GSK983M Milling CNC System

User Manual

(Volume I: Specifications and Programming)

广州数控设备有限公司

GSK CNC EQUIPMENT CO., LTD.

✧ 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 works of the system due to the length limit of the manual, specific operations of the product and other causes. Therefore, the operations not specified herein may be considered impractical or unavailable.

Company Profile

GSK, GSK CNC Equipment Co,. Ltd, is the largest CNC system production and marketing enterprise in China at present. It is the Numerical Control industrial base of South China, and the undertaking enterprise of the national 863 main project Industrialization Support Technology for Medium Numerical Control System. It is also one of the 20 basic equipment manufacture enterprises in Guangdong province. It has been taking up the research and development, design and the manufacture of machine CNC system (CNC device, drive unit and servo motor) in recent 10 years. Now it has developed into a large high-tech enterprise integrated with technology, education, industry and trade by enhancing the popularization and trade of CNC machine tools. There are more than 1400 staffs in this company that involves 4 doctors, more than 50 graduate students and 500 engineers; more than 50 among these staffs are qualified with senior engineer post titles. The high performance-cost ratio products of GSK are popularized in China and Southeast Asia. And the market occupation of GSK's product dominates the first and the turnout and sale ranks the top for successive 7 years in domestic market for the same product from the year 2000 to 2006, which makes GSK the largest CNC manufacture base throughout China.

Field technical support services

Field support services are available when you encounter a problem insolvable through telephone. GSK CNC Equipment Co. Ltd will designate a technical support engineer to your place to solve technical problems for you.

Chinese version of all technical documents in Chinese and English languages is regarded as final.

Foreword

Dear user,

We are really grateful for your patronage and purchase of GSK983M milling CNC system made by GSK CNC Equipment Co., Ltd.

This manual consists of two volumes. Volume I mainly describes the specifications and programming of the system while Volume II operations, all codes, parameters, I/O interfaces and other appendices.

⚠ This system can only be operated by authorized and qualified personnel as improper operations may cause accidents. Please carefully read this user manual before usage.

All specifications and designs herein are subject to change without further notice.

We are full of heartfelt gratitude to you for supporting us in the use of GSK's products.

CONTENTS

1. GENERAL ... *1*

1.1 Overview ... 1

1.2 Introduction to the Manual ... 1

2. SPECIFICATIONS ... *2*

3. PROGRAMMING ... *13*

3.1 What Is Programming? ... 13

3.2 Program Make-up ... 13

3.2.1 Block ... 14
3.2.2 Program word ... 14
3.2.3 Input format ... 16
3.2.4 Decimal programming ... 18
3.2.5 Maximum instruction value ... 19
3.2.6 Program number ... 20
3.2.7 Sequence number ... 21
3.2.8 To skip over an optional block ... 21

3.3 Dimension Word ... 23

3.3.1 Controllable axis ... 23
3.3.2 Set unit ... 25
3.3.2.1 Minimum set unit and minimum travel unit ... 25
3.3.2.2 Input unit×10 ... 26
3.3.3 Maximum travel ... 26
3.3.4 Program origin and coordinate system ... 26
3.3.5 Coordinate system and origin of processing ... 27
3.3.6 Workpiece coordinate system ... 27
3.3.7 Reference (position) point ... 28
3.3.8 Absolute value instruction and incremental value instruction ... 29

3.4 Feed Function (F function) ... 30

3.4.1 Rapid traverse (positioning) function ... 30
3.4.2 Cutting feedrate ... 30
3.4.3 To reduce feedrate to 1/10 ... 31
3.4.4 Synchronous feed (feed per rotation) ... 31
3.4.5 F1 digit feed ... 32
3.4.6 Automatic acceleration and deceleration ... 33
3.4.7 Automatic angle adjustment ... 34
3.4.7.1 Automatic adjustment of inner angle ... 34
3.4.7.2 Switch of inner arc cutting feedrate ... 37

3.5 Preparation Function (G function) ... 38

3.5.1 Selection of planes (G17, G18, G19) ... 40
3.5.2 Positioning (G00) ... 41

3.5.3 Unidirectional positioning (G60) 42
3.5.4 Linear interpolation (G01) 42
3.5.5 Arc interpolation (G02, G03) 44
3.5.5.1 Arc interpolation without any additional axis 44
3.5.5.2 Arc interpolation with an additional axis 48
3.5.6 Sine-curve interpolation 48
3.5.7 Thread cutting (G33) 49
3.5.8 Automatic return to reference point (reference positions G27～G30) 51
3.5.8.1 Check of return to reference point (G27) 51
3.5.8.2 Automatic return to reference point (G28) 52
3.5.8.3 Automatic return from reference point (G29) 53
3.5.8.4 Return to the 2nd, 3rd or 4th reference point (G30) 55
3.5.9 Dwell (G04) 55
3.5.10 Accurate stop detection (G09) 56
3.5.11 Accurate stop detecting mode (G61) and cutting mode (G64) 56
3.5.12 Coordinate system setting (G92) 56
3.5.13 Workpiece coordinate system (G54～G59) 57
3.5.14 To change workpiece coordinate system by program instruction 59
3.5.15 Automatic setting of a coordinate system 59
3.5.16 To switch between Inch and metric systems (G20, G21) 59
3.5.17 Storage travel limit (G22, G23) 60
3.5.18 Skip function (G31) 63

3.6 Compensation 65
3.6.1 Tool length compensation (G43, G44, G49) 65
3.6.2 Tool offset (G45～G48) 67
3.6.3 Tool radius compensation (G40～G42) 74
3.6.3.1 Tool radius compensation function 74
3.6.3.2 Offset (D codes) 74
3.6.3.3 Offset vector 75
3.6.3.4 Plane selection and vector 75
3.6.3.5 G40, G41 and G42 75
3.6.3.6 Details about tool radius compensation C 77
3.6.4 D and H functions 109
3.6.5 External tool offset 110
3.6.6 To input offset through program (G10) 110
3.6.7 Zooming function (G50, G51) 110

3.7 Cycle Machining Function 113
3.7.1 External moving function 113
3.7.2 Fixed cycles (G73, G74, G76 and G80～G89) 114
3.7.2.1 Repeating a fixed cycle 126
3.7.3 Specifying an origin and Point R in a fixed cycle (G98, G99) 132

3.8 Spindle Function (S function), Tool Function (T function), Miscellaneous Function (M function) and Secondary Miscellaneous Function (B function) 133
3.8.1 Spindle function (S function) 134
3.8.1.1 S 2 digits 134

3.8.1.2 S 4 digits 134
3.8.2 Constant surface speed control 134
3.8.2.1 Instructed methods 135
3.8.2.2 Spindle speed 135
3.8.2.3 Restraint of maximum rotational speed of spindle 135
3.8.2.4 Rapid traverse (positioning) (G00) 136
3.8.3 Tool function (T function) 137
3.8.4 Miscellaneous function (M function) 137
3.8.5 Secondary miscellaneous function (B function) 138

3.9 Subprogram 138
3.9.1 Creation of a subprogram 139
3.9.2 Execution of a subprogram 139
3.9.3 Special methods of application 140

3.10 User Macro 142
3.10.1 General 142
3.10.2 Variables 144
3.10.2.1 Indication of a variable 144
3.10.2.2 Introduction of variables 144
3.10.2.3 Undefined variables 145
3.10.2.4 Display and setting of variable 146
3.10.3 Types of variables 146
3.10.3.1 Local variables # 1～# 33 147
3.10.3.2 Common variables #100～ #149 , #500～ #509 147
3.10.3.3 System variables (for user Macro B) 147
3.10.4 Operation instructions 160
3.10.4.1 Definition and substitution of variable 160
3.10.4.2 Additive operation 160
3.10.4.3 Multiply operation (Macro B option) 160
3.10.4.4 Function (Macro B option) 160
3.10.4.5 Hybrid operation 161
3.10.4.6 Changing operational order using [ ] 162
3.10.4.7 Accuracy 162
3.10.4.8 Cautions regarding deterioration of precision 162
3.10.5 Control instruction 163
3.10.5.1 Branch (GOTO) 163
3.10.5.2 Repeat (how to select Macro B) 165
3.10.6 Creation and storage of user macro body 169
3.10.6.1 Creation of user macro body 169
3.10.6.2 Storage of user macro body 170
3.10.6.3 Macro statement and NC statement 170
3.10.7 Macro call instruction 173
3.10.7.1 Simple calling 173
3.10.7.2 Modal calling 180
3.10.7.3 Multiple calling 181
3.10.7.4 Multiple modal calling 181
3.10.7.5 Calling a macro with G codes 182

3.10.7.6 Calling a subprogram with an M code 183
3.10.7.7 Calling macros with M codes 184
3.10.7.8 Calling a subprogram with a T code 185
3.10.7.9 The position of the decimal point of an independent variable 185
3.10.7.10 The difference between M98 (calling a subprogram) and G65 (calling a macro body) 186
3.10.7.11 The nesting and local variables of user macro 186
3.10.8 The relationship with other functions 187
3.10.9 Special codes and words used in user programs 189
3.10.10 Restrictions 190
3.10.11 Descriptions for P/S alarm 191
3.10.12 Examples of user macros 191
3.10.12.1 Groove machining 191
3.10.13 External output instructions 193
3.10.13.1 OPEN instruction: POPEN 194
3.10.13.2 Data output instruction BPRNT DPRNT 194
3.10.13.3 Close instruction PCLOS 195
3.10.13.4 Necessary settings for using the function 195
3.10.13.5 Cautions 195
3.10.14 Macro interruption function (Macro B) 196

3.11 Tool Life Management 196
3.11.1 Setting of tool groups 196
3.11.2 Setting in machining processes 198
3.11.3 Performance of tool life administration 200
3.11.3.1 Tool life calculation 200
3.11.3.2 Tool change signal and tool change reset signal 200
3.11.3.3 New tool selection signal 201
3.11.3.4 Tool skip signal 201
3.11.4 The display and input of tool data 201
3.11.4.1 The display and modification of tool group number 201
3.11.4.2 The display of tool life data during the execution of machine program 201
3.11.4.3 Presetting tool life counter 202
3.11.5 Other cautions 203

3.12 The Indexing Function of Indexing Worktable 203
3.12.1 Instructing methods 203
3.12.1.1 Input unit 203
3.12.1.2 Absolute / incremental instruction 203
3.12.1.3 Concurrently controlled axes 204
3.12.2 Minimum travel unit: 0.001 degree/pulse 204
3.12.3 Feedrate 204
3.12.4 The clamping and release of indexing worktable 204
3.12.5 Jog/step/handwheel 205
3.12.6 Other cautions 205

1. GENERAL

1.1 Overview

As a high-accuracy and high-performance closed-loop CNC system with firmware, **GSK983M** milling CNC system (hereinafter called “System”) was newly developed and launched by GSK CNC Equipment Co., Ltd. to satisfy the demands of NC market and intended for CNC milling machine and processing center. By employing a high-speed microprocessor, a general large scale integrated circuit (LSI), a semiconductor memory and up-to-date storage elements, the reliability as well as performance-to-price ratio of its control circuit are greatly improved, thereby facilitating the reliability, functions/ performance to price ratio to a large extent.

With a widely applicable advanced servo, GSK983M milling CNC system uses a high-duty pulse encoder as its detecting element, thereby composing a closed-loop CNC system.

This operating manual describes all available functions of the system. However, a real unit may not necessarily be provided with all the optional functions. Should the specifications of the control panel and operating mode of the machine change, the instruction manual supplied with the machine by the manufacturer shall be referred to.

1.2 Introduction to the Manual

The functions of a NC machine system are not only dependent on its NC, but also on its mechanical part. That is to say, its heavy current circuit, servo system, NC and mechanical operation panel jointly determines its performance. The manual only gives a general account from the angle of NC as it is difficult to describe all the functions of an NC machine and programming and operating procedures in detail. As for a specific NC machine, please refer to the instruction manual supplied with the machine by the manufacturer. The contents of the machine instruction manual are more important than those herein.

The performance of a NC machine system is jointly determined by its NC system, mechanical structure, heavy current control and servo system including its mechanical operation panel. The operating manual only describes GSK983M CNC system. To know the performance, programming and operating instructions of the entire NC machine system further, please refer to the machine instruction manual supplied by the manufacturer.

The operating manual describes all matters in detail as much as possible. However, the listing of all matters is unnecessary and impossible as this will complicate the contents of the manual. Therefore, the functions that are not described in this manual may be considered impractical.

The functions that are not described in this manual may be deemed unavailable so far.

Some items are specifically explained in Note sections. Therefore, if any explanation is not available in the current Note section, it is advisable to skip over it, thoroughly read the manual and then return to the part.

2. SPECIFICATIONS

<table>
<tr><th>Serial No.</th><th>Designation</th><th colspan="4">Descriptions</th></tr>
<tr><td>1</td><td>Controllable axes</td><td colspan="4">Standard: 3 axes, namely axes X, Y and Z
(4 or 5 axes are also available upon customer's request. The address of the 4th axis can be selected from A, B, C, U, V or W. That the 4th axis is linear or rotary may be set by parameters. The address of the 5th axis can be designated as U, V, W, A, B or C. It is possible to specify whether the 5th axis is linear or rotary by parameters.)</td></tr>
<tr><td>2</td><td>Number of concurrently controllable axes</td><td colspan="4">Standard: 3, axis 3, link,
(The following configurations are also available upon customer's request: 4, axis 3, link; 4, axis 4, link; 5, axis 3, link; and 5, axis 4 and link.)</td></tr>
<tr><td rowspan="3">3</td><td rowspan="3">Incremental system</td><td>Min. set increment</td><td>0.001mm</td><td>0.0001 inch</td><td>0.001°</td></tr>
<tr><td>Min entered increment</td><td>0.001mm</td><td>0.0001 inch</td><td>0.001°</td></tr>
<tr><td colspan="4">The minimal increment entered in accordance with parameter setting metric system may be 0.01mm.</td></tr>
<tr><td>4</td><td>Bit detecting device</td><td colspan="4">Pulse encoder</td></tr>
<tr><td>5</td><td>Max. instruction value</td><td colspan="4">±99999.999 mm
±9999.9999 inch
±99999.999°</td></tr>
<tr><td>6</td><td>Input format</td><td colspan="4">Changeable block, character and address formats are employed.</td></tr>
<tr><td>7</td><td>Decimal point programming</td><td colspan="4">It is possible to enter numerical values including a decimal point. The addresses that may include a decimal point are X, Y, Z, A, B, C, U, V, W, I, J, K, Q, R and F.</td></tr>
<tr><td>8</td><td>Rapid traverse</td><td colspan="4">Axial speed can be up to 24,000m/min or 960 inch/min. In addition, rapid traverse speed may be changed to FO，25，50 or 100% using rapid traverse override (optional).</td></tr>
<tr><td>9</td><td>Miscellaneous function (Bit M2)</td><td colspan="4">Feedrate can be set within the following ranges: 1 mm/min～15,000mm/min and 0.01inch/min～600.00 inch/min. It is possible to set the upper speed limit of cutting feed by parameters. A feedrate may be selected from 0 to 200% by setting the override of feedrate to 10% as a step. The unit of feedrate may be changed to 0.01 mm/min, 0.001 mm/min or 0.001 inch/min according to parameter setting.</td></tr>
</table>

10	Automatic acceleration/deceleration	Both manual rapid traverse and automatic rapid traverse are performed in linear acceleration/deceleration mode so as to reduce positioning time.
11	Absolute/incremental value instruction	G codes may be used to select absolute or Incremental programming. G90: Absolute programming G91: Incremental programming
12	Coordinate system setting (G92)	When the instruction value for subsequent axes of G92 is used to establish a coordinate system, the actual position of cutting tool will become the instruction value for the coordinate system.
13	Positioning (G00)	All axes individually move to the end point rapidly and decelerate until stop through instruction G00, and the machine performs positioning (whether the machine reaches the instructed position) detection by parameters setting.
14	Linear interpolation (G01)	Linear interpolation may be conducted at the feedrate specified by F codes using instruction G01.
15	Buffer register	When executing a block, the unit reads the next one in buffer register in advance. In this way the intermittence of the NC instruction operation caused by the time required for reading may be avoided. While inputting data in buffer register, BUF will appear on the lower right of LCD screen.
16	Dwell (G04)	The operation of the next block can be delayed before execution using instruction G04. The delay time must be specified with address P or X.
17	Accurate stop detection (G09)	The block of instruction G09 decelerates and performs positioning detection by the end of the block.
18	Accurate stop detecting mode/cutting mode (G61, G64)	If G61 is instructed, the travel instructions following G61 decelerate from doubtful points of all blocks and continue to execute the next block when it is positioned. If G64 is instructed, the travel instructions following G64 rather than positioning instruction do not decelerate but execute the following blocks. As a rule, they are applicable for cutting mode.
19	Miscellaneous function (Bit M2)	The instructions of M2 with 2 digits following may be used to control the ON/OFF signal on machine side. Only one M code can be instructed in one block.
20	Dry run	In dry running mode, feedrate becomes Jog (JOG) feedrate. The rapid traverse speed keeps constant and rapid traverse override (option) is still valid in rapid traverse instruction (G00). However, the dry running of instruction is available when it is set to rapid traverse (positioning) according to parameters.

21	Interlock	It is possible for all axes to individually disable the feed of an instructed axis. If any instructed axis in interlocked during traveling, all axes of the machine will decelerate until stop. Once the interlock signal is cancelled, the machine will accelerate and restart.
22	Single block	A block instruction can be executed each time.
23	To skip over an optional block	The block preceded by a “/” (slash) code may be ignored by turning on the OPTIONAL BLOCK SKTP switch on the machine side.
24	External mirroring	The switch may be used to reverse the program instructions of axes X and the 4th axis as well as the travel of MDI. It is possible to set mirroring with the MDI/LCD panel or the switch on the machine side.
25	Manual Absolute Value ON/OFF	It is possible to determine whether to add the traveling amount through the manual traveling tool to absolute coordinates by switching the manual absolute value switch on the machine side. Manual Absolute Value switch ON: To add; OFF: Not to add.
26	To lock auxiliary function	BCD code signal and strobe signal of M, S, T and B are sent to the machine side.
27	To lock the machine	The machine keeps still. However, position display is still valid as the machine is moving and machine locking is also valid even in the execution of a block.
28	To cancel the instruction for axis Z	The function is only equivalent to Z-axis locking. It is to be used when NC program is checked by drawing a picture with a pen.
29	To feedrate feed	The function is used to temporarily stop the feed of all axes that can be restarted by pressing Cycle Start button. Before the restart of feed, it is possible to insert manual operation in manual mode.
30	To cancel override	The function is used to fix the cutting feedrate at 100% depending on the signals from the machine side.
31	Emergent stop	By pressing the Emergent stop button all feed instructions stops (interrupt immediately) and the machine stops running at the same time.
32	External reset, reset signal	The function is used to reset NC from the outside of NC. The signal stops all feed instruction and the machine decelerates until stop by reset. Additionally, the Reset button on MDI/LCD outputs reset signals to the machine side during emergent stop and external reset.

33	Overtravel	The moving parts of the machine receives arriving signal when they come to the end of travel. Then the movement of axis decelerates until stop and overtravel warning is displayed at the same time.
34	NC Ready signal	When the power supply is switched on, NC gives the signal in controllable state. It stops sending signals to the machine side once the power supply is cut off or the control unit overheats.
35	Servo Ready signal	When the servo system is ready, the signal is sent to the machine side. For the axis to be braked, it is locked when the signal has not been given off. LCD displays NO READY in the absence of the signal.
36	NC warning signal	The signal is sent out when NC is in warning state.
37	Distribution Completed signal	NC outputs the signal when the motion order ends up. If the functions of M, S, T or B and motion instruction are active in a block, the signal is given after the completion of the execution of traveling instruction and the functions of M, S, T or B can be executed.
38	Cycle Operation signal	NC gives the signal in cycle operation.
39	Cycle Operation Start Indicator signal	NC sends out the signal at the start of cycle operation.
40	Feed Hold Indicator signal	NC outputs the signal when it is in holding state through feed hold.
41	Manual continuous feed	(1) JOG feed: For JOG feedrate, the rotary switch may be used for 24-step switching. The ratio of 24-step is geometric progression. (2) Manual rapid traverse: Rapid traverse may also be achieved by manual means. Rapid traverse override may be used for the rapid traverse speed set by parameters. Rapid traverse override is an optional function. Manual continuous feed is available for 2 axes at one time.
42	Incremental feed	Since the following increments may exert positioning control, manual positioning can be performed efficiently. Incremental feed is available for 2 axes at one time (incremental feed amount).
43	Sequence number searching	It is possible to search the sequence numbers in the currently selected program with the MDI/LCD panel.
44	Program number searching	It is possible to search the program numbers of 4 digits following O with the MDI/LCD panel.

45	Clearance compensation	It is this function that compensates the lost amount of movement of the machine. The amount of compensation is set in the minimum amount of movement within 0 ~ 255 by parameters.
46	Program key locking	The function disables the display, setting and editing of the program with a program number of 9000～9899 by means of key locking.
47	Environmental conditions	(1) Ambient temperature 0℃～45℃ for operation and -20℃～55℃ for storage and shipment. (2) Relative humidity ≤90%(without condensation), ≤95%(40℃) (3) Vibration Less than 0.5G for operation and 1G for storage and shipment (4) Ambient air To install an NC in an environment with high density of dust, machining fluid and organic solvent, contact the manufacturer.
48	Self-diagnostics	(1) Servo system a. To give an alarm when the error of the error register goes beyond the setting in halted state. b. To give an alarm when the value of the error register goes beyond the maximum setting. c. To give an alarm in case of malfunction of positioning detection system. d. To give an alarm when the drift voltage is excessive. e. To give an alarm in case of malfunction of speed control unit. (2) NC a. To give an alarm in case of malfunction of the memory. b. To give an alarm in case of malfunction of ROM and RAM. c. To give an alarm in case of malfunction of the microprocessor. (3) Status display a. To display the status of NC on LCD. b. To display the status of I/O on LCD.
49	S function/T function (BCD2 bit)	Once the instructions of 2 digits following S and T are instructed, the code signal of BCD2 bit may be sent out while codes S and T ad other codes are individually output. They are reserved until the following S and T are instructed.

50	S4 Bit (binary 12-bit output) A/S4 Bit (analog output) A	The binary 12-bit or analog voltage corresponding to the speed of principal axis is output to the machine side. The analog voltage is up to ±10V, 2mA and the speed (r/min) of principal axis is directly specified by S4 Bit. The speed of principal axis may be regulated in the following range depending on the contact signal on the machine side: 50, 60, 70, 80, 90, 100, 110 and 120%.
51	S4 Bit (binary 12-bit output) A/S4 Bit (analog output) B	When the speed (r/min) of principal axis is directly specified, the voltage for the current speed of principal axis is output by the presently selected gear numbers 1 to 4.The switching of gear is performed in a heavy-current circuit, resulting that the signal of GRA or GRB is input in NC side. As the judging message for switching gears on the heavy-current side, NC outputs the higher 2 bits or lower 2 bits of S4 bit in BCD codes.
52	Thread cutting /synchronous feeding	A position coder is installed on the principal axis. It is possible to perform thread cutting using the pulse synchronous speed of the position coder.
53	Position coder	To achieve the feed that is in step with the rotation of the principal axis, a device that may generate pulse voltage of which frequency is proportional to the number of revolutions of the principal axis and generates 1024 pulses in each rotation shall be directly connected.
54	Constant surface speed control	Generally surface speed is instructed with B codes. In this way the principal axis accordingly changes when the position of the tool changes so that its surface cutting speed is always equal to the linear speed set by S codes.
55	Second auxiliary function (B3 bit)	Address B is followed by three digits. Once it is instructed, a BCD three-bit code signal will be sent out to position the index table.
56	T-function (BCD4 bit)	Addresses S and T are followed by a 2-digit instruction. Once they are instructed, the code signal of BCD2 bit will likely to be sent out and other codes of the addresses S and T are individually sent out and held until the next S and T are instructed.
57	Code standards	ISO codes (ISO840) and EIA codes (EIA RS-244-A) can be used for program code. The identification of ISO codes and EIA codes may be performed automatically.

58	Rapid traverse override	The automatic or manual rapid traverse speed can be set in 4 levels, i.e. FO, 25, 50 and 100% and FO is likely to be set to a specific speed by parameters.
59	Return to reference point A	Return to reference point A consists the following procedures: (1) Manual return to the reference point; (2) Examination of the return to the reference point (G27); (3) Automatic return to the reference point (G28).
60	Return to reference point B	Besides the functions of returning to reference point A, the return to reference point B also includes the return to the 2nd reference point (G30).
61	Return to the 3rd and 4th reference points	It is possible to set the 3rd and 4th reference points by setting their distances from the 1st reference point and return to these reference points.
62	Storage travel limits 1 and 2	For storage travel limit 1, the area beyond those set by parameters are exclusion areas. For storage travel limit 2, the inside or outside of the area specified by parameters or programs are exclusion areas. The validity or invalidity of storage travel limit 2 is set by G codes. G22: Valid G23: Invalid
63	Memory type pitch error compensation	The function is designed to compensate the pitch error caused by the mechanical wear of the feed screw so as to maximize processing accuracy and mechanical life. Compensation data is saved in memory, thereby omitting the compensation mechanism such as the stop and the relevant setting operations.
64	The selection of a workpiece coordinate system	It is possible to preset one of the 6 workpiece coordinate systems using six G codes, i.e. G54～G59 and subsequent programming can be made in the selected coordinate system.
65	Tool offset (G45～G48)	Tools can be offset using instructions G45～G48. Tool offset refers to the side-play mount corresponding to a move instruction's elongation or reduction of an instruction of D or H codes in axial direction. D or H codes may instruct 1～32 and the maximum offset shall be ±999.999mm or ±99.999 inches. G45: To increase the set value; G46: To reduce the set value; G47: To increase the set value by twice; G48: To reduce the set value by twice.

66	Automatic setting of a coordinate system	To manually return to a reference point, parameters must be preset to establish a coordinate system, namely automatic execution is similar to the condition that the instruction G92 is used at the reference point.
67	Tool length offset (G43, G44, G49)	Tool length offset (tool length compensation) is possible in the direction of the axis Z using instructions G43 and G44. The offset number must be selected within 01～32 using H codes. The offset shall fall within ±999.999mm or ±99.999 inches.
68	Tool radius compensation B, C (G40～G42)	Tool radius compensation is possible using instructions G40～G42. The offset number must be specified within 01～32 using D codes. The offset shall fall within ±999.999mm or ±99.999 inches. Cutter compensation B is unavailable for a tool of a medial angle less than 90°. Cutter compensation 0 can be used for the tool of a medial angle less than 90°.
69	Tool length measurement	Manually position the standard tool to the fixed point of the machine tool. Then manually position the machine tool to be measured to the same mechanical point. The length compensation of the tool will be input as offset once $\boxed{z}$ $\boxed{\text{INPUT}}$ is pressed.
70	Tool life management function	Divide the tools in the cutting tool room into several groups and specify service life for each group of cutting tools. Accumulate the tool processing time or processing number that serves as the evaluation standard for tool life whenever a tool in the groups is used. The next cutting tool in the predetermined sequence is automatically selected in the same group when the tool reaches its service life.
71	Additional offset memory A	The numbers of tool offset and tool radius compensation may be increased to 64.
72	Additional offset memory B	The numbers of tool offset and tool radius compensation may be increased to 99.
73	Additional offset memory C	The number of cutter compensation is increased to 200.
74	F1- bit feed	Once the number of F followed by one digit of 1~9 is instructed, the feedrate corresponding to the number will be set. The instruction FO is a rapid traverse speed. The feedrate of the currently selected number can be increased or reduced by turning the manual pulse generator when the machine side gives a speed-changing signal.

75	External moving function (G80, G81)	Instruction G81 is used to output external movement signals after the positioning of X and Y axes and G80 to cancel it.
76	Fixed cycle A (G80, G81, G82, G84, G85, G86 and G89)	It is possible to perform 6 fixed cycles including drilling cycle, tapping cycle and boring cycle.
77	Fixed cycle B (G73, G74, G76, G80～G89)	It is possible to perform 12 fixed cycles including gun drilling cycle, finish boring cycle, tapping cycle and reverse-tapping cycle.
78	Switch between Inch and metric systems (G20, G21)	Input in Inch or metric system can be selected by switching G codes. G20: Input in Inch system G21: Input in metric system
79	Arc interpolation (G02, G03)	Using G02 (or G03) may achieve any arc interpolation within 0～360° with the feedrate of F code instruction. G02: Clockwise (CW) G03: Counterclockwise (CCW)
80	Sine-curve interpolation	When an axis in the arc plane does not move (the axis is considered as an imaginary axis) in the spiral-curve interpolation instruction, the other 2 axes may be used for sine-curve interpolation.
81	The arc interpolation using arc radius R for programming	In arc interpolation, directly specifying radius with radius value R rather than I, J and K simplifies programming. An arc over or below 180° can be instructed.
82	External deceleration	The mechanical vibration at the end of travel can be minimized and the range of effective travel maximized through the function. External deceleration is not applicable for an additional axis.
83	External workpiece number search A	The function is used to input any program number among 1～31 for NC from the outside such as machine side and to select these programs from the memory of NC.
84	External data input	The function is used to transmit the following data from the outside such as machine side: (1) External workpiece number search C; (2) External cutter compensation C; (3) External warning message; (4) External operation information.
85	Automatic acceleration/ deceleration of cutting feed	Cutting feed and manual continuous feed can be set by parameters to make power type acceleration/ deceleration in 8ms～4000ms time constant.

86	Additional skipping over selected blocks	9 switches for skipping over selected blocks can be set on machine side by adding a digit (1~9) after the switching instruction"/" of a block. When a switch n for skipping over selected blocks is enabled, the blocks with "/n" will be skipped over.
87	Skip function (G31)	When G31 is followed by X, Y, Z, the 4^{th} or 5^{th} axis instruction as same as G01, it is possible to make line interpolation. If skip signal is input from the outside during the execution of the instruction, the remaining part of the instruction stops executing and the blocks that follow will be executed.
88	Program restart	When the sequence number to be restarted is specified, the program will restarts from here.
89	Unidirectional positioning	Positioning can only be performed in one direction in order to eliminate clearance and achieve accurate positioning.
90	Addition of the number of storable programs	96 programs can be added to the standard programs to reach 191 programs in total.
91	Zoom	The tool path instructed in a program can be zoomed in a range of 0.001～99.999x.
92	Insertion of a manual tray	The tool movement overlapping an automatic running instruction can be done only with the pulses of the manual pulse generator without processing interruption.
93	Automatic angle adjustment	When cutting the inner side of an angle in tool radius compensation mode, adjustment can be automatically added in the set area for processing at low speed.
94	Manual feed at any angle	The function is used to set the angle of the positive direction of axis X on the dial gauge of the operation panel of the machine and to make Jog feed in the set direction. In the XY plane, this function is only valid for the increment of 5° spacing within 0°～360° range.
95	Sequence number comparison stop	Once the block with the same sequence number as the preset one appears in the execution of a program, a single block will be in stop status when the execution of the program ends. The function is used to check a program.
96	Indication of running time	The NC automatic running time can be indicated on LCD in seconds, minutes and hours.
97	Menu switch	The ON/OFF can be controlled by using the settings on MDI/LCD rather than the switches on the operation panel on the machine

<table>
<tr><td>98</td><td>User macros
A and B</td><td>These are the intrinsic functions available for manufacturer and user and types A and B are provided due to functional limits.</td></tr>
<tr><td>99</td><td>Graphic display</td><td>The tracks of cutting tools are traced out on LCD.</td></tr>
<tr><td>100</td><td>Manual pulse generator</td><td>The manual pulse generator on the operation panel can be used to perform Jog feed of the machine. The manual pulse generator sends out 100 pulses each turn. The travel of each pulse can be switched between 1x, 10x and 100x depending on the signals on the machine side.</td></tr>
<tr><td>101</td><td>PLC</td><td>PLC MODELS A and B are available.
<table>
<tr><td></td><td>PLC MODEL-A</td><td>PLC MODEL-B</td></tr>
<tr><td>Quantity of input points</td><td>192 points</td><td>192 points</td></tr>
<tr><td>Quantity of output points</td><td>128 points</td><td>128 points</td></tr>
<tr><td>Program step</td><td>Up to 2,000 steps</td><td>Up to 5,000 steps</td></tr>
</table></td></tr>
</table>

3. PROGRAMMING

3.1 What Is Programming?

NC processing machine operates by developed programs. To process a part on the NC machine, the route and other processing conditions of cutting feed shall be included in the program. The program is called “part program”.

The following diagram indicates the processes from preparation of detail drawing to NC’s execution of the processing program.

Detail drawing → Processing plan → Part programming → NC’s execution of processing program

(1) Determine the NC processing range and select an NC machine to be used.
(2) Determine the assembling method of the workblank on the machine and select a necessary clamping device and a tool.
(3) Determine the cutting sequence (process type, home point of the cutting tool, the cut depths of rough cutting and finish cutting and route of cutting feed).
(4) Select a cutting tool and a tool clamping device and determine the mounting position on the machine.
(5) Set cutting conditions (spindle rotating speed, feed speed and whether to use cooling fluid, etc).

Part program is an NC instruction for controlling feed route and the auxiliary movement of a machine written according to the rules for NC. The instructions are usually written in a program list.

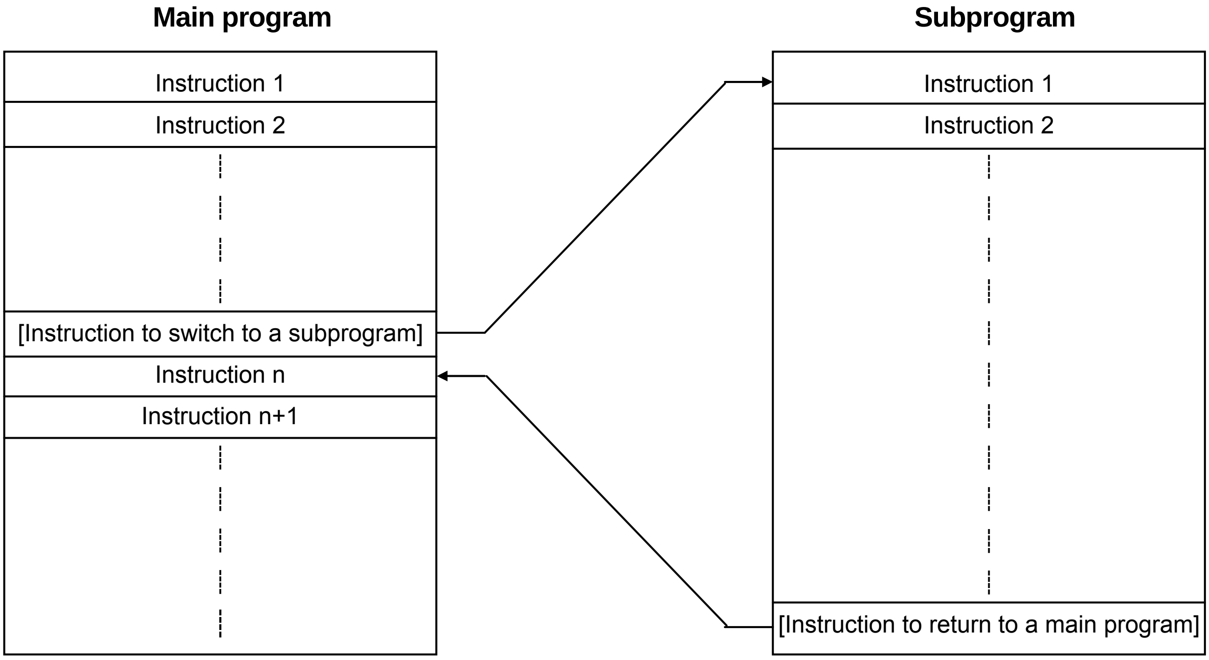
This chapter will describe how to develop a part program.

3.2 Program Make-up

A program consists of main program and subprogram. As a rule, NC moves by the instructions of a main program. When the main program gives an instruction to switch to a subprogram, NC moves by the subprogram.

When there is an instruction to return to a main program in a subprogram, NC will return to the main program and continue to move by the instructions of the main program.

NC memory can store 95 main programs and subprograms in total. When one of the main programs is selected, NC machine can move by its instructions.



Note: The number of programs that can be stored will be increased to 191 if the function "Addition of the number of storable programs" (Optional) is selected.

Refer to Chapter 4: Operations for the storage method of program.

3.2.1 Block

A program is composed of a number of instructions. Each instruction unit in a program is called a block. Blocks are distinguished from each other by end code. As described below, end-of-block code is indicated with a ";".

For example:

XXXX;
XXXX;
XXXX;

Note 1: The maximum number of characters a block is not limited in.

Note 2: End-of-block code: CR for EIA codes and LF for ISO codes.

3.2.2 Program word

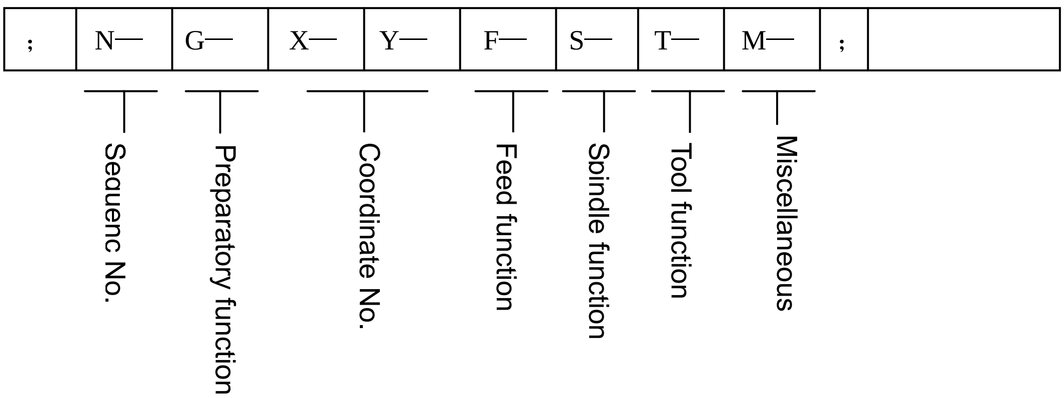
The elements composing a block are program words. The program word below consists of an address and a subsequent figure. A "+" and a " –" can also be added before the figure.

$$\underbrace{\underbrace{X}_{\text{Address}}\ -\ \underbrace{1000}_{\text{Figure}}}_{\text{Program word}}$$

An address is indicated with a letter among A～Z. An address determines the meaning of its subsequent figure. The addresses that can be used in an NC and their meanings are as follows. An address may have different meanings depending on different instructions of preparation function in a program.

Name	Address	Meaning
Program No.	: (ISO) /O (EIA)	Program number
Sequence No.	N	Sequence number
Preparatory function	G	Instructed move mode (linear, arc, etc)
Coordinate word	X, Y, Z	Move instruction of coordinate axis
	A, B, C, U, V, W	Move instruction of additional axis
	R	Arc radius
	I, J, K	Coordinates of arc center
Feed function	F	Designation of feedrate
Spindle function	S	Designation of spindle rotational speed
Tool function	T	Designation of tool number and tool offset number
Auxiliary function	M	Designation of ON/OFF control on machine side
	B	Worktable indexing, etc.
Offset No.	H, D	Designation of offset number
Dwell	P, X	Designation of dwell time
Designation of program No.	P	Designation of subprogram number
Designation of sequence No.	P	Designation of sequence number: Program is repeatedly executed at this sequence number.
Number of repetition	L	Numbers of repetition of subprogram and fixed cycles
Parameter	P, Q, R	Parameters of fixed cycle

For example, these program words may compose the following block.



In the following program list, a line indicates a block and a case in the block indicates a program word.

Name															S57.10.10				Page/	
Program No. 0 (:) 2002																				
/	N	G	X	Y	Z	A/B/C	C/V/W	R/I	J	K	F	S	T	M	B	H/D	L	P	Q	；
	N20	G92	X 100.0	Y 200.0	Z 300.0															；
	N21	G00	X 196.0	Y 315.0	Z 500.0							S400	T15	M03						；
	N22	G01									F10.0									；

(Note) CR(EIA)，LF(ISO)

3.2.3 Input format

All program words composing a block shall be instructed in the formats as specified below. The input format of this system is a variable block format. Therefore, the number of the program words in a block and the number of characters in a program word are variable, which makes programming easier.

(1) Input in metric system

$$\mathrm{NO4 \cdot G02 \cdot XL{+}053 \cdot YL053 \cdot ZL{+}053 \cdot}$$

$$\mathrm{\alpha L{+}053 \cdot \beta L{+}053 \cdot} \left\{\begin{matrix} \mathrm{RD053} \\ \mathrm{ID053 \cdot JD053 \cdot KD{+}053} \end{matrix}\right\} \mathrm{\cdot F050 \cdot} \left\{\begin{matrix} \mathrm{D02} \\ \mathrm{H02} \end{matrix}\right\} \cdot$$

$$\left\{\begin{matrix} \mathrm{S02} \\ \mathrm{S04} \end{matrix}\right\} \cdot \left\{\begin{matrix} \mathrm{T02} \\ \mathrm{T04} \end{matrix}\right\} \mathrm{\cdot B03 \cdot M02;}$$

(2) Input in Inch system

$$\mathrm{NO4 \cdot G02 \cdot XL{+}044 \cdot YL{+}044 \cdot ZL{+}044 \cdot}$$

$$\mathrm{\alpha L{+}053 \cdot \beta L{+}053 \cdot} \left\{\begin{matrix} \mathrm{RD044} \\ \mathrm{ID044 \cdot JD044 \cdot KD044} \end{matrix}\right\} \mathrm{\cdot F032 \cdot} \left\{\begin{matrix} \mathrm{D02} \\ \mathrm{H02} \end{matrix}\right\} \cdot$$

$$\left\{\begin{matrix} \mathrm{S02} \\ \ \end{matrix}\right\} \cdot \left\{\begin{matrix} \mathrm{T02} \\ \ \end{matrix}\right\} \mathrm{\cdot B03 \cdot M02;}$$

S04 T04

Note 1: α and β are for any one of A, B, C, U, V and W.

Note 2: The address and digits in the above formats have the following meanings.

X L + 0 5 3

- 3: The three digits behind decimal point
- 5: The five digits before decimal point
- 0: Leading zero may be omitted.
- +: With a symbol
- L: Absolute value or incremental value
- X: Address

J D 0 5 3

- 3: The three digits behind decimal point
- 5: The five digits before decimal point
- 0: Leading zero may be omitted.
- D: Incremental value with a symbol
- J: Address

For example: When a cutting tool moves to 50.123mm along X axis, its move instruction is as follows:

G00 X50 123

- 123: The three digits behind decimal point
- X50: The five digits before decimal point: 00050
  The leading zero and T are omitted.
- G00: G00 must not be omitted even the leading zero may be omitted. (G00 specifies rapid traverse (positioning)).

Note 3: When an address word in a block is instructed twice or more times, the last instruction will be valid and not give any alarm in principle.

For example: G01 M03 S200 M08;

Now M08 is valid and M03 invalid. For G codes, the lastly specified one in each group of G codes in a block is valid. However, G90/G91 is only valid at the specified location in a block (see Section 3.3.8.).

Example G90 X10.0 G91 Y20.0 ;

G90: Absolute; G91: Incremental

If R and I, J or K are concurrently instructed in an arc interpolation instruction, R is always valid and this is not related to instruction sequence.

Note 4:F050 in Inch-system input format may also be changed into F051 through parameter switching. Refer to Section 3.4.3 “To multiply feedrate by 1/10”.

Note 5: Since P and Q have a number of meanings, they are omitted in the above format.

Note 6: Refer to Section 3.2.4 ”Decimal entries”.

Note 7: Multiply the value input in metric system of X, Y, Z, A, B, C, U, V, W, I, J, K, Q and R by 10 by means of parameter setting.

$$\mathrm{XL{+}052} \bullet \mathrm{YL{+}052} \bullet \mathrm{ZL{+}052} \bullet \alpha\mathrm{L{+}052} \bullet \beta\mathrm{L{+}052} \bullet \left\{ \begin{array}{l} \mathrm{RD052} \\ \mathrm{ID052} \bullet \mathrm{JD052} \bullet \mathrm{KD052} \end{array} \right\}$$

(α and β are A, B, C, U, V and W) (Input in metric system) Refer to Section 3.3.2.2 “To multiply an input unit by 10”.

Note 8: Refer to Section 3.3.2.2 “To multiply an input unit by 10”.

3.2.4 Decimal programming

It is possible for this unit to input the numerical values with a decimal point. Decimal point is used in the values in the unit of distance, time or speed. However, some addresses may not use decimal for entry. The location of a decimal point indicates the position of mm, inch, degree or second.

X15.0 X15mm or x15 inches

F10.0 10mm/min for 10 inch/min

G04×1 Feedrate for 1 second

B90.0 B90deg

The addresses that can be entered with decimal point are as follows:

X, Y, Z, A, B, C, I, J, K, R, Q and F.

Note 1: During the dwell of an instruction, X can be input with the decimal point but P cannot (because P is also specified with a sequence number).

Note 2: When G codes are used to change the location of the decimal point, it is necessary to predetermine G codes even in a block. G20; (specify in Inch system) X1.0G04；X1.0 is considered as a travel distance (in inch) because it does not indicate time. The result is held for 10 seconds in relation to X10000G04.

When G04 is entered, its indication changes from 1.0 to 10.0.

G04X1.0 is treated as G04X1000 and the result is held for 1 second.

Note 3: Take note that the condition with or without a decimal point is quite different. Its programming mode differs from an electronic computer.

G21; (specified in metric system)

X1……X1 mm

X1……X0.001 mm

G20; (specified in Inch system)

X1……X1 inch

X1……X0.0001 inch

Note 4: The numerical values with/without a decimal point can be mixed.

X1000 Y23.7;

X10 Y22359;

Note 5: If a specified value is less than the minimum setting, the value will be rounded down. The instruction of X1.23456 is considered as X1.234 when it is entered in metric system and 1.2345 in Inch system . It has cumulative error for an incremental value instruction and has no cumulative error but rounding error for an absolute value instruction. The instructed digits must not exceed the allowable maximum digit.

X1.23456789……There is an error because it has more than 8 digits.

X1.2345678…… There is no error when it is up to 8 digits inclusive.

Note 6: When a figure with a decimal point is entered, the figure will be converted into the integer of the minimum input increment.

(Example)X12.34 ⟶ 12340(entered in metric system)

In addition, it is necessary to perform digit number verification for the converted integer.

(Example)X1234567.8 ⟶ X1234567800(entered in metric system). It gives an alarm because the number of digits exceeds 8.

3.2.5 Maximum instruction value

The maximum instruction values of all addresses are as listed in the table below. Take note that what the table lists are the ranges of the maximum specified values for the NC unit rather than the mechanical moving ranges of the NC machine tool. For instance, the moving range of X axis is about 100m (entered in metric system) for an NC unit while the travel distance of axis X is likely to be limited to 2m for a machine tool and so is its feedrate. The cutting feedrate of an NC unit is up to 15m/min while that of an NC machine tool is likely to be limited to 6m/min. In actual programming, both this manual and the instruction manual supplied by machine builder shall be referred to. Programming shall be performed on the basis of full knowledge about the programming ranges of a specific machine tool.

Table 2.5: Basic addresses and ranges of instruction values (including additional options)

Name	Address	Input in mm Output in mm	Input in inch Output in mm	Input in mm Output in inch	Input in inch Output in inch
Program No.	: (ISO) O (EIA)	1～9999	See the left	See the left	See the left
Sequence No.	N	1～9999	〃	〃	〃

Preparatory function	G	0～99	〃	〃	〃
Coordinate word	X, Y, Z, I, J, K, Q, R, A, B, C, U, V, W	±99999.999mm ±99999.999°	±3937.0078inch ±99999.999°	±99999.999mm ±99999.999°	±937.0078inch ±9999.999°
Feed per minute	F	1～15000 mm/min	0.01～600.00 inch/min	1～15000 mm/min	0.01～600.00 inch/min
Feed per minute (feedrate l/10) (parameter setting)	F	0.1～15000.0 mm/min	See the above	0.1～15000.0 mm/min	See the above
Spindle function	S	0～30000	See the left	See the left	See the left
Tool function	T	0～9999	〃	〃	〃
Miscellaneous function	M	0～99	〃	〃	〃
Dwell	X, P	0s～99999.99s	〃	〃	〃
Sequence No. setting	P	1～9999	〃	〃	〃
Repeated times	L	1～9999	〃	〃	〃
Offset No.	S, H	0～200	〃	〃	〃
2nd auxiliary function	B	0～999	〃	〃	〃

3.2.6 Program number

The control unit can store a number of programs in the memory of the NC. Program numbers are added to all programs in order to distinguish them.

Program number

$\underbrace{\bigcirc\ \square\ \square\ \square\ \square}_{\text{4 digits}}$ (From 1 to 9999.0 and preceding 0 may be omitted)

The program starts from the program number and ends at M02, M30 or M99.

01111························M02;	02222····························M30;
#1111 program	#2222 program

M02 and M30 indicate the end of a main program. M99 indicates the end of a subprogram.

05555…………………………M99;

#5555 subprogram

Note 1: For ISO codes, replace O with a ":".

Note 2: The block with an optional block skip code such as /M02, /M30 and /M99 shall not be deemed as the end of a program.

Note 3: When a program is not provided with a preceding program number, the first sequence number (N…) other than NO of the program may be used to replace the program number.

Note 4: If a program is not provided with a preceding program number or sequence number, MDI/LCD panel shall be used to specify a program number when storing the program into the memory.

Note 5: For several programs, the second program and the succeeding ones are not necessarily provided with an EOB code skipped wit a mark. However, an EOB code shall be used before the program when the foregoing program ends by ER (EIA) or %(ISO).

Note 6: It is possible to operate without a program number. Subprogram shall be provided with a program number.

Note 7: In some cases, the program number from 9000～9899 can only used by machine manufacturer rather than user.

Note 8: When it has a robot option, program numbers 9900～9999 are used as robot data.

Note 9: When there is no M02, M30 or M99 but ER (EIA)%(ISO) at the end of the program, or the next program number is 0, the end of the program shall be set by parameter No.306 BIT3 (NEOP).

3.2.7 Sequence number

A sequence number may be specified with N followed by four or less digits (1~9999) at the start of a block. The order of sequence numbers is random and discontinuous. It is likely that all blocks are provided with sequence numbers or sequence numbers are only added at the necessary locations of a program.

It is recommended to specify sequence number in succession at key locations, e.g. when replacing new tools or change the worktable indexing to a new machined surface.

Note 1: The use of the sequence number NO. is not allowed for the purpose of compatibility with the program formats of other NC units.

Note 2: Since 0 cannot be used as a program number, the sequence number treated as a program number shall not be 0.

3.2.8 To skip over an optional block

When a slash followed by a digit /n(n=1～9) is specified at the beginning of a block and the

OPTIONAL BLOCK SKTP switch n is enabled on the operation panel , the block with a /n corresponding to switch number n will be ignored.

When the OPTIONAL BLOCK SKTP switch n is disabled, the block with a /n is valid, namely operator may choose to skip over the block with a /n. The 1 in /1 can be ignored. However, the 1 in /1 must not be ignored when two or more OPTIONAL BLOCK SKTP switches are used.

When an OPTIONAL BLOCK SKTP switch is enabled, the ignored area is as follows:

；/2N123G01X4……………………；N7856

Ignored area

Example: N100X100；

N101/2z100；

N102/2/3X200；

N103/3z200；

In the above example, blocks N101 and N102 will be skipped over when No. 2 switch is enabled and N102 and N103 skipped over when No. 3 switch enabled.

Note 1: Slash (/) shall be specified at the beginning of a block. If it is specified at other places of a block, the information from the slash (/) to EOB code will be ignored but the information before the slash (/) will remain valid.

Note 2: When the OPTIONAL BLOCK SKTP switch is enabled, TH and TV checkouts shall be performed for the skipped part as the switch is disabled.

Note 3: The block to be skipped over is identified when the memory sends information to the buffer register. When the block preceded by a slash is written in the buffer register, it will not be ignored even if the OPTIONAL BLOCK SKTP switch is enabled.

Note 4: The function remains available during sequence number searching.

Note 5: The function is invalid when saving a program in the memory. That is, the block preceded by a slash (/) will be written in the memory regardless the state of OPTIONIAL BLOCK SKIP.

Note 6: The programs may be completely output from the memory regardless the state of OPTIONIAL BLOCK SKIP.

Note 7: Some OPTIONIAL BLOCK SKIP switches are likely to be unavailable for some machines. Therefore, it is necessary to consult the machine manufacturer for the number of usable switches.

Note 8: For the system with additional function to skip over an optional block, the 1 in /1 must not be omitted if there are two or more marks for skipping over an optional block in one block.

/1 shall be specified as per the above requirements.

Example: Incorrect: //3 G00 X10.0；

Correct: /1/3 G00 X10.0；

3.3 Dimension Word

A dimension word prescribes the movement of a tool. It composes a relevant instruction with the address of a kinematical axis and a value indicating the direction and amount of travel. It varies depending on the different absolute and incremental programming modes. (See Section 3.3.8.)

Address of dimension world		**Meaning**
Basic axes	X, Y, Z	The addresses of all axes in a rectangular coordinate system: To indicate the location of the axis or the distance in axial direction.
Additional axes	A, B, C, U, V, W	The addresses of the 4th and 5th axes: To indicate the angle of axis of rotation or the location and distance of linear axis.
Arc interpolation parameters	R	To specify the radius of an arc.
	I, J, K	To indicate the distance from the home point to the arc center along axis X, Y or Z or an axis parallel to them.

3.3.1 Controllable axis

The kinematical axis of a machine under the control of the NC system is called controllable axis.

Each control shaft is recalled with the dimension word address of the control unit.

The number of axes that can be controlled by the NC system may be 3 (axes X, Y and Z) and can be added to 4 or 5.

An additional axis may use any address among A, B, C, U, V or W. A, B and C are recommended for axis of rotation and U, V and W for linear axis.

2 standard axes can be controlled in a block at the same time. The number of concurrently controllable may be increased to 3 or 4 by means of additional selection. Separate control of additional axis is only used for 3-axis link. The selection of additional axis gang control function enables the link of 2 or 3 additional axes. If 3-axis gang control function is selected, it is not necessary to select additional axis gang control function any more.

Number of controllable axes	Number of concurrently controllable axes				
	Number of standard axes	Selection of concurrently controllable additional axes	Concurrent selection of 3 axes	Concurrent selection of 3 axes + Concurrent control selection of additional axes	Concurrent selection of 4 axes
3	2		3		

4 5		2 (Including additional axes)	X, Y, Z	3 (Including additional axes)	4

Note 1: The system gives #17 alarm if additional axes (A, B, C, U, V, W) are specified without additional control function.

Note 2: The number of link axes is always 2 in manual operation.

Note 3: When the system has the 5th axis, the following function control shall be exerted.

① Thread cutting and synchronous feed cannot be performed.

② The capacity of part program is reduced to 75%.

③ Additional S4 digit analog output function is not available.

④ Additional constant surface speed control function is not available.

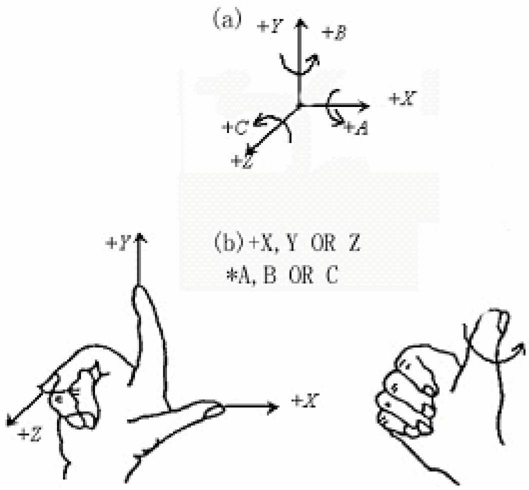
Coordinate axes and movement symbols

If the machine coordinate axes specified by the machine are not identical with the tool movement symbols, programming is subject to serious disorder. The concerned basic concepts are provided in EIA RS-267-A or ISO841.

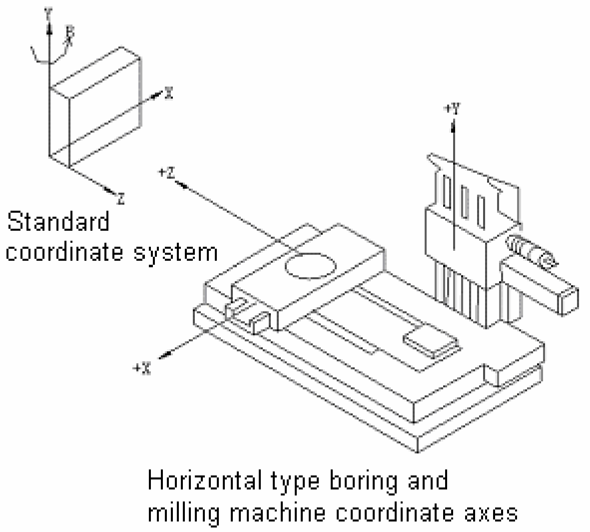
However, the following points shall be observed during programming.

a) Program shall be developed on the basis of a standard coordinate system (right-manual rectangular coordinate system).

b) When programming, assume that workpiece does not move and a cutting tool moves around it.



Right-hand rectangular coordinate system



3.3.2 Set unit

3.3.2.1 Minimum set unit and minimum travel unit

1) Minimum set unit (input unit)

   Minimum unit of tool travel is input by instruction. The minimum unit is given in mm, inch or degree.

2) Minimum travel unit (output unit)

   The minimum unit output to the machine is indicated in mm, inch or degree. Any one of the following combinations may be employed.

Input/output		Minimum set unit	Minimum travel unit
Linear axis	Input in mm and output in mm	0.001mm	0.001mm
	Input in inch and output in mm	0.0001inch	0.001mm
	Input in mm and output in inch	0.001mm	0.0001inch
	Input in inch and output in inch	0.0001inch	0.0001inch
Axis of rotation		0.001°	0.001°

Note: The set unit of an axis of rotation cannot be converted between Inch and metric systems.

Whether the minimum travel unit is 0.001 mm or 0.0001inch shall be determined by the machine and selected by presetting parameter No.006 BIT0 (SCW).

Whether the minimum travel unit is 0.001 mm or 0.0001inch shall be selected by G codes or by setting parameter through MDI/LCD panel.

G20……The minimum set unit of a linear axis is 0.0001inch.

G21……The minimum set unit of a linear axis is 0.001mm.

The state of G20 and G21 remains unchanged when the system is powered on/off.

3.3.2.2 Input unit×10

The minimum set unit in mm may be changed to 0.01mm by parameter No.006 BIT1 setting. The minimum set unit in inch cannot be changed.

	Address	Minimum set unit	
		Input in mm	Input in inch
Dimension word	X, Y, Z, Q, R, I, J, K, U, V, W	0.01mm	0.0001inch
Revolution axis	A, B, C	0.01°	0.01°
Dwell time	X	0.01s	0.001s
	P	0.01s	0.001s

It cannot be changed in the following cases:

a) Inputting differs from the above conditions

b) Display unit

c) Maximum range of instruction value

d) Units of step feed and manual feed

e) Offset input

f) Others

Note 1: In the following descriptions herein, input unit is either 0.0001inch or 0.001mm.

Note 2: Display unit may become 0.01mm or 0.01 degree by parameter No.006 BIT2 (MDL) setting.

3.3.3 Maximum travel

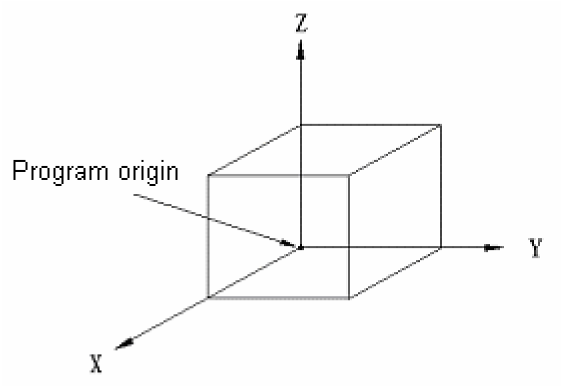
The maximum travels that can be instructed of the device are as listed in the table below:

Input in mm and output in mm	Input in inch and output in mm	Input in mm and output in inch	Input in inch and output in inch
±99999.999mm ±99999.999°	±3937.0078inch ±99999.999°	±99999.999mm ±99999.999°	±9999.9999inch ±99999.999°

Note: The travels listed in the above table change depending on different machines.

3.3.4 Program origin and coordinate system

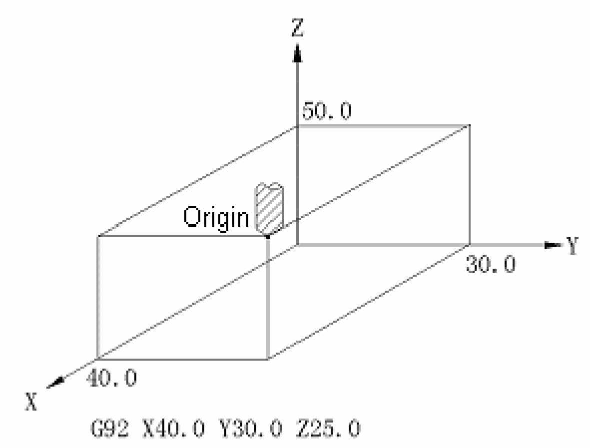
Program origin and coordinate system shall be determined during programming. Generally a specific point on the workpiece is determined as the program origin.



This coordinate system is a workpiece coordinate system.

3.3.5 Coordinate system and origin of processing

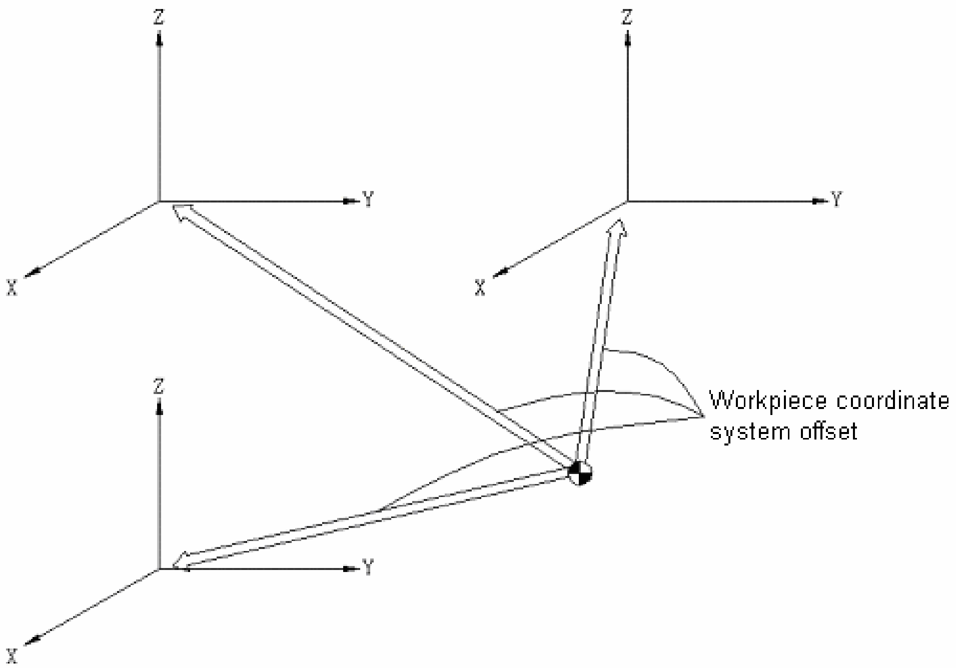
A workpiece coordinate system needs to be used when a program is sent to the NC. The cutting tool starts motion from the origin and the program starts from the origin, but the NC always need to know the coordinates of the cutting tool at the origin through G92 instruction (coordinate system setting).



3.3.6 Workpiece coordinate system

Several workpiece coordinate systems are needed when the mounting positions of the several trailing bars for the machine are different. In this case, the 6 coordinate systems preestablished on the machine may be selected with 6 G codes (G54～G59) and the following programs are executed in the selected coordinate system. All coordinate systems shall be determined with the set distance between a reference point (a fixed point on the machine) and their respective origins of coordinate (workpiece origin offset). See the following diagram for details.

Refer to Section 4.4.13 for the setting method of workpiece origin offset.

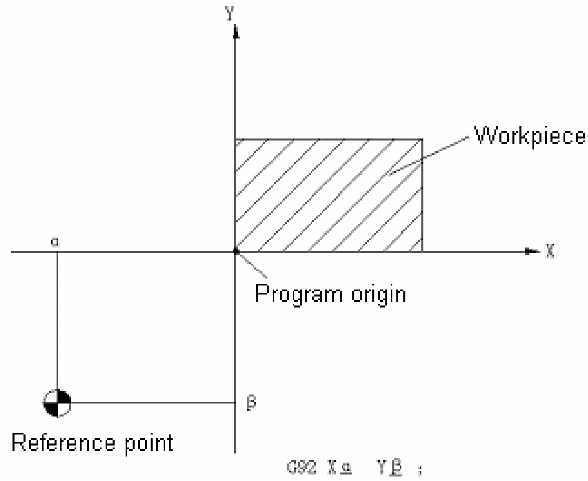


It is not necessary to establish a coordinate system with G92 instruction when the above-mentioned workpiece coordinate system is used. Concurrent use of G54～G59 and G92 may substitute the coordinate system established by G54～G59. Therefore, usually G92 shall not be used in conjunction with G54～G59.

Note: When using the workpiece coordinate system established by G54～G59, make sure to return to the first reference point after switching on so that G54 will automatically generate a workpiece system. Hence it is not necessary to set an automatic coordinate system.

3.3.7 Reference (position) point

Reference point is a fixed point on the machine. Reference point return function allows the cutting tool to return to the reference point. Therefore, a program is likely not to start from one point on a workpiece coordinate system but from the reference point. In this case, the return of the tool to the position of the reference point shall always be specified with G92 in the workpiece coordinate system because the reference point is a specific point on the machine and the program is developed on the basis of the workpiece coordinate system with a point on the workpiece as its origin.



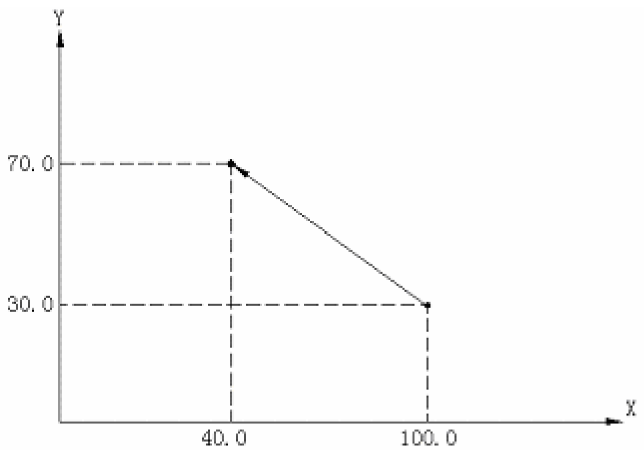
Note: G92 instruction is not necessary when the workpiece coordinate systems established with G54～G59 are used.

3.3.8 Absolute value instruction and incremental value instruction

The distance that a cutting tool moves along all axes may be programmed with an incremental instruction or absolute instruction.

It is possible to directly program a running distance in a block using an incremental instruction (G91).

When using an absolute instruction (G90), the end position of the cutting tool is indicated with a coordinates in the block.



In the above diagram, the program will give the following position when an incremental instruction is employed:

G91 X-60.0 Y40.0；

In the above diagram, the program will give the following position when an absolute instruction is employed:

G90 X40.0 Y70.0；

The instruction mode of G90/G91 must not be changed for all addresses in a block so that the program can be compatible with other NC.

3.4 Feed Function (F function)

3.4.1 Rapid traverse (positioning) function

All axes of the machine operate at the specified rapid traverse speed during rapid traverse.

As a rule, the rapid traverse speed has been set by manufacturer before shipment (set by parameters RPDFX through RPDF4).

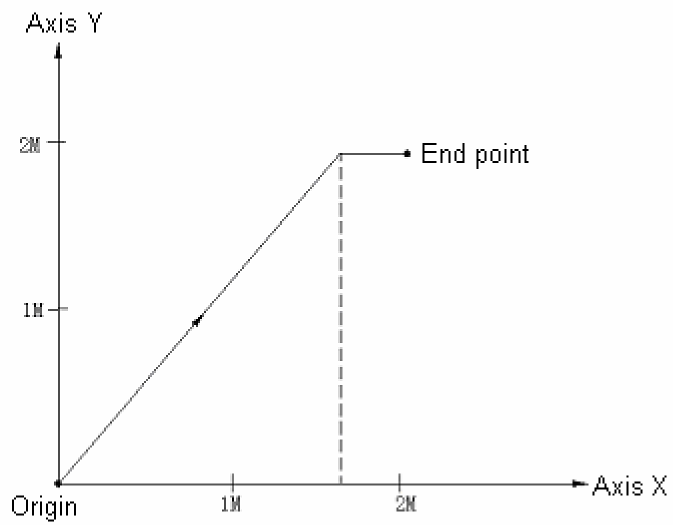
Since all axes of the machine moves individually, the times for all axes to move from their home positions to end positions are different.

For example, when the rapid traverse speeds of axes X and Y reach 5m/min and 8m/min respectively and movement program is as follows:

G91 X200.0 Y200.0；

The times for axes X and Y of the machine to run from the starting point to the end point are 24s and 15s respectively.

The locus of the cutting tool in the above example is as indicated in the figure below:



The control over the override of the rapid traverse speed can be achieved with the switch on the operation panel of the machine. (FO, 25%, 50%, 100%) FO shall be determined by setting parameter No.113 (SPDFL) and its unit shall not be indicated in percentage (%) but mm/min or inch/min.

3.4.2 Cutting feedrate

The cutting feedrate shall be specified in the form of the distance per minute. The feedrate is specified as follows with F codes:

F1 (1mm/min; 0.01inch/min)

F15000(15000mm/min) or F60000(600.00 inch/min)

The feedrate is restrained to the upper limit.

According to the upper limit, the manufacturer of the machine sets feedrate by parameter No.106 (FEDMX) or exerts override restraint from 0 to 200% (10% each step) with the switch on the operation panel. The control over upper limit speed is also effective for override feedrate. The means of specifying feedrate with F codes is also applicable for the axis of rotation.

Example: To input in metric system　F050

To input in Inch system　F032

When inputting in metric and Inch system s, decimal point may be used for inputting and it is located in the position of degree/min.

To input in metric system　F12　0.12 degree/min

To input in Inch system　F12　0.12 degree/min

To input in metric system　F12.0　12 degree/min

To input in Inch system　F12.0　12 degree/min

Note 1: Except the acceleration and deceleration processes in the row of NC, NC's calculus error regarding instructed feedrate is kept within ±2%. Moreover, the error is determined by measuring a travel distance over 500mm when NC is in stable running state.

Note 2: The number of the digits of F codes is up to 7. Should the input feedrate is above the upper limit, the feedrate is restrained to the upper limit.

3.4.3 To reduce feedrate to 1/10

The input speed in metric system can be changed to 1/10 of the original speed by parameter No.006 BIT3 (FMIC) setting.

Item	Minimum input unit	Range
Feed per minute	0.1mm/min	F1 to F150,000 (0.1 mm/min to 15000.0mm/min)

3.4.4 Synchronous feed (feed per rotation)

It is possible to specify a feedrate in relation to the feed amount per minute of the spindle. G95 specifies synchronous feed while G94 feed per minute (with travel per minute as the feedrate).

<table>
<tr><th colspan="2"></th><th>Feed per minute</th><th>Synchronous feed</th></tr>
<tr><td colspan="2">Meaning</td><td>Feed amount per minute of cutting tool</td><td>Feed amount of cutting tool per rotation of spindle</td></tr>
<tr><td colspan="2">Address</td><td>F</td><td>F</td></tr>
<tr><td colspan="2">G code</td><td>G94</td><td>G95</td></tr>
<tr><td rowspan="2">Range</td><td>Input in mm</td><td>1mm/min～15000mm/min
(F1～F15000)</td><td>0.01mm/r～500.00mm/r
(F1～F50000)</td></tr>
<tr><td>Input in inch</td><td>0.01inch/min～600.0inch/min
(F1～F60000)</td><td>0.0001inch/r～50.0000inch/r</td></tr>
<tr><td>Restraining value</td><td colspan="3">Both the feed per minute and synchronous feed at specific feedrate are restrained. The restraining value is set by manufacturer (only the feedrate with override is restrained).</td></tr>
<tr><td>Override</td><td colspan="3">The override from 0 to 200% (10% each step) is effective for both feed per minute or synchronous feed.</td></tr>
</table>

A restraining value is set in mm/min or inch/min. Synchronous speed is converted to mm/min or inch/min with the following formula:

fm=fr×R

Where fm: the feed per minute in mm/min or inch/min

fr: the synchronous feedrate in mm/r or inch/r

R: the rotational speed of the spindle in r/min

Note 1: Both G94 and G95 are modal. Once they are specified, they will be effective until other G codes appear.

Note 2: The spindle shall be provided with a synchronous feed position coder.

Note 3: When the rotational speed of the position coder is as low as 1 rpm, the feedrate will become irregular. The irregularity has no impact on the machining of the machine. Therefore it is still usable when the rotational speed is as low as 1 rpm. However, the degree of irregularity must not worsen more as the further reduction in rotational speed is subject to deterioration.

3.4.5 F1 digit feed

Specifying a digit (1～9) behind F sets the feedrate of the corresponding number. Feedrate is preset for each number using parameter. Specify FO as the rapid traverse speed and set the F1 digit feedrate switch on the machine panel to ON. Then rotate the manual pulse generator to increase or reduce the feedrate of the currently selected number.

The increase or reduction in feedrate:

$\triangle F=\dfrac{Fma\times 1}{100X}$ /Manual pulse generator per case

Where: Fma×1 is used as the upper limit of feedrates of F1～F4 (set with a parameter).

Fma×2: is used as the upper limit of feedrates of F5～F9 (set with a parameter).

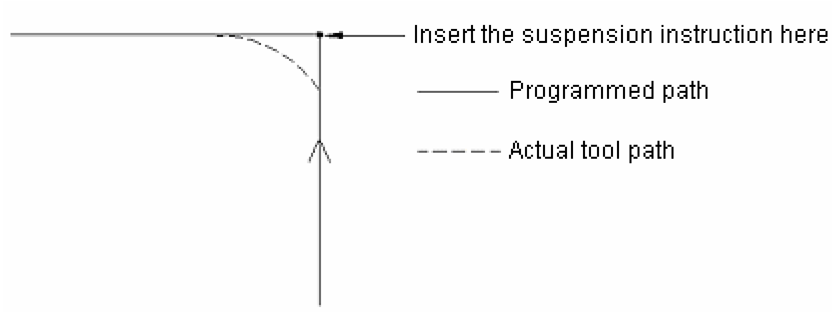
X: Any figure between 1 and 127 (set with a parameter).

Set or modified feedrate can be saved even in case of power failure. The current feedrate is displayed on the LCD.

3.4.6 Automatic acceleration and deceleration

During the start or stop of feed, acceleration or deceleration is performed at some time constant so as to prevent the mechanical system from shock. Hence it is possible to consider the problem of acceleration and deceleration during programming.

It is impossible to machine a sharp angle under the influence of automatic acceleration and deceleration. To machine a sharp angle, dwell instruction (G04) must be added between two blocks.



Once the dwell instruction is inserted, the actual tool path tallies with the programmed path. The rapider feedrate is, the greater time constant of acceleration and deceleration and angle error will be.

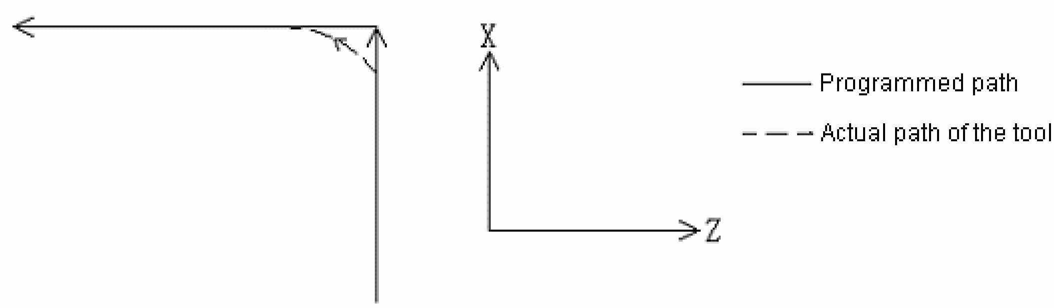
Note 1: The changes of feedrate between the blocks that have specified different moving modes are as listed in the table below.

Foregoing blocks / New blocks	**Positioning**	**Cutting feed**	**Not to move**
Positioning	×	×	×
Cutting feed	×	○	×
Not to move	×	×	×

×: The next block will be executed when the instruction speed is reduced to zero.

○: Continue to execute the next block so that the feedrate will not change excessively.

Note 2: Acceleration/deceleration is independently performed on both axes (axes X and Z) and their feedrates changes between blocks, resulting in the discrepancy between the actual path of the tool and programmed path. For example, if the tool only moves along axis X in a block and along axis Z in the next block, the movement in the direction of axis X starts to decelerate near the angle. Meanwhile, it starts accelerated movement in the direction of axis Z. The actual path of the tool is as indicated in the figure below.



In arc interpolation, the actual arc radius is less than the programmed on (see the Appendix). The deviation may be increased by minimizing the time constant of acceleration and deceleration.

3.4.7 Automatic angle adjustment

Tool cutting is subject to overload if the tool performs the rough machining with tool compensation in inner angle and inner arc area at programmed feedrate. The function automatically reduces feedrate so as to lower the tool's overload in the above machining areas, thereby obtaining a smooth machining surface.

3.4.7.1 Automatic adjustment of inner angle

1) Working conditions

Feedrate can be automatically regulated provided that both the two blocks in succession passing the angle satisfy the following requirements.

a. The G codes of Group 01 are G01, G02 or G03.

b. The offset is not 0 in offset mode.

c. Offsetting shall be performed inside the angle to be processed.

d. The axis moves along the offset surface.

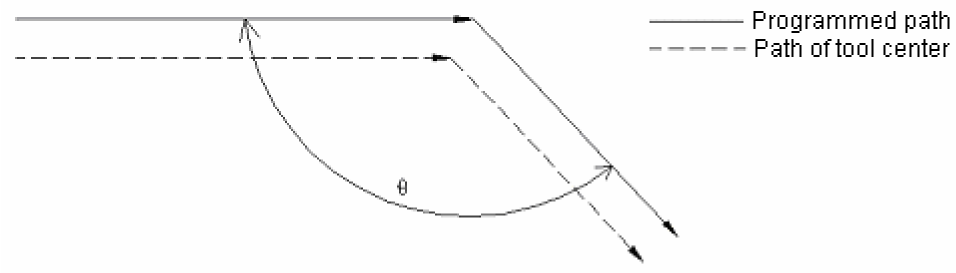
e. Instructions G41 and G42 do not exit in the succeeding block.

f. Instructions G41 and G42 do not exist in the preceding block or the block is not started though it has the two instructions.

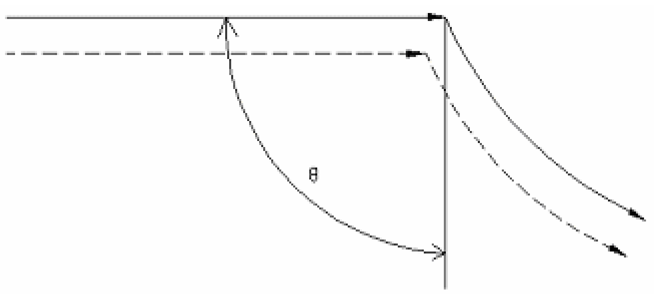
g. The inner angle is less than the θ preset by parameter.

The angle judgment with regard to programmed path:

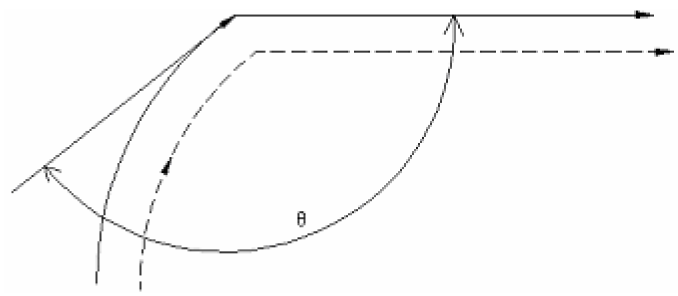
(I) Straight line ——— Straight line



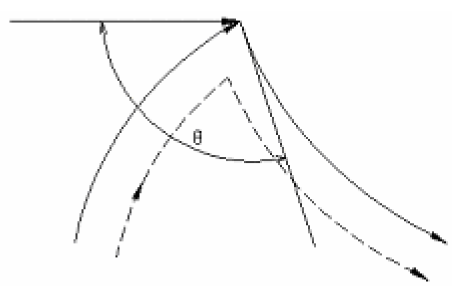
(II) Straight line Arc



(III) Arc—— Straight line



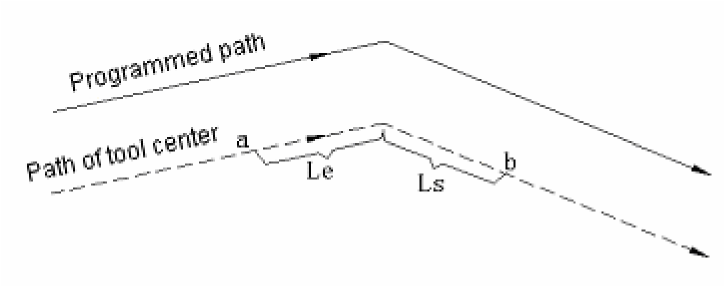
(IV) Arc ——— Arc



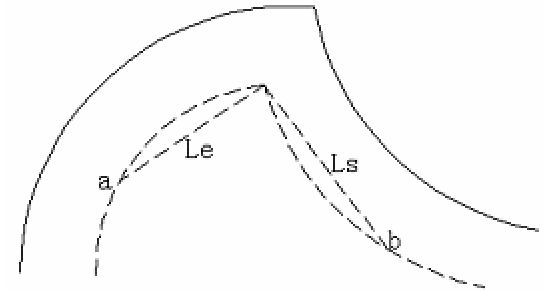
The angle may be considered as an inner angle when θ≤θP. The value of θP shall be set by parameter (NO·335) (1°≤θP≤179°). It is subject to error in judgment less than 0.001° provided that θ is nearly equal to θP.

Operation range

When an angle is determined as an inner angle, feedrate is regulated from the range specified by the Lθ in the block on one side of the intersection point of the angle to the range specified by the Ls in the block one the other side of the intersection point. Ls and Lθ are the straight distances from one point on the path of the tool center to the intersection point. L θ and Ls are set by different parameters (N0·355 and 356).



Feedrate is regulated in the range from point a to point b.



Feedrate is regulated within the range from Point a to Point b.

For an arc, the adjustment function is effective for the end point of a block provided that the following requirements are fulfilled.

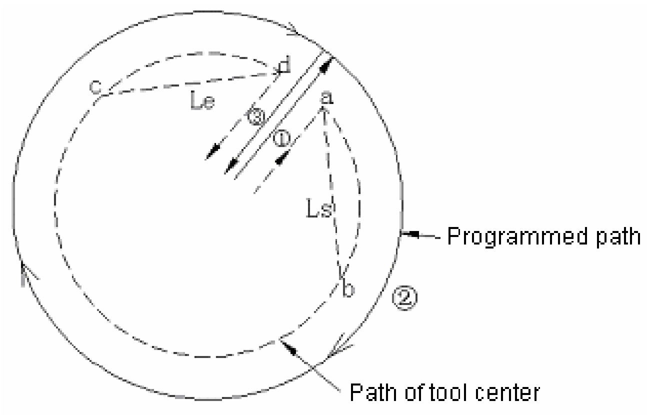
① Within the range of Lθ;

② The origin and end point of the arc are in the same quadrant or the origin is in the quadrant adjacent to the one in which the end point is. Similarly, the adjustment of the origin of the block is effective when the following conditions are met.

① Within the range of Ls;

② The origin and end point of the arc are in the same quadrant or the origin is in the quadrant adjacent to the one in which the end point is.

(Example) For a disc:



For the program ② for an arc, feedrate is regulated from point a to point b and from point c to point d.

Amount of adjustment

Amount of adjustment is set by No. 335 parameter.

1≤amount of adjustment (1% each step) ≤100 (%)

This is effective for commissioning and F1 digit instruction. For F4 digit instruction, the actual feedrate will become:

Fx (inner angle adjustment) × (feedrate adjustment)

Whether inner angle adjustment is effective

Whether to adjust an inner angle or not may be selected with G codes. The inclusion of G62 in the group 15 in which G61 and G64 are is as shown in the table below. These G codes are related to accurate stop checking mode.

	Accurate stop detecting mode	Inner angle adjustment
G61	Valid	Invalid
G62	Invalid	Valid
G64	Invalid	Invalid

Note 1: It is in G64 mode when powered on or cleared.

Note 2: It is necessary to specify G09 when you plan to perform accurate stop detection in G62 mode.

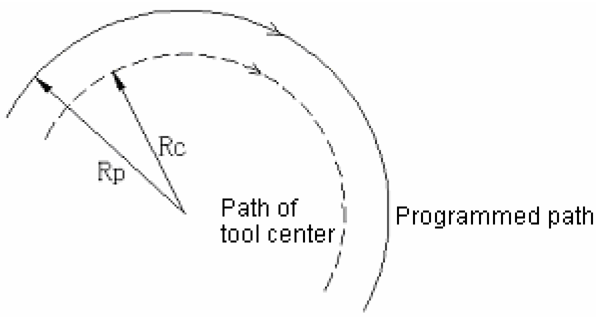
Note 3: The switch of inner arc cutting feedrate is always valid as described in Section 4.7.2 and is not subject to the influence of G codes.

3.4.7.2 Switch of inner arc cutting feedrate

For inner arc offset cutting, the feedrate of programmed path shall be specified by F codes while its actual feedrate $F \times R_C/R_P$ (where R_C is the radius of the path of tool center and R_P the radius of program path.

The switch is also effective for commissioning and F1 digit instruction.

(Example 1)



However, if R_C is much less than R_P, i.e. $R_C/R_P=0$, the tool will stop. Therefore, when $R_C/R_P \leq MDR$ after the minimum reduction ratio (MDR) is set, the actual feedrate will be FX (MDR).

MDR is set by N0·333 parameter. When 1≤MDR (1% each step)≤100, it is applicable for F1 digit and commissioning. The reduction ratio of automatic inner angle adjustment is not under the influence of MDR.

Note: If inner arc cutting is overlapped on automatic inner angle adjustment, now the actual feedrate will be F× $\frac{Rc}{Rp}$ × (angle adjustment)× (feedrate override).

3.5 Preparation Function (G function)

The two digits following address G determine the meaning that the concerned block involves.

G codes may be divided into the following two types:

Type	Meaning
Primary valid G codes	It is only valid in its instructed block.
Modal G codes	After being instructed to be valid, the G codes become invalid only when another G code in the same group is instructed.

(Example) G01 and G00 are modal G codes.

G01 X ______ ;
Y ______ ;
X ______ ;

G01 is valid in this range.

G00 Y ______ ;

Table 5.1: G codes list

G codes	Group	Function
G00	01	Positioning (rapid traverse)
G01		Linear interpolation (feed)
G02		Arc interpolation CW (clockwise)
G03		Arc interpolation CCW (counterclockwise)
G04	00	Hold
G07		Speed sine-curve control (specified imaginary axis)
G09		Accurate stop detection
G10		Offset setting, workpiece zero offset setting
G17	02	To select XY plane
G18		To select ZX plane
G19		To select YZ plane
G20	06	Input in inch
G21		Input in mm
G22	04	Storage travel limit ON

G23		Storage travel limit OFF
G27	00	Reference point return check
G28		To return to reference point
G29		To return from reference point
G30		To return to the 2nd, 3rd and 4th reference points
G31		To skip over cutting
G33	01	Thread cutting
G40	07	To cancel tool compensation
G41		Tool compensation – left side
G42		Tool compensation – right side
G43	08	Forward compensation of tool length
G44		Reverse compensation of tool length
G49		To cancel tool length compensation
G45	00	To increase tool offset
G46		To reduce tool offset
G47		To increase tool offset by twice
G48		To reduce tool offset by twice
G50	11	Zooming OFF
G51		Zooming ON
G54	14	To select workpiece coordinate system 1
G55	14	To select workpiece coordinate system 2
G56		To select workpiece coordinate system 3
G57		To select workpiece coordinate system 4
G58		To select workpiece coordinate system 5
G59		To select workpiece coordinate system 6
G60	00	unidirectional positioning
G61	15	Accurate stop detecting mode
G62		To enable automatic angle adjustment
G64		Cutting mode
G65	00	Simple recall of a microinstruction
G66	12	Modal recall of microinstruction
G67		To cancel modal recall of microinstruction
G73	09	Gun drilling cycle
G74		Reverse-tapping cycle
G76		Finish boring
G80		To disable fixed cycle
G81		Drilling cycle, spotter
G82		Drilling cycle, counterboring
G83		Peck drilling cycle
G84		Tapping cycle
G85		Boring cycle
G86		Boring cycle
G87		Reverse boring cycle
G88		Boring cycle
G89		Boring cycle

G90	03	Absolute value programming
G91◥		Incremental value programming
G92		Absolute zero programming
G94◥	05	Feed per minute
G95◥		Feed per rotation
G98◥	10	To return fixed cycle to the initial point
G99		To return fixed cycle to point R
G96	B	
G97		

(Note 1) The G codes marked with ◥ are the initial G codes of all groups. That is, these G codes are established when the system parameters specifying initial G codes are validated by powering on or pressing the RESET key. For G22 and G23, G22 is selected when power is switched on and G22 or G23 (one of them is valid before reset) is established after reset.

The selection of the state of the initial G codes such as G00, G01, G43, G44, G49, G90, G91 or G94 and G95 shall be set by parameters ((G00, G43, G44, G90 and G95).

For G20 or G21, the valid one is selected before powering off or pressing the RESET key.

(Note 2) The G codes in group 00 are modal and are only valid in the blocks they belong to.

(Note 3) When a G code not listed in the above table is specified or an optional G code not defined by control unit is specified, (N0.0/0) will give an alarm. (N0.0/0) But G38 and G39 are ignored.

(Note 4) Some G codes may be specified in the same block even they do not belong to the same group. When 2 or more G codes than belong to the same group are specified in a block, the lastly specified G code will be valid.

(Note 5) If any G code in group 01 in fixed cycle mode, the fixed cycle will be automatically disabled and the system will be in G80 state. However, the G codes in group 01 are not subject to the influence of the G codes of fixed cycle.

(Note 6) G70 and G71 replaces G20 and G21 (special G codes) by parameter No.008 BIT5 (GSP) setting.

(Note 7) The G codes of all groups are displayed.

3.5.1 Selection of planes (G17, G18, G19)

A plane is selected with the instruction for arc interpolation and tool compensation.

G17……XY plane

G18……ZX plane

G19……YZ plane

Travel instruction is not related to the selection of planes G17/G18/G19. For example, when G17Z—— is specified, Z will move.

3.5.2 Positioning (G00)

Using this code positions the tool at the points programmed by addresses X, Y and Z or A, B, C, U, V and W. Coordinates shall be always specified in absolute instructions while the distance from the origin to the end point in incremental instructions. All axes move the tool at rapid traverse (positioning) rates individually and the tool path is not always straight line during positioning.

2 axes (2 addresses) can be programmed concurrently in a block. But only one can be programmed for the 4th axis.

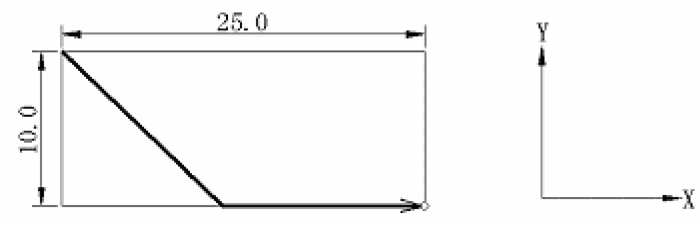
G00 specifies a position.

G00α——β——:

(α • β=X, Y or Z)

Example: The program shall be as follows when the rapid traverse (positioning)rate is 9.600mm/min in the direction of axis X and 9.600mm/min in the direction of axis Y:

G00 X25.0 Y-10.0



Note 1: The rapid traverse (positioning) rate in G00 instruction is set for all axes respectively by manufacturer. Therefore rapid traverse (positioning) rate can not be specified through program.

In the positioning mode of G00, the tool start speeding up until it reaches the predetermined speed. Then it rapidly travels and finally slows down to the end point. When it comes to the “proper position”, the next block is executed in sequence (Note 2).

Note 2: The so-called “proper position” means motor feed is within the specified range (The range is determined by manufacturer). The travel instruction in the following format may be specified provided that the system selects three-axis link function.

G00 X——Y——Z——:

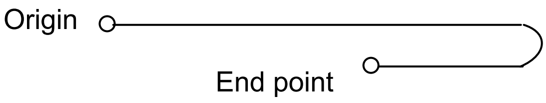
In this example, axes X, Y and Z move the tool to designated position in specified rapid traverse.

When the system selects the linked control function with additional axes, not only addresses X, Y and Z, but also the addresses of additional axes may be instructed. If they are instructed in this way, 3 or 4 axes can move at the same time.

Example: X500.0 Y300.0 Z25.0 B20.0:

3.5.3 Unidirectional positioning (G60)

For the accurate positioning without offset, final positioning can be achieved in only one direction.



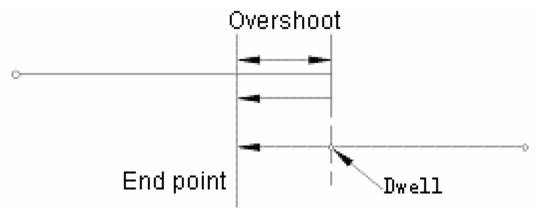
(The direction for final positioning is from right to left.)

The condition that replaces G00 with G60 is as follows:

G60 α——β——γ——δ——:

(α, β and γ= X, Y and Z or additional axes A, B, C, U, V or W. This is the case of concurrent control of 3 or 4 axes. The situation of concurrent control of 2 or 3 axes includes the selection of one additional axis.

Overshoot and positioning direction are set by parameter. The tool stops once before the end point even the positioning direction of instruction is identical with parameter setting.



Note 1: G60 is an one-off valid G code.

Note 2: Axis Z must not be positioned in a single direction in fixed drilling cycle.

Note 3: The axis whose overshoot has not been set by parameter shall not be positioned in a single direction.

Note 4: When the travel distance instruction is O, unidirectional positioning will not be performed.

Note 5: When mirroring function is used, it is invalid to set its direction by parameter.

Note 6: Unidirectional position does not apply to fixed cycles G76 and G87.

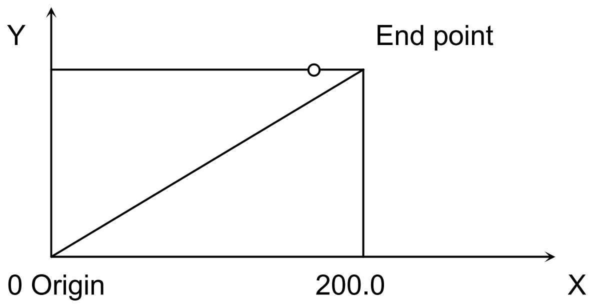
3.5.4 Linear interpolation (G01)

G01α——β——F——:

(α, β and γ=X, Y, Z, A, B, C, U, V and W. Gang control is exerted for 4^{th} and other axes.)

This actually specifies linear interpolation mode. α and β defines the travel distance of the tool. Whether the distance is in absolute or incremental mode depends on the current state of G90/G91. Feedrate shall be specified by F codes, which are modal.

Example of program: (G91) G01 X200.0 Y100.0 F200.0



The feedrate instructed by F codes is the traveling speed of the tool. If F codes are not specified, the feedrate is considered as 0.

The travel instruction (linear interpolation) with 3-axis gang control function is as follows:

G01 X——Y——Z——F——:

It is possible for the instruction to perform linear interpolation for 3 axes at one time.

When additional gang control function is selected, the addresses (A, B or C) of the fourth axis may be used to replace X, Y or Z. In this way the 3-axis gang control involving the 4th axis can be exerted.

Example: G01 X500.0 Y300.0 B20.0 F10.0:

When the system is provided with optional 4-axis gang control functio0n, the use of the following instruction is allowed.

G01α——β——γ——δ——F——:

Where α, β, γ, δ= X, Y, Z, A, B, C, U, V or W.

Note 1: Axial feedrates are as follows:
G01α$\underline{\alpha}$β$\underline{\beta}$ F $\underline{f}$:

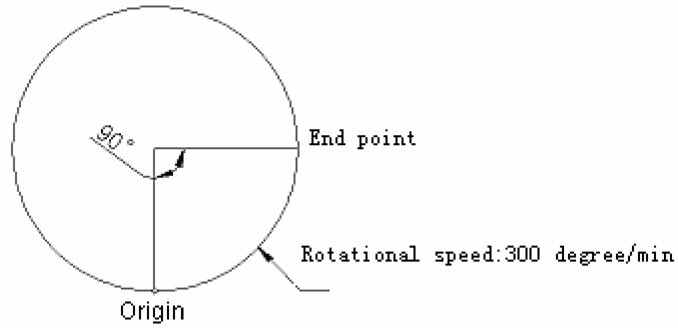
The feedrate in the direction of axis α: Fα= $\frac{\alpha}{L}\cdot f$

The feedrate in the direction of axis β: Fβ= $\frac{\beta}{L}\cdot f$

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

Note 2: The feedrate of an axis of rotation shall be instructed in degree/min (input in metric system: F050; input in inch system: F032).

Example: G91 G01 B90.0 F300:



Note 3: In the linear interpolation involving the 4th axis (axis of rotation A, B or C), the unit of cutting feed is changed from degree to inch (or mm) and the cutting feedrate in α—β rectangular coordinate system is controlled so that it keeps identical with the speed specified by F codes. The feedrate of an axis of rotation is determined with the formulas in Note 1 and its unit is changed into degree/min.

E: G91 G01 X20.0 B40.0 F300.0:

When the unit (degree) of B-axis movement instruction is changed into mm or inch, machining time shall be determined as follows:

$\frac{\sqrt{20^2+40^2}}{300}$ =0.014907 (min)

The feedrate of axis B is:

$\frac{40}{0.14907}$ =268.3 °/min

Note 4: For 3- or 4-axis link, the method for calculating the feedrate in rectangular coordinate system is the same as 2-axis control.

Note 5: For inputting in Inch system and inputting in metric system, the upper limit of the feedrate of an axis of rotation is approximately 6000°/min. The speed is fixed at the upper limit even the feedrate of an instruction exceeds the upper limit.

3.5.5 Arc interpolation (G02, G03)

3.5.5.1 Arc interpolation without any additional axis

The instruction below moves the tool along the arc.

The arc in the X——Y plane:

$$\text{G17}\left\{\begin{matrix}\text{G02}\\ \\ \text{G03}\end{matrix}\right\}\text{X——Y——}\left\{\begin{matrix}\text{R——}\\ \\ \text{I——J——}\end{matrix}\right\}\text{F——;}$$

The arc in the Z——X plane:

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

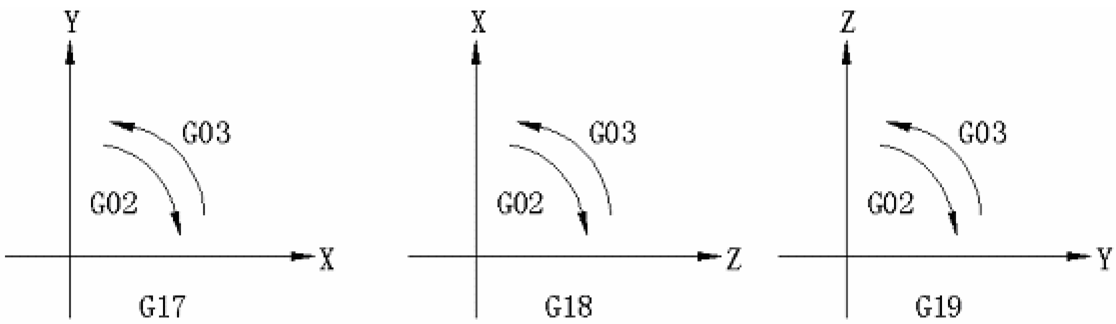
The arc in the Y——Z plane:

$$G19\begin{Bmatrix}G02\\ \\G03\end{Bmatrix}X\text{——}Z\text{——}\begin{Bmatrix}R\text{——}\\ \\J\text{——}K\text{——}\end{Bmatrix}F\text{——}:$$

	Item		**Instruction**	**Meaning**
1	Plane selection		G17	The arc in the plane of XY
			G18	The arc in the plane of ZX
			G19	The arc in the plane of YZ
2	Rotating direction		G02	Clockwise (CW)
			G03	Counterclockwise (CCW)
3	The position of end point	G90 mode	2 of axes X, Y and Z	The position of the end point in the workspace coordinate system
		G91 mode	2 of axes X, Y and Z	The distance from the origin to the end point
4	The distance from the starting point to the center		2 of axes I, J and K	The distance from the origin to the center
	Arc radius		R	Arc radius

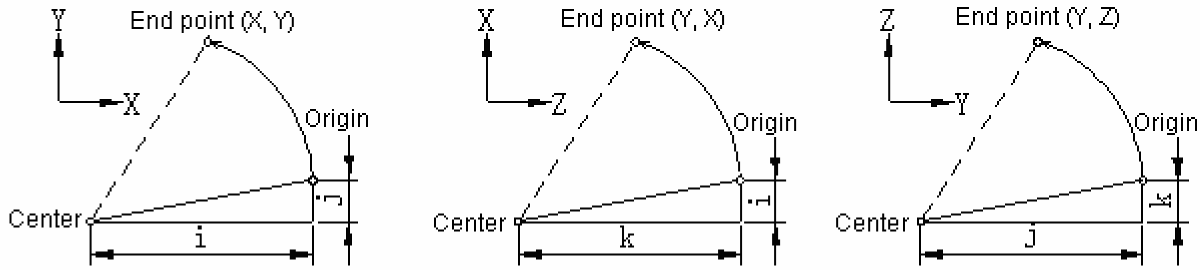
Once the unit is switched on, G17 is enabled as the initial code for plane selection.

Whether it is in clockwise or counterclockwise direction depends on the left-manual or right-manual coordinate system.



The end point of the arc is determined by address X, Y or Z and its indication in absolute or incremental value by G90 or G91. In incremental indication, the coordinate of the end point is specified from the origin of the arc.

The center of the arc is determined by the addresses I, J and K corresponding to axes X, Y and Z. The digit following I, J or K is a coordinate component from the origin to the center of the arc and they are always specified as an incremental value. The provision has no bearing upon G90 and G91.



The marks of I, J and K correspond to the specified directions.

Arc interpolation may substitute I and J with R or gives an instruction with K. The instruction is given in the following format:

$\left\{\begin{matrix} \text{G02} \\ \\ \text{G03} \end{matrix}\right\}$ X——Y——R——:

There are two types of arc in the arc interpolation with R (specified by radius) – the arc less than 180° and more than 180°. Its analysis is as indicated in the diagram below.

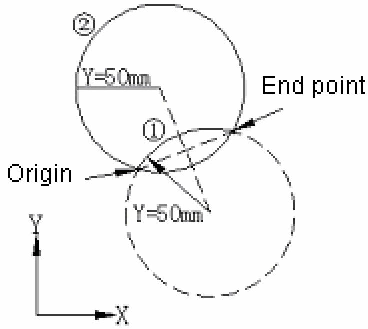
Example of Instruction:

1. The arc less than 180°:

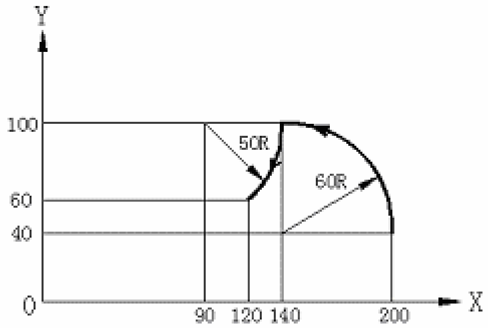
   G02 X6.0 Y2.0 R5.0:

2. The arc over 180°:

   G02 X6.0 Y2.0 R-5.0:



Example of program



a) Absolute programming

(1) G92　X200.0　Y40.0　Z0;

G90　G03　X140.0　Y100.0　I-60.0　F300.0;
G02　X120.0　Y60.0　I-50.0;

(2) G92　X200.0　Y40.0　Z0;

G90　G03　X140.0　Y100.0　R60.0　F300;
G01　X120.0　Y60.0　R50.0;

b) Increment programming

(I) G91　G03　X-60.0　Y60.0　I-60.0　F300;

G02　X-20.0　Y-40.0　I-50.0;

(II) G91　G03　X-60.0　Y60.0　R60.0　F300;

G02　X-20.0　Y-40.0　R50.0;

The tangential feedrate of arc interpolation is equal to the cutting feedrate specified by F codes. However, the arc interpolation involving the fourth axis is not allowed.

Note 1: In arc interpolation, I0, J0 or K0 may be omitted.

Note 2: When the end point of an arc coincides with its origin, I, J and K are used to instruct a center of a circle for programming a 360° arc (full circle) and X, Y and Z may be omitted.

Note 3: It will give No. 023 alarm if an arc of radius of 0 is programmed.

Note 4: The deviation of instruction feedrate from actual tool feedrate is less than or equal to ±2%. In tool radius compensation, the actual tool feedrate is the speed of the tool center path.

Note 5: If the addresses I, J, K and R are assigned to the same block, the arc specified by address R is valid and other I, J and K are ignored.

3.5.5.2 Arc interpolation with an additional axis

The arc interpolation with an additional axis is allowable. An axis (X, Y or Z) shall be set to parallel with the additional axis by parameter setting. If the additional axis does not parallel with any axis, arc interpolation will be impractical. G code for plane selection shall be specified for arc interpolation. An address of an axis is specified with the G code for plane selection so as to determine the axes performing arc interpolation.

Example: Assuming that additional axes U and W parallel with axes X and Y respectively.

a) G17X-Y-…………………XY plane

b) G17U-Y-…………………UY plane (U parallels with X)

c) G17Y-……………………XY plane

d) G17………………………XY plane

e) G17 X-Y-U-……… …Alarm

f) G18X-W-………………XW plane (W parallels with Z)

Addresses I, J and K may also be used to specify the center of the arc. This is similar to the arc interpolation without any additional axis. The addresses I, J and K are used for the axes parallel to axes X, Y and Z.

The arc interpolation specified with R is also valid.

3.5.6 Sine-curve interpolation

In spiral cutting instruction, sinusoidal interpolation is realized by specifying an arc instruction axis not to move during arc interpolation. The imaginary axis is specified as follows:

G07α0: (Specify α as the imaginary axis)

G07α1: (Specify α as the real axis)

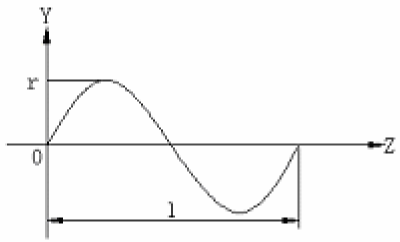
(α= X, Y, Z or additional axes A, B, C, U, V and W)

After instruction G07α0, axis α is deemed as an imaginary axis until G07α1 instruction is given.

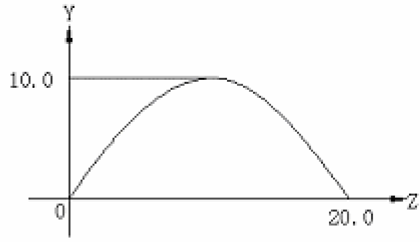
For the single-period sine interpolation in Y-Z plane, axis X serves as an imaginary axis.

$X^2+Y^2 = r^2$ (r: arc radius)

$Y = r\sin(\frac{2\pi}{\ell})Z$ (ℓ : the travel distance along axis Z in a single period)



Example of program



N001　G07　X0:
N002　G91　G17　G03　X-200　Y0.0　I-10.0　Z20.0　F100:
N003　G01　X10.0:
N004　G07　X1:

Axis X is deemed as an imaginary axis during N002～N003 blocks.

In the N003 block, spiral-curve cutting instruction is given in this way when axis Z is used a linear axis. However, axis Y moves only when axis Z is performing sine interpolation since axis X does not move.

In N003 block, the machine is in suspended state after interpolation because axis X does not move.

Note 1: Imaginary axis is only available for automatic operation but not for manual operation.

Note 2: Interlock, travel limit and external deceleration are also available for imaginary axis.

Note 3: Manual insertion is also valid for imaginary axis. That is, the axis moves by manual insertion.

3.5.7 Thread cutting (G33)

The possibility of cutting determines the threads of screw pitch.

G33Z Z F: f:

Where Z is the length of thread (incremental instruction) or end point of thread (absolute instruction)

f: screw pitch

	Minimum input increment	**Range**
Input in mm	0.01mm	F1 to F50000 (0.01mm to 500.00mm)
Input in inch	0.0001inch	F1 to 500000 (0.0001inch to 50.0000inch)

Spindle axis is restricted as follows:

1≤R≤(Maximum feedrate/screw pitch) or allowable rotational speed of the position coder)

Where:

R: Spindle speed (r/min)

Screw pitch: mm or inch

Maximum feedrate: mm/min or inch/min

The instruction signal of maximum feed per minute or the maximum feedrate due to the restriction of motor or machine, whichever is lower;

Allowable rotational speed of position coder: 4,000r/min (position coder A)

6, 000r/min (position coder B)

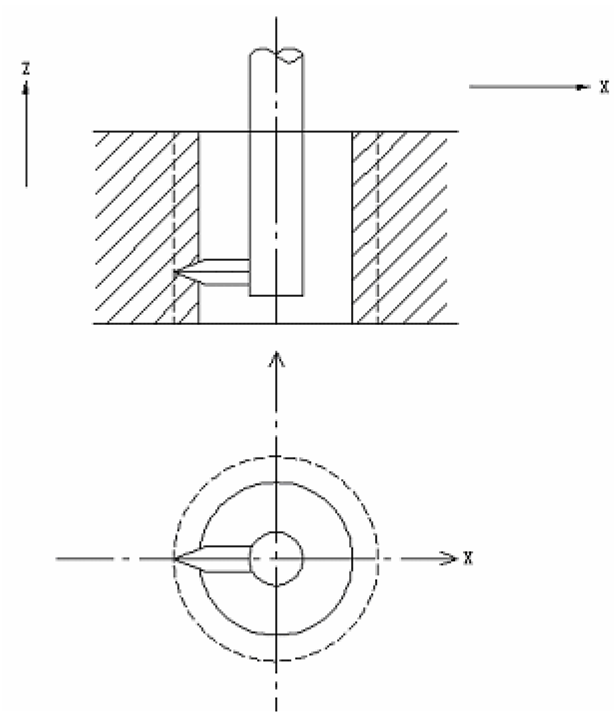
Note 1: Spindle speed may be read continuously through the position coder installed on the spindle. The coder converts the spindle speed into the cutting feed per minute for purposes of feed.

Note 2: The converted cutting feedrate is not overridden but fixed at 100%.

Note 3: The converted cutting feedrate shall be fixed.

Note 4: Feed hold is invalid during thread cutting.

Example:



N20 G90 G00 X100.0 Y… S45 M03;

N21 Z200.0 ;

N22 G33 Z120.0 F5.0 ;

```
N23                                              M19;
N24  G00    X105.0                               M03;
N25                    Z200.0                    M00;
N26         X100.0                               M03;
N27  G04    X2.0                                    ;
N28  G33               Z120.0  F5.0                 ;
```

Notes:

N20, N21: To locate the tool at the center of the hole and to clockwise (CW) rotate the spindle.

N22: To perform thread cutting for the first time with screw pitch specified by address F.

N23: M19 orders the spindle to stop in a fixed position on the circumference (M19: Spindle stops in a fixed position).

N24: To withdraw the cutting tool in the direction of axis X.

N25: To move the cutting tool above the hole: M00 orders the program to stop and allows operator to adjust the tool for the thread cutting for the second time.

N26: To align the cutting tool with the center of the hole and to start forward rotation of the spindle.

N27: When the travel instruction in the N26 block is relatively short, it is necessary to add another dwell instruction so that the spindle has adequate time to reach the rated rotational speed.

N28: To perform thread cutting for the second time.

3.5.8 Automatic return to reference point (reference positions G27 ~ G30)

3.5.8.1 Check of return to reference point (G27)

A point fixed on a machined plane is called reference point (reference position). Once a tool is manually returned to a reference point, it is positioned at the point.

G27 instruction function is used to check that the tool is positioned at the reference point.

G27α—β—:

(α, β are selected from address X, Y and Z and additional axes A, B, C, U, V and W) The instruction is used to rapidly position the tool at the reference point.

Once the tool reaches the reference point, the indicator indicating controllable axis’s return to the reference point is lit.

After returning to the reference point, the next block will be executed if M00 or M01 does not exist in M00 or M01. If the return to the reference point is not required in all cycles, optional program skip function can be used.

If the system is provided with 3-axis gang control function, the G27 instruction may be expressed in the following format:

G27α—β— r —:

(The addresses α, β and r shall be selected from X, Y, Z and additional axes A, B, C, U, V and W. However, only one axis can be controlled at one time if concurrent control of additional axes is not selected.)

The use of the following instruction is allowable when 4-axis gang control is selected:

G27α—β—r—δ—:

Where α, β, r, δ= X, Y, Z, A, B, C, U, V or W

Note 1: In tool compensation mode, the position that the tool reaches with G27 is the position with offset. In this case, the tool is also not at the reference point and the indicator for return to the reference point does not illuminate. As a rule, G27 is only used for compensation cancellation mode.

Note 2: In a inch mechanical system that inputs in metric system, the indicator illuminates even the programmed position of the tool offsets from the reference point by 1μ as a result of that the minimum input increment is less than the minimum traveling increment of the mechanical system.

3.5.8.2 Automatic return to reference point (G28)

G28 α—β—:

(The addresses α and β shall be selected from X, Y, Z and additional axes A, B, C, U, V and W. However, additional axes can only be individually controlled without additional axis gang control function.)

The axis specified by the instruction can be automatically positioned at the reference point. As move instructions, α and β shall be specified in absolute/incremental value in G90/G91 mode.

The end point of the instruction is called “intermediate point” and the coordinates specified by this instruction are saved to the NC.

The procedures in the G28 block are as follows:

First all controlled axes are positioned at intermediate points rapidly. Then they move from the intermediate points to the reference points. If now the machine is not locked, the indicator for return to the reference point is lit.

The positioning at the intermediate point and reference point in this way is equivalent to the positioning of G00.

The instruction for 3-axis link is as follows:

G28α—β— r —:

(The addresses α, β and r shall be selected from X, Y, Z and additional axes A, B, C, U, V and W.) The following instruction is allowed for 4-axis link function.

G28α—β— r —δ—:

Where α, β, r, δ= X, Y, Z, A, B, C, U, V or W

Generally the G28 instruction is used for automatic tool change (ATC). In principle, tool radius compensation, tool offset compensation and tool length compensation shall be cancelled before executing the instruction.

Note 1: Not only the coordinates of move instruction, but also the coordinates of intermediate point is saved to memory in the blocks of G28. This is, for the axes without instruction in the blocks of G28, the foregoing coordinates in G28 serves as the coordinates of the intermediate point of the axis.

Example: N1 G90 X100.0 Y200.0 Z300.0:

N2 G28 X400.0 Y500.0:
N3 G28 Z600.0:

Remarks:

N2: Intermediate point - 400.0, 500.0

N3: Intermediate point - 400.0, 500.0, 600.0

Note 2: After the system is switched on without manual return to the reference point, it moves from the intermediate point when G28 is specified. This is similar to the manual return to the reference point. Now the moving direction from the intermediate point has become the direction of return to the reference point set by parameter.

Note 3: If G28 is instructed for an axis of rotation, the moving direction from the intermediate point to the reference point has become the direction of return to the reference point. Furthermore, the travel is within 360°.

3.5.8.3 Automatic return from reference point (G29)

G29α—β—:

(The addresses α and β shall be selected from X, Y, Z and additional axes A, B, C, U, V and W. However, additional axes can not be interlocked with one the three basic axes if additional axis gang control function is not selected.)

The tool may be positioned at the specified point through an intermediate point using this function. As a rule, the instruction is used after a G28 instruction.

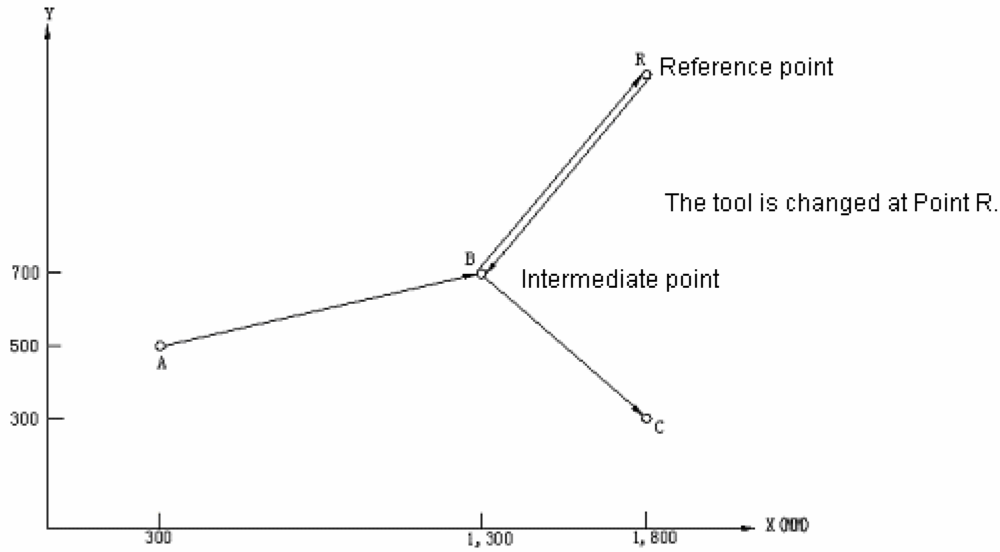
A and β are specified in absolute/incremental value depending on the current G90/G91 state.

In an incremental instruction, an incremental value relative to the intermediate value shall be specified.

Using the movement of the blocks of G29 allows all instructed axes to pass through the intermediate point previously defined by G28 instruction and to reach the specified point at rapid traverse (positioning) rate.

The procedures for positioning at the intermediate point and then the specified point are similar to the positioning with G00.

Example of G28 and G29 application:



For G91:

G28 X1000.0 Y200.0: (From A to B)

M06:

G29 X500.0 Y-400.0: (From B to C)

The example shows that programmer needs not to calculate the actual travel distance from the intermediate point to the reference point. The G29 instruction for a system with 3-axis concurrent control function is as follows:

G29α—β— r —:

(The addresses α, β and r shall be selected from X, Y, Z and additional axes A, B, C, U, V and W. However, additional axes can not be interlocked with one the three basic axes if additional axis concurrent control function is not selected.)

The following instruction is allowed for the system with 4-axis link function:

G29α—β— r —δ—:

Where α, β, r, δ = X, Y, Z, A, B, C, U, V and W

Note: After the tool passes through the intermediate point and reaches the reference point under the instruction of G29/30, the intermediate point shall move to a new coordinate system in case of change in the position of workpiece coordinate system. Thereafter the tool shall pass through the intermediate point that has moved to the new coordinate system and reach the specified point when G29 is instructed.

3.5.8.4 Return to the 2nd, 3rd or 4th reference point (G30)

The following instruction moves the specified axis to the 2nd, 3rd or 4th reference points (G30).

$$G30 \left\{ \begin{matrix} P2 \\ P3 \\ P4 \end{matrix} \right\} \alpha—\beta—; \text{ (P2 may be omitted)}$$

P2: The 2nd reference point
P3: The 3rd reference point
P4: The 4th reference point
Determining the positions of the 2nd, 3rd and 4th reference points is to preset the distance between the 1st reference points for field adjustment. Except that the tool does not return to the 1st reference point but to the 2nd, 3rd and 4th reference points, the function is similar to G28's instructions of return to reference point. After the G30 instruction, G29 instruction positions the cutting tool to the specified position through the intermediated established by G30 instruction. Its movement is the same as the situation of specifying G29 instruction after G28 instruction.

When the G30 instruction's normal position of automatic tool change (ATC) varies from the reference point, the G30 instruction is as follows if 3-axis link function is provided:

G30α—β— r —:

(The addresses α, β and r shall be selected from X, Y, Z and additional axes A, B, C, U, V and W. However, additional axes can not be concurrently controlled along with one the three basic axes if additional axis concurrent control function is not selected.)

The following instruction may be used when 4-axis gang control function is selected.

$$G30 \left\{ \begin{matrix} P2 \\ P3 \\ P4 \end{matrix} \right\} \alpha—\beta— r —\delta—;$$

Where α, β, r, δ = X, Y, Z, A, B, C, U, V or W
Note: After switching on, it is necessary to manually or automatically return to the reference point once (G28) before executing the G30 instruction.

3.5.9 Dwell (G04)

G04X (t): or

G04P (t):

Either of the above methods may be used for hold. After executing the previous block, It is necessary to wait for (t) seconds before executing the next block.

The maximum instruction time is 99999.999s and time error is about 16ms.

Example: To suspend for 2.5s

G04 X2.5 or G04 P2500:

Note 1: Address P does not program with a decimal point.

Note 2: Dwell and time delay are applicable for the two conditions below and are enabled by parameter setting.

1. After the speed of the foregoing block drops to 0
2. After the tool reaches the instruction value (after the inspection of locating point)

3.5.10 Accurate stop detection (G09)

A block involving G09 reduces its feedrate to 0 at the end point and affirms the state of locating point (Note 2). Then the next block is executed. The function is used to machine a sharp edge or corner. G09 is only valid in its specified block.

Note 1: Locating point detection is automatically performed in the locate modes (G00, G60) without G09.

Note 2 Locating point means that a feed motor is within the range of the specified end point.

3.5.11 Accurate stop detecting mode (G61) and cutting mode (G64)

(1) Accurate stop detecting mode (G61)

The move instructions of all blocks following G61 decelerate to 0 at its end point until G64 instruction. They are determined as being in positioned state and then the next block is executed.

(2) Cutting mode (G64)

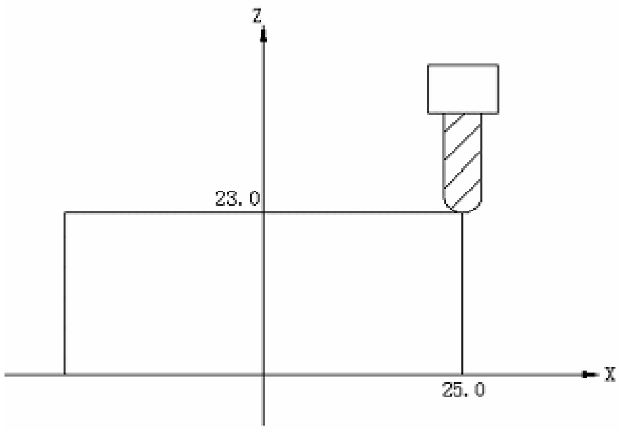
The blocks following G64 do not accelerate until the end point of the move instruction of G61 but immediately go to the next block. Even in the G64 mode, feedrate is reduced to 0 and positioning detection is performed under a positioning instruction (G00 or G60) or in the blocks that has determined (G09) accurate stop detection.

3.5.12 Coordinate system setting (G92)

To move the cutting tool to a specific point with absolute instruction, make sure to preset the coordinate system, which is established with the following instruction.

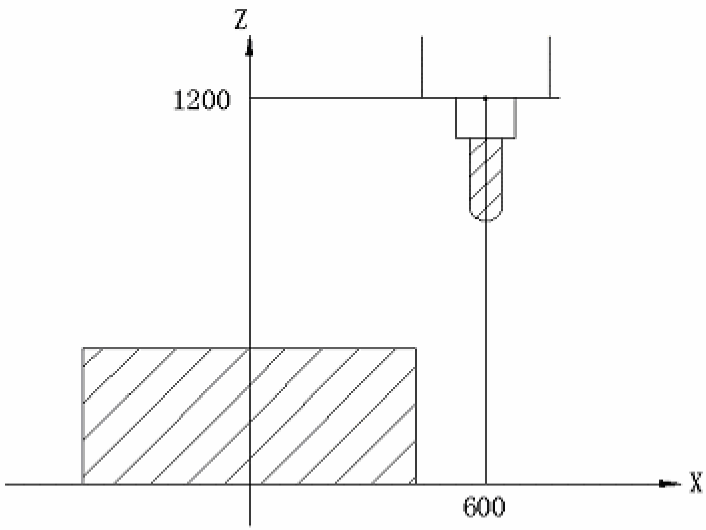
G92X (X) Y (Y) Z (Z) r (r) δ (δ):
(r, δ= A, B, C, U, V, W)

The instruction establishes a coordinate system whose origin is located at a point keeping a specified distance from the tool. The following absolute instruction will refer to the coordinates in the workpiece coordinate system.



G92 X25.0 Z23.0:

As shown in the above program, G92 is used at the beginning of the block to ensure the nose of the tool coincides with the origin of the program.



G92 X600.0 Z1200.0:

As shown in the above diagram, G92 will confirm that the tool point coincides with the origin of the program and execute an absolute instruction. The standard point is positioned at the specified point. To position the nose of the tool to the specified point, the deviation of the nose of the point from the reference point shall be corrected by tool length compensation.

Note 1: In the coordinate system established with G92 instruction in offset mode, the coordinates of the tool in specified position does not include offset.

Note 2: Tool radius compensation is temporarily disabled by G92 instruction.

3.5.13 Workpiece coordinate system (G54 ~ G59)

G92 instruction is not used to establish a workpiece coordinate system. However, the machine's six exclusive coordinate systems may be preset and selected with G54 and 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

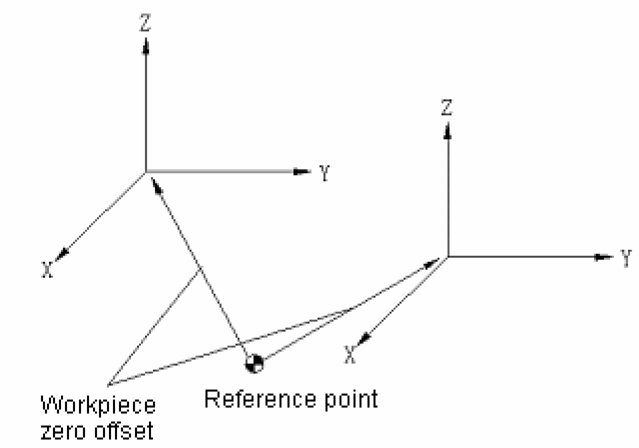
G59…………………Workpiece coordinate system 6

The six coordinate systems are determined by setting all axes' distances (the offset of the zero point of workpiece) from the reference point to their respective points.

Example: G55 G00 X100.0 Z20.0:

X15.5 Z25.5:

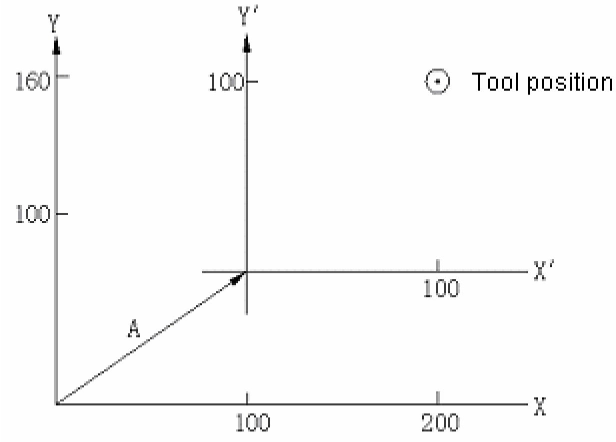
In the above example, it is positioned in the workpiece coordinate system 2 (X =100.0, Z =20.0) and (X =15.5, Z =25.5).



Workpiece coordinate systems 1 to 6 are established after switching on and returning from the reference point. G54 coordinate system is enabled upon switching on.

Note 1: When the range of the workpiece zero point offset of all axes compensated by external data inputting (optional) is 0～±0.7999mm or 0～±0.7999mm, check the machine instruction manual for the function.

Note 2:When G54～G59 are used, the coordinate system is not set with G92 but established with G92. G54～G59 are used to move the coordinate system. G54～G59 must not be incorporated with G92 except the special situation that G54～G59 are used to move the coordinate system.



When the tool is positioned at (200, 160) in G54 mode, G92 X10 Y100; The specified workpiece coordinate system 1 (X', Y') is moved by vector A. However, all other workpiece coordinate systems shall concurrently offset vector A.

Note 3: When automatic coordinate system setting is not selected, proper parameter values shall be set (309 APX～AP4).

3.5.14 To change workpiece coordinate system by program instruction

When the number of workpiece coordinate system becomes inadequate (although 6 are ready) or when they are to be moved as required, we move them through programming instructions.

G10L2P_P_X—Y—Z— r —δ:

Where P=1～6: the 1～6 X, Y, Z, r, δ (r, δ= one of A, B, C, U, V and W) of the corresponding coordinate system: the workpiece zero offsets of all axes.

Note: External workpiece origin offset may be changed by setting P= 0.

3.5.15 Automatic setting of a coordinate system

A coordinate system may be set in the preset parameter number (375PPRTMX～441PPRTI5) when returning to the reference point for the first time after switching on, namely it functions as that G92 automatically sets a coordinate system at the reference point.

Note: If workpiece coordinate system setting function is enabled, it is necessary to set No. 375～378 and 400 parameters to 0 or No. 379～382 and 441 parameters to 0. If they are not set to 0, the workpiece coordinate system (1～6) will offset.

3.5.16 To switch between Inch and metric systems (G20, G21)

Inputting in metric or Inch system may be selected using G codes.

System of units	G codes	Minimum input unit
Inch (Inch system)	G20	0.0001inch
mm (metric system)	G21	0.001mm

Both the G codes must be instructed with a single block before the start of the program and setting of the workpiece coordinate system.

N10 G20:

N20 G92 X—Y—:

The following items changes with the G codes:

(1) The feedrate instructed by F;

(2) Position indication;

(3) Offset

(4) The unit of the scale of the manual pulse generator;

(5) The travel of incremental feed;

(6) A part of parameters

Note 1: Once it is switched on, it enters into the state before power off.

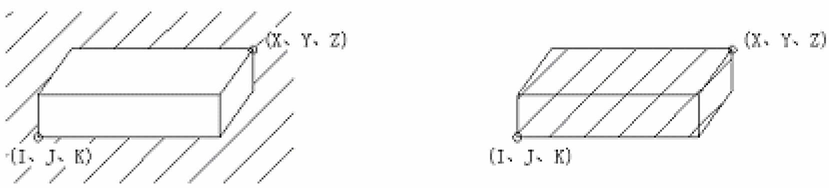
Note 2: Switching between G20 and G21 in the midway of a program is not allowed.

Note 3: When the system of the units of the machine differs from that of the program, the maximum deviation is 1/2 of the minimum traveling unit. The deviation value does not accumulate.

3.5.17 Storage travel limit (G22, G23)

The movable range of the tool may be restricted by the two means below.

(The tool does not enter the shadow area.)



(Exclusion area: outside) (Exclusion area: inside)

Storage travel limit 1:

The outside of the boundary set by parameter is an exclusion area. As a rule, it shall not change once it is set by manufacturer. Therefore it is set at the maximum travel of the machine and is equivalent to the current so-called software limit.

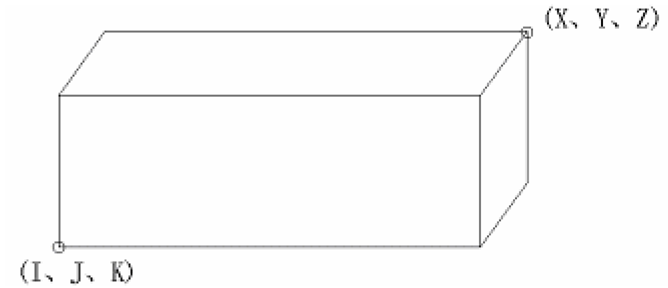
Storage travel limit 2:

The outside or inside of the boundary set by parameter or instruction is an exclusion area. Whether it is inside or outside is determined by parameter No.009 BIT6(RWL).

G22 instruction is used to stop the tool from entering the exclusion area and G23 to cancel the restriction.

The instruction below is used to establish or change an exclusion area.

G22 X—Y—Z—I—J—K—:



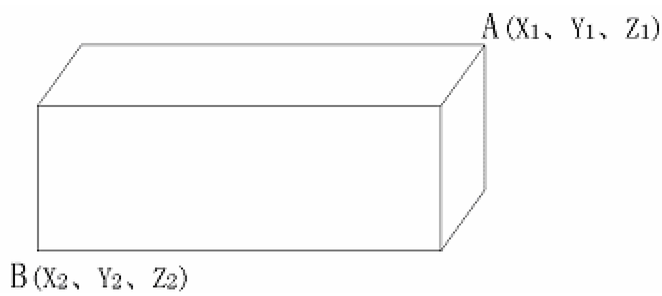
X＞I, Y＞J, Z＞K

X－I＞2000 (Minimum instruction increment)

Y－J＞2000 (Minimum instruction increment)

Z－K＞2000 (Minimum instruction increment)

The points A and B in the figure below shall be involved during parameter setting.



$X_1 > X_2$, $Y_1 > Y_2$, $Z_1 > Z_2$

$X_1 - X_2 > 2000$ (Minimum instruction increment)

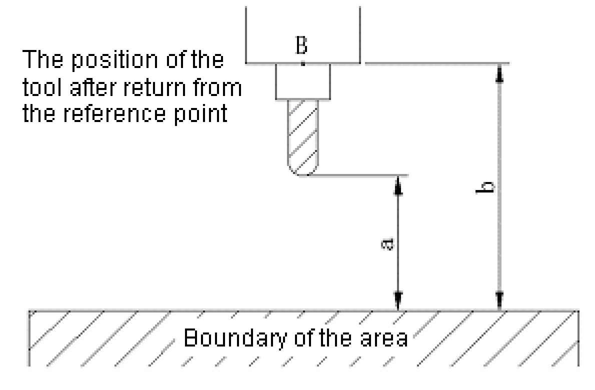
$Y_1 - Y_2 > 2000$ (Minimum instruction increment)

$Z_1 - Z_2 > 2000$ (Minimum instruction increment)

If the exclusion area is set by parameter, X, Y, Z, I, J and K shall be set by the minimum travel unit (input unit) of the mechanical system with the reference as its origin.

If the exclusion area is set by G22 instruction, X, Y, Z, I, J and K shall be set by the minimum travel unit (input unit) of the mechanical system with the reference as its origin. Then the programming data is changed into the numerical value in the minimum traveling unit and the value is set a parameter.

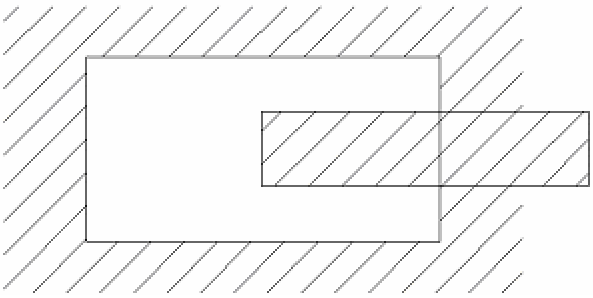
Since it is necessary to detect which part of the tool nose or clamp has entered the exclusion area, the computing methods for X, Y, Z, I, J and K are different.



If point A is detected to enter into the exclusion area, a shall be set. If point B is detected to enter into the exclusion area, b shall be set.

When point A is used to detect the nose of the tool, it is safe without changing the settings of the tool each time if the tool of varied lengths are set to the required length.

Overlapped setting is possible for the area.



Note 1: All limits will become effective after power on and manual return from the reference point or G28 completes automatic return from the reference point.

Note 2: If the reference point is within the exclusion area of all limits, the system immediately gives an alarm when the storage travel limits are valid after power on and manual return from the reference point. (For storage limit 2, only G22 mode is applicable.)

When G23 is changed to G22 and the tool is in the exclusion area, the system will give an alarm in the next block.

Note 3: Pressing the emergent stop switch to cancel the restrictions and move the tool out of the exclusion area in G23 mode when the tool remains still in the exclusion area under the conditions of Note 2. The settings shall be changed if they are set incorrectly. Then the tool returns from the reference point.

Note 4: Since the axis without the function of returning from the reference point, the axis does not give an alarm in the exclusion area.

Note 5: For the setting of the exclusion area, the area is described as follows:

When the exclusion area is beyond the specified area, all the areas are movable.

When the exclusion area is within the specified area, all the areas are restricted for movement (in G22 mode).

Note 6: The settings beyond the travel of the machine are not necessary.

Note 7: The tool may reversely move provided that the tool is in the exclusion are and an alarm is given.

Note 8: Even the sequence of the coordinates of the two points In the set area are set incorrectly, the rectangle with the two points as its acmes can also be created into a limit area.

Note 9: G22 and G23 shall be instructed in single blocks.

Note 10: Storage travel limit 2 cannot be used for additional axes.

3.5.18 Skip function (G31)

The G31 followed by a move instruction is capable of instructing linear interpolation as G01. If skip signal is input form the outside in the midway of the instruction, the remaining part of the instruction will be interrupted and the next block will be executed.

G31 instruction is one-off and only valid in the instructed block.

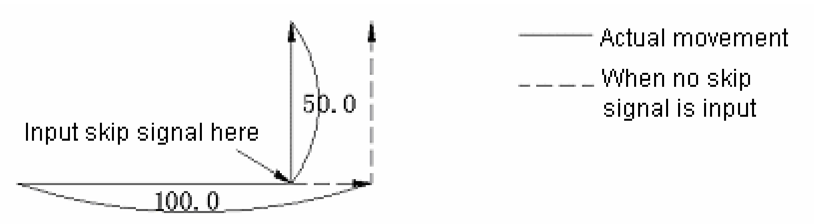
The movement after skip signal depends on whether the next block is incremental or absolute.

1) The next block is an incremental instruction.

To make incremental movement from the break point:

Example: G31　G91　X100.0:

Y50.0:

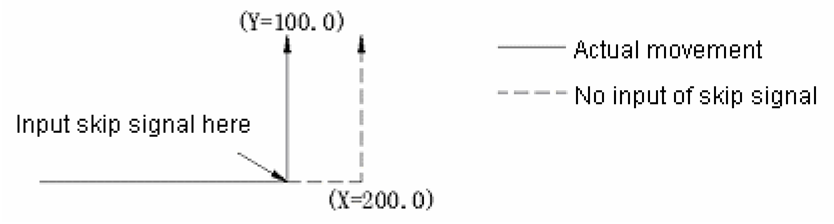


The next block is an absolute instruction (only one axis)

The axis instructed in one block moves to the instructed position while those not instructed remain in the positions input by skip signal.

Example: G31　G90　X200.0:

Y100.0:

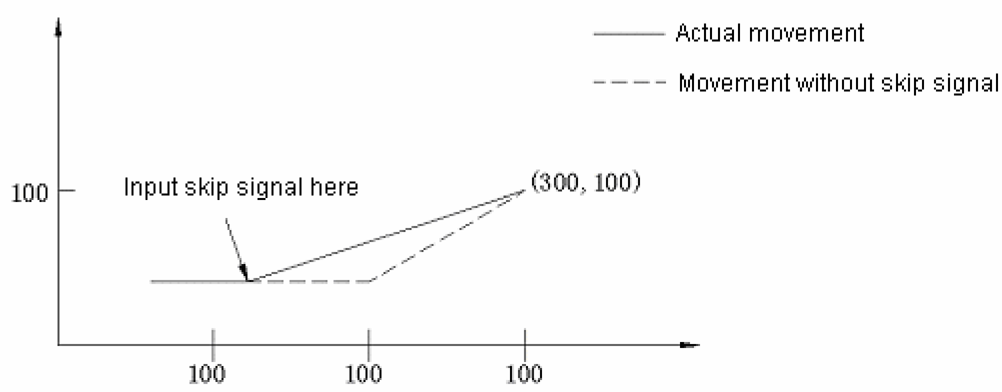


2) The next block is an absolute instruction (to instruct 2 axes)

The next block moves to the instructed position wherever skip signal is input.

Example: G31 G90 X200.0:

X300.0 Y100.0:



The feedrate of a block is specified （G31）with No. 306 (SKPF) parameter in the following two ways:

a) Feedrate is specified with F codes (may be specified in the foregoing blocks or G31 block)

b) No. 342 parameter is set by No. 342 parameter.

The coordinates when skip signal is switched on are stored in the system variables #5061 to #5065 of the user macro. Therefore they may be used in macros.

#5061………………………X coordinates

#5062………………………Y coordinates

#5063………………………Z coordinates

#5064………………………The 4th coordinates

#5065………………………The 5th coordinates

The skip function may be used to the situation with unknown travel. Therefore, it is applicable in the following conditions.

a) For the standard-dimension feed of the milling machine;

b) To enable tool contact pick-off to measure

Note 1: Once G31 instruction is performed in the active state of tool compensation C, No. 035 alarm will be given. Tool compensation shall be canceled with G40 before G31 instruction.

Note 2: If the feedrate instructed by G31 is related to the speed set by parameter, it will have a bearing upon parameter setting in no-load operation.

Note 3: If the feedrate instructed by G31 is related to the speed set by parameter, the automatic acceleration/deceleration will be invalid. In this way the accuracy of automatic measurement in skip function applications is improved.

3.6 Compensation

3.6.1 Tool length compensation (G43, G44, G49)

Using $\left.\begin{array}{l}\text{G43}\\ \text{G44}\end{array}\right\}$ Z—H—; or

$\left.\begin{array}{l}\text{G43}\\ \text{G44}\end{array}\right\}$ H—;

The end position of the axis Z move instruction is moved forward or reversely by setting the offset set in the offset storage. Using this function may set the difference between the tool length estimated in programming and the actual one in service into the offset storage to achieve compensation without changing the program. H instructs the offset set into the offset storage by instructing the offsetting direction with G43 and G44.

Offsetting direction

G43 Offset in + direction

G44 Offset in - direction

Whether in the situation of absolute or incremental instruction, the offset specified by H codes and saved in the offset storage is added to the coordinates of the end point of the spindle move instruction for G43 and detracted from it for G44. The calculated coordinates become the end point of the coordinates. It may be illustrated in the same way when axis Z move instruction is ignored as follows:

$\left.\begin{array}{l}\text{G43}\\ \text{G44}\end{array}\right\}$ G91 ZOH——;

The offset is positive for G43 and negative for G44.

G43 and G44 are modal G codes and, when instructed, are always valid if no G codes in the same group are coded. G 43 or G44 code is valid upon power on depends on parameter setting.

Offset specification

Offset number is specified with H codes. The offset set in the offset storage of the number is added to or detracted from the programming value of axis Z. Offset number may be specified with H00 or H200. The number of the D codes using tool radius compensation hits 32 (It is also possible to choose 64, 99 or 200).

Offset corresponds to offset number and may be preset in the offset storage through MDI/LCD or communicating operations. The setting range of the offset is as follows:

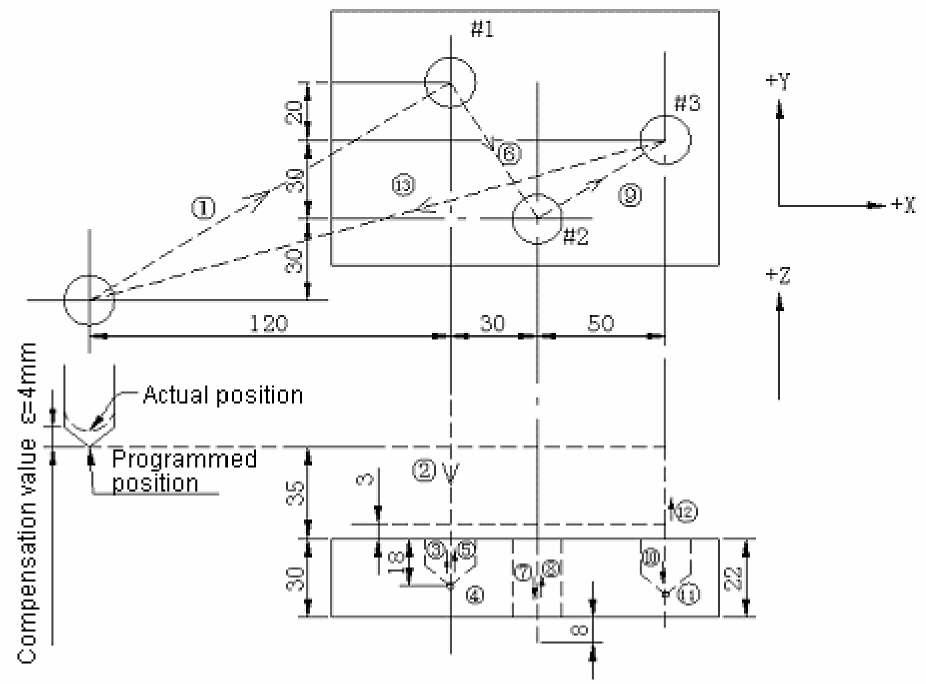
	Input in mm	Input in inch
Offset	0mm～±999.999mm	0 inch～±999.999inch

The offset of the corresponding offset No.00, i.e. H00 is generally 0. The offset corresponding to H00 shall not be set.

Cancellation of tool length compensation

To cancel tool length compensation, G49 or H00 shall be instructed. Once H00 or G49 is instructed, canceling operation will be performed.

(1) Example of tool length compensation (machining of #1, #2 and #3 holes)



H01= －4.0 (Offset)

N1 G91 G00 X120.0 Y80.0 ; ……………①

N2 G43 Z-32.0 H01 ; ……………②

N3 G01 Z-21.0 F1000 ; ……………③

N4 G04 P2000 ; ……………④

N5 G00 Z21.0 ; ……………⑤

N6 X30.0 Y-50.0 ; ....⑥

N7 G01 Z-41.0 ; ....⑦

N8 G00 Z41.0 ; ....⑧

N9 X50.0 Y30.0 ; ....⑨

N10 G01 Z-25.0 ; ....⑩

N11 G04 P2000 ; ....⑪

N12 G00 Z57.0 H00 ; ....⑫

N13 X-200.0 Y-60.0 ; ....⑬

Note 1: When offset is changed due to the change of the offset number, the new offset will not be added to the original offset.

H01...........Offset 20.0

H02...........Offset 30.0

G90 G43 Z100.0 H01: ......Z will hit 120.0

G90 G43 Z100.0 H02: ......Z will hit 130.0

Note 2: D codes cannot be used for tool length compensation. Other axes except axis Z may employ tool length compensation.

Which axis will be added with tool length compensation is instructed with axis address α in the same block with G43 and G44

$$\left.\begin{array}{l}\text{G43}\\ \text{G44}\end{array}\right\}\ \alpha\_\_H\_\_:\quad (\alpha\text{: any axis})$$

Tool length compensation can only be added to an axis at one time. Therefore the system gives an alarm in the following instructions. To switch an axis for tool length compensation, it is necessary to cancel the one-off tool length compensation.

G43 Z__H__:

G43 X__H__: (Alarm)

3.6.2 Tool offset (G45 ~ G48)

The moving distance of the specified axis may be zoomed in or zoomed out by the numerical values set in the offset storage through instructions G45～G48. G codes and their functions are listed in Table 6.2.

Table 6.2: Tool offset and G codes

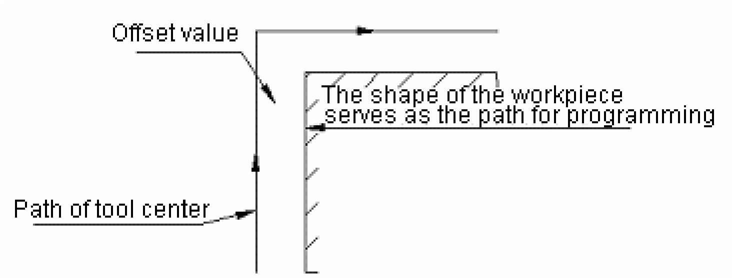
G codes	Function
G45	To zoom in an offset
G46	To zoom out an offset
G47	To zoom in an offset by twice
G48	To zoom out an offset by twice

These G codes are modal and only valid in instructed blocks.

These amounts of compensation are instructed by D or H codes and remain unchanged once they selected before selecting other amounts of compensation.

Whether tool offset compensation uses H or D codes is set by parameter No.010 BIT3 (OFSD).

In offset storage, the shape of the workpiece serves as the path of the tool for programming during the setting of the radius of the tool.



Range of offset

	Input in metric system	Input in Inch system
Offset value	0mm～±999.999mm	0”～±99.9999”
Offset value	0^0～$\pm 999.999^0$	0^0～$\pm 999.999^0$

The offset function is valid for an additional axis (the 4th axis).

When the offset number is 00 (H00 or D00), the offset will also be 0.

Zooming is made in the moving direction of the tool of the axis. When the absolute value instructs, the tool starts to move to the position instructed by the instructions in the blocks of G45～G48 and makes zooming compensation.

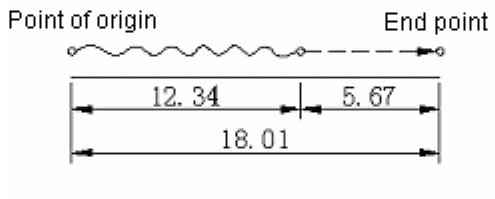
1) G45 instruction (only to lengthen the offset)

～～～～～～～→ Move instruction value

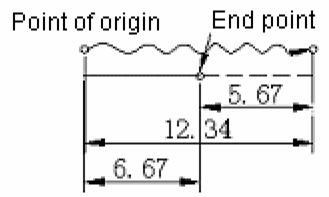
- - - - - - - - -→—→ Offset

Actual travel

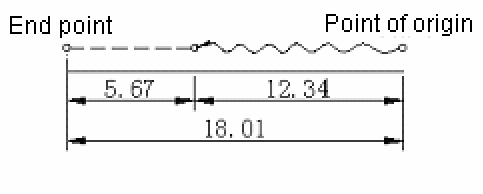
a) Move instruction +12.34 offset +5.67



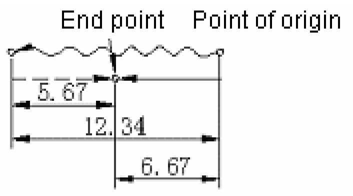
b) Move instruction+12.34 offset



c) Move instruction offset +5.67



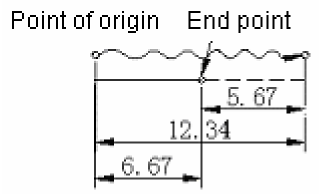
d) Move instruction -12.34 offset -5.67



2) G46 instruction (only to reduce by one offset)

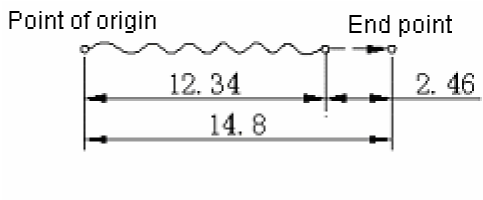
If the sign of the offset is reversed in G45 instruction, it will be identical with G46.

a) Move instruction+12.34 offset+5.67(b)～(d) (omitted)

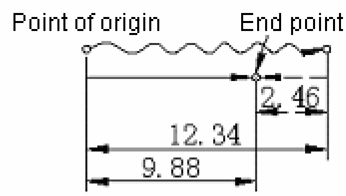


3) G47 instruction (lengthen the offset by twice)

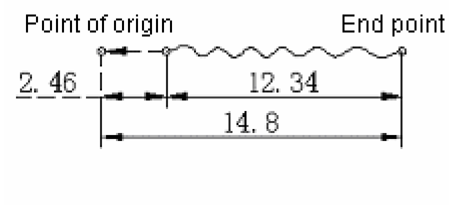
a) Move instruction+12.34 offset+1.23



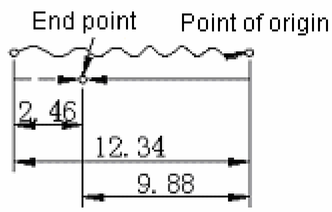
b) Move instruction+12.34 offset-1.23



c) Move instruction-12.34 offset+1.23



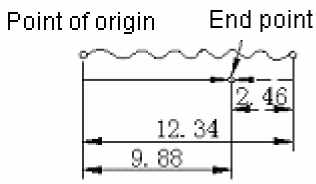
d) Move instruction-12.34 offset-1.23



4) G48 instruction (to reduce the offset by twice)

If the sign of the offset is reversed in G47 instruction, it will be identical with G48.

a) Move instruction+12.34 offset +1.23(b)～(d) (omitted)



When offset is moved in the mode of incremental instruction (G91), the instruction for travel will be 0. No movement will be performed if the travel instructed in the mode of absolute instruction (G90).

Offset +12.34 (offset No. 01)

NC instruction	G91 G45 X0 D01:	G91 G46 X0 D01:	G91 G45 X-0 D01	G91 G46 X-0 D01:
Equivalent instructions	X12.34	X-12.34	X-12.34	X12.34:

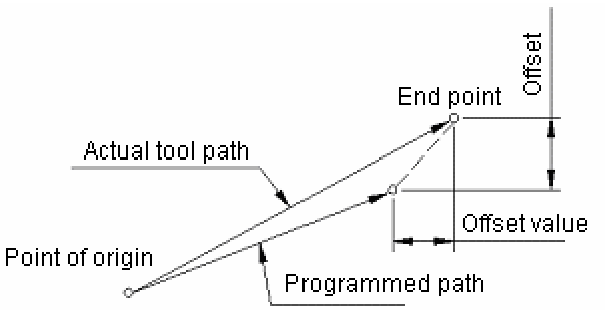
Note 1: If one of G45～G48 is specified for 2-axis gang control, the tool offset is valid for both axes.

For G45

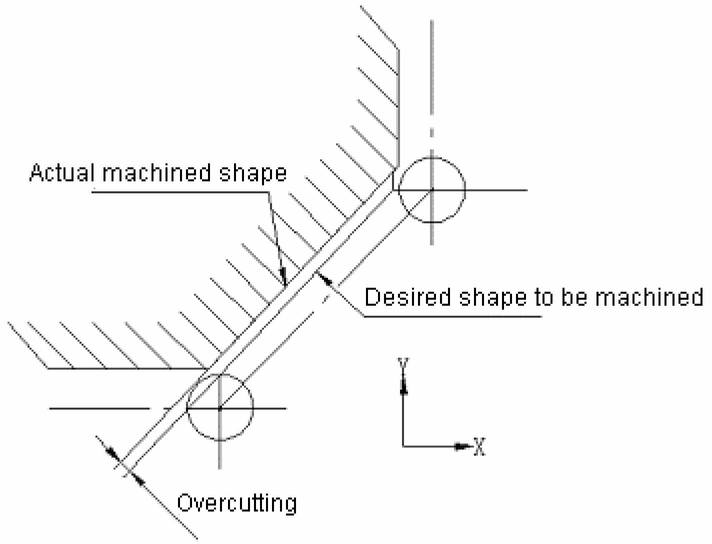
Move instruction X1000.0　Y5000.0

Offset +200.0　 Offset No. 02

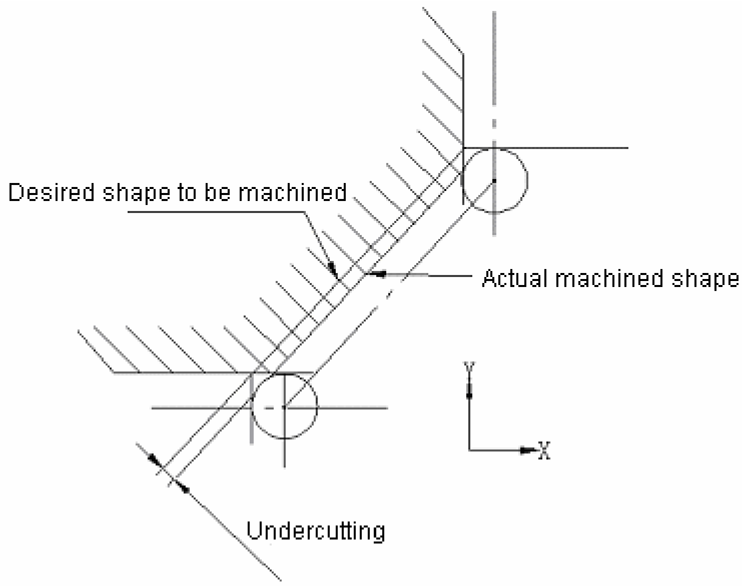
Programming instruction G45　G01　X1000.0　Y500.0　D02:



Note 2: During bevel machining, it is subject to overcutting or under-cutting if tool offsetting is performed.

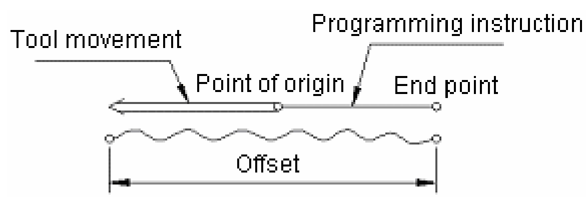



```
G01 X__F__;
G47 X__Y__;
     Y__;
```




```
G01 G45 X__F__D__;
        X__Y__;
G45 Y__;
```

Note 3: When the offset is higher than the move instruction value, the actual moving direction of the tool will be contrary to the programmed direction.



Example: G46　X2.50;
(Incremental instruction)
Offset+3.70
} Equivalent to X-1.20:

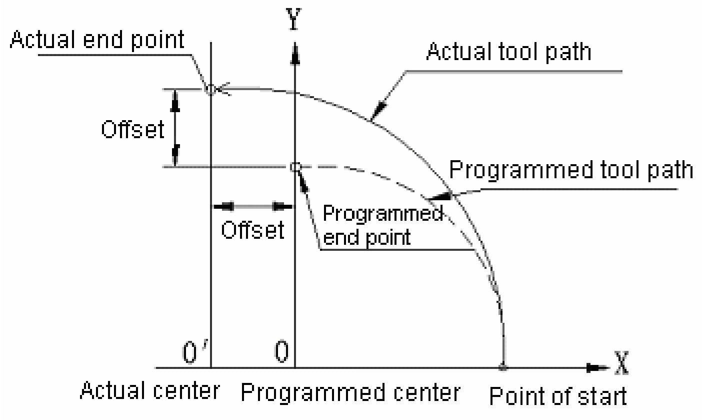
Note 4: For arc interpolation (G02, G03), only the tool offset under the conditions of 1/4 and 3/4 circles may be resulted by G45 to G48 instructions. That is, tool compensation is only available for 1/4 and 3/4 arc instructions.

Example 6.21: Offset +20.0, Offset No. 01

Example of programming:

(G91)

G45　G03　X-70.0　Y70.0　I-70.0　D01:



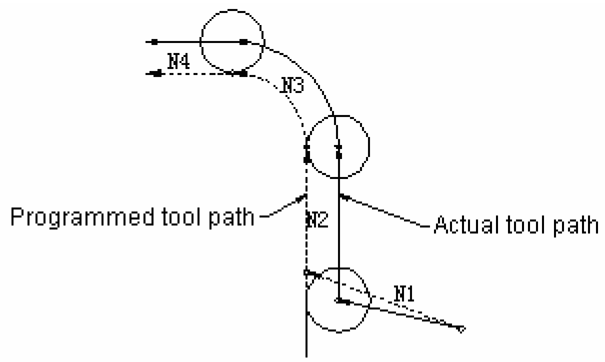
Example 6.22: The tool position offset during arc interpolation

N1　G46　G00　X—Y—D—;

N2　G45　G01　Y—F—;

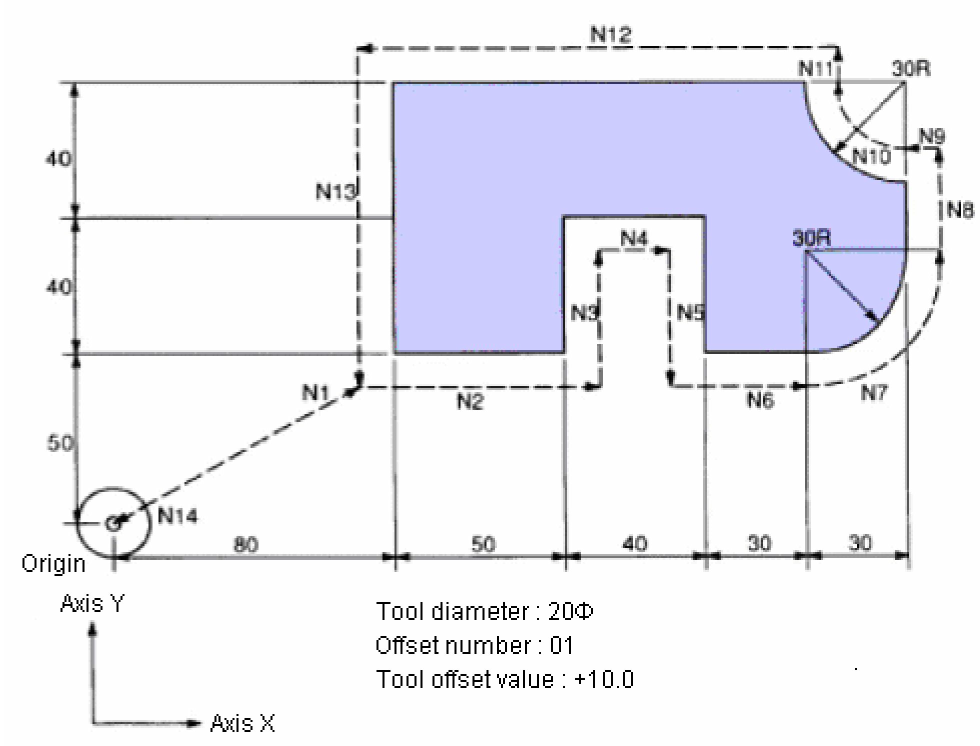
N3　G45　G03　X—Y—I—;

N4　G01　X—;



Example 6.23: The program with tool offset

<u>Tool radius compensation</u>



1. G91 G46 G00 X80.0 Y50.0 D01;
2. G47 G01 X50.0 F120;
3. Y40.0;
4. G48 X40.0;
5. Y-40.0;

6. G45 X30.0;
7. G45 G03 X30.0 Y30.0 J30.0;
8. G45 G01 Y20.0;
9. G46 X0; …… Only to move the offset in the direction of -X
10. G46 G02 X-30.0 Y30.0 J30.0;
11. G45 G01 Y0; …………Only to move the offset in the direction of -Y
12. G47 X-12.0;
13. G47 Y-80.0;
14. G46 G00 X-80.0 Y-50.0;

Note 5: Only axis Z moves an offset if H codes are used in G43 or G44 mode. Therefore, it is recommended not to use H codes but D codes in the offsets of G45～G48 as much as possible.

Note 6: G45～G48 modes are ignored in fixed cycles. Hence G45～G48 shall be programmed before instructing fixed cycles and they must always be cancelled after fixed cycles.

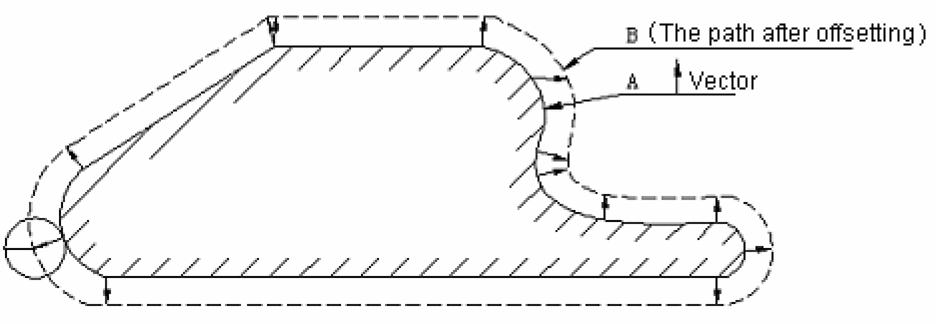
Note 7: Tool offset modes (G45～G48) are not allowed in the modes of G41 or G42 (tool compensation); otherwise P/S alarm will be given (alarm no. 36).

3.6.3 Tool radius compensation (G40 ~ G42)

3.6.3.1 Tool radius compensation function

As shown in the figure below, to machine the workpiece indicated by A in the figure with a tool of radius R, the corresponding tool center must always be B path keeping a distance of R away from A. Moving the tool from the workpiece by one distance in this way is called offsetting. The tool compensating function is used to determine the path (i.e. offset) of the tool that has moved for a distance.

Therefore, programmer may use the tool offset mode to program the outline of a workpiece. Furthermore, if tool radius (offset) is measured and set in the NC during machining, the tool path will be offset (path B) regardless the programmed path.



Offsetting and vector

Two types of tool compensation (B and C) are available. This section only describes type C. The difference between B and C is as follows:

In the mode of type B tool compensation, inward offsetting cannot be performed for angles equal to or less than 90°. In this condition, make sure to program it into a proper inward arc.

3.6.3.2 Offset (D codes)

At most 32 amounts of offset may be set in the offset storage (64, 99 and 200 are optional) (including 32 amounts of offset for tool length compensation and tool position offsetting). The offset depends upon the D instructed in program and the number of digits is set with MDI/LCD.

The range of programmable offsets is as follows:

	Input in mm	Input in inch
Offset	0mm ~ ± 999.999mm	0inch ~ ± 99.9999inch

The offset corresponding to No. 00 or D00 is always 0.

So the offset corresponding to D00 need not be set.

3.6.3.3 Offset vector

Two-dimensional offset vector is equal to the vector of the offset specified by D codes. It is determined in the control unit and its direction is duly corrected in accordance with tool feed of all axes. The offset vector (hereafter called "vector") is generated in the control unit so as to determine the amount of tool offsetting movement and calculate the actual path that the tool radius offsets from the programmed path. The offset vector will be cleared away by reset.

The vector changes with the movement of the tool. It is very important to know the status of the vector during developing a program. Please read the following sections and carefully make certain how the vector is generated.

3.6.3.4 Plane selection and vector

The calculation of offset is performed in the plane established by G17, G18 and G19, which is called offset plane. For instance, (X, Y) or (I, J) is used to calculate the offset as well as vector when XY plane is selected. The coordinates of the axes beyond the offset plane are not under the influence of offset but follows instructed value in the instruction.

In 3-axis gang control, the tool path projected on the offset plane is compensated.

Plane selection switching shall be performed in the mode of offset cancellation. No. 027 alarm will be given if plane selection switching is performed in offset mode.

G codes	Offset plane
G17	X—Y plane
G18	Z—X plane
G19	Y—Z plane

To set an offset plane with an additional axis, it is necessary to preset which one in axes X, Y and Z that the additional axis parallels with by parameter. When none axis it parallels with, the offset plane cannot be defined.

While setting the offset plane with an additional axis, the additional axis shall be instructed at the same time with the G codes, i.e. G17, G18 and G19.

a) G17 X_Y_; ……XY plane

b) G17 U_Y_; ……UY plane (U parallels with axis X)

c) G17 Y_; ………XY plane

d) G17; …………XY plane

e) G17 X_Y_U_; ……Alarm

f) G18 X_W_; ……XW plane (W parallels with axis Z)

3.6.3.5 G40, G41 and G42

G40, G41 and G42 are used to specify the cancellation and generation of a tool radius compensation vector. To determine the direction of an offset vector and the moving direction of the tool, G40, G41 and G42 may be instructed concurrently with G00, G01, G02 or G03.

G codes	Function
G40	To cancel tool compensation
G41	Tool compensation, left
G42	Tool compensation, left

G41 or G42 instruction allows the system to enter into offset mode.

G40 instruction allows the system to enter into cancellation mode.

An example of offsetting course will be illustrated in the following figure.

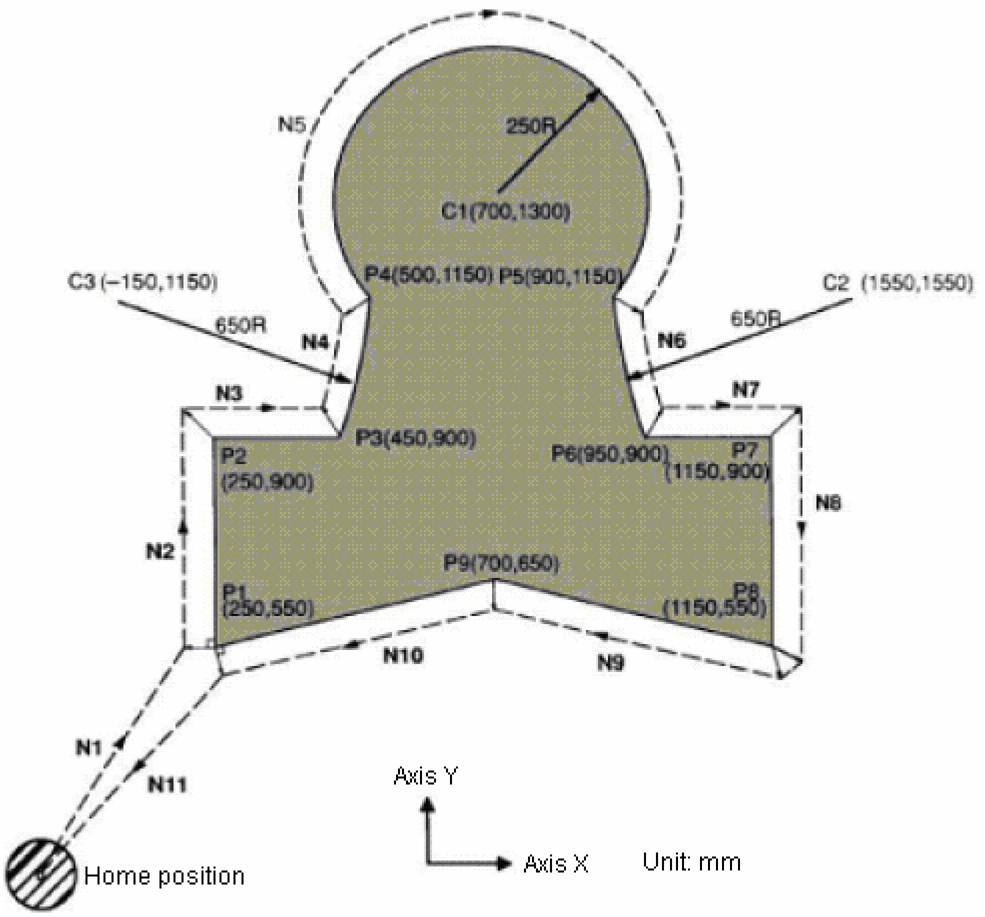
Start-up block ①: Offset cancellation mode becomes offset mode (G41) in this block. The tool center is offset at the end point (P_1) of the block by the radius perpendicular to the lower section of program path (from P_1~P_2). The tool compensating value is specified by D07, i.e. offset No.7. G41 means that the tool offsets to the left.

After workpiece shape $P_1 \rightarrow P_2$——$\rightarrow P_8 \rightarrow P_9 \rightarrow P_1$ programming and starting, the system automatically performs tool compensation.

In block ①, the tool returns to the point of origin (offset cancellation) through G40. The tool center moves (from P_9～P_1) normal to the programmed path at the end point of the 10^{th} block.

G 40 instruction (program cancellation) shall be instructed at the end point of the program.

Example of program for tool compensation C:



G92 X0 Y0 Z0

① N1 G90 G17 G00 G41 D07 X250.0 Y550.0:　(The offset is preset in D07 by MDI)

② N2 G01 Y900.0 F150:

③ N3 X450.0:

④ N4 G03 X500.0 Y1150.0 I-600.0 J250.0:

⑤ N5 G02 X900.0 I200.0 J150.0:

⑥ N6 G03 X950.0 Y900.0 I250.0 J0:

⑦ N7 G01 X1150.0:

⑧ N8 Y550.0:

⑨ N9 X700.0 Y650.0:

⑩ N10 X250.0 Y550.0:

⑪ N11 G00 G40 X0 Y0:

3.6.3.6 Details about tool radius compensation C

The tool radius compensation C is described in detail as follows.

(1)　Cancellation mode

The system will be in offset cancellation mode after power on, reset or completion of the program by executing M02 and M03.

In cancellation mode, the vector is always 0 and the tool center path coincides with the programmed path. It shall be ended up by means of cancellation at the end of the program.

When the program ends in offset mode, positioning the tool at the end point of the program cannot be performed and the position of the tool will offset from the end position by a vector.

(2)　Start-up

When a block satisfying all the following requirements is executed in cancellation mode, the system will be in offset mode. This block is called start-up block.

a) G41or G42 has been instructed and the system is in the state of G41 or G42.

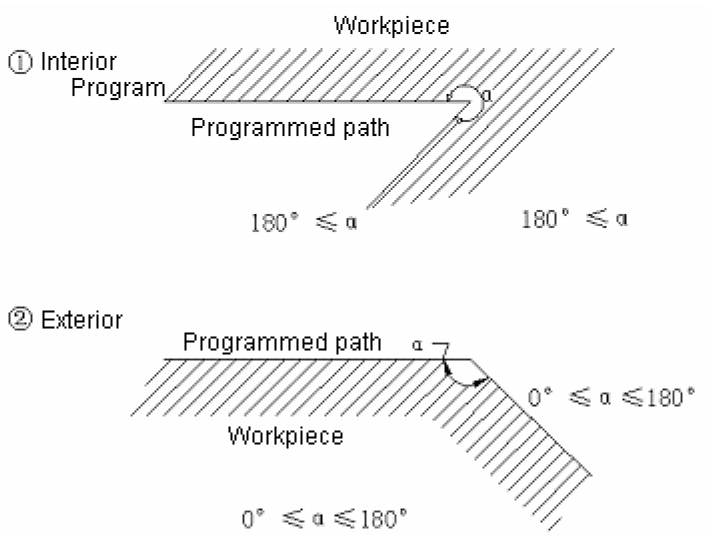
b) Tool compensation number is not D00.

c) Axes (other than axes I, J and K) (even one axis is acceptable) in the offset plane is allowable and their travel is not 0.

In a start-up block, arc instructions (G02, G03) shall not be used; otherwise the NC will gives No. 34 alarm and stops operation. The NC reads in two blocks. The next block enters into the tool compensation buffer register (the contents of the register cannot be displayed) once the first block is read in and executed.

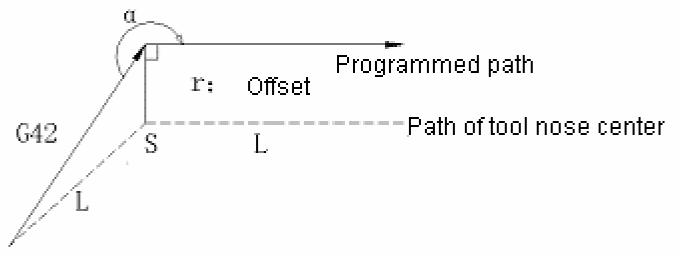
In addition, two blocks in succession are read in when the system is in single block mode. The block is firstly read in stops after execution. As a rule, two blocks are normally read in after that. Hence there are three blocks in the NC, namely the executing block and the next two blocks.

Note: The meaning of the so-called interior angle and other terms are as indicated in the following figure. The intersection angle of the move instructions of two blocks is called "interior angle" when its angular degree on the workpiece side is above 180° and "exterior angle" when it falls within　0°～180°.



(i)　To machine along the interior side

Straight line → Straight line



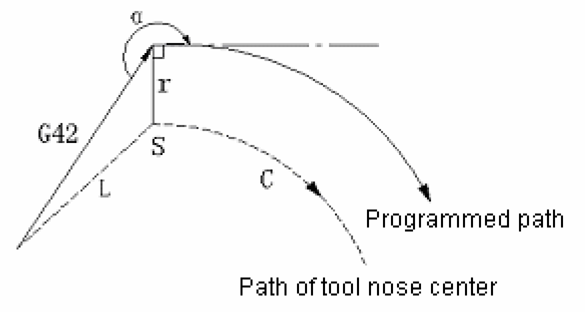
Where:

S indicates the dwell point of single block.

L indicates line motion.

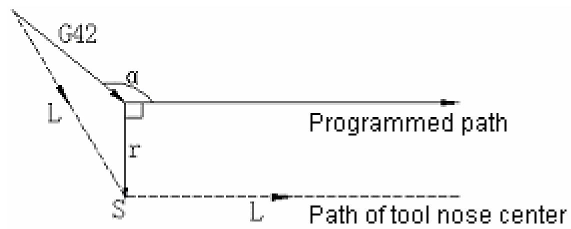
C indicates arc motion.

Straight line → Arc

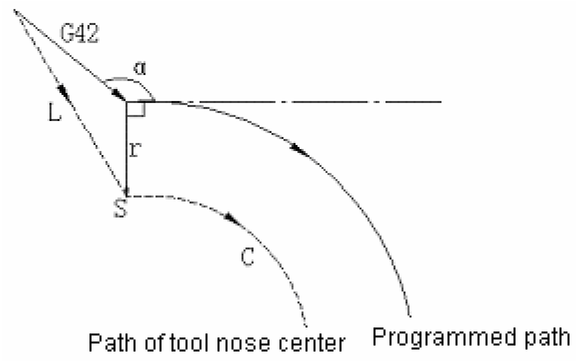


(ii) When the tool machines one side of an obtuse angle, the start-up and cancellation of a (90°≤α≤180°) tool path have the noses A and B and they are selected by parameter No.011 BIT1 (SUPM).

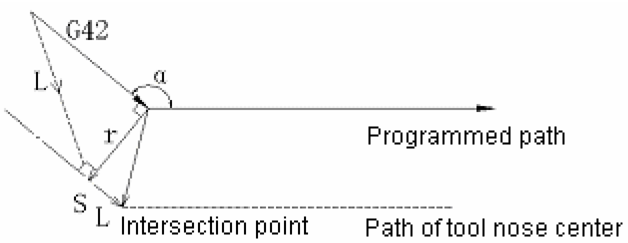
Type A: (Straight line → Straight line)



(Straight line → Arc)

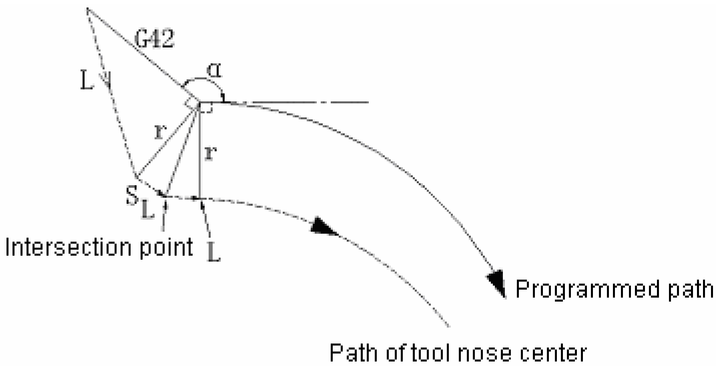


Type B: (Straight line → Straight line)



Intersection point is a point at which the offset paths determined by two blocks in succession intersect.

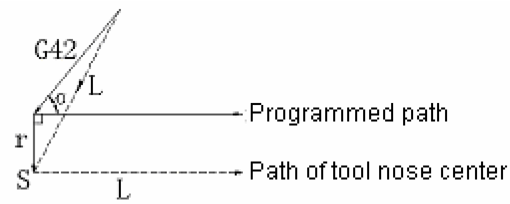
(Straight line → Arc)



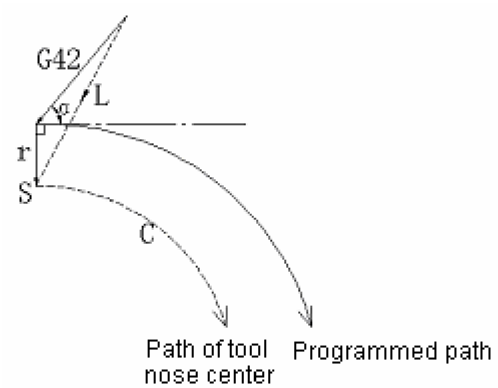
The intersection point in the above diagram is a point where the offset paths of the r length of two blocks.

(iii) To machine an acute angle (α＜90°＝exterior)

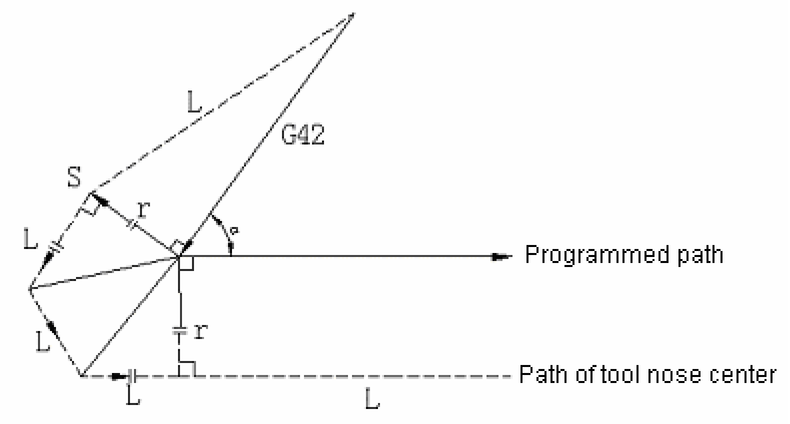
Type A (Straight line → Straight line)



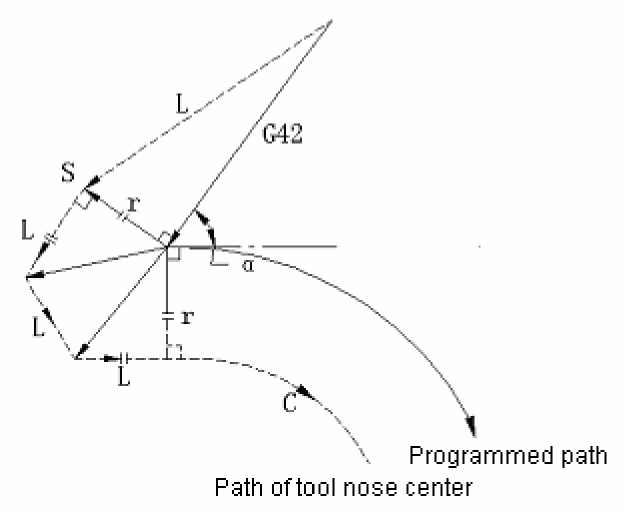
(Straight line → Arc)



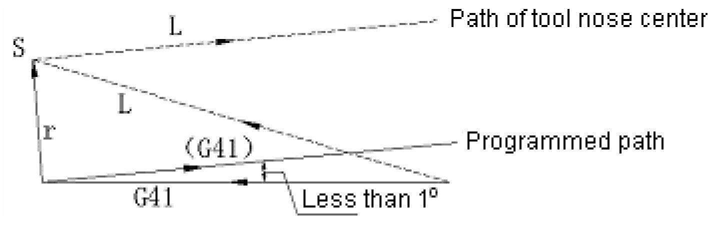
Type B (Straight line→ Straight line)



(Straight line→ Arc)



Note: For Type B, compensation shall be performed as follows when the tool moves on one side of a sharp angle less than 1^0.



(3)　Offset mode

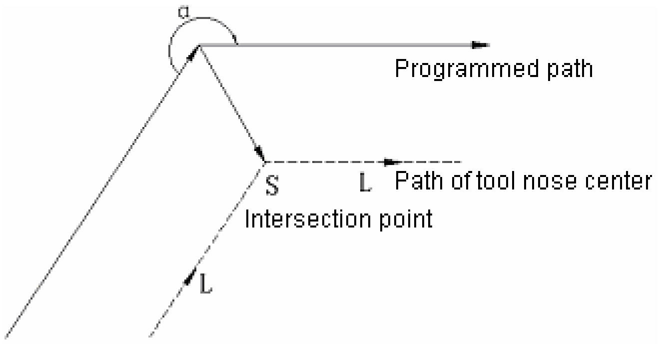
In offset mode, arc interpolation may offset even linear interpolation is instructed.

In offset mode, the blocks without move instruction but miscellaneous function, hold, etc cannot be instructed in two blocks in succession; otherwise it is subject to undercutting or overcutting.

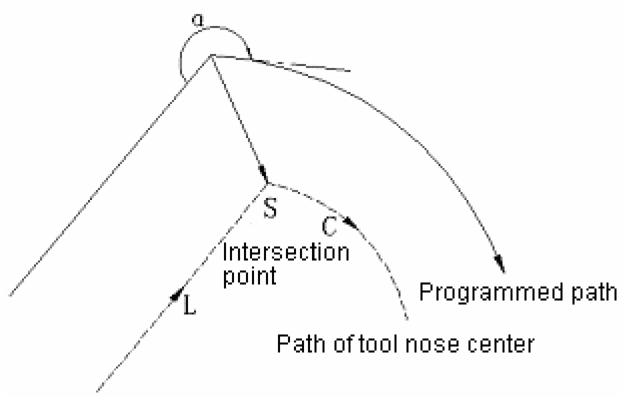
Offset plane cannot be changed in offset mode; otherwise the system will stop after No. 37 alarm.

(i) Situations of interior angle (180°≤α)

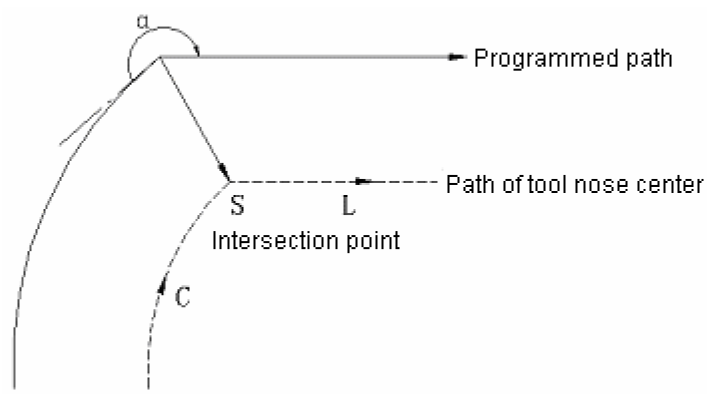
Straight line→ Straight line



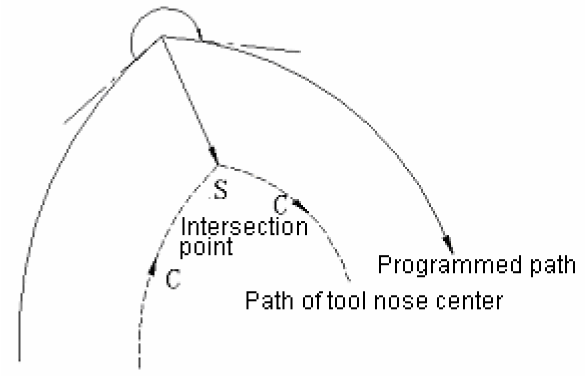
Straight line→ Arc



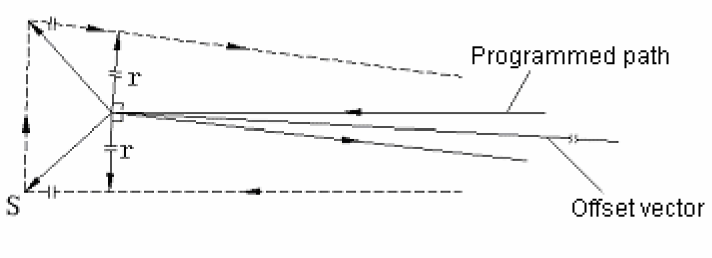
Arc→ Straight line



Arc→ Arc



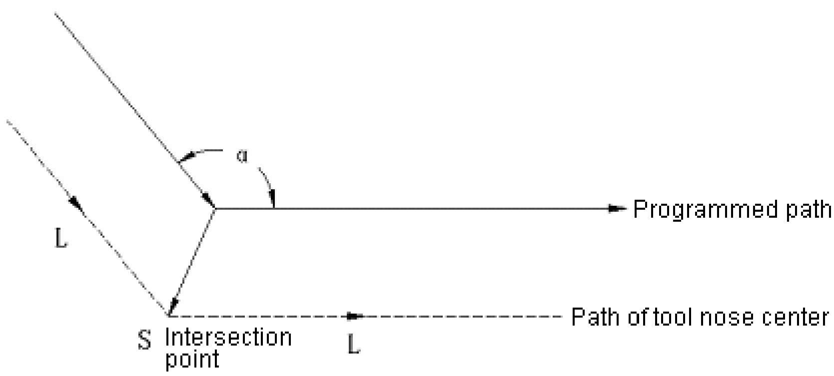
For the condition of straight line to straight line, a narrow and closed angle less than 1° is machined. Now its offset vector becomes extremely big.



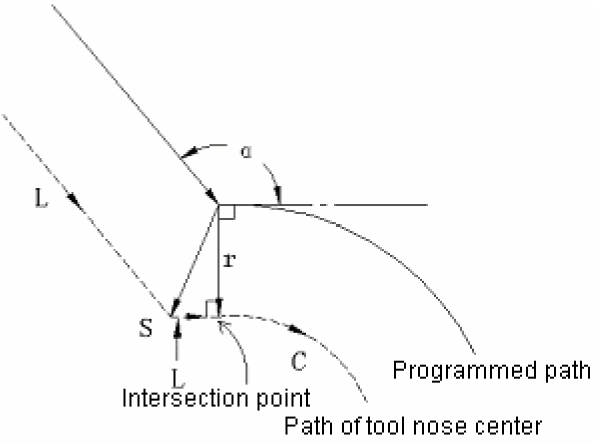
User may follow the above procedures for the conditions of arc to straight line, straight line to arc and arc to arc.

(ii) To machine an obtuse angle along the exterior side (90°≤α＜180°)

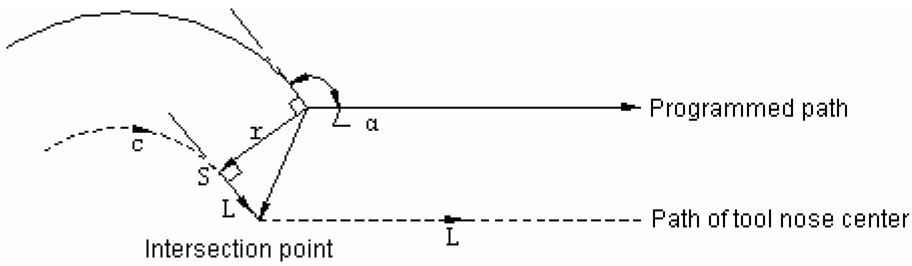
Straight line →Straight line



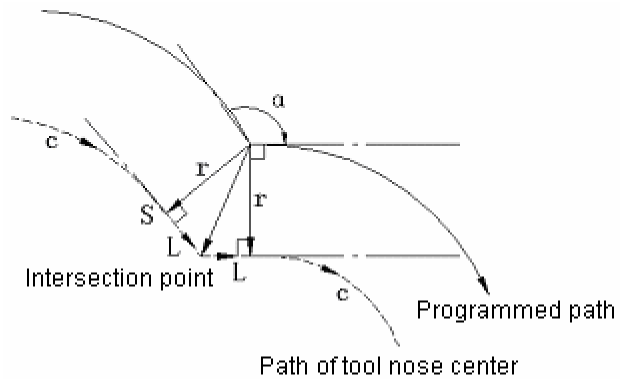
Straight line →Arc



Arc→ Straight line

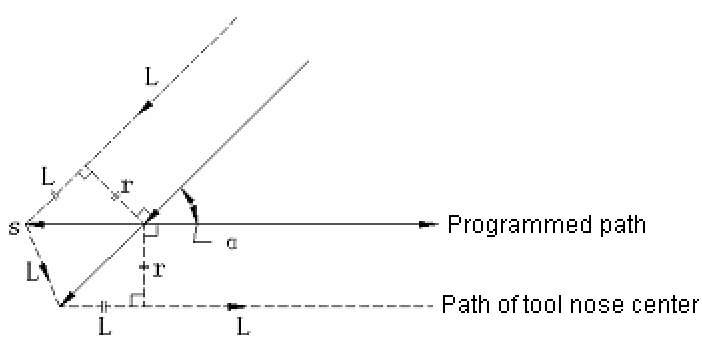


Arc→ Arc

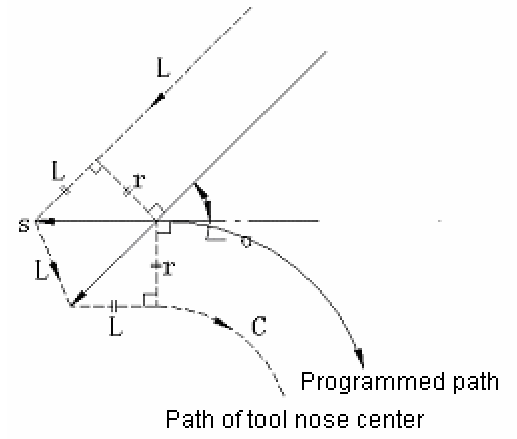


(iii) To machine an acute angle along the exterior side

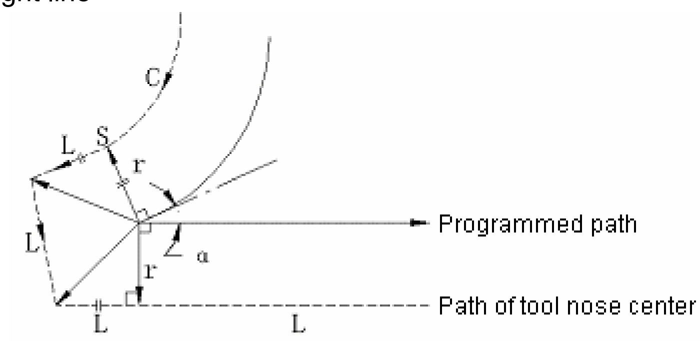
Straight line →Straight line



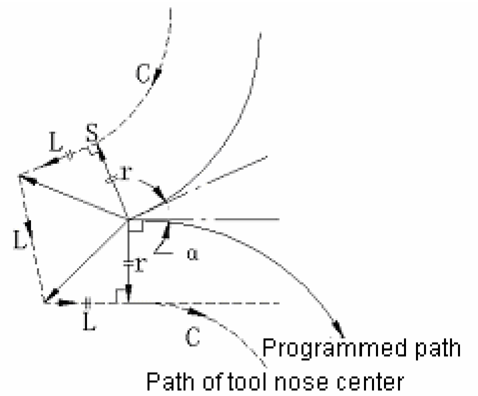
Straight line →Arc



Arc→ Straight line

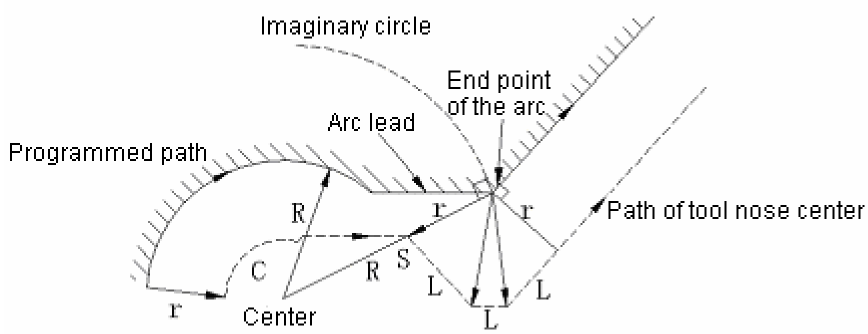


Arc→ Arc



Note 1: Special cases

When the end point of an arc is not on it

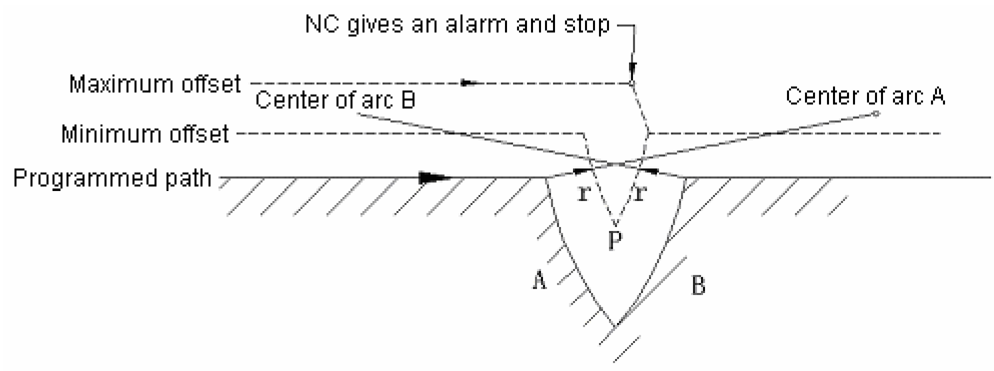


If there is a lead on the arc as shown in the diagram, use the arc center connecting to the end point of the arc as the center of a circle to make a imaginary arc. Make a vector for compensation by using the imaginary arc as the arc of tool compensation. Its result varies from the tool center path that uses the arc lead a lead for tool compensation.

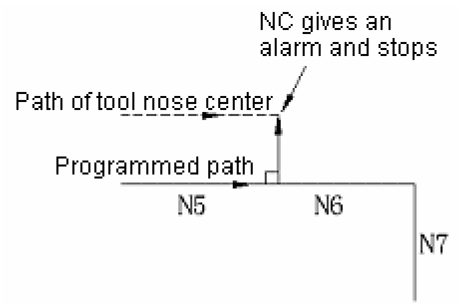
Follow the same procedures for the situation of arc → arc.

<u>The situation without an internal intersection point</u>

As shown in the diagram below, arc intersection point exists on compensation path when the offset is small. Increase in the offset eliminates the intersection point. Now the system gives No. 33 alarm and stops at the end point of the previous block.



As shown in the diagram below, arc intersection point P exists on compensation paths of arc A and arc B when the offset is small. Increase in the offset eliminates the intersection point. For that center coincides with the origin or end point, the system gives No. 38 alarm and stops program execution at the end point of the previous block.



（G41 mode）
N5 G01 X1000;
N6 G02 X1000 I0 J0;
N7 G03 Y-1000 J-1000;

(4)　Offset cancellation

When executing a block satisfying one of the following requirements in offset mode, the system enters into tool cancellation mode. The function of the block is called offset cancellation.

(a) G40 is instructed　　(b) D00 is instructed as a tool compensation number.

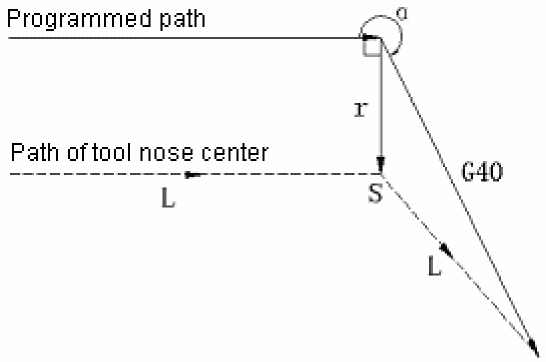
(G02) and (G03) cannot be instructed when offset cancellation is performed. If they are instructed, No. 34 alarm will be sent out and NC will stop.

When a block is read in offset cancellation mode, two blocks including that stored in the buffer (not display) for tool compensation will be executed. In single block mode, a block is read in and executed before stop. The next block is read in and executed by pressing the START button.

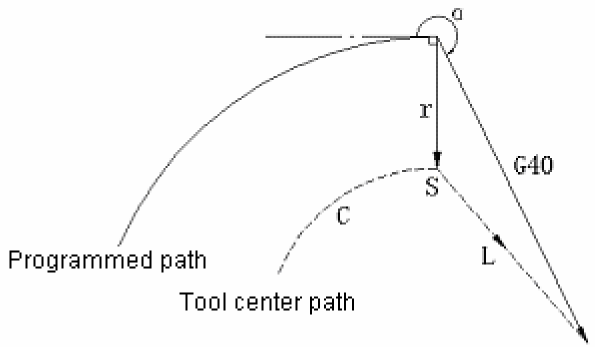
After the control system enters into cancellation mode, the next executed block will be normally saved in the buffer register rather than typed in the tool compensation buffer register.

(a) To machine an interior angle (α≥180°)

Straight line →Straight line



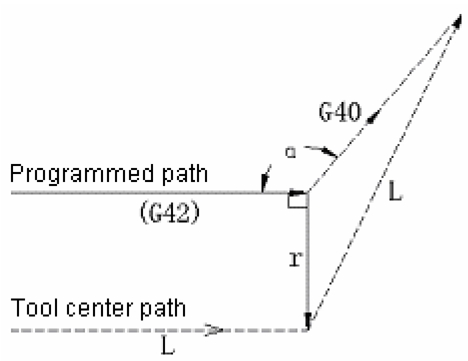
Arc→ Straight line



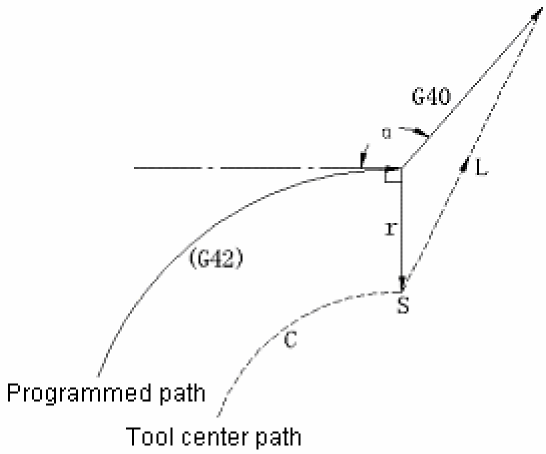
(b) To machine an exterior angle (90°≤α＜180°, obtuse angle)

(i) Type A

Straight line →Straight line

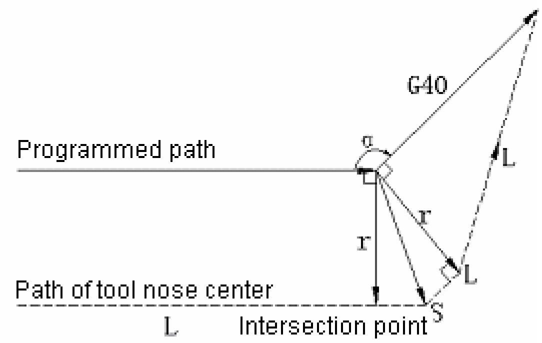


Straight line →Arc

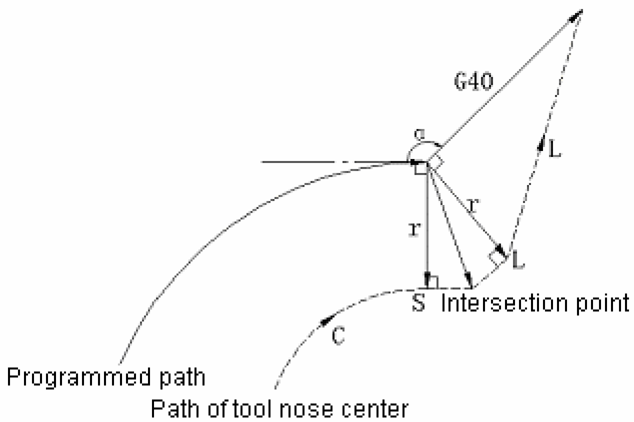


(ii) Type B

Straight line →Straight line



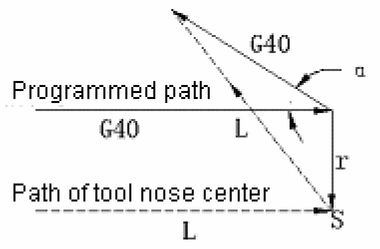
Arc→ Straight line



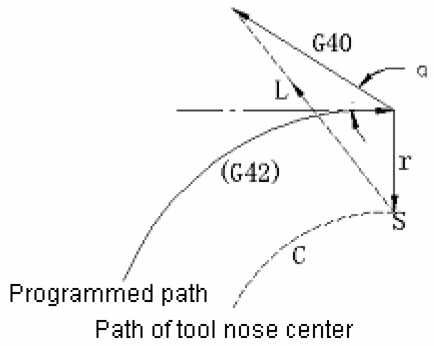
(c) To machine the exterior angle of an acute angle (α＜90°)

(i) Type A

Straight line →Straight line

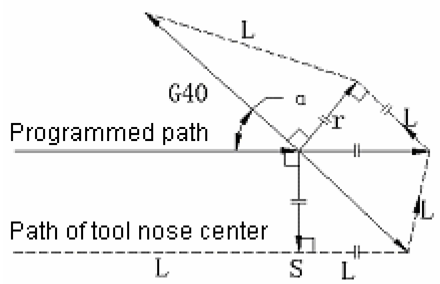


Arc→ Straight line

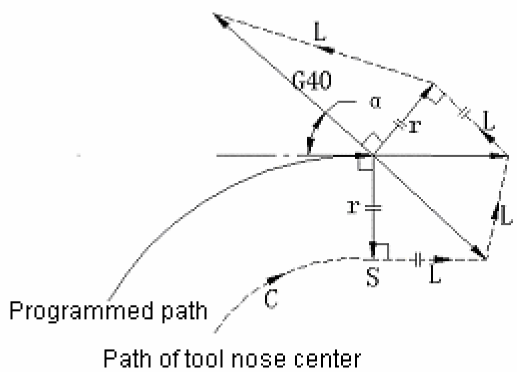


(ii) Type B

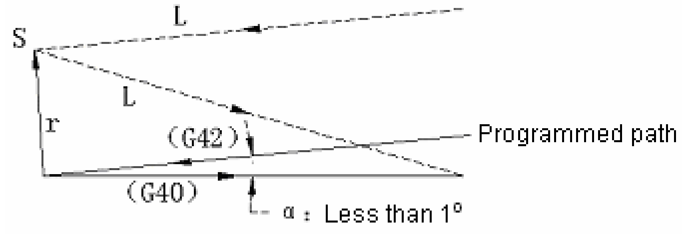
Straight line →Straight line



Arc→ Straight line



Note: For the situation of type B, compensation is performed as follows when the tool machine an acute angle below 1^0 by means of straight line to straight line from the exterior side:



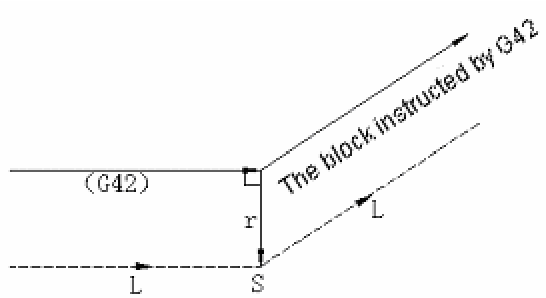
(5) G codes for tool compensation in offset mode

It is possible set an offset vector in relation to the direction of the foregoing block by individually instructing G codes (G41, G42) for tool compensation in offset mode so as to create a correct angle. It is independent of the inner wall or outer wall to be processed.

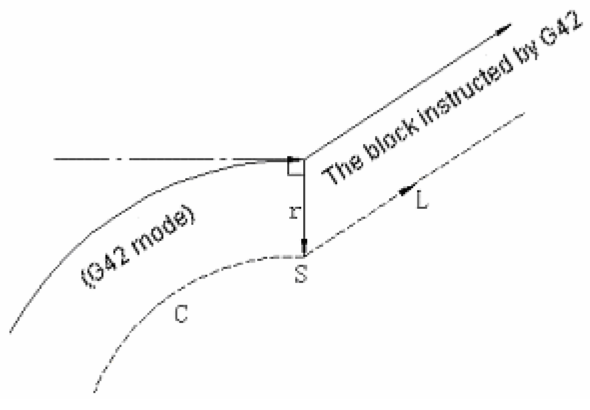
Correct arc motion cannot be performed provided that the codes (G41, G42) are included in an arc instruction.

To reverse the direction of compensation through the G codes (G41, G42) for tool compensation, refer to Note 2: To change offsetting direction in offset mode.

Straight line →Straight line



Arc→ Straight line



Note 2: Change offsetting direction in offset mode.

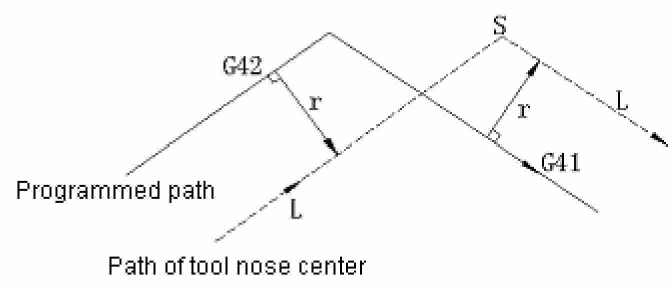
Offsetting direction is identified with the G codes (G41 and G42) for tool compensation and offset signs as follows:

Offset signs / G codes	+	–

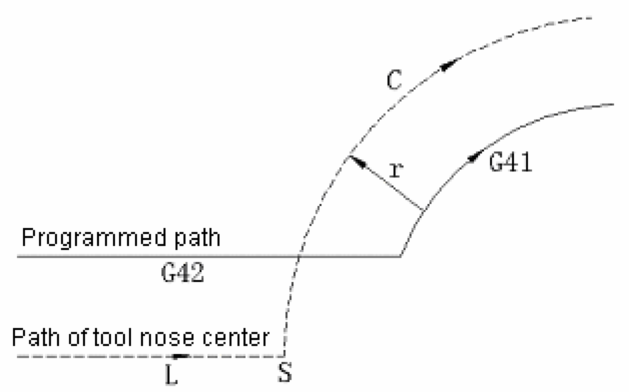
G41	Offsetting to the left	Offsetting to the right
G42	Offsetting to the right	Offsetting to the left

In special cases, offsetting direction can be changed by switching between G41 and G42 in offset mode. However, it cannot be changed for the start-up block and the block that follows. In the situation of changing offsetting direction, the concepts of interior side and exterior side are canceled to adapt to all conditions. The amounts of offset are assumed to be positive in all the following examples.

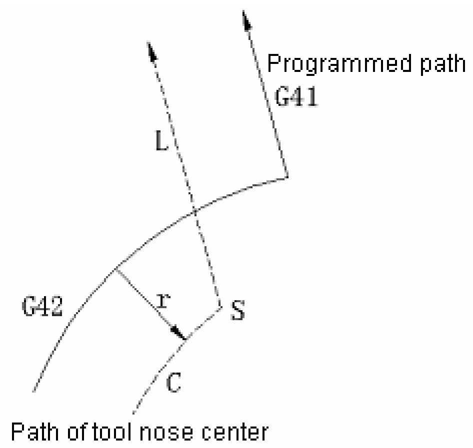
(Straight line →Straight line)



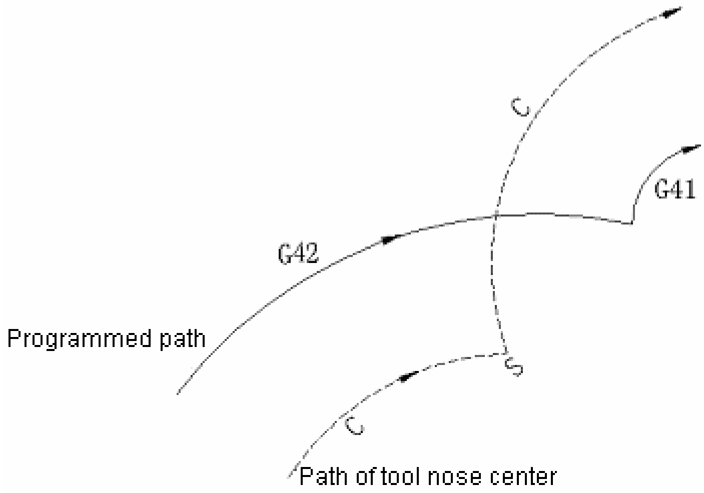
(Straight line →Arc)



(Arc→ Straight line)



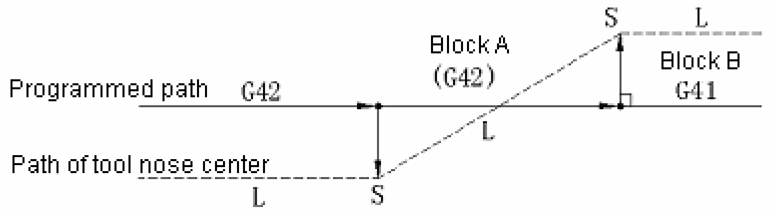
(Arc→ Arc)

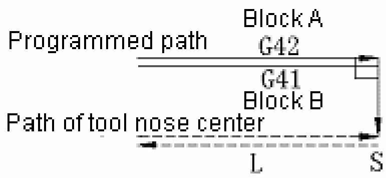


When a path with offset has not intersection point:

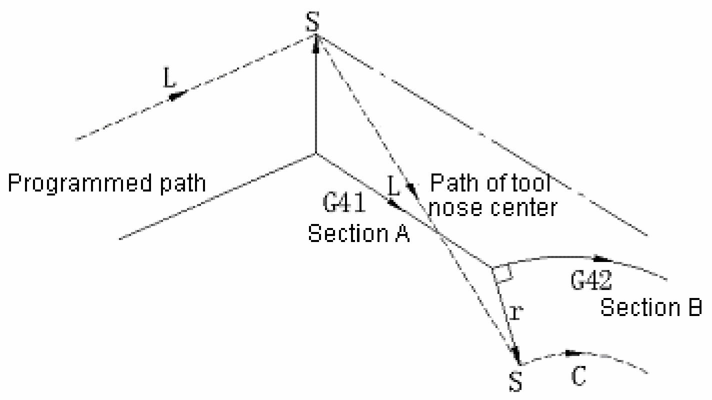
If the switch between G41 and G42 is performed and there is no intersection point of offset path from block A to block B, a vector perpendicular to the programmed direction will be established at the point of origin of block B.

a) Straight line →Straight line

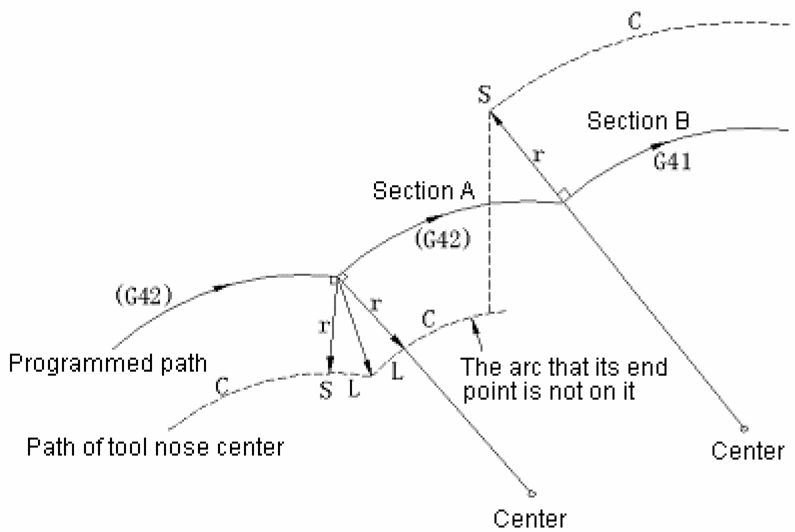




b) Straight line →Arc

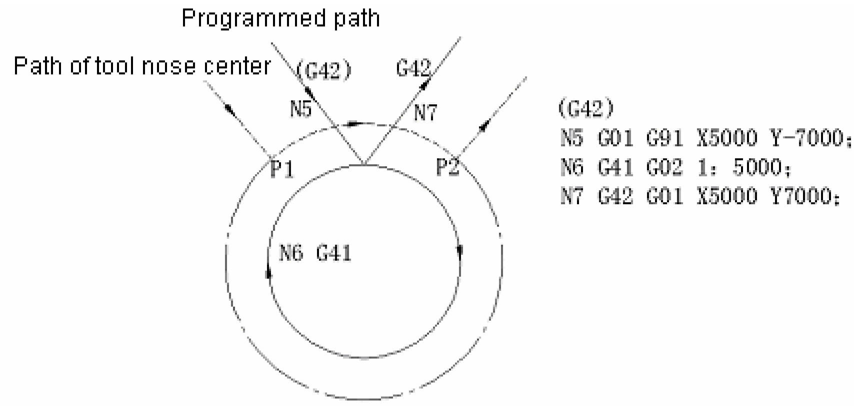


c) Arc→ Arc



The situation in which the length of the tool center path caused by tool compensation:

Generally the above condition will not arise. It is only possible when G41and G42 is switched or G 40 is instructed with address I, J and K.



In the above situation, the tool center path does not go around the circle for one turn but only makes arc motion along P1～P2. Its cause will be described in the alarm caused by interference verification. If you want the tool to make movement along the full circumference, make sure to instruct the circumference by sections.

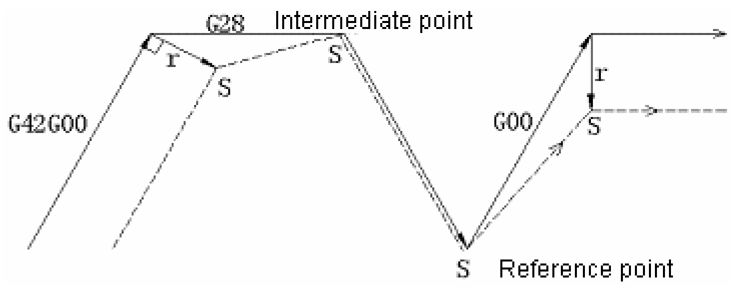
(6) The cancellation of temporary offset and execution of the following instructions in offset mode, "cancellation of temporary offset" will be triggered. Then the system will be automatically restored to offset mode.

Refer to Section 6.3.6 (4) "Offset cancellation" and Section 6.3.6 (2) "Start-up" for the details of the above operations.

(a) G28 Automatic return to reference point

If G28 is instructed in offset mode, offset will be cancelled at the intermediate point. When the reference point is reached, the offset mode will be automatically restored.

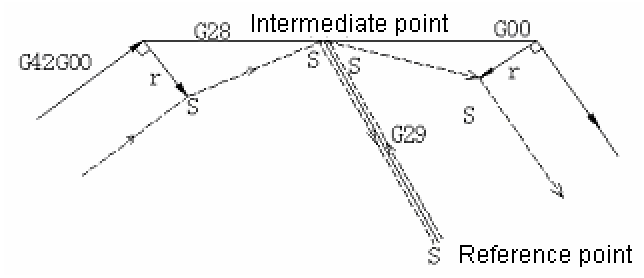
If offset vector is maintained at the intermediate point, the NC will zero the vector of all axes returned to reference point.



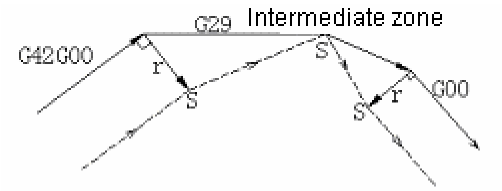
(b) G29 Automatic return from reference point

If G29 is instructed in offset mode, the offset will be cancelled at the intermediate point and then automatically restored in the next block.

To directly instruct G29 after G28:



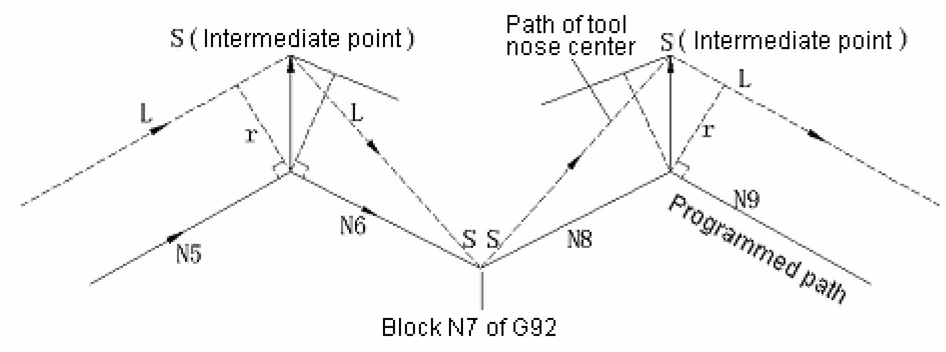
The situations rather than that G29 is directly instructed after G28:



(7) The instruction for temporary cancellation of an offset vector

In the offset mode, offset vector shall be cancelled and then it will be automatically restored if G92 (absolute zero programming).

In this case, offset cancellation will not be performed. The tool directly moves to the point that is specified to cancel its offset vector from the intersection point, and directly moves to the intersection point when the offset mode is restored.



(G41 mode)

N5 G01 X3000 Y7000:

N6 X－3000 Y6000:

N7 G92 X1000 Y2000:

N8 G01 X4000 Y8000:

Note: SS indicates the points at which tool stops twice in single block mode.

(8) The blocks without tool movement

The following blocks are free from tool movement. The tool does not move even tool radius compensation is valid in these blocks.

M05; ……………………………M code input

S21;　…………………………S code input

G04　X1000;　………………Dwell time

G22　X100000; ………………Machining area setting

G10　P01　X100; ……………Offset setting

(G17) Z2000; …………………The movement beyond offset plane

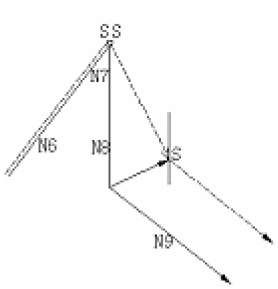
G90; ……………………………Only G codes

G91　X0; ………………………Offset is 0

Not move

a) The situations in which a block is instructed during start-up

If a block without tool movement is instructed at the origin of the program, it does not generate offset vector.



G04　G91 --------------

⋮

N6　X1000.0　Y1000.0;

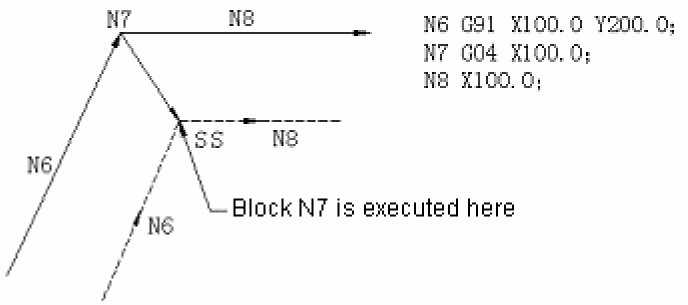
N7　G41　X0;

N8　Y－1000.0;

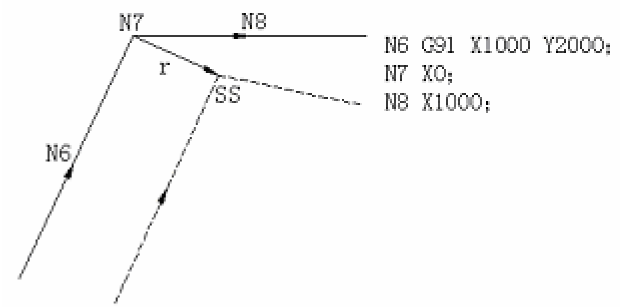
N9　X1000.0　Y—1000.0;

b) The situations in which a block is instructed in offset mode

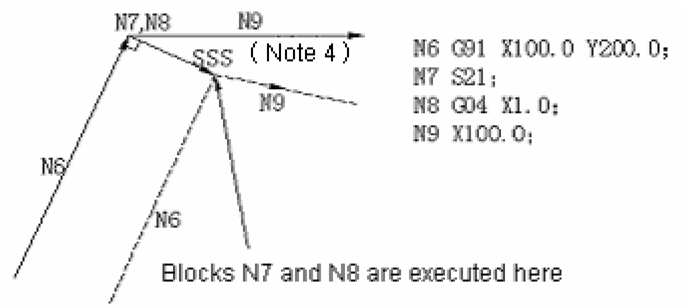
When a single block without tool movement is instructed in offset mode, its vector and tool center path are the same as when the block is not instructed (refer to Section 6.3.6 (a): Offset mode). The block is executed at the stop point of the single block.



When the travel is 0, however, the tool movement of the block is identical with when more than a block without tool movement is instructed. It is specifically described as follows:



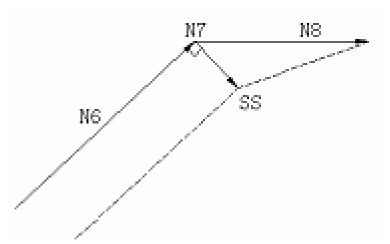
More than 2 blocks without tool movement cannot be specified in succession; otherwise a vector which has a length equal to the offset and direction perpendicular to the moving direction of the tool in the previous block will be generated, thereby causing overcutting.



Note 4: SSS means that the tool operated by a single block stops here for three times.

c) When the block is instructed along with offset cancellation

When a block without tool movement and offset cancellation are specified at one time, its vector whose length is equal to the offset is generated in the direction normal to the tool movement of the previous block. The vector will be cancelled in the next move instruction.

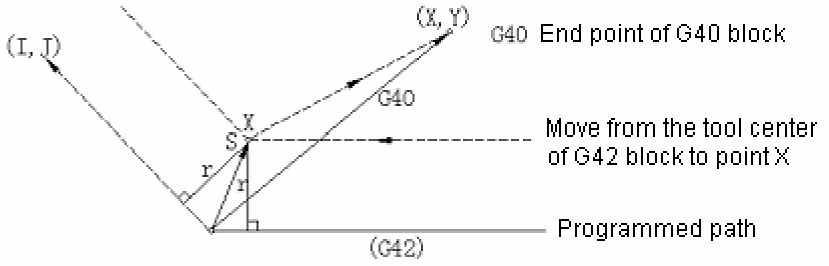


N6 G91 X100.0 Y100.0:
N7 G40:
N8 X100.0 Y0:

(9) When G40 and the one among I, J and K that is in the offset plane are instructed and the previous block mode is G41 or G2.

When the above instruction is performed in the offset mode, it will become the following condition with G17 plane as an example. Analogies may be drawn for the situations of other planes.

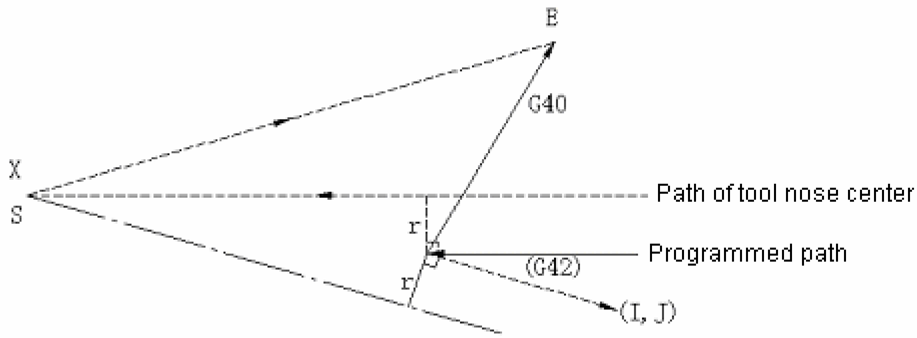
Now the direction of the vectors (I, J) that start from the end point of the previous block are determined by the above instructions. Its offsetting direction is the same as the previous block.



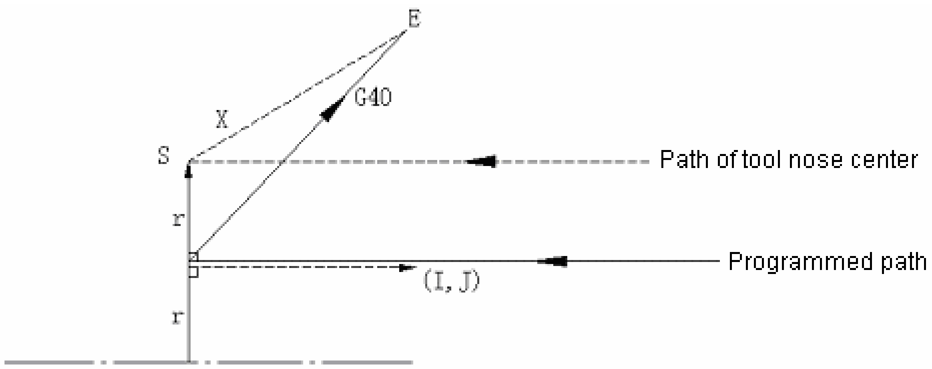
(G42 mode)

G40 X X Y Y I — J --:

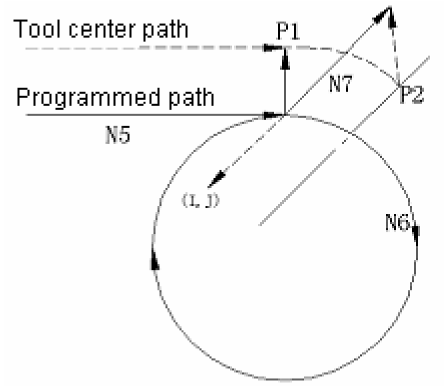
Note 5: Take note that in this situation the NC gets a intersection point of tool path independent of the specified machined inner wall or outer wall.



Note 6: When it is impossible to get an intersection point, the tool reaches the position normal to the previous block path at the end of the previous block.



Note 7: When the length of the tool center path is bigger than the circumference:



(G41)
N5　G01　G91　X10000:
N6　G02　J－6000:
N7　G40　G01　X5000:
　　Y5000　I－100　J－100:

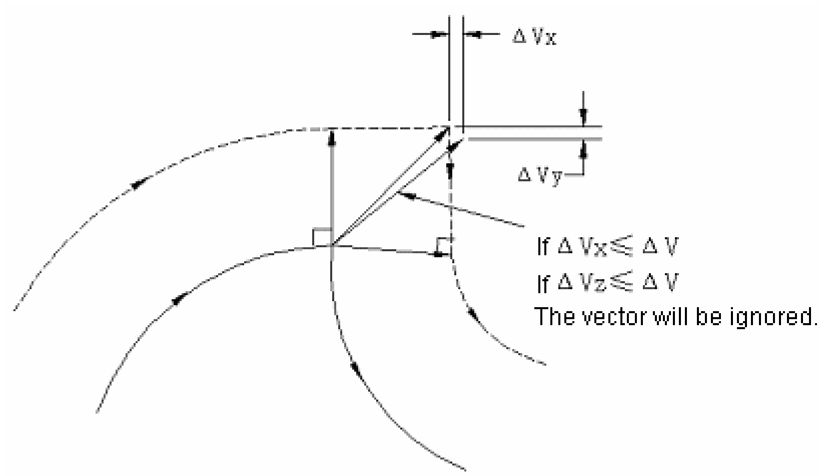
In the above situation, the tool center does not go around the circumference but the section of arc between P_1 and P_2.

The alarm caused by interference verification is related to the following conditions. (If you want the tool to go around the circumference, you need to instruct a circumference by sections.)

(10) Corner movement

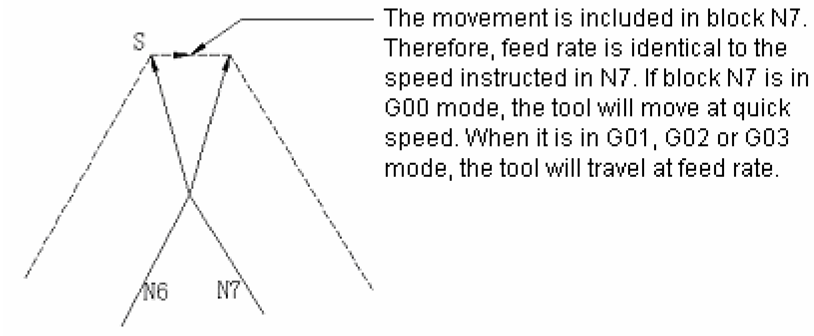
When 2 or more vectors are generated at the end point of a block, the tool makes line motion from one vector to the other.

If these vectors almost coincide, corner movement will not be performed and the following vectors will be ignored.



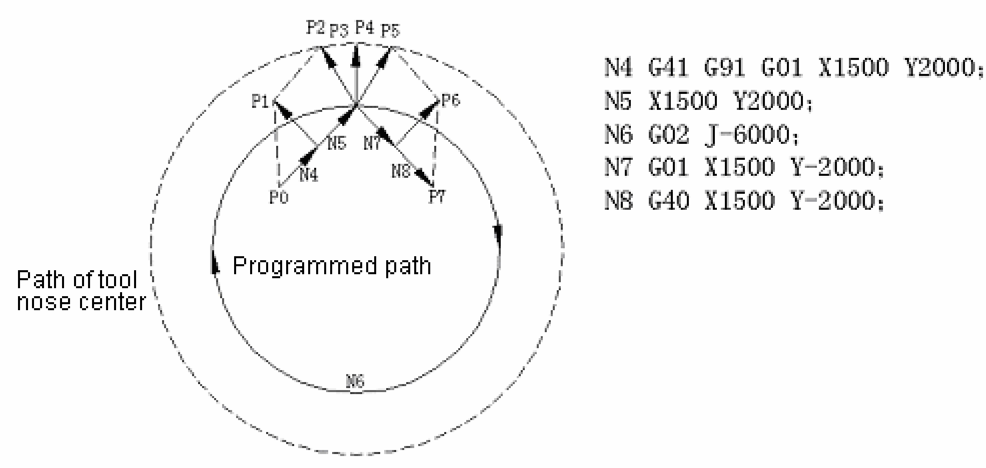
If $\triangle V_x < \triangle V$ limits or $\triangle V_y < \triangle V$ limits, the vectors that follow will be ignored. $\triangle V$ limit shall be preset by parameter No.069(CRCDL).

If these vectors do not coincide, movement around the angle will be caused. The movement is included in the blocks that follow.



Note 8: If the path of the next block is an arc over a semicircle, however, the above-mentioned function cannot be achieved.

The cause is as follows:



If the vector is not ignored, the tool path is as follows:

P_0→P_1→P_2→P_3 (Circumference)→P_4→P_5→P_6→P_7

If the distance between P_2 and P_4 is small, P_3 will be ignored. Now the tool path is as follows:

P0→P1→P2→P3→P4→P5→P6→P7

The arc cutting instructed by block N6 will be ignored.

(11) General cautions about compensation

a) To specify offset

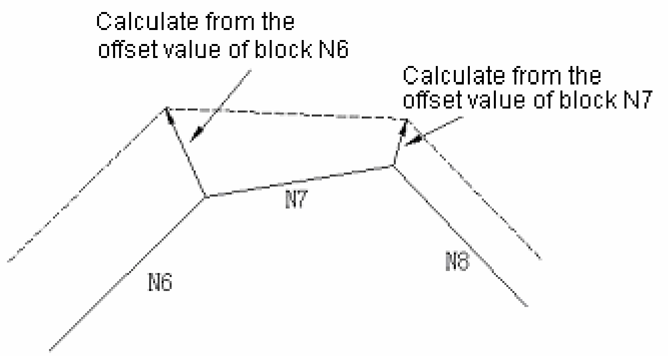
The D codes that are specified with an offset number instructs offset.

Once they are specified, the D codes will remain valid until another D code is specified or they are cleared.

Besides that D codes are used for the offset of tool radius compensation, they are also intended for specifying the offset of tool position offset. If both tool compensation (G41/G42) and tool offset (G45～G48) are included in a block, No. 36 alarm will be given.

b) To change offset

In general, if offset is changed during tool change in offset cancellation mode, the vector of the end point of the block is fit for the new offset.

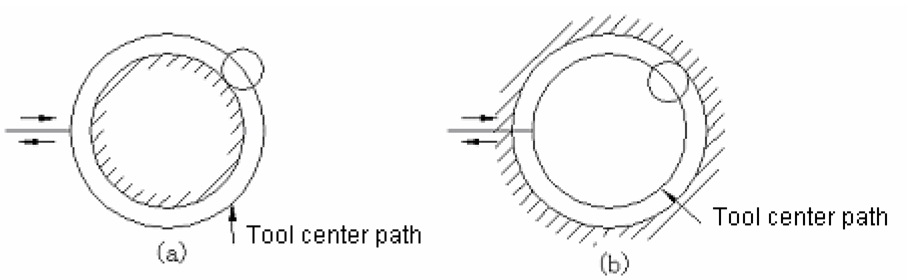


c) Positive and negative offsets and tool center path

A negative (-) offset is equivalent to the exchange of G41 and G42 in the program. If originally the tool center moves along the exterior side of a workpiece, now it will move along its interior side, vice versa.

The figure below is an example. In general, offset is programmed as a positive value.

When the offset is negative when the tool path is programmed as shown in the Figure (a), the tool center will move as indicated in (b), vice versa. Therefore machining a female mould and a male mould may use the same program. The clearance between them may be adjusted by selecting an offset (it is also applicable if the start-up and cancellation is of type A.)

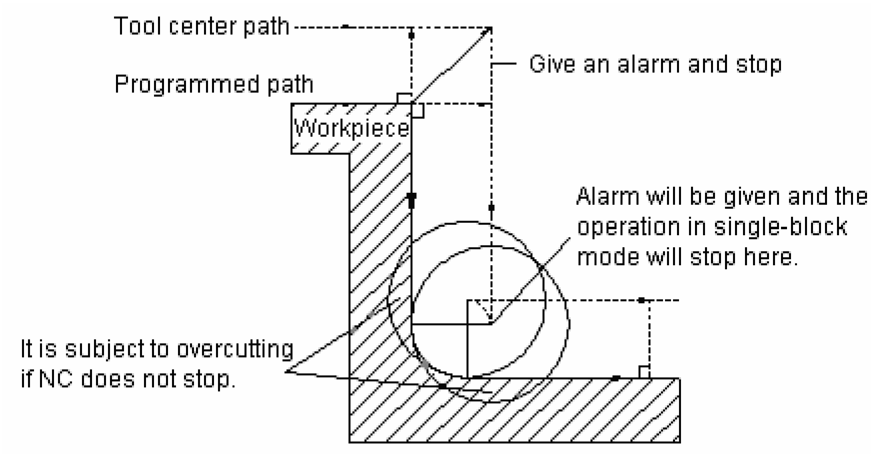


d) Overcutting resulting from tool compensation

(i) To machine the inner side of an arc whose radius is less than that of a tool

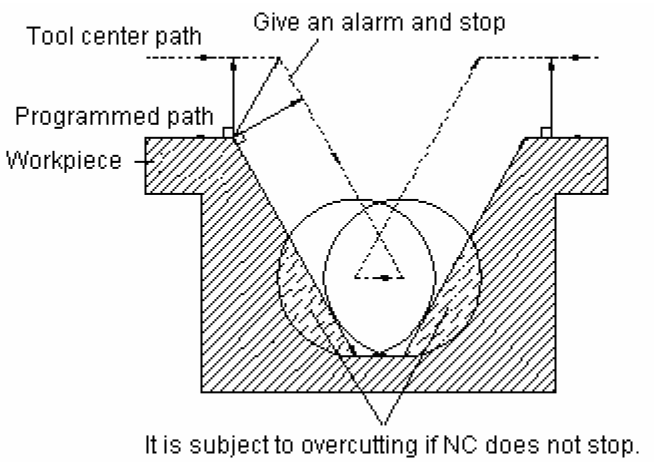
When the instructed arc radius is less than that of the tool, the system will give No.41 alarm and stop since the offset of the interior side of the tool may lead to overcutting. During single block operation, however, it is likely to cause overcutting because the tool only stops when the program ends up. Now the tool movement is similar to the No. 41

alarm below.



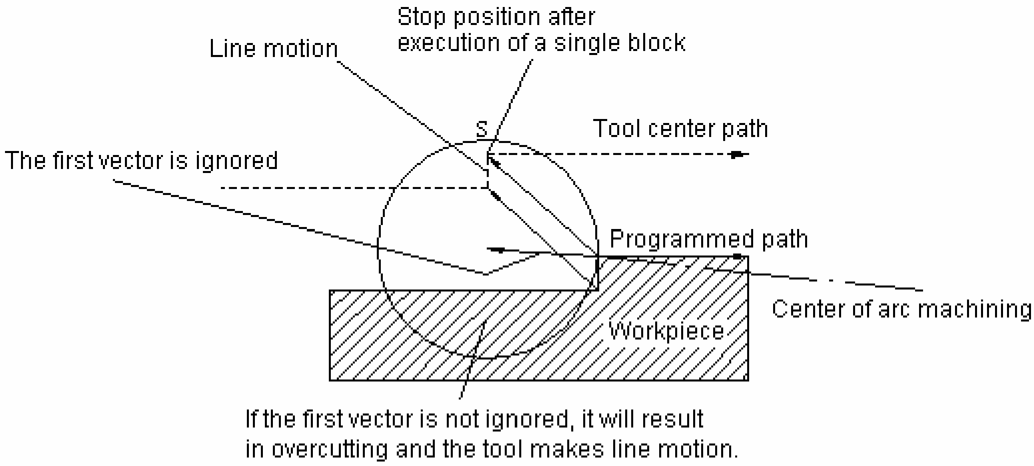
(ii) To machine a groove smaller than tool diameter

Since that tool compensation forces the tool center path to move in the direction contrary to the programmed direction may cause overcutting, No. 41 alarm will be given and NC will stop at the origin of the block.



(iii) To machine a step less than the tool radius

When the program includes a step smaller than the tool radius and it is machined under circular cutting instruction, the tool center path using normal offset 6.3.6 (3) will become the reverse programmed direction. Now the first vector will be ignored and the tool moves the position of the second vector. The tool stops at this point. The program continues to execute if it is not machining in single block mode.



(iv) The startup associated with tool compensation C and the movement in the direction of axis Z

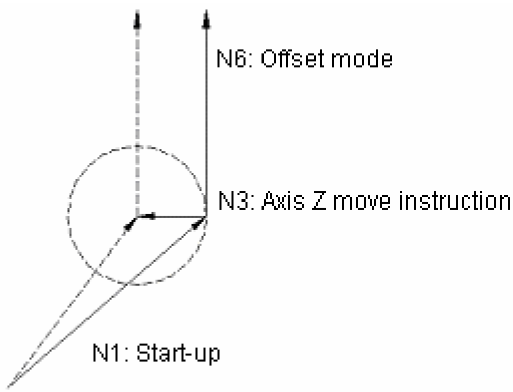
At the beginning of cutting, tool compensation (generally XY plane) shall be preset at a distance from the workpiece and then the tool feed by means of movement along axis Z. Now take note of the following issues if the rapid traverse (positioning) of axis Z shall not be independent of the cutting feed.

Refer to the following program

N1 G91 G00 G41 X50000 Y50000 D1;

N3 G01 Z-30000 F1;

N6 Y100000 F2;



In the above example, N3 and N6 will also be read in buffer register during execution of block N1 and the relationship between them will be properly compensated as shown in the right figure.

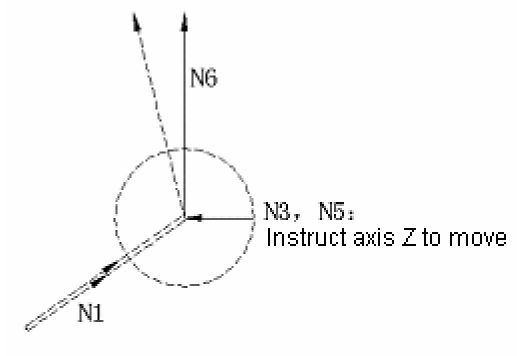
Secondly, if N3 (axis Z move instruction) is separated,

N1 G91 G00 G41 X50000 Y50000 D1;

N3 Z-25000;

N5 G01 Z-5000 F1;

N6 Y100000 F2;



Since both the move blocks N3 and N5 do not include the XY plane, block N6 cannot enter into the buffer register when N1 starts execution. As a result, the tool center path is calculated by the information N1 in the right figure. In this situation, tool vector will not be generated during start-up, thereby resulting in the overcutting as shown in the right figure.

In this situation, overcutting may be prevented by specifying an instruction of the same moving direction in the blocks before and after axis Z feed instruction using the above rules.

N1 G91 G00 G41 X50000 Y40000 D1;

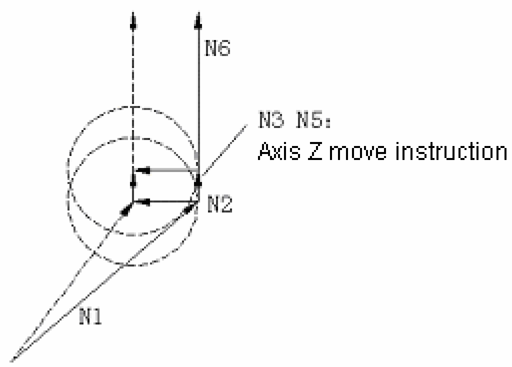
N2 Y10000;

N3 Z-25000;

N5 G01 Z-5000 F1;

N6 Y100000 F2;

(The moving directions of N2 and N6 are identical)



Blocks N2 and N3 are read in the buffer register and correctly compensated according to the relationship between N1 and N2 during the execution of block N1.

Note 9: Interference verification

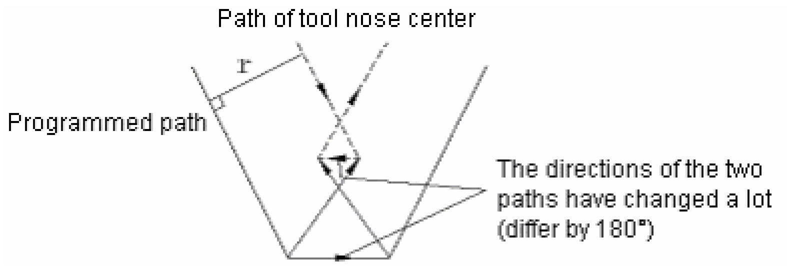
Tool overcutting is called “interference”. Interference verification is a function designed to precheck tool overcutting. However, not all overcuts can be checked by the function. There are also some cases that overcutting is not checked.

1）The benchmark for overcutting verification

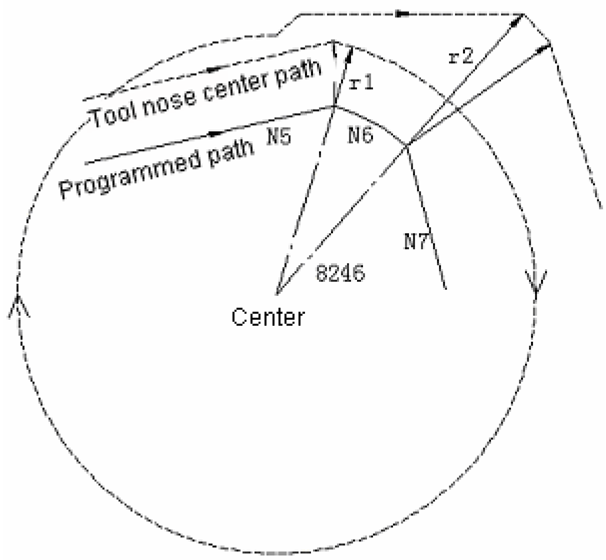
a) In tool compensation, the traveling direction of the tool center path differs from that of sine path (differ by 90°～270°) .

b) During arc machining, the angle difference between the starting point and end point of the tool center path extremely varies from that between those of programmed path except the above state a (more than 180°).

Example of state a:



Example of state b:



(G41)

N5 G01 G91 X8000 Y2000 D01:

N6 G02 X3200 Y-1600 I-2000 J-8000 D02:

N7 G01 X2000 Y-5000:

(Corresponding offset of D01: r_1 =2000)

(Corresponding offset of D02: r_2 =6000)

In the above example, the arc in block N6 is within a quadrant. However, the arc extends to the four quadrants after tool compensation.

2）Precorrection of interference

(a) Vector travel associated with interference

When tool compensation is performed for blocks A, B and C, the vectors V_1, V_2, V_3 and V_4 of blocks A and B as well as the V_5, V_6, V_7 and V_8 will be created. The vectors close each other will be first verified. They will be neglected in case of interference. If the vector to be neglected due to interference is a vector at the end of the angle, it will not be neglected.

The interference verification before vectors N4 and N5 →interference→V_4 and V_5 are neglected.

Verify V_2 and V_6 → interference → neglect

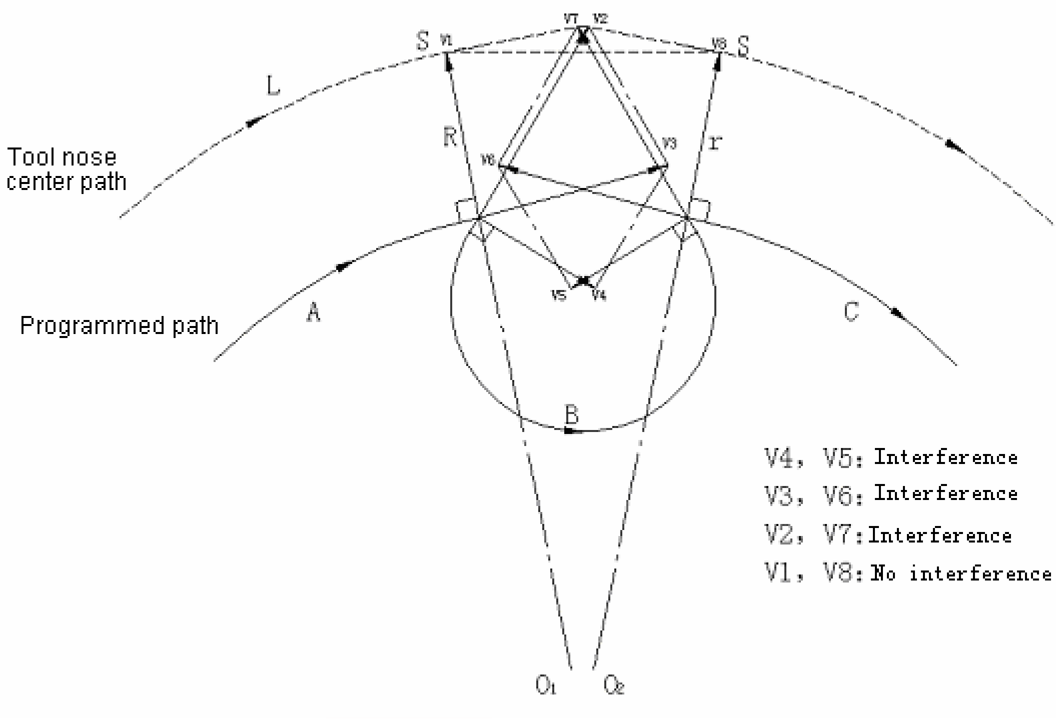
Verify V_2 and V_7 → interference → neglect

Verify V_1 and V_8 → interference → not neglect

If no vector interference is found during verification, the verification will stop.

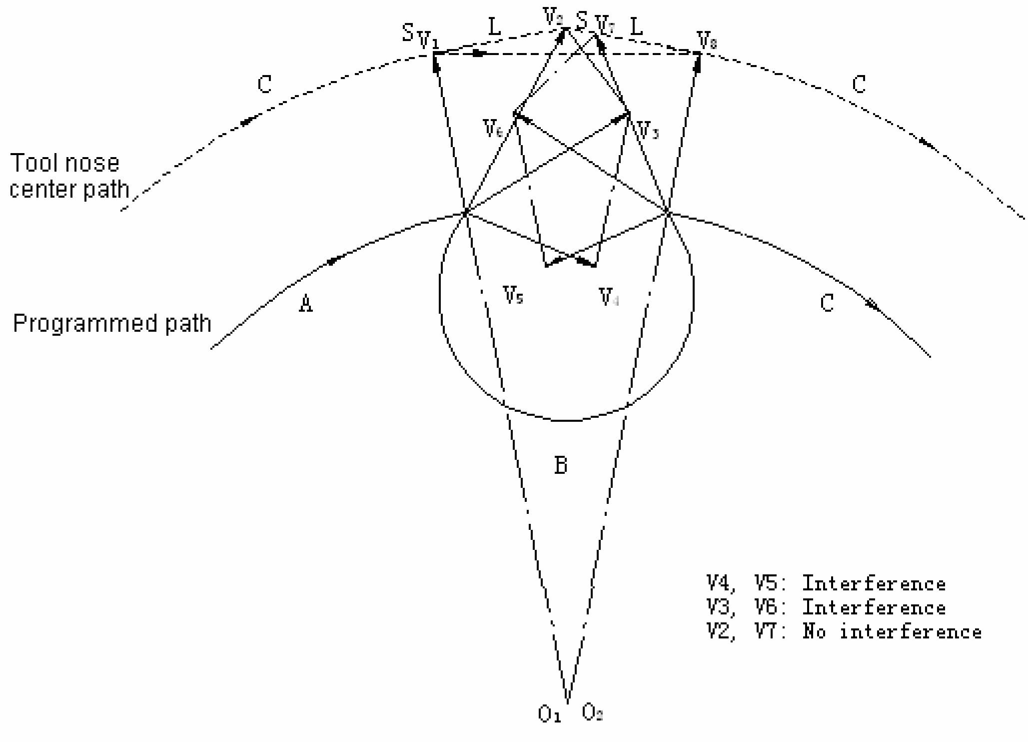
In case block B is an arc, the arc will make line motion if interference arises.

Example 1: The tool makes line motion from V_1 to V_8.



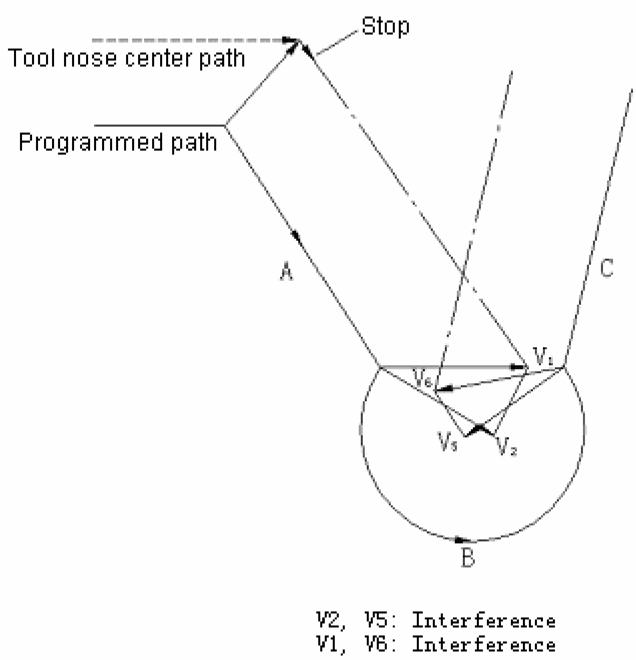
Example 2: The line motion of the tool is as follows:

Tool path: V_1→V_2→V_7→V_8



(b) The tool will stop due to alarm if interference arises after correction (a).

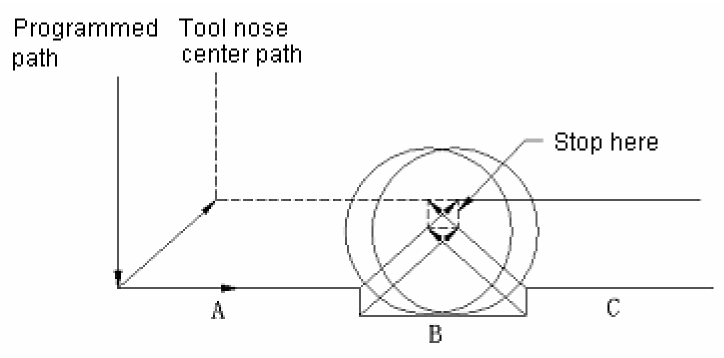
If interference arises at the last vector during correction (a) or there is a pair of vectors at the beginning of verification and interference arises, the system will give No. 41 alarm and stop after the completion of the previous block.



Though vectors V_2 and V_5 are neglected due to interference, interference will arise between V_1 and V_6. Now the system gives an alarm and stop.

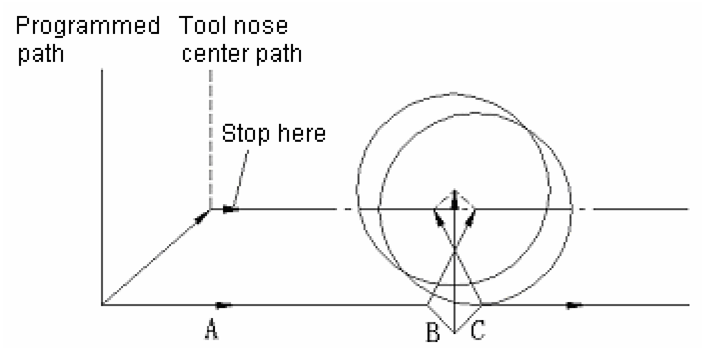
3) Verification is performed even actually no interference arises. See the following examples:

(a) Concave depth less than tool compensation



Though actually no interference occurs, the tool stops due to No. 41 alarm because the direction of tool path after tool compensation differs from that of programmed path.

(b) Groove depth less than tool compensation



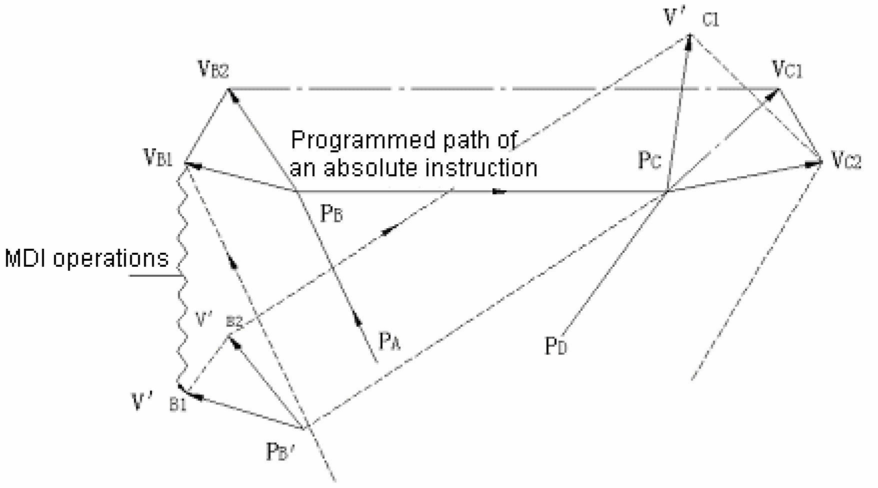
It is similar to (a) and the direction of tool path differs from that of programmed path.

(12) Instructions input through MDI

Compensation is not applicable for the instructions input through MDI. However, the tool path is as follows after the use of the programmed operations constituted of absolute instructions, dwell of execution of single block, performing of MDI operations and the restart of automatic operations.

Under these conditions, the vector at the starting point of the block that follows is translated while other vectors are created by the next two blocks.

Therefore, compensation is automatically performed from point Pc.



When points P_A, P_B and P_C are specified in an absolute instruction, the tool is stopped at the end point of the blocks P_A to P_B by the single block function. Now the means for translation are operated by MDI. Vectors V_{B1} and V_{B2} are translated to V'_{B1} and V'_{B2}. The V_{C1} and V_{C2} between P_B —— P_C and P_C —— P_D need to be recalculated.

Since the vector V'_{B2} is not to be calculated, compensation will be accurately performed from point P_C.

(13) Manual inputting interference

Refer to Volume II, Section 4.3.4.3, e), Note 1 for the manual inputting interference during tool compensation.

(14) The tool compensation involving with the 4th axis

Tool compensation cannot be performed for the 4th axis because the offset plane involving with the 4th axis is not provided.

3.6.4 D and H functions

Address D and H are used to specify tool offset and tool compensation. They use the same number and the same compensation value.

Address D differs from H as follows:

D———For tool diameter compensation (tool diameter compensation and tool position offset)

H———For tool length compensation (tool diameter compensation and tool position offset)

The codes and offsets are input in storage through MDI/LCD panel correspondingly so as to specify one two-digit code to perform the relevant compensation. The specified usable value is selected in the following range. No. 30 alarm will be given if a value beyond the range is specified.

The compensation specified by H00 and D00 are always 0. H00 and D00 are made certain after power on.

The standard number of tool compensation is 32 (01～32). When options A, B and C are selected for the number of tool compensation, they correspond to 64 (01～64), 99(01～99) and 200 (01～200) respectively.

Note 1: Tool compensation in G40, G41 and G42 modes always use D codes. Tool length compensation (G43, G44, G49) always use H codes. Whether tool offset (G45, G46, G47, and G48) uses D codes or H codes are specified by parameter No.010 BIT3 (OFSD).

Note 2: When an additional tool offset number B is used, make sure to select a part program to save/edit B (40m)～F (1280m).

Note 3: When an additional tool offset number C is used, make sure to select a part program to save/edit C (40m)～F (1280m).

3.6.5 External tool offset

The function is designed for correction of an offset from the outside. For example, using the function on the machine side may input a tool offset and add it to the offset corresponding to the offset number specified by program. In addition, the input value itself may be specified as an offset.

When the machine is provided with an automatic measurement function for tool and workpiece, a difference value in relation to the accurate value may be input in the NC as the amount of correction with the function.

Make sure to operate by following the instruction manual supplied by manufacturer because of machine manufacturers' difference in programming, operating functions and restrictions.

3.6.6 To input offset through program (G10)

The offsets for tool position offset, tool length compensation and tool radius compensation may be specified by G10 instruction in programming. The instruction has the following format:

G10　P p R r

p: Offset number

r: Offset

Whether the offset is absolute or incremental depends on G90 or G91 mode.

3.6.7 Zooming function (G50, G51)

The zooming of the switch specified in machine program may be performed through instruction. First the zooming scale shall be enabled in No. 64 parameter.

G51 I　J　K　P　:

I, J, K: X, Y and Z coordinates in zooming

P: Zooming ratio (minimum input incremental: 0.001)

With the instruction, the move instruction that follows is converted by the zooming ratio specified by P and centered on the point specified by I, J and K.

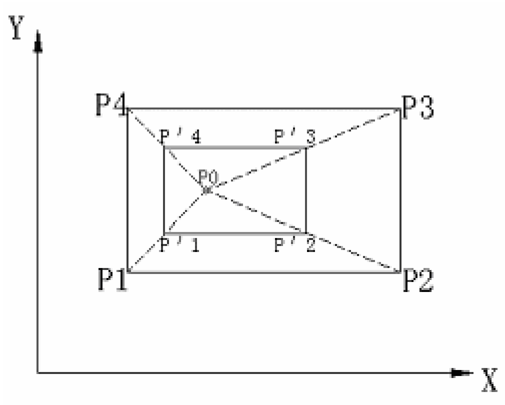
The conversion mode is cancelled by G50.

G50: Zooming mode cancel instruction

G51: Zooming mode

The zooming range that may be instructed is as follows:

0.001x～99.999x (P1～P99999)



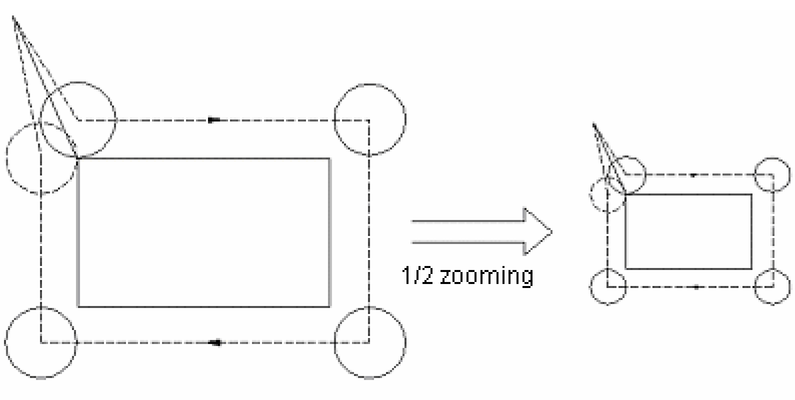
P1～P4: The outline of machine program

P1～ P' 4: The outline after zooming

P0: Zooming center

If P is not specified, the zooming ratio may also be set by MDI/LCD. For the situations that I, J and K are omitted, the point instructed by G51 serves as the zooming center.

The zooming can not be used for offset, e.g. tool radius compensation, tool length compensation and tool position offset.



Note 1: In a single block, G51 shall be specified in G40 mode. G50 may also be specified in offset mode. G51 must always be cancelled by G50.

Note 2: A position is indicated at coordinates after zooming.

Note 3: If a setting is used as a zooming ratio that does not specify P, the setting will be the zooming ratio specified by G51 and it is invalid for any other instruction to specify the setting.

Note 4: Whether the zooming function of all axes is valid may be set by parameter. For G51 mode, the function is always valid for the arc radius instructed by R and is dependent of parameter setting.

Zooming function is always invalid for additional axes.

Note 5: Zooming function is invalid for manual operations, but valid for DNC, automatic operations and MDI operations.

Note 6: Zooming is not applicable for the following movement in the event that axis Z moves in a fixed cycle.

* The cutting depth Q and backing of peck drilling cycle (G83, G73).

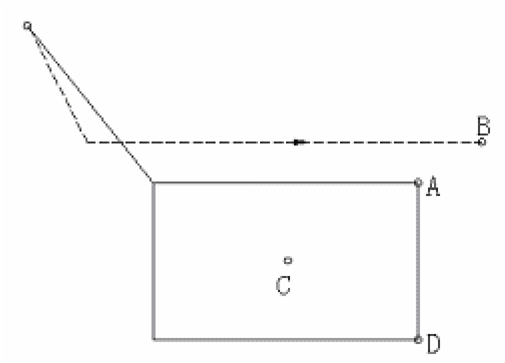
* The travel of X and Y in finish boring (G76) and reverse boring (G87) cycles

Note 7: G27, G28, G29, G30 and G92 are specified in G50 mode.

Note 8: Rounding of the zooming result may zero the travel. Now the block is considered as an unmovable block. Therefore, it may affect the tool movement caused by tool compensation (see Section 6.3.6 (8)).

Note 9: Reset

(a) Reset in G51 mode changes the original programming coordinate to the current coordinates or zoomed coordinate. Hence the travel after reset differs depending on whether the instruction is incremental or absolute.



If it is reset at Point B, Point A in the program shall be regarded as the current B. While executing the move instruction of Point D, the generation of the following movements depends on whether they are incremental or absolute.

* Incremental

If the travel from Point A to Point D is incremental, D' will become the point of destination on the programmed path and Point D' is converted to Point E. The tool moves to Point E_Y because it is only a move instruction for axis Y.

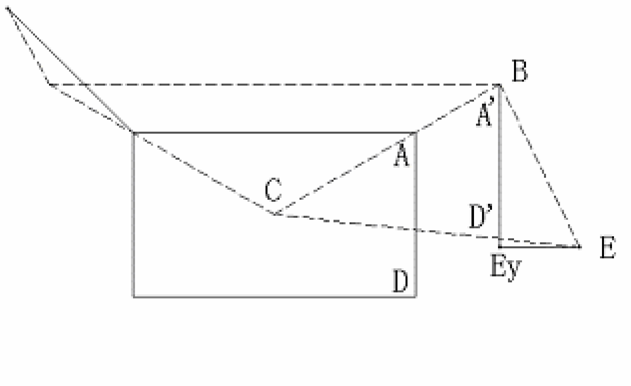


Fig. A

AD-A′D′

* Absolute

If Point D is absolute, the tool moves to Point E, which is converted from Point D.

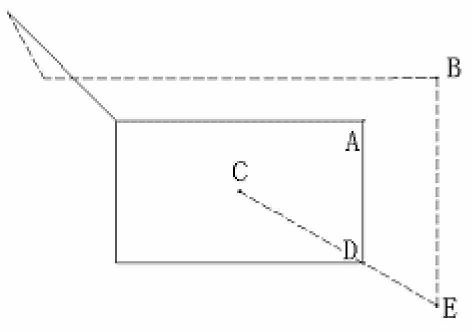


Fig. B

(b) When clearance is made by reset by parameter No.007 BIT3 CLER setting.

After switching from G51 to G50, the tool moves to point D' if the move instruction is incremental (see Fig. A) and to Point D if it is absolute (see Fig. B).

3.7 Cycle Machining Function

3.7.1 External moving function

As instructed by G81X—Y—L—, NC sends out external moving function signal once X—Y— positioning is completed. The machine side performs clamping, drilling and other specific operations and executes cycles according to the signal. Each positioning operation sends out the signal before clearance with G80 instruction. Whether G81 will be reset may be set by parameter No.007 BIT3 (CLER). The system is in G80 mode when it is switched on.

Circular positioning is conducted for L times depending on the numerical value behind the address L and external moving signal is given after each positioning. External moving signal will not be sent out when the blocks excluding X and Y are executed. Besides that G81 is used for external moving function, it can be applied to the following fixed cycles through parameter No.009 BIT5 (MCF) setting.

3.7.2 Fixed cycles (G73, G74, G76 and G80 ~ G89)

In general, a fixed cycle may employ a block including G codes instead of the several blocks for instructing machining operations in order to simplify programming.

At present, two types (A and B) of fixed cycles are available for your selection. Type A may use G80, G81, G82, G84, G85, G86 and G89 as listed in the table below while Type B all the G codes in the table.

Refer to Table 7.2 for the summary of fixed cycles.

G codes	Drilling operations (-Z direction)	Operations in hole bottom	Backing operations (＋Z direction)	Applications
G73	Intermittent feed	——	Rapid traverse (positioning)	High-speed peck drilling
G74	Cutting feed	Spindle forward rotation	Cutting feed	Reverse tapping
G76	Cutting feed	Oriented spindle stop	Rapid traverse (positioning)	Finished boring cycle (only for the 2nd group of fixed cycles)
G80	——	——	——	Cancellation
G81	Cutting feed	——	Rapid traverse (positioning)	Drilling cycle (fixed-point cycle)
G82	Cutting feed	Hold	Rapid traverse (positioning)	Drilling cycle (counterboring)
G83	Intermittent feed	——	Rapid traverse (positioning)	Peck drilling
G84	Cutting feed	Spindle reverse rotation	Cutting feed	Tapping
G85	Cutting feed	——	Cutting feed	Boring cycle
G86	Cutting feed	Spindle stop	Rapid traverse	Boring cycle
G87	Cutting feed	Spindle stop	Manual operation of rapid running	Boring cycle (reverse)
G88	Cutting feed	Hold, spindle stop	Manual operation of rapid running	Boring cycle
G89	Cutting feed	Hold	Cutting feed	Boring cycle

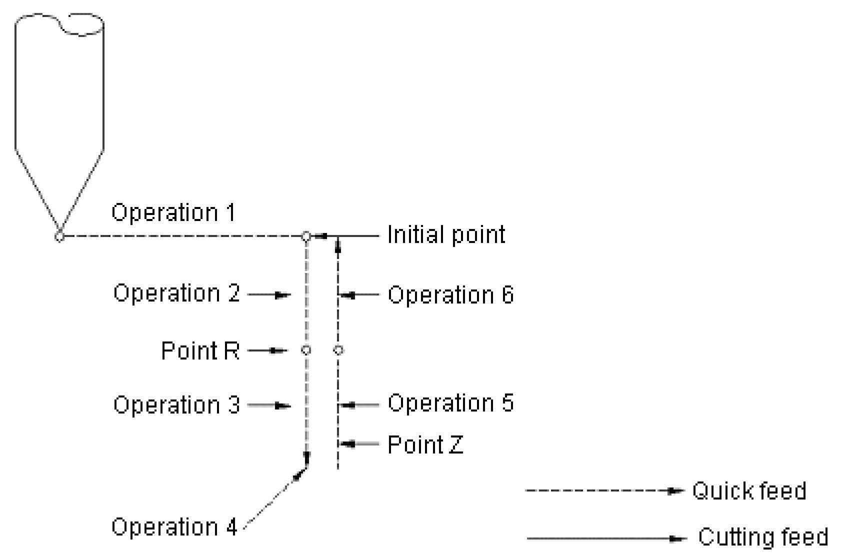
Note 1: Whether the signal [(SRV, SSP) fixed cycle I] output by NC or M codes (fixed cycle II) is used

to control the reverse rotation and stop of spindle is determined by parameter No.009 BIT7 (FIX2).

Note 2: In G87 mode, different operations are performed for fixed cycle I and II.

A fixed cycle normally consists of the following six operations.

Operations
1. Positioning of axes X and Y
2. Rapidly moving to Point R
3. Drilling
4. Performing relevant operations at hole bottom
5. Returning to Point R
6. Rapidly moving to initial point



Positioning is performed on XY plane and drilling is conducted in the direction of axis Z. Drilling in other planes or axial directions is not allowed. This is independent of plane selection G instruction.

These fixed cycles are specified in three modes with each specified by a special modal G code.

① Data type
- G90 Absolute!
- G91 Incremental

② Plane of return point
- G98 Initial-point plane
- G99 Point-A plane

③ G73, G74, G76, G80, G81 …… G89: See Table 7.2.

Note: Initial-point plane refers to the position of the absolute value in Z direction when fixed cycle cancellation mode changes to fixed cycle mode.

The data indicated in Fig. 7.2.2 depends on G90 or G91 mode.

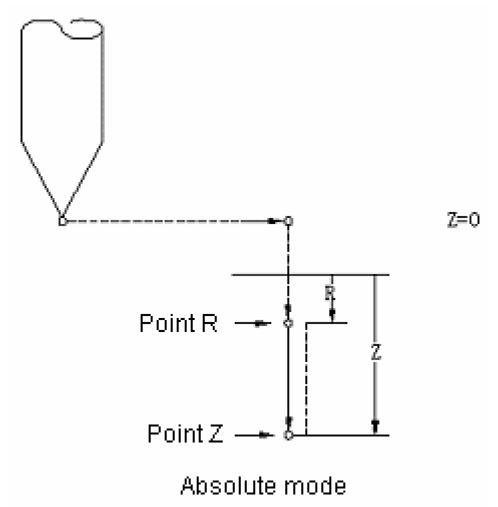
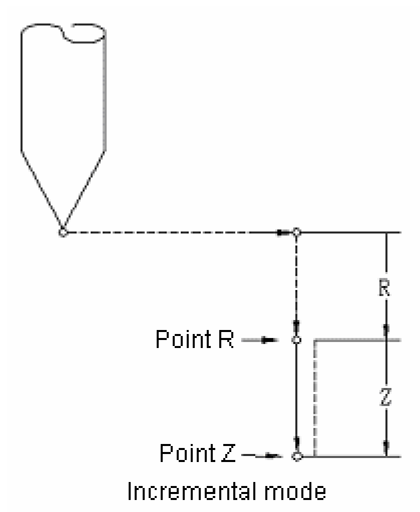
G90	G91
	

Figure 7.2.2: Absolute and increment programming

（A） During the returning operation, whether the tool returns to Point-R plane or initial-point plane depends on the setting of G98 or G99. See Fig. 7.2.3.

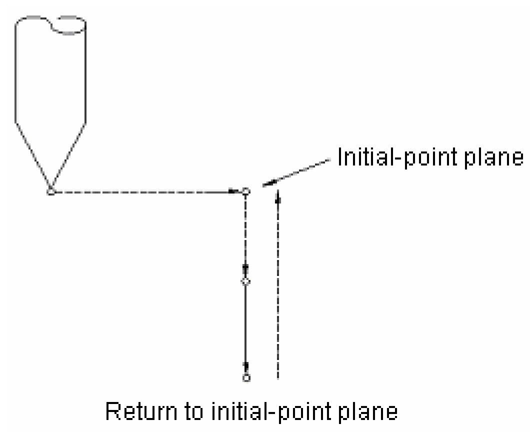
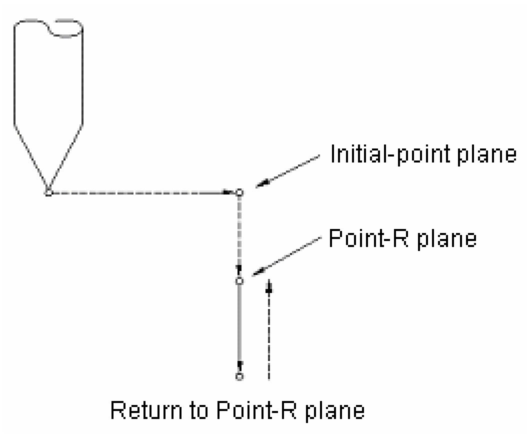
G98	G99
	

Fig. 7.2.3: Initial-point plane and Point-R plane

The initial point remains constant even touring operations in G99 mode. If the position of previous return is located in Point-R plane, the starting point is Point R. If the position of previous return is in initial-point plane, the initial point is used as the starting point.

（B） After G73, G74, G76 and G81 to G89, specifying the data related to touring may create a block. The data saved in control unit as modal value through the instruction data and related to the machining in a fixed cycle shall be instructed in the following format.

$$\underbrace{G\square\square}_{\text{Hole machining mode}}\quad \underbrace{X\text{———}Y\text{———}}_{\text{Hole position data}}\underbrace{Z\text{———}R\text{———}Q\text{———}P\text{———}F\text{———}}_{\text{Hole machining data}}\underbrace{L\text{———}:}_{\text{Cycle times}}$$

Hole machining mode:

G□□   (See Table 7.2)

Hole position data X, Y:

Hole position is specified by an absolute or incremental value. Whether the selection of tool path and feedrate will follow group 01 G codes (G02 and G03 are regarded as G01) or unconditionally follow G00 shall be determined by parameter No.009 BIT3 (FCUT).

Hole machining data:

Z: It is necessary to specify the increment of the distance from point R to hole bottom or the absolute coordinates of hole bottom. The feedrate of the operation 3 in Fig. 7.2.1 depends on F codes. Operation 5 feeds at rapid traverse (positioning)rate or the speed specified by F codes depending on different drilling modes.

R: R specifies the incremental value of the distance from the initial-point plane to point R or the absolute coordinates of point R. The feedrate in operations 2 and 6 is a rapid traverse (positioning) rate.

Q: The depth of each machining is specified in G73 or G83 mode and travel value in G76 or G87 (fixed cycle II) mode. (It is always an incremental value.)

P: To specify the dwell time in hole bottom: The relationship between time and specified value is similar to G04 instruction.

F: To specify cutting feedrate

Number of repetitions L: L is used to specify the number of repetitions of fixed cycles (Job 1 to 6). It is considered as 1 when L is omitted.

If L=0 is specified, machining will not be performed though the machining data of hole is saved in the system.

Once a drilling mode (G□□) is specified, it remains constant until other drilling modes or the G codes for fixed cycle cancellation are specified. So it needs not to be specified in each block when the same holes are machined in succession. The G codes for fixed cycle cancellation are the G codes in G80 or Group 01.

Once hoe machining data is specified, they remain constant until the data is changed or fixed cycle is disabled. Hence you only need to specify all necessary hole machining holes at the beginning of fixed cycles and change hole coordinate data in fixed cycles.

When necessary, the number of repetitions L shall be specified. The data of L is valid in the specified blocks. The cutting speed specified by F is maintained even fixed cycles are cancelled. Hole machining mode and data remain unchanged, but hole position data and repetitions are cancelled when the control unit is reset during fixed cycles. When the parameter (CLER) for specifying G80 mode for reset is set, however, hole machining data is also cleared.

The example of maintaining and canceling the above data is as follows:

① G00 X—M30:

② G81 X—Y—Z—R—F—L—: The original data Z, R and F of fixed cycles must be specified. Drilling operation is specified by G81 for L times.

③ Y—: When drilling mode and drilling data are the same as block ②, G81, Z, R and F may be omitted. Hole position is moved for Y— distance and drilling is performed once in G81 mode.

④ G82 X—P—L—: Hole position only moves X— in relation to ③. Drilling is performed for L times in G82 mode. Now the drilling data is used as follows.

Z, R and F ………………………Specified in block ②

⑤ G80 X—Y—M05:

Now touring operation will not be performed and all touring data except F codes will be cleared.

⑥ G85 X—Z—R—P—: Since addresses Z and R are cancelled in block ⑤, they shall be specified once again. Now F codes are identical with block ② and may be omitted. Address P is not needed in this block, but it is still saved.

⑦ X—Z—: Hole position has only moved X— and now drilling data is used as follows:

Z…………………Specified in block ⑦

R…………………Specified in block ⑥

F…………………Specified in block ②

⑧ G89 X—Y—: After the positioning movement specified by addresses X and Y, drilling is performed in G89 mode. Now the drilling data is applied as follows:

Z…………………Specified in block ⑦

R…………………Specified in block ⑥

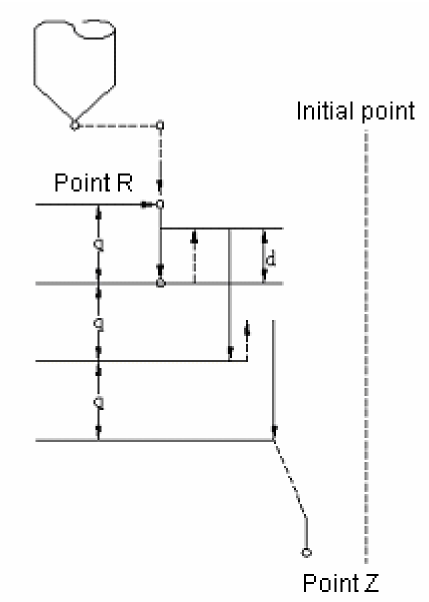
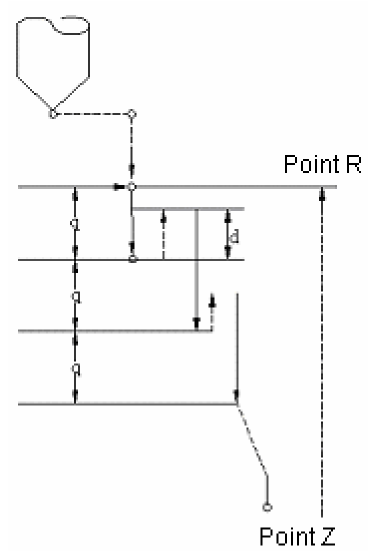
F…………………Specified in block ②

⑨ G01 X—Y—:

Now all hole machining modes and hole machining data (except F) are cancelled.

All machining means are described as follows:

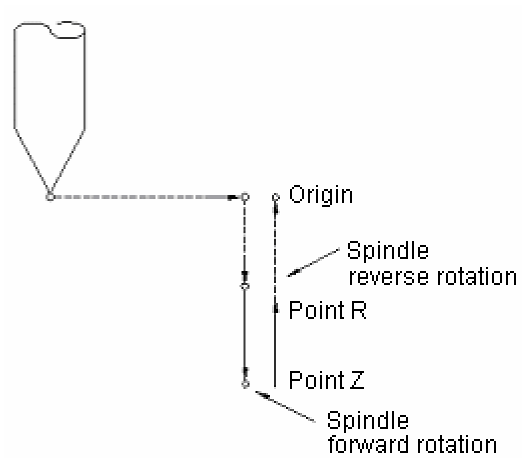
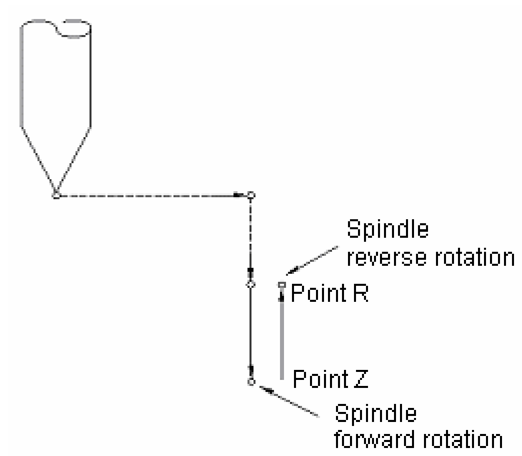
（1） G73 (high-speed peck drilling)

G73 (with G98)	G73 (with G99)
	

Backing “α” value is set by parameter No.067 (CYCR).

Drilling may be efficiently performed and cutting easily prevented by axis Z’s discontinuous feed. Backing is carried out at rapid traverse (positioning)rate.

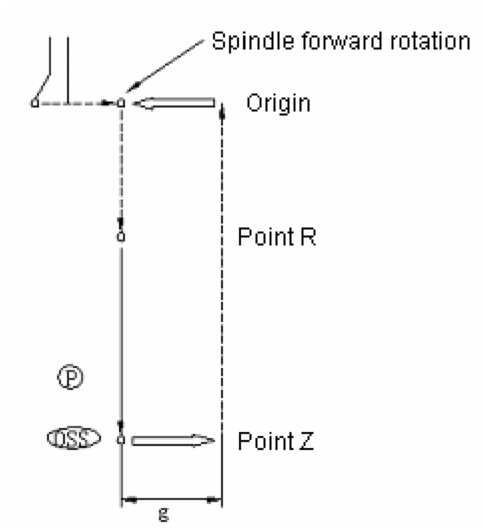
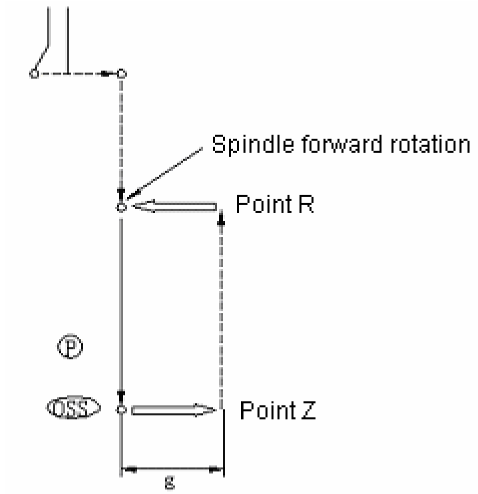
(2) G74 (left-hand tapping cycle)

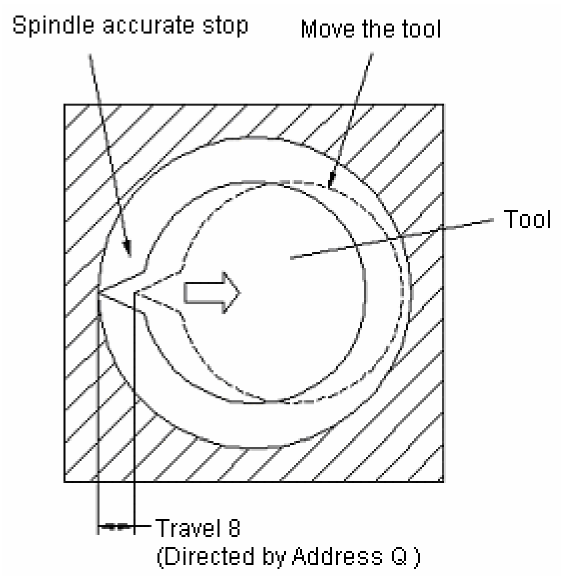
G74 (with G98)	G74 (with G99)
	

The instruction specifies that the spindle forward rotate at hole bottom and then perform a left-hand tapping cycle.

Note: If feedrate override is neglected or feed hold occurs during tapping with G74, machining will not stop until the fixed cycle is completed.

(3)　G76 (Finish boring)

<table>
<tr><th>G76 (with G98)</th><th>G76 (with G99)</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>



Ⓟ　Hold

OSS　Oriented spindle stop (spindle stops at a fixed point)

⇨　Tool motion

┈┈>　Rapid traverse

──►　Cutting feed

～～～>　Manual feed　　“d” is set by parameter.

Note 1: G76 can be used only when the output M codes set by parameter No.009 BIT7 (FIX2) are used as the output signals for spindle reverse rotation, spindle forward rotation and spindle accurate stop.

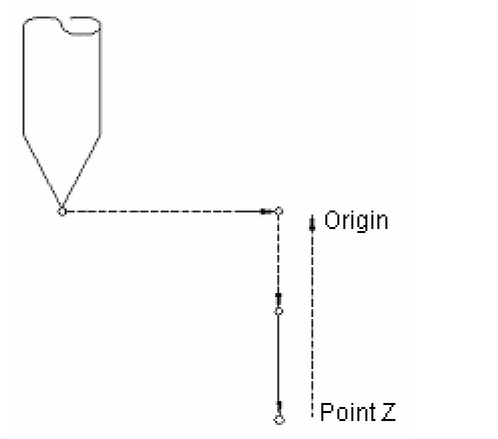
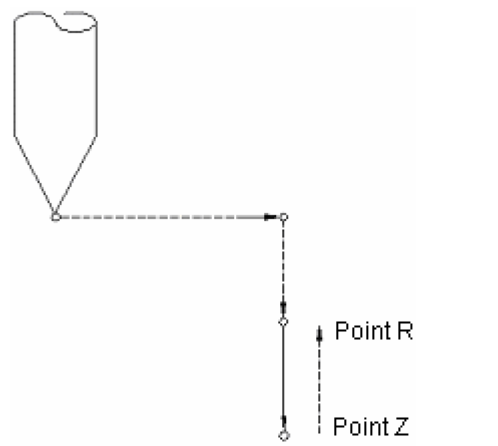
It is possible to accurately and efficiently perform hole machining and not to damage workpiece surface by stopping the spindle in oriented position at hole bottom and then removing the tool bit after it offsets for departing machining. Offset is specified by the address Q (always positive). The minus sign will be neglected if a negative number is used. An offset direction shall be preset between (＋X, ＋Y) and (－X,－Y) by parameter No.022 BIT4,5 (PMXY1, 2). Note that Q value is modal in a fixed cycle mode and indicates the cutting depth in G73 and G83.

Even at the hole bottom, address I and J may specify offset for the tool. Axes X and Y moves using linear interpolation according to parameter No.022 BIT6 (SIJ) setting and replace Q with the incremental value specified by I and J. Hence they may offset upward in any direction. Feedrate is identical with the speed specified by F codes. I and J are modal in mode of fixed cycles. Hole machining can not be performed only by specifying I and J. The instructions are only used to specify I and J again.

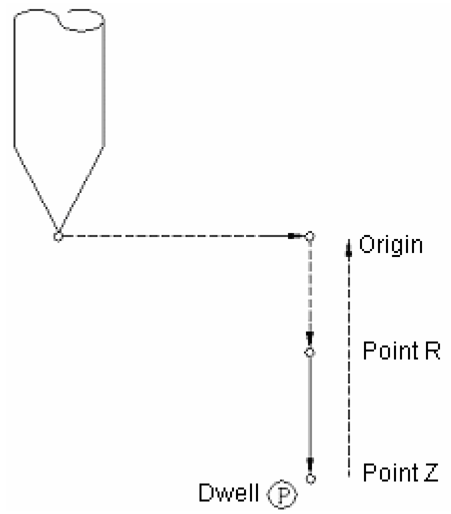
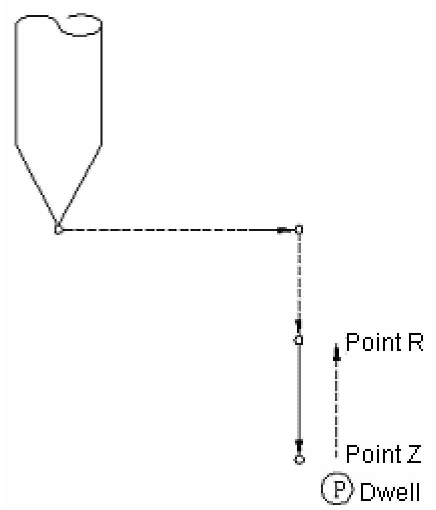
(4) G80 (To disable fixed cycles)

The instruction disables fixed cycles (G73, G74, G76, G81～G89). Then NC starts to perform normal operations. The data of Point R and Point Z is also disabled, namely the tool does not travel and other machine data is cancelled.

(5) G81 (Drilling cycle, fixed-point drilling)

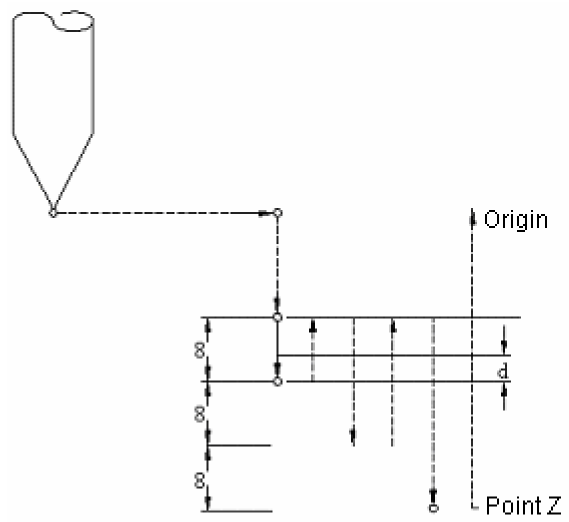
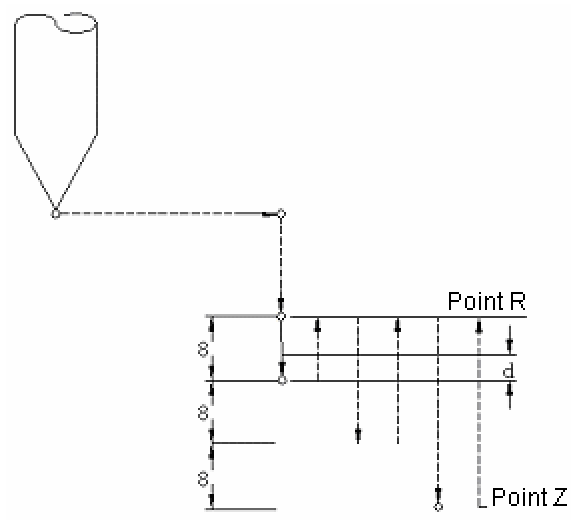
G81 (with G98)	G81 (with G99)
	

(6) G82 (drilling cycle, boring)

<table>
<tr><th>G76 (with G98)</th><th>G76 (with G99)</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Except that it holds at hole bottom and then retracts, the instruction is similar to G81 (The dwell time is specified by address P). The accuracy of hole depth is improved because it holds at hole bottom.

(7) G83 (Peck drilling cycle)

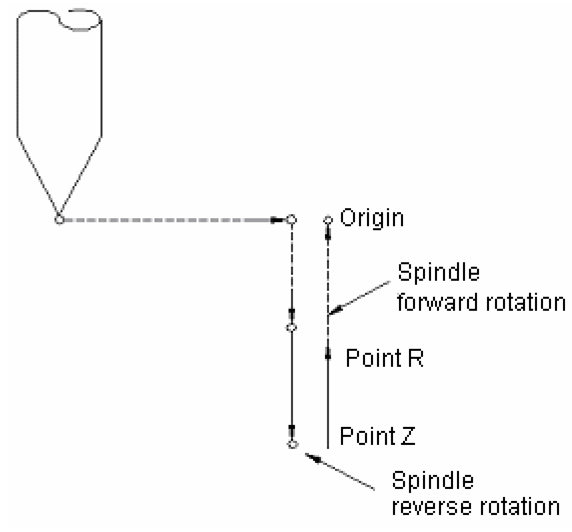
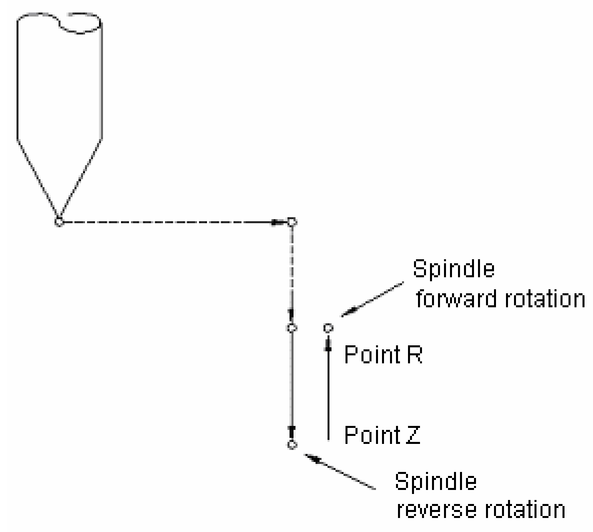
<table>
<tr><th>G83 (with G98)</th><th>G83 (with G99)</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Now the format of the instruction is as follows:

G83 X—Y—Z—Q—R—F—;

Q indicates the cutting depth of each time and always specifies it in incremental value. Rapid traverse (positioning) is changed into cutting feed at a distance of a millimeter or inch from the machined position. Q value is specified in positive number. Minus sign is neglected if it is specified with a negative number. Distance “d” is set by parameter (CYCD).

(8) G84 (Tapping cycle)

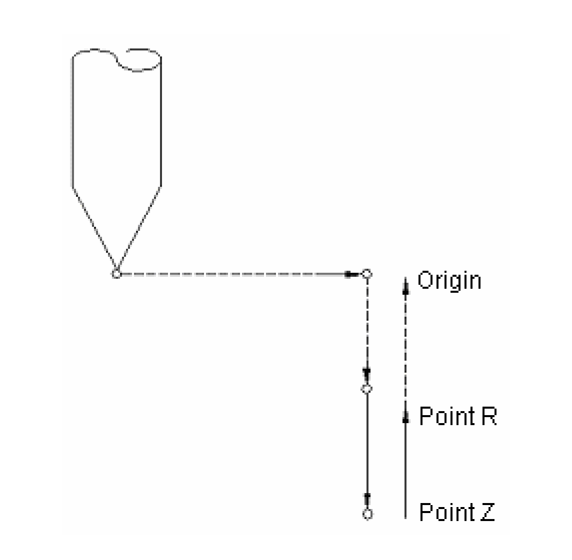
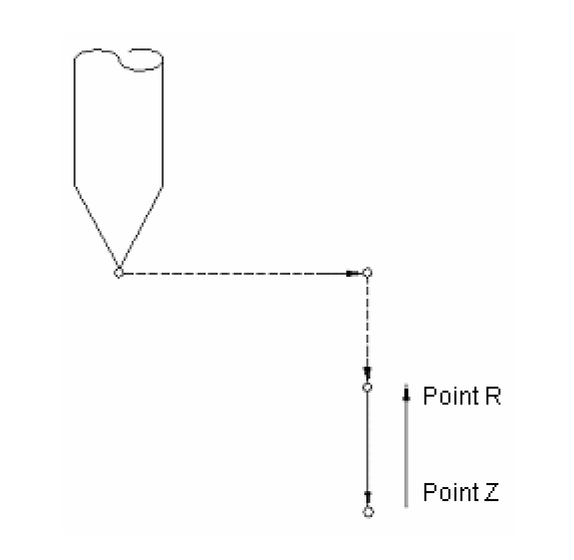
G84 (with G98)	G84 (with G99)
	

Rapid traverse (positioning) Cutting feed

The instruction requires the spindle to counterclockwise rotate at hole bottom and perform a tapping cycle.

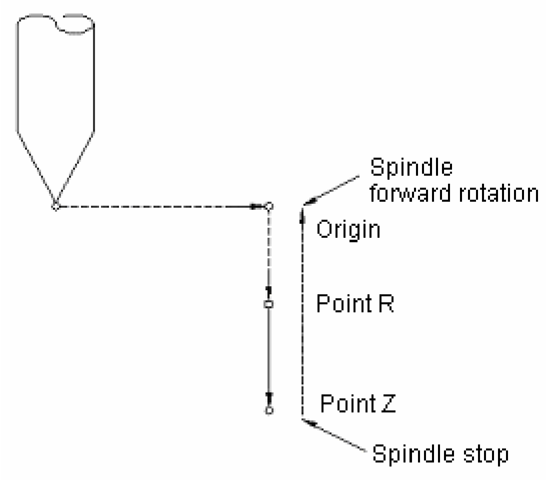
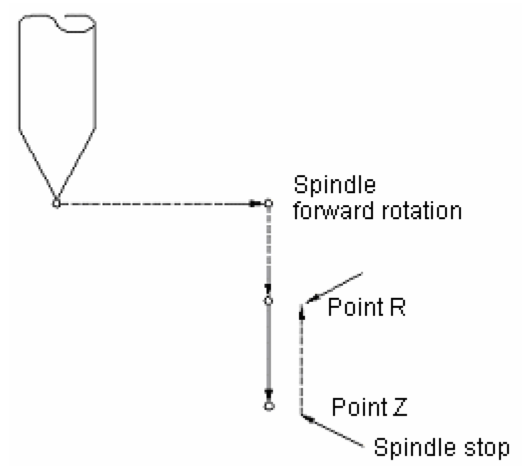
Note: Feed override is neglected and it shall not stop until the end of the cycle during the tapping instructed by G84. It will not be affected even Feedrate occurs at this time.

(9) G85 (Boring cycle)

G85 (with G98)	G85 (with G99)
	

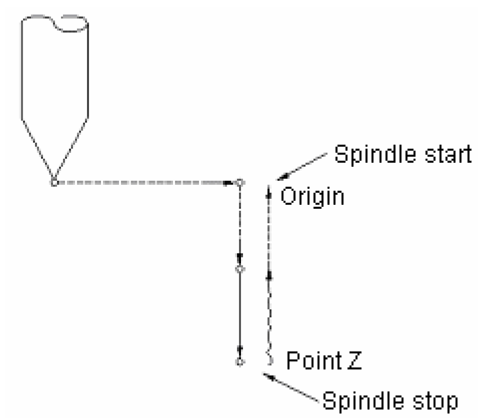
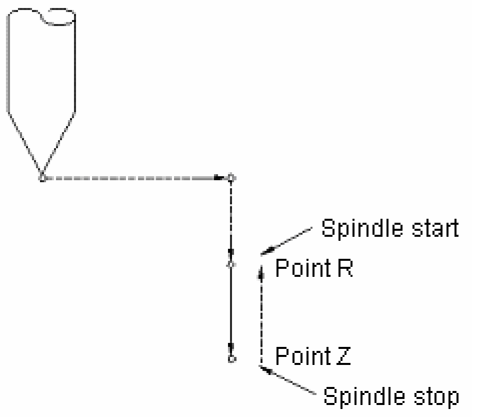
The instruction is similar to G84 except that spindle does not rotate reversely at hole bottom.

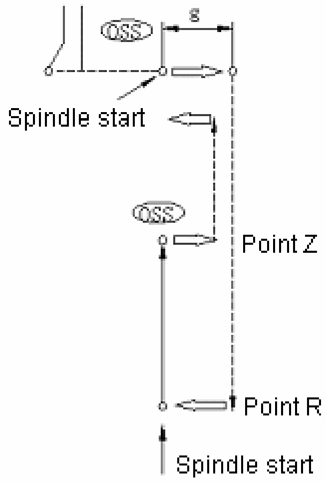
(10) G86 (Boring cycle)

<table>
<tr><th>G86 (with G98)</th><th>G86 (with G99)</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

The instruction is equivalent to G81 except stop at hole bottom and return by rapid traverse.

(11) G87 (Boring cycle/Reverse boring cycle)

<table>
<tr><th></th><th>G87 (with G98)</th><th>G87 (with G99)</th></tr>
<tr><td>Fixed cycle I</td><td>
</td><td>
</td></tr>
</table>

<table>
<tr>
<td>Fixed cycle II</td>
<td>
</td>
<td>Not use</td>
</tr>
</table>

Rapid traverse

Cutting feed

Manual feed

Tool offset

Spindle accurate stop

Fixed cycle (boring cycle)

The control system enters into Feedrate mode when the tool reaches the hole bottom and the spindle stops. In this condition, the tool may travel by manual means. Any manual operation may be performed. For the purpose of safety, however, the tool shall be withdrawn from the hole.

To restart machining, it is necessary to switch to DNC or automatic mode and press the CYCLE START key. The spindle rotates forward after the tool returns to the position of origin or Point R through G98 and G99. Then the instruction of the next block is executed.

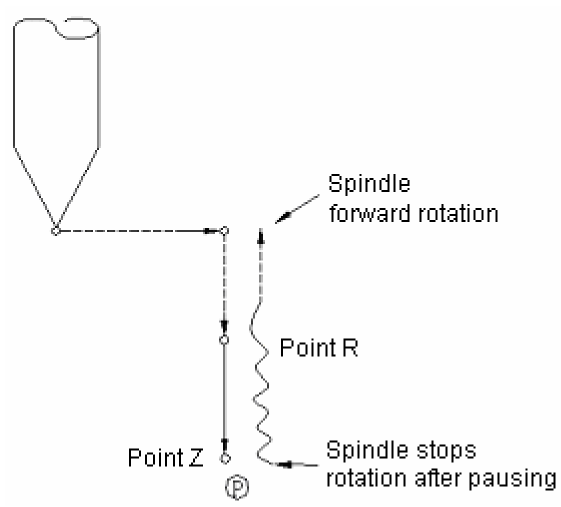
Fixed cycle II (reverse boring cycle)

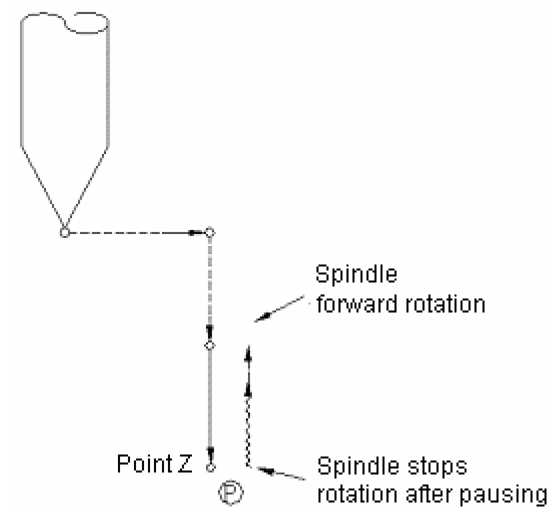
Once axes X and Y are positioned, the spindle stops and the tool offset in the direction contrary to the tool nose and is rapidly positioned at the hole bottom (Point R). Here the tool returns in the original amount of offset. The spindle starts clockwise and performs machining in the direction of axis Z until Point Z. It does not suspend even under P instruction. After the spindle stops here again, the tool backs out from the original offset and moves upward. Then the tool returns to the origin and backs out in the original offset. The spindle rotates forward and the next block starts. The offsets and directions of axes X and Y are identical with G76 (Refer to G76 and G87 for the setting of direction.)

Note: Fixed cycle I is set by parameter No.009 BIT7(FIX2) and signals SRV and SSP are used as the output signals for spindle reverse rotation and spindle stop.

Fixed cycle II is set by parameter No.009 BIT7 (FIX2) and M codes are used as the output signals for spindle reverse rotation, spindle stop and spindle oriented stop.

(12) G88 (boring cycle)

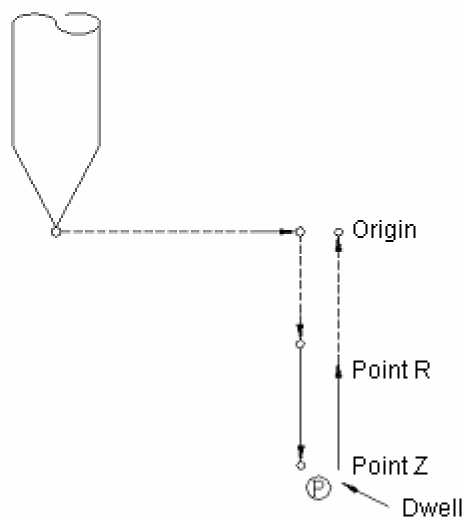
<table>
<tr><th>G88 (with G98)</th><th>G88 (with G99)</th></tr>
<tr><td>


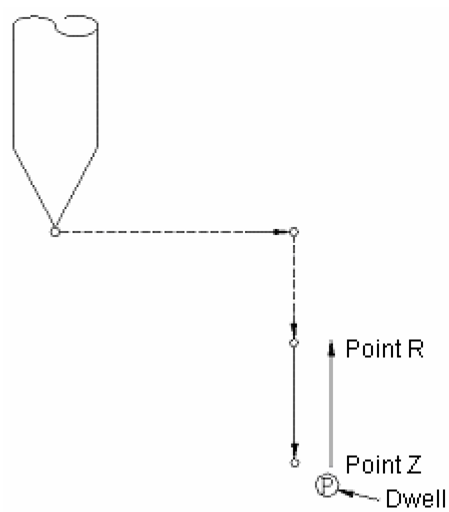
</td><td>


</td></tr>
</table>

Except that the spindle stops rotation after pausing at hole bottom, the instruction is similar to G87 (fixed cycle I).

(13) G89 (boring cycle)

<table>
<tr><th>G89 (with G98)</th><th>G89 (with G99)</th></tr>
<tr><td>


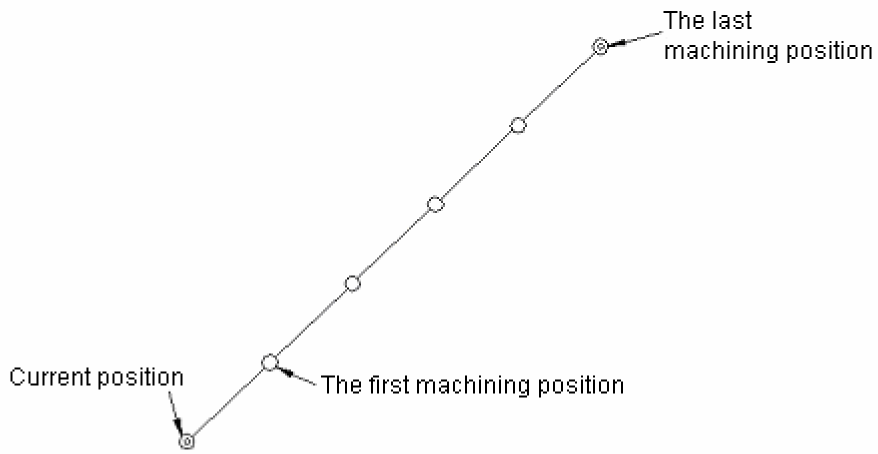
</td><td>


</td></tr>
</table>

Though it is similar to G85, the spindle needs to dwell at hole bottom.

3.7.2.1 Repeating a fixed cycle

To repeatedly machine equally spaced holes with the same fixed cycle, it is possible to specify the number of repetitions with address L, which is up to 9999 and is only valid in some existing block.



G81 X—Y—Z—R—L 5 F—;

X—Y— specifies the first machining position in an incremental value (in G91 mode). If the instruction specifies it in an absolute value (in G90 mode), drilling will be repeatedly performed at the same point.

In a fixed cycle, the time constant of automatic acceleration/deceleration can be automatically switched. It is switched to the time constant of rapid traverse (positioning) or cutting feed depends on all feed motions. After deceleration is completed at the end of the motion, the system switches to the next operation.

In G98 instruction, however, it returns to Point R from hole bottom at rapid traverse (positioning)rate and then to the origin without deceleration.

Notes regarding fixed cycles:

Note 1: Spindle rotation function shall be specified with M codes before specifying a fixed cycle.

ER:

M03: Spindle forward rotation (clockwise)

⋮

G□□……………… ; Correct

⋮

M05: …………………… Spindle stop

⋮

G□□ ………… ; Incorrect (so specify M03 or M04 before the block)

⋮

Note 2: In the mode of fixed cycle, touring operation may be performed provided that the position data corresponding to axes X, Y and Z and the 4th axis are specified in the block.

Touring cannot be performed if the position data is not specified. It will not be performed even the dwell instruction (G04 P— ;) is incorporated. If address X is used to specify the dwell time (G04 X— ;), touring operation will not be performed.

G00　X— :

G81　X—Y—Z—R—F—P—L— :

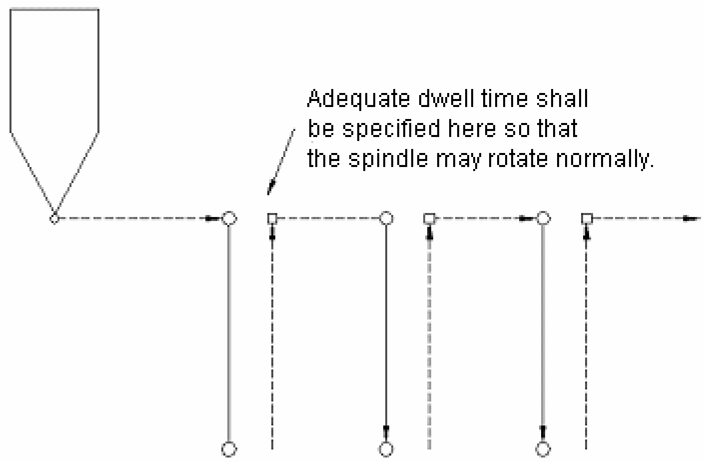
;　(Not to drill)

F— ;　(Not to drill but modify the value of F)

M— ;　(Not to drill but perform M function)

G04　P— ; (Not to drill; the P data for drilling will not be modified by the instruction)

Note 3: For the fixed cycles G74, G84 and G86 for controlling spindle rotation, the spindle fails to reach the normal rotational speed before drilling because drilling is continuously performed when the spacing between holes and the distance from the origin to Point R are relatively short. In this case, a G04 (HOLD) shall be instructed to insert between all drilling motions. Therefore, the program below does not specify the number of repetitions with through L.



G00　M—;

G86　X—Y—Z—R—F—;

G04　P—;　(Not to drill but to feedrate)

　X—Y—;

G04　P—;　(Not to drill but to feedrate)

⋮

Whether it is necessary to perform the operation or not depends on the performance of the machine. Refer to the manual supplied by the manufacturer of the machine.

Note 4: The above-mentioned fixed cycles may be deleted with G00, G01, G02 or G03. When G00～G03 is instructed in a fixed cycle, the following motions will be carried out.

# indicates 0, 1, 2 or 3.

G0# indicates G00, G01, G02 or G03.

G□□ indicates a fixed cycle.

G0# G□□ X—Y—Z—R—Q—P—F—L—;

(To perform a fixed cycle)

G□□ G0# X—Y—R—Q—P—F—L—;

The tool's travel along axes X and Y depends on G0# codes. The values of R, P and L will be neglected and F codes saved.

G□□ G0# X—Y—Z—R—Q—P—F—L—;

If 3-axis link function is not provided, the block will cause an alarm.

Note 5: When M codes and a fixed cycle are specified in the same program, M codes and MF signals will be sent out during the first positioning (Operation 1). The machining of the next hole will be performed after receiving the signal FIN at the end of the cycle. The M codes and MF signals are only sent out in the first cycle rather than the cycles that follow provided that the cycles are provided with L instruction for repeated operation.

Note 6: Tool offset instructions (G45～G48) are ignored in the mode of fixed cycles.

Note 7: Tool length compensation (Operation 2 in Fig. 7.2.1) will be performed during the positioning of Point R if it (G43 or G44) is instructed in the mode of fixed cycles.

Note 8: Cautions of operation

(a) Reset

When the control unit is stopped by pressing the RESET key of EMERGENT STOP bottom in a fixed cycle, normally touring mode and touring data will remain unchanged. Fully note this during restarting. Touring mode and touring data may be cancelled by parameter No.007 BIT3 (CLER).

(b) Single block

When a fixed cycle is performed in the single block mode of operation, the control unit will stop at the end points of the operations 1, 2 and 6 in Fig. 7.2.1. As a result, it needs to be started for three times for drilling a hole.

The FEEDRATE lamp illuminates at the end points of Operation 1 and 2. For Operation 6, if repeated cycles do not end up in the block, it will stop in Feedrate mode or other stop mode.

(c) Feedrate

Once the FEEDRATE button is pressed during operations 3 to 5 in G74 or G84, the FEEDRATE lamp will illuminate and the control unit will continue to perform Operation 6 before stop. If Feedrate is available during Operation 6, the operation will immediately stop.

(d) Feedrate override

Feedrate is fixed at 100% during the fixed cycle G74 or G84.

(e) Manual absolute value

The MANUAL ABSOLUTE switch is used to switch between the manual operations of G87 (fixed cycle I) and G88 in the two modes below:

ON: Point R and origin are identical with the programming.

OFF: Point R and origin are offset through manual unit.

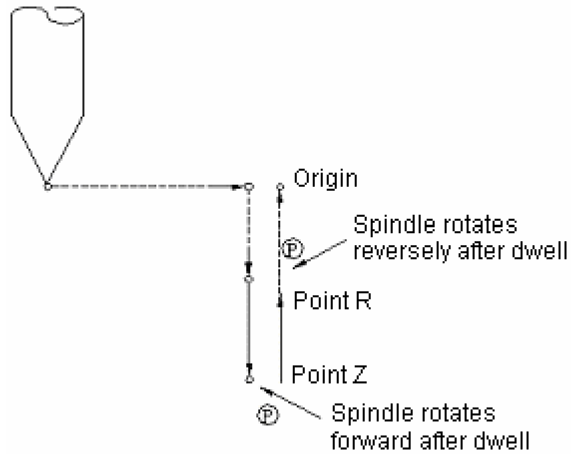
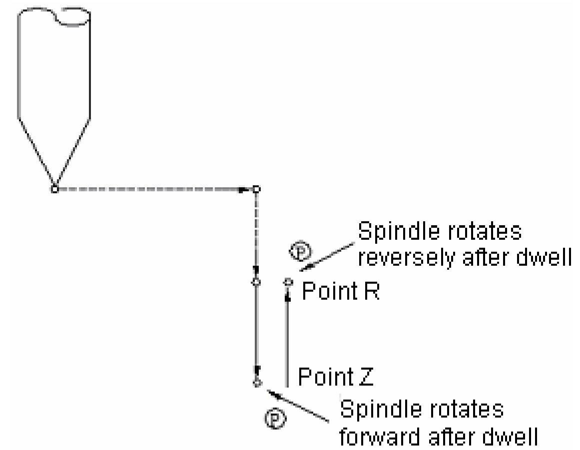
Note 9: Fixed cycles G74 and G84 may be changed by parameter No.022 BIT2 (FXCD).

The spindle may dwell before forward and reverse rotation using the time instructed by P by setting the parameter.

It is very necessary when using a dedicated tapping device. Thread machining is performed through the forward/backward movement caused by the rotary force of the tap during dwell without the travel of axis Z.

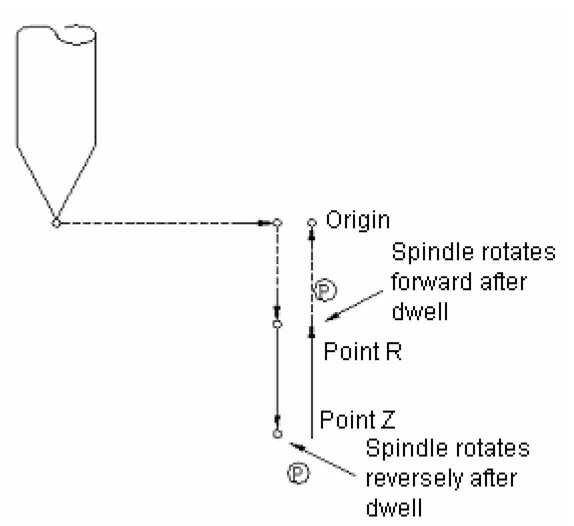
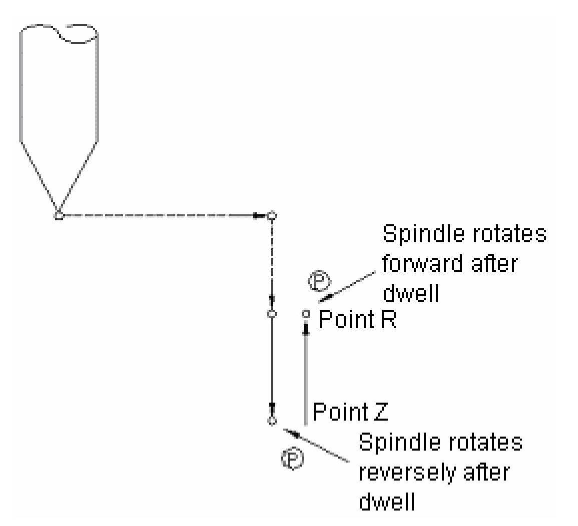
(a) Left-hand tapping cycle G74

G74 X—Y—Z—R—P—F—;

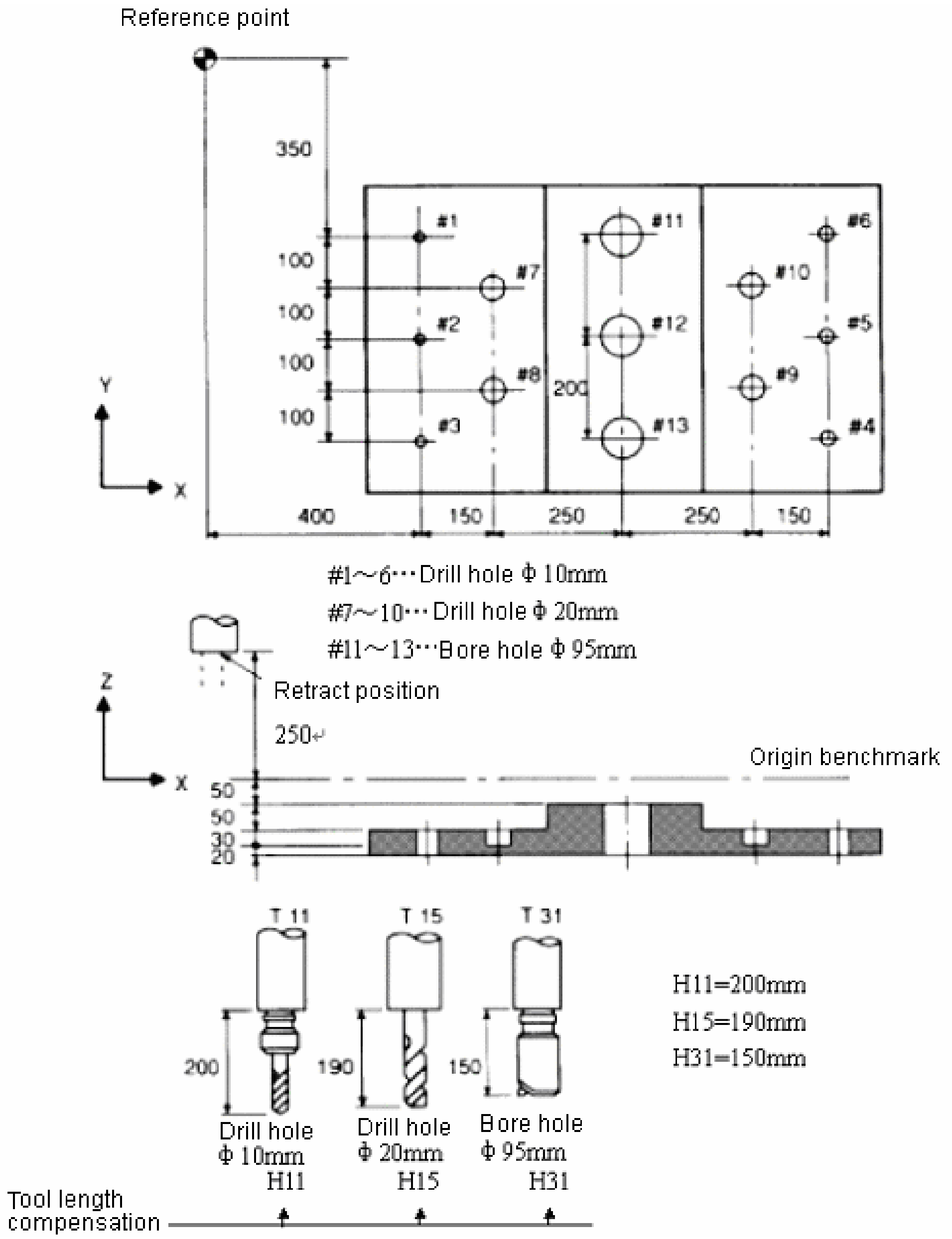
<table>
<tr><th>G74 (with G98)</th><th>G74 (with G99)</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

(b) G84 (Tapping cycle)

G84 X—Y—Z—R—P—F—;

<table>
<tr><th>G84 (with G98)</th><th>G84 (with G99)</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Example 7.2.2: Programming using tool length compensation and fixed cycle.



Example of program

N001　G92　X0　Y0　Z0; Set the coordinate system at the reference point

N002　G90　G00　Z250.0　T11　M06; Tool change

N003　G42　Z0　H11; Origin & tool length compensation

N004　S30　M03; Spindle start

N005　G99　G81　X400.0　Y－350.0　Z－153.0

R－97.0　F120; Drill 1# hole after positioning

N006　Y－550.0; Drill 2# hole after positioning and returning to Point R

N007　G98　Y－750.0; Drill 3# hole after positioning and returning to the origin

N008　G99　X1200.0; Drill 4# hole after positioning and return to the origin

N009　Y－550.0; Drill 5# hole after positioning and return to Point R

N010　G98　Y－350.0; Drill 6# hole after positioning and return to the origin

N011　G00　X0　Y0　M05; Return to the reference point and stop the spindle

N012　G49　Z250.0　T15　M06; Tool length compensation and tool change

N013　G43　Z0　H15; Origin, tool length compensation

N014　S20　M03; Spindle starts

N015　G99　G82　X550.0　Y－450.0　Z－130.0

R—97.0　P300　F70; Drill 7# hole after positioning and return to Point R

N016　G98　Y－650.0; Drill 8# hole after positioning and return to the origin

N017　G99　X1050.0; Drill 9# hole after positioning and return to Point R

N018　G98　Y－450.0; Drill 10# hole after positioning and return to the origin

N019　G00　X0　Y0　M05; Return to the reference point and stop the spindle

N020　G49　Z250.0　T31　M06; Cancel tool length compensation and change a tool

N021　G43　Z0　H31; Origin, tool length compensation

N022　S10　M03; Spindle starts

N023　G85　G99　X800.0　Y－350.0　Z－153.0　R－47.0　F50;

Drill 11# hole after positioning and return to Point R

N024　G91　Y－200.0　L2; Drill 12# and 13# holes after positioning and return to Point R

N025　G00　G90　X0　Y0　Z0　M05; Return to the reference point and the spindle stops

N026　G49　G91　Z0; Cancel tool length compensation

M02: Program stops

Note: When the number of repetitions L is programmed in G98/G99, the tool returns from the first drill hole to the origin (G98) or Point R (G99).

3.7.3 Specifying an origin and Point R in a fixed cycle (G98, G99)

G98 and G99 specifies whether the return point in a fixed cycle is the origin or Point R as shown in Fig. 7.3 respectively. If the return position of the first fixed cycle is the origin, the starting point of the cycle will be the origin. If it is at point R, the starting point of the cycle will be Point R. In general, G99 is employed to drill the first hole while G98 the last. To program the number of repetitions L, G98 shall be specified for the first hole so that the tool may return to the origin.

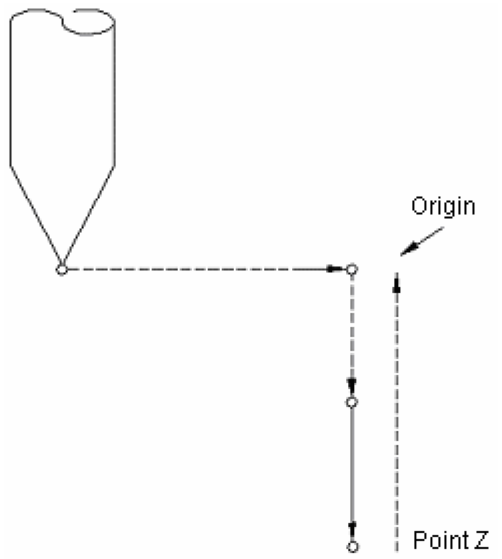
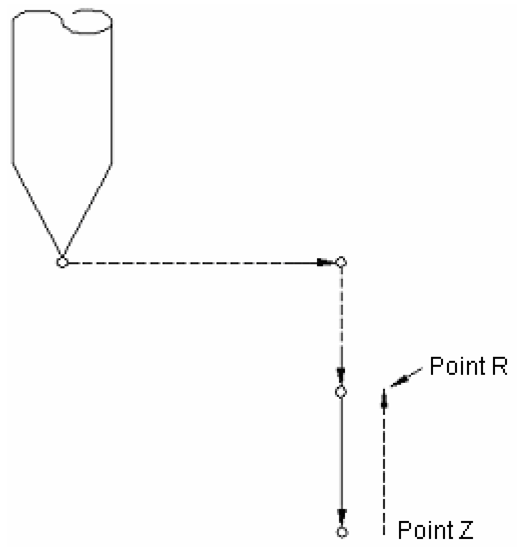
<table>
<tr><th>G98</th><th>G99</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Fig. 7.3: Origin and Point R

3.8 Spindle Function (S function), Tool Function (T function), Miscellaneous Function (M function) and Secondary Miscellaneous Function (B function)

A numerical value is specified behind address S, T, M or B so that BCD signal or strobe pulse is transmitted to the NC machine. This is mainly used for the switching function of the numerically controlled machine.

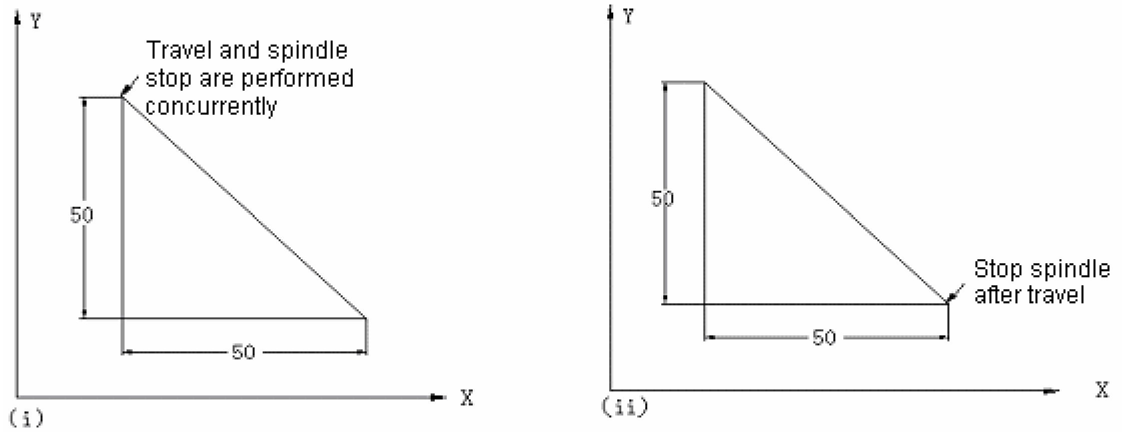
S codes are used for spindle control, T codes for tool change, M codes for switching all the functions controlling the machine and B codes for swivel table division. Refer to the application instructions for these addresses and codes supplied by manufacturer.

When a move instruction is specified in the same block as the codes S, T, M or B, the instruction shall be executed by one of the following two methods.

(i) The move instruction is executed at the same time as S, T, M or B,

(ii) S, T, M or B function is performed after the move function.

(Example): N1　G01　G91　X50.0　Y－50.0　M05: (Spindle stop)



Which method shall be selected depends on manufacturer's setting. In general, both the two methods are available for a numerically controlled machine. Refer to the manual supplied by the manufacture of the machine.

3.8.1 Spindle function (S function)

3.8.1.1 S 2 digits

Spindle speed is controlled by address S and the 2 digits that follow. Refer to the manual supplied by the manufacture of the machine for details.

Note: When S 4 digits are specified after S 2 digits, the last 2 digits will be valid.

3.8.1.2 S 4 digits

Spindle speed (r/min) is directly specified by the 5 digits following address S. The unit of spindle speed (up to 30000r/min) is set by the manufacturer of the machine.

3.8.2 Constant surface speed control

When surface speed (the tool speed relative to workpiece) is specified after S instruction, the constant speed control function always keeps surface speed unchanged with the change of the tool position and supplies a control voltage corresponding to the calculated spindle speed so that the spindle may rotate at correct surface speed.

The units of surface are as listed in the table below:

Input unit	Unit of surface speed
M (Metric system)	m/min
Inch (Inch system)	Inch/min

The units of surface speed may vary depending on different manufacturers.

3.8.2.1 Instructed methods

The following G codes are specified for the control of constant surface speed.

G code	Meaning	Unit
G96	Constant surface speed control ON	m/min or inch/min
G97	Constant surface speed control OFF	r/min

To exert constant surface speed control, it is necessary to establish a workpiece coordinate system so that the coordinates of center of the rotation axis are 0 (the axis that exerts constant surface speed control).

The axis for constant surface speed control may be selected through programming instruction.

$$\text{G96P}\left\{\begin{matrix}1\\2\\3\\4\end{matrix}\right\}\text{——;}$$

P1……………………Constant surface speed control is designated to axis X.
P2……………………Constant surface speed control is designated to axis Y.

P3……………………Constant surface speed control is designated to axis Z.

P4……………………Constant surface speed control is designated to the 4th axis.

P5……………………Constant surface speed control is designated to 5th axis.

For P0 or that no axis is specified, the present concerned axis is predetermined by No. 315 parameter.

Note 1: To set an axis for constant surface speed control by programming, make sure to specify Pα (α=1, 2, 3, 4 or5). Otherwise the axis set by parameter setting will be under control. Whenever G96 is set again, Pα must be always specified. This is independent of whether G96 Pα is specified before that.

Note 2: The surface speed (S) in G96 is considered as S=0 until M03 or M04 is specified. That is to say, S function cannot be performed before specifying M03 or M04 [only when parameter TCW (the 7th digit of No. 010 parameter) is 1].

3.8.2.2 Spindle speed

The specified surface speed or spindle speed may be overridden in 50, 60, 70, 80, 90, 100, 110 or 120% by the signal from the operation control panel of the machine.

3.8.2.3 Restraint of maximum rotational speed of spindle

During constant surface speed control, the value in rpm following G92 specifies the maximum rotational speed of spindle.

G92 S—:

During constant surface speed control, the system will automatically restrain the rotational speed of spindle to the maximum value if the speed is higher than the programmed value.

3.8.2.4 Rapid traverse (positioning) (G00)

In the rapid traverse (positioning) blocks specified by G00, it is impossible to exert constant surface speed control by calculating the surface speed in the position of the tool at all times. However, the control may be achieved by calculating the surface speed from the starting point to the end point of the block. Machining cannot be carried out in the conditions of rapid traverse (positioning).

Note 1: Constant surface speed control is disabled (G97) during power on.

Note 2: Spindle override becomes effective when parameter SOV (the 5th digit of No. 010 parameter) is preset to 1.

Note 3: The maximum spindle speed cannot be preset (or restrained) during power on.

Note 4: The maximum spindle rotational speed is only restrained in G96 mode. It needs not to be retrained in G97 mode. However, the spindle motor is restrained by No.136 parameter in G97 mode.

Note 5: G92 S0 means that the spindle speed is restrained to 0 rpm.

Note 6: The S value specified in G96 mode is still saved even after it is switched from G96 mode to G97 mode. The value will be restored when it is returned from G97 mode to G96 mode.

G96 559: (50m/min or 50 inch/min)

G97 S1000: (1000r/min)

G96 X3000: (50m/min or 50inch/min)

Note 7: When tool length compensation (G43 or G44) was made earlier, programming coordinates shall be used to calculate the constant surface speed. When tool position offset (G45～G48) was made, the existing value shall be used to calculate the constant surface speed.

Note 8: When the machine is locked, calculation about constant surface speed shall be made depending on the change in the coordinates of the axis with constant speed control.

Note 9: Constant surface speed control is also available during thread machining. Therefore, it is recommended to use G97 instruction to disable constant surface speed control before face thread and taper thread chasing because servo system does not respond during the change of spindle rotational speed.

Note 10: Constant surface speed (feed per minute) control mode (G96) is also allowable in G94 mode.

Note 11: When switching from G96 mode to G97 mode, the final spindle rotational speed, in G96 mode, is used as the S codes in G97 mode if S codes (r/min) is predetermined in G97 block.

N111 G97 S800: 800r/min

┆

N222 G96 S100: 100m/min

┆

N333 G97: X r/min

X the spindle rotational speed X rpm in the blocks before N333, alternatively the spindle speed remains unchanged when the mode has been changed from G96 to G97. ,

The S value newly specified in G96 mode will be enabled when switching from G97 to G96 mode.

If S is not specified, S=0m/min (inch/min).

3.8.3 Tool function (T function)

Tool function is specified by the 2 or 4 digits after address T. The relationship between T codes and the tool is set by the manufacturer of the machine.

3.8.4 Miscellaneous function (M function)

When 2 digits are set after address M, a 2-digit BCD code and strobe signal will be sent out. These signals are designed for the ON/OFF control of the functions of the machine. An M code may be set in a block. When more than two M codes are concurrently set, the last code will be valid. The setting of M codes may vary depending on different manufacturers of machine.

The following M codes have special meanings.

(1) M02, M30: End of program

(i) This indicates the end of the main program and is necessary for recording NC instructions in storage in communication mode.

(ii) Automatic operation stop and reset of NC unit (It is different depending on manufacturers).

(iii) It will return to the starting point of program after the end of the program for M30, and return to the starting point for M02.

(2) M00: Program stops

Automatic operation stops after the end of the block with M00. When the program stops, all modal data remains unchanged as in the operations of a single block. The cycle operation restarts by setting NC start-up (This may vary depending on different manufacturers).

(3) M01: Selection stops

Just like M00, automatic operation stops after the end of the block with M01. The code is valid only when the SELECTION STOP switch on the operation panel of the machine is pressed.

(4) M98: Call of subprogram

The code is intended to call a subprogram. Refer Chapter 9 for details.

(5) M99: End of subprogram

The code indicates the end of a subprogram. It returns to the main program once M99 is executed. See Chapter 9 for details.

Note 1: The block following M00, M01, M02 or M30 cannot be read in the buffer register. Similarly, the next block may set the M codes not to make intermediate conversion through 2 parameters. Refer to the section regarding M codes in the manual supplied by the manufacturer of the machine.

Note 2: Code signal or strobe signal will not be sent out during the execution of M98 or M99.

Note 3: The M codes other than M98 and M99 will be processed by the machine side instead of the NC unit. Refer to the instruction manual supplied by the manufacturer of the machine.

3.8.5 Secondary miscellaneous function (B function)

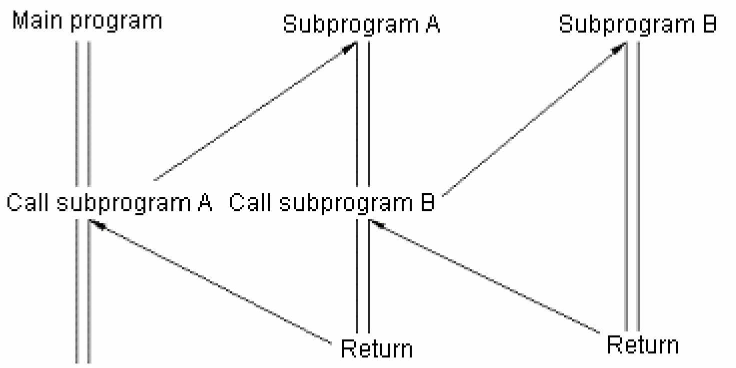
Worktable indexing is set by address B and the 3 digits that follow. Different manufacturers may have different settings for the division value corresponding to B codes.

3.9 Subprogram

When some areas in fixed sequences or those repeatedly appear are included in a program, these sequences or areas may be saved in storage as subprograms so simplify programming.

A subprogram can be called in automatic mode of operation. A subprogram can also call another subprogram.

That a main program calls a subprogram is called primary call. Secondary call of subprogram is performed as follows.



A call instruction can be used to repeatedly call a subprogram. A call instruction may repeat calling for 9999 times.

3.9.1 Creation of a subprogram

Create a subprogram in the following format:

O (:)××××;

– ;

– ;

⋮

– ;

M99;

Specify a subprogram number after "O" (EIA) or ":" (ISO) at its beginning. M99 is not necessarily set in a single block at the end of a subprogram.

Example:

X – – – – – – – – – – – – – – – – – – M99;

Refer to Sections 5.17 to 5.19 for how to save a subprogram in a register.

Note 1: The subprogram number in the preceding block can also be indicated as "N××××" rather than the (:) following O so that NC program may be compatible with other systems.

The system records the numerical value following N as a subprogram number.

3.9.2 Execution of a subprogram

A subprogram will be executed once it is called by a main program or other subprograms.

A subprogram may be called in the following format:

M98 P— — — — L— — — —:

Number of repeated subprogram calls

Subprogram number

A subprogram shall be only repeated once in case of omission of L.

Example:

M98 P1002 L5:

The No.1002 subprogram will be called repeatedly for five times when the instruction is executed.

The called subprogram instruction (M98 P—L—) and move instruction may be set in the same block.

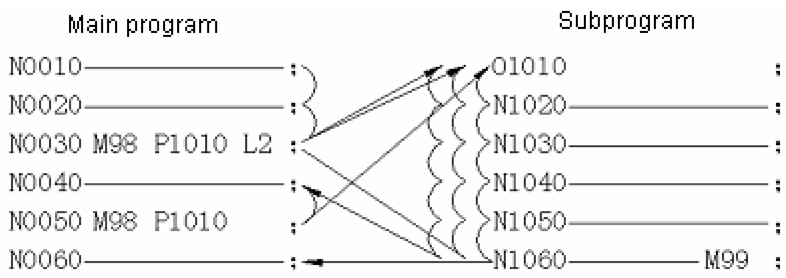
Example:

X1000 M98 P1200;

In the above example, No.1200 subprogram is called after the travel in the direction of axis X.

Example:

The executing sequence for a main program to call a subprogram is as follows:



The executive process for a subprogram to call another subprogram is similar to the above example.

Note 1: M98 and M99 signals are not sent to the machine.

Note 2: If the subprogram number specified by P is not found, No.78 alarm will be given.

Note 3: The M98 P××××: instruction input through MDI cannot call a subprogram. In this case, the following program shall be developed in the editing mode and executed with storage.

O××××

M98 P××××:

M02:

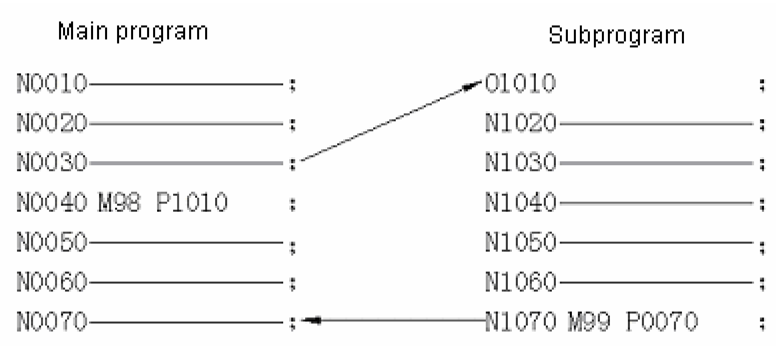
Note 4: The blocks or single block containing M98 P—; M99; are invalid. When the blocks of M98 and M99 contain the addresses other than O, N, L and P, however, the stop of a single block will be valid.

3.9.3 Special methods of application

The following special methods of application are available.

3.9.3.1 When a sequence number is specified in address P in the last block of a subprogram, control cannot return to the next block of the called subprogram in a main program, but to the block specified by the sequence number set by address P. Yet the main program is only valid in the storage operation mode.

The time required to return to the main program by this means is longer than that in normal conditions.



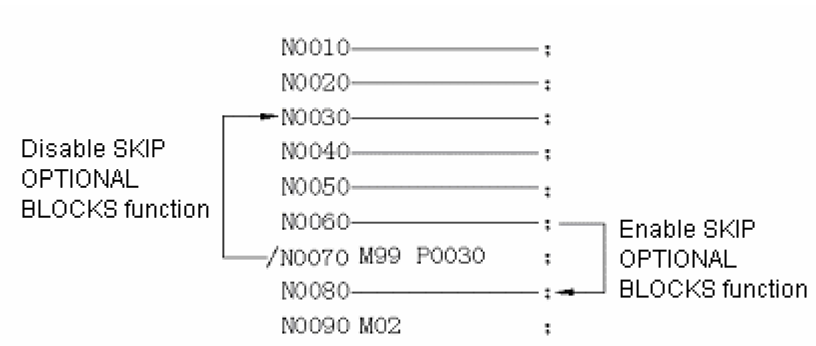
3.9.3.2

If M99 is executed in a main program, the control will return to its beginning.

If a "/M99;" block is inserted in the proper position in the main program, execution of M99 at this time will enable the control to return to the beginning of the main program and it will be executed again.

"/M99;" will be neglected and the control will go to the next block provided that SKIP OPTIONAL BLOCKS function is enabled.

If "/M99 Pn;" is inserted, the control cannot return to the head of the program but to the blocks of sequence number "n". The time required for switching to the blocks of sequence number n is longer than that for returning to the starting point of the program.

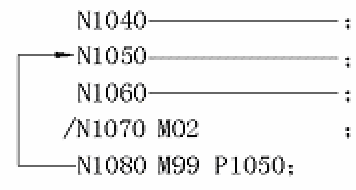


3.9.3.3

It may be executed in automatic mode as a main program from the beginning of MDI Search subprogram.

In this case, it will return to the beginning of the subprogram if a block containing M99 instruction is executed. When "M99 Pn;" is executed, it will switch to the block of the sequence number "n" and repeatedly execute it.

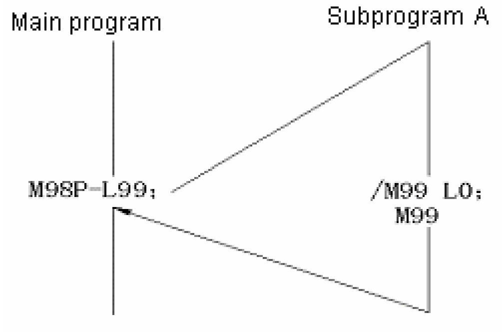
To stop execution in the above process, it is possible to insert "/M02;" or "/30;" in a proper position if you need to stop the execution. The execution of the above instruction may stop it and disconnect the switch when the SKIP OPTIONAL BLOCKS switch is set to ON.



3.9.3.4 M99 Lα

Number L of repeated recalls of a subprogram is changed to α in midway when the above instruction is executed.

If the SKIP OPTIONAL BLOCKS switch is set to OFF in the process, the number of repetitions will be changed to zero and the execution of the main program will go on when END OF SUBPROGRAM instruction (M99) is executed.

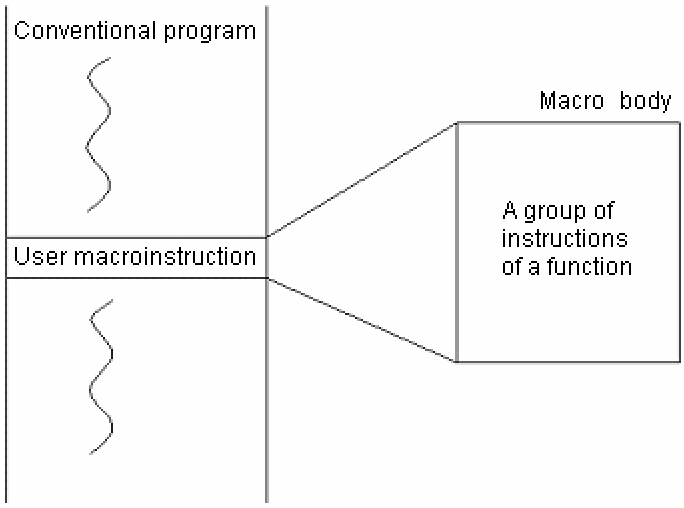


3.10 User Macro

3.10.1 General

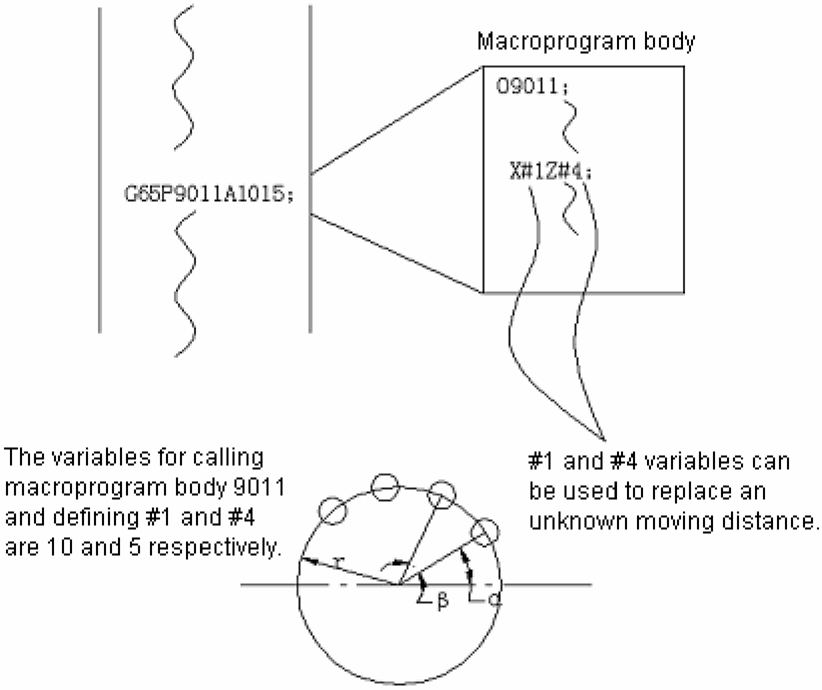
The functions of user macros A and B are basically equivalent. The differences between them are described in section 10.10 (9).

A function is composed of a group of instructions and saved in the storage as a subprogram. The saved function is symbolized by an instruction. Therefore, it is only necessary to specify its symbolic instruction for the function to be executed. The group of saved functions is called “macro body” and is symbolic instruction is called “user macroinstruction”. Macro body may also be called “macro” for short and user instruction called “macro call instruction”.



Programmer only needs to memorize macroinstructions in stead of the instructions in macro bodies.

Three points regarding macro: Variables may be used in macro bodies; operations can be performed on the basis of variables; and the variables that may be actually specified in user macroinstructions.



The above shows that annularly distributed screw holes are easy to machine.

Once a macro body for annular holes is programmed and recorded, NC may start machining in it has annular hole machining function.

Programmer may call the annular hole machining function through the following instruction.

G65 P_p_ R_r_ A_a_ B_b_ K_k_ :

p: Macro number of annular holes

r: Radius

a: Starting angle of holes

b: Angle between holes

k: Number of holes

In this way user may improve the performance of NC himself. Macro bodies may be supplied by the manufacturer of machine or developed by user.

3.10.2 Variables

In macros, variables are used to replace data of figures. User may define any of their values (in allowable range). The application of macros enables them to more general and flexible.

When using some variables, they are identified by signs.

3.10.2.1 Indication of a variable

As shown in Example 10.2.1, a variable consists of a # followed by a variable number.

#1 (i=1, 2, 3, 4……)

Example 10.2.1: #5

#109

#1005

The format below may also be usable, in which figure is replaced with a formula.

#〔<Formula>〕

Example 10.2.2: #〔#100〕

#〔#1001-1〕

#〔# b/2〕

The variable # i is replaced with #〔<formula>〕 hereinafter.

3.10.2.2 Introduction of variables

The values following addresses may be replaced by variables. If a program is <Address># 1 or < Address >-# 1, it means that a variable value or its complement serves as the instruction value of its address.

Example 10.2.3:

F #33 Assuming #33=1.5, it is similar to F1.5.。

Z-#18 Assuming #18=20.0, it is similar to Z-20.0.

G #130 Assuming #130=3.0, it is similar to G3.

(1) The addresses prohibited from using variable: 1, :, O and N, namely # 27 or N # 1 or equivalent expressions are not allowed.

The n folds ((n = 1 ～ 9) in SKIP OPTIONAL BLOCKS /n cannot be used as a variable.

(2) The method for replacing a variable number with a variable is as follows: When the “5” in # 5 is replaced with “# 30”, it shall not be written as “# #30” but “# (30)

(3) For all addresses, their variable value must not exceed their maximum instruction value. For instance, when #140=120, M #140 exceeds its maximum value (M code shall be less than 99).

(4) Numerical number shall not be used for identification purpose. When #30=02, for instance, F #30 shall be deemed as F2.

(5) – 0 and + 0 are not identifiable, namely when # 4 = - 0, X # 4 is deemed as X 0.

(6) When a variable is used for address data, the numerical value below significant bit shall be rounded (half-adjust)

(7) The numerical value following an address can also be replaced with <Formula>. If < Formula >[< Formula >] or < Formula > - [<Formula >] is used as a program, the value or complement of < Formula> shall be the instruction value of the address.

Caution: For a constant without a decimal point in parentheses, assume there is a decimal point at its end.

Example: 10.2.4:

X [#24+ #18*COS (#1)]

Z- (#18+ #26)

3.10.2.3 Undefined variables

Undefined variable value is called <Empty>. Variable #0 is used for the variable that is always empty.

An undefined variable has the following characteristics:

(1) Introduction of variable

The address itself will be neglected when an undefined variable is introduced.

When #1=<Empty>	When #1=0
G90 X100 Y#1 ↓ G90 X100	G90 X100 Y#1 G90 X100

(2) Operational formula

Except <Empty> is used for replacement, it is equivalent to variable O.

When #1=<Empty>	When #1=0
#2=#1 ↓ #2=<Empty>	#2=#1 ↓ #2=0
#2=#1*5 ↓ #2=0	#2=#1*5 ↓ #2=0
#2=#1+#1 ↓ #2=0	#2=#1+#1 ↓ #2=0

(3) Conditional expression

<Empty> and 0 differ in relation to E Q and N E.

When #1=<Empty>	When #1=0
#1EQ#0 ↓ Definite	#1EQ#0 ↓ Indefinite
#1 NEO ↓ Definite	#1 NEO ↓ Indefinite
#1GE#0 ↓ Definite	#1GE#0 ↓ Definite
#1GT0 ↓ Indefinite	#1GT0 ↓ Indefinite (unestablished)

3.10.2.4 Display and setting of variable

A variable may be displayed on LCD and set by MDI means. See Chapter IV, Section 5.8.2.

3.10.3 Types of variables

Variable may be classified into local variable, common variable and system variable depending on their variable numbers. And these variables have different use and characteristics.

3.10.3.1 Local variables # 1～# 33

A local variable is a variable that is locally used in a macro, namely the local variable # 1 for calling a macro at a point of time varies from the # 1 for calling a macro at another point of time (regardless the macros are the same). Therefore, the local variable for Macro A will never be misused for Macro B and damage its variable value as nesting when calling Macro B from A.

Local variable is used for independent variable conversion. See Section 10.7 for the relationship between variable and address. The local variable without independent variable conversion is empty in its original state and user may use them as he like.

3.10.3.2 Common variables #100～ #149 , #500～ #509

Local variable is common in a macro. Common variable is common for all subprograms called by main programs and all macros are common. That is, the # 1 for some macro and the # 1 for another. Therefore, the operation result of the common variable # 1 in a certain macro is usable in another macro.

In this system, no special requirement is made for the use of common variable. They may be used by user.

Common variables #100 to #149 can be cleared by powering off. However, the #500 to #509 common variables cannot be removed by powering off.

3.10.3.3 System variables (for user Macro B)

In this system, the use of system variables is definite.

(1) Interface signals #1000 to #1015 and #1032, #1100 to #1115 and #1132

[Input signal]

The state of interface input signal is determined by the readings of #1000 to #1032 system variables.

System variables	Interface input signals
#1000	2^0 UI0
#1001	2^1 UI1
#1002	2^2 UI2
#1003	2^3 UI3
#1004	2^4 UI4
#1005	2^5 UI5
#1006	2^6 UI6
#1007	2^7 UI7
#1008	2^8 UI8
#1009	2^9 UI9
#1010	2^{10} UI10
#1011	2^{11} UI11
#1012	2^{12} UI12
#1013	2^{13} UI13
#1014	2^{14} UI14
#1015	2^{15} UI15

Variable value	Input signal
1	Contact ON
0	Contact OFF

Since the reading of variable value 1.0 or 0.0, it is not necessary to take its unit into account. However, unit shall be considered in the development of a macro.

All input signals are read in by reading variable system #1032.

$$\#1032=\sum_{i=0}^{15}\#[1000+i]*2^i$$

In operational instructions, system variables #1000 to #1032 shall not be used as the item on the left side [output signal]

Interface output signals may be given by assigning a value to #1100 to1132 system variables.

System variables	Interface input signals

#1100	2^0 UO0
#1101	2^1 UO1
#1102	2^2 UO2
#1103	2^3 UO3
#1104	2^4 UO4
#1105	2^5 UO5
#1106	2^6 UO6
#1107	2^7 UO7
#1108	2^8 UO8
#1109	2^9 UO9
#1110	2^{10} UO10
#1111	2^{11} UO11
#1112	2^{12} UO12
#1113	2^{13} UO13
#1114	2^{14} UO14
#1115	2^{15} UO15

Variable value	**Output signal**
1	Contact ON
0	Contact OFF

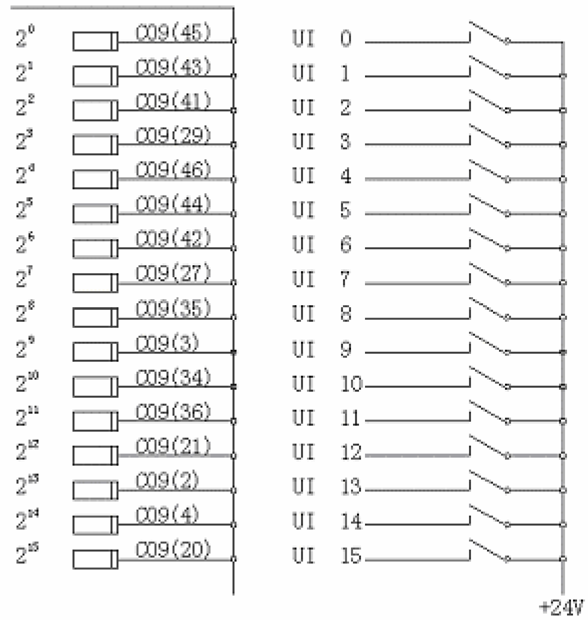
All output signals can be output at a time by assigning a value to system variable #1132.

$$\#1132=\sum_{i=0}^{15}\#[1100+i]*2^i$$

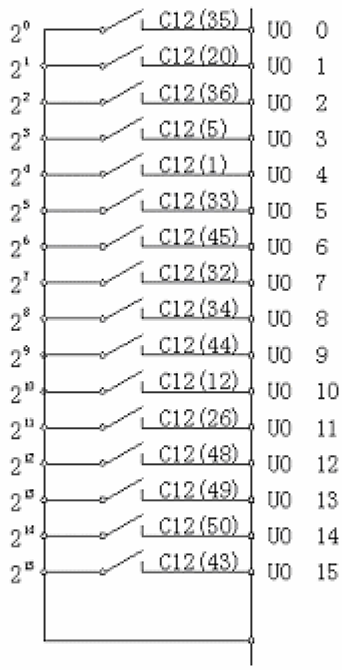
For #1100 to #1132 system variables, the given final numerical values are stored as 1.0 or 0.0.

Note: When the values of different 1.0 or 0.0 are assigned to #1100 to 1115, <Empty> is deemed as 0 and all conditions other than <Empty> and 0 are regarded as 1. But the numerical values less than 0.00000001 are undefined.

Note 1: The figure below indicates the connection for input signals of user macros.



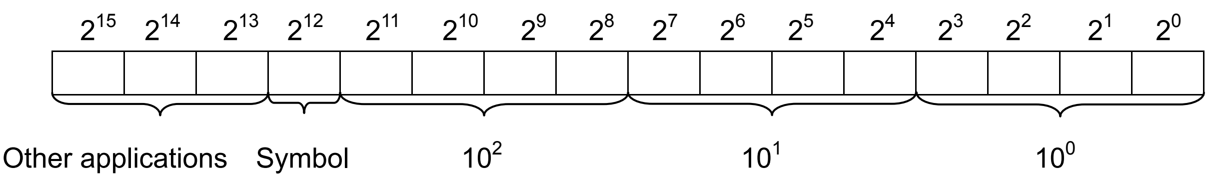
Note 2: The figure below indicates the connection for output signals of user macros.



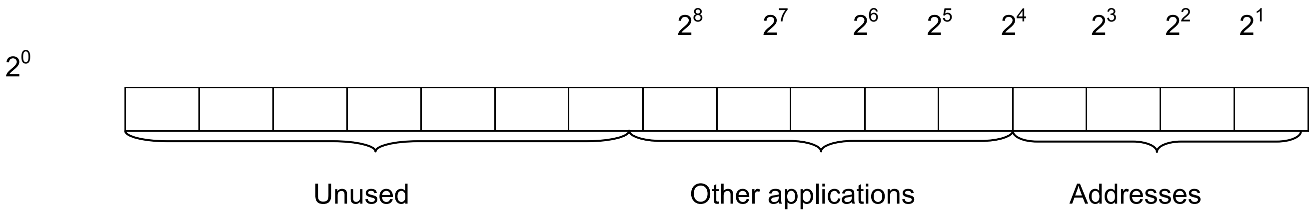
Example 10.3.1

① The 3-digit BCD data with symbol is read to #100 by changing address.

Composition of DI:



Composition of D0:



Macro call instruction:

G65 P9100D (address):

The macro body is required as follows:

09100:

#1132=#1132 AND496 OR #7: : To deliver the address

G65 P9101 T60: : Time macroinstruction

#100=BIN [#1032 AND 4095]; : To read 3-digit BCD data

IF [#1012 EQO] GOTO 9100; : With symbol

#100= - #100:

N9100 M99:

② Eight types of 6-digit BCD data (3 digits on the left side of decimal point + 3 digits on the right of decimal point) with symbol are read to #101 through address change.

When D0$2^0$=0, 3 digits on the right of decimal point

=1, 3 digits on the left side of decimal point

When D0$2^3$～2^1=000, No.1 data

=001, No.2 data

┆

=111, No.8 data

Macro call instruction:

G65 P9102 D (data number):

User macro body is required as follows:

09102:

G65 P9100D [#7*2];

#101=#101+#100/1000;

M99;

(2) Tool offsets #2000～#2200, workpiece offsets #2500～#2906。

Tool offsets use system variables #2001～#2200 while workpeice offsets #2500 and #2906.

The offsets are established by reading these variables and changed by assigning values to the system variable #1.

Tool offset No.	Tool offset
1	#2001
2	#2002
3	#2003
┆	┆
┆	┆
199	#2199
200	#2200

#2000 is readable, but its numerical value is always 0.

Axes	Workpiece offset numbers	Workpiece offsets
X	External workpiece offset G54 ┆ G59	#2500 #2501 ┆ #2506
Y	External workpiece offset G54 ┆ G59	#2600 #2601 ┆ #2606
Z	External workpiece offset G54 ┆ G59	#2700 #2701 ┆ #2706
The fourth	External workpiece offset G54 ┆ G59	#2800 #2801 ┆ #2806
The fifth	External workpiece offset G54 ┆ G59	#2900 #2901 ┆ #2906

Example 10, 3, 2　#30=#2005

Substitute the tool offset of the offset number for variable #30.

When the offset is 1.500mm (0.500inch), the value of #30 will change into 1.5 (0.15).

#2210=#8

Alter the offset of Offset number 10 so that it is equal to the value of variable #8.

(3) Alarm #3000

When error is found in a macro, an alarm will be given. If the alarm number is specified in system variable #3000, the warning lamp will be lit and NC unit will be in warning state.

#3000=n (ALARM　MESSAGE):

Select the alarm number not used in standard specifications and set it in the macro. (n＜200= A warning less than 26 characters may be specified between the note start code and note end code.

(4) Clocks #3001, #3002

It is possible to know the time of the clock by reading the values of system changes #3000 and #3002 for the clock. Time can also be preset by evaluation to the system variable.

Type	System variable	Unit of time	Power on	Counting condition
Clock 1	#3001	1ms	Reset to 0	Any time
Clock 2	#3002	1hr	The same as power off	When there is an STL signal

The accuracies of all clocks fall within 16ms. The accuracy beyond the range of Clock 1 will be reset to 0 at 6536ms. Clock 2 will continue to increase provided that it is not predetermined.

Time cannot be correctly determined in the event that it goes beyond the maximum value 9544hr.

Example 10.3.3: Timing

Macro call instruction

G65 P9101T (wait time)ms;

The macro may be set as follows:

09101;

#3001=0;　　　　　　　　: Initial setting

WHILE [#3001 LE #20] D01;　: Wait for the set time

END1;

M99;

(5) Disable SINGLE BLOCK STOP and wait for end signal of miscellaneous functions

When the following numerical value is assigned to the system variable #3003, SINGLE BLOCK STOP function will be disabled and the next block will be executed in advance without waiting for the end signal (FIN) of miscellaneous functions (M, S, T and B). Distribution end signal (DEN) will not be sent without waiting for end signal. Note that a miscellaneous function

without waiting for end signal will be set after that.

#3003	Single block stop	End signal of miscellaneous functions
0	Not disable	Wait
1	Disable	Wait
2	Not disable	Not wait
3	Disable	Not wait

Example 10.3.4: Drilling cycle (for increment programming) is equivalent to G81.

Macro call instruction

G65 P9081L (Number of repetitions) R (Point R) Z (Point Z):

The macro body is required as follows:

09081;

#3003=1;

G00 Z#18;

G01 Z#26;

G00Z-[ROUND (#18)+ROUND (#26)];

#3003=0;

M99;

Single block will not stop. #18 corresponds to R and #26 to Z.

(6) Set FEEDRATE, FEEDRATE OVERRIDE and ACCURATE STOP CHECK INVALID in #3004

If the following numerical values are assigned to system variable #3004, the FEEDRATE and FEEDRATE OVERRIDE of the blocks that follow will become invalid and accurate stop check will not be performed any more. Press the FEEDRATE key during the execution of FEEDRATE INVALID block.

① Pressing down and hold the key: FEEDRATE will be executed at the beginning of the first block of FEEDRATE VALID.

② Pressing and releasing the key: Now the FEEDRATE lamp illuminates. However, Feedrate is not performed as above, but started at the end point of the first block of

FEEDRATE VALID.

#3004	Feedrate	Feedrate override	Accurate stop check
0	○	○	○
1	×	○	○
2	○	×	○
3	×	×	○
4	○	○	×
5	×	○	×
6	○	×	×
7	×	×	×

○: Valid　　　　×: Invalid

Example 10.3.5: Tapping cycle (for increment programming) (equivalent to G84)

Macro call instruction

G65 P9084 L (Number of repetitions)R (Point R) Z (Point Z):

The macro body is developed as follows:

09084;

#3003=1;　　　　: Disable SINGLE BLOCK STOP

G00Z#18;

#3004=7;

G01Z#26;

M05;　　FEEDRATE, FEEDRATE OVERRIDE and ACCURATE STOP CHECK

M04;

Z-#26;

#3004=0;

M05;

M03;

G00Z-#18;

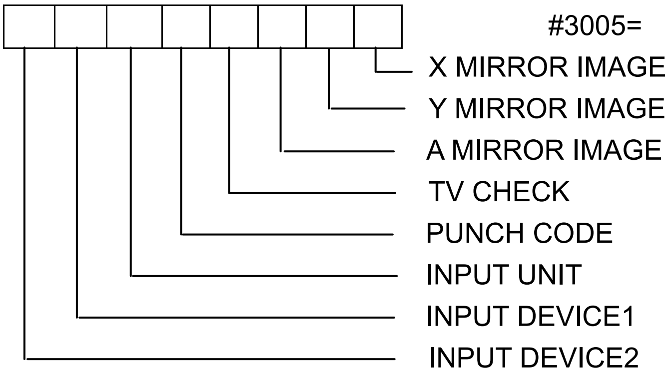
#3003=0;

M99;

Note: “M05” may be omitted for some machines.

(7) Variable corresponding to the setting data #3005

Setting data is established by assigning a value to the system variable #3005. Its digits correspond to all setting data when the system variable #3005 is expressed as a binary number.



Example: If #3005=55: instruction is executed, the setting data is established as follows:

X MIRROR IMAGE (X mirror image)=1

Y MIRROR IMAGE (Y mirror image)=1

A MIRROR IMAGE (A mirror image)=1

TV CHECK (vertical check)=0

PUNCH CODE (code standard)=1

INPUT UNIT (input unit)=1

INPUT DEVICE1 (input device 1)=0

INPUT DEVICE2 (input device 2)=0

(8) Modal messages #4001～#4120

The present value instructed by a modal message may be determined by readable system variables #4001 to #4120. Its unit is the unit during instruction

System variables	Modal messages
#4001	G codes (Group 01)
⋮	⋮
#40021	G codes (Group 21)
#4102	B codes
#4107	D codes
#4109	F codes
#4111	H codes
#4113	M codes
#4114	Sequence numbers
#4115	Program numbers
#4119	S codes
#4120	T codes

Example 10.3.6: Boring cycle (equivalent to G86) during incremental value/absolute value mixed programming

Macro call instruction

G65 P9086L (Number of repetitions) R (Point R) Z (Point Z):

The macro is developed as follows:

```
09086;
#1=#4003;      : Save Group 03 G codes
#3003=1;       : Disable SINGLE BLOCK STOP
G00 G91 Z#18;
G01  Z#26;
M05;
G00 Z-[#18+#26];
M03;
#3003=0;
G#1 M99;       : Restore Group 03 G codes
```

System variables #4001 to #4120 cannot be used for the left item on the left side of the operational instruction.

(9) Position messages #5001～#5105

Position messages may be determined by readable system variables #5001 to 5015. Whether its unit is mm or inch depends on the input system.

System variables #5001# to #5105 cannot be used for the left item on the left side of the operational instruction

System variables	Position messages	Reading in during travel
#5001 #5002 #5003 #5004 #5005	End position of Axis X block (ABSIO) End position of Axis Y block (ABSIO) End position of Axis Z block (ABSIO) End position of the 4th axis block (ABSIO) End position of the 5th axis block (ABSIO)	Applicable
#5021 #5022 #5023 #5024 #5025	Actual position of Axis X (ABSMT) Actual position of Axis Y (ABSMT) Actual position of Axis Z (ABSMT) Actual position of the 4th axis (ABSMT) Actual position of the 5th axis (ABSMT)	Not applicable
#5041 #5042 #5043	Actual position of Axis X (ABSMT) Actual position of Axis Y (ABSMT) Actual position of Axis Z (ABSMT)	Not applicable

#5044 #5045	Actual position of the 4th axis (ABSMT) Actual position of the 5th axis (ABSMT)	
#5061 #5062 #5063 #5064 #5065	Axis X SKIP position (ABSKP) Axis Y SKIP position (ABSKP) Axis Z SKIP position (ABSKP) The 4th axis X SKIP position (ABSKP) The 5th axis X SKIP position (ABSKP)	Applicable
#5083	Tool length offset	Not applicable
#5101 #5102 #5103 #5104 #5105	Axis X servo position deviation Axis Y servo position deviation Axis Z servo position deviation The 4th servo position deviation The 5th servo position deviation	Not applicable

Abbreviation	ABSIO	ABSMT	ABSOT	ABSKP
Meaning	The end position of the previous block	The actual position of an instruction (identical with indication POS, MACHINE)	The actual position of an instruction (identical with indication POS, MACHINE)	The position of the ON SKIP signal in G31 block
Coordinate system	Workpiece coordinate system	Workpiece coordinate system	Workpiece coordinate system	Workpiece coordinate system
Tool position Tool length	N/A	Applicable	Applicable	Applicable
Tool compensation	Tool position	Tool calibrating spot	Tool calibrating spot	Tool calibrating spot

Note: The tool length offset is not valid just between the blocks to be executed, but in the executing blocks. If SKIP signal is not switched on in G31 block, its position is at the end point of the block.

Example 10.3.7:

The tool moves to a fixed point (The distances from XP, YP and ZP to the reference point) and returns to the original position after treatment by programming an intermediate point.

Macro call instruction

G65 P9300X (Intermediate point) Y (Intermediate point) Z (Intermediate point):

The macro body is developed as follows:

0 9300;

#1=#5001;

#2=#5002;

#3=#5003;

G00 Z#26;

X#24 Y#25;

G04; Stop travel because of reading #5021～#5023.

G91 X[XP-#5021]Y[yp-#5022]Z[ZP-#5023]:

┆ (Treatment)

X#24 Y#25 Z#26;

X#1 Y#2;

Z#3;

M99;

(10) The setting and display of variable names

The names consisting of up to 8 characters may be assigned to variables #500 to #511 through the following instructions.

SETVNn[$\alpha_1\alpha_2$…………α_8, β_1, β_2, …………β_8…………]:

n is a starting number of the variable numbers with names.

α_1, α_2…………α_8 are the names of variable number n.

β_1, β_2, …………β_8 are the names of variable number n+1; the same applies below.

All character strings are separated by “, ”. All characters except NOTE END, NOTE START, [,], EOB, EOR, : (colon for program number) can be used for valid messages. Variable names will not be cleared after power off.

LCD displays in the sequence of NO., NAME and DATA.

Note: Since the function is not available for some devices, #510 and #511 may not be used.

Example 10.3.8

SETVN 500[ABCDEFGH, COUNTER, POINTER];

MACRO	VAL： 06：	01234 N3456
NO。	NAME	DATA
0500	ABCDEFGH	-1234。5678
0501	COUNTER	00020。000
0502	POINTER	
0503	1st	-0000。4025
0504	2nd	-00004。500
0505		124000。00]
0506		
0507		
0508	START	
0509		
0510	TOOL-PT	000045。00
0511		
P		LSK

3.10.4 Operation instructions

All types of operations may be carried out between variables. Operation instruction is equivalent to general arithmetically developed program.

#i=＜Formula＞

The right ＜Formula＞ of an operation instruction is the combination of constant, variable, function and operator. Constant replaces #j and #k. The constant with a decimal point in ＜Formula＞ may be deemed that there is a decimal point at its end.

3.10.4.1 Definition and substitution of variable

#i=#j Definition and substitution

3.10.4.2 Additive operation

#i=#j+#k Summation

#i=#j-#k Subtraction

#i=#joR#k Logical sum (for each one of 32 digits)

#i=#jXOR#K Exclusive or (for each one of 32 digits)

3.10.4.3 Multiply operation (Macro B option)

#i=#j*#k Arithmetic product

#i=#j/#k Quotient

#i=# JAND#K Logical multiply (for each one of 32 digits)

3.10.4.4 Function (Macro B option)

#i=SIN[#j] Sine (unit: degree)

#i=COS[#j] Cosine (unit: degree)

#i=TAN[#j] Tangent (unit: degree)

#i=ATAN[#j]/ [#k] Arc tangent (unit: degree)

#i=SQRT[#j] Square root (unit: degree)

#i=ABS[#j] Absolute value

#i=BIN[#j] Switching from BCD to BIN

#i=BCD[#j] Switching from BIN to BCD

#i=ROUND[#j] Rounding

#i=FIX[#j] Rounding off the part below decimal point

#i=FUR[#j] Rounding the decimal part to integer part

Note: How to use ROUND function

(1) If function ROUND is used in an operation instruction or in IF or WHILE conditions, the original data with a decimal point shall be rounded off.

Example: #1=ROUND [1.2345];

#1 changes to 1.0.

IF [#1 LEROUND (#2)]GOTO 10:

When #2=3.567, ROUND[#2]=4.0

(2) When the function ROUND is used in an address instruction, it shall be rounded off in its minimum setting unit.

Example: G01 X[ROUND (#1)];

If #1 is 1.4567 and the minimum input increment of X is 0.001, the block will change to G01 X1.457;

This instruction is equivalent to G01 X#1: in the instruction.

The function ROUND in an address instruction mainly applies to the following conditions.

Example: [It is only moved by #1 and #2 in increment and then return to the initial point.]

N1 #1=1.2345;

N2 #2=2.3456;

N3 G01 X#1 F100; : X moves by 1.235

N4 X#2; : X moves by 2.346

N5 X-[#1+#2]; : X moves (because #1+#2=3.5801)

Since 1.235+2.346=3.581, the program cannot return to the starting point through N5.

With N5X-[ROUND[#1]+ROUND[#2]];

This is equivalent to N5x-1.235+2.346] and the program may return to the starting point.

3.10.4.5 Hybrid operation

The above operations and functions may be combined. The precedence order of operation is function, multiply operation and then additive operation.

Example 10.4.1 #i #j+#K*SIN [#ℓ]

——①——Operational order

——②——

——③——

3.10.4.6 Changing operational order using [ ]

The part that is to be preferred may be put in [ ]. [ ] may be nested for 5 times (including the parenthesis in a function).

Example 10.4.2:

#i=SIN [[[#j+#K] *#ℓ +#m] *n] (triple nesting)

①

②

③

④

⑤

3.10.4.7 Accuracy

To arrange an order with macro order, make sure it has an adequate accuracy.

(1) Data format

The floating point format of the data processed by macro is as follows:

M*2E

Where: M: 1 sign bit +31 binary numbers

E: 1 sign bit +7 binary numbers

(2) Operational precision

Executing an operation once generates the following error. These errors are totalized in repeated operations.

Operational format	**Average error**	**Maximum error**	**Error type**
a=b*c	1.55×10^{-10}	4.66×10^{-10}	Relative error
a=b/c	4.66×10^{-10}	1.86×10^{-9}	$\left\lvert \frac{\varepsilon}{a} \right\rvert$
a= $\sqrt{b}$	1.24×10^{-9}	3.73×10^{-9}	
a=b+c a=b-c	2.33×10^{-10}	5.32×10^{-10}	min $(\frac{\varepsilon}{b}, \frac{\varepsilon}{c})$
a=SiNb a=comb	5.0×10^{-9}	1.0×10^{-9}	Absolute error $\lvert\varepsilon\rvert$ degree
a=ATANb/c	1.8×10^{-6}	3.6×10^{-6}	

Note: Function TAN executes SIN/COS.

3.10.4.8 Cautions regarding deterioration of precision

(1) Add-subtract operation

Note that when absolute values operate subtraction in addition or subtraction, the relative error will not be maintained below 10^{-8}. For example, assuming the actual values of #1and #2 are as follows:

#1=9876543210123.456

#2=9876543277777.777

Execute the operation of #2-#1:

#2-#1=67654.321

The above numerical value cannot be achieved. Since a macro has only a decimal 8-bit precision, the precision of #1and #2 numerical values is degraded and approximates:

#1=9876543200000.000

#2=9876543300000.000

Strictly speaking, the above value differs from internal value because they are binary.

#2-#1=100000.000

Causing a bigger error.

(2) Logic operation

EQ, NE, GT, LT, GE and LE are basically equivalent to add-subtract operation. Hence note the error and make sure that the #1 and #2 are equal in the above example. For instance, IF[#IEQ#2] always fails to make correct judgment.

If IF[ABS[#1-#2]LT50000] is used to make error judgment, #1and #2 are considered equal when the difference between #1and #2 falls within its error range.

(3) Trigonometric function

Absolute may be well guaranteed in trigonometric functions. Since they are not below 10^{-8}, note the conditions of multiplication and division after an operation of trigonometric function.

3.10.5 Control instruction

The control of a program is achieved by using the following instructions.

3.10.5.1 Branch (GOTO)

IF [＜Conditions＞]＝GOTOn

If ＜Conditions＞ is satisfied, the next operation will go to the block with a sequence number n in the program. The sequence number n may be substituted by a variable or [＜Formula＞].

If the condition is not satisfied, it will continue to execute the next block.

IF[＜Conditions＞= can also be omitted, and the program is unconditionally forwarded to block n in this case.

＜Conditions＞ is of the following types:

#JEQ#K=

#JNE#K≠

#JGT#K＞

#JLT#K＜

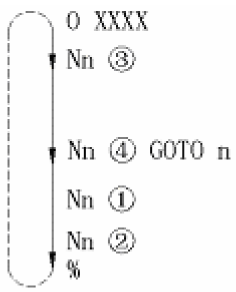
#JGE#K≥

#JLE#K≤

#j and #k may also be substituted with ＜Formula＞ and n with a variable or＜Formula＞.

Note 1: The blocks with a sequence number following n are executed after GOTOn and block shall be preceded by the sequence number n.

Note 2: When executing GOTON, the more distant Nn block is in the same direction, the longer executing time will be.



In the above figure, the executing time increases in the order of ①②③④. The GOTOn that has been executed for more times only keeps a short distance from Nn block. When the contents of a variable are used for measurement as well as warning, it is recommended not to use the warning program to approach GOTOn statement. The warning program departs from GOTOn.

Example 10.5.1: When #1≥10, No.150 alarm will be given.

┆

IF [#1GE10]GOTO150;

The conditions in which no alarm will be given:

M99;

┆

N150 #3000=150;

M99;

(If NEOP (NO.306) is set by parameter, the program will be saved in storage and M99 will not be used as the end of program.)

Note3: During the execution of GOTO, alarm may be given in the following events:

① When a macro operation cannot be properly executed in an address。

If GOTO is executed when #1=-1,

No. 119 alarm will be given in the block X[SQRT[#1]];.

② When the conditions specified by WHILE cannot be correctly executed

If GOTO is executed when #1=0,

No. 112 alarm will be given in the block WHILE [10/#1 GE2]D0 1.

In this case, change the following program:

① #2=SQRT[#1];

x#2;

② #2=10/#1

WHILE[#2 GE 2]D0 1;

┆

#2=10/#1;

END　1;

Operation instruction does not give an alarm even GOTO is executed.

3.10.5.2 Repeat (how to select Macro B)

WHILE[＜Conditions＞］＝DOm (m=1,2,3)

ENDm

When ＜Conditions＞ is satisfied, the block from DOm to ENDm will be repeatedly executed. That is, when the conditional expression ＜Conditions＞ for judging DOm block is satisfied, the program will go to the next block. When the conditional expression is not satisfied, the blocks following ENDm will be executed.

When WHILE[＜Conditions＞＝is equivalent to IF, it may also be omitted. If it is omitted, the program will repeatedly executed from DOom to ENDm.。

WHILE[＜Conditions＞＝DOm and ENDm shall be used in pair and identification number m be used for mutual identification.

Example 10.5.2

#120=1;

N1 WHILE [#120LE10] DO1;

N2 WHILE [#30EQ1] DO2;　　Repeat for 10 times

N3 END 2;　　Repeat when #30=1

```
#120=#120+1;
N4  END1;
```

Note 1: REPEAT (pay attention to the following points during REPEAT programming)

① DOm shall always be set before ENDm.

```
    ┆
END1;
    ┆ (N/A)
DO1;
    ┆
```

② In the same program, DOm shall correspond to ENDm one by one.

```
    ┆
DO1;
    ┆
DO1; (N/A)
    ┆
END1;
    ┆
DO1;
    ┆
END1 (N/A)
    ┆
END1;
    ┆
```

③ The same identification number may be used for many times.

```
    ┆
DO1;
    ┆
END1;
    ┆   (Applicable)
DO1;
    ┆
```

```
END1;
  ┆
```

④ DO statement may be nested for 3 times.

```
  ┆
DO1;
  ┆
DO2;
  ┆
DO3;
  ┆
END3;
  ┆
END2;
  ┆
END1;
  ┆
```

⑤ DO area must not be crossed.

```
  ┆
DO1;
  ┆
DO2;
  ┆
END1;
  ┆
END2;
  ┆
```

⑥ It is possible to transfer from the inside to the outside of DO area.

```
  ┆
DO1;
  ┆
GOTO 9000;
```

```
  ┆      (Applicable)
END1;

  ┆

N9000….;

  ┆
```

⑦ Transferring from the outside to the inside of the DO area is not allowed.

```
  ┆
GOTO 9000;

  ┆
DO1;     (N/A)
  ┆
N9000…..;

  ┆

END1;

  ┆

DO1;

  ┆

N9000…..;

  ┆    (N/A)

END1;

  ┆

GOTO 9000;
```

⑧ It is possible to call a macro and subprograms from the inside of the DO area. DO statement may be nested for three times in a macro body or a subprogram.

```
  ┆

DO1;

  ┆

G65…..; (Applicable)

  ┆

G66…..; (Applicable)

  ┆

G67     ; (Applicable)

  ┆

END 1;
```

┆;

DO1 ;

┆

M98….. ; (Applicable)

┆

END1;

┆

Note 2: As a rule, the time required for TRANSFER is shorter than that for REPEAT.

Example 10.5.3: Wait for the cyclic program whose signal (#10000) is 1

┆

N 10 I F[#1000 EQ 0]GOTO 10;

┆

If

┆

WHILE [#1000 EQ 0] DO 1

END1

Is used for programming, the executing time will be shorter.

3.10.6 Creation and storage of user macro body

3.10.6.1 Creation of user macro body

Macro and subprogram have the same format.

$\overline{O}$ (Program number):

Instruction

M99;

Program number is prescribed as follows:

(1) $\overline{O}$ 1～ $\overline{O}$ 7999

The numbers are applicable for the programs that can be freely saved, cleared and edited.

(2) $\overline{O}$ 8000～ $\overline{O}$ 8999

The numbers cannot be used for the programs for saving and clearing way edition without the setting of relevant devices.

(3) $\overline{O}$ 9000～ $\overline{O}$ 9019

The numbers are applicable for calling type special macros.

(4) $\overline{O}$ 9020～ $\overline{O}$ 9899

The numbers cannot be used for the programs for saving and clearing way edition without the setting of the parameters.

(5) $\overline{O}$ 9900～ $\overline{O}$ 9999

The numbers are applicable for robot operational programs.

Fictitious variable (the variable that macro calls in instruction to receive data) are fixed, namely the addresses of the specified parameters correspond to macro body number through macro call instruction.

Example 10.6.1:

O 9081;

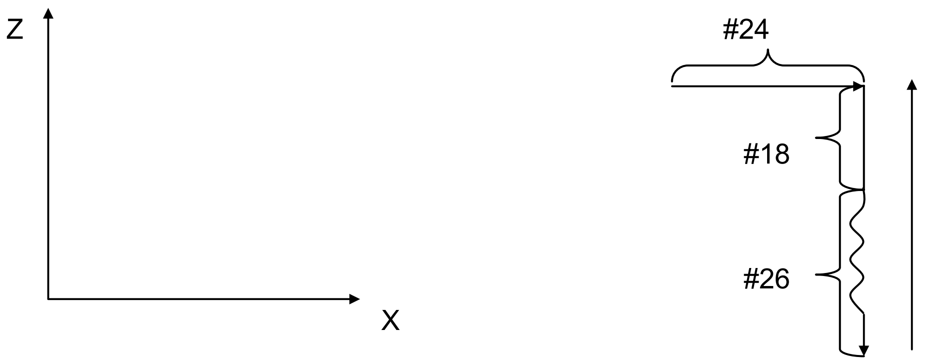
G 00 X#24;

Z#18;

G 01 Z#26;

G00 Z-[ROUND[#18]＋ROUND[#26]];

M99;



3.10.6.2 Storage of user macro body

A macro is a subprogram and is stored and edited in the same way as a subprogram. Its storage capacity is set with that is combined with a subprogram.

3.10.6.3 Macro statement and NC statement

The following block is called macro statement.

(i) Operation instruction (including =block)

(ii) Control instruction (including block GOTO, DO or END)

(iii) Macro call instruction (including the blocks of the macros called by G65, G66, G67 and G codes)

The block other than macro statement is called NC statement.

Macro state differs from NC statement in the following aspects:

(i) Single block does not stop in general single block mode.

(ii) Tool radius compensation C does not serve as “block without travel”.

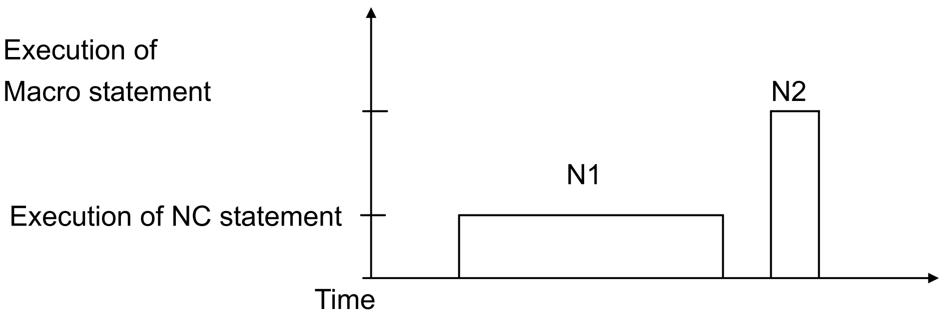
(iii) The executing time may vary depending on different statements.

(a) If a macro statement are provided after a block without buffer (the M codes without buffer and the blocks of G31), the statement will be executed after the block.

Example 3.1:

N1 X1000 M00;　Executing block

N2 #1100=1;　Macro statement



(b) When a block with buffer is followed by macro statements,

(i) When tool radius compensation C is not used,

The current block will be executed, and at the same time, the next macro statement will be executed until the next NC statement.

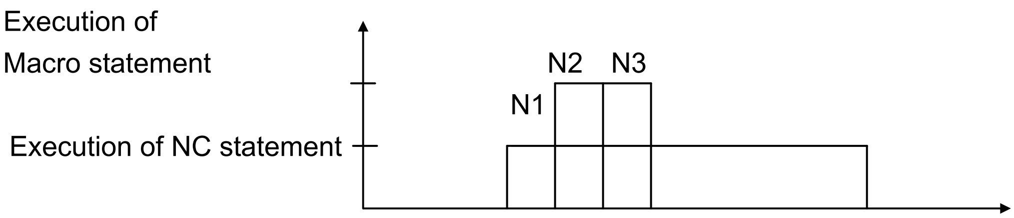
Example 3.2

N 1 G01 X1000; Executing block

N2 #1100＝1; Executed macro statement

N3　#1=10; Executed macro statement

N4　X2000; Next NC statement



(ii) Tool radius compensation C mode

(2－1) The first NC statement following the currently executing block is not a block without travel (no block without move instruction in the tool radius compensation plane).

(2－1－1) The 2nd NC statement is also not a block without travel.

The statements after the first NC statement following the executing block are executed.

Example 3.3:

N1 X1000; Executing block

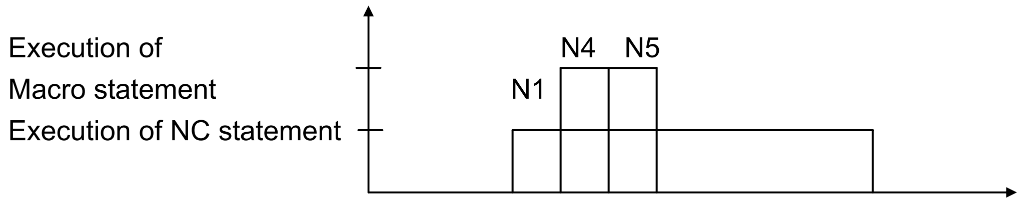
N2 #10＝100; Executed macro statement

N3 Y1000; The first NC statement

N4 #11001; Executed macro statement

N5 #1＝10; Executed macro statement

N6 X－1000; The 2nd NC statement



(2－1－2) The 2nd NC statement after the executing block is a “block without travel”.

The statements after the 2nd NC statement (i.e. block without travel) following the executing block are executed.

Example 3.4

N1 X 1000; Executing block

N2 #10＝100; Executed macro statement

N3 Y 1000; The 1st NC statement

N4 #1100＝1; Executed macro statement

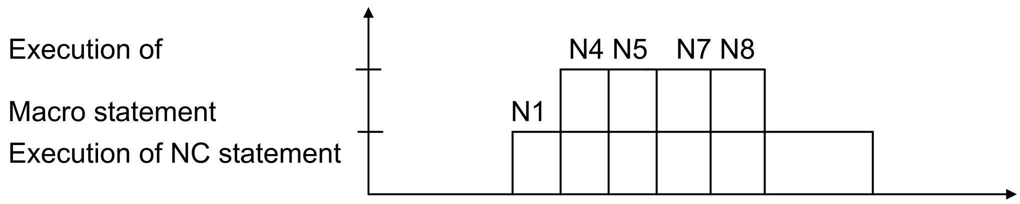
N5 #1＝10; Executed macro statement

N6 Z 1000; The 2nd NC statement

N7 #1101＝1; Executed macro statement

N8 #2＝20; Executed macro statement

N9 X－1000: The 3rd NC statement



(2－2) The first NC statement following the currently executing block is a block without travel. The macro statement cannot be executed.

Example 3.5

N1 Y1000; Executing block

N2 1100＝1; Executed macro statement

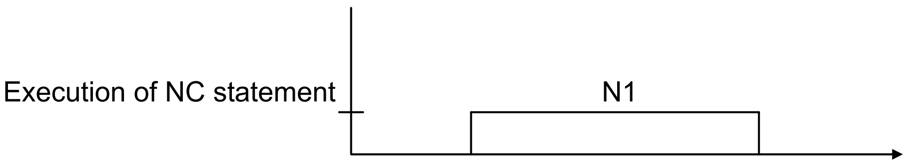
N3 #1＝10; Executed macro statement

N4 # Z 1000; The 1st NC statement (the block without travel)

N5 #1101＝1; Executed macro statement

N6 #2＝20; Executed macro statement

N7 X－1000; The 2nd NC statement



3.10.7 Macro call instruction

Macro may be easily called from a single block or modally from each block by means of calling.

3.10.7.1 Simple calling

The macro body specified by P (programmer) is called during the execution of the following instruction.

G 65 P (program number)L (Number of repetitions) <independent variable assignment>:

When an independent variable is required to change to macro, it is set by < independent variable assignment >. The following two types of <independent variable assignment> may be set. The independent variable herein is the actual numerical value assigned to a variable.

(Note) G65 shall be always specified before independent variable in a block. Minus sigma and decimal are usable and independent of the address in <independent variable assignment>.

(1) Independent variable assignment I

A__ B__ C__ D__ _ _ _ Z__

An independent variable other than G, L, N, O and P may be assigned to other addresses. An address needs not to be assigned alphabetically but specified in the format of words. The addresses that do not need to be specified may be neglected.

The use of I, J and K shall always be assigned alphabetically.

B__A__D__ _ _ _I__ K__ _ _ _True

B__A__D__ _ _ _J__ I__ _ _ _ False

The coincidence relation between the addresses assigned to variable assignment I and the variable number in macro body is as follows:

Addresses of independent variable assignment I	**The variables in macro body**
A	#1
B	#2
C	#3
D	#7
E	#8
F	#9
H	#11
I	#4
J	#5
K	#6
M	#13
Q	#17
R	#18
S	#19
T	#20
U	#21
V	#22
W	#23
X	#24
Y	#25
Z	#26

(2) Independent variable assignment II

A__ B__ C__ I__J__K__ I__J__K__ _ _ _ _

Besides that independent variable may be assigned to addresses A, B and C, up to ten groups of independent variables can be specified with addresses I, J and K. When several numerical values need to be assigned to the same address, they shall be assigned in the specified sequence. Unnecessary addresses may be omitted.

The addresses allocated as per independent variable assignment II correspond to the variable numbers of macro as follows:

Addresses of independent assignment I	Variables in macro
A	#1
B	#2
C	#3
I_1	#4
J_1	#5
K_1	#6
I_2	#7
J_2	#8
K_2	#9
I_3	#10
J_3	#11
K_3	#12
I_4	#13
J_4	#14
K_4	#15
I_5	#16
J_5	#17
K_5	#18
I_6	#19
J_6	#20
K_6	#21
I_7	#22
J_7	#23
K_7	#24
I_8	#25
J_8	#26
K_8	#27
I_9	#28
J_9	#29
K_9	#30
I_{10}	#31
J_{10}	#32
K_{10}	#33

The subscripts 1 to 10 of I, J and K indicate the order of the assigned group.

(3) Concurrently existing independent variable assignments I and II

Alarm will not be given even the independent variables of assignments I and II are within the same block with G65 instruction.

If Type I and II independent variables correspond to the same variable, the latterly specified independent variable will be valid.

G 65 A1.0 B2.0 I－3.0 I4.0 D5.0 P1000:

<Variable>

#1: 1.0

#2: 2.0

#3:

#4: －3.0

#5:

#6:

#7 5.0

In the example, the D5.0 that follows is valid though independent variables I 4.0 and D5.0 are set to variable #7.

Example 10.7.1: Reference point setting

Before instructing hole-group machining, the reference point of the hole group shall be set.

X_0 The X coordinate of hole-group reference point

Y_0 The Y coordinate of hole-group reference point

Macro call instruction:

G 65 P9200 X_x Y_y;

Shall use the following variables:

#100: hole counting

#101: the X coordinate of the reference point for hole-group macro

#102: the Y coordinate of the reference point for hole-group macro

#24: the X coordinate of the reference point is assigned with a macro call instruction

#25: the Y coordinate of the reference point is assigned with a macro call instruction

Macro body is developed as follows:

09200;

#101＝#24; Notify the reference point to hole-group macro

#102＝#25;

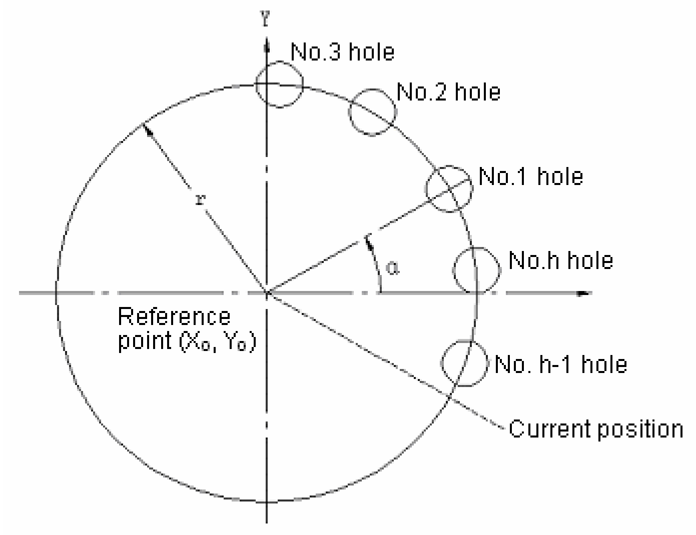
#100＝0; : Reset of hole counting

M99;

Example 10.7.2 Bolt-hole ring

The reference point set with the reference-point setting macro is used as the center of the circular ring. The h holes to be machined distribute on the circular ring at equal spacing. The

1st hole is located on the straight line of “a” angle (see the figure below).



The coordinates of the ring reference point for XO and YO bolt-holes

R: radius

A: starting angle

H: number of holes

Macro call instruction:

G65　P9207　Rr　Aa　Hh;

When h＜0, however, workpiece will be machined clockwise in –h counting.

The following variables shall be used.

#100　hole counting

#101　the X coordinate of reference point

#102　the Y coordinate of reference point

#18　radius r

#1　starting angle a

#11　number of holes h

#30　the storage of the X coordinate of reference point

#31　the storage of the Y coordinate of reference point

#32　Counting shows that the No.1 hole is being machined.

#33　the angle of No.1 hole

Macro body is developed as follows (for absolute programming)

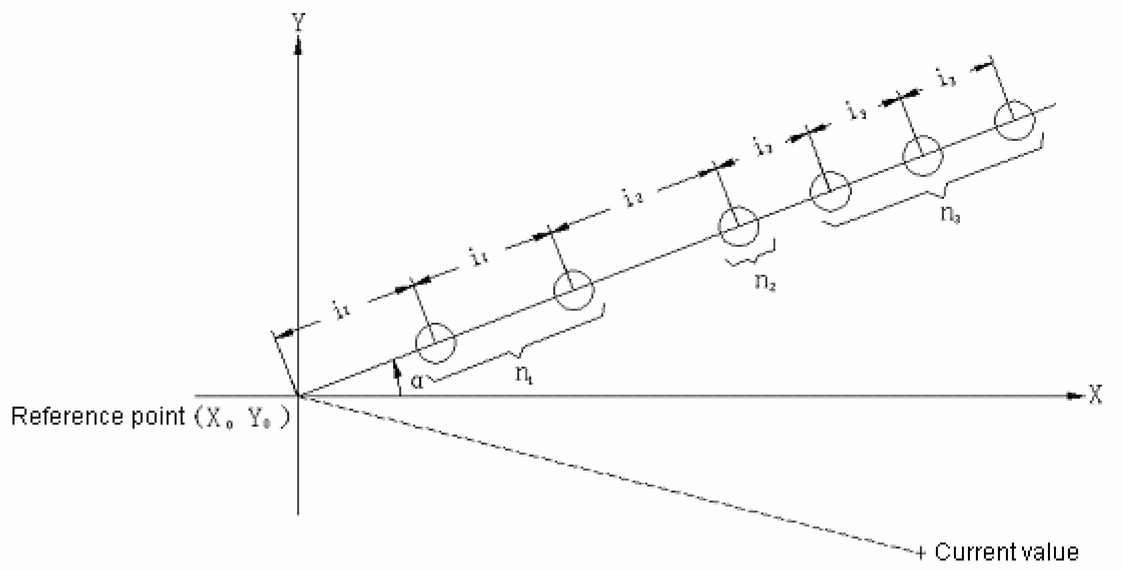
09207;

#30=#101;　　: Storage of reference point

```
#31=#102;
#32=1;
WHILE[#32LEABS[#11]]DO1;       : Repeat depending on number of holes
#33=#1+360*[#32-1]/#11;
#101=#30+#18*COS [#33];        : Hole position
#102=#31+#18*SIN [#33];
X#101   Y#102;
#100=#100+1;                   : Hole count plus 1
#32=#32+1;
END1;
#101=#30;                      ; Return to reference point
#102=#31;
M99;
```

Example 10.7.3: Unequally spaced oblique line

The point established by the reference point setting macro is used as the reference point. The holes to be machined are arranged by different spacing (1, 12……) in the direction forming an a angle with axis X



Coordinates of X0, Y0

A Angle

I Spacing between holes

K Number of holes set by equal spacing (to be assigned with decimal point)

Macro call instruction

G65 P9203 Aa, I1, Kn1, I12, Kn2………..;

When n=1, Kn may be omitted.
Using the following variables:

#100 : hole counter

#101 : X coordinate of reference point

#102 : Y coordinate of reference point

#1 : angle a

#4 : No.1 spacing 1_1

#6 : number of holes n_1 of the 1^{st} spacing group 1_1

#7 : No. 2 spacing 1_2

#9 : number of holes n_2 of the 2^{nd} spacing group 1_2

⁞

#2 : the X coordinate of storage reference point

#3 : the Y coordinate of storage reference point

#5 : Take out hole spacing I_1 count

#8 : the distance from reference point to the current hole

The macro is developed as follows (for absolute programming)

09203;

#2=#101; : Storage reference point

#2=#102;

#5=4;

#8=0;

WHILE[#5 LE 31]D01; : Hole spacing assignment I is limited to 10.

IF[#[#5]EQ#0]GOTO 9001; : If the assignment I is＜＞, it will ends.

D02;

#8=#8+#[#5];

#101=#2+#8*COS [#1]; : Hole position

#102=#3+#8*SIN [#1];

X#101 Y#102;

#100=#100+1; : Hole count plus 1

#[#5+2]=#[#5+2]-1;

IF[#[#5+2]LEO]GOTO 9002; : Repeat for k times

END2;

N9002 #5=#5+3; : Move to the next assignment I

END1;

N9001 #101=#2; : Return to the reference point

#102=#3;

M99;

3.10.7.2 Modal calling

When executing the following instruction, the macro calling mode may be instructed. During the execution of macro calling mode, calling the specified macro each time executes a move instruction.

G66P (<u>Program number</u>)L (<u>Number of repetitions</u>)＜Independent variable assignment＞:

＜Independent variable assignment＞ is equivalent to the condition of simple calling.

Macro calling mode will be cleared when the following instruction is executed.

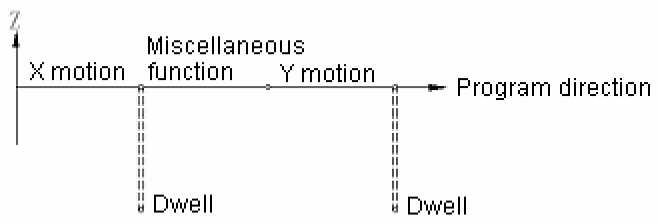
G67;

(Note): In G66 block, G66 shall be instructed before all independent variables.

The addresses for＜Independent variable assignment＞ may use minus sign and decimal point.

Example 10.7.4 Drilling cycle

Drilling cycle is performed at all locating points.



G66 P9082 R (<u>Point R</u>)X (<u>Point Z</u>)X (<u>Dwell time</u>):

X ;

M ;

Y Some move block performs drilling cycle within the area.

⋮

G67 ;

The macro is as follows (for incremental programming)

G9082;

G00 Z#18;

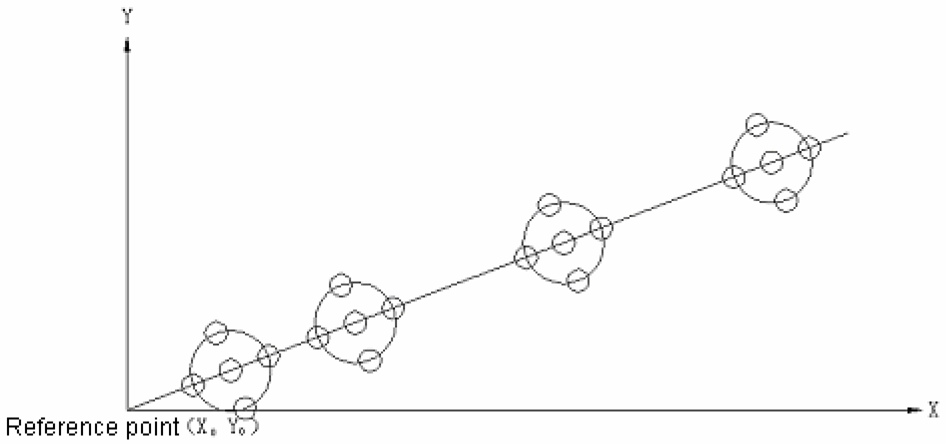
G01 Z#26;

G04 X#24;

G00 Z-[ROUND[#18]+ROUND[#26]];

M99;

Example 10.7.5 Combined type hole group



For a drilling process with the bolt-hole ring as described in 10.7.2 and a unequally spaced hole group on an oblique line as described in the above example 10.7.3, it is necessary to perform it with macro and fixed cycle. The program is as follows:

G81…………;

G65 P9200 X (coordinate of reference point) Y (coordinate of reference point);

G66 P9207 R (radius) A (starting angle) K (number of holes);

G65 P9203 A (angle) I (spacing) K (number) I (spacing);

G67;

3.10.7.3 Multiple calling

It is equivalent to calling a subprogram from another one. It is also possible to call a macro from another one. Multiple calling includes single calling and modal calling. Its number of repetitions is up to 4.

3.10.7.4 Multiple modal calling

In modal calling mode, move instruction is executed whenever the specified macro is called. When several modal macros are specified, the move instruction of the previous macro will be executed whenever the next macro is called. Macros are called by the following specified instructions continuously.

Example 10.7.6

G66 P9100;
Z10000; (1-1)

G66 P9200;

Z15000; (1-2)

G67; : P9200 Cancelled

G67; : P9100 Cancelled

Z-25000; (1-3)

09100;

X5000; (2-1)

M99;

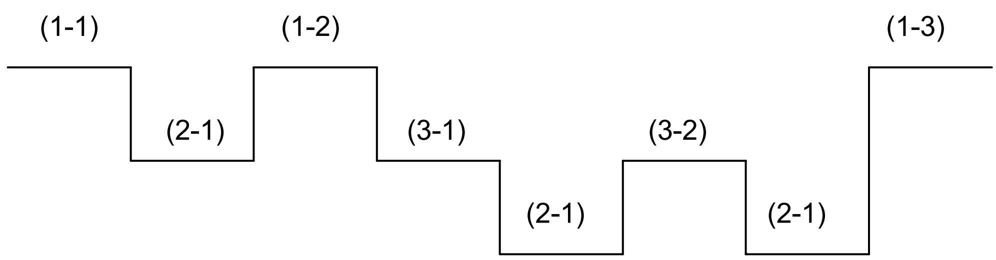
09200;

Z6000; (3-1)

Z7000; (3-2)

M99;

Executing order (the blocks without instruction are omitted)



(Note): Since (1-3) is not followed by macro calling mode, modal macro is not called.

3.10.7.5 Calling a macro with G codes

A G code for calling macro may be set by parameter, namely substituted by N_G65 P△△△△<Independent variable assignment>. The same motion may use the following simple instruction.

N__G××<Independent variable assignment>

The correspondence between calling macro with G code xx and calling the program number △△△△ may be set by parameter.

The program number △△△△ for calling G code xx and calling macro is set in parameters.

Except G00, up to 10 of G01 to G255 can be selected to call macro. These G codes cannot be specified through MDI panel like G65 instruction. The G codes cannot be set in the macro calling instructions for G codes and used in the subprogram calling instructions with M codes.

Set the following parameters:

0323	The G code for calling macro: 9010
0324	The G code for calling macro: 9011
⁞	
0332	The G code for calling macro: 9019

Example 10.7.7 : Clockwise machining with G02

G02 I (Radius)D (Offset number);

(1) Set the following parameter

Macro body: 9010 calls G code=12

(2) Record the following macro body

```
09010;
#1-ABS[#4]-#[2000+#7];
IF[#1 LEO]GOTO 1;
#2=#1/2;
#3003=3;
G01 X[#1-ROUND[#2]]Y#2;
G17 G02 X#2 Y-#2R-#2;
I-#1;
X-#2 Y-#2 R#2;
G01 X[#-ROUND[#2]]Y#2];
#3003=0;
N1 M99;
```

3.10.7.6 Calling a subprogram with an M code

The M code set by parameter may be used to call a subprogram. The instruction of N__G__X Y__……M98P△△△△ : may be substituted by the following simple instruction.

N G X Y ……M××:

For M98, a subprogram is indicated on COMND page, but MF and M codes are not sent.

The correspondence between calling macro with M code xx and calling the program number △△△△ may be set by parameter.

Except the No. 35 and 36 M30 parameters $MBUF_1$ and $MBUF_2$, at most 3 of M03 to M97 can be used for macro calling.

Instruction may be specified through MDI keypad. In the macros called by the means that a G code calls a macro or in the subprogram called by the means that a T code calls a subprogram, the subprograms of the specified M codes cannot be called like common M codes.

Set the following parameters:

0320	The M code for calling macro: 9001
0321	The M code for calling macro: 9002
0322	The M code for calling macro: 9003

Example 10.7.8 : Through the ATC fixed cycle of M06

(1) Set the following parameter

Subprogram: the M code called by 9001=06。

(2) Record the following macro bodies:

```
09001;
#1=#4001;
#3=#4003;
G28   G91   Z0   M20;
G28   Y0;
M21;
G00   Z10000;
M22;
G28   Z0;
M23;
G#1   G#3   M99;
```

3.10.7.7 Calling macros with M codes

The M codes set by parameter may call a macro, i.e. N—G65 P△△△△<Specified variable>

Operation is performed when using the following instruction instead.

N—M××<Specified variable>

The calling of the program number △△△△ of macro is set by relevant parameter.

Except a part of specified M codes, up to 10 of M06 to M255 can call macro. However, this type of M code cannot be input through MDI like G65 or used in the subprograms called with G code, M code and T code.

Set the parameters as follows:

043	User macro body: 9020 calls M mode
⁞	⁞
052	User macro body: 9029 calls M mode

3.10.7.8 Calling a subprogram with a T code

The T codes set by parameter may call a subprogram.

N__G__X__Y__………………………Tt;

Operation is performed when using the following two instructions of the program instead.

#149 = t;

N　G　X　Y　………………………M98　P90000;

The t in T codes is saved as the independent variable in variable #149. T codes are displayed on COMND page, but TF and T codes are not sent. They may be specified through MDI but cannot be instructed in the blocks with the M codes for calling a subprogram.

In the macro called by the means for G codes to call a macro as well as in the subprogram called by the means for M or T codes to call a subprogram, these T codes does not call a subprogram but are treated like common T codes when instructed.

Set the following parameter:

0306							TMCR	

3.10.7.9 The position of the decimal point of an independent variable

Independent variable is usually specified with a decimal point. If a decimal point is not specified, the position of the decimal point is assumed as follows:

Addresses	Input in mm	Input in inch
A, C	3 (2)	3
B (B 3-digit is not selected)	3 (2)	3
B (B 3-digit is selected)	0	0
D, H	0	0
E, F (in G94 mode)	0 (1)	2
E, F (in G95 mode)	2 (3)	4
I, J, K	3 (2)	4
M, S, T	0	0
Q, R	3 (2)	4

U, V, W	3 (2)	4
X, Y, Z	3 (2)	4

The positions of the decimal point are calculated from the least significant bit for the numerical values listed in the above table.

The numerical values in () indicate the number of the digits on the right side of the decimal point. For addresses E and F, parameter FMIC=1.For other addresses, MIC=1.

3.10.7.10 The difference between M98 (calling a subprogram) and G65 (calling a macro body)

(1) G65 may contain an independent variable, but M98 not.

(2) After executing an instruction other than M, P or L in M98 block, the use of M98 goes to a subprogram but G65 only transfers.

(3) When M98 block contains an address other than O, N, P and L, a single block stops execution, but G65 block not.

(4) G65 may change the level of a local variable, but M98 cannot. That is to say, the #1 specified before G65 is different from the #1 in a macro. The #1 specified before M98 is identical to the #1 in a calling subprogram.

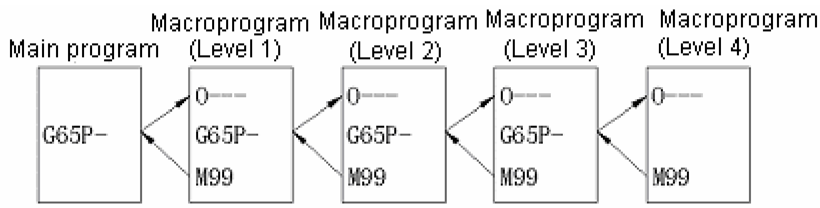
(5) It is possible to call a nesting for 4 times when G65 is combined with G66. M98 can also call for 4 times (when Macro A or B is selected).

(6) During automatic operation, M98 can achieve 4 calls at most in TAPE mode or MEMORY mode when an operation is inserted through MDI. One or two codes may achieve 4 calls in MDI mode. G65 can achieve up to 4 calls in all modes.

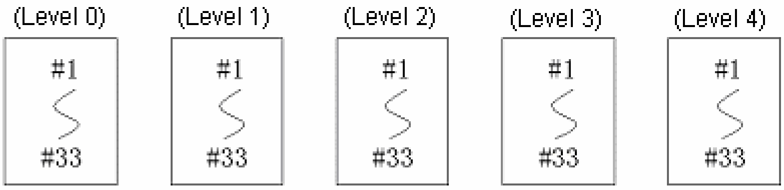
3.10.7.11 The nesting and local variables of user macro

When G65, G66 and G codes are used to call a macro, the nesting degree (level) of its macro increases by 1 and the level of its local variable also increases by 1.

The relationship between macro calling and local variable is as follows:



Local variables



① Note that local variables (Level 0) #1 to #33 are provided in a main program.

② When a macro (Level 1) is called by G65, the local variable (Level 0) of the main program is saved, the local variables #1～#33 (Level 1) of the new macro (Level 1) are prepared and the substitution of the independent variables are possible (the same for ③).

③ Each group of local variables (Levels 1, 2 and 3) are saved ant new local variables (Levels 2, 3 and 4) are prepared whenever a macro (Levels 2, 3 and 4) is called.

④ When using M99 to return from each macro, the local variables (Levels 0, 1, 2 and 3) saved in ② and ③ are restored as they are saved.

3.10.8 The relationship with other functions

(1) MDI operation

Macro call instruction, operation instruction and control instruction cannot be specified with MDI.

During the execution of a macro and the stop of a single block, the MDI instructions other than those are related to macros may be executed.

In macro calling mode (G66), inputting a move instruction by MDI cannot perform macro calling.

(2) Sequence number retrieving

The sequence numbers in a macro body cannot be retrieved.

(3) Single block

The blocks other than macro call instruction, operation instruction, control instruction sometimes may be processed in single block stop mode in a macro.

The blocks of macro call instruction (G65, G66, and G67), operation instruction and control instruction do not stop in the workpiece with single block.

However, the blocks other than macro call instruction may perform single block stop and be set through the following settings and parameters.

For the check of a macro body

0318		MCS9						
0319		MCS8						MCS7

When MCS7 = 1, single block stop will be performed in the macro statements in 01 to 07999 and 09900～09999.

When MCS8 = 1, single block stops in the macro statements in 08000～8999.

When MCS9 = 1, single block stops in the macro statements in 09000～9899.

However, when single block stops in a macro in offset compensation mode C, it is assumed that it does not to move. Sometimes wrong compensation is also performed (strictly speaking, instructing movement is similar to that the amount of movement is zero). The assumption is preferential for the single block stop restraint of #3003. In a word, when MCS7, 8 and 9 are equal to 1, #3003 will be equal to 1 (also called 3) in the programs in all program sequence numbers. All single blocks will be restrained. Here MCS7, 8 and 9 are the parameters for the inspection of macros. Therefore, the parameter shall be set to 0 at the end of macro inspection.

(4) Skip optional blocks

When / code appears in <Expression> (on the right side of working equation or in [ ]), it may be deemed as a division operator rather than an optional block.

(5) Operation in EDIT mode

In order to prevent damage caused by misoperation, the recorded macro bodies and subprograms may be set as follows.

0318	PRG9							

0319	PRG8							

Here PRG8 = 1 corresponds to the user macros and subprograms of program numbers 8000～8999 while PRG9 = 1 to those of 9000～9899. Recording, clearance and edition are not allowed. However, clearance of all blocks and output of single programs can be carried out upon tenderization.

(6) Indication of the program numbers other than EDIT mode

As a rule, the called programs will be displayed when calling a user macro and a subprogram. The following setting may be used to maintain the foregoing programs.

0318			MPD9					

0319			MPD8					

MPD8 = 1 corresponds to the user macros and subprograms of program numbers 8000～8999 while MPD9 = 1 to those of 9000～9899. These programs are not displayed in the PROGRAM page for the modes other than EDIT.

(7) Reset

When reset function is used for clearance, all local variables and public variables #100 to #149 are in <Empty> mode and system variables #10000 through #1132 cannot be cleared.

The calling mode of user macro and subprogram as well as the state of D0 will be cleared and the main program be returned to in the cases other than the clearance in MDI mode. For the clearance in MDI mode, only the calling state in MDI mode is cleared.

(8) Macro statements and NC statements

The following blocks indicate the statements of a macro.

① Operation instruction (= is also included in the block)

② Control instruction (G0T0, D0 and END are included in the block)

③ Macro call instruction (G65, G66, G67 and the G codes for calling a macro are included in the block).

The blocks other than macro statements are NC statements.

(9) MDI's interference in automatic operation

When the MDI in automatic operation is used to insert a macro, up to 4 levels of the nesting degree called by macros and that of D0 can be continuously called from the beginning of automatic operation.

(10) The display of PROGRAM RESTART page

The M and T codes used for calling a subprogram are not displayed like M98.

(11) Feedrate

When Feedrate is set to ON, the execution of macro statement stops (also stops during alarm clearance).

3.10.9 Special codes and words used in user programs

Besides those for common programs, the codes used in user macros include the following codes:

(1) ISO

Meaning	Binary encoding	Symbol
[	1 1 0 1 1 ° 0 1 1	[
]	1 1 0 1 1 ° 1 0 1	]
#	1 0 1 0 0 ° 0 1 1	#
*	1 0 1 0 1 ° 0 1 0	*
=	1 0 1 1 1 ° 1 0 1	=
0	1 1 0 0 1 ° 1 1 1	0
+	0 0 1 0 1 ° 0 1 1	+

(2) EIA

Meaning	Binary encoding	Symbol
[	0 0 0 1 1 ° 1 0 0	
]	0 1 0 0 1 ° 1 0 0	
#	Parameter setting	
*	0 0 0 0 1 ° 1 1 0	&

=	0 0 1 1 1 ° 0 1 1	,
+	0 1 1 1 0 ° 0 0 0	+

0: The 0 code similar to the 0 of program number shall be used.

The # of EIA codes and code format shall be set by parameter.

However, Chinese characters cannot be used. Latin letters are usable. If # is used, note that its original meaning is not applicable.

Parameter No.

0317								

The special characters used by Macro A are as follows: OR, XOR, IF, GOTO, EQ, GT, LTT, GE, LE.

The special characters used by Macro B are as follows:

AND, SIN, COS, TAN, ATANN, SQRT, ABS, BIN, RCD, ROUND, FIX, FUP, WHILE, DO, END.

3.10.10 Restrictions

(1) Usable variables

#0, #1～#33, #100～#149, #500～#509 and system variables

(2) Usable variable value

Maximum $\pm 10^{47}$, minimum $\pm 10^{-39}$

(3) Rated numerical value used in <Expression>

Maximum ±99999999, minimum ±0.0000001

Decimal point: usable

(4) Operational precision: decimal 8 digits

(5) Nesting degree of macro calling: up to 4 levels

(6) Repetition Identification number: 1～3

(7) Nesting of [　]: up to 5 levels

(8) Nesting degree of subprogram calling: up to 4 levels

(9) The above-mentioned functions: User Macro B may perform all of them while A can only perform the following operations.

(i) The variables beyond the amount are applicable.

(ii) The following operations may be performed between variables:＋, －, OR, XOR.

(iii) IF [<Conditions>] GOTO　n is applicable.

(iv) Simple calling and modal calling are possible.

3.10.11 Descriptions for P/S alarm

① Alarm No.004

Addresses are not found in proper positions

(Example): X1*1:

No.004 alarm will be given when "*" instead of the next address appears after X1.

② Alarm No.114

The formats other than <Formula> are incorrect. This type of alarm indicators up a lamp in the following conditions:

(a) The characters following an address shall not be numerical values, • , －, #, [, ＋

(Example): XF1000:

XSIN [10]:

(b) The formats other than IF (also called WHILE)[<Formula>ΔΔ<Formula>]

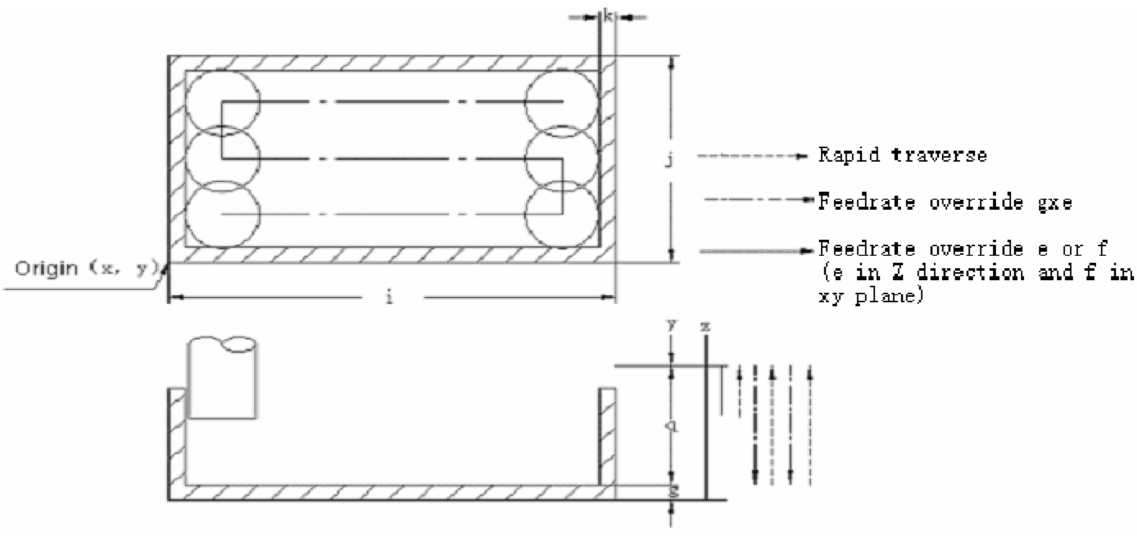
(Example) IF [#1　EQ　#2] GOTO　10:

WHILE [#1　SIN#2] DO1:

3.10.12 Examples of user macros

3.10.12.1 Groove machining

User macro performs fixed cycles of groove in the range of the figure below, where Z is the machining range of certain depth and z the cutting amount of the machining range.



(1) User macro call instruction

G65 P9802 X x Y y Z z R r Q q I I J j K k T t D d F f E e *

The meaning of all addresses

xy: Absolute coordinates of the axes X and Y (Left bottom corner of the groove) at the origin.

zr: Absolute coordinates of Point Z and Point R (See the figure)

g: The cutting amount of each machining (positive)

ij: The lengths (positive) in the directions of X and Y in machining area (see the figure) (the machining efficiency will be higher when i> j.

k: Allowable amount at the end

t: The machining width shall not exceed the tool radius xt%

d: Tool radius compensation number (01～99)

f: The feedrate in xy plane

e: The feedrate during cutting in, the feed is at 8Xe feedrate 1mm before cutting in.

(2) User macro body

```
0 9802;
#27＝#[2000＋＝#7];
#28=#6+#27;
#29=#5-2*#28;
#30=2*#27*#23/100;
#31=FUP[#29/#30];
#32=#29/#31;
#10=#24+#28;
#11=#25+#28;
#12=#24+#4-#28;
#13=#26+#26+#6;
G00 X#10 Y#11;
Z#18;
#14=18;
D01;
#14=#14－#17;
IF[#14GE13]GOTO 1;
#14=#13;
```

```
N1  G01  Z#14  F#8;
X#12 F#9;
#15=1;
WHILE[#15  LE  #31] D02;
Y[#11+#15*#32];
IF[#15  AND  1 EQO]GOTO02;
X#10;
GOTO  3;
N2 X#12;
N3 #15=#15+1;
END2;
G00 Z #18;
X#10 Y#11;
IF[#14 LE#13]GOTO 4;
G01 Z[#14+1F[8*#8];
END1;
N4 M99;
```

3.10.13 External output instructions

Besides typical user macro instructions, the following macro instructions (external output instructions) may be executed.

(a) BPRNT

(b) DPRNT

(c) POPEN

(d) POLOS

These instructions are for the purpose of outputting values and texts of variable through RS232 interface.

These instructions shall be instructed in the following order:

① OPEN instruction: POPEN

Get external I/O equipment interface ready before outputting a series of data instructions.

② Data output instruction: BPRNT and DPRNT

Execute the necessary data output instructions.

③ Close instruction: POLOS

The instruction shall be used at the end of all data output instructions to disable external I/O devices and interfaces.

3.10.13.1 OPEN instruction: POPEN

POPEN:

While an external I/O device and its interface is executing an instruction, the instruction shall be executed to output DC2 control code from the NC side before outputting a series of data instructions.

3.10.13.2 Data output instruction BPRNT DPRNT

(1) BPRNT [a #b [o] — — — —]

Character Variable Number of effective digits after decimal point

The output of characters and binary output of variables are performed during the execution of BPRNT instruction.

(a) Characters: Instructed characters output as ISO codes. The characters that can be instructed include:

- Latin letters (A～Z)
- Numerical values
- Special characters (*,/,+,-)

  "*" is output as a space code.

(b) Since all variables with a decimal point will be saved. The number of the valid digits after decimal point is indicated with the parentheses following a variable instruction. The variable value that takes the digits after decimal point into account is indicated with a 2-character data (32bit) and starts from high byte outputting in binary data.

(c) EOB codes outputs with ISO codes after outputting instruction data.

(d) The variables of <Empty> cannot be output (with 114#p/s alarm)

(2) DPRNT [a #b [c d]

Number of the digits after decimal point
Number of the digits before decimal point
Variable
Character

The output of characters and digits of numerical values may be performed with ISO codes during the execution of DPRNT.

(a) Refer to the descriptions for the points (a), (c) and (d) of BPRNT instruction.

(b) During the output of a variable value, the variable number is specified after character #. Here the numbers of the digits before and after decimal point are specified in parentheses.

The number of digits of a variable value starts from high-byte valid digit. Each digit and its decimal point are output through ISO codes.

A variable value consists of up to eight digits. If its high-byte digit is 0, then no code will be output when No. 315 parameter PRT=1 and space code is output when PRT=0.

Whenever no decimal is output, + code outputs space code when No.315 parameter PRT=0 in the situation of its sign is positive (+). No code will be output when PRT=1.

3.10.13.3 Close instruction PCLOS

PCLOS

To release the machining link of external I/O unit, the instruction is specified at the end of all data output instructions. DC4 control codes are output through NC.

3.10.13.4 Necessary settings for using the function

(1) Set No.341 parameter so as to use the output unit RS232C for communication outputting.

(2) Set all data (baud rate, etc) of the RS232C interface for one of No. 310 to 313 parameters according to the above output unit predetermined for No.341 parameter.

(3) Set the ISO codes as output codes.

(4) Set No.315 parameter so as to determine whether to put space for the preceding 0 when outputting data with DPRNT instruction.

0	3	1	5

PRT							
7	6	5	4	3	2	1	0

The leading zero is treated by PRT DPRNT instruction as follows during the output of data.

0: Output space 1: Not output

3.10.13.5 Cautions

(1) It is unnecessary to continuously set the open instruction (POPEN), data output instruction (BPRENT, DPRNT) and close instruction (PCLOS). After setting the open instruction at the beginning of a program, it is not necessary to set the open instruction until the close instruction is set.

(2) Open instruction and close instruction shall be set in pair without omission.

That is, close instruction shall be given at the end of a program. It is impossible to individually set the close instruction without the open instruction.

(3) Reset the data output instruction in the execution of stop program and cancel the data

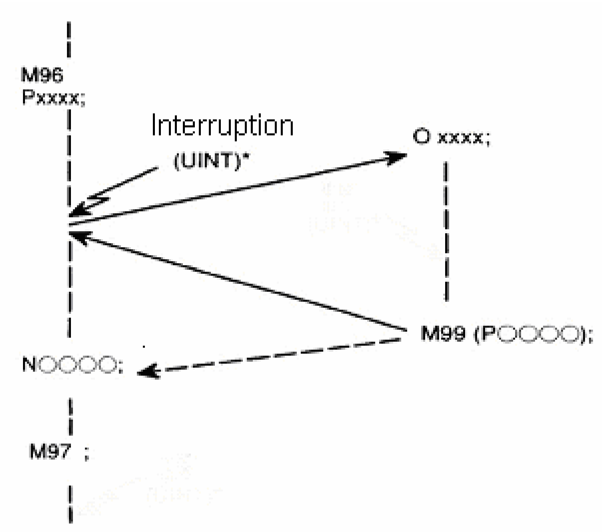
that follows.

If the reset process is instructed by with M30 or a similar instruction at the end of a data output program, you need to specify the close instruction at the end of the program and to wait until all data is output before the start of M30 or other reset process.

(4) It is necessary to select Macro B and I/O interface for the function.

3.10.14 Macro interruption function (Macro B)

If a interruption is input for NC between M96 PX X X X; and M97; blocks, control will go to PX X X X program.



Setting M99; program returns from the original program. The sequence number of the original program returned to may be set with address P.

(Note 1): Refer to Appendix 11 for the details of the functions of macro.

(Note 2): Make sure to refer to the operation manual supplied by the manufacture of the machine when using this function.

3.11 Tool Life Management

Tools are divided into serial groups. Specific tool life (in time or number of cycles) is specified for each group. The so-called tool life management function refers to the capability of totalizing the tool life of all groups in service and replacing a tool in the predetermined order in the same group.

3.11.1 Setting of tool groups

The order of the tools in each group and the life of each tool are preset in the NC device in the following format.

Format	Meaning
O □□□□	Program number
G 1 0 L 3	To set the beginning of tool groups
P □□□ LΔΔΔ	What follows P is the group number of N01~128

	What follows L is the tool life of No.1～9999 tools (Note 1)
TΔΔΔΔH O O D □□	(1) What follows T is a tool number
TΔΔΔΔH O O D □□	(2) What follows H is a tool length offset number
┆	What follows D is a tool offset number
TΔΔΔΔH O O D □□;	(N) Tool selecting order: (1), (2) till (N)
P □□□ LΔΔΔ;	The data of the next group
TΔΔΔΔH O O D □□;	
┆	
G11:	To set the end of tool group
M02(M30):	End of program

The setting procedures are as follows:

(i) Like the general DNC functions, press ENTER in "EDIT" mode after activating the DNC communication interface. Programs will be loaded into the part program storage and get ready for display and edition.

(ii) In storage mode, perform a cycle starting operation so as to run the programs. Data will be saved in the tool life data area. At the same time, the tool life data of all tool groups saved earlier will be deleted and the tool life counter cleared. Once data is saved, it will not lose even after power failure.

(iii) In the operation of Step (i), perform a cycle starting operation in DNC mode and save the contents of the program directly into the tool life data area through RS232. Now display and edition cannot be performed as Step (i).

(Note 1) Whether tool life will be indicated in time (min) or frequency (number of cycles) shall be set by parameter (309－LOTM).

(Note 2) One of the following four groups may be selected for tool group number and tool numbers (309－GST1,GST2)

In any type of combination, up to 256 tools can be saved. At most 16 groups can be selected for group ①, each group having 16 tools. Group ② may have 32 groups at most and each group may have 8 tools, and so on. A type of combination may be changed by modifying its parameters and then switching off and on the power supply.

	Group No.	Tool No.
①	16	16
②	32	8
③	64	4
④	128	2

(Note 3) H codes and D codes can be omitted when they are not to be used.

(Note 4) The same tool number may appear for many times or appear in any position in set data. The following is an example of program format.

```
00001;
G10L3;
P001    L0150;
T0011 H02 D13;
T0132 H05 D08;
T0068 H14 D16;
```

Data of No.1 group

```
P002    L1400;
T0061 H15 D07;
T0241 H25 D04;
T0134 H17 D03;
T0074 H08 D21;
P003 L0700;
T0012 H14 D08;
T0202 H22 D02;
G11;
M02;
```

Data of No.2 group

Data of No.3 group

(Note 5) The tool group numbers specified by P are not necessarily continuous and all savable tool groups need not to be set.

3.11.2 Setting in machining processes

Tool groups are set by T codes as follows in machining processes.

Program format	**Meaning**

T▽▽▽▽	To use tool group number + tool life administration invalidation number (Note 1) at the beginning of the next M06 instruction.
MO6T □□□□;	The tools specified by □□□□ (Note 2) are concluded and those specified by ▽▽▽▽ are started.
H□□;	99 : The tool offset specified by valid group number 00 : Cancellation of tool length offset
D00	99 : The tool compensation specified by valid group number. 00: Cancellation of tool compensation
T△△△△ ;	To use the tools set by △△△△ after the next M06 instruction.
MO6T▽▽▽▽;	Tool end number ▽▽▽▽ and tool start number △△△△
M02 (M30);	End of machine program

(Note 1) It is set by tool life administration invalidation number △△△△ from T 0 0 0 0 to T△△△△ as common T code instructions without tool life administration. When T tool code △△△△ plus group number is specified, the tool life administration of the concerned group is administrated. Tool life administration invalidation number is set by parameter.

For example, when the value is 100, T0000 through T0100 will be output as common T codes. When T0101 is specified, the T codes of the tools in No.1 group that have not reached their service life will be output.

(Note 2) The above program format is used for tool return number instruction mode. Tool return instruction is required for tool change. It is not required for other instructions. After that, the T codes of M06 can be neglected. Now the similar tool change operation is performed as above.

The following is an example of the program format whose tool life administration invalidation number is 100 in a tool return number instructing method.

Program format	Meaning

……………	
T 0 1 0 1;	What follows the next M06 instruction is No.1 group of the used tools.
……………	
M 0 6 T 0 0 0 3;	End of the currently used No.0003 tool and start of using No.1 group of tools
……………	
G 4 3 H 9 9;	The setting of the tool length offset number used in No. 1 group
……………	
G 4 1 D 9 9;	The setting of tool offset number used in No. 1 group
…………	
D 0 0;	Cancellation of tool compensation
…………	
H 0 0;	Cancellation of tool length compensation
……………	
T 0 0 0 5;	To use No.0005 tool after the next M06 instruction
…………………	
M 0 6 T 0 1 0 1;	End of No.1 group of tools and start of No.0005 tool

3.11.3 Performance of tool life administration

3.11.3.1 Tool life calculation

(1) When tool life is determined in time (min)

Now, T△△△△ (△△△△＝tool life administration invalidation number＋tool group number) and M06 are instructed in succession. M06 is specified in machine program again. In cutting method, the actual time of tool usage is calculated by specified time interval (4s). The time for single block stop, Feedrate, rapid traverse (positioning), dwell, etc is not included. The maximum set life value is 4300min.

(2) When tool life is determined in number of cycles

Whenever a cycle starting operation is performed, it operates until M02 or M30 is instructed and NC reset. Then the counter for the used tool group increases by 1. The counter increases by 1 even a group is instructed for several times in the same program. Life value is up to 9999. Each group of calculated life and contents of counter will not lose after power failure.

(Note) For specifying life in number of cycles, EXTERNAL RESET (ERS) or RESET AND REWIND (RRW) signal is input in NC when M02 or M30 is executed.

3.11.3.2 Tool change signal and tool change reset signal

Another tool will be selected in the predetermined order after the end of tool life. When the last tool in the same tool group has reached its service life, a tool change signal will be given. The tool to be changed is displayed on LCD. Then the relevant group number is specified and tool change reset signal input or MDI panel (see 3.11.4.3) operated. All data such as life counter *,@, etc (see 3.11.4.2) are cleared. All tool groups are changed and reset when tool change signal is automatically released at the end of tool life. After machining is restored, the group starts selection from the first tool.

(Note) For specifying tool life in time, tool change signal is output once it has reached service life and machining goes on until the end of program. For specifying tool life in number of cycles, tool change signal is output in case of M02 or M30 reset at the end of tool life.

3.11.3.3 New tool selection signal

When a new tool is selected from a group, tool T code and new tool selection signal are output at the same time. When a new tool is selected, the signal may be used for the automatic measurement of tool compensation.

3.11.3.4 Tool skip signal

It is possible to forcibly change a tool even it has not reached its service life.

(I) Set the group where the tool is and input tool skip signal. Use the next T code instruction to select the next tool in the group.

(II) Input tool skip signal without specifying a group number but assuming the selected tool is specified. Follow (i) for other issues.

Following (I) or (II) shall be set by parameter. Service life starts from 0 no matter which method is followed. However, output a tool change signal when the tool skip signal inputs the last tool.

(Note) When STL or SPL or both of them are lit, it indicates that what is input is neither tool change signal nor tool skip signal.

3.11.4 The display and input of tool data

3.11.4.1 The display and modification of tool group number

In the part program storage and edition area, tool group data may be displayed and modified like the edition of common programs. As described in Section 11.1, modified program shall be executed; otherwise it cannot be saved in the tool life data area.

3.11.4.2 The display of tool life data during the execution of machine program

Pressing the DIAGNOSIS button twice in any mode displays the first page of tool life data in the following format on the screen of LCD.

Group No. Service life

```
T00L   LIFF   DATA   01: 00001   N0012
                     SELECTED  GRP  001
GRP    001:   LIFE   9800   COUNT   6501
      *0034  *0078  *0012  *0056
      *0090  @0076   0032   0098
       0054   0010   0099   0087
       0077   0066   0055   0024
GRP    002:   LIFE   0500   COUNT  0112
      *0011  *0022  *0201  *0144
      *0155  *0066  *0176  *0188
      #0019   0234   0007   0112
       0156   0090   0016   0232
GRP    TO  BE  CHANGED :
       003  006  012  013  014  -------->
       N                 LSK  INC
```

Life counter

The tools in Group 1
*: The tool whose life has come to an end
(a): The tool in service
#: The skipped tool

The tool number in Group 2

The tool group number to be changed

Two groups of data are displayed on each page. Pressing the page turning button ⇩ displays all data groups in sequence. Tool change signal may be given to up to 5 groups, which are displayed at the bottom of each page. For 6 or more groups, an arrow will be indicated in the diagram.

To review the data, select address N, enter the group number and press the INPUT button, or pressure press the CURSOR ⇩ button to move the cursor to the GRP of the next group and to display its data.

3.11.4.3 Presetting tool life counter

To modify the tool life counter, select the MDI mode.

(I) P□□□□ and press the INPUT button.

Now the group counter that is identified by the cursor is preset □□□□ while other data in the group remains unchanged.

(II) Type in P－9999 and press the INPUT button.

All the executed data in the group identified by the cursor including * are cleared. It functions as the reset of the tools in the group (see Section 3.11.3.2).

3.11.5 Other cautions

Part program storage and edition area will reduce the storage area in the last part for the purpose of tool life data area. When data is saved in the part program and edition area in EDIT mode as described in Section 11.1, more area will be occupied.

3.12 The Indexing Function of Indexing Worktable

The 4th axis (e.g. Axis B) may be used for the indexing of indexing work. Indexing instruction only employees the angle specified by Address B. The process becomes simple as it is unnecessary to set the M codes for worktable tensioning and releasing.

3.12.1 Instructing methods

3.12.1.1 Input unit

Decimal point, B 1…1° shall not be used.

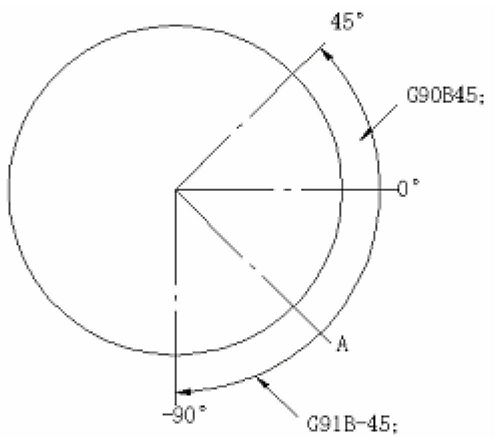
Note: When a decimal is used, PS alarm will be given one the digits after it are instructed. (N O 180) means that the value less than 1° cannot be instructed.

3.12.1.2 Absolute / incremental instruction

Absolute / incremental instruction can be instructed by G90/G91.

Absolute instruction G90 B45; Indexing of 45° position

Incremental instruction G 91 B－45; Indexing of negative rotation 45°



The point A in the above figure is the actual position. The mentioned instruction moves as shown in the above figure.

3.12.1.3 Concurrently controlled axes

Axis B shall be individually specified. PS alarm will be given when X, Y, Z or the 5th axis is instructed along with Axis B (NO181).

3.12.2 Minimum travel unit: 0.001 degree/pulse

3.12.3 Feedrate

As a rule, the feedrate of axis B is a rapid one regardless the state of Group 01 G codes (G 00, G01, G02 and G03). When Axis B is instructed in G 00, G01, G02 or G03 mode, the G 00, G01, G02 and G03 in the blocks regarding other axes are still valid, and hence it is not necessary to specify G 00, G01, G02 and G03 again.

G01 X10 F5; Axis X operates at cutting feedrate.

B45 ; Axis B operates at cutting feedrate.

X29 ; Axis X operates at cutting feedrate.

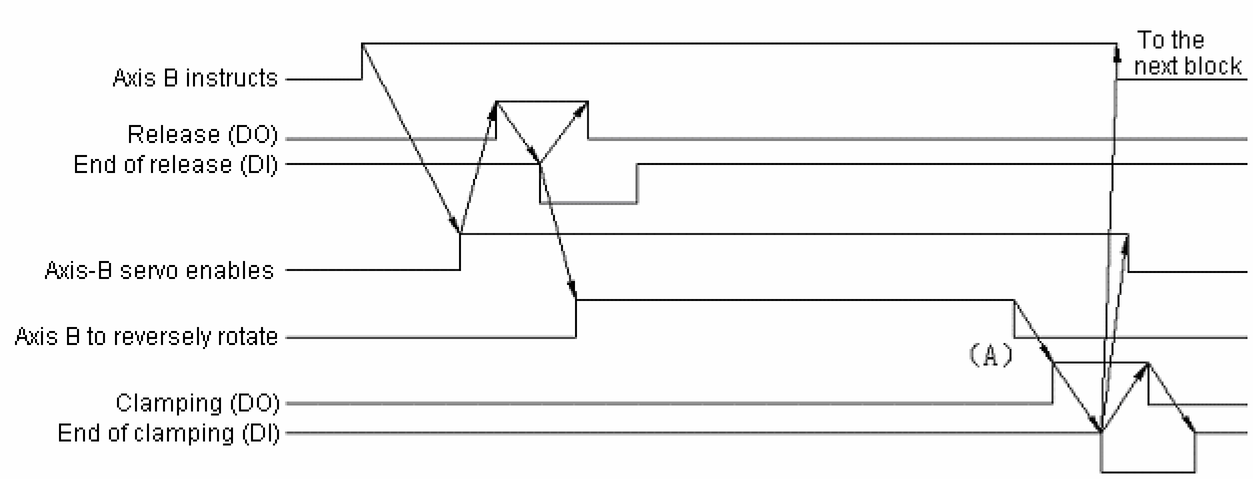
(G01 is still valid.)

No-load operation is invalid.

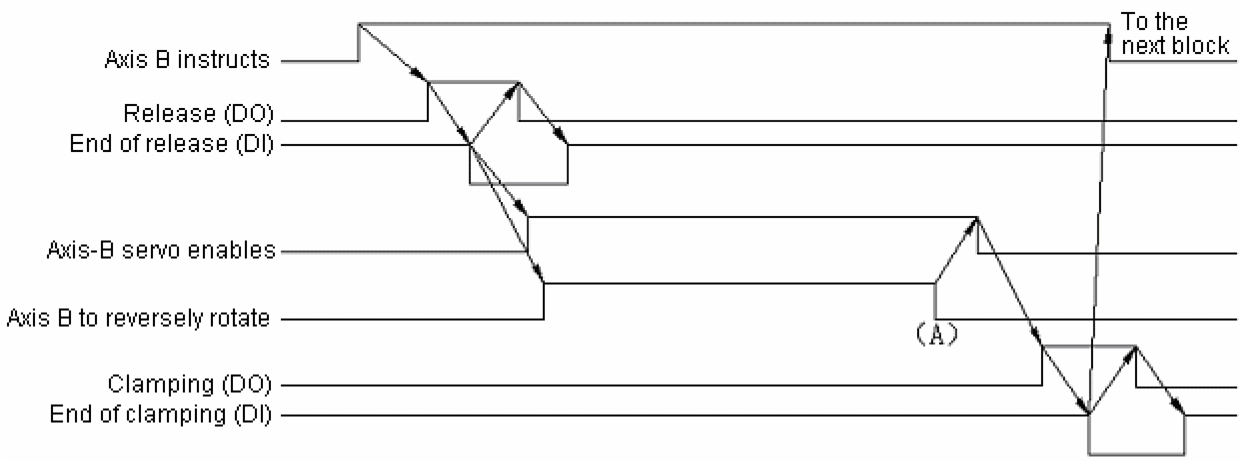
3.12.4 The clamping and release of indexing worktable

The clamping and release of indexing worktable is automatically performed before and after the movement of Axis B.

(1) Indexing sequence A



(2) Indexing sequence B



Positioning check is always performed at the Point A when the above indexing sequence A/B is selected by parameter setting.

Note 1: Clamping or releasing signals are cleared when the NC is reset in the waiting state after clamping or release. NC unit then completes the waiting state and enters into reset state.

Note 2: In clamping or releasing mode, these states remain unchanged even the unit is reset, namely the sequence of release or clamping cannot be automatically executed through reset, but clamping are releasing signals are cleared.

Note 3: The waiting state after clamping or releasing is displayed in diagnosis mode display state (DGN701-BCNT).

3.12.5 Jog/step/handwheel

Operations in Jog/step/handwheel mode cannot be performed for Axis B. However it may return to the reference point in Jog mode. Travel stops once the axis selection signal becomes “0” when manually returning to the reference point. Clamping instruction is not performed. In order to avoid the problem, set the sequence program on the machine side so that the axis selection signal will not become “0” before returning to the reference point.

3.12.6 Other cautions

(1) The indication of the actual position on the screen of LCD, indication of external position and the indication on the screen of COMND have a decimal point.

Example: B180.000

(2) Whether the internal absolute coordinates of NC for Axis B use 360° full circle is set by parameter (314－IRND).

(i) when IRND＝0, absolute coordinate is rounded to 360° and starts from 0° position. If G90 B720; is specified, Axis B rotates by 720° (2 turns) and the actual position indication and absolute coordinate is 720°.

(ii) When IRND＝1, absolute coordinate and the actual position are rounded to 360°. However the rounding of absolute coordinates is performed after the travel for increment determination. That is, if G90 B720; is specified from 0° position, Axis B will rotate by 720° (2 turns) and absolute coordinate will be 0°. Now the actual position changes as follows:

0° ⟶ 90° ⟶ 180° ⟶ 270° ⟶ 0° ⟶ 90° ⟶ 180° ⟶ 270° ⟶ 0°.

The result of 360° rounding, absolute coordinates and actual position are displayed between 0° and 359°.

For the RELATIVE in the display of the actual position (ABSOLUTE RELATIVE), rounding is only performed when parameter No. 007 PPD＝1.

No matter it is in the condition of (i) or (ii), the mechanical coordinate system often uses 360° for rounding. When automatic return to the reference point is specified, (G28) calculates the amount of movement with the mechanical coordinate system. The movement between the intermediate point and reference point is less than 360° (on turn).

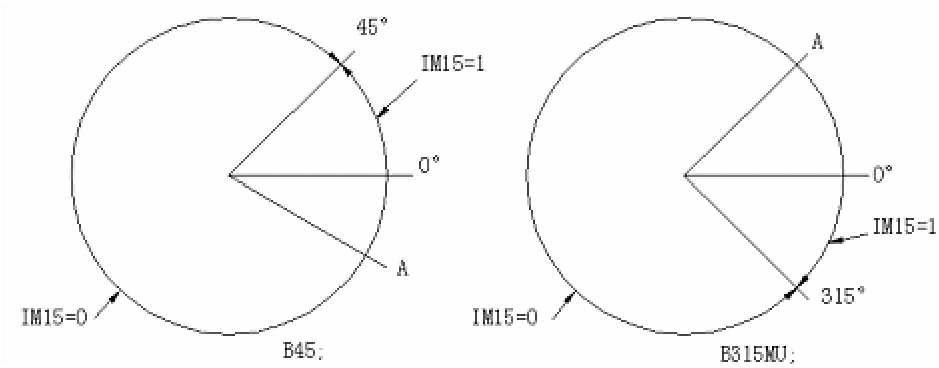
(3) The following requirements are made through parameter setting (314－IM15):

When IM15＝1:

(i) Axis-B instruction shall be always considered as absolute instruction regardless the G90 or G91 mode.

(ii) The rotating direction is positive.

(iii) When M15 is pacified in the same block as the Axis-B instruction, the rotating direction is negative.



(Note) Though M15 is processed inside the NC, FIN signal returns to the NC because MF and M codes are sent to the machine side.

(4) During the movement of Axis B, Feedrate, reset and emergent stop are valid. Proper workpiece shall be done on the machine side in order to prevent stop in midway.

(5) When the option is adopted, the additional axis servo ON signal (*8VF4) will become invalid.

(6) Standard additional axis still applies to the requirements, parameters and inter-unit connection that are not described in the instruction manual.

广州数控设备有限公司

GSK CNC EQUIPMENT CO., LTD.

Add: No.52, 1st . Street, Luochong North Road, Luochongwei, Guangzhou, 510165, China

Website: http://www.gsk.com.cn E-mail: gsk@gsk.com.cn

Tel: 86-20-81796410/81797922 Fax: 86-20-81993683

All specification and designs are subject to change without notice. Aug. 2007/Edition 4

Aug. 2007/Printing 4