. Press or key to move the cursor to the item to beset;

3. Press U_W and R_V key to shift the ON-OFF state: press U_W key, "*" moves to the left toset the switch for OFF, press R_V key, "*" moves to the right to set the switch for ON. Only the PARM SWT is set for ON, could the parameter be altered; so are PROG SWT and AUTO SEG.

Note: When the PARM SWT is shifted from "OFF" to "ON", an alarm will be issued by CNC system. By pressing the CANCEL, RESET key together, the alarm can be cancelled. If the PARM SWT is shifted again, no alarm is issued. For security it should set the PARM SWT for "OFF" after the parameter alteration is finished.

9.1.2 Graphic display

Press SETTING key to enter the path window

Graphic parameter meaning

Coordinate system setting: 6 types of graphic paths can be displayed in this TAC2000 CNC system depending on the front or rear tool post coordinate system

A: Graphic scaling up and down

In Graphic window, the graphic path can be scaled up and down by the keys I A , M [in the edit keypad.

B: The START, STOP and CLEAR of the graphic path

In Graphic window, press the S] key once, it starts the drawing; press the T Y key once, it stops drawing; press J B key once, it clears the current graphic path.

C: Move of graphic path

In Graphic window, it may press the direction keys to move the graphic path

9.1.3 Parameter setting

By the parameter setting, the characteristics of the driver and machine can be adjusted. See Appendix 1 for their significance.

Press PARAVETER key to enter the Parameter interface, then press or key to window the parameter interface, as is shown in the following figure:

A. Alteration of the bit parameter

1. Byte alteration

1) Turn on the parameter switch;
2) Enter the MDI mode;
3) Move the cursor to the parameter No. to be set:

Method 1: Press or key to enter the window containing the parameter to be set, press or key to move the cursor to the No. of the parameter to be set;

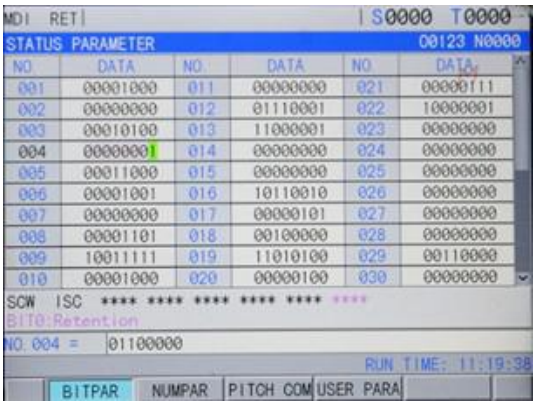
Method 2: Press address key P Q , key in parameter No., then press DATA INPUT key.

4) Key in the new parameter value;
5) Press DATA INPUT key, the parameter value is entered anddisplayed.
6) For security, the PARM SWT needs to be set for OFF after all parameter settings are finished.

Example:

Set the bit parameter No.004 Bit (DECI) to 1, and the other bits unchanged.

Move the cursor to No.004, input 01100000 by sequence in the prompt row, the display is as follows:



Press DATA INPUT key to finish the parameter alteration. The window is shown as follows:

2. Alteration by bit:

1) Turn on the parameter switch;
2) Enter the MDI mode;
3) Move the cursor to the No. of the parameter to be set;

Method 1: Press or key to enter the window of the parameter to be set, press or key to move the cursor to the No. of the parameter to be set;

Method 2: Press address key P, key in parameter No., then press DATA INPUT key.

4) Press CHANGE key to skip to a bit of the parameter, and the bit is backlighted. Press ⇨ or ⇦ key to move the cursor to the bit to be altered, then key in 0 or 1;

5) After all parameters setting is finished, the PARM SWT needs to be set for OFF for security.

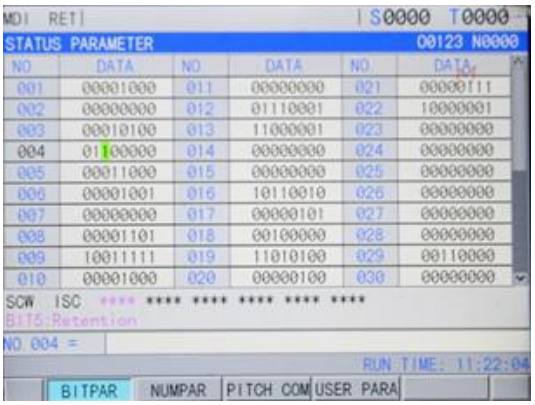
Note: After entering a bit of the parameter, press CHANGE key, it may skip out of the bit and back to the parameter No..

Example:

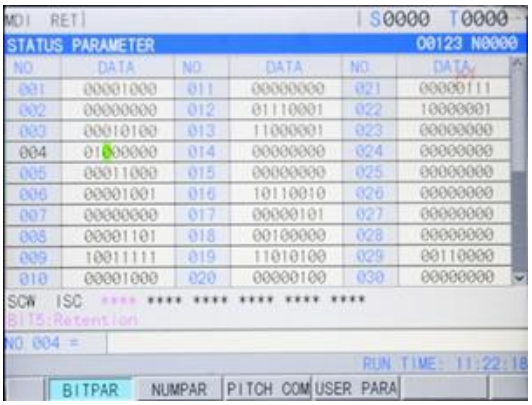
Set the BIT5 of the bit parameter No.004 to 1, and the other bits unchanged.

Move the cursor to "No.004" by the steps above, press CHANGE key to skip to a bit of the parameter as follows:

Move the cursor to “BIT5”by pressing ⇨ or ⇦ key as follows:



Input “0” to finish the alteration.

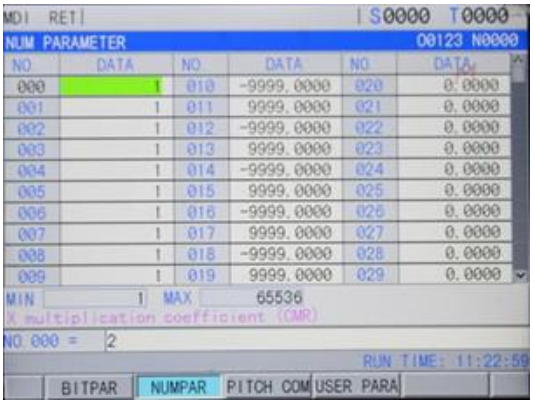


B Altering data parameter and screw-pitch parameter

1) Turn on the parameter switch;
2) Enter the MDI mode;
3) Move the cursor to the No. of the parameter to be set;
4) Key in the new parameter value;
5) Press DATA INPUT key, the value is entered and displayed;
6) After all parameters setting is finished, the PARM SWT needs to be set for OFF for security.

Explanation: The screw-pitch parameter can only be altered under the 2 level password authority.

Example 1: set the data parameter No.000 to 2.
Move the cursor to “No.000” by the steps above, key in “2” by sequence in the cue line as follows:



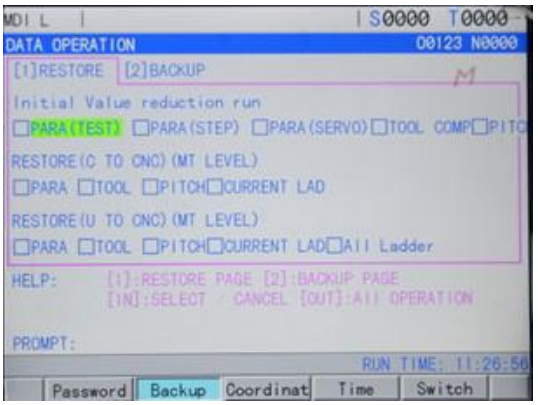
Press DATA INPUT key to finish the alteration. The window is shown as follows:

Example 2: set X value of the screw-pitch parameter No.000 to 12, Z axis value of that to 30. Move the cursor to screw-pitch parameter No.000 by the steps above, key in “X12” by sequence in the cue line.Press DATA INPUT key to finish the alteration.

The same as above, key in “Z30” by sequence in the cue line, press DATA INPUT key to finish the alteration. The window is as follows:

9.2 Data recovery and backup

The user data (such as bit parameter, data parameter, and screw-pitch parameter) can be backup (saved) and restored (read) in this TAC2000 system. It doesn’t affect the part programs stored in the CNC system while backuping and restoring these data. The backup window is shown as follows:



1. Turn on the parameter switch;

2. Press MDI key to enter the MDI mode, then press SETTING key (or key if necessary) to enter OPERATE DATAwindow;

3. Press 2 key to enter Backup DATAWindow,press 1 to enter RECOVERY DATA Window

4. Move cursor to operated option, press DATA OUTPUT key to select / cancle operation option

Note 1: Don't cut off the power in the backup and restore operation of the data, and no other operation is suggested to be performed before the operation is prompted to be finished.

Note 2: The user above the 3 password level can perform the backup and restore operation of the bit parameter, data parameter and the screw-pitch parameter.

9.3 Password setting and alteration

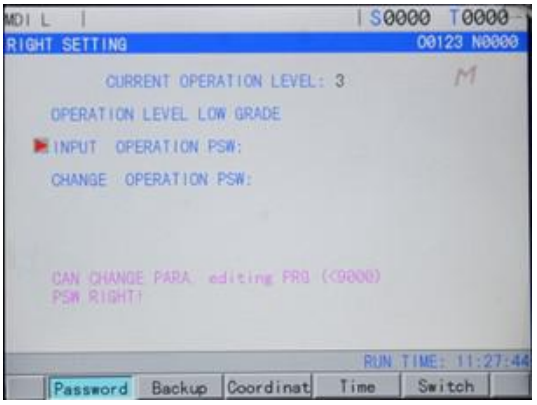
To protect the part programs, CNC parameters from malignant alteration, this TAC2000 provides password setting function that is graded for 4 levels. By descending sequence,
they are machine builder (2) level, equipment management (3) level, technician (4) level, machining operation (5) level. The current password level of the CNC system is displayed for "CURRENT LEVEL:_" in the PASSWORD SETTING window.

2 level: the CNC bit parameter, data parameter, screw-pitch parameter, tool offset data, part program edit, PLC ladder transmission etc. are allowed;

3 level: the initial password is 12345, the CNC bit parameter, data parameter, tool offset data, part program edit operations areallowed;

4 level: the initial password is 1234, tool offset data (for tool setting), macro variables, part program edit operations are allowed; but the CNC bit parameter, data parameter, screw-pitch parameter operations are forbidden.

5 level: no password. Only the machine panel operation is allowed, and the operations of part program edit and selection, the alteration operations of CNC bit parameter, data parameter, screw-pitch parameter, tool offset data are forbidden.



After entering the PASSWORD SETTING window, the cursor locates at the "INPUT PASSWORD:" row. It may press the ⇧ or ⇩ key to move the cursor to the corresponding item.

a) Press ⇧ key once, the cursor shifts a row upward. If the current cursor locates at the

"SET LOWER LEVEL" row (1st row), press ⇧ key, the cursor shifts to the "ALTER PASSWORD:" row(endrow);

b) Press ⇩ key once, the cursor shifts a row downward. If the current cursor locates at the end row, by pressing ⇩ key once, the cursor shifts to the 1st row.

9.3.1 Operation level entry

1 Move the cursor to the "INPUT PASSWORD:" row after the system enters the PASSWORD SETTING window;

2 Input the password (an "*" sign added each time inputting a character);

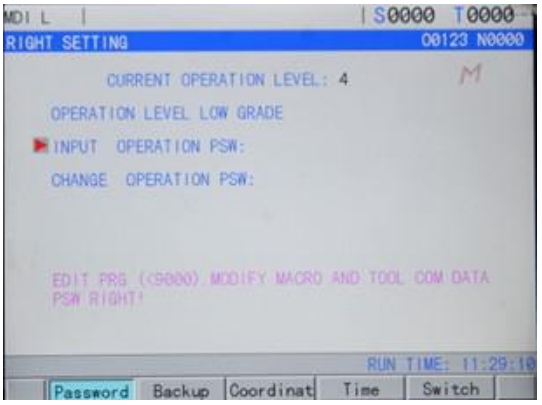
3 Press DATA INPUT key to finish the inputting, and the system enters the corresponding password level.

Note: The length of TAC2000 password corresponds to the operation level, which can't be or reduced by user at will. The detailed is as follows:

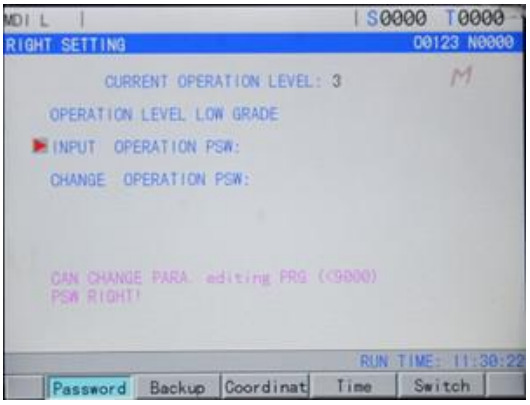
Operation level	Password length	Initial password
3	5 bytes	12345
4	4 bytes	1234
5	No	No

Example:

The current CNC operation level is 4 level, as the following window shows, the 3 level password of CNC is 12345, please alter the current level to the 3 level.



Move the cursor to the "INPUT PASSWORD:" row, key in 12345, then press the DATA INPUT key, the CNC prompts "Modify parameter and edit program", "PASSWORD PASSED.", and the current level is the 3 level. The display is as follows:



Note: When current operation level is lower than or equal to the 3 level (3, 4, 5 level), this level is not changed if the CNC system is turned on again. If previous level is the 2 level, it defaults the 3 level when the system is turned on again.

9.3.2 Altering the password

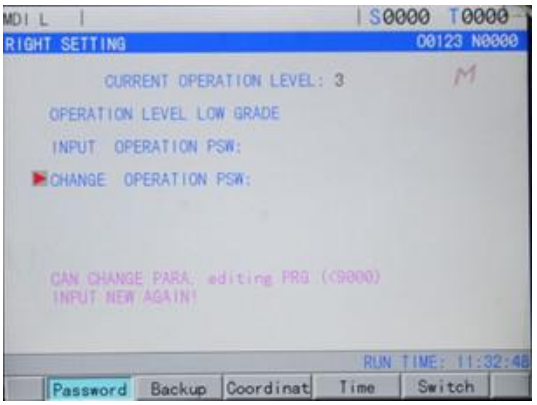
Steps for password alteration:

1 After entering the PASSWORD SETTING window, enter the password by the methods in Section10.3.1;

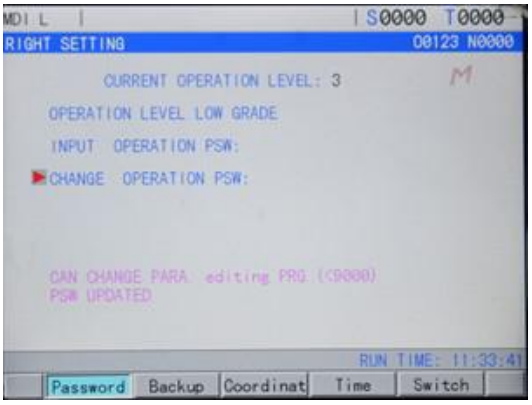
2 Move the cursor to the "ALTER PASSWORD:" row;

3 Key in the new password, then press DATA INPUT key;

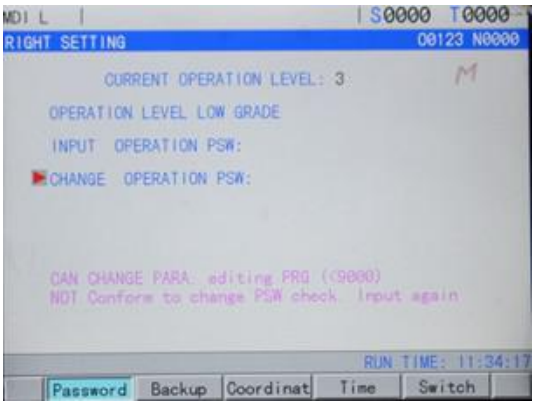
4 The CNC system prompts "PLEASE INPUT USER PASSWORD AGAIN!", the window display is as follows:



5 After re-inputting the password, press DATA INPUT key, if the passwords input are identical, CNC prompts "PASSWORD UPDATED.". So the password alteration is successful.



6 If the inputs of the passwords are not identical, CNC prompts “PASSWORD CHECKOUT ERROR.”, the window is as follows:

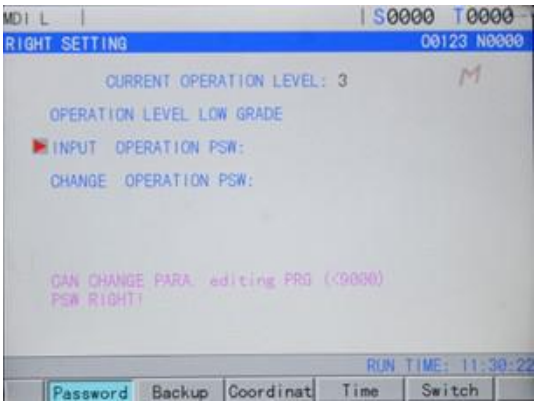


9.3.3 Setting the lower password level

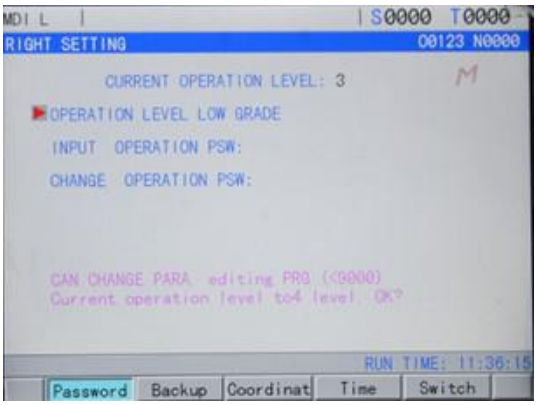
The demotion of the operation level is used to enter a lower level from a higher level, the steps are as follows:

1 After entering the PASSWORD SETTING window, key in the password by the method in Section 9.3.1;

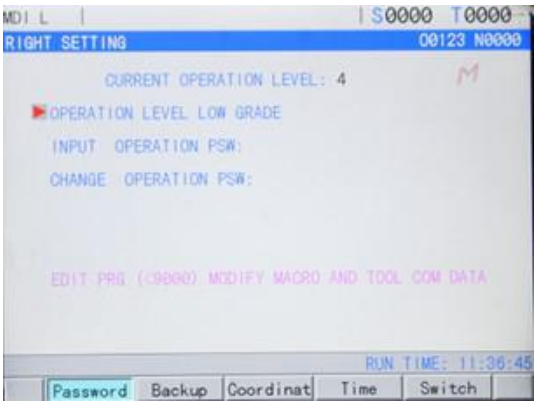
2 Move the cursor to the “SET LOWER LEVEL:” row, if the current CNC operation is the 3 level, the window is as follows:



3 Press DATA INPUT key, the CNC system prompts “CURRENT LEVEL TO 4, MAKE SURE? ”, the window is as follows:



4 Press DATA INPUT key again, if the demotion is successful, the window is as follows:



Note: If the current level is the 5 level, the demotion operation is forbidden.

CHAPTER x U DISK OPERATION FUNCTION

10.1 File catalog window

In non-edit mode, press PROGRAM to enter the program window, press to enter [File catalog] window, press CHANGE to identify it after U disk is inserted as follows:

The left displays CNC catalog information and the right displays USB disc catalog information. When the system has not checked the U disc, the right does not display the content. The bottom displays the file capacity and user operation prompt. The system only displays “.CNC”,”.NC” and “.txt” in the current file and other extension names are notdisplayed.

Press CHANGE and the cursor is switched from CNC to USB, press ⇧ or ⇩ to moveit.

10.2 File copy

Move the cursor the required CNC format file , press DATA OUTPUT to copy.

Copy file from U disk to system:Press CHANGE key to switch U disk catalog,and then press key to copy DATA OUTPUT

Volume III
Installation and Connection

CHAPTER 1 INSTALLATION

LAYOUT **1.1 TAC2000 system connection**

1.1.1 TAC2000 back cover interface layout

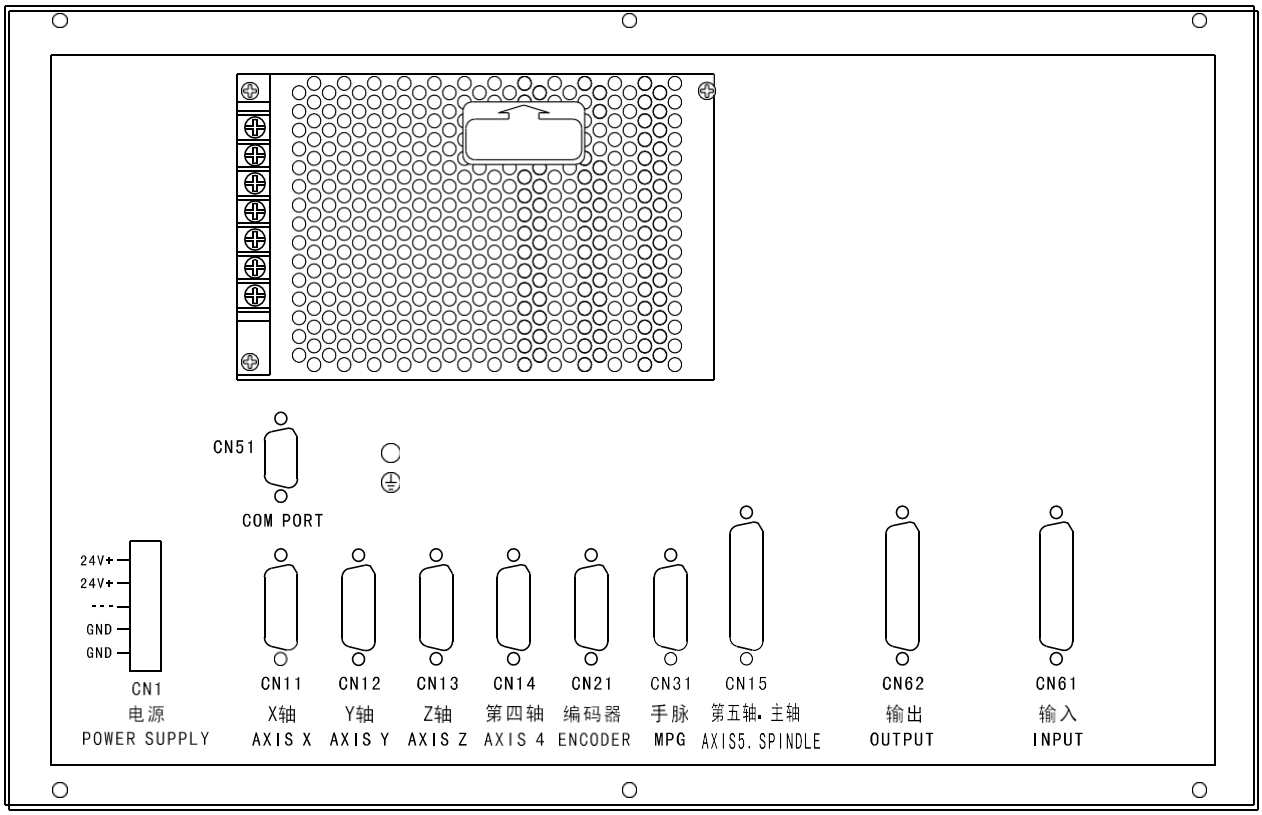


Fig. 1-1 TAC2000 back cover interface layout

1.1.2 Interface explanation

- Power box: for +24V, GND power supply
- Filter(optional): Input terminals for 220V AC power, PE terminal for grounding, output terminals to L, N terminals of KY-PB2 power box
- CN1：power supply interface
- CN11：X axis，pin15 D female, connect with X drive unit
- CN12：Y axis，pin15 D female, connect with Y drive unit
- CN13：Z axis，pin15 D female, connect with Z drive unit
- CN14：4th axis，pin15 D female, connect with 4th drive unit
- CN15：spindle, pin 25 D female, connect with spindle drive unit
- CN21：encoder, pin15 D male, connect with spindle encoder
- CN31：MPG, pin26 D male, connect with MPG
- CN51：communication, pin9 D female, connect PC RS232 interface
- CN61：input, pin44 D male, connect with machine input
- CN62：output, pin44 D female, connect with machine output

1.2 TAC2000 installation

1.2.1 TAC2000 external dimensions See

Appendix III,IV.

1.2.2 Preconditions of the cabinet installation

- The dust, cooling liquid and organic resolution should be effectively prevented from entering the cabinet;
- The designed distance between the CNC back cover and the cabinet should be not less than 20cm, the inside and outside temperature difference of the cabinet should be not more than 10℃ when the cabinet inside temperature rises;
- Fans can be fixed in the cabinet to ventilate it;
- The panel should be installed in a place where the cooling can't splash;
- The external electrical interference should be taken into consideration in cabinet design to prevent it from interfering the CNC system.

1.2.3 Measures against interference

In order to insure the CNC stable working, the anti-interference technology such as space electromagnetic radiation shielding, impact current absorbing, power mixed wave filtering are employed in CNC design. And the following measures are necessary during CNC connection:

1. Make CNC far from the interference devices (inverter, AC contactor, static generator, high-voltage generator and powered sectional devices etc.);
2. To supply the CNC via an isolation transformer, the machine with the CNC system should be grounded, the CNC and drive unit should be connected with independent grounding wires at the grounding point;
3. To inhibit interference: connect parallel RC circuit at both ends of AC winding (Fig. 1-3), RC circuit should approach to inductive loading as close as possible; reversely connect parallel freewheeling diode at both ends of DC winding (Fig. 1-4); connect parallel surge absorber at

the ends of AC motor winding (Fig. 1-5);

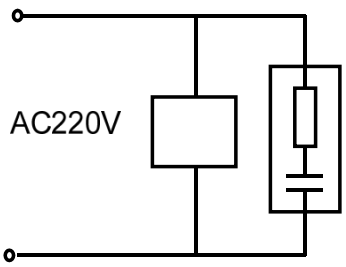


Fig. 1-3

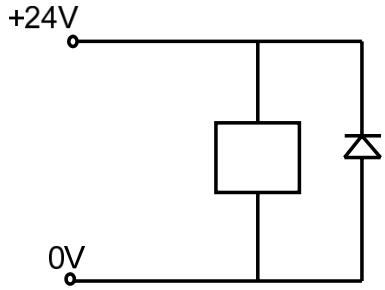


Fig. 1-3

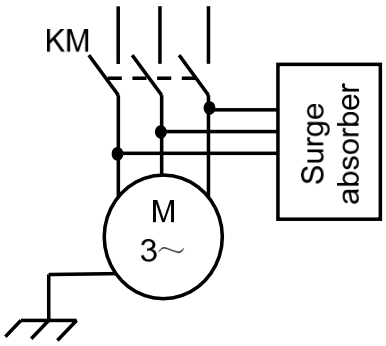


Fig. 1-3

4. The CNC leadout cables use the twisted shield cable or shield cable, the cable shield tier is grounded by an terminal at CNC side, signal cable should be as short as possible;
5. To reduce the mutual interference among the CNC signal cables, and among the strong current, the wiring should follow the following:

Table 1-1 The Wiring requirement

Group	Cable type	Wiring requirement
A	AC power cable	Tie up A group cables with a clearance at least 10cm from that of B, C groups, or shield A group cables from electromagnetism
	AC coil	
	AC contactor	
B	DC coil(24VDC)	Tie up B and A group cables separately or shield B group cables; and the further B group cables are from that of C group, the better it is
	DC relay(24VDC)	
	Cables between CNC and strong-power cabinet	
	Cables between CNC and machine	
C	Cables between CNC and servo drive unit	Tie up C and A group cables separately, or shield C group cables; and the cable distance between C group and B group is at least 10cm and they are twisted pair cables.
	Position feedback cable	
	Position encoder cable	
	Handwheel（MPG）cable	
	Other cables for shield	

Chapter 2 Interface Signal and Connection

2.1 Connection with the Drive Unit

2.1.1 Definition of the Drive Interface

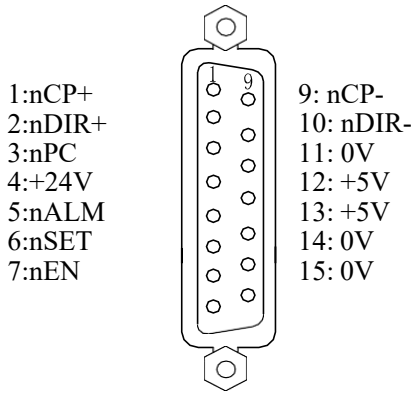


Fig. 2-1 Interfaces of CN11, CN12, CN13 and CN14 (female socket of 15-cord in D type)

SIGNAL	REMARK
nCP+, nCP-	Command pulse signal
nDIR+, nDIR-	Command direction signal
nPC	Zero signal
nALM	Drive unit alarm signal
nEN	Axis enable signal
nSET	Pulse forbid signal

2.1.2 Command Pulse Signal and Command Direction Signal

nCP+ and nCP- are command pulse signals, -nDIR+ and nDIR- are command direction signals, and the signals of two groups are differential output（AM26LS31）, and the external is suggested to use AM26LS32 for receiving, and the internal circuit is shown as the following figure 2-2:

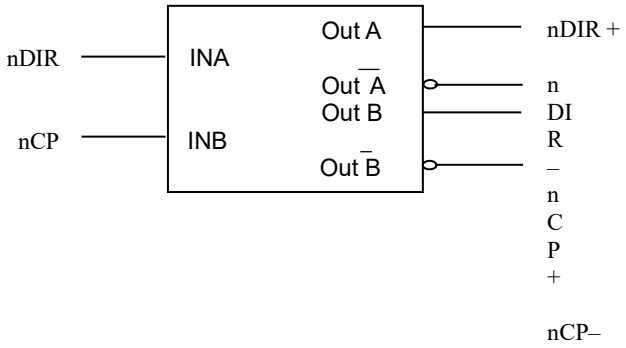


Fig. 2-2 The internal circuit of the command pulse signal and the command direction signal

2.1.3 Drive Unit Alarm Signal nALM

Whether the drive unit alarm level is low or high is set by Bit0, Bit1, Bit2, Bit3 and Bit4 of CNC parameter No.009. About the internal circuit, refer to figure 2-3:

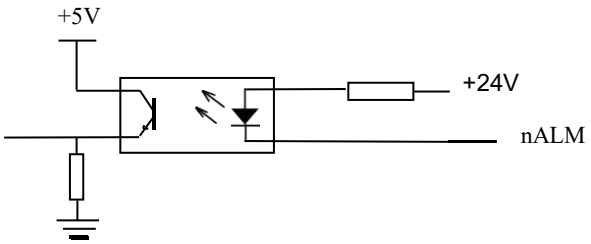


Fig. 2-3 The internal circuit of the drive unit alarm signal

Based on the input circuit of the type, the drive unit should be adopted the following methods (shown as fig.2-4) to provide the signals:

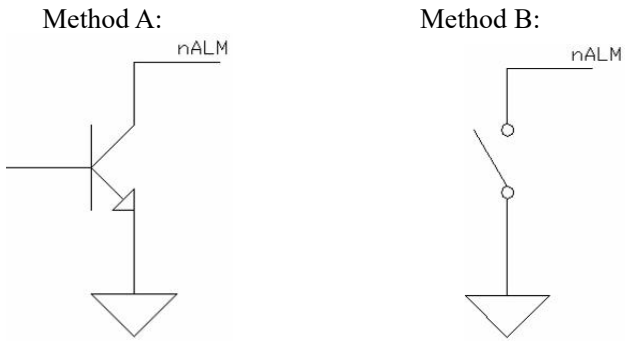


Fig. 2-4 Method of the drive unit providing signals

2.1.4 Axis Enable Signal nEN

When CNC works normally, nEN signal output is valid (nEN signal is connected with 0V); when the drive unit alarms, CNC switches off nEN signal output (nEN signal is disconnected with 0V). The internal interface circuit is shown as the following figure 2-5:

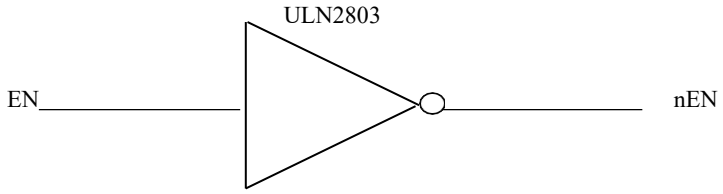


Fig. 2-5 Axis enable signal internal interface circuit

2.1.5 Pulse Forbid Signal nSET

The nSET signal is to control the servo input forbid, to improve anti-interference between CNC and the drive unit, and the signal is high impedance state when CNC outputs the pulse signal, and it is the low level when the pulse signal isn't output. The internal interface circuit is shown as the following figure 2-6:

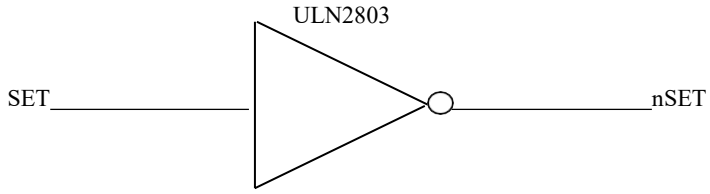


Fig. 2-6 Pulse forbid signal circuit

2.1.6 Zero Signal nPC

During the machine zero return, one-turn signal of the motor encoder or the proximity switch signal is taken as the zero signal. The internal connection circuit is shown as the following figure 2-7:

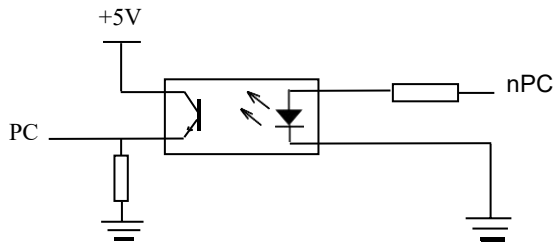


Fig. 2-7 Zero signal circuit

Note: The nPC signal is adopted +24V level.

a) The user should provide the wave of PC signal, which is shown as the following figure 2-8:

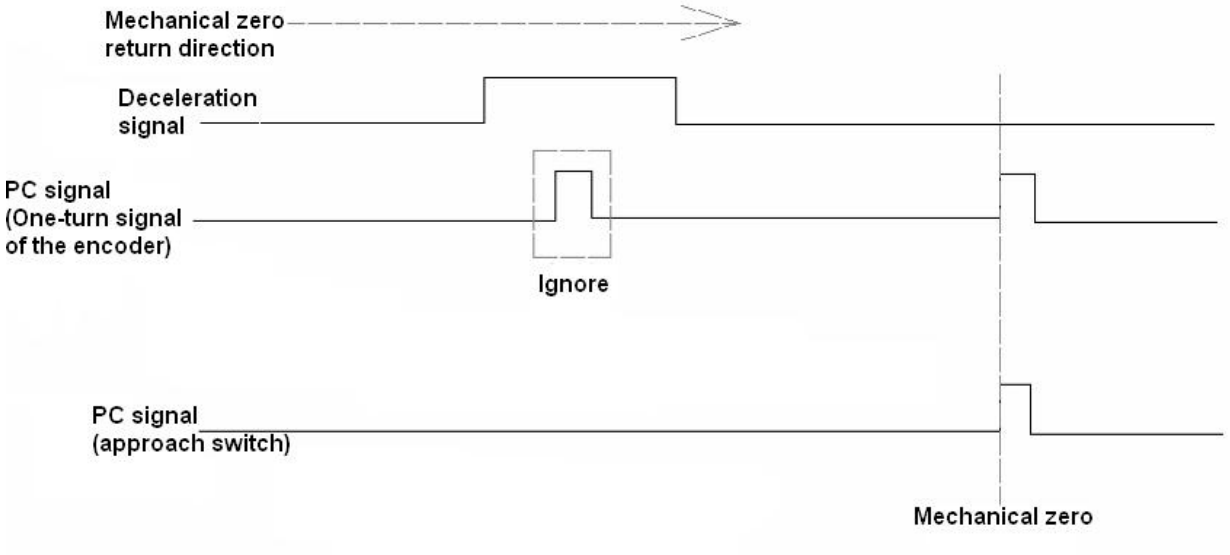


Fig. 2-8 PC signal waveform

Note: During the machine zero return, CNC can judge the location of the reference position through detecting PC signal after the deceleration switch is OFF, and the rising edge or the falling edge all are valid.

b) The wiring of NPN Hall element taken as both DEC signal and zero signal is shown in Fig. 2-9:

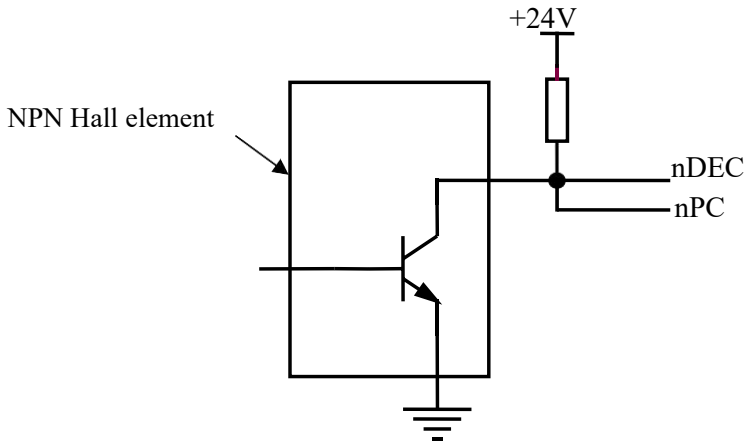


Fig. 2-9 Wiring by a NPN Hall element

c) The wiring of PNP Hall elements taken as both DEC signal and zero signal is shown in Fig. 2-10:

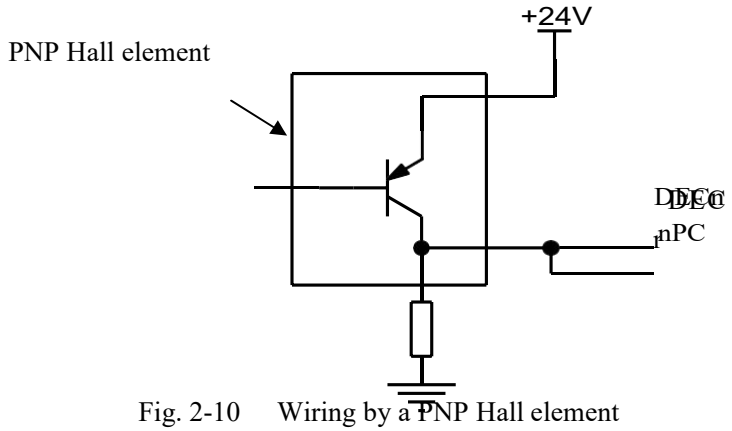
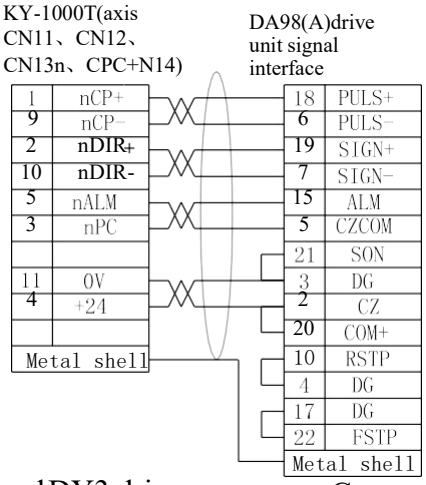


Fig. 2-10 Wiring by a PNP Hall element

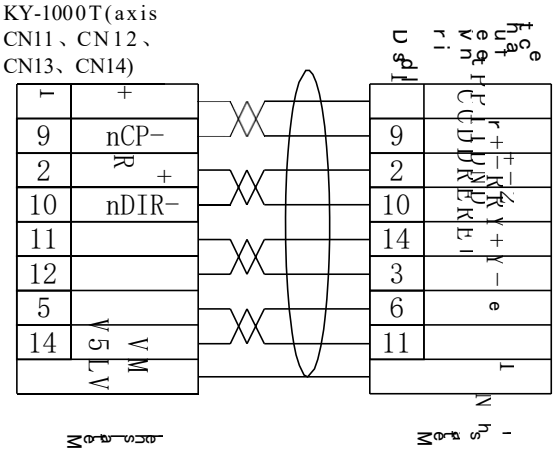
2.1.7 Connection to a drive unit

TAC2000 is connected with our drive unit shown in Fig. 2-11:

TAC2000 is connected with DA98(A)drive unit



Connection of TAC2000 and DY3 driver



Connection of TAC2000 and DF3 driver

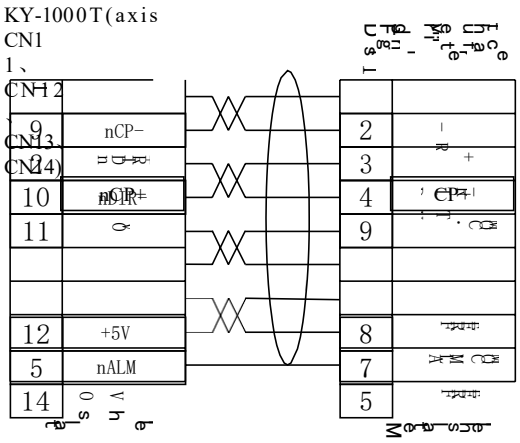


Fig. 2-11 Connection of TAC2000 and drive unit

2.2 Connection of the Spindle Encoder

2.2.1 Definition of the Spindle Encoder Interface

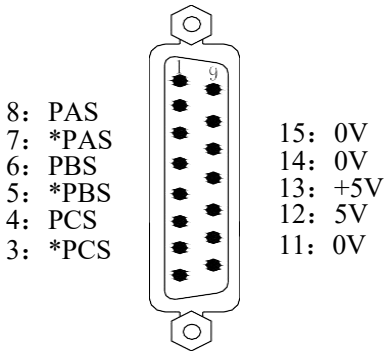


Fig. 2-12 CN21 encoder interface (male socket of 15-cord in D type)

NAME	REMARK
*PAS/PAS	Encoder phase A pulse
*PBS/PBS	Encoder phase B pulse
*PCS/PCS	Encoder phase C pulse

2.2.2 Signal Explanation

*PCS/PCS, *PBS/PBS and *PAS/PAS are respectively the differential input signals of the encoder phases C, B and A, and it adopts 26LS32 for receiving; *PAS/PAS and *PBS/PBS are orthogonal square waves of difference 90°, and the highest signal frequency is<1MHz; the encoder resolution is set by parameters (range 100～5000).

The internal connection diagram is shown as the following figure 2-13 (n in figure= A, B, C):

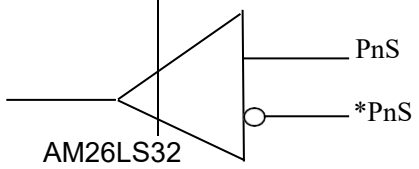


Fig. 2-13 Encoder signal circuit

2.2.3 Connection of the Spindle Encoder Interface

The connection of TAC2000 and the spindle encoder is shown as the following figure 2-14, and the connection is used with the twisted wire. (The encoder of ZLF-12-102.4BM-C05D from Changchun First Optical, Co., Ltd is taken as the example):

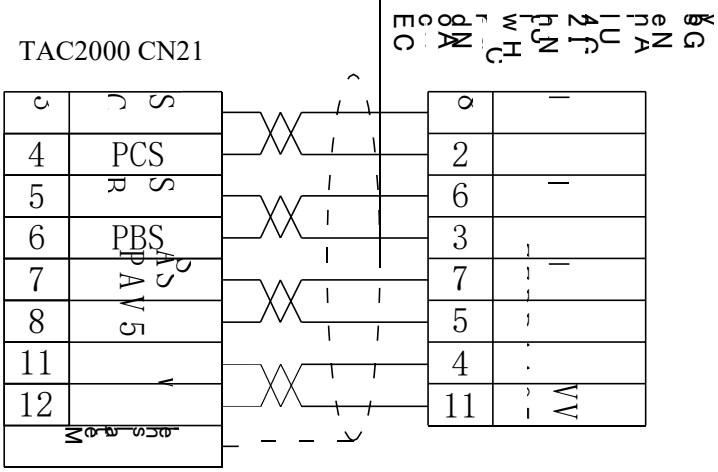


Fig. 2-14 Connection of TAC2000 and the encoder

2.3 Connection with MPG

2.3.1 MPG Interface Definition

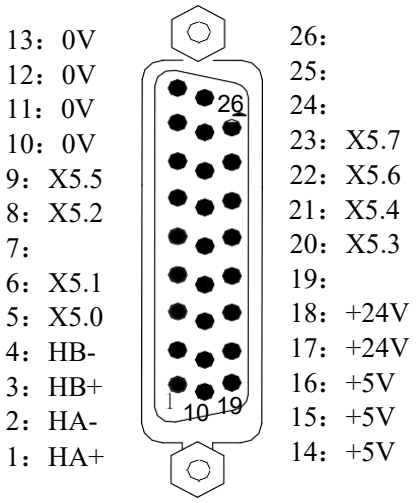


Fig. 2-15 CN31 MPG interface (male socket of 26-cord in type D)

Signal	Explanation
HA+、HA-	MPG A phase signal
HB+、HB-	MPG B phase signal
X5.0	X MPG axis selection
X5.1	Y MPG axis selection
X5.2	Z MPG axis selection
X5.3	4th MPG axis selection
X5.4	5th MPG axis selection
X5.5	increment×1
X5.6	increment×10
X5.7	increment×100
+24V	DC power supply
VCC、GND	

2.3.2 Signal Explanation

HA and HB are separately the input signals of MPG phases A and B. The internal connection diagram is shown as the following figure 2-16:

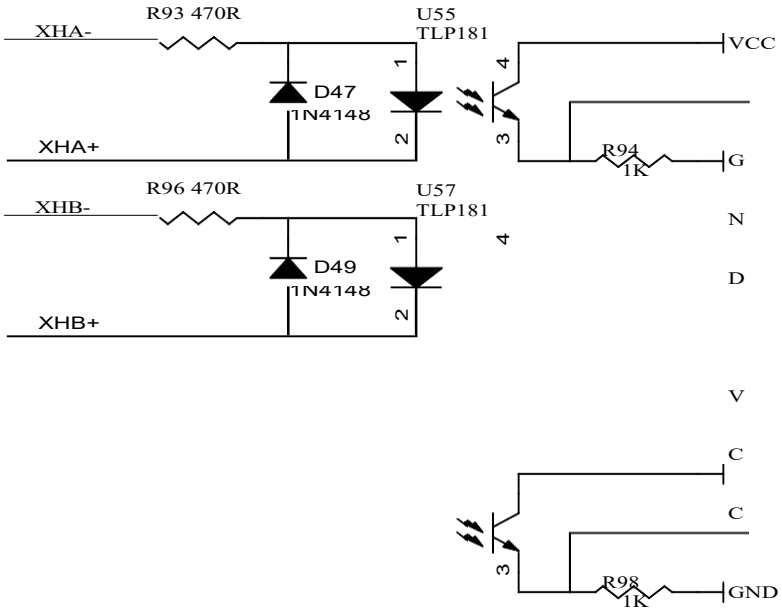
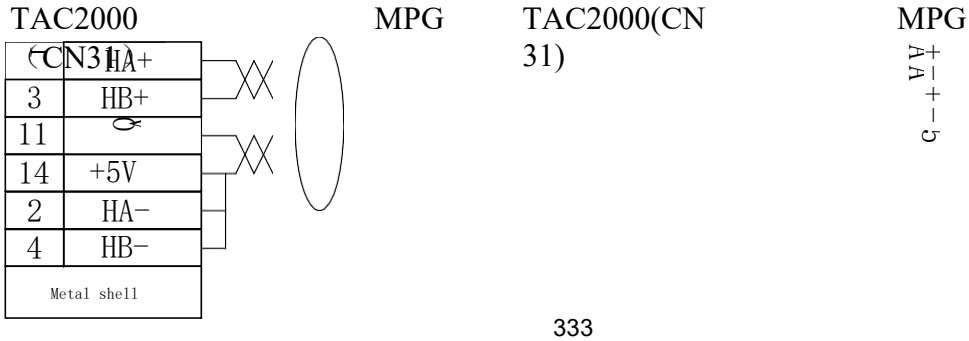
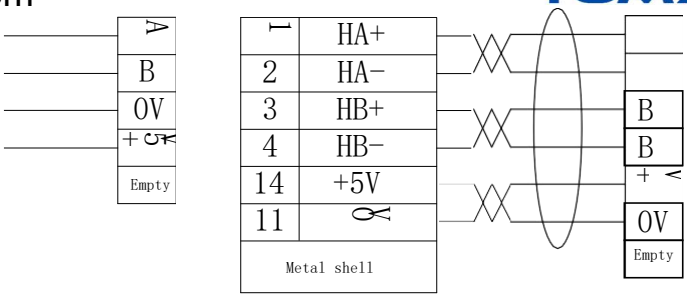


Fig. 2-16 MPG signal circuit

Connection of TAC2000 and MPG is shown as the following figure:2-17:





Single-terminal input differential input

Fig. 2-17 TAC2000 is connected with MPG

2.4 Spindle Interface

2.4.1 Spindle Interface Definition

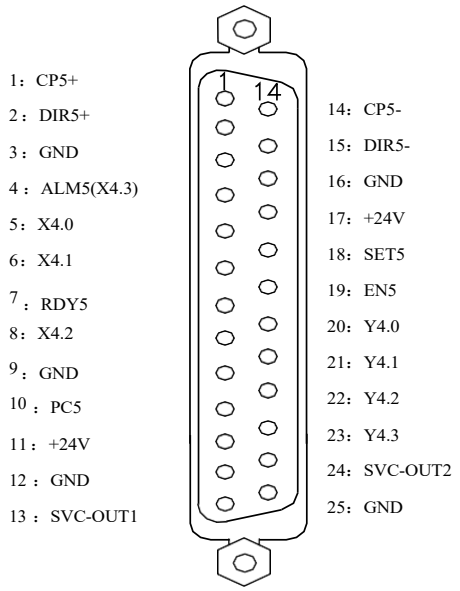


Fig. 2-18 CN15 spindle interface (female socket of 25-cord in type D)

CP5+,CP5-	Spindle pulse signal
DIR5+,DIR5-	Spindle direction signal
ALM5	The 5th axis/spindle abnormal alarm signal
RDY5	Spindle ready signal
PC5	Spindle zero signal
SVC-OUT1	Analog voltage output 1
SVC-OUT2	Analog voltage output 2
SET5	Spindle setting signal
EN5	Spindle enable signal
X4.0~X4.3	PLC address, only low level is valid
Y4.0~Y4.3	PLC address

3:GND
4:ALM5(X4.3)
5:X4.0
6:X4.1

Note 1: Conduction between PC5 and 0V is valid, and it is different with the other feeding axes（PC of CN11～CN14 axis interfaces is conductive with＋24V, and it is valid.）.

Note 3: The internal circuit of the signals of PC5 is shown as the following figure:

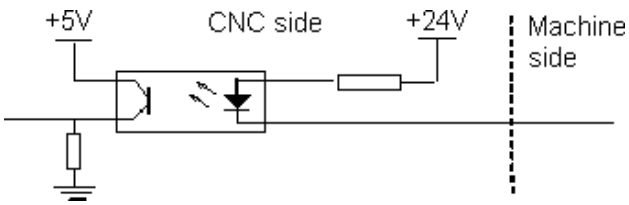


Fig. 2-19 Circuit of PC5

2.4.2 Common Transducer Connection

The analog spindle interface SVC port can output 0~10V voltage, the signal internal circuit is shown as the following figure 2-20:

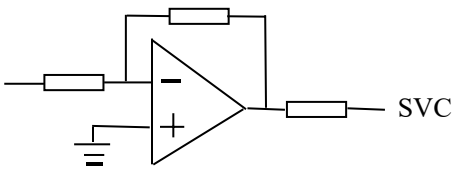


Fig. 2-20 SVC signal circuit

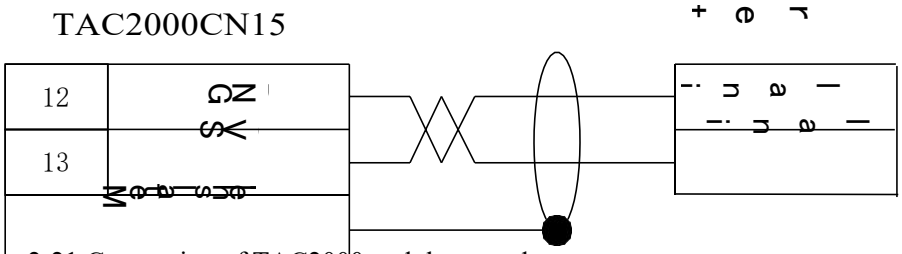
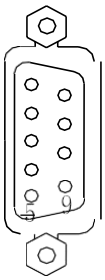


Fig. 2-21 Connection of TAC2000 and the transducer

2.5 Connection of TAC2000 and PC Serial Port

2.5.1 Communication Interface Definition

1:
2: RXD
3: TXD
4:
5: GND



6:485+
7:485-
8:
9:GND

SIGNAL	REMAKR
RXD	Receiving the data
TXD	Sending the data
GND	Signal ground
485+	Absolute value 485+
485-	Absolute value 485-

Fig.2-22 CN51communication interface（9 holes）

2.5.2 Communication Interface Connection

TAC2000 can communicate with CN51 interface and PC (the communication software is optional). TAC2000 and PC connection is shown as the following figure 2-23A:

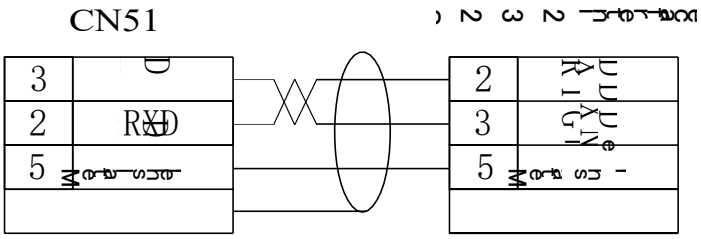


Fig. 2-23A TAC2000 is connected with PC

The communication between a TAC2000 system to another TAC2000 system can be done by CN51 shown in Fig. 2-23B:

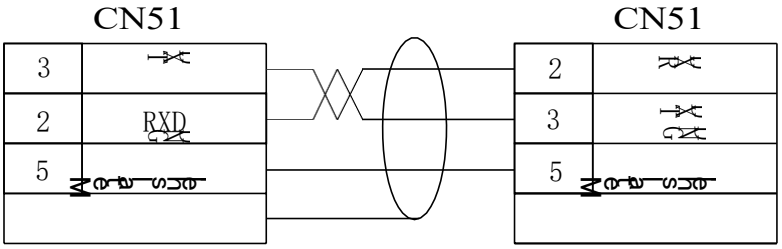


Fig. 2-23B Communication between a TAC2000 system and another TAC2000 system

2.6 Power interface connection

The power box interface has been done for its delivery from factory, and the user only need to connect it to a 220V AC power in using.

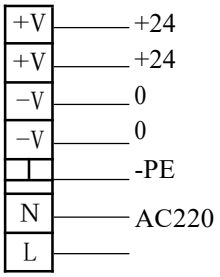


Fig. 2-24 system power interface CN1

2.7 I/O Interface Definition:

NOTE!

The meaning of the fixed address I/O function which doesn't mark in TAC2000 CNC system is defined by PLC program (ladder diagram). When TAC2000 CNC system is equipped with the machine tool , I/O function is set by the machine tool manufacturer; and about the details, please refer to the user manual from the machine tool manufacturer.

The fixed address I/O function which doesn't mark in the chapter is mainly for standard PLC program of TAC2000. Please pay special attention to that the described content also applies to TAC2000 system without especial explanation.

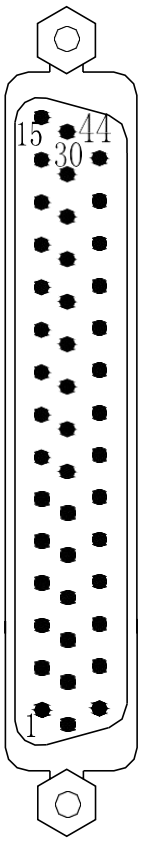


Fig. 2-25 CN61 input interface

PIN NO.	ADDRESS	FUNCTION	REMARK
21～24	0V	Power supply interface	Power supply 0V terminal
18～20 25～28	Suspend	Suspend	Suspend
1	X0.0	DECX	X axis deceleration signal
2	X0.1	SP1	External pause signal
3	X0.2	ST1	Y axis deceleration signal
4	X0.3	DECX(DEC1)	Z axis deceleration signal
5	X0.4	PRES1	4th axis deceleration signal
6	X0.5	ESP	Emergency stop signal
7	X0.6	ESP1	Over travel release input signal
8	X0.7	T11	Start working door switch X4
9	X1.0	T21	Tool location signal 2 of channel 1
10	X1.1	T31	Tool location signal 3 of channel 1
11	X1.2	T41	Tool location signal 4 of channel 1
12	X1.3	DECZ(DEC2)	Z axis deceleration signal
13	X1.4	TCP1	Tool post locked signal of channel 1
14	X1.5	SAGT1	Protection door detection signal of channel 1
15	X1.6	DIQP1	External chuck control signal of channel 1
16	X1.7	DITW1	External tailstock control signal of channel 1
29	X2.0	SAGT2	Protection door detection signal of channel 2
30	X2.1	SP2	Pause signal of channel 2
31	X2.2	ST2	Cycle start signal of channel 2
32	X2.3	DECY(DEC3)	Y axis deceleration signal
33	X2.4	DEC4	The 4th axis deceleration signal
34	X2.5	DEC5	The 5th axis deceleration signal
35	X2.6	ESP2	External emergency stop signal of channel 2
36	X2.7	T12	Tool location signal 1 of channel 2
37	X3.0	T22	Tool location signal 2 of channel 2
38	X3.1	T32	Tool location signal 3 of channel 2
39	X3.2	T42	Tool location signal 4 of channel 2
40	X3.3	TCP2	Tool post locked signal of channel 2
41	X3.4	PRES2	Pressure detection of channel 2
42	X3.5	SKIP	G31 skip signal
43	X3.6	DIQP2	External chuck control signal of channel 2
44	X3.7	DITW2	External tailstock control signal of channel 2

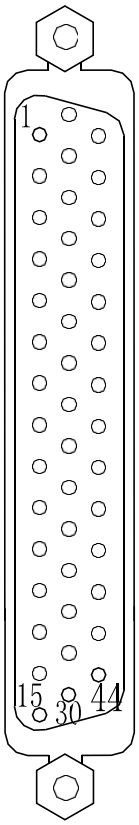


Fig. 2-26 CN62 output interface

PIN NO.	ADDRESS	FUNCTION	REMARK
17～19, 26～28	0V	Power supply interface	Power supply 0V terminal
20～25	+24V	Power supply interface	Power supply ＋24V terminal
1	Y0.0	COOL1	Channel cooling signal
2	Y0.1	LUBR1	Lubricating output signal of channel 1
3	Y0.2	SCLP	Spindle clamped output signal of channel 1
4	Y0.3	SFR1	Spindle CW rotation signal of channel 1
5	Y0.4	SRV1	Spindle CCW rotation signal of channel 1
6	Y0.5	DOTWJ1	Channel 1 tailstock advance
7	Y0.6	DOTWS1	Channel 1 tailstock retraction
8	Y0.7	SPZD1	Spindle brake signal of channel 1
9	Y1.0	S11/M411	Spindle mechanical gear signal 1 of channel 1
10	Y1.1	S21/M421	Spindle mechanical gear signal 2 of channel 1
11	Y1.2	S31/M431	Spindle mechanical gear signal 3 of channel 1
12	Y1.3	S41/M441	Spindle mechanical gear signal 4 of channel 1
13	Y1.4	DOQPJ1	Channel 1 chuck clamped
14	Y1.5	DOQPS1	Channel 1 chuck released
15	Y1.6	TL1+	Channel 1 tool post CW rotation
16	Y1.7	TL1-	Channel 1 tool post CCW rotation
29	Y2.0	COOL2	Channel 2 cooling signal
30	Y2.1	LUBR2	Channel 2 lubrication output signal
31	Y2.2	SVF	Channel 1 spindle servo OFF
32	Y2.3	SFR2	Channel 2 spindle CW rotation signal
33	Y2.4	SRV2	Channel 2 spindle CCW rotation signal
34	Y2.5	DOTWJ2	Channel 2 tailstock advance
35	Y2.6	DOTWS2	Channel 2 tailstock retraction
36	Y2.7	SPZD2	Channel 2 spindle brake signal
37	Y3.0	S12/M412	Channel 2 spindle mechanical gear signal 1
38	Y3.1	S22/M422	Channel 2 spindle mechanical gear signal 2
39	Y3.2	S32/M432	Channel 2 spindle mechanical gear signal 3
40	Y3.3	S42/M442	Channel 2 spindle mechanical gear signal 4
41	Y3.4	DOQPJ2	Channel 2 chuck clamped
42	Y3.5	DOQPS2	Channel 2 chuck released
43	Y3.6	TL2+	Channel 2 tool post CW rotation
44	Y3.7	TL2-	Channel 2 tool post CCW rotation

Note 1: Some input and output interfaces can define many functions, which is represented by "/" in the above list. Note 2: When the output function is valid, the output signal internal is conducted with 0V. When the output function is invalid, the output signal is high impedance cut off.

Note 3: When the input signal is conducted with +24V, the input is valid. When the input signal and +24V are cut off, the input is invalid.

Note 4: +24V and 0V and the terminals with the same name of the power supply box equipped by TAC2000 are equivalent.

2.7.1 Input Signals

The input signals are ones from the machine tool to the CNC, and the input is valid when the input signal is connected with +24V; the input is invalid when the input signal is disconnected with +24V. The input signals should satisfy the following conditions when it is on the machine side: Contact capacity: DC30V, above 16mA.

The leakage current among the contacts in the open circuit: below 1mA.

Voltage drop of the contacts in the access: below 2V (current 8.5mA, including the voltage drop of the cable).

The external input of the input signals has two methods: One is input with the contact switch, the signals are from the buttons and the limit switch from the machine side and the contacts of the relay, the connection is shown as the figure 2-27:

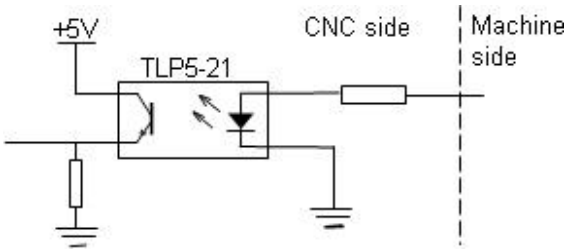


Fig. 2-27

The other is input without the contact switch (transistor), and the connection is shown as the figures 2-28A and 2-28B.

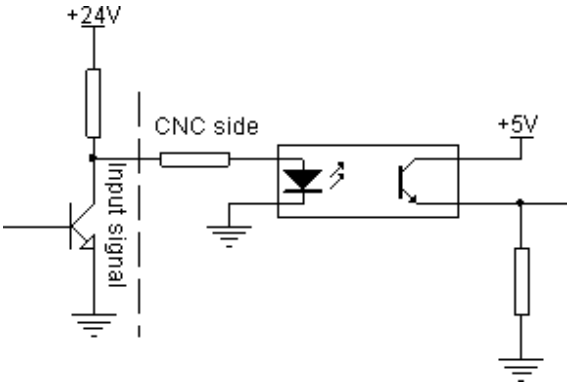


Fig. 2-28A Connection of NPN type

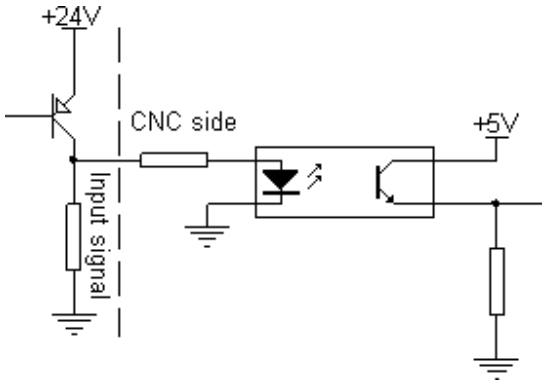


Fig. 2-28B Connection of PNP type

The input interfaces of TAC2000 standard PLC definition function includes the signals XDEC,YDEC,ZDEC,KYP,ST,SP and PRKY etc.

2.7.2 Output Signals

The output signals are used for driving the relay and the indicator on the machine side; when the output signals are connected with 0V, the output function is valid; when 0V is cut off, the output function is invalid. The digital of totally 36 routes is output in I/O interface, and they are all with the same structure, which is shown as figure 2-29:

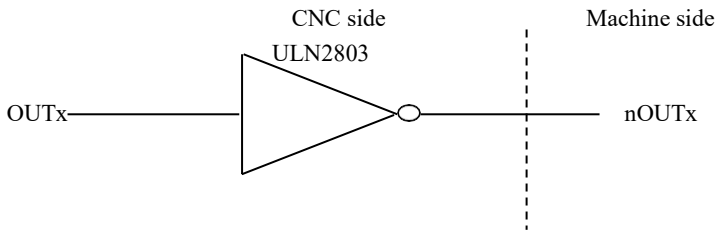


Fig. 2-29 Figure of digital output module circuit structure

The logic signal OUTx is output by the main board via the connecter and sent into the input port of the phase inverter (ULN2803), nOUTx has two output status: 0V output or high resistance. The typical application is shown as below:

- Drive LED

Output the drive LED with ULN2803, and one resistance should be serial connected to limit the current via LED (normally it is 10Ma), which is shown as the following figure 2-30:

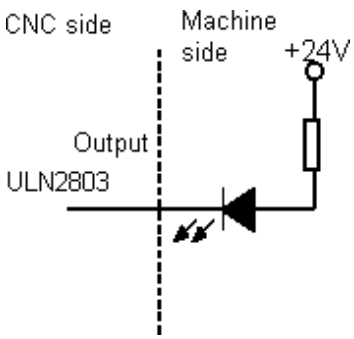


Fig. 2-30

- Drive indicator in lamp filament type

  Output the drive indicator in lamp filament type with ULN2803, one preheat resistance should be connected externally to reduce the electric shock during conducting, and the preheat resistance value should NOT make the indicator lamp ON, which is shown as the figure 2-31.

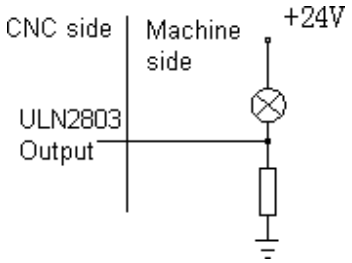


Fig. 2-31

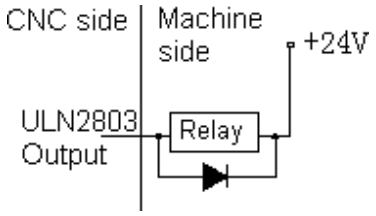


Fig.2-32

- Drive inductive load (such as the relay)

Output the drive inductive load of ULN2803 type, connect the fly-wheel diode around the coil to protect the output circuit and reduce the interference, which is shown as the above figure 2-32.

The meaning of the signals output from I/O interface is defined by PLC program, and the output signals defined by standard PLC program include S1~S4(M41~M44), M3~M5, M8, M32 etc.

2.8 I/O Function and Connection

NOTE!

The I/O function meaning of TAC2000 CNC system is defined by PLC program (ladder diagram). When TAC2000 CNC system is equipped with the machine tool , I/O function is set by the machine tool manufacturer, and about the details, please refer to the user manual from the machine tool manufacturer.

The I/O function in the chapter is mainly for standard PLC program of TAC2000. Please pay special attention to that the described content also applies to TAC2000 system without especial explanation.

2.8.1 Emergency Stop and Stroke limit

- **Relative signals**

KYP: emergency stop signal, alarm issued if the system is not connected with +24V

LMIX：X overtravel limit check input

LMIY：Y overtravel limit check input

LMIZ：Z overtravel limit check input

- **Diagnosis data**

0	0	0
Interface pin		

ESP							
CN61.6							

- **Signal diagnosis**

Signal	KYP	LMIX	LMIY	LMIZ
Diagnosis address	X0.5	X3.0	X3.1	X3.2
Interface pin	CN61.6	CN61.37	CN61.38	CN61.39

- **Control parameter**

Bit parameter

0	2	1

					KYP		

KYP =0: Check KYP signal

=1: Do not check KYP signal

- **PLC bit parameter**

K	1	0

LMIT	LMIS						

LMIT = 1: Travel limit check function of each axis is valid.

=0: Travel limit check function of each axis is invalid

LMI = 1: The system alarms for overtravel when the travel limit check signal is not connected with +24V.

=0: The system alarms for overtravel when the travel limit check signal is connected with +24V

- **Signal connection**

The circuit of the emergency stop signal（KSP）is shown as the following figure 2-33:

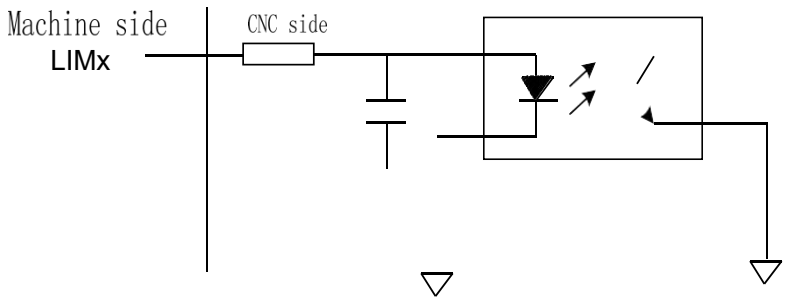


Fig. 2-33

- **Machine external connection**

① The limit switch is serial connected with the system emergency stop, the connection method is shown as the following figure 2-34A:

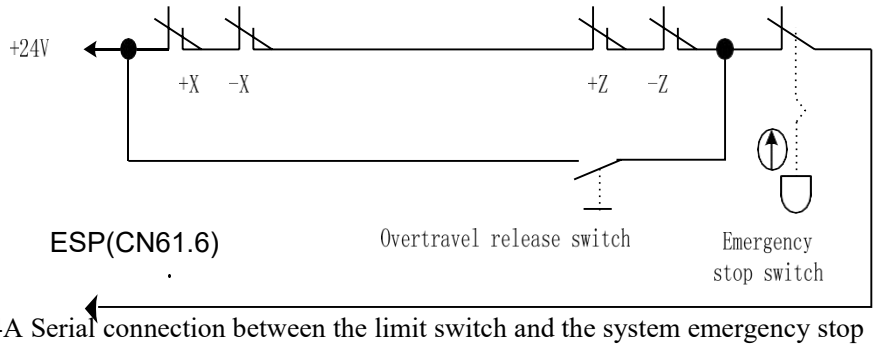


Fig. 2-34A Serial connection between the limit switch and the system emergency stop

② The limit switch is independently connected with the external emergency stop of each channel, and the connection method is shown as the following figure 2-34B:

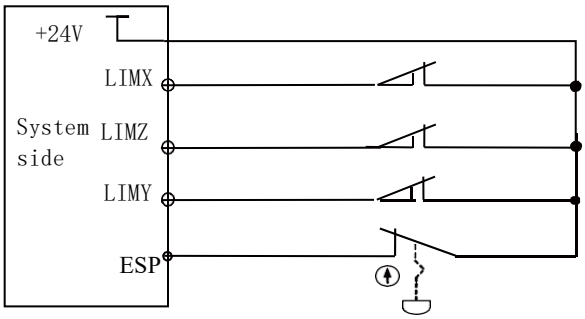


Fig. 2-34B Serial connection between the limit switch and the external emergency stop of each channel

- **Control logic**

1. The limit switch is serial connected with the system emergency stop

When overtravel occurs or the emergency stop button is pressed, “emergency stop” alarm occurs in the CNC system. If the overtravel occurs, press the overtravel release button without releasing, and press resetting key to cancel the alarm, and it moves in the opposite direction, then the overtravel alarm can be released. When the emergency stop alarm occurs, CNC stops pulse output. Except for the above mentioned CNC function, the other functions can be defined by PLC program when the emergency stop alarm occurs.

2. The limit switch is independently connected with the external emergency stop

1) There is only 1 overtravel contact.Whether positive or negative overtravel alarm is determined by movement direction of axis

2).If the overtravel alarm occurs, it moves in the opposite direction, then the overtravel alarm can be released by pressing reset key

2.8.2 Machine zero return

- Relative signal

  DECX：X deceleration signal；

  DECY：Y deceleration signal；

  DECZ：Z deceleration signal；

  DEC4：4th deceleration signal；

  DEC5：5th deceleration signal；

- Diagnosis data

0	0	0
Interface pin		

			DEC5	DEC4	DECZ	DECY	DECX
			CN61.34	CN61.33	CN61.12	CN61.32	CN61.4

- Control parameter

K	2	2

DEC4T	DECY	DECZ	DECX				

DEC4T=0: 4THdeceleratesasDECsignalisLOWlevel 4THdeceleratesasDECsignalisHIGHlevel

=1:

DECY=0: YdeceleratesasDECsignalisLOWlevel

=1: YdeceleratesasDECsignalisHIGHlevel

DECZ=0: ZdeceleratesasDECsignalisLOWlevel

=1: ZdeceleratesasDECsignalisHIGHlevel

DECX=0: XdeceleratesasDECsignalisLOWlevel

=1: XdeceleratesasDECsignalisHIGHlevel

0	0	6

				ZPLS			ZMOD

ZMOD =1: machine zero return block before

=0: machine zero return block after

ZPLS =1: machine zero return mode selection,have one-urn single

=0: machine zero return mode selection,have not one-turn single

0	1	2

							ISOT

ISOT =1: Manual rapid traverse active prior to machine zero return after power on

=0: Manual rapid traverse inactive prior to machine zero return after power on

0	2	6

			MZR5	MZR4	MZRY	MZRZ	MZRX

MZRx =1: The direction of machine zero return isnegative

=0: The direction of machine zero return is positive

● Data parameter

0	8	0	ZRNFL

ZRNFL =Low rate of axes reference return

0	7	0	ZRNFHX

ZRNFH =High-speed of X axes reference return

0	7	1	ZRNFHY

ZRNFY =High-speed of Y axes reference return

0	7	2	ZRNFHZ

ZRNFHZ =High-speed of Z axes reference return

0	7	3	ZRNFH4

ZRNFH4 =High-speed of 4TH axes reference return

0	7	4	ZRNFH5

ZRNFH5 =High-speed of 5TH axes reference return

● **Signal connection**

The interior wiring circuit of deceleration signal is shown as follows:

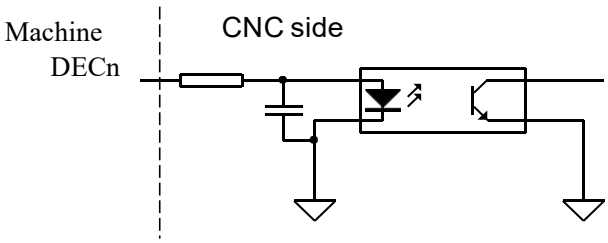
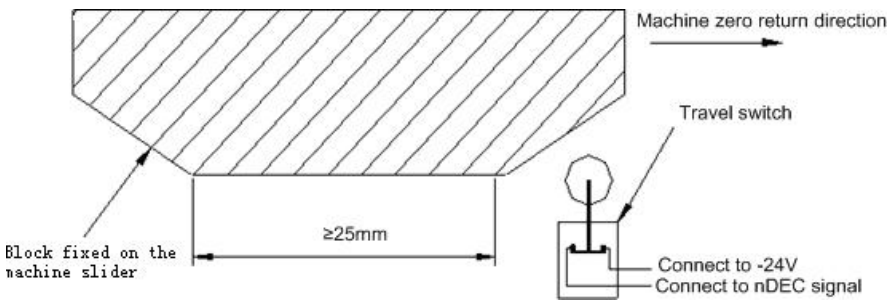


Fig.2-35

● **Machine zero return type B by regarding servo motor one-rotation signal as zero signal**

①Its sketch map is shown as follows:



②The circuit of deceleration signal

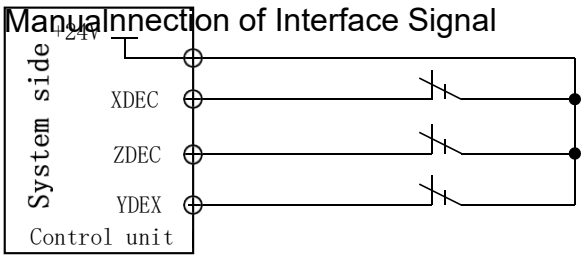
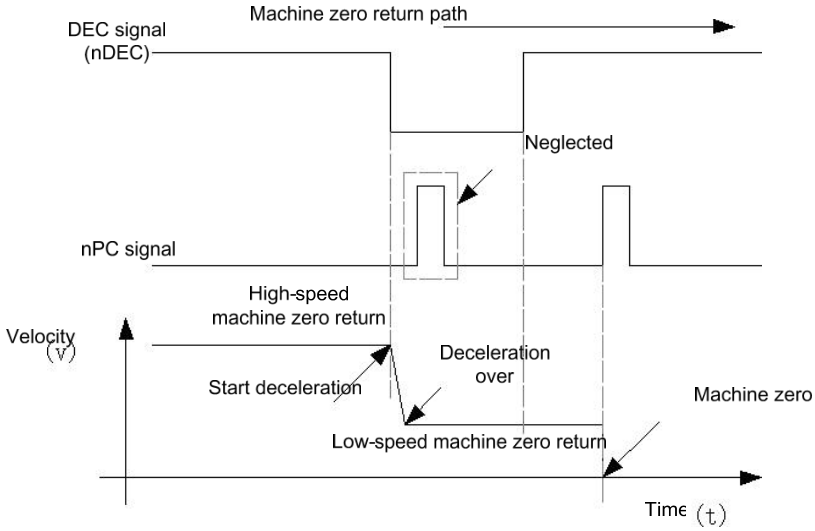


Fig. 2-40

③Sequence of machine zero return(Take X axis as an example)

When BIT4 of the K022 is set to 0, the deceleration signal low level is active. So the sequence of machine zero return is shown as follows:



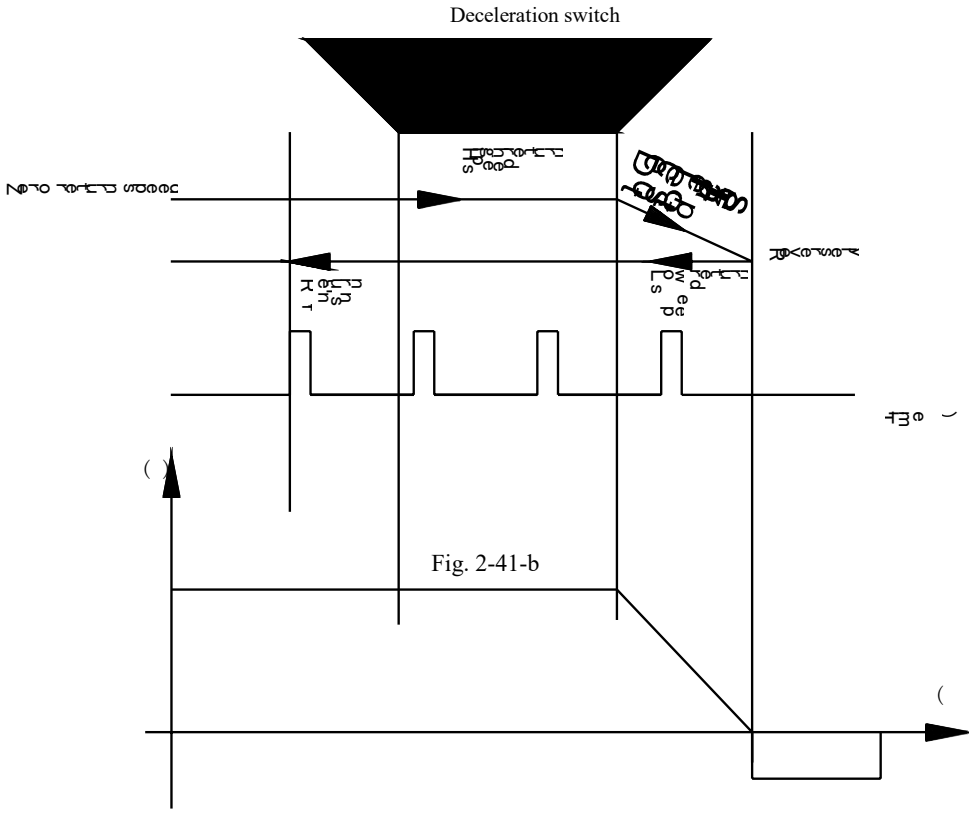
Return process of machine zero Fig. 2-41-a

A: Select Machine zero mode, press the manual positive or negative feed key(machine zero return direction set by bit parameter No.026), the corresponding axis moves to the machine zero by a rapid traverse speed(parameter No.70~No.74). As the axis press down the deceleration switch to cut off deceleration signal, the feeding slows down immediately, and it continues to run in a fixed low speed(parameter No.80).

B: When the deceleration switch is released, the deceleration signal contact is closed again. And CNC begins to detect the encoder one-turn signal (PC), if this signal level skips, the motion will be halted. And the corresponding zero return indicator on the operator panel lights up for machine zero return completion.

When the BIT1 (ZMOD) of the bit parameter No.006 is set to 1, and the BIT4 of the K022 is set to 0, it chooses the machine zero return block before, and the deceleration signal low level is active.

So the sequence of machine zero return block before is shown as follows:



Process of machine zero return block before

A: Select Machine zero mode, press the manual positive or negative feed key (return direction set by bit parameter No.026), the corresponding axis moves to the machine zero by a rapid traverse speed(parameter No.70~No.74). As the axis press down the deceleration switch to cut off deceleration signal, the feeding keeps rapid rate and depart from the deceleration switch, when the DEC signal contact is closed, the feeding slows down to zero, then run reversely to return to machine zero in a low speed.

B: In the reverse running, it presses down the deceleration switch to cut off the DEC signal contact and continues returning; as it departs from the deceleration switch, the deceleration signal contact is closed again. And CNC begins to detect the encoder one-turn signal (PC), if this signal level skips, the motion will be halted. And the corresponding axis zero return indicator on the operation panel lights up for zero return completion.

2.8.4 Spindle control

- **Relevant signal (by standard PLC program)**

Type	Symbol	Interface	Address	Function	Remark
Input signal	SAR	CN15.6	X4.1	Spindle speed arrival signal	It is valid when 0V is input
	SALM	CN15.4	X4.3	Spindle abnormity alarm input	
Output	M03	CN62.4	Y0.3	Spindle rotation(CW)	
	M04	CN62.5	Y0.4	Spindle rotation(CCW)	
	M05	CN62.6	Y0.5	Spindle stop	
	SCLP	CN62.7	Y0.6	Spindle clamped	
	SPZD	CN62.8	Y0.7	Spindle brake	
	SVF	CN62.37	Y3.0	Spindle servo OFF	

signal					
Command format	M03			Spindle rotation(CW)	
	M04			Spindle rotation(CCW)	
	M05			Spindle stop	
	M20			Spindle clamped	They are valid in analog spindle
	M21			Spindle released	

- **Control parameter**

Bit parameter

K	1	0							RSJG	

RSJG =1: CNC not turn off M03, M04, M08, M32 output signals when pressing  key;

=0: CNC turns off M03, M04, M08, M32 output signals when pressing  key.

0	2	0			SAR						

Bit6 1：The spindle SAR signal is checked before cutting; 0：The spindle SAR signal is not checked before cutting.

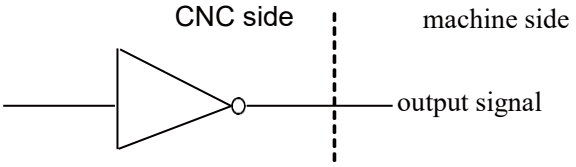
Data parameter

D	9	9	

Spindle zero speed output range(r/min)

- **Signal connection**

M03,M04,M05,SCLP,SPZD,SVF signal output cricurt is shown as 2-45A:



ULN2803

Fig. 2-45A

SAR、SALM signal input cricurt is shown as 2-45B:

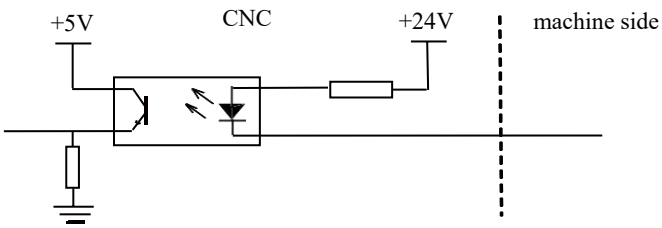


Fig. 2-45B

- **Movement time sequence (standard PLC program definition)**

The movement time sequence of the spindle is shown as the following figure 2-46:

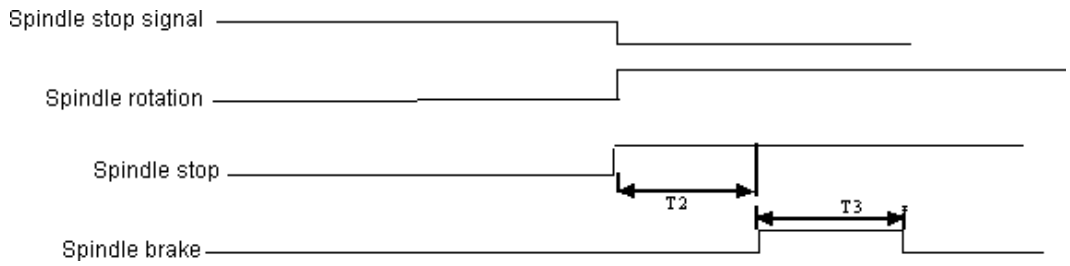


Fig. 2-46 Spindle CW and CCW rotation time sequence diagram

Note: T2 is the delay time from sending the signal of the spindle stop to sending the signal of the spindle brake; T3 is the spindle brake hold time.

- **Function description（defined by standard PLC program）**

① After the CNC is turned on, when M05 output is valid, M03 or M04 is executed, M03 or M04 output is valid and remains, at the time, M05 output is closed; when M03 or M04 output is valid, M05 is executed, M03 or M04 is closed, M05 output is valid and remains;

② When M03（M04）output is valid, M04（M03）is executed, the alarm occurs.

Note 1: In the emergency stop, it turns off M03, M04, M08 signals, and outputs M05 signal;
Note 2: Whether M03, M04 is cancelled is set by BIT3 of the bit parameter No.009 when CNC is reset.
If Bit 1=0, CNC turns off M03, M04 at reset; If
Bit 1=1, M03, M04 is kept at reset.

2.8.5 Spindle switching volume control

- **Relevant signal(defined by standard PLC program)**

S01～S04: The spindle speed switch value control signal, S01～S04 signal interfaces defined by the standard PLC program are the multiplex interfaces, S01～S04 and M41～M44 are the common interfaces.

- **Signal diagnosis**

Signal	S4	S3	S2	S1
Diagnosis address	Y1.3	Y1.2	Y1.1	Y1.0
Interface pin	CN62.12	CN62.11	CN62.10	CN62.09

- **Control parameters**

Bit parameter

0	0	1

			ACS				

Bit4 =1: Analog voltage control of spindle speed
=0: Switching volume control of spindle speed

- **Control logic (defined by standard PLC program)**

S1～S4 output are inactive at power on. If any code of them is executed, the corresponding S signal

output is active and held on, and the other S signal outputs are cancelled. S1～S4 outputs are cancelled when executing S00 code, and only one of them is active at a time.

2.8.6 Spindle automatic gearing control

- **Relevant signal (defined by standard PLC program)**

M41～M44： spindle automatic gear shifting output signals. It supports 4-gear spindle automatic gear shifting control when the system selects the spindle analog value control(0～ 10V analog voltage output)

M41I,M42I：spindle automatic gear shifting No.1, 2 gear in-position signals to support gear shifting in-position check function

- **Signal diagnosis**

Signal	M42I	M41I	M44	M43	M42	M41
Diagnosis address	X1.6	X1.5	Y1.3	Y1.2	Y1.1	Y1.0
Interface pin	CN61.15	CN61.14	CN62.12	CN62.11	CN62.10	CN62.09

- **Signal connection**

The circuit for M41~M44 is shown in Fig.2-47:

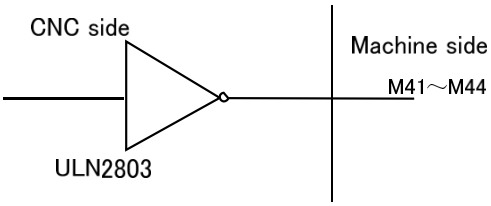


Fig. 2-47

- **Control parameter**

Bit parameter

0	0	1				ACS				

Bit4 =1: Spindle analog volume control, set to 1 if using spindle automatic gearing

=0: Spindle switching volume control

K	1	5					SHT	AGIM	AGIN	AGER

AGER =1: Spindle automatic gearing active

=0: Spindle automatic gearing inactive

AGIN =1: Detect M41I, M42I signal when shifting to gear 1, 2

=0: Not detect M41I, M42I signal when shifting to gear 1, 2

AGIM =1:Active when M41I, M42I signals disconnecting to +24V

=0: Active when M41I, M42I signals connecting to +24V

SHT =1：spindle gear power-down executes the memory

=0：spindle gear power-down does not execute the memory

Data parameter

2	1	0	GRMAX1
2	1	1	GRMAX2
2	1	2	GRMAX3
2	1	3	GRMAX4

GRMAX1,GRMAX2, GRMAX3, GRMAX4: The respective max. speeds of spindle gear 1, 2, 3, 4 when analog voltage output is 10V. Spindle speeds for M41, M42, M43, M44 when spindle automatic gearing is active.

2	1	4	SFTREV

Output voltage of spindle gearing (0~10000, unit: mV)

- **Function description (defined by standard PLC program)**

The spindle automatic gearing is active only under the spindle analog voltage control (BIT4 of the bit parameter No.001 set to 1) and the BIT0 of the K parameter No.15 is set to 1; if the spindle auto gearing is inactive, alarm will be issued when M41～M44 is being executed and only one of them is active at a time.

When spindle auto gearing is used to control automatic spindle mechanical gear switching, as CNC executes S□□□□ code, it calculates the analog voltage output to spindle servo or frequency inverter based on the parameter of the current gear by M4n (M41～M44 to data parameters No.210~No.213 respectively) to make the actual speed to be consistent with the S code.

When CNC is powered on, the spindle gear memorizing is set by the BIT3 of K parameter No.15.

If the BIT4 of bit parameter No.001 is 0, the spindle gear is not memorized at repowering after power down, and the gear 1 will be defaulted, M41~M44 are not output. If BIT4 of bit parameter No.001 is 1, the spindle gear is memorized at repowering after power down.

No gearing is done if the specified gear is consistent with the current gear. If not, gearing will be performed, and the process defined by standard ladders is shown in the following:

①Execute any of M41, M42, M43, M44 codes, output analog voltage to spindle servo or frequency inverter according to a value set by data parameter No.214 (Unit: mV);

②After a delay (gearing time 1) by the data parameter DT000, turn off the original gear output signal and output the new gearing signal;

③If the gear is 1 or 2, and the BIT1(AGIN) of the K parameter No.15 is 1, it jumps to ④, or else it jumps to ⑤;

④Check the gear in-position input signal M41I, M42I, it jumps to ⑤if the gear in-position is done; if not, the CNC waits the gear in-position signal;

⑤After a delay (gearing time 2) by the data parameter DT001, output spindle analog voltage by the current gear according to a value set by data parameter No.210~No.213 (gear 1~4) and finish the gearing.

Note: The output of M41～M44 is held on when CNC is reset or i emergency stop, which is defined by standard PLC ladder.

2.8.7 External cycle start and feed hold

- **Relevant signal (defined by standard PLC program)**

ST: External cycle start signal, whose function is the same with the CYCLE START key on the machine panel;

SP: External feed hold signal, whose function is the same with the FEED HOLD key on the machine panel

- **Signal diagnosis**

Signal	SP	ST
Diagnosis address	X0.1	X1.4
Interface pin	CN61.2	CN61.13

- **Signal connection**

The interior circuit of SP/ST signal is shown in Fig. 2-48:

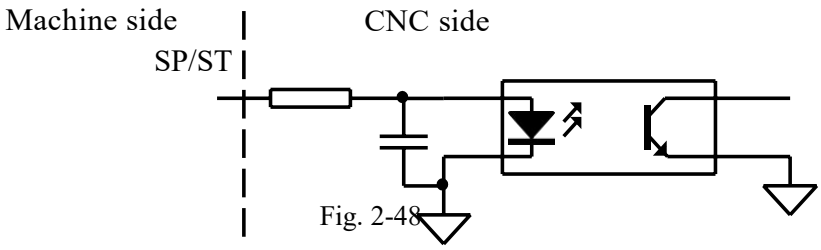


Fig. 2-48

- **Control parameter**

Bit parameter

0	2	1								MSP	MST

MST =1: External cycle start signal (ST) inactive

=0: External cycle start signal (ST) active

MSP =1: External feed hold signal (SP) inactive

=0: External feed hold (SP) active, the External feed hold switch is needed, or "feed hold" is displayed by CNC.

- **External connection circuit**

The external connection circuit of SP, ST signals is shown in Fig. 2-49:

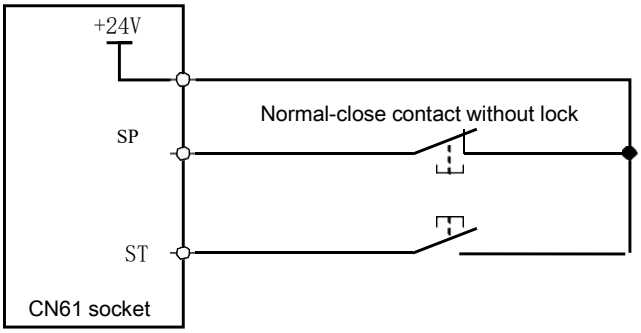


Fig. 2-49

2.8.8 Cooling Pump Control

- **Relevant signal (defined by standard PLC program)**

Type	Symbol	Interface	Address	Function	Remark
Output signal	M08	CN62.1	Y0.0	Cooling control output	
Command format	M08			Cooing ON	
	M09			Cooling OFF	

- **Signal connection**

  Its internal circuit is shown in Fig. 2-50:

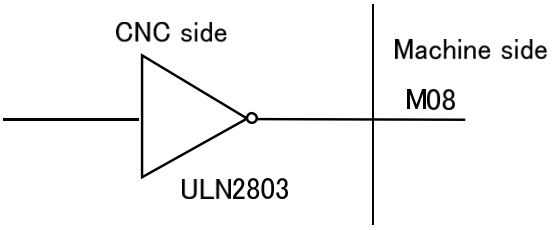


Fig. 2-50 M08 internal circuit

- **Function description (defined by standard PLC program)**

M09 is active, i.e. M08 is inactive, after CNC power on. To execute M08, M08 output is active and cooling is turned on; to execute M09, M08 output is cancelled and cooling is turned off.

Note 1: M08 output is cancelled at CNC emergency stop.

Note 2: Whether M08 is cancelled is set by BIT3 of the bit parameter No.009 when CNC is reset.

When Bit1=0, M08 output is cancelled as CNC is reset; When

Bit1=1, M08 output is not cancelled as CNC is reset;

Note 3: There is no corresponding output signal for M09, and M08 output is cancelled if M09 is executed. Note 4:

The cooling can be controlled by the 

key on operation panel, see details in OPERATION.

2.8.9 Lubrication Control

- **Relative command signals (standard PLC program definition)**

TYPE	CODE	INTERFACE	ADDRESS	FUNCTION	REMARK
Output signal	M32	CN62.2	Y0.1	lubrication control output	
Format	M32			Lubrication ON	
	M33			Lubrication OFF	

- **Signal connection**

  The internal circuit is shown as the following figure 2-51:

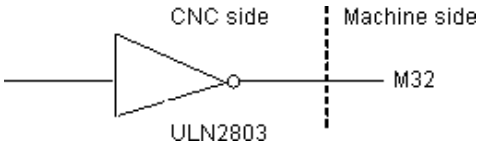


Fig. 2-51

- **Control parameters**

PLC bit parameters

K	1	0							RSJG	

RSJG =1: When RESET key is pressed, the output signals M03, M04, M08 and M32 are NOT shut down

=0: When RESET key is pressed, the output signals M03, M04, M08 and M32 are shut down

K	1	6						M32A		

M32A =1: The lubrication is output at power-on when automatic lubrication is valid.

=0: The lubrication is NOT output at power-on when automatic lubrication is valid.

PLC data

D	T	0	5	3	

Automatic lubrication interval time (0~65535ms).

D	T	0	1	3	

Automatic lubrication output time (0~65535ms).

- **Function description**

There are two types of lubrication function defined by TAC2000 standard PLC program: Manual lubrication and automatic lubrication, and they are set by parameters: DT13 =0: Manual lubrication

>0: Automatic lubrication, the lubrication time is set by DT13 and the lubrication interval time is set by DT53.

1. Manual lubrication function

For lubrication turnover output, press the machine operation panel LUBRICATING key, the lubrication is output, and press it again, the lubrication output is canceled. During executing M32, the lubrication is output, and M33 is executed, the lubrication output is canceled.

When DT13>1, the lubrication is output in fixed time, press the machine operation panel LUBRICATING key, the lubrication is output, and after the time set by DT13, the lubrication output is canceled; M32 is executed, the lubrication is output, and the lubrication output is canceled in the time set by DT13. If the time set by DT13 is not up, execute M33 or press LUBRICATING key once again, and then, the lubrication output is canceled.

2. Automatic lubrication:

When K16.2 is set as 1, after power-on, the system starts lubricating in the time set by DT13, and then the output stops. And then, the lubrication is output, again after the time set by DT53 is up, the process is executed in cycle. During automatic lubricating, codes of M32 and M33 and the machine operation panel 

key are also valid, and the lubrication time is still set by DT13.

Note 1: During CNC emergency stop, the lubrication output is OFF.

Note 2: During CNC resetting, whether cancel the lubrication output is set by Bit1 of K10: When Bit1＝0, close lubrication output by CNC resetting;

When Bit1＝1, the lubrication output state remains unchanged during CNC resetting.

2.8.10 Safety door detection

- **Relevant signal**

SAGT: Safety door detection input signal

- **Signal diagnosis**

Signal	SAGT
Diagnosis address	X0.0
Interface pin	CN61.1

- **Control parameter**

State parameter

K	1	4						SPB4	PB4		

PB4 =0: Safety door detection inactive

=1: Safety door detection active

SPB4 =0: For safety door closing as SAGT is cut off with +24V

=1: For safety door closing as SAGT is connected with +24V

- **Function description (defined by standard PLC program)**

①When PB4＝1, SPB4＝0, CNC confirms that the safety door is closed as SAGT is disconnected to +24V;

②When PB4＝1, SPB4＝1, CNC confirms that the safety door is closed as SAGT is cut off with +24V;

③The protection door detection function is valid in Auto mode; however, when the protection door is open, the alarm of "the protection door is open" occurs in all modes, but it doesn't affect the operation;

④In Auto mode, during the automatic cycle start, if CNC has detected the protection door open, the alarm is issued;

⑤During automatic running, if CNC has detected the protection door is open, the axis feeding dwells, and the cooling output is closed. If SGSP is set as 0, the spindle output is also closed meanwhile; otherwise, the spindle output isn't closed.

2.8.11 CNC macro variables

- **Relevant signal**

  Macro output signal: standard PLC defines 5 macro output interfaces #1100～#1105； Macro input signal: standard PLC defines 16 macro output interfaces #1000～#1015

- **Signal diagnosis**

Macro variable number	#1105	#1104	#1103	#1102	#1101	#1100
Diagnosis address	Y3.7	Y3.6	Y3.5	Y3.4	Y3.3	Y3.2

Macro variable number	#1007	#1006	#1005	#1004	#1003	#1002	#1001	#1000
Diagnosis address	X0.7	X0.6	X0.5	X0.4	X0.3	X0.2	X0.1	X0.0

Macro variable number	#1015	#1014	#1013	#1012	#1011	#1010	#1009	#1008
Diagnosis address	X1.7	X1.6	X1.5	X1.4	X1.3	X1.2	X1.1	X1.0

- **Function description (defined by standard PLC program)**

U00～U05 signal output may be changed if macro variable ＃1100～＃1105 are assigned. If they are assigned for “1”, it outputs 0V, if they are assigned for “0”, it turns off their output signals.

Detect the values of the macro variables ＃1000～＃1015 in channel 1, the input status of the input interfaces X0.0~X0.7 and X1.0~X1.7 can be got.

2.8.12 Tri-colour indicator

Relevant signals and function definitions:
Y2.2 (CN62.31): yellow indicator,normal (non-running, non-alarming) Y2.3 (CN62.32): green indicator,running
Y2.4 (CN62.33): red indicator,alarming

2.8.13 External MPG

- **Related signals**

CN31(MPG)	PLC address	Address character	Function	Remark
5	X5.0	EHDX	X MPG	Applied to PSG-100-05E/L, ZSSY2080 MPG
6	X5.1	EHDY	Y MPG	
8	X5.2	EHDZ	Z MPG	
9	X5.3	EMP0	Increment ×1	
22	X5.4	EMP1	Increment ×10	
23	X5.5	EMP2	Increment ×100	
11, 12, 13	GND			
14,15	+5V			
17,18	+24V			

- **Related parameters**

Bit parameter

0	0	1						MPG			

Bit3 =0：Step working mode.

=1：MPG working mode.

PLC bit parameter

K	1	6	SINC								

SINC = 0：MPG, STEP mode ×1000-gear increment is valid.

=1：MPG, STEP mode×1000-gear increment is invalid.

- **Function description**

① When SINC is set to 1, MPG/STEP mode ×1000-gear selection is disabled. When x1000-gearis selected before modifying the parameter, the system automatically changes into ×100mm-gear

② When the external MPG, its axis selection does not lock, that is, the axis selection of MPG is disabled, the system changes to the non-axis selection state.

③ When the external MPG axis selection and gear selection input are enabled, the axis selection on the panel and the gear selection keys are disabled; when the external MPG axis selection and gear selection input are disabled, the axis selection on the panel and the gear selection keys are enabled and self-locked.

CHAPTER 3 PARAMETERS

The CNC bit and data parameters are described in this chapter, various functions can be set by these parameters.

3.1 Parameter description (by sequence)

3.1.1 Bit parameter

The state parameter is expressed as follows:

Parameter No.			BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0	0	1	***	***	***	SPTY	SOHW	***	***	INI

Bit0 0: Metric input
1: Inch input
Bit3 0: Step mode
1: MPG mode
Bit4 0: Spindle switching volume control
1: Spindle analog voltage control
Default:0 0 0 1 1 0 0 0

0	0	3	***	***	SCRW	TLC	***	***	CIM	OIM

Bit0 0: Not convert the tool compensation value during the metric and inch system switch
1: Automatically convert the tool compensation value during the metric and inch system switch Bit1 0:
Not convert the workpiece coordinate value during the metric and inch systemswitch
1: Automatically convert workpiece coordinate value during the metric and inch system switch
Bit4 0: Select mode B of tool length compensation 1:
Select mode A of tool length compensation
Bit5 0: Screw pitch error compensation function invalid 1:
Screw pitch error compensation function valid
Default:0 0 1 1 0 0 1 1

0	0	4	SCW	***	***	***	***	***	***	***

Bit7 1: Inch system output 0:
Metric system output
Default:0 0 0 0 0 0 0 0

0	0	5

***	***	***	M30	M02	***	***	***

Bit3 0: Cursor to beginning after M02 execution

1: Cursor not to beginning after M02 execution

Bit4 0: Cursor to beginning after M30 execution

1: Cursor not to beginning after M30 execution

Default:0 0 0 1 1 0 0 0

0	0	6

***	***	***	MAOB	ZPLS	***	***	ZMOD

Bit0 0：Reference return mode selection: in front of the block 1：

Reference return mode selection: behind the block

Bit3 0：Zero type selection: non-one-revolution signal 1：

Zero type selection: one-revolution signal

Bit4 0: Zero return mode when have not one-turn signal: A mode

1: Zero return mode when have not one-turn signal: B mode Default:0 0

0 0 1 0 0 0

0	0	8

***	***	***	DIR5	DIR4	DIRZ	DIRY	DIRX

Bit0 0: Direction signal (DIR) is high level as X axis moves negatively 1:

Direction signal (DIR) is high level as X axis moves positively

Bit1 0: Direction signal (DIR) is high level as Y axis moves negatively 1:

Direction signal (DIR) is high level as Y axis moves positively

Bit2 0: Direction signal (DIR) is high level as Z axis moves negatively 1:

Direction signal (DIR) is high level as Z axis moves positively

Bit3 0: Direction signal (DIR) is high level as 4th axis moves negatively 1:

Direction signal (DIR) is high level as 4th axis moves positively

Bit4 0: Direction signal (DIR) is high level as 5th axis moves negatively 1:

Direction signal (DIR) is high level as 5th axis moves positively

Default:0 0 0 1 1 1 0 1

0	0	9

SALM	***	***	5ALM	4ALM	ZALM	YALM	XALM

Bit0 0: X alarm signal signal is high level alarm 1:

X alarm signal signal is low level alarm

Bit1 0: Y alarm signal is high level alarm 1:

Y alarm signal is low level alarm

Bit2 0: Z alarm signal is high level alarm 1:

Z alarm signal is low level alarm

Bit3 0: 4th alarm signal is high level alarm 1:
4th alarm signal is low level alarm

Bit4 0: 5th alarm signal is high level alarm
1: 5th alarm signal is low level alarm

Bit7 0: Spindle alarm signal is high level alarm 1:
Spindle alarm signal is low level alarm

Default:0 0 0 0 0 0 0 0

0	1	1

RVCS	***	***	***	***	***	***	***

Bit7 0: Backlash compensation mode:fixed frequency 1:
Backlash compensation mode:acc and dec

Default:0 0 0 0 0 0 0 0

0	1	2

***	***	***	***	***	***	***	ISOT

Bit0 0: Prior to machine zero return after power on, manual rapid traverse active 1:
Prior to machine zero return after power on, manual rapid traverse inactive

Default:0 0 0 0 0 0 0 0

0	1	3

HPF	RHPG	***	***	***	***	***	HNGD

Bit0 0：Coordinates increase in all axis MPG (CCW) rotation 1：
Coordinates decrease in all axis MPG (CW)rotation

Bit6 0：Not use electronic MPG drive function
1：Use electronic MPG drive function

Bit7 0： MPG rotate displacement run completely 1：
MPG rotate displacement run incompletely

Default:1 0 0 0 0 0 0 1

0	1	4

***	***	***	***	***	***	RFO	LRP

Bit0 0：Positioning(G00) interpolation track:non-linear 1：
Positioning(G00) interpolation track:linear

Bit1 0：Not stop while rapid feeding when rapid feed override is Fo 1：
Stop while rapid feeding when rapid feed override is Fo

Default:0 0 0 0 0 0 0 0

0	1	5

JAX	***	***	***	DLF	ZRN	AZR	SJZ

Bit0 1: Memory mechanical zero is memorized
0: Memory mechanical zero is not memorized

Bit1 1: G28 instruction alarms when the reference point is not set up
0: G28 instruction uses the block when the reference point is not set up Bit2 1:
Instructions except G28 alarms when the reference point is not set up
0: Instructions except G28 do not alarm when the reference point is not set up Bit3
1: After the reference point, the point is returned to for manual speed
0: After the reference point, the point is returned to for quick speed.

Bit7 1: Not choose multi axis when manual back to zero 0:
Choose multi axis when manual back to zero

Default:0 0 0 0 0 1 0 0

0	1	6

WLOE	HLOE	CLLE	CBLS	CBOL	FLLS	FBLS	FBOL

Bit0 0: Rapid run mode:front acceleration and deceleration 1:
Rapid run mode:rear acceleration anddeceleration

Bit1 0: Rapid run front acceleration and deceleration :linear type
1: Rapid run front acceleration and deceleration :S type Bit2 0:
Rapid run rear acceleration and deceleration :linear type
1: Rapid run rear acceleration and deceleration:exponential type

Bit3 0: Cutting feed mode in none-preread:front acceleration and deceleration 1:
Cutting feed mode in none-preread:rear acceleration anddeceleration

Bit4 0: Cutting feed mode front acceleration and deceleration in none-preread way:linear type 1:
Cutting feed mode front acceleration and deceleration in none-preread way:S type

Bit5 0: Cutting feed mode rear acceleration and deceleration in none-preread way:linear type
1: Cutting feed mode rear acceleration and deceleration in none-preread way:exponential type

Bit6 0: Manual(JOG) run:linear acceleration and deceleration
1: Manual(JOG) run:exponential acceleration and deceleration Bit7
0: Manual run:linear acceleration and deceleration
1: Manual run:exponential acceleration and deceleration

Default:1 0 0 0 1 1 0 1

0	1	7

***	***	***	PIIS	PPCK	ASL	PLAC	STL

Bit0 0: select non-prereading working type 1:
select prereading working type

Bit1 0: Interpolation rear acceleration and deceleration in preread way:linear type
1: Interpolation rear acceleration and deceleration in preread way:exponential type

Bit2 0：Auto corner deceleration function in preread way:angle control
1：Auto corner deceleration function in preread way:speed difference control Bit3
0：Not carry on detection of in place in preread way
1：Carry on detection of in place in preread way

Bit4 0：Overlapping interpolation ineffective in acceleration/deceleration blocks before forecasting 1：
Overlapping interpolation effective in acceleration/deceleration blocks before forecasting

Default:1 1 0 0 0 0 0 1

0	1	8

***	***	***	***	CANT	***	CLV	CCV

Bit0 0: Macro program public variable #100~#199,not clear after reseting 1:
Macro program public variable #100~#199,clear after reseting

Bit1 0: Macro program local variable #1~#50,not clear after reseting
1: Macro program local variable #1~#50, clear after reseting Bit3
0：Single workpiece machinning time not clearautomatically
1：Single workpiece machinning time clear automatically

Default:0 0 0 0 0 0 0 0

0	1	9

G39	ODI	CCA	CCN	SUP	CNI	***	***

Bit2 0: Not carry on radius compensation intervene check 1:
Carry on radius compensation intervenecheck

Bit3 0: Tool start and tool retract's form are A type in tool radius compensation
1: Tool start and tool retract's form are B type in tool radius compensation

Bit4 0：Not cancle tool radius compensation when G28,G30 instruction move to middle point 1:
Cancle tool radius compensation when G28,G30 instruction move to middle point

Bit5 0：Cancle tool compensation standard action when G28,G30 instruction move to middle point 1:
Cancle tool radius verticality when G28,G30 instruction move to middlepoint

Bit6 0：Tool radius compensation value is set by radius value
1: Tool radius compensation value is set by diameter value

Bit7 0：Corner circular arc function is invalid in radius compensation 1:
Corner circular arc function is valid in radius compensation

Default:1 1 0 1 0 1 0 0

0	2	0

SPFD	SAR	***	VAL5	VAL4	VALY	VALZ	VALX

Bit7 1：In cutting feed, do not permit the spindle stopping rotation 0：
In cutting feed, permit the spindle stops rotation

Bit6 1: Detect spindle SAR signal prior to cutting
0: Not detect spindle SAR signal prior to cutting

Bit4 1：Not flip 5th axis movement key direction
0：Flip 5th axis movement key direction Bit3
1：Not flip 4th axis movement key direction
0：Flip 4th axis movement key direction

Bit2 1：Not flip Y axis movement key direction
0：Flip Y axis movement key direction

Bit1 1：Not flip Z axis movement key direction
0：Flip Z axis movement key direction

Bit0 1：Not flip X axis movement key direction
0：Flip X axis movement key direction

Default:0 0 0 0 0 0 0 0

0	2	1

***	***	***	***	***	MESP	MSP	MST

Bit0 1: External cycle start signal inactive； 0:
External cycle start signal active.

Bit0 1: External pause signal inactive； 0:
External pause signal active.

Bit2 1: Do not detect emergency signal； 0:
Detect emergency signal.

Default:0 0 0 0 0 0 1 1

0	2	2

AD2	***				BFA	LZR	UOUT2

Bit0 0: Inside the off-limits areas of 2nd travel limit 1:
Outside the off-limits areas of 2nd travel limit

Bit1 0: Soft limit is invalid before returning machine zero 1:
Soft limit is valid before returning machine zero

Bit2 0: Alarm before over-distance when give out over-distance instruction 1:
Alarm after over-distance when give out over-distance instruction

Bit7 0: Not alarm when instruct more than 2 same adress in a block 1: Alarm when instruct more than 2 same adress in a block

Default:1 0 0 0 0 0 0 1

0	2	5

NAT	RRW	***	***	***	WARP	PETP	SPOS

Bit0 0: RELATIVE POS displayed in POS&PRG page 1: DIS TO GO displayed in POS&PRG page

Bit1 0: Not switch to program interface by pressing edit key 1: Switch to program interface by pressing edit key

Bit2 0: Not switch to alarm interface when alarm occurs 1: Switch to alarm interface when alarm occurs

Bit2 0: Cursor return to the beginning of program in edit mode when reset 1: Cursor return to the beginning of program in all mode when reset

Bit7 0: Function ATAN, ASIN range is -90.0～90.0; 1: Function ATAN, ASIN range is 90.0～270.0

Default:0 0 0 0 0 0 1 0

0	2	6

***	***	***	ZMI5	ZMI4	ZMIY	ZMIZ	ZMIX

Bit0 0: Set the direction of the X axis returning to the reference point: positive direction 1: Set the direction of the X axis returning to the reference point: nagative direction

Bit1 0: Set the direction of the Z axis returning to the reference point: positive direction 1: Set the direction of the Z axis returning to the reference point: nagative direction

Bit2 0: Set the direction of the Y axis returning to the reference point: positive direction 1: Set the direction of the Y axis returning to the reference point: nagative direction

Bit2 0: Set the direction of the 4th axis returning to the reference point: positive direction 1: Set the direction of the 4th axis returning to the reference point: nagative direction

Bit4 0: Set the direction of the 5th axis returning to the reference point: positive direction 1: Set the direction of the 5th axis returning to the reference point: nagative direction

Default:0 0 0 0 0 0 0 0

0	2	9

***	***	NE9	NE8	***	***	***	***

Bit4 0: Not ban editing subprogram of No. 8000~8999 1:
Ban editing subprogram of No. 8000~8999

Bit5 0: Not ban editing subprogram of No. 9000~9999 1:
Ban editing subprogram of No. 9000~9999

Default:0 0 1 1 0 0 0 0

0	3	0

***	***	***	***	***	***	PRPD	PLA

Bit1 1: Axis rapid traverse rate of PLC by input value
0: Axis rapid traverse rate of PLC by parameter value Bit0
1: PLC axis control active
0: PLC axis control inactive

Default:0 0 0 0 0 0 0 0

0	3	3

***	***	RG90	***	***	AXSZ	AXSY	AXSX

Bit0 0: X axis is set to be linear axis 1:
X axis is set to be rotate axis

Bit1 0: Y axis is set to be linear axis 1:
Y axis is set to be rotate axis

Bit2 0: Z axis is set to be linear axis 1:
Z axis is set to be rotate axis

Bit5 0: Scale division instruction:specified by G90/G91 1:
Scale division instruction:absolute instruction

Default:0 0 0 0 0 0 0 0

0	3	4

SATP	***	RCS4	***	***	***	ROS4	ROT4

Bit0 0: Set 4th axis to be the linear axis 1:
Set 4th asix to be the rotary axis

Bit1 0: Set 4th to be the rotary axis(B type) 1:
Set 4th to be the rotary axis(A type)

Bit5 0 : 4th Cs axis function is valid 1:
4th Cs axis function is invalid

Bit0 0: 3 axis linkage system
1: 4 axis linkage system

Default:0 0 0 0 0 0 0 0

0	3	5

***	***	***	***	***	RRL4	RAB4	ROA4

Bit2 1：When 4th is the rotary axis, the relative coordinate cycle function is valid

0：When 4th is the rotary axis, the relative coordinate cycle function is invalid Bit1

1：4th rotates according to the symbol when it is the rotary axis

0：4th rotates contiguously when it is the rotary axis

Bit0 1：The absolute coordinate cycle function is valid when 4th is the rotary axis 0：The absolute coordinate cycle function is invalid when 4th is the rotary axis

Default:0 0 0 0 0 0 0 0

0	3	6

***	***	RCS5	***	***	***	ROS5	ROT5

Bit5 0：5th Cs function is valid；

1：5th Cs function is invalid

Bit1 0： sets 5th to be the rotary axis(A type)，

1： sets 5th to be the rotary axis(B type)， Bit0

0: sets 5th to be the linear

1: sets 5th to be the rotary

Default:0 0 0 0 0 0 0 0

0	3	7

***	***	***	***	***	RRL5	RAB5	ROA5

Bit2 1：When 5th is the rotary axis, the relative coordinate cycle function is valid 0：

When 5th is the rotary axis, the relative coordinate cycle function is invalid

Bit1 1：5th rotates according to the symbol when it is the rotary axis

0：5th rotates contiguously when it is the rotary axis

Bit0 1：The absolute coordinate cycle function is valid when 5th is the rotary axis 0：The absolute coordinate cycle function is invalid when 5th is the rotary axis

Default:0 0 0 0 0 0 0 0

0	4	2

***	***	***	***	RIN	***	***	SSC

Bit0 0：Not use constant surface cutting speed control function 1：

Use constant surface cutting speed control function

Bit3 0：Rotation angle of G68 coordinate rotation:abslute instruction

1：Rotation angle of G68 coordinate rotation:G90/G91 instruction

Default:0 0 0 0 1 0 0 0

0	4	3

XSC	***	***	SCLZ	SCLY	SCLX	***	SCL

Bit0 0: Not use zoom function 1:
Use zoom function

Bit2 0: X axis zoom function is invalid 1:
X axis zoom function is valid

Bit3 0: Y axis zoom function is invalid 1:
Y axis zoom function is valid

Bit4 0: Z axis zoom function is invalid 1:
Z axis zoom function is valid

Bit7 0: Mode of every zoom ratios:every axis uses P instruction 1:
Mode of every zoom ratios:every axis uses IJK instruction

Default:1 0 0 1 1 1 0 1

0	4	4

QZA	***	RD2	RD1	MUNI	***	***	***

Bit3 0: G76,G87 displacement:Q instruction
1: G76,G87 displacement:I,J,K instruction

Bit4 0: Set tool retract direction of G76,G87:positive Set
1: tool retract direction of G76,G87:negative

Bit5 0: Set tool retract axis of G76,G87:X axis Set
1: tool retract axis of G76,G87:Y axis

Bit7 0: In deep-hole drilling(G73,G83),not alarm without instruction penetration value
1: In deep-hole drilling(G73,G83),alarm without instruction penetration value

Default:1 0 0 0 0 0 0 0

0	4	5

***	***	***	***	***	***	***	DWL

Bit0 0: Not clear F,H,D code when reset or emergencystop 1:
Clear F,H,D code when reset or emergency stop

Default:0 0 0 0 0 0 0 0

0	4	6

C07	C06	C05	C04	C03	C02	C01	***

Bit1 0: Not clear 01 group G code when reset or emergency stop 1:
Clear 01 group G code when reset or emergency stop

Bit2 0: Not clear 02 group G code when reset or emergencystop 1:
Clear 02 group G code when reset or emergency stop

Bit3 0: Not clear 03 group G code when reset or emergencystop 1:
Clear 03 group G code when reset or emergency stop

Bit4 0: Not clear 04 group G code when reset or emergency stop Clear
1: 04 group G code when reset or emergency stop

Bit5 0: Not clear 05 group G code when reset or emergency stop Clear
1: 05 group G code when reset or emergency stop

Bit6 0: Not clear 06 group G code when reset or emergency stop Clear
1: 06 group G code when reset or emergency stop

Bit7 0: Not clear 07 group G code when reset or emergency stop
1: Clear 07 group G code when reset or emergency stop

Default:1 0 0 0 0 0 0 0

0	4	7

C15	C14	C13	C12	C11	C10	C09	C08

Bit0 0: Not clear 08 group G code when reset or emergency stop 1:
Clear 08 group G code when reset or emergency stop

Bit1 0: Not clear 092group G code when reset or emergency stop 1:
Clear 09 group G code when reset or emergency stop

Bit2 0: Not clear 10 group G code when reset or emergency stop 1:
Clear 10 group G code when reset or emergency stop

Bit3 0: Not clear 11 group G code when reset or emergency stop 1:
Clear 11 group G code when reset or emergency stop

Bit4 0: Not clear 12 group G code when reset or emergency stop 1:
Clear 12 group G code when reset or emergency stop

Bit5 0: Not clear 13 group G code when reset or emergency stop 1:
Clear 13 group G code when reset or emergency stop

Bit6 0: Not clear 14 group G code when reset or emergency stop 1:
Clear 14 group G code when reset or emergency stop

Bit7 0: Not clear 15 group G code when reset or emergency stop 1:
Clear 15 group G code when reset or emergency stop

Default:0 0 0 0 0 0 0 1

0	4	8

***	***	G13	G91	G19	G18	G17	G01

Bit0 0: Set to G00 way when power on or in the state of clearing 1: Set
to G01 way when power on or in the state of clearing

Bit1 0: Plane selection is not G17 when power on or in the state of clearing 1:
Plane selection is G17 when power on or in the state ofclearing

Bit2 0: Plane selection is not G18 when power on or in the state of clearing 1:
Plane selection is G18 when power on or in the state ofclearing

Bit3 0: Plane selection is not G19 when power on or in the state of clearing 1:
Plane selection is G19 when power on or in the state ofclearing

Bit4 0: Set to G90 way when power on or in the state of clearing 1:
Set to G91 way when power on or in the state of clearing

Bit5 0: Set to G12 way when power on or in the state of clearing 1:
Set to G13 way when power on or in the state of clearing

Default:0 0 1 0 0 0 1 0

0	4	9

***	***	***	***	WZ0	MCV	GOF	WOF

Bit0 0: Input tool wear offset value in MDI mode
1: Ban inputing tool wear offset value in MDI mode

Bit1 0: Input tool geometry offset value in MDI mode
1: Ban inputing tool geometry offset value in MDI mode

Bit2 0: Input macro program variable in MDI mode
1: Ban inputing macro program variable in MDI mode

Bit3 0: Input workpiece origin offset value in MDI mode
1: Ban inputing workpiece origin offset value in MDI mode Default:0 0

0 0 0 0 0 0

0	5	0

DAL	***	***	***	MCL	MKP	MSL	SEQ

Bit0 0：Not insert number automatically
1：Insert number automatically

Bit0 0：Start line is 1st line when program status interface executes 1：Start line is cursor line when program status interface executes

Bit2 0：Not delete program after executive program at the interface 1：Delete program after executive program at the interface

Bit2 0：Not delete program when reset under the state interface 1：Delete program when reset under the state interface

Bit2 0：Absolute location display not considers tool length compensation 1：Absolute location display considers tool length compensation

Default:0 0 0 0 0 0 0 0

0	5	1

ITL	***	***	***	***	***	***	SCBM

Bit7 1： All axis interlocking signals are valid 0：
All axis interlocking signals are invalid

Bit0 1: Travel detection before moving
0: Not travel detection before moving

Default:0 0 0 0 0 0 0 0

0	5	2

MDLY	SBM	***	SIM	***	MDL	***	***

Bit2 0：Single direction localization G code is not set to modal code 1：
Single direction localization G code is set to modal code

Bit4 0：Do not make alarm if indexing instruction and other axes instructions are in the same block 1：
Make alarm if indexing instruction and other axes instructions are in the same block

Bit6 0：Can not use "single block" in the macro program instruction
1：Can use "single block" in the macro program instruction Bit7
0：Delay in the macro program instruction
1：Not delay in the macro program instruction Default:0 0
0 0 0 0 0 0

0	5	3

ZCL	RLC	***	***	***	***	***	***

Bit6 0：Relative coordinate system not cancle after reseting 1：
Relative coordinate system cancle after reseting

Bit7 0：Not cancle relative coordinate which is for returning reference point 1：
Cancle relative coordinate which is for returning reference point

Default:0 0 0 0 0 0 0 0

0	5	4

***	***	***	***	***	***	HPC	NPC

Bit0 0：Feeding is invalid when it doesn't install position encoder 1：
Feeding is valid when it doesn't install position encoder

Bit1 0：The system has not install position encoder 1：
The system has installs position encoder

Default:0 0 0 0 0 0 1 0

0	5	5

***	***	***	***	***	CALT	ALS	CPCT

Bit0 0：Cutting feed does not control in place precision 1：
Cutting feed controls in place precision

Bit1 0：Automatic corner ratio function is invalid
1：Automatic corner ratio function is valid

Bit2 0：Exponential acceleration and deceleration cutting feed accelerated speed does not clamp down 1：
Exponential acceleration and deceleration cutting feed accelerated speed clamps down

Default:0 0 0 0 0 0 0 1

0	5	6

***	***	***	***	***	TDR	FDR	RDR

Bit0 0: Dry running is invalid during cutting feed 1:
Dry running is valid during cutting feed

Bit1 0: Dry running is invalid during rapid localization 1:
Dry running is valid during rapid localization

Bit2 0: Dry running is invalid during the operation of tapping 1:
Dry running is valid during the operation of tapping

Default:0 0 0 0 0 0 0 0

0	5	7

DWL	***	SOC	RSC	***	***	***	***

Bit7 1: G04 is pause in every turn of the feed mode
0: G04 is not pause in every turn of the feed mode

Bit5 1: After the G96 spindle speed clamps down spindle override 0:
Before the G96 spindle speed clamps down spindle override

Bit4 1: G90 spindle speed when G0 positioning according to the current coordinate 0: G90
spindle speed when G0 positioning according to the Final coordinate

Default:0 0 0 0 0 0 0 0

0	5	8

OVU	DOV	T R	***	ORI	***		SSOG

Bit7 1: Rigid tapping knife back rate is 10% 0:
Rigid tapping knife back rate is 1%

Bit6 1: Rigid tapping knife back rate is valid 0:
Rigid tapping knife back rate is invalid

Bit5 1: Rigid tapping knife, knife back use the same time constant
0: Rigid tapping knife, knife back dose not use the same time constant Bit3
1: The spindle stop when flexible tapping at the beginning
0: The spindle does not stop when flexible tapping at the beginning

Bit1 1: Tapping into high speed deep hole tapping cycle 0:
Not into high speed deep hole tapping cycle

Bit0 1: Tapping mode of spindle control for servo 0:
Tapping mode of spindle control as follow

Default:0 0 0 0 0 0 0 0

0	5	9

LEDT	LOPT	OHPG	HISP	***	SOVD	FOVD	ROVD

Bit7 1: Use external editor lock

0: Not use external editor lock

Bit6 1: Use external operation panel lock

0: Not use external operation panel lock

Bit5 1: Use extenrnal MPG

0: Not use external MPG

Bit4 1: Use the external hand wheel/single-step interrupt function

0: Not use the external hand wheel/single-step interrupt function Bit2

1: Use band switch on the main shaft speed adjustment

0: Use the operating panel on the speed adjustment of the main shaft

Bit1 1: Use band switch on the cutting feed rate adjustment

0:Use the operating panel on the cutting feed rate adjustment Bit0 1:

Use band switch on the fast running rate adjustment

0: Use the operating panel on the fast running rate adjustment

Default:0 0 0 0 0 0 0 0

2	1	5

***	***	***	AALM	LALM	EALM	SALM	FALM

Bit4 1: Ignore external user alarm 0：Not

ignore external user alarm

Bit3 1：Ignore hard limit alarm

0: Not ignore hard limit alarm Bit2

1：Ignore emergency stop alarm

0: Not ignore emergency stop alarm Bit1

1: Ignore alarm of the spindle drive

0: Not ignore alarm of the spindle drive

Bit0 1：Ignore alarm of the feed shaft drive

0：Not ignore alarm of the feed shaft drive

Default:0 0 0 0 0 0 0 0

3.1.2 Data parameter

0 0 0	CMRX(X axis)multiplier coefficient
0 0 1	CMRY(Y axis) multiplier coefficient
0 0 2	CMRX(Z axis)multiplier coefficient
0 0 3	CMR4th(4th axis) multiplier coefficient
0 0 4	CMR5th(5th axis)multiplier coefficient

［Data range］ 1 ～65536
Default value 1

0 0 5	X axis frequency division coefficient(CMD)
0 0 6	Y axis frequency division coefficient(CMD)
0 0 7	Z axis frequency division coefficient(CMD)
0 0 8	4th axis frequency division coefficient(CMD)
0 0 9	5th axis frequency division coefficient(CMD)

［Data range］ 1 ～65536

Electronic gear ratio formula:

$$\frac{CMR}{CMD} = \frac{P}{L \times 1000}$$

P: Feedback corresponding to the number of pulses when motor rotation L:

Movement of machine tools when motor rotation

Default value 1

0 1 0	X axis negative max. travel(1st travel limit)
0 1 2	Y axis negative max. travel(1st travel limit)
0 1 4	Z axis negative max. travel(1st travel limit)
0 1 6	4th axis negative max. travel(1st travel limit)
0 1 8	5th axis negative max. travel(1st travel limit)

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ -9999.9999～9999.9999

Default value -9999.9999

0	1	1	X axis positive max. travel(1st travel limit)
0	1	3	Y axis positive max. travel(1st travel limit)
0	1	5	Z axis positive max. travel(1st travel limit)
0	1	7	4th axis positive max. travel(1st travel limit)
0	1	9	5th axis positive max. travel(1st travel limit)

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ -9999.9999～9999.9999

Default value 9999.9999

0	2	0	X axis negative max. travel(2nd travel limit)
0	2	2	Y axis negative max. travel(2nd travel limit)
0	2	4	Z axis negative max. travel(2nd travel limit)
0	2	6	4th axis negative max. travel(2nd travel limit)
0	2	8	5th axis negative max. travel(2nd travel limit)

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ -9999.9999～9999.9999

Default value -9999.9999

0	2	1	X axis positive max. travel(2nd travel limit)
0	2	3	Y axis positive max. travel(2nd travel limit)
0	2	5	Z axis positive max. travel(2nd travel limit)
0	2	7	4th axis positive max. travel(2nd travel limit)
0	2	9	5th axis positive max. travel(2nd travel limit)

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ -9999.9999～9999.9999

Default value 9999.9999

0	3	0	Reverse gap compensation to determine the reverse accuracy (X0.0001

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ 0.000.1~1

Default value 0.01

0 3 1	X axis backlash compensation.
0 3 2	Y axis backlash compensation
0 3 3	Z axis backlash compensation.
0 3 4	4th axis backlash compensation.
0 3 5	5th axis backlash compensation.

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ 0~0.5000

Default value 0

Note： X is the diameter value.

0 3 6	Compensation step for X axis space with fixed frequency
0 3 7	Compensation step for Z axis space with fixed frequency
0 3 8	Compensation step for Y axis space with fixed frequency
0 3 9	Compensation step for 4TH axis space with fixed frequency
0 4 0	

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ 0~99.9999

Default value 0.003

0	4	1	Time constant of the reverse gap to the lifting speed mode

[Data unit] ms

[Data range] 0~400

Default value 20

0	5	0	X coordinate value of 1st reference point in the machine coordinate system
0	5	1	Y coordinate value of 1st reference point in the machine coordinate system
0	5	2	Z coordinate value of 1st reference point in the machine coordinate system
0	5	3	4th coordinate value of 1st reference point in the machine coordinate system
0	5	4	5th coordinate value of 1st reference point in the machine coordinate system

[Data unit]

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

[Data range] -9999.9999~9999.9999

Default value 0

0	6	0	X coordinate value of 3rd reference point in the machine coordinate system
0	6	1	Y coordinate value of 3rd reference point in the machine coordinate system
0	6	2	Z coordinate value of 3rd reference point in the machine coordinate system
0	6	3	4th coordinate value of 3rd reference point in the machine coordinate system
0	6	4	5th coordinate value of 3rd reference point in the machine coordinate system

[Data unit]

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

[Data range] -9999.9999~9999.9999

Default value 0

0	6	5
0	6	6
0	6	7
0	6	8
0	6	9

X coordinate value of 4th reference point in the machine coordinate system
Y coordinate value of 4th reference point in the machine coordinate system
Z coordinate value of 4th reference point in the machine coordinate system
4th coordinate value of 4th reference point in the machine coordinate system
5th coordinate value of 4th reference point in the machine coordinate system

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ -9999.9999~9999.9999

Default value 0

0	7	0
0	7	1
0	7	2
0	7	3
0	7	4

High speed of X axis returning to machine zero
High speed of Y axis returning to machine zero
High speed of Z axis returning to machine zero
High speed of 4th axis returning to machine zero
High speed of 5th axis returning to machine zero

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 10～9999

Default value 4000

0	7	5
0	7	6
0	7	7
0	7	8
0	7	9

X axis grid offset vlaue or reference point offset value
Y axis grid offset vlaue or reference point offset value
Z axis grid offset vlaue or reference point offset value
4th axis grid offset vlaue or reference point offset value
5th axis grid offset vlaue or reference point offset value

［Data unit］

Setting unit	Data unit
Metric input	mm/min
Inch input	inch/min
rotary axis	deg

Default value 30

0	8	0

Low speed of returning to machine zero(universal for all axis)

[Data unit]

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

[Data range] 1~60

Default value 40

0	8	2

Dry running speed

[Data unit]

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

[Data range] 0~9999

Default value 5000

0	8	3

Cutting feed speed after powering on

[Data unit]

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

[Data range] 0~9999

Default value 300

0	8	5

Fo speed of rapid running override for all axis(universal for all axis)

[Data unit]

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

[Data range] 0~1000

Default value 30

0	8	6

Rapid localization and max. controlled speed in none-preread mode(universal for all axis)

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 300~30000

Default value 8000

0	8	7

Rapid localization and min. controlled speed in none-preread mode(universal for all axis)

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0~300

Default value 0

0	8	8

Max. controlled speed in preread mode(universal for all axis)

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 300~9999

Default value 6000

0	8	9

Min. controlled speed in preread mode(universal for all axis)

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

Default value 50

0 9 0	X axis G0 rapid localization speed
0 9 1	Y axis G0 rapid localization speed
0 9 2	Z axis G0 rapid localization speed
0 9 3	4th axis G0 rapid localization speed
0 9 4	5th axis G0 rapid localization speed

［Data unit］

Setting unit	Data unit
Metric input	mm/min
Inch input	inch/min

［Data range］ 0～30000

Default value 5000

1 0 0	Exponential acc/dec accelerated speed clamping constant

［Data unit］ ms

［Data range］ 0~1000

Default value 50

1 0 2	Max. clamping speed when MPG runs imcompletely

［Data unit］

Setting unit	Data unit
Metric input	mm/min
Inch input	inch/min

［Data range］ 0～3000

Default value 2000

1 0 3	Accelerated speed clamping constant when MPG runs imcompletely

［Data range］ 0～1000

Default value 50

1	0	4

MPG linear acc/dec time constant

［Data unit］ ms

［Data range］ 1～4000

Default value 120

1	0	5

MPG exponential acc/dec time constant

［Data unit］ ms

［Data range］ 0～4000

Default value 80

1	0	8

Maximum clamp speed of step feed

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0～3000

Default value 1000

1	1	0

Feedrate of manual continuous feed for axes (JOG)

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0～9999

Default value 2000

1	1	1

Linear acc/dec time constant when every axis manual feeds

［Data unit］ ms

［Data range］ 0～400

Default value 100

Default value 50

1 1 2	**Exponential acc/dec time constant when every axis manual feeds**

［Data unit］ ms

［Data range］ 0～400

Default value 120

1 1 3	**X axis manual rapid localization speed**
1 1 4	**Y axis manual rapid rapid localization speed**
1 1 5	**Z axis manual rapid rapid localization speed**
1 1 6	**4th axis manual rapid rapid localization speed**
1 1 7	**5th axis manual rapid rapid localization speed**

［Data unit］

Setting unit	Data unit
Metric input	mm/min
Inch input	inch/min

［Data range］ 0～30000

Default value 5000

1 2 0	**Front acceleration&deceleration linear time constant of X axis rapid traverse**
1 2 1	**Front acceleration&deceleration linear time constant of Y axis rapid traverse**
1 2 2	**Front acceleration&deceleration linear time constant of Z axis rapid traverse**
1 2 3	**Front acceleration&deceleration linear time constant of 4th axis rapid traverse**
1 2 4	**Front acceleration&deceleration linear time constant of 5th axis rapid traverse**

［Data unit］ ms

［Data range］ 3～400

Default value 100

1 2 5	**Front acceleration&deceleration S type time constant of X axis rapid traverse**
1 2 6	**Front acceleration&deceleration S type time constant of Y axis rapid traverse**
1 2 7	**Front acceleration&deceleration S type time constant of Z axis rapid traverse**
1 2 8	**Front acceleration&deceleration S type time constant of 4th axis rapid traverse**
1 2 9	**Front acceleration&deceleration S type time constant of 5th axis rapid traverse**

［Data unit］ ms

［Data range］ 3～400

Default value 100

1 3 0	Rear acceleration&deceleration linear time constant of X axis rapid traverse
1 3 1	Rear acceleration&deceleration linear time constant of Y axis rapid traverse
1 3 2	Rear acceleration&deceleration linear time constant of Z axis rapid traverse
1 3 3	Rear acceleration&deceleration linear time constant of 4th axis rapid traverse
1 3 4	Rear acceleration&deceleration linear time constant of 5th axis rapid traverse

［Data unit］ ms

［Data range］ 3～400

Default value 80

1 3 5	Rear acceleration&deceleration S type time constant of X axis rapid traverse
1 3 6	Rear acceleration&deceleration S type time constant of Y axis rapid traverse
1 3 7	Rear acceleration&deceleration S type time constant of Z axis rapid traverse
1 3 8	Rear acceleration&deceleration S type time constant of 4th axis rapid traverse
1 3 9	Rear acceleration&deceleration S type time constant of 5th axis rapid traverse

［Data unit］ ms

［Data range］ 3～400

Default value 60

1 4 0	Max. number of merged program segment in none-preread mode

［Data range］ 0～10

Default value 0

1 4 1	Controlled precision of merged program segment in none-preread mode

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0.001～0.5

Default value 0.01

Default value 100

1	4	2

In place precision of cutting feed in none-preread mode

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0.001～0.5

Default value 0.03

1	4	4

Front acc.&dec. linear time constant of cutting feed in none-preread mode

［Data unit］ ms

［Data range］ 3～400

Default value 100

1	4	5

Front acc.&dec. s type time constant of cutting feed in none-preread mode

［Data unit］ ms

［Data range］ 3～400

Default value 100

1	4	6

Rear acc.&dec. linear time constant of cutting feed in none-preread mode

［Data unit］ ms

［Data range］ 3～400

Default value 80

1	4	7

Rear acc.&dec. exponential time constant of cutting feed in none-preread mode

［Data unit］ ms

［Data range］ 3～400

Default value 60

1	4	8

Min. speed of exponential acc.&dec. in none-preread mode

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0～9999

Default value 10

1	5	0

Max. number of merged program segment in preread mode

［Data range］ 0～15

Default value 0

1	5	1

Controlled precision of merged program segment in preread mode

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0.001～0.5

Default value 0.01

1	5	2

In place precision of cutting feed in preread mode

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0.01～0.5

Default value 0.01

1	5	4

Front acc.&dec. linear time constant of cutting feed in preread mode

［Data unit］

Setting unit	Data unit
Metric input	mm/s/s
Inch input	inch/s/s

［Data range］ 0～2000

Default value 250

1	5	5

Front acc.&dec. s type time constant of cutting feed in preread mode

[Data unit] ms

[Data range] 3～400

Default value 100

1	5	6

Rear acc.&dec. linear time constant of cutting feed in preread mode

[Data unit] ms

[Data range] 3～400

Default value 80

1	5	7

Rear acc.&dec. exponential time constant of cutting feed in preread mode

[Data unit] ms

[Data range] 3～400

Default value 60

1	5	8

Min. speed of exponential acc.&dec. in preread mode

[Data unit]

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

[Data range] 0～400

Default value 10

1	6	0

2 program segment critical angle of automatic corner deceleration in preread mode

[Data unit] angle

[Data range] 1～45

Default value 5

1	6	1

Min. feed speed of automatic corner deceleration in preread mode

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 10～1000

Default value 120

1	6	2

Every axis allowable deviation for deceleration function in speed difference way in preread mode

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 60～1000

Default value 80

1	6	3

Precision level of cutting machining in preread mode

［Data range］ 0～8

Default value 2

1	6	5

Length condition of forming spline in preread mode

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

1	6	6

Angle condition of forming spline in preread mode

［Data unit］ angle

［Data range］ 0～30

Default value 5

1	7	0

Accelerated speed limit outside circular interpolation

［Data unit］

Setting unit	Data unit
Metric machine	mm/s/s
Inch machine	inch/s/s

［Data range］ 100～5000

Default value 1000

1	7	1

Low speed lower bound of accelerated speed-clamped outside circular interpolation

［Data unit］

Setting unit	Data unit
Metric machine	mm/min
Inch machine	inch/min

［Data range］ 0～2000

Default value 200

1	7	2

Controlled precision of circular interpolation

［Data range］ 0～0.5

Default value 0.03

1	7	3

Limit value of circular radius error

［Data unit］

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

［Data range］ 0.0001～1

Default value 0.01

1	8	0
1	8	1
1	8	2
1	8	3
1	8	4

Pitch error compensation No. of the X axis reference point
Pitch error compensation No. of the Y axis reference point
Pitch error compensation No. of the Z axis reference point
Pitch error compensation No. of the 4th axis reference point
Pitch error compensation No. of the 5th axis reference point

［Data range］ 0～255

Default value 0

1	8	5
1	8	6
1	8	7
1	8	8
1	8	9

Pitch error compensation points of the X axis
Pitch error compensation points of the Y axis
Pitch error compensation points of the Z axis
Pitch error compensation points of the 4th axis
Pitch error compensation points of the 5th axis

［Data range］ 0～256

Default value 256

1	9	0
1	9	1
1	9	2
1	9	3
1	9	4

Pitch error compensation interval of the X axis
Pitch error compensation interval of the Y axis
Pitch error compensation interval of the Z axis
Pitch error compensation interval of the 4th axis
Pitch error compensation interval of the 5th axis

［Data unit］

Setting unit	Data unit
Metric input	mm
Inch input	inch

1 9 5	Pitch error compensation override of the X axis
1 9 6	Pitch error compensation override of the Y axis
1 9 7	Pitch error compensation override of the Z axis
1 9 8	Pitch error compensation override of the 4th axis
1 9 9	Pitch error compensation override of the 5th axis

[Data range] 0～99.9999
Default value 0.001

2 0 0 Spindle upper limit speed

[Data unit] r/min

[Data range] 0～99999
Default value 6000

2 0 1 Spindle encoder lines

[Data unit] line/r

[Data range] 100～5000
Default value 1024

2 0 2 The Max. setting value of the frequency-converter

[Data range] 4000～65536
Default value 65535

2 0 3 Spindle override lower limit

[Data range] 0～1
Default value 0

2	0	5

Gain adjustment data for spindle analog output

[Data range] 0.98～1.02

Default value 1

2	0	6

Compensation value of offset voltage for spindle analog output

[Data range] -0.2～0.2

Default value 0

2	0	8

Spindle speed in the spindle orientation or JOG

[Data unit] r/min

[Data range] 0～9999

Default value 50

2	0	9

Spindle upper limit speed in tapping cycle

[Data unit] r/min

[Data range] 0～5000

Default value 2000

2	1	0
2	1	1
2	1	2

Spindle maximum speed to gear 1

Spindle maximum speed to gear 2

Spindle maximum speed to gear 3

[Data unit] r/min

[Data range] 0~99999

Default value 6000

2	1	4

Output voltage(mV) when spindle shifts gear

[Data unit] mV

[Data range] 0~10000

Default value 100

2	2	0
2	2	1
2	2	2

Tooth number of spindle side gear (the 1st gear)
Tooth number of spindle side gear (the 2nd gear)
Tooth number of spindle side gear (the 3rd gear)

[Data range] 1~999

Default value 1

2	2	3
2	2	4
2	2	5

Tooth number of position encoder side gear (the 1st gear)
Tooth number of position encoder side gear (the 2nd gear)
Tooth number of position encoder side gear (the 3rdgear)

[Data range] 1~999

Default value 1

2	3	0
2	3	1
2	3	2

Spindle instruction multiplication coefficient (CMR) in tapping(the 1st gear)
Spindle instruction multiplication coefficient (CMR) in tapping(the 2nd gear)
Spindle instruction multiplication coefficient (CMR) in tapping(the 3rd gear)

[Data range] 1~999

Default value 512

2	3	3
2	3	4
2	3	5

Spindle instruction frequency division coefficient (CMD) in tapping(the 1st gear)
Spindle instruction frequency division coefficient (CMD) in tapping(the 2nd gear)
Spindle instruction frequency division coefficient (CMD) in tapping(the 3rd gear)

[Data range] 1~999

Default value 215

2 4 0	**Spindle clearance in rigid tapping (the 1st gear)**
2 4 1	**Spindle clearance in rigid tapping (the 2nd gear)**
2 4 2	**Spindle clearance in rigid tapping (the 3rd gear)**

[Data unit]

Setting unit	Data unit
Metric input	mm
Inch input	inch

[Data range] 0~99.9999

Default value 0

2 4 4	**Maximum spindle speed in rigid tapping (the 1st gear)**
2 4 5	**Maximum spindle speed in rigid tapping (the 2nd gear)**
2 4 6	**Maximum spindle speed in rigid tapping (the 3rd gear)**

[Data unit] r/min

[Data range] 0~9999

Default value 6000

2 5 0	**Linear acc./dec. time constants of spindle and tapping axis (the 1st gear)**
2 5 1	**Linear acc./dec. time constants of spindle and tapping axis (the 2nd gear)**
2 5 2	**Linear acc./dec. time constants of spindle and tapping axis (the 3rd gear)**

[Data unit] m

[Data range] s

Default value

0

~

9

9

9

9

200

2 5 3	2 5 5
2 5 4	

[Data unit]

［Data range］	**Linear acc./dec. time constant of spindle and tapping axis in retraction(the 1st gear)**
Default value	**Linear acc./dec. time constant of spindle and tapping axis in retraction(the 2nd gear)**
	Linear acc./dec. time constant of spindle and tapping axis in retraction(the 3rd gear)
	r/min
	0~9999
	200

2	6	0
2	6	1
2	6	2
2	6	3
2	6	4

External workpiece' origin offset amount along the X axis
External workpiece' origin offset amount along the Y axis
External workpiece' origin offset amount along the Z axis
External workpiece' origin offset amount along the 4th axis
External workpiece' origin offset amount along the 5th axis

[Data unit]

Setting unit	Data unit
Metric input	mm
Inch input	inch

[Data range] -999.999~999.9999

Default value 0

2	6	5
2	6	6
2	6	7
2	6	8
2	6	9
2	7	0
2	7	1
2	7	2
2	7	3
2	7	4
2	7	5
2	7	6
2	7	7
2	7	8
2	7	9
2	8	0
2	8	1
2	8	2
2	8	3
2	8	4
2	8	5
2	8	6
2	8	7
2	8	8
2	8	9

Workpiece' origin offset amount along X axis in workpiece coordinate1 in G54
Workpiece' origin offset amount along Y axis in workpiece coordinate1 in G54
Workpiece' origin offset amount along Z axis in workpiece coordinate1 in G54
Workpiece' origin offset amount along 4th axis in workpiece coordinate1 in G54
Workpiece' origin offset amount along 5th axis in workpiece coordinate1 in G54
Workpiece' origin offset amount along X axis in workpiece coordinate1 in G55
Workpiece' origin offset amount along Y axis in workpiece coordinate1 in G55
Workpiece' origin offset amount along Z axis in workpiece coordinate1 in G55
Workpiece' origin offset amount along 4th axis in workpiece coordinate1 in G55
Workpiece' origin offset amount along 5th axis in workpiece coordinate1 in G55
Workpiece' origin offset amount along X axis in workpiece coordinate1 in G56
Workpiece' origin offset amount along Y axis in workpiece coordinate1 in G56
Workpiece' origin offset amount along Z axis in workpiece coordinate1 in G56
Workpiece' origin offset amount along 4th axis in workpiece coordinate1 in G56
Workpiece' origin offset amount along 5th axis in workpiece coordinate1 in G56
Workpiece' origin offset amount along X axis in workpiece coordinate1 in G57
Workpiece' origin offset amount along Y axis in workpiece coordinate1 in G57
Workpiece' origin offset amount along Z axis in workpiece coordinate1 in G57
Workpiece' origin offset amount along 4th axis in workpiece coordinate1 in G57
Workpiece' origin offset amount along 5th axis in workpiece coordinate1 in G57
Workpiece' origin offset amount along X axis in workpiece coordinate1 in G58
Workpiece' origin offset amount along Y axis in workpiece coordinate1 in G58
Workpiece' origin offset amount along Z axis in workpiece coordinate1 in G58
Workpiece' origin offset amount along 4th axis in workpiece coordinate1 in G58
Workpiece' origin offset amount along 5th axis in workpiece coordinate1 in G58

2	9	0	Workpiece' origin offset amount along X axis in workpiece coordinate1 in G59
2	9	1	Workpiece' origin offset amount along Y axis in workpiece coordinate1 in G59
2	9	2	Workpiece' origin offset amount along Z axis in workpiece coordinate1 in G59
2	9	3	Workpiece' origin offset amount along 4th axis in workpiece coordinate1 in G59
2	9	4	Workpiece' origin offset amount along 5th axis in workpiece coordinate1 in G59

[Data unit]

Setting unit	Data unit
Metric input	mm
Inch input	inch

[Data range] -9999.999~999.9999

Default value 0

3	0	0	DNC mode select(0:U disk 1:Xon/Xoff 2:XModem)

[Data range] 0～2

Default value 0

3	0	1	Baudrate of communication channel (DNC)

[Data unit] bit/s

[Data range] 2400,4800,9600,14400,19200,28800,38400,57600,115200

Default value 115200

3	0	2	Baudrate of communication channel (file transmission)

[Data unit] bit/s

[Data range] 2400,4800,9600,14400,19200,28800,38400,57600,115200

Default value 115200

3	0	3	Axes controlled by the CNC

[Data range] 3～4

Default value 3

3	0	4	Current used ladder No.

[Data range] 0～99

Default value 1

3	0	5

CNC language selection(0: CH 1: EN 2: RuS 3: ESP)

[Data range] 0～3

Default value 0

3	1	3

Program name of the 4th axis(3:A 4:B 5:C)

[Data range] 3～5

Default value 3

3	1	6

Incremental amount for automatic sequence number insertion

[Data range] 0～1000

Default value 10

3	1	7

Tool offset heading number input by MDI disabled

[Data range] 0～9999

Default value 10

3	1	8

Tool offset numbers input by MDI disabled

[Data range] 0～9999

Default value 10

3	2	1

Output time of reset signal

[Data unit] ms

[Data range] 50~400

Default value 200

3	2	2

Bits allowable for M codes

[Data range] 1～2

Default value 2

3	2	3

Bits allowable for S codes

[Data range] 1～6

Default value 5

3	2	4

Bits allowable for T codes

[Data range] 1～6

Default value 4

3	2	7

Axis as counting for surface speed control

[Data range]

Setting value	meaning
0	X axis
1	Y axis
2	Z axis
3	4th axis
4	5th axis

Default value 0

3	2	8

Spindle minimum speed for constant surface speed control (G96)

[Data unit] r/min

[Data range] 0~9999

Default value 100

3	3	0

Limit with vector ignored when moving along outside corner in tool radius compensation C

[Data unit]

Setting unit	Data unit
Metric intput	mm
Inch intput	inch

[Data range] 0～9999.9999

Default value 0

3	3	1

Maximum value of tool wear compensation

[Data unit]

Setting unit	Data unit
Metric intput	mm
Inch intput	inch

[Data range] 0～999.9999

Default value 400

3	3	2

Maximum error value of tool radius compensation C

[Data unit]

Setting unit	Data unit
Metric intput	mm
Inch intput	inch

[Data range] 0.0001～0.01

Default value 0.001

3	3	3

Helical infeed radius coefficient in groove cycle

[Data range] 0.01～3

Default value 1.5

3	3	4

Retraction amount of high-speed peck drilling cycle G73

[Data unit]

Setting unit	Data unit
Metric intput	mm
Inch intput	inch

[Data range] 0～999.9999

Default value 2

3	3	5

Reserved space amount of canned cycle G83

[Data unit]

Setting unit	Data unit
Metric intput	mm
Inch intput	inch

[Data range] 0～999.9999

Default value 2

3	3	6

Minimum dwell time at the hole bottom

[Data unit] ms

[Data range] 0~1000

Default value 250

3	3	7

Maximum dwell time at the hole bottom

[Data unit] ms

[Data range] 1000~9999

Default value 9999

3	3	8

Override for retraction in rigid tapping

[Data range] 0.8～1.2

Default value 1

3	3	9

Retraction or spacing amount in peck tapping cycle

[Data unit]

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

[Data range] 0～100

Default value 0

3	4	2

Rotational angle with no rotational angle specified in G68 coordinate rotation

[Data unit] angle

[Data range] 0~9999.9999

Default value 0

3	4	4

Scaling with no scaling specified

[Data range] 0.0001～9999.9999

Default value 1

3	4	5

Scaling override of the X axis

[Data range] 0.0001～9999.9999

Default value 1

3	4	6

Scaling override of the Y axis

[Data range] 0.0001～9999.9999

Default value 1

3	4	7

Scaling override of the Z axis

[Data range] 0.0001～9999.9999

Default value 1

3	5	0

Dwell time unidirectional positioning

[Data unit] s

[Data range] 0~10

Default value 0

3	5	1
3	5	2
3	5	3
3	5	4
3	5	5

External workpiece' origin offset amount along the X axis

External workpiece' origin offset amount along the Y axis

External workpiece' origin offset amount along the Z axis

External workpiece' origin offset amount along the 4th axis

External workpiece' origin offset amount along the 5th axis

[Data unit]

Setting unit	Data unit
Metric machine	mm
Inch machine	inch

[Data range] -99.999~99.9999

Default value 0

3	6	0

Number of machined workpiece

[Data range] 0～9999

Default value 0

3	6	1

Total workpiece to be machined

[Data range] 0～9999

Default value 0

Chapter 4 Machine Debugging Methods and Modes

The trial run methods and steps at initial power on for this TAC2000-V are described in this chapter. The corresponding operation can be performed after the debugging by the following steps.

4.1 Emergency Stop and Limit

TAC2000-V system has a software limit function. It is suggested that hardware limit should be employed by fixing the stroke limit switches in the positive or negative axis. The connection is as follows (taking example of 2 axes):

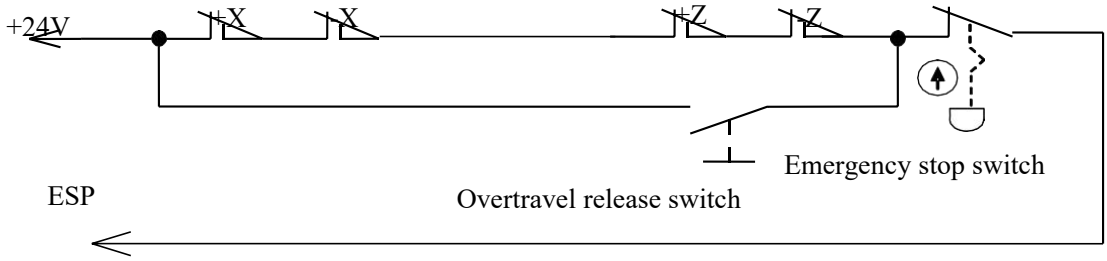


Fig. 4-1

So the BIT2 (MKYP) of bit parameter No.21 should be set to 0.

The diagnostic message DGN000.7 monitors the emergency stop input signal.

In Manual or MPG mode, slowly move the axes to testify the validity of stroke limit switch, correctness of alarm display, validity of overtravel release button. When the overtravel occurs or Emergency Stop button is pressed, “emergency stop” alarm will be issued by CNC system. The alarm can be cancelled by pressing down the OVERTRAVEL key for reverse moving.

4.2 Drive Unit Setting

BIT4, BIT3, BIT2, BIT1, BIT0 (5ALM, 4ALM, YALM, ZALM, XALM separately corresponds to 5th, 4th, Y, Z, X) of bit parameter No.009 for CNCmakers drive unit are all set to 1 according to the alarm logic level of the drive unit.

If the machine moving direction is not consistent with the move code, modify BIT4, BIT3, BIT2, BIT1 and BIT0 (DIR5, DIR4, DIRY, DIRZ, DIRX separately corresponds to 5th, 4th, Y, Z, X) of bit parameter No.008.

The manual move direction can be set by BIT4,BIT3,BIT2,BIT1, BIT0 (5VAL, 4VAL, YVAL, ZVAL, XVAL separately corresponds to 5th, 4th, Y,Z, X movement key) of bit parameter No.175.

4.3 Gear Ratio Adjustment

№000 ～ №009 can be modified for electronic gear ratio adjustment to meet the various mechanical transmission ratios when the machine travel distance is not consistent with the displacement distance displayed by the CNC.

Formula:

$$\frac{CMR}{CMD} \quad \frac{\delta *360}{\alpha *L} * \frac{Z_M}{Z_D}$$

CMR: Code multiplier coefficient (data parameter No.000,No.001,No.002,No.003,No.004)

CMD: Code frequency division coefficient (data parameter No.005,No.006,No.007,No.008,No.009)

: Pulse volume, motor rotation angle for a pulse L:

Screw lead

δ: Current min. input code unit of CNC

ZM : gear teeth number of lead screw ZD:

gear teeth number of motor

Example: if gear teeth number of lead is 50, gear teeth number of motor is 30, pulse volumer =0.075 degree,screw lead is 4mm.α

gear ratio is:

$$\frac{CMR}{CMD} = \frac{\delta \times 360}{\alpha \times L} \times \frac{Z_M}{Z_D} = \frac{0.001 \times 360}{0.075 \times 4} \times \frac{50}{30} = \frac{2}{1}$$

4.4 Acceleration & Deceleration Characteristic Adjustment

Adjust the relative CNC parameters according to the factors such as the drive unit, motor characteristics and machine load:

Data parameter No.090~No.094,No.113~No.117: rapid traverse rate of each axis;

Data parameter No.120~No.139: ACC&DEC time constant of each axis rapid traverse rate; Data parameter No.111~No.112: ACC&DEC time constant in manual feeding for every axis;

Data parameter No.154: Front acc.&dec. linear time constant of cutting feed in preread mode(mm/s/s); Data parameter No.102~No.105: ACC&DEC time constant and MPG spped;

Data parameter No.108: Maximum clamp speed of step feed;

Data parameter No.110: Feedrate of manual continuous feed for axes (JOG);

The larger the ACC&DEC time constant is, the slower the ACC&DEC is, the smaller the machine movement impact and the lower the machining efficiency is, and vice versa.

If ACC&DEC time constants are equal, the higher the ACC&DEC start/end speed is, the faster the ACC&DEC is, the bigger the machine movement impact and the higher the machining efficiency is, and vice versa.

The principle for ACC&DEC characteristic adjustment is to properly reduce the ACC&DEC time constant and increase the ACC&DEC start/end speed to improve the machining efficiency on the condition that there is no alarm, motor out-of-step and obvious machine impact. If the ACC&DEC time constant is set too small, and the start/end speed is set too large, it is easily to cause faults such as drive unit alarm, motor out-of-step or machine vibration.

4.5 Machine zero adjustment

Relatedsignal

DECX：X axis deceleration signal；

DECY：Y axis deceleration signal；

DECZ：Z axis deceleration signal；

DEC4：4THaxisdecelerationsignal；

DEC5：5THaxisdecelerationsignal； DGN

DATA

0	0	0				DEC5	DEC4	DECZ	DECY	DECX
Interface pin						CN61.34	CN61.33	CN61.12	CN61.32	CN61.4

ControlPAR

K	2	2	DEC4T	DECY	DECZ	DECX				

DEC4T=0：4th decelerates signal is low level；

=1：4th decelerates signal is high level。

DECY=0：Y decelerates signal is low level；

=1：Ydecelerates signal is high level。

DECZ=0：Z decelerates signal is low level；

=1：Zdecelerates signal is high level。

DECX=0：X decelerates signal is low level；

=1：Xdecelerates signal is high level。

0	0	6					ZPLS			ZMOD

ZMOD =1: Zero return mode selection: in front of the block.

=0: Zero return mode selection: behind the block.

ZPLS =1: Zero type selection: one-revolution signal Zero

=0: type selection: non-one-revolution signal

0	1	2								ISOT

ISOT=1：After electric power, the machine can move quickly and effectively；

=0：After the power, the machine to the zero point, the manual is invalid。

0	2	6				ZMI5	ZMI4	ZMIZ	ZMIY	ZMIX

MZRX=1：Set the direction of the axis returning to the reference point: nagative direction；

=0：Set the direction of the axis returning to the reference point: positive direction

DataPAR

0	8	0	ZRNFL

ZRNFL：Low rate back to zero。

0	7	0	ZRNFHX

ZRNFHX：X high rate back to zero。

0	7	1	ZRNFHY

ZRNFHY：Y high rate back to zero。

0	7	2	ZRNFHZ

ZRNFHZ：Z high rate back to zero。

0	7	3	ZRNFH4

ZRNFH4：4th high rate back to zero。

0	7	4	ZRNFH5

ZRNFH5：5th high rate back to zero。

Adjust the relevant parameters based on the active level of the connection signal, zero return type and direction applied:

BIT4, BIT5, BIT6，BIT7 of the K parameter No.022: valid level of deceleration signal as machine zero return

BIT0(ZMOD) of the bit parameter No.006: Zero return mode selection:(0:behind the block 1:in front of the block)

BIT3(ZPLS) of the bit parameter No.006: Zero type selection: (0:no 1:yes) have one-revolution signal Data parameter No.080: low deceleration speeds of each axis in machine zero return.

Data parameter No.070~No.074：high speed of each axis in machine zero return.

BIT0, BIT1, BIT2, BIT3, BIT4(ZMIX, ZMIY, ZMIZ, ZMI4, ZMI5) of the bit parameter No.026: each axis zero return direction: negative or positive.

Only the stroke limit switch validity is confirmed, can the machine zero return be performed.

The machine zero is usually fixed at the max. travel point, and the effective stroke of the zero return touch block should be more than 25mm to ensure a sufficient deceleration distance for accurate zero return. The more rapid the machine zero return is, the longer the zero return touch block should be. Or the moving carriage will rush over the block and it may affect the zero return precision because of the insufficient deceleration distance.

Usually there are 2 types of machine zero return connection:

① The connection to AC servo motor: using a travel switch and servo motor one-turn signal separately

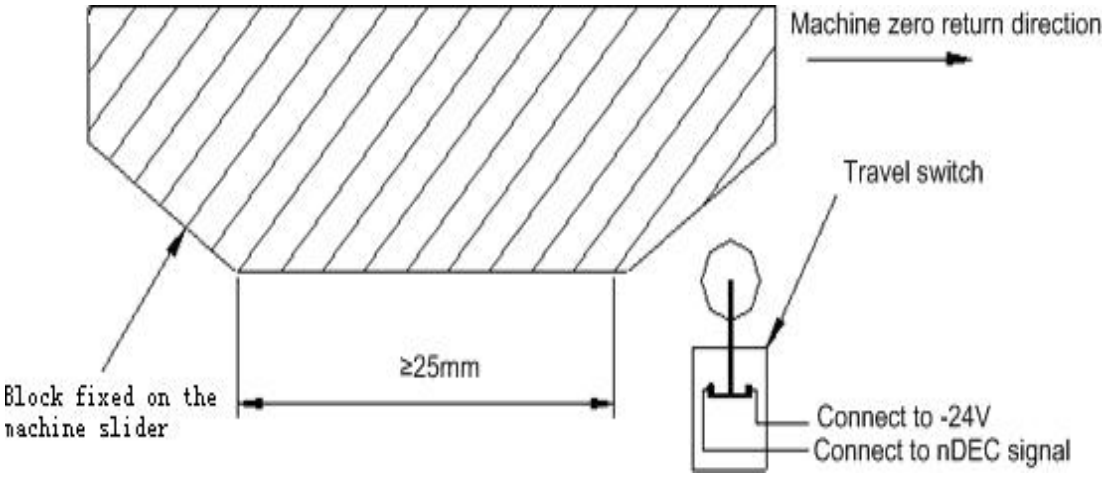


Fig. 4-2

By this connection, when the deceleration switch is released in machine zero return, the one-turn signal of encoder should be avoided to be at a critical point after the travel switch is released. In order to improve the zero return precision, and it should ensure the motor reaches the one-turn signal of encoder after it rotates half circle. The parameter setting is as follows:

Bit parameter No.006 BIT0(ZMOD) =0

Bit parameter No.006 BIT3(ZPLS) =1

Data parameter No.080=200

Bit parameter No.026 BIT0(ZMIX) , BIT1(ZMIY) , BIT2(ZMIZ), BIT3(ZMI4) , BIT4(ZMI5) =0

② The connection to stepper motor: schematic diagram of using a proximity switch taken as both deceleration signal and zero signal

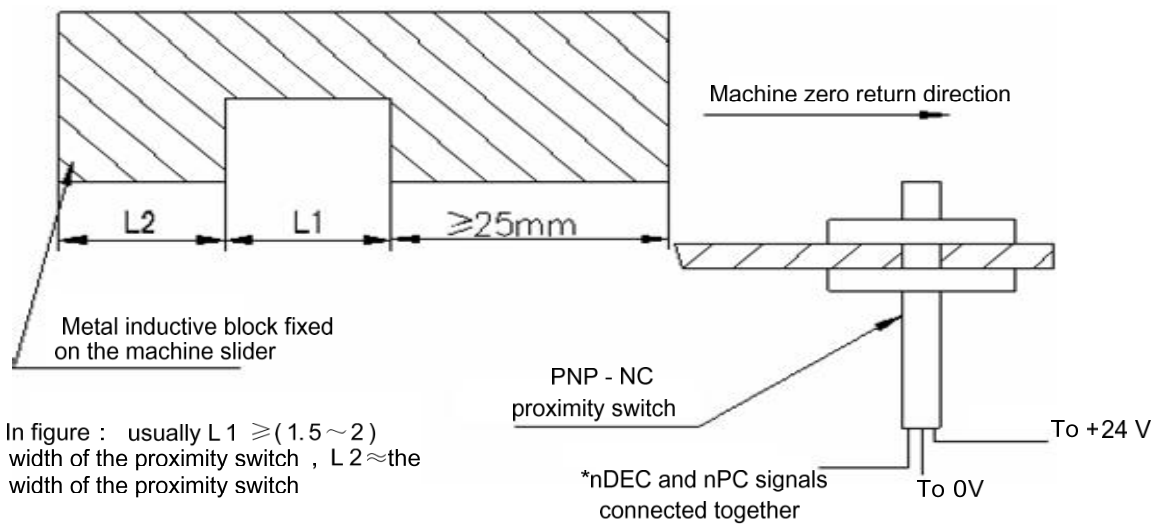


Fig. 4-3

When matching the stepper motor, the parameter settings are as follows:

Bit parameter No.006 BIT5(ZMOD) =0
Bit parameter No.006 BIT3(ZPLS) =0

Data parameter No.080=200
Data parameter No.026 BIT0(ZMIX), BIT1(ZMIY), BIT2(ZMIZ), BIT3(ZMI4), BIT4(ZMI5) =0

4.6 Spindle adjustment

4.6.1 Spindle encoder

Encoder with the pulses 100~5000p/r is needed to be installed on the machine for flexible tapping. The pulses are set by data parameter No.201. The transmission ratio(spindle gear teeth/ encoder gear teeth) between encoder and spindle is 1/255～255. The spindle gear teeth are set by CNC data parameter No.220, and the encoder gear teeth are set by data parameter No.223. Synchronous belt transmission should be applied for it (no sliding transmission).

4.6.2 Switch volume control of spindle speed

When the machine is controlled by a multi-speed motor, the motor speed codes are S01～S04.
The relevant parameters are as follows:

State parameter No.001 Bit4=0: select spindle speed switch control.

4.6.3 Analog voltage control of spindle speed

This function can be obtained by the parameter setting of CNC. By interface outputting 0V～10V analog voltage to control frequency inverter, the stepless shift can be obtained. And the related parameters needed to

be adjusted are:

Bit parameter No.001 Bit4=1: for spindle speed analog voltage control;
Data parameter No.026: offset value(mv) as spindle speed code voltage is 10V; Data parameter No.210～No.212: max. spindle speed of each gear;

Basic parameters are needed to adjust the inverter:
CW or CCW code mode selection:by common terminal VF;
Frequency setting mode selection:by common terminal FR; The concrete is referred to the user manual about inverter.

When the speed by programming is not consistent with that detected by the encoder, it can be adjusted to be consistent with the actual one by adjusting the data parameter No.210～No.212.

Speed adjustment method: select the corresponding spindle gear, determine the data parameter is 9999 as for this system gear, set the spindle override for 100%. Input spindle run command in MDI mode to run the spindle: M03/M04 S9999, view the spindle speed shown on the right bottom of the screen, then input the speed value displayed into the corresponding system parameter.

When entering S9999 code, the voltage should be 10V, S0 for 0V. If there is a voltage error, adjust bit parameter No.0206 to correct the voltage offset value (corrected by manufacturer, usually not needed).

For the max. speed of current gear, w the analog voltage output by CNC is not 10V, the CNC output analog voltage is set 10V by adjusting the data parameter No.206;

If the machine is not fixed with an encoder, the spindle speed can be detected by a speed sensing instrument, input S9999 in MDI mode to set the speed value displayed by the instrument into the data parameter No.210～No.212.

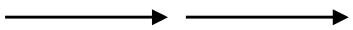
4.7 Backlash Offset

The backlash offset is input by actual measured offset The unit is mm or inch.It can be measured by a dial indicator,micrometer or a laser detector.Because the backlash offset can improve the machining precision only by accurate compensation, it is not recommended to measure it in MPG or Step mode, but the following method is suggested:

- Program editing (taking example of Z)

```
O0001；
N10 G01 Z10 F800 G91；  N20
Z15 ；
N30 Z1 ；
N40 Z-1 ；
N50 M30 。
```

- Set the backlash error offset to 0 before measuring;
- Run the program by single blocks, search the measuring benchmarkA after 2 positioning operations, record the current data, move 1mm in the same direction, then move 1mm to point B reversely, read the current data.



Reverse
position

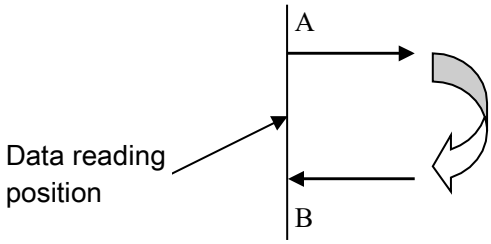


Fig. 4-4 Schematic map of backlash measuring method

- Backlash error offset value= | data of point A –data of point B |; then input its outcome to the CNC data parameter No.031~No.035.

Data A: dial-indicator data at point A Data
B: dial-indicator data at point B

Note 1: The backlash offset mode can be set by Bit7 of CNC parameter No.011; the Step size of backlash offset frequency can be set by data parameter No.036~No.040.
Note 2: Check the machine backlash every 3months.

4.8 Step/MPG Adjustment

The  key on the panel can be used to select the Step mode or MPG mode, which is set by the BIT3 of bit parameter No.001.

Bit3 =1: MPG mode is active, Step mode inactive;
=0: Step mode is active, MPG mode inactive;

4.9 Other Adjustment

0	2	1								MSP	MST

MST =0: External Cycle Start(ST) signal valid;
=1: External Cycle Start(ST) signal invalid.

MSP =0: External Stop(SP) signal valid.It is must connected with an external stop switch, or “HALT”will be shown by CNC
=1: External Dwell(SP) signal invalid.

CHAPTER 5 DIAGNOSIS MESSAGE

Diagnosis messages for TAC2000 system are described in this chapter.

5.1 CNC diagnosis

The part is used to check the CNC interface signals and internal running and it can't be modified.

5.1.1 I/O status and data diagnosis message

0	0	0
Pin		
PLC fixed address		

ESP	***	***	DEC5	DEC4	DECZ	DECY	DECX
CN61.6			CN61.34	CN61.33	CN61.12	CN61.32	CN61.4
X0.5			X2.5	X2.4	X1.3	X2.3	X0.3

DECX, DECY, DECZ, DEC4, DEC5：machine zero return signal of X, Y, Z, 4th, 5th ESP: emergency stop signal

0	0	1
Pin		
PLC fixed address		

***	***	***	***	***	***	***	SKIP
							CN61.42
							X3.5

SKIP：skip signal

5.1.2 CNC motion state and data diagnosis message

0	0	4

***	***	***	EN5	EN4	ENZ	ENY	ENX

EN5～ENX：enabling signal

0	0	5

***	***	***	SET5	SET4	SETZ	SETY	SETX

SET5～SETX：pulse prohibit signal

0	0	6

***	***	***	DRO5	DRO4	DROZ	DROY	DROX

DRO5～DROX：X, Y, Z, 4th, 5th motion direction output

0	0	9

***	***	***	5ALM	4ALM	ZALM	YALM	XALM

5ALM～XALM：X, Y, Z, 4th, 5th axis alarm signal

0	9	0	X output pulse quantity
0	9	1	Z output pulse quantity
0	9	2	Y output pulse quantity
0	9	3	4TH output pulse quantity
0	9	4	5TH output pulse quantity
1	4	0	MPG count value
1	4	4	Spindle encoder count value

5.1.3 Diagnosis keys

DGN.010 ~ DGN.016 are the diagnosis messages of edit keypad keys; When pressing a key in the operation panel, the corresponding bit displays “1”, and “0” after releasing this key. If it displays reversely, it means there is a fault in the keypad circuit.

0 1 0	9	8	7	P/Q	G	N	O	RST
Key	9	8	7	P Q	G *	N #	O _	// RESET //

0 1 1	6	5	4	W	U	Z	X	PGU
Key						Y &		

0 1 2				R	K			PGD
Key					K C	J B		

0 1 3	-	0	.	T	S	M	RIGHT	CRU
Key	>	0	<		S	M	⇨	⇧
	ALT	INS	EOB	F/E	D/L	H	LEFT	CRD
Key	ALTER	INSERT	EOB					

0 1 5	PLC	DGN	PAR	SET	ALM	OFT	PRG	POS
Key	PLC	DIAGNOSE	PARAMETER	SETTING	ALARM	OFFSET	PROGRAM	POSITION

0 1 6	IN	OUT	CHG	CAN	DEL	***	***	***
Key	DATA INPUT	DATA OUTPUT	CHANGE	CANCEL	DELETE			

5.1.4 Others

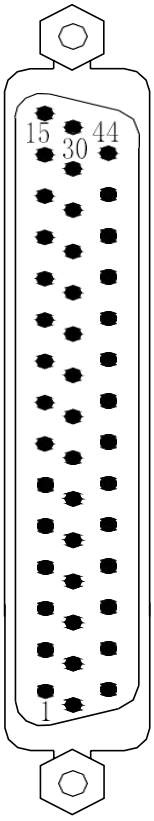
1	4	5
1	4	6

PLC execution time(ms)
Execution all time (h)

5.2 PLC state

This part of diagnosis is used to detect the signal state of machine→PLC(X), PLC→machine(Y), CNC→PLC (F), PLC→CNC (G) and alarm address A, and internal relay (R, K) states.

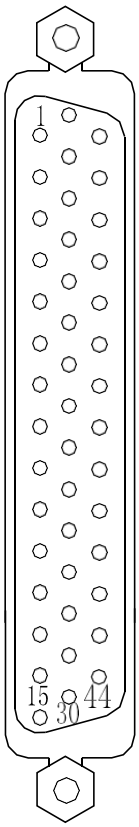
5.2.1 X address (machine→PLC , defined by standard PLC ladders)



PIN	Address	Function	Explain
21～24	0V	Power on	Power 0V
18～20 25～28	Floating	Floating	Floating
1	X0.0	SAGT	Guard door check signal
2	X0.1	SP	External pause
3	X0.2	THAN	External manual clamp/loose tool
4	X0.3	DECX	X deceleration signal
5	X0.4		Retention
6	X0.5	KYP	Emergency stop signal
7	X0.6	LIMU	Release overtravel input signal
8	X0.7	PRKY	Pressure detecion input signal
9	X1.0	TOPE	Spindle tool loose in-position signal
10	X1.1	TCLO	Spindle tool clamp in-posotion sign
11	X1.2	TZER	Tool magazine return zero key signa
12	X1.3	DECZ	Z deceleration signal
13	X1.4	ST	External cycle start
14	X1.5	M41I	Spindle auto gear shift 1-gear in-p
15	X1.6	M42I	Spindle auto gear shift 2-gear in-p
16			
29	X2.0	TFRX	Tool magazine forward in-position si
30	X2.1	TBAX	Tool magazine backward in-position s
31	X2.2	TCUX	Tool magazine count switch signal
32	X2.3	DEXY	X deceleration signal
33	X2.4	DEC4	4TH deceleration signal
34	X2.5	TZEX	Tool magazine return zero in-positi
35	X2.6	TRSW	Current tool pan tool detection swi
36	X2.7	TMSW	Spindle tool detection switch signa
37	X3.0	LMIX	X overtravel input

38	X3.1	LMIY	Y overtravel input
39	X3.2	LMIZ	Z overtravel input
40	X3.3	TCW	Tool CW key signla
41	X3.4	TCCW	Tool CCW key signal
42	X3.5	SKIP	G31 skip signal
43	X3.6	TFRX	Tool magazine forward key signal
44	X3.7	TBAX	Tool magazine backward key signal

5.2.2 Y address (PLC→machine, defined by standard PLC ladders)



(CN62)

PIN	Address	function	Explain
17～19 26～28	0V	Power on	Power 0V
20～25	+24V	Power on	Power＋24V
1	Y0.0	COO L	cooling output
2	Y0.1	M32	lubricating output
3	Y0.2	TCLA	1/0 Tool loose/clamp
4	Y0.3	M03	Spindle rotation CW
5	Y0.4	M04	Spindle rotation CCW
6	Y0.5	M05	Spindle stop
7	Y0.6	SCL P	Spindle clamped
8	Y0.7	SPZ D	Spindle brake
9	Y1.0	S1/M41	Spindle mechanical 1-gear
10	Y1.1	S2/M42	Spindle mechanical 2-gear
11	Y1.2	S3/M43	Spindle mechanical 3-gear
12	Y1.3	S4/M44	Spindle mechanical 4-gear
13	Y1.4		Spindle tool loose indicator
14	Y1.5		Retention
15	Y1.6		Retention
16	Y1.7		Retention
29	Y2.0	TLP	Retention
30	Y2.1		Retention
31	Y2.2	CLP Y	Three-color yellow-lamp
32	Y2.3	CLP G	Three-color green-lamp
33	Y2.4	CLP R	Three-color red-lamp
34	Y2.5		Retention
35	Y2.6		Retention
36	Y2.7		Retention
37	Y3.0	STA O	Spindle oriented output signal
38	Y3.1	TCCY	Tool magazine totation CW
39	Y3.2	TCW Y	Tool magazine totation CCW
40	Y3.3	TFR Y	Tool magazine forward
41	Y3.4	TBAY	Tool magazine backward
42	Y3.5	TBAL	Tool magazine backward indicator
43	Y3.6		Retention
44	Y3.7		Retention

5.2.3 Machine panel

TAC2000 controller panel

5.2.4 F address (CNC→PLC)

F000	OP	SA	STL	SPL	***	***	***	

OP: Auto run signal SA:
Servo ready signal
STL: Cycle start indicator signal
SPL: Feed hold indicator signal

F001	MA	***	TAP	ENB	DEN	***	RST	AL

MA: CNC ready signal
TAP: Tapping signal
ENB: Spindle enable signal
DEN: Designation end signal
RST: Reset signal
AL: Alarm signal

F002	MDRN	CUT	MSTOP	SRNMV	THRD		RPDO	

MDRN: Dry run detection signal
CUT: Cutting feed signal
MSTOP: Select stop detection signal
SRNMV: Program start signal
THRD: Threading signal
RPDO: Rapid feed signal

F003	***	MEDT	MMEM	MRMT	MMDI	MJ	MH	MINC

MEDT: Memory edit selection detection signal
MMEM: Auto run selection detection signal
MRMT: DNC run selection detection signal

MMDI: MDI selection detection signal MJ:

JOG selection detection signal

F004	***	MPST	MREF	MAFL	MSBK	MABSM	MMLK	MBDT

MH: MPG selection detection signal
MINC: Increment feed detection signal
MPST: Program beginning return detection signal
MREF: Manual reference return detection signal
MAFL: MST lock detection signal
MSBK: Single block detection signal
MABSM: Manaul absolute detection signal
MMLK: All machine axes lock detection signal
MBDT: Optional block skip detection signal

F007	***	***	***	***	TF	SF	***	MF

TF: Tool function strobe signal SF:
Spindle speed strobe signal
MF: MST function strobe signal

F009	DM00	DM01	DM02	DM30	***	***	***	RCT

DM00: M01 decoding signal
DM01: M02 decoding signal DM02:
M03 decoding signal DM30: M04
decoding signal RCT:
executing changing tool

F010	MB07	MB06	MB05	MB04	MB03	MB02	MB01	MB00

MB07: Miscellaneous function code M07
MB06: Miscellaneous function code M06
MB05: Miscellaneous function code M05
MB04: Miscellaneous function code M04
MB03: Miscellaneous function code M03
MB02: Miscellaneous function code M02
MB01: Miscellaneous function code M01
MB00: Miscellaneous function code M00

F014							DRUN	PDBG

PDBG: PLC enter debug mode DRUN:
No switching signal

F015				EN5T	EN4T	ENZ		

EN5T: 5TH axis selection
EN4T: 4TH axis selection
ENY: Z axis selection

F018	. AR07	. AR06 .	AR05 A	AR04	AAR03 A	AR02	AR01	AAR00

AR07:Actual speed of spindle AR07

AR06:Actual speed of spindle AR06
AR05:Actual speed of spindle AR05
AR04:Actual speed of spindle AR04
AR03:Actual speed of spindle AR03
AR02:Actual speed of spindle AR02

AR15	AR14	AR13	AR12	AR11	AR10	AR09	AR08

AR01:Actual speed of spindle AR01
AR00:Actual speed of spindle AR00

F019

AR15:Actual speed of spindle AR15
AR14:Actual speed of spindle AR14
AR13:Actual speed of spindle AR13
AR12:Actual speed of spindle AR12
AR11:Actual speed of spindle AR11
AR10:Actual speed of spindle AR10
AR09:Actual speed of spindle AR09
AR08:Actual speed of spindle AR08

F020

						BCLP	BUCLP

BCLP: 4
BUCLP: T

SB07	SB06	SB05	SB04	SB03	SB02	SB01	SB00

H axis indexing table clamp signal 4TH
axis indexing table release signal

F022

SB07: Spindle speed code signal SB07
SB06: Spindle speed code signal SB06
SB05: Spindle speed code signal SB05
SB04: Spindle speed code signal SB04

TB07	TB06	TB05	TB04	TB03	TB02	TB01	TB00

S
B

03: Spindle speed code signal SB03
SB02: Spindle speed code signal SB02
SB01: Spindle speed code signal SB01
SB00: Spindle speed code signal SB00

F026

TB07: Tool code signal TB07
TB06: Tool code signal TB06
TB05: Tool code signal TB05
TB04: Tool code signal TB04
TB03: Tool code signal TB03
TB02: Tool code signal TB02
TB01: Tool code signal TB01
TB00: Tool code signal TB00

F030	R08O	R07O	R06O	R05O	R04O	R03O	R02O	R01O

R08O: S12 bit code signal R08O
R07O: S12 bit code signal R07O
R06O: S12 bit code signal R06O
R05O: S12 bit code signal R05O
R04O: S12 bit code signal R04O
R03O: S12 bit code signal R03O
R02O: S12 bit code signal R02O
R01O: S12 bit code signal R01O

F031	***	***	***	***	R12O	R11O	R10O	R09O

R12O: S12 bit code signal R12O
R11O: S12 bit code signal R11O
R10O: S12 bit code signal R10O
R09O: S12 bit code signal R09O

F032	X1000	X100	X10	X1			RGSPM	RGSPP

X1000: Step X1000 soft key.
X100: Step X100 soft ke.
X10: Step X10 soft ke.
X1: Step X1 soft ke.
RGSPM: The reversal in rigid tapping
RGSPP: Rigid tapping spindle is in turn

F033	MTAP	DTAP						RTAP

MTAP: G63 tapping mode signal
DTAP: During rigid tapping signal
RTAP: Rigid tapping mode signal

F034	SSTOP	SCW	Z-	Z+	Y-	Y+	X-	X+

SSTOP：Spindle stop softkey

SCW：Rotating softkey

Z-: Z- softkey Z+：Z+ softkey Y-:
Y- softkey Y+: Y+ softkey

X-：X- softkey X+：X+ softkey

F035	SCCW	MSTOP	AFLO	BDTO	SBKO	MLKO	DRNO	QFAST

SCCW：Spindle rotate(CCW) softkey

MSTOP：Choose to stop softkey AFLO：
The auxiliary function lock key BDTO：Hop
key program

SBKO：Single program softkey

MLKO：Machine lock key

DRNO：Dry run softkey

QFAST：Fast moving softkey

F036	S-	S+	FAST-	FAST+			FEED-	FEED+

S-：Spindle override decrease soft key S+：
Spindle override increase soft key FAST-：
Rapid override decrease soft key FAST+：
Rapid override decrease soft key FEED-：Feed
override decrease soft key FEED+：Feed
override increase soft key

F037				ZP5	ZP4	ZP3	ZP2	ZP1

ZP5：Reference point return end signal ZP5 ZP4：
Reference point return end signal ZP4 ZP3 ：
Reference point return end signal ZP3 ZP2 ：
Reference point return end signal ZP2 ZP1 ：
Reference point return end signal ZP1

F038				MV5	MV4	MV3	MV2	MV1

MV5：AxismovesignalMV5

MV4：AxismovesignalMV4

MV3：AxismovesignalMV3

MV2：AxismovesignalMV2

MV1：AxismovesignalMV1

F039				MVD5	MVD4	MVD3	MVD2	MVD1

MVD5： Axis move direction signal MVD5

MVD4： Axis move direction signal MVD4

MVD3： Axis move direction signal MVD3

MVD2： Axis move direction signal MVD2

MVD1：AxismovedirectionsignalMVD1

F040				ZRF5	ZRF4	ZRF3	ZRF2	ZRF1

ZRF5 ： Reference point creation signal ZRF5

ZRF4 ： Reference point creation signal ZRF4

ZRF3 ： Reference point creation signal ZRF3

ZRF2 ： Reference point creation signal ZRF2

ZRF1：ReferencepointcreationsignalZRF1

F041				ZP15	ZP14	ZP13	ZP12	ZP11

ZP15：5TH axis 1st reference point return end signal ZP14：4TH axis 1st reference point return end signal ZP13：Y axis 1st reference point return end signal ZP12：Z axis 1st reference point return end signal ZP11：Xaxis1streferencepointreturnend signal

F042				PRO5	PRO4	PRO3	PRO2	PRO1

PRO5 ： Program zero return end signal PRO5

PRO4 ： Program zero return end signal PRO4

PRO3 ： Program zero return end signal PRO3

PRO2 ： Program zero return end signal PRO2

PRO1：Programzeroreturnendsignal PRO1

F043								MSPHD

MSPHD：Spindle jog detection signal

F044				SIMSPL			FSCSL	

SIMSPL：Analog spindle valid

FSCSL：Cs contour control switch end signal

F047	Total tool number

F048		MST	MSP		MESP			

MST：Shield external cycle start signal MSP：Shield external pause signal

MKYP: Shield external emergency stop sign

F051				VAL5	VAL4	VAL3	VAL2	VAL1

VAL5：5TH axis direction selection

VAL4：4TH axis direction selection

VALY：Y axis direction selection

VALZ：Z axis direction selection

VALX：X axis direction selection

F054	UO07	UO06	UO05	UO04	UO03	UO02	UO01	UO00

UO07: Macro output signal UO07

UO06: Macro output signal UO06

UO05: Macro output signal UO05

UO04: Macro output signal UO04

UO03: Macro output signal UO03

UO02: Macro output signal UO02

UO01: Macro output signal UO01

UO00: Macro output signal UO00

F0055	UO15	UO14	UO13	UO12	UO11	UO10	UO09	UO08

UO15: Macro output signal UO15

UO14: Macro output signal UO14

UO13: Macro output signal UO13

UO12: Macro output signal UO12

UO11: Macro output signal UO11

UO10: Macro output signal UO10

UO09: Macro output signal UO09

UO08: Macro output signal UO08

F057				ZP25	ZP24	ZP23	ZP22	ZP21

ZP25：5THaxis2ndreferencepointreturnendsignal ZP24：4TH axis 2nd reference point return endsignal ZP23：Z axis 2nd reference point return end signal ZP22：Y axis 2nd reference point return end signal ZP21：X axis 2nd reference point return endsignal

F058				ZP35	ZP34	ZP33	ZP32	ZP31

ZP35：5TH axis 3rd reference point return end signal ZP34：4TH axis 3rd reference point return end signal ZP33：Y axis 3rd reference point return end signal ZP32：Z axis 3rd reference point return end signal ZP31：X axis 3rd reference point return end signal

F059				ZP45	ZP44	ZP43	ZP42	ZP41

ZP45：5TH axis 4th reference point return end signal
ZP44：4TH axis 4th reference point return end signal ZP43：Y axis 4th reference point return end signal ZP42：Z axis 4th reference point return end signal ZP41：X axis 4th reference point return endsignal

F061								ESEND

KYEND：Required parts to arrive signal

F008							SCHK	

SCHK：Checking grammar signal

F016					ZP4	ZP3	ZP2	ZP1

ZP1：X axis return zero point end signal ZP2：Y axis return zero point end signal ZP3：Z axis return zero point end signal ZP4：4TH axis return zero point end signal

F022	SB07	SB06	SB05	SB04	SB03	SB02	SB01	SB00

SB07 ： Spindle spped code signal SB07
SB06 ： Spindle spped code signal SB06
SB05 ： Spindle spped code signal SB05
SB04 ： Spindle spped code signal SB04
SB03 ： Spindle spped code signal SB03
SB02 ： Spindle spped code signal SB02
SB01 ： Spindle spped code signal SB01
SB00：Spindle spped code signal SB00

5.2.5 G address(PLC→CNC)

G004					FIN			

FIN: MST function end signal

G005	LEDT	AFL		LAXIS				

LEDT: Edit lock signal
AFL: MST lock signal
LAXIS: All axis interlock sianal

G006		SKIPP		OVC		ABSM	MSTOP	SRN

SRN: Program restart signal
ABSM: Manual absolute signal
OVC: Feedrate override cancel signal
SKIPP: Skip signal
MSTOP: Selective stop signal

G007						ST		

ST: Cycle start signal

G008			SP	ESP				

ESP: Emergency stop signal
SP: Feed hold signal

G009						M12	M32	COOL

M12: 0:/1 Spindle tool clamp/loose signal
M32: Lubricating signal
COOL: Cooling signal

G0010	JV07	JV06	JV05	JV04	JV03	JV02	JV01	JV00

JV00: JOG override signal JV00
JV01: JOG override signal JV01
JV02: JOG override signal JV02
JV03: JOG override signal JV03
JV04: JOG override signal JV04
JV05: JOG override signal JV05
JV06: JOG override signal JV06
JV07: JOG override signal JV07

G0011	JV15	JV14	JV13	JV12	JV11	JV10	JV09	JV08

JV08: JOG override signal JV08
JV09: JOG override signal JV09
JV10: JOG override signal JV10
JV11: JOG override signal JV11
JV12: JOG override signal JV12
JV13: JOG override signal JV13
JV14: JOG override signal JV14
JV15: JOG override signal JV15

G0012	FV07	FV06	FV05	FV04	FV03	FV02	FV01	FV00

FV00: Feedrate override signal FV00
FV01: Feedrate override signal FV01
FV02: Feedrate override signal FV02
FV03: Feedrate override signal FV03
FV04: Feedrate override signal FV04
FV05: Feedrate override signal FV05
FV06: Feedrate override signal FV06
FV07: Feedrate override signal FV07

G0014	RV8	RV7	RV6	RV5	RV4	RV3	RV2	RV1

RV1: Rapid feedrate override signal RV1 RV2:
Rapid feedrate override signal RV2 RV3:
Rapid feedrate override signal RV3 RV4:
Rapid feedrate override signal RV4 RV5:
Rapid feedrate override signal RV5 RV6:
Rapid feedrate override signal RV6 RV7:
Rapid feedrate override signal RV7 RV8:
Rapid feedrate override signal RV8

G016				SAR				

SAR：Spindle speed arrival signal

G017					DECA	DECY	DECZ	DECX

DECA: 4TH axis back to zero deceleration signal
DECY: Y axis back to zero deceleration signal DECZ:
Z axis back to zero deceleration signal DECX:
X axis back to zero deceleration signal

G018					H4TH	HY	HZ	HX

H4TH: 4TH axis MPG feed selection signal HY:
Y axis MPG feed selection signal
HZ: Z axis MPG feed selection signal HX:
X axis MPG feed selection signal

G019	RT		MP2	MP1				

RT: Manual rapid feed selection signal
MP2: MPG override signal MP2
MP1: MPG override signal MP1

G021	SOV7	SOV6	SOV5	SOV4	SOV3	SOV2	SOV1	SOV0

SOV7: Spindle override signal SOV7
SOV6: Spindle override signal SOV6
SOV5: Spindle override signal SOV5
SOV4: Spindle override signal SOV4
SOV3: Spindle override signal SOV3
SOV2: Spindle override signal SOV2
SOV1: Spindle override signal SOV1
SOV0: Spindle override signal SOV0

G022	R08I	R07I	R06I	R05I	R04I	R03I	R02I	R01I

R01I: Spindle motor speed code signal R01I R02I:
Spindle motor speed code signal R02I R03I:
Spindle motor speed code signal R03I R04I:
Spindle motor speed code signal R04I R05I:
Spindle motor speed code signal R05I R06I:
Spindle motor speed code signal R06I R07I:
Spindle motor speed code signal R07I R08I:
Spindle motor speed code signal R08I

G023	SIND	SGN			R12I	R11I	R10I	R09I

R09I: Spindle motor speed code signal R09I R10I:
Spindle motor speed code signal R10I R11I:
Spindle motor speed code signal R11I R12I:
Spindle motor speed code signal R12I SGN:
Spindle motor code polarity selection signal
SIND: Spindle motor speed code selection signal

G024	MRDYA							

MRDYA：Machine ready signal

G025			SRVB	SFRB				

SRVB：Spindle rotate(CCW) signal
SFRB: Spindle rotate(CW) signal

G026	CON							

CON：Cs contour control switch signal

G027					+J4	+J3	+J2	+J1

+J4:Feed axis and direction selection signal+J4
+J3:Feed axis and direction selection signal+J3
+J2:Feed axis and direction selection signal+J2
+J1: Feed axis and direction selection signal +J1

G028					-J4	-J3	-J2	-J1

-J4:Feed axis and direction selection signal-J4
-J3:Feed axis and direction selection signal-J3
-J2:Feed axis and direction selection signal-J2
-J1: Feed axis and direction selection signal -J1

G030					+L4	+L3	+L2	+L1

+L4: Axis overtravel signal+L4
+L3: Axis overtravel signal+L3
+L2: Axis overtravel signal+L2
+L1: Axis overtravel signal+L1

G031					-L4	-L3	-L2	-L1

-L4: Axis overtravel signal-L4
-L3: Axis overtravel signal-L3
-L2: Axis overtravel signal-L2
-L1: Axis overtravel signal-L1

G036	BEUCL	BECLP						SPD

BEUCL: Indexing table release signal
BECLP: Indexing table clamp signal SPD:
Spindle point function signal

G037	NT07	NT06	NT05	NT04	NT03	NT02	NT01	NT00

NT07: Current tool No. NT07
NT06: Current tool No. NT06
NT05: Current tool No. NT05
NT04: Current tool No. NT04
NT03: Current tool No. NT03
NT02: Current tool No. NT02
NT01: Current tool No. NT01
NT00: Current tool No. NT00

G043	ZRN		DNC1			MD4	MD2	MD1

ZRN：Cueerntworkmodeselection4 DNC1：

DNC run selction signal MD4：Cueerntwork

modeselection 3 MD2：Cueerntworkmode

selection 2 MD1：Cueerntworkmodeselection

G044	HDT						MLK	BDT

HDT：Manual change tool by sequence signal

MLK：Machine locked signal（PLC → CNC）

BDT：Program skip signal(PLC → CNC)

G046	DRN				KEY1		SBK	

DRN：Dry run signal KEY1：

Storage protect signal

SBK：Signal program senment signal(PLC → CNC)

G048							GR2	GR1

GR2：Gear selection signal2

GR1：Gear selection signal1

G053	CD2	SMZ						

CDZ：Chamfer signal

SMZ：Error check signal

G054	UI07	UI06	UI05	UI04	UI03	UI02	UI01	UI00

UI07 ： Macro input signal UI07

UI06 ： Macro input signal UI06

UI05 ： Macro input signal UI05

UI04 ： Macro input signal UI04

UI03 ： Macro input signal UI03

UI02 ： Macro input signal UI02

UI01 ： Macro input signal UI01

UI00：MacroinputsignalUI00

5.2.6 Address A (message display requiery signal, defined by standard PLC ladders)

A0002.0	1216	Alarm of safe door not closed
A0002.1	1217	Alarm of chuck low pressure
A0002.3	1219	Chuck released unallowed in spindle running
A0002.4	1220	Clamping in-position signal inactive alarm in spindle running
A0002.5	1221	Spindle start unallowed if chuck clamping in-position signal inactive
A0002.6	1222	Spindle start unallowed for chuck releasing
A0004.0	1232	Illegal M code
A0004.1	1233	Spindle jog disabled in non-analog spindle mode
A0004.2	1234	M03, M04 designation error
A0004.4	1236	Spindle gear change time is too long
A0004.5	1237	Spindle speed/position control switch time is too long
A0005.1	1241	Alarm for the abnormal spindle servo or frequency converter for abnormality
A0007.1	1257	Safety door has been opened
A0007.3	1259	Alarm for the tool pot unclocked

CHAPTER 6 MEMORIZING PITCH ERROR COMPENSATION

6.1 Function description

There are more or less precision errors in the pitch of machine axes lead screw, and it will definitely affect the parts machining precision. This TAC2000 CNC system has the memorizing pitch error offset function that it can accurately compensate the pitch error of the lead screw.

6.2 Specification

1) The set offset amount is concerned with the offset origin, offset intervals etc.;

2) Pitch error offset value is get by searching the table about machine coordinates and pitch error compensation origin;
3) Points to be compensated: 256 points for each axis
4) Axis compensated: X, Y, Z ,4th,5th axis
5) Compensation range: 0～±99 x least input increment
6) Compensation range interval: 1～9999.9999;
7) Compensation amount of compensation pointN(N=0,1,2,3,…255) is determined by the mechanical error between point N and point N-1;
8) The setting is the same as the CNC parameters input, see **Volume IIOperation.**

6.3 Parameter setting

6.3.1 Pitch compensation

Bit parameter

0	0	3

		SCRW					

Bit5=1: Pitch error offset active;
Bit5=0: Pitch error offset inactive;

6.3.2 Pitch error compensation origin

A position which the pitch error offset starts from in the offset list, which is determined from the machine zero, is called pitch error offset origin (reference point). This position may be set from 0 to 255 in each axis by data parameter No.180~No.184, depending on the mechanical requirement.

Data parameter

1	8	0	X axis pitch error compensation origin position No.
1	8	1	Y axis pitch error compensation origin position No.
1	8	2	Z axis pitch error compensation origin position No.
1	8	3	4th axis pitch error compensation origin position No.
1	8	4	5th axis pitch error compensation origin position No.

6.3.3 Offset interval

Pitch offset interval: No.190~No.194;

Input unit: Metric machine:mm; Inch machine:inch Setting range: 1～9999.9999

State parameter

1	9	0	Pitch error compensation interval of X axis
1	9	1	Pitch error compensation interval of Y axis
1	9	2	Pitch error compensation interval of Z axis
1	9	3	Pitch error compensation interval of 4th axis
			Pitch error compensation interval of 5th axis

6.3.4 Offset value

Every axes pitch offset values is set according to the parameter No. in the following table. The offset value is input by mm(metric machine) or inch(inch machine), which is not irrelevant to diameter or radius programming.

Offset No.	X	Z	Y
000	…	…	…
001	5	-2	3
002	-3	4	-1
…	…	…	…
255	…	…	…

6.4 Notes of offset setting

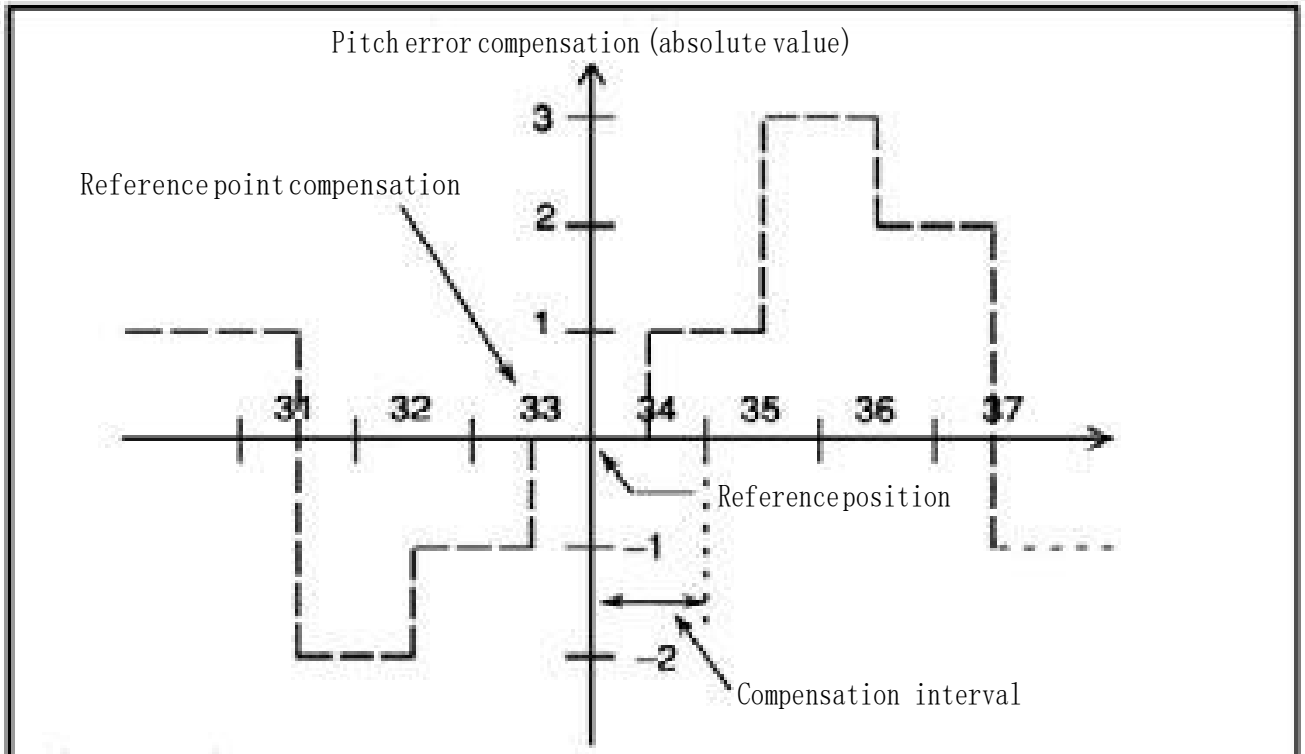
①The setting and alteration of pitch offset can only be done at the authority of password level 2.

②After the parameter of pitch offset is set, only the machine zero is returned could the offset be done.

6.5 Setting examples of offset parameters

①Data parameter No.180(pitch error origin) =33, Data parameter No.185 (offset interval)=10.000mm

When the pitch error origin is set to 33:



Compensation point	31	32	33	34	35	36	37
Compensation value	+3	-1	-1	+1	+2	-1	-3

Appendix

AppendixI Listofalarm

1、CNCAlarm

NO.	Content	Remark
0000	Please power off!	
0001	Fail opening file	
0002	Edited data exceeding limit	
0003	Copy or rename program No. existing.	
0004	No searched address	
0005	No data behind address	
0006	illegal minus	
0007	illegal decimal point	
0008	File too capacity not be loaded.	
0009	Input illegal address	
0010	Incorrect G codes	
0011	No feedrate instruction	
0012	Insufficient disc.	
0013	Too many Files	
0014	Not command G95, spindle not support	
0015	Command too axes	
0016	Cur pitch error comp. point too many!	
0017	No right to alert!	
0018	Not permit to alert	
0019	Cann't use scale!	
0020	Exceed radius tolerance	
0021	Command illegal plane axis	
0022	R and IJK is 0 in arc	
0023	IJK and R specified simultaneously in arc	
0024	Screw interpolation chamfer is 0	
0025	G12 cann't specify with other G code	
0026	Format not supported by system.	
0027	Offset can't share a block with G92.	
0028	illegal plane selection	

0029	illegal offset value	
0030	illegal comp. No.	
0031	illegal P commanded in G10	
0032	illegal comp. value in G10	
0033	No intersection in C	
0034	Cann't start or cancel tool comp. in arc	
0035	Not cancel C offset before M99	
0036	Not command G31	
0037	Not change plane in C	
0038	interference in arc block	
0039	Tool nose position error in C	
0040	Workpiece coordinate changed in comp. C	
0041	interference in C	
0042	Over 10 non-traverse instructions in comp. C	
0043	Unauthorized	
0044	No permitting G27~G30 in canned cycle	
0045	Address Q not found (G73/G83)	
0046	illegal reference point return instruction	
0047	Executing machine zero return before it	
0048	Z plane should be higher than R	
0049	Z plane should be lower than R	
0050	Should traverse pos before chang fixed cycle	
0051	Mistaken traverse after chamfer	
0052	Mirror disabled in grooving cycle	
0053	Over address instruction	
0054	DNC carry setting error	
0055	Mistaken traversing value in chamfer or R	
0056	M99 can't share a block with macro	
0057	Savefailed.	
0058	Not found end point	
0059	Not found program No.	
0060	Not found sequence No.	
0061	X axis not in reference point	

0062	Y axis not in reference point	
0063	Z axis not in reference point	
0064	4TH axis not at reference point	
0065	5TH axis not at reference point	
0066	Cancel fixed cycle before executing G10	
0067	SettingformatnotsupportedbyG10.	
0068	PARA SWITCH hasn't turned on	
0069	Need close "U" disk interface as cnc runing	
0070	Memory capacity insufficiency	
0071	Not found data	
0072	Over program quantities	
0073	Program number used	
0074	illegal program number	
0075	Protection	
0076	Address P no defined	
0077	Mistaken subprogram embedding	
0078	Not found sequence number	
0079	Systemexpired.	
0080	Improper input data	
0082	Command H in G37	
0083	illegal axis instruction in G37	
0084	Key overtime or short circuit	
0085	Communication error	
0087	X axis reference point return unfinished	
0088	Z axis reference point return unfinished	
0089	Y axis reference point return unfinished	
0090	4TH axis reference point return unfinished	
0091	5TH axis reference point return unfinished	
0092	Axis not in reference point	
0094	Not permit P type (coordinates)	
0095	Not permit P type (EXT OFS CHG)	
0096	Not permit P type (WRK OFS CHG)	
0097	Not permit P type(automatically execute)	

0098	Found G28 in sequence return	
0099	Not permit executing MDI after searching	
0100	Valid parameter write	
0101	Power-off memory data confused	
0110	Dataoverflow.	
0111	PC data overflow	
0112	Divided by zero	
0113	Mistaken instruction	
0114	Mistaken G39 format	
0115	illegal variable	
0116	Write protection variable	
0118	Mistaken big brackets embed	
0119	M00~M02,M06,M98,M99,M30 can't at the same block with other block	
0122	Fourfold macro mold-calling	
0123	Not use macro instruction in DNC	
0124	Program illegal completion	
0125	Mistaken macro program format	
0126	illegal cycle number	
0127	NC & macro instruction in the same block	
0128	Sequence number of illegal macro instruction	
0129	illegal independent variable address	
0130	illegal axis operation	
0131	Over external alarm information	
0132	Not found alarm number	
0133	System not support axis instruction	
0134	Axis more than 3 can not use rigid tapping	
0135	illegal angle instruction	
0136	illegal axis instruction	
0139	Can't change PLC control axis	
0142	illegal proportional rate	
0143	Scaling motion data overflow	
0144	illegal plane selection	
0148	illegal data setting	

0149	Format error in G10L3	
0150	illegal tool group No.	
0151	Not found tool group No.	
0152	Tool data not memorize	
0153	Not cancle C before changing tool	
0154	Tool in unused tool life	
0155	Illegal T code in M06	
0156	Not found P/L instruction	
0157	Over tool group	
0158	illegal tool life data	
0159	Tool data setting unfinished	
0160	Arc only use R prg in polar coordinates mode	
0161	Not execute the instr in polar coordinates	
0162	Have used G70~G76 instructions in MDI mode	
0163	Not execute the instruction in rotation mode	
0164	Not execute the instruction in scaling mode	
0165	Specify the instruction in sole block	
0166	Axis not specified in reference point return	
0167	Coordinates in intermediate point too big	
0168	Min. dwell time should smaller than max.4	
0170	Not cancel comp. in entering or Esc subprg	
0172	P not int or less than 0 in calling subprg	
0173	Subprogram calling times less than 9999	
0175	G17 executed only in canned cycle	
0176	Spindle rotate speed not set	
0177	Not support spindle oriented function	
0178	Spindle rotate speed not set before canned cycle	
0181	illegal M code	
0182	illegal S code	
0183	illegal T code	
0184	Selected tool exceeding limit	
0185	L too small	
0186	L too large	

0187	Tool radius too large	
0188	U too large	
0189	U smaller than tool radius	
0190	V too small or V has not defined	
0191	W too small or W has not defined	
0192	Q too small or Q has not defined	
0193	I has not define or I is zero	
0194	J has not define or J is zero	
0195	D has not define or D is zero	
0198	Illegal axis selection	
0199	Macro instruction not defined	
0200	illegal S mode instruction	
0201	Not found feedrate in rigid tapping	
0202	Position LSI overflow	
0203	Program error in rigid tapping	
0204	Illegal axis operation	
0205	Rigid mode DI signal closed	
0206	Not change plane (rigid tapping)	
0207	Tapping data error	
0208	Cann'texe.theinstructioninG10.	
0212	illegal plane selection	
0224	Reference point return	
0231	illegal format in G10 L50 or L51	
0232	Commanded spiral interpolation axes too many	
0233	Device busy	
0235	Error completion	
0236	Program restart parameter error	
0237	No decimal point	
0238	Address repetition error	
0239	Parameter 0	
0240	No permitting G41/G42 in MDI	
0241	MPG abnormal	

0251	Emergency stop alarm
0260	Name of axis is repeated.Please alter parameters NO.225~227
0451	Xaxisdriveralarm.
0452	Zaxisdriveralarm.
0453	Yaxisdriveralarm.
0454	4THaxisdriveralarm.
0455	5THaxisdriveralarm.
0456	Spindledriveralarm.
0500	Software limit overtravel:-X
0501	Software limit overtravel:+X
0502	Software limit overtravel:-Y
0503	Software limit overtravel:+Y
0504	Software limit overtravel:-Z
0505	Software limit overtravel:+Z
0506	Software limit overtravel:-4Th
0507	Software limit overtravel:+4Th
0508	Software limit overtravel:-5Th
0509	Software limit overtravel:+5Th
0510	Hardware limit overtravel:-X
0511	Hardware limit overtravel:+X
0512	Hardware limit overtravel:-Y
0513	Hardware limit overtravel:+Y
0514	Hardware limit overtravel:-Z
0515	Hardware limit overtravel:+Z
0516	Hardware limit overtravel:-4TH
0517	Hardware limit overtravel:+4TH
0518	Hardwarelimitovertravel:-5TH

0519	Hardwarelimitovertravel:+5TH
0740	Rigidtappingalarm:overproof
0741	Rigidtappingalarm:overproof
0742	Rigid tapping alarm:LSI overflow
0751	Checkthefirstspindle alarm(AL-XX)
0754	Abnormal torque alarm
1001	Address of relay or coil not set
1002	Input code inexistence
1003	COM/COMEusedbymistake.
1004	ladderexceedingmax.linageorstep.
1005	Error inEND1/END2.
1006	illegaloutputinNET.
1007	Hardware failure or system interrupt error causes PLC to communicate
1008	Notconnectedcorrectly.
1009	Networkhorizonnotconnected.
1010	Networkmissingforpower-offineditladder.
1011	Addressdatanotinputcorrectly.
1012	Symbolundefinedordataexceedinglimit.
1013	Definedillegalcharacters.
1014	CTRadressisrepeated.
1015	JMP/LBLdealerrorexceedingitscapacity.
1016	Networkstructisincomplete.
1017	Networkstructisn'tsupported.
1019	TMRaddressrepeat.
1020	Noparameterinfunctioninstruction.
1021	PLC execution timeout, the system automatically stops PLC.
1022	Functioninstructionnamelost.
1023	Functionaladdressorconstantoverflow.
1024	Unnecessaryrelayorcoilexist.
1025	Functioninstructionnotcorrectlyoutput.
1026	Linenumberofnetworkconnectionoverflow.
1027	Onesymbolnamedefinedinanotherplace.
1028	Ladderformaterror.

1029	Ladderbeingusedlost.	
1030	IncorrectverticallineinNET.	
1031	Datafull,reducingCODinstr.datacapacity.	
1032	Firstgradeofladdertoobig.	
1033	SFTinstructionexceedingmax.capacity.	
1034	DIFU/DIFD used mistakenly.	
1035	Current opened ladder convert failed	
1036	PLC emergency stop alarm	
1037	Opened and data para setting ladder isn't same	
1039	Instruction or network not within range	
1040	CALL/SP/SPEusedmistakenly.	
1041	Horizonallineparallelstonodenet.	
1042	PLC parameter file not loaded	

Appendix II Operation list

Type	Function	Operation	Opera -tion mode	Display window	Password level	Program switch	Para -meter switch	Remark
Clear	X relative coordinate clear	× , CANCEL		Relative coordinate				Volume II Section 1.3.1

Type	Function	Operation	Opera -tion mode	Display window	Password level	Program switch	Para -meter switch	Remark
	Y relative coordinate clear	Y & , CANCEL		Relative coordinate				
	Z relative coordinate clear	: , CANCEL		Relative coordinate				
	Machining numbers clear	CANCEL + N #		relative coordinate or absolute coordinate				Volume II Section 1.3.1
	Cutting time clear	CANCEL + M [		relative coordinate or absolute coordinate				Volume II Section 1.3.1
Data setting	state parameter	parameter value, DATA INPUT	MDI mode	state parameter	2-level, 3-level,		ON	Volume II Section 10.1.3
	Data parameter	parameter value, DATA INPUT	MDI mode	data parameter	2-level, 3-level,		ON	
	X pitch parameter input	X ,compensation value, DATA INPUT	MDI mode	pitch compensation parameter	2-level		ON	Volume II Section 10.1.3
	Y pitch parameter input	Y & ,compensation value, DATA INPUT	MDI mode	pitch compensation parameter	2-level		ON	Volume II Section 10.1.3
	Z pitch parameter input	Z . ,compensation value, DATA INPUT	MDI mode	pitch compensation parameter	2-level		ON	Volume II Section 10.1.3
	Macro variable	macro variable value, DATA INPUT		Macro variable	2-level, 3-level, 4-level			Volume II Section 1.3.3
	Tool offset	compensation value, DATA INPUT		tool offset	2-level, 3-level, 4-level			Volume II Section 7.4.2
Search	Search downward from the cursor's current position	character, ⇩	EDIT mode	program content	2-level, 3-level, 4-level	ON		Volume II Section 6.1.3
	Search upward from the cursor's current position	character, ⇧	EDIT mode	program content	2-level, 3-level, 4-level	ON		Volume II Section 6.1.3

Type	Function	Operation	Opera-tion mode	Display window	Password level	Program switch	Para-meter switch	Remark
	Search downward from the current program	O, ⇩	EDIT mode or AUTO mode	program content program contents or program state	2-level,3-level, 4-level			Volume II Section 6.4.1
	Search upward from the current program	O, ⇧			2-level,3-level, 4-level			Volume II Section 6.4.1
	Search specified program	O ,program name,			2-level,3-level, 4-level			Volume II Section 6.4.2
	Search bit parameters, data parameters or pitch compensation parameters	P, parameter No., DATA INPUT		Corresponding page of data				Volume II Section 10.1.3
	PLC state, PLC data search	P,address No., DATA INPUT		PLC state PLC data				Volume II Section 1.3.7
Delete	character deletion at the cursor	DELETE	EDIT mode	program content	2-level,3-level, 4-level	ON		Volume II Section 6.1.6
		CANCEL	EDIT mode	program content	2-level,3-level, 4-level	ON		
	Single block deletion	Move the cursor to the head line, DELETE	EDIT mode	program content	2-level,3-level, 4-level	ON		Block No. in block, Volume II Section 6.1.7
	Blocks deletion	CHANGE, N# sequence No. , DELETE	EDIT mode	program content	2-level,3-level, 4-level	ON		Volume II Section 6.1.8
	Segment deletion	CHANGE ,character, DELETE	EDIT mode	program content	2-level,3-level, 4-level	ON		Volume II Section 6.1.9
	Single program deletion	O ,program name, DELETE	EDIT mode	program content	2-level,3-level, 4-level	ON		Volume II Section 6.3.1
	All programs deletion	O , -+ 999, DELETE	EDIT mode	program content	2-level,3-level, 4-level	ON		Volume II Section 6.3.2

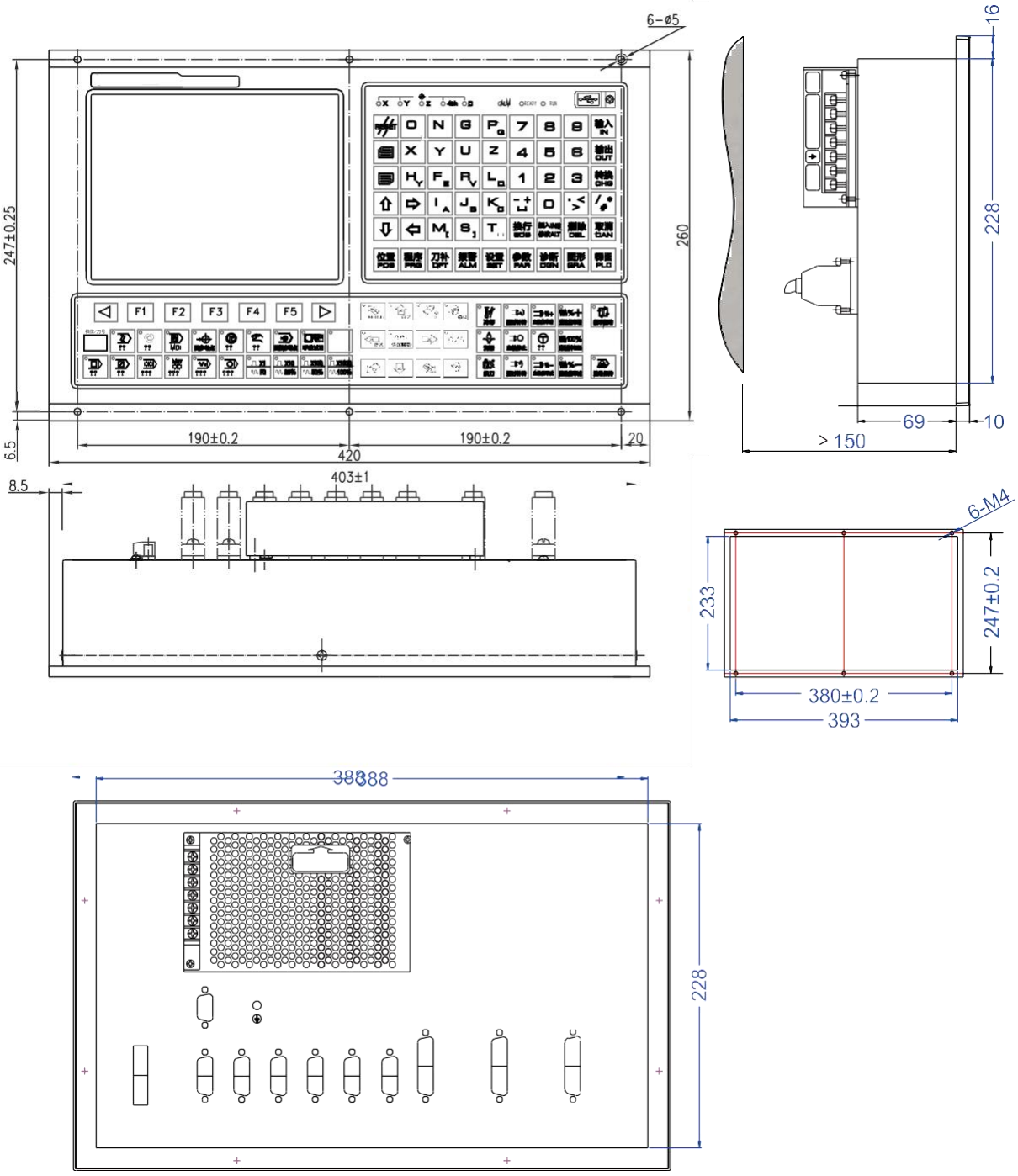
Type	Function	Operation	Opera -tion mode	Display window	Password level	Program switch	Para -meter switch	Remark
Rename	Program rename	O L ,program name, ALTER	EDIT mode	program content	2-level,3-level, 4-level	ON		
Copy	Program copy	O L ,program name, CHANGE	EDIT mode	program content	2-level,3-level, 4-level	ON		
Switch setting	Parameter switch ON	D L		Switch setting	2-level,3-level			Volume II Section10.1.1
	Program switch ON	D L		Switch setting	2-level,3-level, 4-level			
	Automatic sequence No. ON	D L		Switch setting				
	Parameter switch OFF	U W		Switch setting	2-level,3-level			
	Program switch OFF	U W		Switch setting	2-level,3-level, 4-level			
	Automatic sequence No. OFF	U W		Switch setting				

Note 1: “,” in “Operation” indicates that the two operations are successive, “+” indicates that the two operations are executed at the same time.

Example: “ X , CANCEL ” indicates that we firstly press X and then press CANCEL ; “ CANCEL + N # ” indicates these two keys are pressed simultaneously.

Note 2: The blanks in Operation Mode, Display Window, Password Level, Program Switch and Parameter Switch column indicate that the corresponding switches are not related to their items correspondingly.

Appendix III TAC2000 contour dimension



Appendix IV Additional panel dimensions