CHAPTER 4  CONTROL FUNCTION of ADDITIONAL AXIS ................................................................. 106

4.1 General ............................................................................................................................................... 106
4.2 Axis Name .......................................................................................................................................... 106
4.3 Axis Display ....................................................................................................................................... 106
4.4 Axis Startup ........................................................................................................................................ 107
4.5 The Additional Axis is Linear Axis ..................................................................................................... 107
Volume II  OPERATION

CHAPTER1  OPERATION MODE AND DISPLAY .................................................................163
  1.1 Panel Division .......................................................................................................163
    1.1.1 State indication ...............................................................................................163
    1.1.2 Edit keypad .....................................................................................................164
    1.1.3 Menu display ...................................................................................................165
    1.1.4 Machine panel .................................................................................................166
  1.2 Summary of Operation Mode ................................................................................169
  1.3 Display Interface .....................................................................................................170
    1.3.1 Position interface .............................................................................................173
    1.3.2 Program interface ............................................................................................175
    1.3.3 Tool offset, macro variable and tool life management interface ....................177
    1.3.4 Alarm interface .................................................................................................181
    1.3.5 Setting interface ...............................................................................................183
    1.3.6 BIT PARAMETER, DATA PARAMETER, PITCH COMP interface ...............188
    1.3.7 CNC DIAGNOSIS, PLC STATE, PLC VALUE, machine soft panel, VERSION
      MESSAGE interface ................................................................................................190
  1.4 List of general operations .......................................................................................193

CHAPTER 2  POWER ON OR OFF AND PROTECTION .......................................................199
  2.1 System Power On ....................................................................................................199
  2.2 System Power Off ...................................................................................................199
  2.3 Overtravel Protection .............................................................................................201
    2.3.1 Hardware overtravel protection ......................................................................200
    2.3.2 Software overtravel protection .......................................................................200
  2.4 Emergency Operation ............................................................................................201
    2.4.1 Reset ................................................................................................................201
    2.4.2 Emergency stop ...............................................................................................201
    2.4.3 Feed hold .........................................................................................................201
    2.4.4 Power off .........................................................................................................201

CHAPTER 3  MANUAL OPERATION ..................................................................................202
  3.1 Coordinate axis moving .........................................................................................202
    3.1.1 Manual feed .....................................................................................................202
    3.1.2 Manual rapid traverse .....................................................................................202
    3.1.3 Manual feedrate Override adjustment .............................................................203
    3.1.4 Manual rapid Override adjustment ..................................................................204
    3.1.5 Relative coordinate clearing ...........................................................................204
  3.2 Other Manual operations .......................................................................................205
    3.2.1 Spindle CCW, CW, stop control ....................................................................205
    3.2.2 Spindle Jog .......................................................................................................205
    3.2.3 Cooling control ...............................................................................................205
    3.2.4 Lubrication control ........................................................................................205
    3.2.5 Spindle override adjustment ...........................................................................206

CHAPTER 4  MPG/STEP OPERATION ...............................................................................207
  4.1 Step Feed ................................................................................................................207
    4.1.1 Increment selection .........................................................................................207
    4.1.2 Moving direction selection .............................................................................208
  4.2 MPG (Handwheel) Feed .......................................................................................208
    4.2.1 Increment selection .........................................................................................208
    4.2.2 Moving axis and direction selection ...............................................................209
    4.2.3 Explanation items ............................................................................................209

CHAPTER 5  MDI OPERATION .........................................................................................210
CHAPTER 6  PROGRAM EDIT AND MANAGEMENT ................................................................. 215

6.1 Program Creation ....................................................................................................... 215
   6.1.1 Creation of the block number ........................................................................... 215
   6.1.2 Input of the program content ........................................................................... 215
   6.1.3 Search of the character .................................................................................... 217
   6.1.4 Insertion of the character ................................................................................ 219
   6.1.5 Deletion of the character ................................................................................ 221
   6.1.6 Modification of the character ........................................................................... 221
   6.1.7 Deletion of a single block ................................................................................ 221
   6.1.8 Deletion of the blocks ..................................................................................... 221
   6.1.9 Segment deletion ........................................................................................... 223

6.2 Program annotation .................................................................................................. 224
   6.2.1 Annotation for program name ......................................................................... 224
   6.2.2 Block annotation ........................................................................................... 226
   6.2.3 Alter program annotation .............................................................................. 226

6.3 Deletion of the Program .......................................................................................... 226
   6.3.1 Deletion a single program .............................................................................. 226
   6.3.2 Deletion of all programs ............................................................................... 226

6.4 Selection of the Program ......................................................................................... 227
   6.4.1 Search method ............................................................................................. 227
   6.4.2 Scanning method ......................................................................................... 228
   6.4.3 Cursor method ............................................................................................. 228
   6.4.4 Select file by using file list ............................................................................ 228

6.5 Execution of the Program ....................................................................................... 229
6.6 Rename of the Program ......................................................................................... 229
6.7 Copy of the Program ............................................................................................... 229
6.8 Program positioning ............................................................................................... 230
6.9 Program preview ...................................................................................................... 230

CHAPTER 7  AUTO OPERATION ....................................................................................... 232

7.1 Auto Run .................................................................................................................. 232
   7.1.1 Selection of the program to be run .................................................................. 232
   7.1.2 Program start ............................................................................................... 233
   7.1.3 Stop of the auto run ..................................................................................... 233
   7.1.4 Auto run from an arbitrary block ................................................................. 235
   7.1.5 Adjustment of the feedrate override, rapid override ..................................... 236
   7.1.6 Spindle override adjustment ........................................................................ 237

7.2 DNC running ............................................................................................................ 237
7.3 Running state .......................................................................................................... 237
   7.3.1 Single block execution ................................................................................. 237
   7.3.2 Dry run ....................................................................................................... 238
   7.3.3 Machine lock .............................................................................................. 238
   7.3.4 MST lock ...................................................................................................... 238
   7.3.5 Block skip .................................................................................................... 239
   7.3.6 Optional stop ............................................................................................. 239

7.4 Memorizing at power-down ................................................................................... 239
   7.4.1 Program interruption in non-DNC auto operation ....................................... 239
   7.4.2 Interruption at power-down on DNC auto operation .................................. 240

CHAPTER 8  MACHINE ZERO RETURN OPERATION ...................................................... 241

8.1 Machine Zero .......................................................................................................... 241
8.2 Machine Zero Return Steps .................................................................................... 241
CHAPTER 9 DATA SETTING, BACKUP and RESTORE ...........................................................243
  9.1 Data Setting ...........................................................................................................243
    9.1.1 Switch setting ....................................................................................................243
    9.1.2 Graphic setting ..................................................................................................243
    9.1.3 Parameter setting ..............................................................................................245
  9.2 The Password Setting and Alteration........................................................................251
    9.2.1 Entry of the operation level ..............................................................................252
    9.2.2 Alteration of the password ..............................................................................253
    9.2.3 Lower level set .................................................................................................254
  9.3 Data Restore and Backup ..........................................................................................256

CHAPTER 10 ADVANCE OPERATION ..................................................................................258
  10.1 Operation path ......................................................................................................258
  10.2 Operation instructions ..........................................................................................260
  10.3 Attentions ...............................................................................................................261

CHAPTER 11 FLASH OPERATION .......................................................................................262
  11.1. File list ...............................................................................................................262
  11.2. Introduction of general file operation function ......................................................263
    11.2.1 Open and close file folder ..............................................................................263
    11.2.2 Copy the file by one key(current list in C disk—current list in U disk) ..........264
    11.2.3 CNC file search .............................................................................................265
    11.2.4 Open CNC file ..............................................................................................266
APPENDIX

Appendix 1. Dimensions of Additional Panel AP01 ................................................................. 351
Appendix 2 Dimensions for Additional Panel AP02 .............................................................. 351
Appendix 4 Alarm Information ............................................................................................ 352
Appendix 5 Function Configuration of Standard Ladder Diagram ........................................ 357

5.1 Information for Ladder Diagram ...................................................................................... 357
  5.1.1 Introduction ............................................................................................................... 357
  5.1.2 Information of Current Version .............................................................................. 357
5.2 ADDRESS DEFINITION................................................................................................. 358
5.3 FUNCTION CONFIGURATION ....................................................................................... 361
  5.3.1 Spindle CCW and CW Control ............................................................................... 361
  5.3.2 Spindle JOG .............................................................................................................. 362
  5.3.3 Switch Value Control for Spindle Speed .............................................................. 363
  5.3.4 Cycle Start and Feed Hold .................................................................................. 364
  5.3.5 Cooling Control .................................................................................................... 365
  5.3.6 Lubricating control ............................................................................................... 365
  5.3.7 Optional Block Skip ............................................................................................. 366
  5.3.8 Machine Lock ....................................................................................................... 367
  5.3.9 MST Lock ............................................................................................................. 367
  5.3.10 Single Block ....................................................................................................... 367
  5.3.11 Dry Run .............................................................................................................. 367
  5.3.12 Optional Stop...................................................................................................... 368
  5.3.13 Stroke Limit and Emergency Stop ................................................................... 368
  5.3.14 Tri-color Indicator ............................................................................................. 369
  5.3.15 Reset and Cursor Return .................................................................................. 369
  5.3.16 Rigid Tapping .................................................................................................... 370
  5.3.17 Spindle Exact Stop .............................................................................................. 370
  5.3.18 External MPG control ........................................................................................ 371
5.4 Standard Ladder Diagram ............................................................................................. 371
VOLUME I  PROGRAMMING
CHAPTER 1 PROGRAMMING FUNDAMENTALS

1.1 Introduction

GSK980MDa Milling Machine is a new generation of CNC system developed by GSK Company. As the upgraded version of GSK980MD, it supports milling, boring and drilling cycle. It employs 32 bits high-capability CPU and very large scale programmable device FPGA, applies real-time multi-task control technology and hardware interpolation technology, and is able to perform μm level precision motion control and PLC logic control. GSK980MDa is the optimum choice for upgrading CNC milling machine.

Characteristics:

- Five axes control (X, Y, Z, 4th and 5th); 3 axes linkage; optional interpolation precision (1μm/0.1μm); maximum speed 60m/min; optional axis types (linear axis or revolving axis) for the 4th and 5th axes; CS axis control available for the 4th and 5th axes.
- Electronic gear ratio: (1~32767):(1~32767)
- Screw-pitch error compensation, backlash compensation, tool length compensation, tool abrasion compensation and tool nose radius compensation.
- Embedded with PLC can be downloaded to CNC from PC.
- DNC function supports for real-time program transmission for machining.
- Compatible with G commands in GSK980MC, GSK928MA and GSK980MD. 26 kinds of canned cycles, such as drilling/boring, circular/rectangular groove rough-milling, full circle/rectangular finish-milling, linear/rectangular/arc continuous drilling.
- Spindle encoder tapping and rigid tapping can be detected during tapping cycle, so that high precision machining can be performed.
- Metric/inch programming; automatic chamfering function and tool life management function.
- Chinese, English, Russian and Spanish display selected by the parameters.
- Full screen program editing; 40MB program capacity for storing up to 40000 of part programs.
- USB data communication; CNC system upgrading, machining programs reading through U disk and bidirectional transfer between CNC and U disk.
- Alarm log; multi-level passwords for equipment maintenance and management.
- Bidirectional transfer between CNC and CNC, CNC and PC; upgrade of CNC software and PLC programs;
- The installation dimensions and the electric ports are compatible with GSK980MD, GSK980MC.

### Specifications

<table>
<thead>
<tr>
<th>Motion control</th>
<th>Controlled axes: five axes (X,Y,Z,4th and 5th); (for the 4th and 5th axes) optional axis types (linear axis or revolving axis) and CS contouring control available;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Interpolation functions: linear interpolation (for X, Y, Z, 4th and 5th axes); helical interpolation (for X, Y and Z axes); circular interpolation (for arbitrary 2 axes).</td>
</tr>
<tr>
<td></td>
<td>Position command range: -99999999~99999999; least command increment: 1μm/0.1μm; (selected via parameters)</td>
</tr>
<tr>
<td></td>
<td>Electronic gear ratio: command multiplier 1<del>32767, command frequency divisor 1</del>32767</td>
</tr>
<tr>
<td></td>
<td>Rapid traverse speed: maximum 60000mm/min</td>
</tr>
<tr>
<td></td>
<td>Rapid traverse override: F0, 25%, 50%, 100% four levels real-time tuning</td>
</tr>
<tr>
<td></td>
<td>Cutting feedrate: maximum 15000mm/min (feed per min.) or 500mm/r. (feed per rotation)</td>
</tr>
<tr>
<td></td>
<td>Feedrate override: 0~150% sixteen-level real-time tuning</td>
</tr>
<tr>
<td></td>
<td>Manual feedrate: 0~1260mm/min sixteen-level real-time tuning</td>
</tr>
<tr>
<td></td>
<td>MPG feed: 0.001, 0.010, 0.100,1.000mm four gears.</td>
</tr>
<tr>
<td></td>
<td>Acceleration/deceleration type: S-type for rapid traverse; exponential-type for cutting feed.</td>
</tr>
<tr>
<td></td>
<td>Automatic chamfering</td>
</tr>
<tr>
<td>Macro command</td>
<td>31 kinds of arithmetic, logical operations and skip can be achieved by macro command G65</td>
</tr>
<tr>
<td></td>
<td>Macro statement command. eg:IF,WHILE,GOTO</td>
</tr>
<tr>
<td>Operation mode</td>
<td>Seven operation modes: EDIT, AUTO, MDI, DNC, MACHINE ZERO, MPG/STEP and MANUAL.</td>
</tr>
<tr>
<td>Tapping</td>
<td>Tapping function: lead 0.001<del>500mm or 0.06</del>25400 pitch/inch</td>
</tr>
</tbody>
</table>
| **Precision compensation** | Encoder tapping: settable line number of encoder (0 or 100p/r to 5000p/r); no detect for spindle encoder (when the line number is set to 0)  
Rigid tapping: by rotary axis  
Drive ratio between encoder and spindle: (1 ~ 255); (1 ~ 255) |
| **M command** | Backlash compensation: 0 ~ 2.000mm  
Pitch error compensation: 255 compensation points per axis; compensation amount of each point: ±0.255mm.  
Tool compensation: 32 groups tool length compensation, tool wear compensation, cutter compensation C |
| **T command** | Special M commands (redefinition unallowed): M02, M29, M30, M98, M99, M9000 ~ M9999.  
Other M commands are defined or disposed by PLC program.  
M commands defined by standard PLC program: M00, M03, M04, M05, M08, M09, M10, M11, M32, M33 |
| **Spindle speed control** | tool number T01 ~ T32 (32 numbers at most); manual tool change or auto-tool change selected by the parameters; auto tool change sequence set by PLC program.  
Tool life management; 32 groups, 8 kinds/groups of tool life management data |
| **PLC function** | Speed switching value control: S command is defined or disposed by PLC program; the standard PLC programs S1, S2, S3 and S4 directly output; The output of S1, S2, S3, and S4 are closed by S0.  
Speed analog voltage control: the spindle speed per minute commanded by S codes; output 0 ~ 10V voltage to spindle converter; spindle stepless speed changing supports 4 spindle mechanical gears  
9 kinds of basic commands; 23 kinds of function commands; 2-level PLC program involving up to 5000 steps (2μs processing time for each step). 8ms refresh cycle for the first level program; Ladder diagram edit software and communication software downloadable  
Integrated machine panel: 44 points input (key), 44 points output (LED)  
Basic I/O: 41 points input/36 points output |
| **Display interface** | Display: 480×234 lattice, 7” wide-screen multi-color LCD,  
Display modes: Chinese, English, Russian, Spanish display selected by parameters; machining path displayable |
| **Program edit** | Capacity: 40MB for up to 40000 part programs; custom macro program call; 4 nesting-levels of subprogram  
Edit modes: full-screen editing; absolute/incremental programming |
| **USB** | CNC system upgrade  
Part programs reading in USB  
Bidirectional files transfer between CNC and USB (including programs, parameters, PLC backup and recovery) |
| **Clock display** | Clock, date and week display. |
| **Serial Communication** | bidirectional transfer between CNC and PC, CNC and CNC (involving programs, parameters, tool compensation data); download and upgrade of system software and PLC program serial ports |
### Matching drive unit
AC servo or step drive device by using the pulse+direction signal input. (DA98 or DY3 series)

### G Code Table

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
<th>Code</th>
<th>Function</th>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>G00</td>
<td>Positioning (rapid traverse)</td>
<td>*G54</td>
<td>Workpiece coordinate system 1</td>
<td>G92</td>
<td>Coordinate system setting</td>
</tr>
<tr>
<td>*G01</td>
<td>Linear interpolation</td>
<td>G55</td>
<td>Workpiece coordinate system 2</td>
<td>*G94</td>
<td>Feed per min.</td>
</tr>
<tr>
<td>G02</td>
<td>Circular/helical interpolation (CW)</td>
<td>G56</td>
<td>Workpiece coordinate system 3</td>
<td>G95</td>
<td>Feed per rotation</td>
</tr>
<tr>
<td>G03</td>
<td>Circular/helical interpolation (CCW)</td>
<td>G57</td>
<td>Workpiece coordinate system 4</td>
<td>*G98</td>
<td>Return to initial plane in canned cycle</td>
</tr>
<tr>
<td>G04</td>
<td>Dwell, exact stop</td>
<td>G58</td>
<td>Workpiece coordinate system 5</td>
<td>G99</td>
<td>Return to R point in canned cycle</td>
</tr>
<tr>
<td>G10</td>
<td>Tool life management</td>
<td>G59</td>
<td>Workpiece coordinate system 6</td>
<td>G110</td>
<td>Inner circle groove roughing (CCW)</td>
</tr>
<tr>
<td>G11</td>
<td>Tool life management end</td>
<td>G65</td>
<td>Macro program/ macro code</td>
<td>G111</td>
<td>Inner circle groove roughing (CW)</td>
</tr>
<tr>
<td>*G17</td>
<td>XY plane selection</td>
<td>G66</td>
<td>Macro program modal call</td>
<td>G112</td>
<td>Inner circle finishing (CCW)</td>
</tr>
<tr>
<td>G18</td>
<td>ZX plane selection</td>
<td>*G67</td>
<td>Macro program modal call cancel</td>
<td>G113</td>
<td>Inner circle finishing (CW)</td>
</tr>
<tr>
<td>G19</td>
<td>YZ plane selection</td>
<td>G73</td>
<td>High-speed peck drilling</td>
<td>G114</td>
<td>Circular outer finish milling (CW)</td>
</tr>
<tr>
<td>G20</td>
<td>Inch input</td>
<td>G74</td>
<td>Counter tapping cycle</td>
<td>G115</td>
<td>Outer circle finishing (CCW)</td>
</tr>
<tr>
<td>G21</td>
<td>Metric input</td>
<td>*G80</td>
<td>Canned cycle cancel</td>
<td>G134</td>
<td>Rectangular groove roughing (CCW)</td>
</tr>
<tr>
<td>G28</td>
<td>Reference position return</td>
<td>G81</td>
<td>Drilling cycle (spot drilling cycle)</td>
<td>G135</td>
<td>Rectangular groove roughing (CW)</td>
</tr>
<tr>
<td>G29</td>
<td>Return from reference position</td>
<td>G82</td>
<td>Drilling cycle (stepped hole boring cycle)</td>
<td>G136</td>
<td>Rectangular groove inner finishing (CCW)</td>
</tr>
<tr>
<td>G30</td>
<td>2nd, 3rd, 4th, reference position return</td>
<td>G83</td>
<td>Peck drilling cycle</td>
<td>G137</td>
<td>Rectangular groove inner finishing (CW)</td>
</tr>
<tr>
<td>G31</td>
<td>Skip function</td>
<td>G84</td>
<td>Tapping cycle</td>
<td>G138</td>
<td>Rectangular outer finishing (CCW)</td>
</tr>
<tr>
<td>*G40</td>
<td>Cutter compensation cancel</td>
<td>G85</td>
<td>Boring cycle</td>
<td>G139</td>
<td>Rectangular outer finishing (CW)</td>
</tr>
<tr>
<td>G41</td>
<td>Cutter compensation left</td>
<td>G86</td>
<td>Drilling cycle</td>
<td>G140</td>
<td>Rectangular continuous drilling (CW)</td>
</tr>
<tr>
<td>G42</td>
<td>Cutter compensation right</td>
<td>G88</td>
<td>Boring cycle</td>
<td>G141</td>
<td>Rectangular continuous drilling (CCW)</td>
</tr>
</tbody>
</table>
### Tool Length Compensation

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
<th>Code</th>
<th>Function</th>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>G43</td>
<td>Tool length compensation direction</td>
<td>G89</td>
<td>Boring cycle</td>
<td>G142</td>
<td>cc continuous drilling (CW)</td>
</tr>
<tr>
<td>G44</td>
<td>Tool length compensation direction</td>
<td>*G90</td>
<td>Absolute programming</td>
<td>G143</td>
<td>cc continuous drilling (CCW)</td>
</tr>
<tr>
<td>*G49</td>
<td>Tool length compensation cancel</td>
<td>G91</td>
<td>Incremental programming</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: mark “*” means initial state.

### PLC Codes List

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>LD</td>
<td>Normal open contact read</td>
</tr>
<tr>
<td>LDI</td>
<td>Normal closed contact read</td>
</tr>
<tr>
<td>OUT</td>
<td>Output coil</td>
</tr>
<tr>
<td>AND</td>
<td>Normal open contact in series</td>
</tr>
<tr>
<td>ORI</td>
<td>Normal closed contact in series</td>
</tr>
<tr>
<td>OR</td>
<td>Normal open contact in parallel</td>
</tr>
<tr>
<td>ORI</td>
<td>Normal closed contact in parallel</td>
</tr>
<tr>
<td>ORB</td>
<td>Serial block in parallel</td>
</tr>
<tr>
<td>ORB</td>
<td>Parallel block in series</td>
</tr>
<tr>
<td>ORB1</td>
<td>First level program end</td>
</tr>
<tr>
<td>ORB2</td>
<td>Second level program end</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>LD</td>
<td>Normal open contact read</td>
</tr>
<tr>
<td>LDI</td>
<td>Normal closed contact read</td>
</tr>
<tr>
<td>OUT</td>
<td>Output coil</td>
</tr>
<tr>
<td>AND</td>
<td>Normal open contact in series</td>
</tr>
<tr>
<td>ORI</td>
<td>Normal closed contact in series</td>
</tr>
<tr>
<td>OR</td>
<td>Normal open contact in parallel</td>
</tr>
<tr>
<td>ORI</td>
<td>Normal closed contact in parallel</td>
</tr>
<tr>
<td>ORB</td>
<td>Serial block in parallel</td>
</tr>
<tr>
<td>ORB</td>
<td>Parallel block in series</td>
</tr>
<tr>
<td>ORB1</td>
<td>First level program end</td>
</tr>
<tr>
<td>ORB2</td>
<td>Second level program end</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>LD</td>
<td>Normal open contact read</td>
</tr>
<tr>
<td>LDI</td>
<td>Normal closed contact read</td>
</tr>
<tr>
<td>OUT</td>
<td>Output coil</td>
</tr>
<tr>
<td>AND</td>
<td>Normal open contact in series</td>
</tr>
<tr>
<td>ORI</td>
<td>Normal closed contact in series</td>
</tr>
<tr>
<td>OR</td>
<td>Normal open contact in parallel</td>
</tr>
<tr>
<td>ORI</td>
<td>Normal closed contact in parallel</td>
</tr>
<tr>
<td>ORB</td>
<td>Serial block in parallel</td>
</tr>
<tr>
<td>ORB</td>
<td>Parallel block in series</td>
</tr>
<tr>
<td>ORB1</td>
<td>First level program end</td>
</tr>
<tr>
<td>ORB2</td>
<td>Second level program end</td>
</tr>
</tbody>
</table>

### 1.2 Program Execution

#### Execution Sequence

The current program can only be run in automatic mode. GSK980MDa cannot run more than 1 program at the same time, so only one program can be performed at a time. The cursor is ahead of the first block when a program is opened, and can be moved in EDIT mode. In automatic mode, when the machine is in stop state, the cycle start signal (key on the panel or external cycle start signal) enables the program to be run from the block where the cursor is located. Usually, blocks are executed in sequence programmed in advanced. Program stops running till M02 or M30 is executed. The cursor
moves along with program execution. The program execution sequence or state will be changed in following conditions:

- Program running stops when the key or the Emergency Stop button is pressed.

Program running stops when the CNC alarm or PLC alarm occurs.

- When the system is switched in DIT or MDI mode, program stops running after the current block is executed. After switching to automatic mode again, when the key on the panel is pressed or external cycle start signal is ON, the program runs from the block where the cursor is located.

- If the operation mode is switched to L/M/P MPG/ST/P/M/CHI/Z/RO R/T/R mode when the program is running, the execution dwells: after switching to automatic mode again, when the key on the panel is pressed or external cycle start signal is ON, the program runs from where it stops.

- The execution dwells when the key is pressed or external pause signal is cut off; program starts running from where it stops when the key on the panel is pressed or external cycle start signal is ON.

- The program dwells at the end of each block when the single block switch is ON: after pressing , key or switching on external cycle signal, program continuously runs from the next block.

- Blocks with mark is skipped when the skip switch is ON.

- The object block is executed when command G65 or macro program skip (GOTO) is specified.

- When M98 or M9000~M9999 command is performed, the corresponding subprogram or macro program is called. M99 is executed at the end of the subprogram or macro program, after returning to the main program, the subsequent block (the one after the block in which the subprogram is called) is executed. (return to a specified block, if it is commanded by M99).

- When M99 command is specified in the middle of a main program which is not called by other programs, the current program is repeatly executed after returning to the head of the program.

### Word Execution Sequence Within Block

When multiple words (such as G, X, Y, Z, F, R, M, S, T) are in one block, most of M, S, and T words are interpreted by CNC and sent to PLC for processing. Other words are processed by CNC directly. M98, M99, M9000~M9999 and S word (which specify the spindle speed in r/min, m/min) are directly processed by CNC as well.

When G words share the same block with M00, M01, M02 and M30, M words are executed after G words, and CNC sends corresponding signals to PLC for processing.

When the G words share the same block with the M98, M99, M9000~M9999, these M words are performed by CNC after G words (the M signal not sent to PLC).
When G words and M, S, T words share the same block, PLC program (ladder diagram) determines the execution consequence (executed at the same time or G words before M, S, T words). Refer to the manual from tool builder for relevant words execution sequence.

### 1.3 Basic Axes Increment System

The increment system consists of the least input increment (for input) and least command increment (for output). The least input increment is the minimum unit for programming moving distance. The least command increment is the minimum unit for moving the tool on the machine. Both increments are represented in mm, inches, or deg.

The basic axes herein means X, Y, Z axes. The basic increment system includes IS-B and IS-C types which can be selected by bit ISC of parameter NO.038.

<table>
<thead>
<tr>
<th>ISC</th>
<th>Increment System</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>IS-B(1U)</td>
</tr>
<tr>
<td>1</td>
<td>IS-C(0.1U)</td>
</tr>
</tbody>
</table>

In different increment system, different pulse output type enables different output speed. (Selected by bit ABPx of parameter NO.039)

<table>
<thead>
<tr>
<th>ABPx</th>
<th>Impulse Mode of Axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Impulse and direction.</td>
</tr>
<tr>
<td>1</td>
<td>AB phases</td>
</tr>
</tbody>
</table>

### 1.3.1 Speed of Increment Systems

<table>
<thead>
<tr>
<th>Output Mode</th>
<th>Metric System (mm/min)</th>
<th>Inch System (inch/min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pulse + direction</td>
<td>60,000</td>
<td>6,000</td>
</tr>
<tr>
<td>AB quadrature phase</td>
<td>240,000</td>
<td>24,000</td>
</tr>
</tbody>
</table>

### 1.3.2 Unit of Increment Systems

In different increment system, the least input/output increment varies with metric/inch system. The specific data is shown as follows.

<table>
<thead>
<tr>
<th>Unit</th>
<th>Least Input Increment</th>
<th>Least Command Increment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric machine</td>
<td>0.001 (mm)</td>
<td>0.001 (mm)</td>
</tr>
<tr>
<td>Inch machine</td>
<td>0.0001 (inch)</td>
<td>0.001 (inch)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit</th>
<th>Metric Input</th>
<th>Inch Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric machine</td>
<td>0.001 (mm)</td>
<td>0.0001 (inch)</td>
</tr>
<tr>
<td>Inch machine</td>
<td>0.001 (mm)</td>
<td>0.001 (deg)</td>
</tr>
</tbody>
</table>

In different increment system, different pulse output type enables different output speed.
<table>
<thead>
<tr>
<th>Increment System</th>
<th>Metric Input (G21)</th>
<th>Inch Input (G20)</th>
<th>Least Command Increment (for Output)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric Machine System</td>
<td>0.0001 (mm)</td>
<td>0.00001 (inch)</td>
<td>0.0001 (deg)</td>
</tr>
<tr>
<td>Inch Machine System</td>
<td>0.0001 (mm)</td>
<td>0.000001 (inch)</td>
<td>0.0001 (deg)</td>
</tr>
</tbody>
</table>

Least input increment (for input) is metric or inch can be set by G20 or G21. Least command increment (for output) is metric or inch is determined by machine tool and set by bit SCW of parameter NO.004.

### Data Range of Increment System

Limited by pulse output frequency, the data ranges may vary due to different increment system.

<table>
<thead>
<tr>
<th>Increment System</th>
<th>Coax and Data Input Range</th>
<th>Data Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1u (IS-B)</td>
<td>-99999.999 - 99999.999 (mm)</td>
<td>5.3</td>
</tr>
<tr>
<td></td>
<td>-99999.999 - 99999.999 (deg)</td>
<td>5.3</td>
</tr>
<tr>
<td></td>
<td>-9999.999 - 9999.999 (inch)</td>
<td>4.4</td>
</tr>
<tr>
<td></td>
<td>-9999.999 - 9999.999 (deg)</td>
<td>4.3</td>
</tr>
<tr>
<td>0.1u (IS-C)</td>
<td>-9999.999 - 9999.999 (mm)</td>
<td>4.4</td>
</tr>
<tr>
<td></td>
<td>-9999.999 - 9999.999 (deg)</td>
<td>4.4</td>
</tr>
<tr>
<td></td>
<td>-9999.999 - 9999.999 (inch)</td>
<td>3.5</td>
</tr>
<tr>
<td></td>
<td>-9999.999 - 9999.999 (deg)</td>
<td>3.4</td>
</tr>
</tbody>
</table>

**Note:** In the table above indicate 5 integers and 3 decimals. Other data are alike.

### Data Range and Unit Increment System

- **n**eed **a**address

  Machine tool types decide the units of linear axes speed, i.e. mm/min for metric machine system is; 0.1 inch/min for inch machine system.

  The range of linear axis speed parameter is codetermined by machine tool type and increment system.

  For example: data parameter NO.070 upper limit of cutting feedrate.
Chapter 1 Programming Fundamentals

- **Machine-tool type**
  - **Metric machine system**
    - Linear axis increment unit: 1 u (IS-B) for 0.1 u (IS-C)
    - Linear axis parameter range: 1 u (IS-B) for 0.1 u (IS-C)
    - Linear axis speed unit: mm/min for 0.1 inch/min
    - Rotation axis parameter range: 5~60000 (deg/min)
    - Rotation axis speed unit: deg/min

  - **Inch machine system**
    - Linear axis increment unit: 1 u (IS-B) for 0.1 u (IS-C)
    - Linear axis parameter range: 1 u (IS-B) for 0.1 u (IS-C)
    - Linear axis speed unit: 5~60000 (deg/min)
    - Rotation axis parameter range: 5~60000 (deg/min)
    - Rotation axis speed unit: deg/min

As rotary axes are not involved in metric-inch interconversion, the rotation speed unit is always deg/min.

The switch between different increment systems may cause the excess of permitted running speed set by data parameter. Therefore, at the first power-on after switching, the system automatically modifies relevant speed parameters and gives an alarm.

- **Increment parameter**
  - The unit and range of linear axis speed parameter are codetermined by machine tool type and increment system.
  - Example: parameter NO135: X axis software limit.

  - Metric, inch machine tool system
    - Linear axis increment unit: 1 u (IS-B) for 0.1 u (IS-C)
    - Linear axis parameter range: 1 u (IS-B) for 0.1 u (IS-C)
    - Linear axis speed unit: 0.001mm for 0.0001mm
    - Rotation axis parameter range: 0.001deg for 0.0001deg
    - Rotation axis speed unit: 0~99999.999 for 0~9999.9999

As rotary axes are not involved in metric-inch interconversion, the rotary axis increment parameter unit is determined by increment system types. The ranges of rotary axis increment parameters are the same as that of metric machine tool.

- **Coordinate data (G54~G59)**
  - The unit of linear axis coordinate data is determined by metric/inch input system, namely, mm for metric system, inch for inch system.
  - The ranges of linear axis coordinate data are codetermined by metric/inch input system and increment system. It is the same as command data input ranges. Shown as follows:
**Volume I  Programming**

**Increment system  Linear axis coordinate data range**

<table>
<thead>
<tr>
<th>Input type</th>
<th>Increment system</th>
<th>Linear axis coordinate data range</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 u (IS-B)</td>
<td>Metric input (G21)</td>
<td>-99999.9999 to 99999.9999 (mm)</td>
</tr>
<tr>
<td></td>
<td>Inch input (G20)</td>
<td>-9999.9999 to 9999.9999 (inch)</td>
</tr>
<tr>
<td>0.1u (IS-C)</td>
<td>Metric input (G21)</td>
<td>-9999.9999 to 9999.9999 (mm)</td>
</tr>
<tr>
<td></td>
<td>Inch input (G20)</td>
<td>-999.9999 to 999.9999 (inch)</td>
</tr>
</tbody>
</table>

As rotary axis is not involved in metric-inch interconversion, the unit of rotary axis coordinate data is deg. The ranges of rotary axis coordinate data is the same as linear axis coordinate data ranges in metric system.

<table>
<thead>
<tr>
<th>Input type</th>
<th>Increment system</th>
<th>Rotary axis coordinate data range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric, inch input</td>
<td>1 u (IS-B)</td>
<td>-99999.9999 to 99999.9999 (deg)</td>
</tr>
<tr>
<td></td>
<td>0.1u (IS-C)</td>
<td>-9999.9999 to 9999.9999 (deg)</td>
</tr>
</tbody>
</table>

- **Tool compensation data**
  
  The unit of tool compensation data is determined by metric/inch input system, namely, mm for metric input, inch for inch input.

  The range of tool compensation data is limited as 9999999, determined by inch input system and increment system. It is smaller than command data. Shown as follows:

<table>
<thead>
<tr>
<th>Input type</th>
<th>Increment system</th>
<th>Tool compensation data unit</th>
<th>Tool compensation data range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric input (G21)</td>
<td>1 u (IS-B)</td>
<td>mm</td>
<td>-99999.9999 to 99999.9999</td>
</tr>
<tr>
<td></td>
<td>0.1u (IS-C)</td>
<td></td>
<td>-9999.9999 to 9999.9999</td>
</tr>
<tr>
<td>Metric input (G21)</td>
<td>1 u (IS-B)</td>
<td>inch</td>
<td>-99999.9999 to 99999.9999</td>
</tr>
<tr>
<td></td>
<td>0.1u (IS-C)</td>
<td></td>
<td>-999.9999 to 999.9999</td>
</tr>
</tbody>
</table>

- **Screw-pitch error compensation data**
  
  The unit and range of linear axis screw-pitch error compensation data is codetermined by machine tool type and increment system.

  Shown as following table:

<table>
<thead>
<tr>
<th>Input type</th>
<th>Increment system</th>
<th>Tool compensation data unit</th>
<th>Tool compensation data range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric input (G21)</td>
<td>1 u (IS-B)</td>
<td>mm</td>
<td>-99999.9999 to 99999.9999</td>
</tr>
<tr>
<td></td>
<td>0.1u (IS-C)</td>
<td></td>
<td>-9999.9999 to 9999.9999</td>
</tr>
<tr>
<td>Metric input (G21)</td>
<td>1 u (IS-B)</td>
<td>inch</td>
<td>-99999.9999 to 99999.9999</td>
</tr>
<tr>
<td></td>
<td>0.1u (IS-C)</td>
<td></td>
<td>-999.9999 to 999.9999</td>
</tr>
</tbody>
</table>
### Rotary Axes and Screw-Pitch Error Compensation

Rotary axes are not involved in metric-inch conversion. The unit of rotary axes screw-pitch error compensation is determined by increment system. The range is the same as that of the metric machine tool.

#### Metric Tool Machine System

<table>
<thead>
<tr>
<th>Unit</th>
<th>Screw-Pitch Error Compensation Data Unit</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric tool</td>
<td></td>
<td>-255 ~ 255</td>
</tr>
<tr>
<td>1 u (IS-B)</td>
<td></td>
<td>0.001mm</td>
</tr>
<tr>
<td>0.1 u (IS-C)</td>
<td></td>
<td>0.0001mm</td>
</tr>
</tbody>
</table>

#### Inch Tool Machine System

<table>
<thead>
<tr>
<th>Unit</th>
<th>Screw-Pitch Error Compensation Data Unit</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inch tool</td>
<td></td>
<td>-255 ~ 255</td>
</tr>
<tr>
<td>1 u (IS-B)</td>
<td></td>
<td>0.0001 inch</td>
</tr>
<tr>
<td>0.1 u (IS-C)</td>
<td></td>
<td>0.00001 inch</td>
</tr>
</tbody>
</table>

### Graphics Setting Data

The maximum and minimum data ranges of X, Y, Z set by graph is in accordance with the command data ranges.

#### Metric Input (G21)

<table>
<thead>
<tr>
<th>Code</th>
<th>μ (μm)</th>
<th>μ (μ”C)</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>0.001</td>
<td>500.000</td>
<td>mm/pitch/lead</td>
</tr>
<tr>
<td>I</td>
<td>0.06</td>
<td>2540</td>
<td>Pitch/lead/inch</td>
</tr>
</tbody>
</table>

#### Inch Input (G20)

<table>
<thead>
<tr>
<th>Code</th>
<th>μ (μ”)</th>
<th>μ (μ”C)</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>0.0001</td>
<td>50.00</td>
<td>inch/pitch/lead</td>
</tr>
<tr>
<td>I</td>
<td>0.06</td>
<td>2540</td>
<td>Pitch/lead/inch</td>
</tr>
</tbody>
</table>

### Definition and Range of the Pitch

- **Pitch Definition**
  
  G94: feed per minute, F unit: mm/min
  
  G95: feed per rotation, F definition and ranges are as follows:
1.4 Additional Axes Increment System

In the least increment system (IS-B or IS-C), under the condition that the additional axes are not involved in simultaneous control and just used for separate motion (such as feeding), and the requirement for precision is not high, when the least increment is 0.01, the feedrate will be much faster, greatly increasing the efficiency. Therefore, the additional axes least increment system is not necessary to be in accordance with the current least increment system. To meet various requirements of users, the system adds optional function to least increment system.

Additional axes increment system is set by state parameter No.026, No.028. Shown as follows:

<table>
<thead>
<tr>
<th>No.</th>
<th>Increment System of 4th</th>
<th>Inc. System of 5th</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Same to the X, Y, Z</td>
<td>Same to the X, Y, Z</td>
</tr>
<tr>
<td>0</td>
<td>IS-A 0.01</td>
<td>IS-A 0.01</td>
</tr>
<tr>
<td>1</td>
<td>IS-B 0.001</td>
<td>IS-B 0.001</td>
</tr>
<tr>
<td>1</td>
<td>IS-C 0.0001</td>
<td>IS-C 0.0001</td>
</tr>
</tbody>
</table>

Note: The least input/output in the table above are described without considering the metric/inch system and rotation axes.

1.4.1 Additional Axes in Current Increment System

When IS-B or IS-C is selected, the speed and range of additional axes are the same as described in 1.3.

1.4.2 Additional Axes in IS-A Increment System

When IS-A is selected, the maximum speed of additional axes can reach 100 times of that of IS-B and IS-C. The relevant data and parameters ranges are the same as that of the current basic axes increment system. (Refer to section 1.3)
2.1 M Codes (Miscellaneous Function)

The M codes are composed by code address M and 1~2 or 4 digits after the codes M is used for controlling the program execution or outputting M code to PLC.

![M Codes Value](image)

M98, M99 and M9000~M9999 are independently processed by CNC, and the M codes are not output to PLC.

The function of M29 is fixed, namely, to output M codes to PLC.

The M02 and M03 are defined as program END codes by NC, meanwhile it also outputs M codes to PLC for the I/O control (spindle OFF, cooling OFF control etc.).

The PLC program can not change the meaning of the above-mentioned codes when the M98, M99 and M9000~M9999 are regarded as program CALL codes and the M02 and M30 are regarded as program END codes. The codes of other M codes are all output to PLC program for specifying the code function; please refer to the manual issued by machine tool manufacturer.

One block only has one M code. The CNC alarm occurs when two or more M codes are existed in one block.

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>M02</td>
<td>End-of-Run</td>
</tr>
<tr>
<td>M29</td>
<td>Rigid tapping designation</td>
</tr>
<tr>
<td>M30</td>
<td>End-of-Run</td>
</tr>
<tr>
<td>M98</td>
<td>Subprogram call</td>
</tr>
<tr>
<td>M99</td>
<td>Return from the subprogram; the program will be repeatedly executed. If the code M99 is used for main program ending (namely, the current program is not called by other programs).</td>
</tr>
<tr>
<td>M9000~M9999</td>
<td>Call macro program (Program No. is larger than 9000)</td>
</tr>
</tbody>
</table>

### End of Program (M02)

**Format:** M02

**Function:** The M02 code is executed in the Auto mode. The automatic run is ended after the other codes of current block are executed: the cursor stops in the block in which the M02 is located and does not return to the head of the program. If the program is to be executed again, the cursor should return to the beginning of the program.

Besides the above-mentioned functions processed by CNC, the functions of code M02 also can be defined by the PLC ladder diagram. The function defined by standard ladder diagram can be that the current input state of CNC is not change after the code M02 is executed.

### Rigid Tapping Designation (M29)

**Format:** M29

**Function:** In auto mode, after the execution of M29, the G74, G84 that followed is processed as...
rigid tapping codes.

### End of Run (M30)

**Format:** M30

**Function:** If the M30 command is executed in the Auto mode, the automatic run is ended after the other commands of the current block are executed; the system cancels the tool nose radius compensation and the cursor returns to the beginning of the program when the workpieces number is added by one (whether the cursor returns to the head of the program is determined by parameters).

The cursor does not return to the beginning of the program when the BIT4 of parameter No.005 is set to 0; when it is set to 1, the cursor returns to the beginning of the program as soon as the program execution is finished.

Besides the above-mentioned functions processed by CNC, the functions of code M30 also can be defined by the PLC ladder diagram. The function defined by standard ladder diagram can be turn OFF the M03, M04 or M08 output signal after the M30 command is executed, and meanwhile output M05 signal.

### Subprogram Call (M98)

**Format:** M98 P

**Function:** In Auto mode, when the M98 is executed, the subprogram specified by P is called after the execution of other codes in the current block. The subprogram can be performed 9999 times at most. M98 cannot be performed in MDI, or an alarm will occur.

### Return from Subprogram (M99)

**Format:** M99 P

**Function:** (in subprogram) as the other commands of the current block are executed, the block specified by P is performed continuously when the main program is returned. The next block is performed continuously by calling current subprogram of M98 command when returning to the main program because of the P is not given. If the main program is ended by using the M99 (namely, the current program is not called by other programs for execution), the current program will be run circularly. So, the M99 command is disabled in MDI.

**Example:** Fig. 2-1 shows that the execution route of the subprogram is called (the P command within M99). Fig. 2-2 shows that the execution route of the subprogram is called (the P command is not in M99.)
This program can call a quadruple subprogram, namely, the other subprogram can be called from the subprogram. (See Fig. 2-3)
2.1.6 Macro program call (M9000–M9999)

*Format*

\[ \text{M} \text{xFFFF} \]

\[ \text{O9000} \quad \text{O9999} \]

Command function: Call the macro program which is corresponded by the command value.

Macro program: Program 09000 ~ 09999 is special space obligated for the machine tool manufacturer for using editing and achieving special function subprogram, which is called macro program. Two-level operation authority is needed when editing the program 09000 ~ 09999, the user cannot modify or run the macro program but the macro calling command if his authority is 3 ~ 5 level. So the M9000 ~ M9999 commands are invalid in MDI mode.

2.1.7 M command defined by standard PLC ladder diagram

- M commands other than the above mentioned commands (M02, M30, M98, M99, M9000 ~ M9999) are defined by PLC. The M commands are defined by standard PLC hereinafter. This GSK980 MDa milling machine is used for machine control. About the function, meaning, control time sequence and logic etc. of the M command, refer to the manual issued by the machine tool builder.

<table>
<thead>
<tr>
<th>Command</th>
<th>Function</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>M00</td>
<td>Program pause</td>
<td></td>
</tr>
<tr>
<td>M03</td>
<td>Spindle CCW</td>
<td></td>
</tr>
<tr>
<td>M04</td>
<td>Spindle CW</td>
<td></td>
</tr>
<tr>
<td>M05</td>
<td>Spindle stop</td>
<td>Function interlock, state hold</td>
</tr>
<tr>
<td>M08</td>
<td>Cooling on</td>
<td></td>
</tr>
<tr>
<td>M09</td>
<td>Cooling off</td>
<td>Function interlock, state hold</td>
</tr>
<tr>
<td>M32</td>
<td>Lubricating on</td>
<td></td>
</tr>
<tr>
<td>M33</td>
<td>Lubricating off</td>
<td>Function interlock, state hold</td>
</tr>
</tbody>
</table>

Note: The command with “*” specified by standard PLC is valid when the power is on.

2.1.8 Program stop M00

Format: M00

Command function: the program is stopped after executing the M00 command, the “pause” is displayed; the program will continue when the key of Cycle Start is pressed.

2.1.9 Spindle CCW, CW, stop control (M03, M04 and M05)

*Format*:

\[ \text{M} \text{03} ; \]
\[ \text{M} \text{04} ; \]
\[ \text{M} \text{05} ; \]

Command function: M03: Spindle forward rotation (CCW); M04: spindle reverse rotation (CW); M05: spindle stop.
2.1.10 Cooling control (M08, M09)

For format:
M08;
M09;

Command function:
M08: cooling on;
M09: cooling off.

Note: The control time sequence and logic of M08 and M09 are specified by standard PLC.

2.1.11 Lubricating control (M32, M33)

Format:
M32;
M33;

Command function:
M32: lubricating on;
M33: lubricating off.

Note: The control time sequence and logic of M32 and M33 are specified by standard PLC.

2.2 Spindle Function

The spindle speed is controlled by the command, there are two ways to control spindle speed for GSK980MDa.

Spindle speed switching value control: the \( S \_2 \_2 \) (2-digit command value) command is processed by the PLC program for exporting the switching value signal to the machine, so that the step speed change of the spindle is achieved.

Spindle speed analog voltage control: the actual spindle speed is specified by the \( S \_4 \_4 \_4 \_4 \) (4-digit command value), the NC output the 0~10V analog voltage signal to the spindle servo device or inverter for achieving the stepless speed regulating of the spindle.

2.2.1 Spindle Speed Switch Value Control

When the spindle speed is controlled in the spindle speed switching value control, the BIT 4 of the bit parameter NO.001 is set to 0. One block only has one command. The CNC alarm occurs when two or more S commands are displayed in the block.

When the S command shares the same block with the command word, the performance sequence is defined by the PLC program. For details, refer to the manual issued by the machine tool builder.

This GSK980MDa milling machine is used for machining control when the spindle speed switching value is controlled. The time sequence and logic for S command should be referred to the manual issued by the machine tool builder. The following S command is defined by GSK980MDa standard PLC, for reference only.
2.2.2 Spindle speed analog voltage control

Spindle speed is analog voltage control when the BIT 4 of the current bit parameter is set to 1. For format S0000 0000 1111 (leading zero can be omitted), spindle speed analog voltage control command function: the CNC outputs 0 ~ 10V analog voltage to control the spindle servo or inverter for achieving the stepless speed regulating of the spindle when the spindle speed is set. The S command value is not stored when the power is turned off; and then the parameter recovers to 0 when the power is turned on.

The CNC owns four mechanical spindle shifts function. Counting the corresponding analog voltage value specified by the speed based upon the current set value (corresponding to data parameter No. 101 ~ No. 104) of the top speed (output analog voltage is 10V) of the spindle shift when the S command is performed, then output the voltage value to spindle servo or inverter, so that the consistency of actual speed and required speed of the spindle are controlled.

The analog output voltage is 0V when the CNC is switched on. The output analog voltage value is variable (unless the cutting feed in constant linear speed control and the absolute value of X axis absolute coordinate value are changed) after the S command is executed. The analog output voltage is 0V when the command S0 is executed. And the analog output voltage value is invariant when the CNC is reset or at emergency stop.

The parameter related to spindle speed analog voltage control:
2.2.3 Spindle override

The spindle actual speed can be modified by using spindle override when the spindle speed analog voltage control is effective. The actual speed modified by spindle override is limited by the top speed of current spindle shift, and it is also controlled by the lowest spindle limit value and the top spindle limit value in constant linear control mode.

This NC offers 8-level spindle override (50% ~ 120%, the change is 10% per level). The actual level and the modulating mode of the spindle override are defined by PLC ladder diagram. Refer to the manual issued by the machine tool builder when attempting to use it. The following description is GSK980MDa standard PLC ladder diagram function, for reference only.

The spindle override defined by GSK980MDa standard PLC ladder diagram has 8 levels. The spindle actual real-time speed can be adjusted by using the spindle override key in the command speed range of 50% ~ 120%, the spindle override will be memorized when the power is turned off. Refer to this manual for modification operation of the spindle override.

2.4 Feeding Function

2.4.1 Cutting feed (G94/G95, F command)

For format: G94 F_; (F 0.0001 ~ F 8000, leading zero can be omitted, for feed rate per minute, mm/min).

Command function: The cutting feed rate is specified by mm/min, G94 is modal G command. If the current mode is G94 that it needs no G94 anymore.

For format: G95 F_; (F 0.0001 ~ F 500, leading zero can be omitted).

Command function: The cutting feed rate is offered by the unit of mm/rev. G95 is modal G command. The G95 command can be omitted if the current mode is G95. When the CNC performs G95 F_, the cutting feed rate is controlled by feed rate command based on the multiplication of F command value (mm/rev) and current spindle speed (rev/min). The actual feed rate varies with the spindle speed. The spindle cutting feed rate per revolution is specified by G95 F_, the even cutting line can be formed on the surface of workpiece. It is necessary to install spindle encoder when the G95 mode is operated.

The G94 and G95 are modal G commands at the same group, one of them is available only.
The G94 is initial state command, so, it defaults the G94 when the CNC is switched on. The following below shows the conversion formula of feed value per rev. and feed value per min:

\[ F_m = F_r \times S \]

There into: \( F_m \): feed value per minute (mm/min);
\( F_x \): feed value per revolution (mm/rev);
\( S \): spindle speed (r/min).

The feed rate value is set by the CNC Data parameter No. 172 when the CNC is switched on, the \( F \) value is invariable after the \( F \) command is executed. The feed rate is 0 after \( F \) 0 is executed. The \( F \) value is invariable when CNC is reset or at emergency stop.

Note: In G95 mode, the cutting feedrate will be uneven when the spindle speed is less than 1 rev./min. The following error will exist in the actual feedrate when the spindle speed vibration occurs.

To guarantee the machine quality, it is recommended that the spindle speed selected in machining is not less than the lowest speed of available torque exported by spindle servo or inverter.

Cutting feed: The CNC makes tool movement path and the path (linear or circular arc) defined by command into consistency (circular interpolation can be performed by two axes in selected plane when it is circular arc, the helical interpolation is formed by third axis linear interpolation link), by which, the CNC controls three directions movement for X axis, Y axis, Z axis, 4th axis and 5th axis at the same time. The instantaneous speed of movement path in a tangential direction is consistent with the \( F \) command value, so this is called CUTTING FEED or INTERPOLATION. The cutting feed rate is supplied by \( F \) command, which is assembled to each interpolation axis according to the programming path when the CNC performs the interpolation command (cutting feed).

Linear interpolation: The CNC can control the instantaneous speed in the directions of X axis, Y axis, Z axis, 4th axis and 5th axis, so the vector resultant speed in these five directions are equal to the \( F \) command value.

\[
\begin{align*}
    f_x &= \frac{d_x}{\sqrt{d_x^2 + d_y^2 + d_z^2 + d_4^2 + d_5^2}} \cdot F \\
    f_y &= \frac{d_y}{\sqrt{d_x^2 + d_y^2 + d_z^2 + d_4^2 + d_5^2}} \cdot F \\
    f_z &= \frac{d_z}{\sqrt{d_x^2 + d_y^2 + d_z^2 + d_4^2 + d_5^2}} \cdot F \\
    f_4 &= \frac{d_4}{\sqrt{d_x^2 + d_y^2 + d_z^2 + d_4^2 + d_5^2}} \cdot F \\
    f_5 &= \frac{d_5}{\sqrt{d_x^2 + d_y^2 + d_z^2 + d_4^2 + d_5^2}} \cdot F \\
    f_6 &= \frac{d}{\sqrt{d_x^2 + d_y^2 + d_z^2 + d_4^2 + d_5^2}} \cdot F
\end{align*}
\]
Chapter 2 MST Codes

Volume I Programming

The vector resultant speed for the instantaneous speed in X, Y and Z axis directions. The dx is instantaneous increment of the X axis, the fx is instantaneous speed of X axis. The dy is instantaneous increment of Y axis, the fy is instantaneous speed of Y axis. The dz is instantaneous increment of Z axis, the fz is instantaneous speed of Z axis. The d4 is instantaneous increment of 4th axis, the f4 is instantaneous speed of 4th axis. The d5 is instantaneous increment of 5th axis, the f5 is instantaneous speed of 5th axis.

Circular interpolation (helical interpolation): Performing the arc interpolation in selected plane, the third axis performs linear interpolation, so the F value is circular interpolation speed. An interpolation of linear and circular arc has the following relation when the linear interpolation speed is f:

\[ f = F \times \frac{\text{linear axis length}}{\text{circular arc length}} \]

Circular interpolation (helical interpolation): Performing the arc interpolation in selected plane, the third axis performs linear interpolation, so the F value is circular interpolation speed. An interpolation of linear and circular arc has the following relation when the linear interpolation speed is f:

\[ f = F \times \frac{\text{linear axis length}}{\text{circular arc length}} \]

Feedrate along the circle between 2 arc interpolation axes is the specified one.

Circular interpolation (helical interpolation): Performing the arc interpolation in selected plane, the third axis performs linear interpolation, so the F value is circular interpolation speed. An interpolation of linear and circular arc has the following relation when the linear interpolation speed is f:

\[ f = F \times \frac{\text{linear axis length}}{\text{circular arc length}} \]

Feedrate along the circle between 2 arc interpolation axes is the specified one.
2.4.2 Manual feed

<table>
<thead>
<tr>
<th>Feedrate override(%)</th>
<th>0</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
<th>110</th>
<th>120</th>
<th>130</th>
<th>140</th>
<th>150</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual feedrate (mm/min)</td>
<td>2.0</td>
<td>3.2</td>
<td>5.0</td>
<td>7.9</td>
<td>12.6</td>
<td>20</td>
<td>32.5</td>
<td>50</td>
<td>79</td>
<td>126</td>
<td>200</td>
<td>320</td>
<td>500</td>
<td>790</td>
<td>1260</td>
<td></td>
</tr>
</tbody>
</table>

Note: The manual feedrate of axis is diameter variation per minute. The feedrate defined by a standard PLC ladder diagram is memorized when the power is turned off.
2.4.4 Automatic acceleration or deceleration

This GSK980MDa performs automatic acceleration or deceleration in order to achieve a smooth transition of the speed at the beginning of the axis movement or before the movement stops; this will diminish the impact when the movement is started or stopped. This GSK980MDa adopts kinds of acceleration or deceleration as follows:

- Rapid traverse: linear type front acceleration or deceleration
- Cutting feed: exponential type rear acceleration or deceleration
- Manual feed: exponential type rear acceleration or deceleration
- MPG feed: exponential type rear acceleration or deceleration
- Step feed: exponential type rear acceleration or deceleration

When the cutting feed is performed, this GSK980MDa adopts exponential rear acceleration or deceleration.
d e c e l e r a t i o n , a n d a r c t r a n s i t i o n w i l l b e f o r m e d f o r t h e a c c e l e r a t i o n o r d e c e l e r a t i o n a t t h e m e e t i n g p o i n t o f t h e p a t h f o r t h e a d j a c e n t t w o c u t t i n g f e e d b l o c k s , w h e n t h e B I T 5 o f t h e b i t p a r a m e t e r N o . 0 0 7 i s s e t t o 0 . A c o n t o u r e r r o r e x i s t s b e t w e e n t h e a c t u a l t o o l p a t h a n d t h e p r o g r a m m e d p a t h w h e n t h e p o s i t i o n i n g i s n o t e n o u g h a c c u r a t e a t t h e m e e t i n g p o i n t o f t h e t w o p a t h s .

I n o r d e r t o a v o i d t h i s k i n d o f e r r o r , t h e e x a c t s t o p c o m m a n d ( G0 4 ; ) c a n b e i n s e r t e d b e t w e e n t h e t w o b l o c k s o r t h e B I T 5 o f t h e C N C b i t p a r a m e t e r N o . 0 0 7 i s s e t t o 1 . N o w , t h e p r e v i o u s b l o c k i s d e c e l e r a t e d t o z e r o s p e e d a n d i t i s p o s i t i o n e d t o t h e e n d o f t h e b l o c k , a n d t h e n t h e n e x t c u t t i n g f e e d b l o c k i s p e r f o r m e d . T h e f o l l o w i n g b l o c k c a n b e p e r f o r m e d b e c a u s e e a c h b l o c k i s a c c e l e r a t i n g f r o m t h e i n i t i a l s p e e d a n d t h e n d e c e l e r a t i n g t o z e r o a t l a s t . I f t h e p r o g r a m t i m e i s i n c r e a s i n g , i t m a y c a u s e t h e l o w e r m a c h i n i n g e f f i c i e n c y .

T h e S M Z o f b i t p a r a m e t e r N o . 0 0 7 i s s e t t o 0 , t h e t r a n s i t i o n b e t w e e n t w o a d j a c e n t b l o c k s i s p r o c e s s e d a c c o r d i n g t o t h e t a b l e 2 - 3 .

<table>
<thead>
<tr>
<th>Previous block</th>
<th>Rapid Position</th>
<th>Cutting feed</th>
<th>Without move</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rapid positioning</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Cutting feed</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Without move</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
</tbody>
</table>

Note: ✗: The subsequent block is performed after the previous block is accurately positioned at the end of the block.

✗: Each axis speed is transmitted according to the acceleration or deceleration between the adjacent blocks. An arc transition is formed at the meeting point of the tool path.

![Table 2-3](image)
Chapter 3 - G Command

3.1 G Command Brief

The G command is composed by the command address G and the 1 to 3 digit command value after the command G. Many kinds of operations are specified such as tool movement relative to workpiece, coordinate set, etc. See Table 3-1 for G commands.

<table>
<thead>
<tr>
<th>Command word</th>
<th>Group</th>
<th>Function</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>G00</td>
<td>00</td>
<td>Initial G command</td>
<td>Rapide traverse</td>
</tr>
<tr>
<td>G01</td>
<td>01</td>
<td>Linear interpolation</td>
<td></td>
</tr>
<tr>
<td>G02</td>
<td>02</td>
<td>Circular interpolation (CW)</td>
<td></td>
</tr>
<tr>
<td>G03</td>
<td>03</td>
<td>Circular interpolation (CCW)</td>
<td></td>
</tr>
<tr>
<td>G73</td>
<td>01</td>
<td>Peck drilling cycle</td>
<td></td>
</tr>
<tr>
<td>G74</td>
<td>01</td>
<td>Left-hand (counter) tapping cycle</td>
<td></td>
</tr>
<tr>
<td>G80</td>
<td>01</td>
<td>Canned cycle cancellation</td>
<td></td>
</tr>
<tr>
<td>G81</td>
<td>01</td>
<td>Drilling cycle (spot drilling cycle)</td>
<td></td>
</tr>
<tr>
<td>G82</td>
<td>01</td>
<td>Drilling cycle (counter bore cycle)</td>
<td></td>
</tr>
<tr>
<td>G83</td>
<td>01</td>
<td>Peck drilling cycle</td>
<td></td>
</tr>
<tr>
<td>G84</td>
<td>01</td>
<td>Tapping cycle</td>
<td></td>
</tr>
<tr>
<td>G85</td>
<td>01</td>
<td>Boring cycle</td>
<td></td>
</tr>
<tr>
<td>G86</td>
<td>01</td>
<td>Drilling cycle</td>
<td>Modal G command</td>
</tr>
<tr>
<td>G88</td>
<td>01</td>
<td>Boring cycle</td>
<td></td>
</tr>
</tbody>
</table>

Table 3-1 G command word list
GSK9 80 MDa Milling CNC System User Manual
Volume I  Programming

28

G89 Boring cycle
G110 Circular groove inner rough-milling CW
G111 Circular groove inner rough-milling CCW
G112 Circular groove inner finish-milling CW
G113 Circular groove inner finish-milling CCW
G114 Excircle finish-milling CW
G115 Excircle finish-milling CCW
G134 Rectangle groove rough-milling CW
G135 Rectangle groove rough-milling CCW
G136 Rectangle groove inner finish-milling CW
G137 Rectangle groove inner finish-milling CCW
G138 Rectangle outer finish-milling CW
G139 Rectangle outer finish-milling CCW

G17 (initial G command) XY plane selection
G18 ZX plane selection
G19 YZ plane selection

Modal G command

G90 (initial G command) Absolute programming
G91 Relative programming

G94 (initial G command) Feed per minute
G95 Feed per revolution

G20 Data inch input
G21 Data metric input

Modal power down memory

G40 (initial G command) Tool nose radius compensation cancellation
G41 Tool nose radius compensation left
G42
G43 Tool length offset in + direction
G44 Tool length offset in - direction
G49 (initial G command)

G140 Rectangle path serially punch CW
G141 Rectangle path serially punch CCW
G142 Arc path serially punch CW
G143 Arc path serially punch CCW

Non-modal G command

G98 (initial G command) Return to initial plane in canned cycle
G99

G67 (initial G command) Macro program call
G66

G54 (initial G command) Work piece coordinate system 1
G55 Work piece coordinate system 2
G56 Work piece coordinate system 3
G57 Work piece coordinate system 4

Modal G command
3.1.1 Modal, non-modal and initial state

The G commands can be set to 12 groups such as 00, 01, 02, 03, 05, 06, 07, 08, 09, 10, 12 and 14. Therefore, G commands of 00 group are non-modal G commands, those of other groups are modal commands. G00, G80, G40, G49, G67 and G94 are initial G commands.

After the G command is executed, the function defined or status is valid until it is changed by other G command in the same group, this kind of command is called modal G command.

After this G command is performed and before the function defined or status is changed, this G command need not be input again when the next block performs this G command.

After the G command is performed, the function defined or status is valid for once, the G command word should be input again while every time the G command is performed, this kind of command is called non-modal G command.

The modal G command is valid without performing its function or status after the system is powered on, this is called initial G command. If the G command is not introduced after the power is turned on, then the initial G command is executed. The initial commands of GSK980MDa are G00, G80, G40, G49, G67 and G94.

3.1.2 Examples

Example 1

O0001
G17 G0 X100 Y100; (Move to G17 plane X100 Y100 at the rapid traverse rate; modal command G0 and G17 valid)
X20 Y30; (Move to X20 Y30 at the rapid traverse rate; modal command G0 can be omitted)
G1 X50 Y50 F300; (Linear interpolation to X50 Y50, feedrate is 300mm/min; modal command G1 valid)
X100; (Linear interpolation to X100 Y50, feedrate is 300mm/min; the Y coordinate is not input, use current value Y50; keep F300, the modal command G01 can be omitted)
G0 X0 Y0; (Move to X0 Y0 at the rapid traverse rate, modal G command G0 valid)
M30;

Example 2

O0002:
G0 X50 Y5; (Move to X50 Y5 at the rapid traverse rate)
G04 X4; (Time delay for 4 seconds)
G04 X5; (Time delay again for 5 seconds non-modal command G04 should be
input again).

M30:

Example 3 (the first operation after the power is turned on) O0003:

G0 G4 G01 X100 Y100 F500; (G4 feed rate is 500mm/min)
G5 G5 G01 X10 F0.01; (G5 feed rate per revolution, input the F value again)
G0G0O X0 Y50;
M30:

3.1.3 Related definition

The words or characters which are not specially described in this manual are as follows:

**Start point**: the position before performing the current block;

**End point**: the position after performing of the current block;

**X**: the end point absolute coordinate of X axis for G0, the incremental value of X axis against current point for G1;

**Y**: the absolute coordinate of Y axis at the end for G0, the incremental value of Y axis against current point for G1;

**Z**: the absolute coordinate of Z axis at the end for G0, the incremental value of Z axis against current point for G1;

**F**: cutting feed rate.

3.1.4 Address definition

Usage of the address in system is as follows:

<table>
<thead>
<tr>
<th>Address</th>
<th>Function</th>
<th>Value range</th>
<th>Rounding</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Punching number of 1 and 3rd side for rectangle serial punch (G140/G141)</td>
<td>-nnnnnnnnnnnnn nnnnnnnnnnnnnn</td>
<td>Decimal part omitted</td>
</tr>
<tr>
<td></td>
<td>4th, 5th axis, axis name address</td>
<td>-nnnnnnnnnnnnn</td>
<td>Round-off</td>
</tr>
<tr>
<td>B</td>
<td>Punching number of 2nd and 4th side for rectangle serial punch (G140/G141)</td>
<td>-nnnnnnnnnnnnn nnnnnnnnnnnnnn</td>
<td>Decimal part omitted</td>
</tr>
<tr>
<td></td>
<td>Radius for arc serially punch (G142/G143)</td>
<td>-nnnnnnnnnnnnn nnnnnnnnnnnnnn</td>
<td>Round-off</td>
</tr>
<tr>
<td></td>
<td>4th, 5th axis, axis name address</td>
<td>-nnnnnnnnnnnnn</td>
<td>Round-off</td>
</tr>
<tr>
<td>C</td>
<td>Punching number for arc serially punch (G142/G143)</td>
<td>-nnnnnnnnnnnnn nnnnnnnnnnnnnn</td>
<td>Decimal part omitted</td>
</tr>
<tr>
<td></td>
<td>4th, 5th axis, axis name address</td>
<td>-nnnnnnnnnnnnn</td>
<td>Round-off</td>
</tr>
<tr>
<td>D</td>
<td>Tool radius offset number</td>
<td>0~32</td>
<td>Decimal</td>
</tr>
</tbody>
</table>
### Chapter 3 G Command

#### Volume I Programming

<table>
<thead>
<tr>
<th>E</th>
<th>Unused</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>F</th>
<th>G code in G system</th>
</tr>
</thead>
<tbody>
<tr>
<td>G9 4 feed per minute</td>
<td>0.15000</td>
</tr>
<tr>
<td>G9 5 feed per rotation</td>
<td>0.0001-500</td>
</tr>
<tr>
<td>Tooth pitch in G74,G84 (unit: G21, mm/r; G20, inch/r)</td>
<td>0.001-500</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>I</th>
<th>Length offset number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation command in G5</td>
<td>0.32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>J</th>
<th>Distance from arc start point to center point in X direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>G110,G115: radius value of circle</td>
<td></td>
</tr>
<tr>
<td>G134,G135: width of rectangle in X direction</td>
<td></td>
</tr>
<tr>
<td>G74,G84: inch screw (unit: tooth/inch)</td>
<td>0.0 25400</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>K</th>
<th>Distance from arc start point to center point in Y direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>G112,G113: distance from start point to center point</td>
<td></td>
</tr>
<tr>
<td>G114,G115: distance from start point to circle</td>
<td></td>
</tr>
<tr>
<td>G134,G135: width of rectangle in Y direction</td>
<td></td>
</tr>
<tr>
<td>G140,G141: length of 2nd side of rectangle</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>L</th>
<th>Distance from arc start point to the center point in Z direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>G110,G111,G134,G135: cutting increment in XY plane each time</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alarm</th>
<th>Decimal efficiency</th>
<th>Round-off</th>
</tr>
</thead>
</table>

<p>| Decimal alarm | | |</p>
<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
<th>Value Range</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>G13~G139</td>
<td>Distance from start point to rectangle side in X axis direction</td>
<td>-9999.999 ~ 9999.999</td>
<td>Absolute value for negative, Round-off</td>
</tr>
<tr>
<td></td>
<td>The length of linear chamfering</td>
<td>-9999.999 ~ 9999.999</td>
<td>Absolute value for negative, Round-off</td>
</tr>
<tr>
<td></td>
<td>Punching number for linear serial punch (use together with the canned cycle punch)</td>
<td>-9999.999 ~ 9999.999</td>
<td>Absolute value for negative, Decimal part omitted</td>
</tr>
<tr>
<td></td>
<td>Tool life management, tool life value</td>
<td>0 ~ 9999</td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>Miscellaneous function</td>
<td>0 ~ 9999</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td>M code subprogram call</td>
<td></td>
<td>00000 ~ 99999</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td>N</td>
<td>Program number</td>
<td>0 ~ 2^31</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td></td>
<td>Tool life: tool life unit (0-time, non-0-time)</td>
<td>0 or other number</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td>P</td>
<td>Delay time in G04 (ms)</td>
<td>-9999999999.999 ~ 9999999999.999</td>
<td>Ignore negative, Decimal alarm</td>
</tr>
<tr>
<td></td>
<td>What kind of number reference return in G30</td>
<td>2 ~ 4</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td></td>
<td>Skip sequence or alarm number in G65</td>
<td>0 ~ 999999999999999</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td></td>
<td>M subprogram call (times: program name)</td>
<td>0 ~ 999999999999999</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td></td>
<td>Sequence number of M subprogram return</td>
<td>0 ~ 999999999999999</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td>R</td>
<td>Specifying G73 and G83 cut-in value per time</td>
<td>-9999999999.999 ~ 9999999999.999</td>
<td>Absolute value for negative, Round-off</td>
</tr>
<tr>
<td></td>
<td>The value of operation in G5</td>
<td>-9999999999999999999</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td></td>
<td>Radius value of arc</td>
<td>-9999999999.999 ~ 9999999999.999</td>
<td>Absolute value for negative, Round-off</td>
</tr>
<tr>
<td></td>
<td>Plane value of canned cycle command</td>
<td>-9999999999.999 ~ 9999999999.999</td>
<td>Absolute value for negative, Round-off</td>
</tr>
<tr>
<td></td>
<td>The value of operation in G5</td>
<td>-9999999999999999999</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td>S</td>
<td>Analog spindle</td>
<td>0 ~ 9999999999999999999</td>
<td>Decimal alarm</td>
</tr>
<tr>
<td></td>
<td>Shift spindle</td>
<td>0 ~ 9999999999999999999</td>
<td>Decimal alarm</td>
</tr>
</tbody>
</table>
### 3.2 Rapid Positioning G00

**Format:**  
G00 X___ Y___ Z___;

**Function:** X, Y and Z axes simultaneously move to end points from start at their rapid traverse rates. See Fig. 3-1.

Two axes move at their respective speeds, the short axis arrives at the end firstly, the long axis moves the rest of distance independently, and their resultant paths are possibly not linear.

**Explanation:** G00, which is initial G command;

The value ranges of X, Y and Z are indicated as -9999.999~+9999.999mm; X, Y and Z axes, one of them can be omitted or all of them can be omitted. When one of them is omitted, it means that the coordinate value of start and end points are same. The start and end points share the same position when they are omitted at the same time.

**Command path figure:**

Tool positions at the rapid traverse rate independently for each axis. Usually, the tool path is not linear.
X, Y and Z axes are separately set by the system data parameter No.059, No.060 and No.061 at their rapid traverse rate, the actual traverse rate can be modified by the rapid override keys on the machine panel.

The rapid traverse acceleration or deceleration time constant of X, Y and Z axes are separately set by the system data parameter No.064, No.065 and No.066.

Example: tool traverses from point A to point B. See Fig. 3-2.

![Diagram of tool traversal](image)

### Fig. 3-2

```
G90 G0 X120 Y253 Z30;  (absolute coordinate programming)
G91 G0 X160 Y-97 Z-50;  (relative coordinate programming)
```

### 3.3 Linear Interpolation G01

**Format:** G01 X_Y_Z_F;

**Function:** Movement path is a straight line from start to end points.

**Explanation:** G01, which is modal G command;

- The value range of X, Y and Z are indicated as -9999.999~+9999.999mm;
- X, Y and Z axes which one of them can be omitted or all of them can be omitted.
When one of them is omitted, it means that the coordinate value of start and end points are consistent. The start and end points share the same position when they are omitted at the same time.

F command value is vector resultant speed of instantaneous rates in X, Y and Z axes directions, the actual feedrate is the product of override and F command value; F command value is invariable after it is performed till the new one is executed. The following G command with F command word uses the same function.

The value range is indicated as follows:

<table>
<thead>
<tr>
<th>Command function</th>
<th>G94 (mm/min)</th>
<th>G95 (mm/rev)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value range</td>
<td>1~15000</td>
<td>0.001~500</td>
</tr>
</tbody>
</table>

**Command path figure:**

The linear interpolation is performed from point O to point A: $G01 \ X \ a \ Y \ \ b \ Z \ y \ F$.

The feedrate specified by F is the tool movement speed along the line. The speed of each axis is as follows:

- Speed in X axis direction: $F_x = \frac{a}{L} \times f$
- Speed in Y axis direction: $F_y = \frac{b}{L} \times f$
- Speed in Z axis direction: $F_z = \frac{y}{L} \times f$

Note: The F initial default value is set by data parameter No.172 when the power is turned on.
Circular and helical interpolation

Format:

**Circular interpolation:**

- **Circular in the XY plane:**

  
  \[
  G17 \begin{cases}
  \text{G02} & \{ \begin{array}{c}
  X_-Y_- \\
  \text{G03} & \{ \\
  R_- I_- \end{array} \\
  F_
  \end{cases} \\
  \end{cases}
  \]

- **Circular in the XZ plane:**

  
  \[
  G18 \begin{cases}
  \text{G02} & \{ \begin{array}{c}
  X_-Z_- \\
  \text{G03} & \{ \\
  R_- I_- \end{array} \\
  F_
  \end{cases} \\
  \end{cases}
  \]

- **Circular in the YZ plane:**

  
  \[
  G19 \begin{cases}
  \text{G02} & \{ \begin{array}{c}
  Y_-Z_- \\
  \text{G03} & \{ \\
  R_- J_- \end{array} \\
  F_
  \end{cases} \\
  \end{cases}
  \]

**Helical interpolation**

- **Circular in the plane of axis linear interpolation linkage:**

  
  \[
  G17 \begin{cases}
  \text{G02} & \{ \begin{array}{c}
  X_-Y_-Z_- \\
  \text{G03} & \{ \\
  R_- I_- \end{array} \\
  F_
  \end{cases} \\
  \end{cases}
  \]

- **Circular in the plane of axis linear interpolation linkage:**

  
  \[
  G18 \begin{cases}
  \text{G02} & \{ \begin{array}{c}
  X_-Z_-Y_- \\
  \text{G03} & \{ \\
  R_- I_- \end{array} \\
  F_
  \end{cases} \\
  \end{cases}
  \]

- **Circular in the plane of axis linear interpolation linkage:**

  
  \[
  G19 \begin{cases}
  \text{G02} & \{ \begin{array}{c}
  Y_-Z_-X_- \\
  \text{G03} & \{ \\
  R_- J_- \end{array} \\
  F_
  \end{cases} \\
  \end{cases}
  \]

**Function:**

- Only two axes of circular interpolation can be linked for controlling tool movement along with the arc on the selected plane in any time. If the third axis is specified simultaneously in linear interpolation mode, it will be linked by linear interpolation type to constitute helical interpolation. Movement path is clockwise from start to end points. Movement path is counterclockwise from start to end points.
**Explanation:**

G02 and G03 are modal G commands; 
R is arc radius, the value range are indicated as ±9999.99999 mm; 
When the circle center is specified by address I, J and K, they are corresponding with the X, Y and Z axes separately.

I is the difference between the center point and the arc start point in the X axis direction, center point coordinate X - coordinate of arc start point, the value range are indicated as ±9999.99999 mm;

J is the difference between the center point and arc start point in the Y axis direction, center point coordinate Y - coordinate of circle arc start point, the value range are indicated as ±9999.99999 mm;

K is the difference between the center point and circle start point in the Z axis direction, center point coordinate Z - coordinate of circle start point, the value range are indicated as ±9999.99999 mm.

Note: When I, J and K are for whole circle that they have signs according to the direction. And they are positive values when I, J and K share the same directions with X, Y and Z axes; otherwise they are negative ones.

<table>
<thead>
<tr>
<th>Item</th>
<th>Specified Content</th>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Plane specification</td>
<td>G19</td>
<td>Specifying YZ plane arc</td>
</tr>
<tr>
<td>2</td>
<td>Rotating direction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>End point</td>
<td></td>
<td>End point in the part coordinate system</td>
</tr>
<tr>
<td>4</td>
<td>Distance from start point to circle center point</td>
<td></td>
<td>Distance from start to end points</td>
</tr>
<tr>
<td>5</td>
<td>Arc radius</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Feedrate</td>
<td>F</td>
<td>Feedrate along the arc</td>
</tr>
</tbody>
</table>

“Clockwise” and “Counterclockwise” are defined when XY plane (Z X plane, YZ plane) is viewed in the positive to negative direction of the Z (Y, X) axis in the Cartesian coordinate system; see the following figure:
The end point of an arc is specified by using the address \( X \), \( Y \) or \( Z \) and is expressed as an absolute or incremental value according to \( G90 \) or \( G91 \). The incremental value is the distance value from start to end points of an arc. The arc center is specified by address \( I \), \( J \) and \( K \) against the \( X \), \( Y \) and \( Z \) respectively. The numerical value following \( I \), \( J \) and \( K \) however is a vector component from start point of an arc to the center point which is an incremental value with sign. See the following figure:

The \( F \) command is circular interpolation rate in helical interpolation so in order to achieve the linkage interpolation between linear axis and arc the speed of linear interpolation by the \( \text{rd} \) axis has the following relationship to the \( F \) command:

\[
f = F \times \frac{\text{Length of linear axis}}{\text{Length of circular arc}}
\]

Helical interpolation path is as follows:
I, J and K have signs according to the direction. The circular center also can be specified by radii other than I, J and K as follows:

- Radius R

Now, the following two arcs can be described, one arc is more than 180°, the other is less than 180°. The arc radius which is less than 180° is specified by the positive value; the arc radius which is more than 180° is specified by the negative value. The radius is either positive or negative when the arc command is equal to 180°.

Example: Arc less than 180°

G02 X60.0 Y20.0 R50.0 F300.0;

Arc more than 180°

G02 X60.0 Y20.0 R-50.0 F300.0;
Example for the programming

To program the above paths using the absolute mode and incremental mode respectively:

**Absolute mode**

```
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
```

```
G92 X200.0 Y40.0 Z0
G90 G03 X140.0 Y100.0 R60.0 F300.0
G02 X120.0 Y60.0 R50.0
```

**Incremental mode**

```
G91 G03 X-60.0 Y60.0 I-60.0 F300.0
G02 X-20.0 Y-40.0 I-50.0
```

```
G91 G03 X-60.0 Y60.0 R60.0 F300.0
G02 X-20.0 Y-40.0 R50.0
```

The feedrate of circular interpolation is specified by the F command; it is the speed of the tool along the arc tangent direction.

**Note 1:** The I, J, and K can be omitted, but it is very necessary to input one of the addresses I, J, or K, or the system alarm is generated.

**Note 2:** The X, Y, and Z can be omitted simultaneously when the end and start points share the same position. When the center point is specified by addresses I, J, and K, it is a 360° arc.

- **G02 I_;** (Full circle)
- **G02 R_;** (not move)

It is recommended that programming uses R, in order to guarantee the start and end points of the arc are consistent with the specified value, the system will move by counting R according to the selected plane when programming using the I, J, and K.
<table>
<thead>
<tr>
<th>Plane selection</th>
<th>Count the radius value again</th>
</tr>
</thead>
<tbody>
<tr>
<td>G17</td>
<td>$R = \sqrt{I^2 + J^2}$</td>
</tr>
<tr>
<td>G18</td>
<td>$R = \sqrt{I^2 + K^2}$</td>
</tr>
<tr>
<td>G19</td>
<td>$R = \sqrt{J^2 + K^2}$</td>
</tr>
</tbody>
</table>

Note 3: The error between the actual tool feedrate and the specified feedrate is ±2% or less. The command speed is movement speed after tool radius offset along the arc.

Note 4: The R is effective when address I, J and K are commanded with the R, but the I and J are disabled at one time.

Note 5: The axis does not exist is specified on the set plane the alarm occurs.

Note 6: If the radius difference between start and end points exceeds the permitted value by parameter No.1, an alarm occurs.

### 3.5 Dwell G04

**Format:**

G04 P_; or

G04 X_;  

**Function:**

Axes stop, the current command mode and the data, status are invariable after delay time specified, the next block will be executed.

**Explanation:**

G04 which is a non-modal command

- Delay time is specified by command words P_ or X_

See the following figure table for time units of P_ and X_ command values:

<table>
<thead>
<tr>
<th>Address</th>
<th>P</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit</td>
<td>0.001 s</td>
<td>0~999999</td>
</tr>
</tbody>
</table>

**Note:**

- X can be specified by the decimal but P not, or the alarm will be generated.
- When the P and X are not introduced or they are negative value, it means exact stop between the
- The P is effective when the P and X are in the same block.
- The operation is held on when feeding during the G04 execution. Only the delay time execution is finished, can the dwell be done.
### Plane Selection Command G17, G18 and G19

**Format:**
- G17  … … XY plane
- G18  … … Z X plane
- G19  … … YZ plane

**Function:**
The plane of arc interpolation and tool radius compensation are chosen by using the G code.

**Explanation:**
- G17, G18 and G19 are modal G commands, the plane will not be changed when a block without any commands inside.

**Command example:**
- G18  X_  Z_
- Z X plane
- X_  Y_
- Invariable plane ( Z X plane)

**Note:**
- Note 1: The plane selection command can share the same block with other group G commands.
- Note 2: The move command is regardless of the plane selection. For example, the Z axis is not On X Y plane, the Z axis movement is regardless of the X Y plane in command G17 Z_.

### Conversion of Inch and Metric G20 and G21

**Format:**
- G20/ G21

**Function:**
The input unit either inch or metric is chosen by G code.

**Explanation:**
- Unit system G code
  - Metric G20 0.0001 inch
  - Inch G21 0.001 mm

The G code should be placed in front of the program when inch and metric is switched each other. Before the coordinate system is set, it is specified by a single block command.

The following units systems vary according to the G code for inch or metric conversion.
1. Feed rate command value by F.
2. Command value related to the position.
3. Offset.
4. Scale value for MPG.
5. Step amount value.
6. Current coordinate value.
Note 1: The G code for inch or metric conversion when the power is turned on is the same as that at the power off.

Note 2: Changing G20 and G21 are unallowed during programming. Or the alarm occurs.

Note 3: When the unit systems between the machine and input are different, the max. error is 0.5 of the min. move unit and the error is not be cumulated.

Note 4: As the inch input (G20) and the metric input (G21) switches each other, the offset should be suited to the reset of the input unit.

### 3.8 Reference Point Return G28

**Format:**

```
G28 X_ Y_ Z_;
```

**Function:**

The middle point position specified by X, Y and Z is reached from the start point at the rapid traverse rate, then it returns to the reference point.

**Explanation:**

G28 is a non-modal G-command;

X: The absolute coordinate of middle point in X axis is indicated by G90, the middle point increment against current point in X axis is indicated by G91;

Y: The absolute coordinate of middle point in Y axis is indicated by G90, the middle point increment against current point in Y axis is indicated by G91;

Z: The absolute coordinate of middle point in Z axis is indicated by G90, the middle point increment against current point in Z axis is indicated by G91.

One of the command address X, Y and Z or all of them can be omitted, as follows:

<table>
<thead>
<tr>
<th>Command</th>
<th>un</th>
</tr>
</thead>
<tbody>
<tr>
<td>G28 X</td>
<td></td>
</tr>
<tr>
<td>G28 Y</td>
<td></td>
</tr>
<tr>
<td>G28 Z</td>
<td></td>
</tr>
<tr>
<td>G28 X, Y</td>
<td></td>
</tr>
<tr>
<td>G28 X, Z</td>
<td></td>
</tr>
<tr>
<td>G28 Y, Z</td>
<td></td>
</tr>
<tr>
<td>G28 X, Y, Z</td>
<td></td>
</tr>
</tbody>
</table>

**Process for command action**

(See the figure 3-10):

1. Positioning from current position to intermediate point of command axis at the rapid traverse rate (From point A to B)
2. Positioning to the reference point from intermediate point at the rapid traverse rate (From point B to R)
3. If the machine tool is unlocked, the zero return indicator lights up when the reference point return is finished.
Note:

- After power on, if G28 is executed prior to the manual machine zero return, the process of G28 machine zero return should be consistent with manual machine zero return, and the deceleration signal and one-rotation signal should be detected. The G28 machine zero return here will not detect the deceleration signal and one-rotation signal, but directly position to zero point.

- During the process of point A-B and B-R, the two axes move at two independent speeds, therefore, the paths may not be linear.

- After the execution of G28 machine zero return, the bit 7 of parameter No.22 decides whether to cancel cutter compensation or not.

- In compensation mode, if command G28 is specified, the compensation will be canceled in the intermediate point. The compensation mode is canceled automatically after reference point return.

- If zero point switch is not equipped on the machine tool, G28 command and machine zero return are disabled.

- The intermediate point can only be established during the movement from the intermediate point to the reference point which followed the movement from the start point to the intermediate point.

- After the modification of parameters which set the zero return point, manual reference point return is necessary; G28 command can be executed later.

3.9 Return from Reference Point G29

Format: G29 X_ Y_ Z_

Function: When a rapid traverse is performed from the current point to mid point, it positions to the specified position by X, Y and Z at the rapid traverse rate.

Explanation:

- X: The absolute coordinate of aim point in X axis is indicated by G90; the aim point increment against the mid point in X axis is indicated by G91;

- Y: The absolute coordinate of aim point in Y axis is indicated by G90; the aim point increment against the mid point in Y axis is indicated by G91;

- Z: The absolute coordinate of aim point in Z axis is indicated by G90; the aim point increment against the mid point in Z axis is indicated by G91;

- One of the command address X, Y and Z or all of them can be omitted, see the following figure:
### Process for command action:

![Diagram showing the process for command action](image)

**Note:**
- Note 1: G29 is specified after G28, if an intermediate point is not specified by any of axes, the system alarm will be generated.
- Note 2: It is incremental distance against the intermediate point in G91 coordinate programming.
- Note 3: Current position is reference point when the G29 command is followed to G28 or G30, it returns from reference point directly; or, it returns from current position if G29 command is not followed by G28 or G30.
Format:

G30 P2 X_ Y_ Z _ ; the machine 2nd reference point return (P2 can be omitted)

G30 P3 X_ Y_ Z _ ; the machine 3rd reference point return

G30 P4 X_ Y_ Z _ ; the machine 4th reference point return

Function:

From the start point, after the intermediate point by X, Y and Z is reached at a rapid traverse rate, the machine 2nd, 3rd and 4th reference points are returned. The command word P2 can be omitted when the machine 2nd reference point is returned.

Explanation:

G30, which is a non-modal G-command;

X: X axis coordinate for intermediate point;

Y: Y axis coordinate for intermediate point;

Z: Z axis coordinate for intermediate point;

One of the command address X, Y and Z or all of them can be omitted, see the following figure:

<table>
<thead>
<tr>
<th>Command</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>G30 Pn</td>
<td>Machine n th reference point return for X axis, Y and Z axes in the original position</td>
</tr>
<tr>
<td>G30 Pn</td>
<td>Y_ Z _ Machine n th reference point return for Y and Z axes, X axis in the original position</td>
</tr>
<tr>
<td></td>
<td>Machine 3 axes in the original position, the next block</td>
</tr>
<tr>
<td>G30 Pn</td>
<td>X_ Y_ Z _ X, Y and Z axes return to the machine n th reference</td>
</tr>
</tbody>
</table>

Note 1: n is 2, 3 or 4 in above table;

Note 2: Deceleration and zero signals check are not needed when the machine 2nd, 3rd and 4th reference points are returned to.
**Chapter 3**

**G Command**

**Command action process**

1. **Positioning to intermediate point of the specified axis from current position at a rapid traverse rate (from point A to point B);**
2. **Positioning to the 2nd reference position set by data parameter No.94 and No.96 at the setting speed by data parameter No.150 and No.152 (from point B to point R2);**
3. **When the reference point return if the machine is unlocked, the Bit 0 and Bit 1 of the reference point returning end signal ZP21 are HIGH.**

**Note 1:** After returning the machine reference point by manual or the G28 command is performed, the machine 2nd, 3rd and 4th reference point return function can be employed only, or the 2nd, 3rd and 4th reference point operation of G30 command, the system alarm will be generated.

**Note 2:** From point A to B or from point B to R2, the axes are moved at their separately rate, so the path is not straight line possibly.

**Note 3:** After machine 2nd, 3rd and 4th reference point returned by the G30 command, the system tool length compensation cancellation is defined by bit 7 of the parameter No.22.

**Note 4:** The 2nd, 3rd and 4th reference point operation of G30 command can not be executed if the zero switch is not installed on the machine tool.

**Note 5:** The workpiece coordinate system is set after the machine 2nd, 3rd and 4th reference point are returned.

### 3.11 Skip Function G31

As G01 linear interpolation is performed, if an external SKIP signal is valid during execution of this command, execution of this command is interrupted and the next block is executed. The skip function is used when the end of machining is not programmed but specified with a signal from the machine, for example, in grinding. It is also used for measuring the dimensions of a workpiece.

**Format:**

`G31 X_ Y_ Z_`

**Explanation:**

1. G31, which is a non-modal G-code, it is effective only in the block in which it is specified.
2. G31 cannot be specified in the tool compensation and chamfering, or the alarm will be generated.
- The next block to G31 is incremental command 1: it moves with incremental value from the position interrupted by the skip signal.

Example: G31 G91 X100.0 F100 Y50.0

- The next block to G31 is absolute command for one axis: the command axis moves to the specified position, and the axis not specified keeps at the skip signal position.

Example: G31 G90 X200.0 F100 Y100.0

- The next block to G31 is absolute command for 2 axes: wherever the skip signal input is, the tool moves to specified position of next block.

Example: G31 G90 X200.0 F100 X300.0 Y100.0

- The next block to G31 is absolute command for 3 axes: wherever the skip signal input is, the tool moves to specified position of next block.

Example: G31 G90 X90 Y00.0 Z00.0
Chapter 3 - G Command

Volume I - Programming

3.12 Tool Nose Radius Compensation (G40, G41 and G42)

Format:

Functions:

Tool nose radius compensation function
To cancel or perform the tool radius compensation vector by using the commands G40, G41 and G42. They are combined with the commands G00, G01, G02 and G03 for specifying a mode which can be confirmed the compensation vector value, direction and the direction of tool movement.

G codes

<table>
<thead>
<tr>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>G40</td>
</tr>
<tr>
<td>G41</td>
</tr>
<tr>
<td>G42</td>
</tr>
</tbody>
</table>

G41 or G42 drives the system into compensation mode; G40 cancels the system compensation mode.

Explanation:

- Compensation plane
  The compensation plane can be confirmed based upon plane selection command; the tool compensation C is calculated in this plane.

- Compensation value (D code)
  This system can be set for 32 compensation values at most. Two digits specified by D code in the program, is called serial number of compensation value, the compensation value should be set by MDI / LCD unit.

  D code determines the compensation value in tool offset page according to the bit 1 of parameter No.003, it is very important to notice that the value applied is diameter or radius.

Plane selection

<table>
<thead>
<tr>
<th>Planes</th>
<th>Plane compensation</th>
</tr>
</thead>
<tbody>
<tr>
<td>G17</td>
<td></td>
</tr>
<tr>
<td>G18</td>
<td></td>
</tr>
<tr>
<td>G19</td>
<td></td>
</tr>
</tbody>
</table>

- Plane selection
  Plane selection is a selection of the plane on which the tool moves. The plane selection is made by the G code.

  G17 Y over X plane
  G18 Z over X plane
  G19 Z over Y plane

- Plane compensation
  Plane compensation is the process of compensating for the tool nose radius in the selected plane. The plane compensation is also made by the G code.

  G41 tool radius left compensation
  G42 tool radius right compensation

- Compensation vector
  Compensation vector is the vector that compensates for the tool nose radius. The compensation vector is determined by the G code.

- Compensation value
  Compensation value is the value that determines the tool nose radius compensation. The compensation value is specified by the D code.

- Compensation method
  Compensation method is the method of compensating for the tool nose radius. The compensation method is determined by the compensation value.

- Compensation result
  Compensation result is the result of compensating for the tool nose radius. The compensation result is displayed on the LCD panel.
Compensation vector

The compensation vector is a two-dimensional vector. It is equal to the compensation value specified with D code. The compensation vector is calculated in the control unit, its direction is real-time modified along with the tool path in each block. You can calculate how much compensation is needed for tool movement when the compensation value is applied in the control unit. Compensation path (tool center path) = programmed path + tool radius (or diameter) (determined by compensation direction).

Note:

- Compensation operation is executed in the plane selected by G17, G18, G19. For example, when XY plane is selected, (X, Y) or (I, J) is used to carry out compensation operation and vector operation. The coordinate value whose axis is not in the compensation plane is not affected by the compensation.
- In 3-axis linkage control, compensation is only performed for the tool path projected on the compensation plane.
- The alteration of compensation plane should be executed posterior to the compensation mode cancelled. Otherwise, the system will give an alarm and machine stops.
- When the cutter compensation is cancelled by G40, movement amount should be specified, otherwise, an alarm will occur.
- In the canned cycle G codes, G40, G41, G42 codes are disabled.
Chapter 3 Command

Example:

In this block, the compensation cancellation mode becomes compensation mode by G41.

At the end of this block, the tool center is compensated in the direction that the tool radius is vertical to the next program path (from P1 to P2). Tool compensation value is specified with D07, so set the compensation number to 7, then the G41 is indicated with tool path compensation left.

After the compensation begins, tool path compensation performs automatically when creating the workpiece as P1→P2......P8→P9→P1.

```
N00 G92 X0 Y0 Z0
N01 G90 G17 G00 G41 D7 X250.0 Y550.0 T
The compensation value should be pre-set with compensation number V
N02 G01 Y900.0 F150
N03 X450.0
N04 G03 X500.0 Y1150.0 R650.0
N05 G02 X900.0 R-250.0
N06 G03 X950.0 Y900.0 R650.0
N07 G01 X1150.0
N08 Y550.0
N09 X700.0 Y650.0
N10 X250.0 Y550.0
N11 G00 G40 X0 Y0
```
3.13 Tool Length Compensation (G43, G44, G49)

Function:
G18, G19

Explanation:
G43 and G44 are modal G codes; they are effective before meeting other G codes in the same group.
The endpoint specified by Z axis moves an offset value, as above figure G17 plane is selected. The difference between supposed and actual machine tool length value is preset at the offset storage when the program is applied. Different length tool can be employed by changing tool length compensation value, so program change is not needed.
Different offset directions were specified by G43 and G44, the offset number is specified by H code.

Offset axis
The offset axes are vertical to the specified planes (G17, G18 and G19)

Specifying plane Offset axes
G17      Z axis
G18      Y axis
G19      X axis

Tool position offset for two or more axes can be used to specify the offset axis and the offset axis changed by 2 ~ 3 blocks
(Example) X and Y axes compensation
G19 G43 H_; … X axis offset
G18 G43 H_; … Y axis offset, composed with the previous block, X and Y axes are compensated.

Function:

<table>
<thead>
<tr>
<th>Offset Value</th>
<th>Compensation</th>
</tr>
</thead>
<tbody>
<tr>
<td>H_</td>
<td></td>
</tr>
<tr>
<td>G18</td>
<td></td>
</tr>
<tr>
<td>G19</td>
<td></td>
</tr>
</tbody>
</table>

Different length tool can be employed by changing tool length compensation value.
Chapter 3: Command

Volume I: Programming

**Offset direction**

G43: Positive offset
G44: Negative offset

Compensation axes can be regarded as Z, Y and X. Either absolute or incremental command, the end point coordinate value specified by Z axis movement command in program adds the offset specified by H codes in G43 (set in the offset storage), or subtracts the offset specified by H codes in G44, finally, the value calculated is regarded as the end point coordinate.

The following command is indicated for Z axis movement omitting: When the offset is positive, G43 is for an offset in the positive direction; G44 is for an offset in the negative direction. It reverses movement when the offset is negative value.

**Specifying the offset**

An offset number is specified by H code and its corresponding offset adds or subtracts Z axis movement command value in program to get a new Z axis movement command value. The offset number is H00~H32.

The offset value corresponded with offset number is preset in the offset storage by using the panel of LCD/MDI. Setting range for offset is as follows:

<table>
<thead>
<tr>
<th>Millimeter input</th>
<th>Inch input</th>
</tr>
</thead>
<tbody>
<tr>
<td>-9999.999 to +9999.999</td>
<td>-999.9999 to +999.9999</td>
</tr>
</tbody>
</table>

Offset number 00, i.e., H00 corresponds to the 0 offset. It is disabled to set offset value to H00.

**Tool length compensation cancellation**

G49 or H00 can be specified when tool length compensation is cancelled. When two or more axes compensations are cancelled, all the axes compensation will be cancelled if G49 is applied. Compensation value of the vertical axis for currently specified plane is cancelled with H00. After G49 or H00 is specified, the system immediately cancels the compensation value.

**Note:**

1. In the block that tool length compensation is specified, G02, G03, G04, G92, and G31 cannot be specified at the same time, otherwise, an alarm will occur.

2. Tool length compensation command can be specified in the block in which canned cycle is specified. But after the canned cycle is executed, tool length compensation is disabled and is not modal.

**Example:**

| Normal Modal Explanation (H1 = 10.0mm, H2 = 20.0mm) |
|------------------|------------------|------------------|------------------|------------------|
| G43 H1           | G44 G01 X50 Y50 Z50 H2 | G90 G00 X100 Y100 Z100 (Z80) | G43 H1 | G44 H2 |

Setting H1, tool length compensation in the positive direction.
Linear interpolation, setting H2 tool length compensation in the negative direction.
Position to X100 Y100 Z100 (Z80) with H2 compensation offset.
In the same block with canned cycle code

G43 H1
G44 G81 X50 R5 Z-70 H2
G90 G00 X100 Y100
G43 H1
G44 H2

Setting H1 tool length compensation in the positive direction.
Setting H2 tool length compensation in the negative direction. Starts the canned cycle from H2.

Specific in the canned cycle

G43 H1
G90 G81 X50 R5 Z-70 G49 H2
G49 G0 X75 Y75 Z75 H0
G43 H1
G43 H1
G43 H1
G49 H0

Setting H1 tool length compensation in the positive direction.

Compensation offset with H1; enters into canned cycle mode.
The tool length compensation (G49, H2) in the canned cycle is ineffective, and the previous block remains modal.
Cancel all axis compensations, and set H0 modal.

Position to X75 Y75 Z75 (Z75).

Command Example:

Tool length compensation (#1, #2 and #3 hole machining)

090 20.0 0.0
090 0.0 0.0
090 0.0 0.0

Compensation value (E=1MN)

Actual position
Programmed position

G0 X0 Y0 Z0
G90 90.00 0.0 0.0 ; ....

(1)
3.14 Workpiece Coordinate system  G54~G59

Format:
- G54 X__ Y__ Z__: Workpiece coordinate system 1
- G55 X__ Y__ Z__: Workpiece coordinate system 2
- G56 X__ Y__ Z__: Workpiece coordinate system 3
- G57 X__ Y__ Z__: Workpiece coordinate system 4
- G58 X__ Y__ Z__: Workpiece coordinate system 5
- G59 X__ Y__ Z__: Workpiece coordinate system 6

Function:
There are 6 workpiece coordinate systems for machine tool regardless of the G92, any of coordinate system can be selected by G54~G59.

Explanation:
- X: New X axis absolute coordinate in current position;
- Y: New Y axis absolute coordinate in current position;
- Z: New Z axis absolute coordinate in current position.

These six workpiece coordinates are set by the distances (workpiece zero offset) from machine zero to each coordinate system origin.
Example:

N10  G55 G90 G00 X100.0 Z20.0;
N20  G56 X80.5 Z25.5;

Rapidly positioning to workpiece coordinate system 3 (X=80.5, Z=25.5) from workpiece coordinate system 2 (X=100.0, Z=20.0). For example, if N20 block is G91, it is incremental movement. The absolute coordinates automatically become the coordinates in coordinate system G56.

The absolute position for the figure is coordinate value under the current coordinate system.

Note:
- Workpiece coordinate systems 1~6 is set up as soon as machine zero return is executed after power-on. When the system is restarted, the coordinate system is the one set by parameter No. 13 bit 17.
- Whether the relative position varies with coordinate system depends on status parameter No005 PP. When PP=0, it changes; when PP=1, it does not change.
- When the workpiece coordinate system function is determined, usually, G92 is not
needed to set coordinate system. If G92 is used, coordinate system 1~6 will be moved. Do not confuse with G92 and G54~G59, unless workpiece coordinate systems G54~G59 are to be moved. When G54~G59 are in the same block with G92, G54~G59 are disabled.

• Workpiece coordinate system can be modified in the program run. The new coordinate system is effective till the system is restarted.

If it performs G92 X100 Y100 commands when the tool is positioned at (200, 160) in the G54 coordinate system; the offset vector for workpiece coordinate system 1 is (X', Y'). And the other workpiece coordinate systems offset for vector A.

3.15 Compound Cycle Command

Generally, the canned cycle is a machining movement completion from one block with G function to the completion of multi-block specified. Canned cycles make it easier for the programmer to create programs. With a canned cycle, a frequently-used machining operation can be specified in a single block with a G function; without canned cycles, multiple blocks are needed, and canned cycles can shorten the program to save memory.

3.15.1 Canned Cycle List

<table>
<thead>
<tr>
<th>G Code</th>
<th>Canned Cycle</th>
<th>Operation at the Bottom of a Hole</th>
<th>Dwell</th>
<th>Feed</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>G73</td>
<td>intermittent feed</td>
<td>———</td>
<td>———</td>
<td>rapid feed</td>
<td>High-speed peck drilling cycle</td>
</tr>
<tr>
<td>G74</td>
<td>feed</td>
<td>well, spindle CCW</td>
<td>feed</td>
<td>left-hand tapping cycle</td>
<td></td>
</tr>
<tr>
<td>G80</td>
<td>———</td>
<td>———</td>
<td>———</td>
<td>canned cycle cancellation</td>
<td></td>
</tr>
<tr>
<td>G81</td>
<td>feed</td>
<td>———</td>
<td>rapid feed</td>
<td>drilling, point drilling</td>
<td></td>
</tr>
<tr>
<td>G82</td>
<td>feed</td>
<td>well</td>
<td>rapid feed</td>
<td>drilling, boring, counter boring</td>
<td></td>
</tr>
<tr>
<td>G83</td>
<td>intermittent feed</td>
<td>———</td>
<td>rapid feed</td>
<td>Peck drilling cycle</td>
<td></td>
</tr>
<tr>
<td>G84</td>
<td>feed</td>
<td>well, spindle CW</td>
<td>feed</td>
<td>Tapping</td>
<td></td>
</tr>
<tr>
<td>G85</td>
<td>feed</td>
<td>———</td>
<td>feed</td>
<td>boring</td>
<td></td>
</tr>
<tr>
<td>G86</td>
<td>feed</td>
<td>spindle stop</td>
<td>rapid feed</td>
<td>boring</td>
<td></td>
</tr>
<tr>
<td>G88</td>
<td>feed</td>
<td>well, spindle stop</td>
<td>manual</td>
<td>boring</td>
<td></td>
</tr>
</tbody>
</table>
## GSK980MDa Milling CNC System User Manual

### G89 Feed Dwell Feed Boring

<table>
<thead>
<tr>
<th>G89</th>
<th>Feed</th>
<th>well</th>
<th>Feed</th>
<th>Boring</th>
</tr>
</thead>
<tbody>
<tr>
<td>G110</td>
<td>Intermittent feed</td>
<td>Full-circle helical rough milling</td>
<td>Rapid feed</td>
<td>Round groove internal rough milling CCW</td>
</tr>
<tr>
<td>G111</td>
<td>Intermittent feed</td>
<td>Full-circle helical rough milling</td>
<td>Rapid feed</td>
<td>Round groove internal rough milling CW</td>
</tr>
<tr>
<td>G112</td>
<td>Feed</td>
<td>Full-circle fine milling</td>
<td>Rapid feed</td>
<td>Full-circle internal fine milling CCW</td>
</tr>
<tr>
<td>G113</td>
<td>Feed</td>
<td>Full-circle fine milling</td>
<td>Rapid feed</td>
<td>Full-circle internal fine milling CW</td>
</tr>
<tr>
<td>G114</td>
<td>Feed</td>
<td>Full-circle fine milling</td>
<td>Rapid feed</td>
<td>External round fine milling CCW</td>
</tr>
<tr>
<td>G115</td>
<td>Feed</td>
<td>Full-circle fine milling</td>
<td>Rapid feed</td>
<td>External round fine milling CW</td>
</tr>
</tbody>
</table>

### G134 Intermitent feed Rectangle rough milling

<table>
<thead>
<tr>
<th>G134</th>
<th>Feed</th>
<th>well</th>
<th>Feed</th>
<th>Boring</th>
</tr>
</thead>
<tbody>
<tr>
<td>G135</td>
<td>Intermittent feed</td>
<td>Rectangle rough milling</td>
<td>Rapid feed</td>
<td>Rectangle groove internal rough milling CCW</td>
</tr>
<tr>
<td>G136</td>
<td>Feed</td>
<td>Rectangle rough milling</td>
<td>Rapid feed</td>
<td>Rectangle groove internal rough milling CW</td>
</tr>
<tr>
<td>G137</td>
<td>Feed</td>
<td>Rectangle fine milling</td>
<td>Rapid feed</td>
<td>Rectangle groove internal fine milling CCW</td>
</tr>
<tr>
<td>G138</td>
<td>Feed</td>
<td>Rectangle fine milling</td>
<td>Rapid feed</td>
<td>Rectangle groove external fine milling CCW</td>
</tr>
<tr>
<td>G139</td>
<td>Feed</td>
<td>Rectangle fine milling</td>
<td>Rapid feed</td>
<td>Rectangle groove external fine milling CW</td>
</tr>
</tbody>
</table>

### 3.15.1.2 Canned circle explanations

Generally, a canned cycle consists of a sequence of the following operations, see the right figure.

**Operation 1**… Positioning of axes X and Y

**Operation 2**… Rapid traverse to point R plane

**Operation 3**… Hole machining;

**Operation 4**… Operation at the bottom of hole;

**Operation 5**… Retraction to point R plane

**Operation 6**… Rapid traverse to the initial Point

### 3.15.1.3 G90 / G91

The data mode corresponded with G90 and G91 are different. The point R plane and the absolute position machined at the bottom of the hole are specified by R and Z values, when the

---

The text above contains a detailed explanation of the canned cycles and their operations, including positioning, machining, and retraction. It also explains the data modes G90 and G91 and their implications for specifying positions and machined features.
command is G 90. The specified R value is the distance relative to the initial plane, and the Z value is the distance relative to the R point plane when the command is G91. See the Fig. 13.1 (B)

<table>
<thead>
<tr>
<th>G90 (Absolute command)</th>
<th>G91 (Incremental command)</th>
</tr>
</thead>
</table>

Fig. 13.1 (B) Absolute and incremental commands for canned cycle

Returning point levels

Tool can be returned to the initial plane or point plane according to G98 and G99 during returning. See the following figure Fig. 13.1 (C).

Normally, the initial hole machining is used by G99, the last machining is used with G98. The initial level will not be changed when the hole machining is done by G99.

<table>
<thead>
<tr>
<th>G98 (Return to initial level)</th>
<th>G99 (Return to point plane)</th>
</tr>
</thead>
</table>

Fig. 13.1 (C) Levels for initial and point plane
The initial point level is an absolute position for hole machining axis direction which is indicated from the canned cycle cancellation to start.

Canned cycle cancellation

There are two ways for canned cycle cancel are listed below:

1. Canceling the canned cycle with the G80
2. The canned cycle is cancelled by the G00, G01, G02 and G03 command in group 01.

   (1) When the canned cycle is cancelled by the command G80, if the G00, G01, G02 and G03 of the 01 group are not specified, then the reserved modal command (G00 or G01) performs motion before using canned cycle.

   For example:

   
   N0010 G01 X0 Y0 Z0 F800; (The modal command is G01 before entering the canned cycle)

   N0020 G81 X10 Y10 R5 Z-50; (Entering canned cycle)

   N0030 G80 X100 Y100 Z100; (The modal G01 command reserved before canned cycle performs cutting feed)  
   
   If the G01 is not specified in the abovementioned program N0010, but G00, the G00 performs rapid positioning for N0030.

   When both command G80 and commands G00, G01, G02 and G03 are specified in block, actions are performed by the latter, G00, G01, G02 and G03.

   For example:

   
   N0010 G01 X0 Y0 Z0 F800; (The modal command is G01 before entering the canned cycle)

   N0020 G81 X10 Y10 R5 Z-50; (Entering canned cycle)

   N0030 G00 G80 X100 Y100 Z100; (The G00 performs positioning at the rapid rate, and the modal command G00 is saved)

Note: The cutting feedrate by F command is still held on even if the canned cycle is cancelled.

3.15.1.6 General command format for canned cycle

Once the hole machining data is specified in the canned cycle, it is held until the canned cycle is cancelled. So the hole machining data should be outright specified at the beginning of the canned cycle, only the modified data is specified in the following canned cycle.

The general command format of canned cycle: G X Y R Z Q P L;

All commands for canned cycle are listed in above-mentioned format. But it is not needed to specify the above-mentioned format in each canned cycle. For example, the canned cycle can be performed as long as the G command (hole machining) and any of X, Y, Z and R are specified; additionally, Q or P is not available in some canned cycle G command (hole machining), the command is disabled even if these data are specified, they are regarded as modal data memories only.
Table 13.1.7 Command explanations for canned cycle

<table>
<thead>
<tr>
<th>Specifying content</th>
<th>Address</th>
<th>Explanation for command address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hole machining</td>
<td>G</td>
<td>Refer to the canned cycle list.</td>
</tr>
<tr>
<td>Hole position data</td>
<td>X, Y</td>
<td>Specifying the hole position with the absolute and incremental value, control is same with G00 position. Unit: mm;</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>See the fig.13.1 (B), the distance from initial point level to point R plane is specified by using the incremental value, or specifying the coordinate value of the point R by absolute value. Unit: mm;</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>Hole depth. See the fig.13.1 (A), the distance from R point to the bottom of a hole is specified by using the incremental value or specifying the coordinate value of the hole bottom by absolute value. Unit: mm;</td>
</tr>
<tr>
<td></td>
<td>Q</td>
<td>Specifying each cut-in in G73 and G83 or translational value in G76 and G87. Unit: mm;</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>Specifying the dwell at the bottom of a hole. Relation of time and the numerical specified are same with G04. Unit: ms;</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>The cutting feedrate is specified, tooth pitch is indicated in G74 and G84;</td>
</tr>
</tbody>
</table>

A part of command of canned cycle such as G110, G111, G112, G113, G115, G133, G135, G136, G137, G138 and G139 are explained in the following chapters or sections.

3.15 Description for canned cycle

3.15.1 High-speed peck drilling cycle G73

Format: G98 / G99 G73 X_ Y_ R_ Z_ Q_ F_ L_;

Function: This kind of cycle performs high-speed peck drilling, it performs intermittent cutting feed to the bottom of a hole, and eliminating the chips from the hole simultaneously.

Explanation: Refer to the command explanation of canned cycle in Table 13.1.7.

Cycle process:
(1) Positioning to XY plane level at the rapid traverse;
(2) Down to the point R plane at the rapid traverse rate;
(3) Cutting feed for Q distance;
(4) Retract d distance in rapid traverse;
(5) Cutting feed for (Q+d) distance
(6) Machine to the Z axis hole bottom by cycling the (4) and (5);
(7) Return to the start point level or point R plane according to G88 or G89 at the rapid traverse.
Related Explanation:

1. This kind of cycle is peck drilling for Q value intermittent feeding along the Z-axis direction. The Q value should be positive, the sign is ineffective even if the negative value is specified. If the Q value is not specified, then it defaults to 0.1mm. If a depth to be cut is less than the Q value, then cut to the bottom of the hole without tool retraction at the rapid traverse for the first time.

2. To remove chips from the hole easily, a small value can be set for retraction. This allows drilling to be performed efficiently. The tool is retracted in rapid feed, the retraction amount d is set by parameter No.51, the default is 1000, unit: 0.001mm.

3. The command P is disabled, but its value is reserved as canned cycle modal value.

3.15 Left-handed tapping cycle G74

Format:  G98  G99  G74  X_  Y_  R_  Z_  P_  F_  L

Function: This cycle performs left-handed tapping. In the left-handed tapping cycle, the spindle rotates clockwise for tapping till the bottom of the hole has been reached, then retracts by counter-clockwise after dwell.

Explanation: For canned cycle explanation, see the Table 13.1. Thereinto, the F is indicated for tooth pitch. The value range are indicated as 0.001-500.00mm (metric), 0.06-25400 teeth/inch (inch)

Cycle Process:

1. Positioning to XY plane level at the rapid traverse;
2. Down to the point R plane at the rapid traverse;
3. Tapping to the bottom of a hole;
4. The spindle stops;
5. Pause for time P if dwell is specified;
6. The spindle rotates CW, and then retracts to point R plane;
(6) The spindle is stopped; pause for time P if dwell is specified;
(8) Spindle rotates CW;
(9) Return to the initial plane if it is G8.

**Command Path:**

<table>
<thead>
<tr>
<th>G98 (Mode for returning to initial plane)</th>
<th>G99 (Mode for returning to R point plane)</th>
</tr>
</thead>
</table>

**Related Explanation:**

1. Tapping to the bottom of a hole it will not be returned immediately even if the P is omitted or regarded as 0 in this cycle, it will be returned after a dwell time (2s), and this time is set by system.
2. The F is tapping modal value, the last tapping F value is taken when it is omitted, or alarm will be generated if it does not exist.
3. The metric or inch of the F value is determined by G20 (metric) or G21 (inch).
4. The command Q is disabled in this cycle, but its value will be reserved as canned cycle modal value.

### 3.15.2.3 Tapping cycle G84

**Format:**

```
G84 G X_ Y_ R_ Z_ P_ F_ L_
```

**Function:**

This cycle is used to machine a thread. The tapping is performed by spindle rotating positively, when the bottom of a hole has been reached, the spindle is retracted in the reverse direction.

**Explanation:** For command explanation of canned cycle, see the Table 13.1. Thereinto, the F is tooth-pitch. The value range is 0.001-500.00mm (metric), 0.06-25400 tooth/inch (inch).

**Cycle Process:**

1. Positioning to the XY plane level at the rapid traverse;
2. Down to the point R plane at the rapid traverse;
3. Tapping to the bottom of a hole;
4. Spindle stops;
5. For dwell time P if it is commanded
6. Spindle returns to the point R plane in reverse direction;
(3) Spindle stops; for dwell time P if the P is commanded;
(8) The spindle is rotated in the positive direction;
(9) Returning to the initial point level if it is G8.

**Command Path:**

<table>
<thead>
<tr>
<th>G8 (Code for returning to initial point level)</th>
<th>G98 (Code for returning to point R plane)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Diagram" /></td>
<td><img src="image2.png" alt="Diagram" /></td>
</tr>
</tbody>
</table>

**Related Explanation:**
Please refer to the related explanation for G74 (Counter tapping cycle)

### 3.15.2.4 Drilling cycle, spot drilling cycle G81

**Format:**

```
G98 G99 G81 X_ Y_ R_ Z_ F_ L_ ;
```

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

**Explanation:** For the command explanation of canned cycle, see the Table 13.1.7.

**Cycle Process:**

1. Positioning to the XY plane level position at the rapid traverse;
2. Down to the point R plane at the rapid traverse;
3. Cutting feed to the bottom of the hole;
4. Returning to the initial point or point R plane at rapid traverse according to the G8 or G98.

**Command Path:**

<table>
<thead>
<tr>
<th>G8 Return to the initial plane at the rapid traverse</th>
<th>G98 Return to the R point plane at the rapid traverse</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image3.png" alt="Diagram" /></td>
<td><img src="image4.png" alt="Diagram" /></td>
</tr>
</tbody>
</table>
Related Explanation:
The command Q or P is disabled in this cycle, but its value will be saved as canned cycle modal value.

3.15.2.5 Drilling cycle: counter boring cycle G82

Format:  G98 / G99   G82   X_  Y_  R_  Z_  P_  F_  L_ ;

Function:  Cutting feed is performed to the bottom of the hole. Hole depth precision is added when the dwell is performed, and then the tool is retracted from the bottom of the hole at rapid traverse.

Explanation:  For the command explanation of these canned cycles, see the Table 13.1.

Cycle Process:
(1) Positioning to the XY plane level at the rapid traverse;
(2) Down to the point R plane at the rapid traverse;
(3) Cutting feed to the bottom of a hole
(4) Dwell for P time if it is commanded.
(5) Returning to the initial point or point R plane according to G98 or G99 at the rapid traverse;

Command Path:

Related Explanation:
(1) They are basically the same as G81 (drilling and spot-drilling machining), it is up after dwell at the bottom of a hole only (the dwell time is specified by P, the dwell will not be executed if it is not specified, and the command action is same as that of G81). In the blind hole, the accuracy of hole can be improved by the dwell.
(2) The command Q is disabled in this cycle, but its value will be reserved as the canned cycle modal value.

3.15.2.6 Peck drilling cycle G83

Format:  G98 / G99   G83  X_  Y_  R_  Z_  Q_  F_  L_ ;

Function:  This cycle performs high-speed peck drilling; it performs intermittent cutting feed to the bottom of a hole while removing chips from the hole.

Explanation:  The command explanation for canned cycle, see the Table 13.1.

Cycle Process:
(1) Positioning to the XY plane level at the rapid traverse;
(2) Down to the point R plane at the rapid traverse;
(3) Cutting feed for Q distance;
(4) Retract to the point R plane at the rapid traverse;
(5) Rapid feed to d distance to the end surface
(6) Cutting feed for (Q+d) distance;
(7) Cycling (4) (5) and (6) to the bottom of a hole along Z-axis;
(8) Return to the initial point or point R plane according to the G8 or G9 at the rapid traverse;

Command Path:

<table>
<thead>
<tr>
<th>Command</th>
<th>Path Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>G8</td>
<td>Returned to the initial plane at the rapid traverse</td>
</tr>
<tr>
<td>G9</td>
<td>Returned to the point R plane at the rapid traverse</td>
</tr>
</tbody>
</table>

Related Explanation:

(1) Same as G73, after feeding for Q, it returns to the point R plane at the rapid traverse firstly, and then rapid feeds to d mm to the end surface, then cutting feed is applied and the cycle is performed in turn. The Q value should be positive, even if the negative value is specified, and the sign is also disabled. Q value 0.001mm is defaulted if Q value is not specified; d, is set by the parameter No.52, its default value is 1000, and the unit is 0.001mm. If the cutting depth is less than the Q value, then cutting to the bottom of a hole at the first time, and rapid traverse retraction is not performed.

(2) The command P is disabled in this cycle, but its value will be reserved as canned cycle modal value.

3.15. Boring cycle G85

Format:  
Function:  After positioning along X and Y axes, rapid traverse is performed to point R; the boring is performed from point R to point Z thereafter. Cutting feed is performed to return point R plane when the Z point has been reached the bottom of a hole.

Command explanation for the canned cycle, see the table 13.1.
**Cycle Process:**

1. Positioning to the XY plane level at the rapid traverse;
2. Down to the point R plane at the rapid traverse;
3. Cutting feed to the bottom of a hole;
4. Cutting feed to the point R plane;
5. Returning to the initial point level if it is G 8;

**Command Path:**

<table>
<thead>
<tr>
<th>G 8 (Code for returning to initial point level)</th>
<th>G 9 9 (Code for returning to point R plane)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Diagram" /></td>
<td><img src="image2" alt="Diagram" /></td>
</tr>
</tbody>
</table>

**Related Explanation:**

1. This cycle is used to bore a hole. The command motion is basically same as the G81 (Drilling, Spot-drilling cycle), the difference is that by the G81 it returns to the point R plane in rapid traverse rate, while by the G85 it returns to the point R plane in feedrate when the cutting feed reaches the bottom of a hole.
2. The Q and P commands are disabled in this cycle, but its value is reserved as the canned cycle modal value.

### 3.15. Boring cycle G 6

**Format:** G 8 G 9 6 X_ Y_ R_ Z_ F_ L_ ;  

**Function:** After positioning along X and Y axes, rapid traverse is performed to R point, and the boring is performed from point R to point Z. The tool is retracted in rapid traverse and spindle is rotated positively when the spindle is stopped at the bottom of the hole.

**Explanation:** For command explanation for canned cycle, see the table 13.1.

**Cycle Process:**

1. Positioning to the XY plane level at the rapid traverse;
2. Down to the point R plane at the rapid traverse;
3. Cutting feed to the bottom of a hole;
4. The spindle stops;
5. Returning to the initial point or point R plane at rapid traverse according to the G 8 or G 9;
**Command Path:**

<table>
<thead>
<tr>
<th>G8 (Code for returning to start point level)</th>
<th>G99 (Code for returning to point R plane)</th>
</tr>
</thead>
</table>

```
Related Explanation:

1. This cycle is used to bore a hole. The command operation is basically same with G81, only spindle rotation status is different. After cut feeds to the bottom of a hole, the M05 is executed (spindle stops), then the point R plane is retracted at the rapid traverse, the M03 is then performed (spindle rotates positively) regardless of the currently spindle rotation status and the positive or negative rotation are specified before the canned cycle.

2. The command Q and P are disabled in this cycle, but its value is reserved as canned cycle modal value.

3.15.2.9 Boring cycle G88

**Format:** G88 G98 G99 X_ Y_ R_ Z_ P_ F_ L_;

**Function:** A dwell is performed at the bottom of a hole, the spindle is stopping. If the manual operation is applied now, tool can be removed manually. It is better to retract the tool safely from the hole regardless of any kind of manual operation. It is rapidly retracted to point R or initial plane when the automatic operation is performed again, the spindle is stopped and G88 is finished.

**Related Explanation:** For the command explanation of the canned cycle, see the table 13.1.

**Cycle Process:**

1. Positioning to the XY plane at the rapid traverse rate;
2. Down to the point R plane at the rapid traverse rate;
3. Cutting feed to the bottom of hole;
4. The spindle is stopped;
5. P time is delayed if it is specified.
6. Annual operation will be performed if the dwell is executed.
7. Restoring the automatic mode, retracting to initial point or point R plane according to the G88 or G99 at the rapid traverse rate.
8. The spindle rotates positively;
**Chapter 3 G Command**

**Command Path:**

<table>
<thead>
<tr>
<th>G8 (Mode for returning to initial plane)</th>
<th>G9 (Mode for returning to point R plane)</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Diagram of G8]</td>
<td>![Diagram of G9]</td>
</tr>
</tbody>
</table>

**Related Explanation:**

The command Q is disabled in this cycle, but its value is reserved as the canned cycle modal value.

**3.15.2.1 Boring cycle G8**

**Format:**

G8 / G9   G89   X_  Y_  R_  Z_  P_  F_  L_ ;

**Function:**

This cycle is used to bore a hole normally. This cycle performs a dwell at the bottom of the hole; the tool is then retracted from the bottom of the hole at the rapid traverse rate.

**Explanation:**

For the command explanation of the canned cycle, see the table 13.1.

**Cycle Process:**

1. Positioning to XY plane at the rapid traverse rate;
2. Down to the point R plane at the rapid traverse rate;
3. Cutting feed to the bottom of a hole;
4. For dwell time P if the P is specified;
5. Cutting feed to the point R plane;
6. Returning to the initial point level if it is G8;
7. Returning to the initial point or point R plane at the rapid traverse according to the G8 or G9;
Command Path:

<table>
<thead>
<tr>
<th>Command Path:</th>
<th>Command Path:</th>
</tr>
</thead>
<tbody>
<tr>
<td>G8 (Mode for returning to initial point level)</td>
<td>G89 (Mode for returning to point R plane)</td>
</tr>
</tbody>
</table>

Related Explanation:

1. G89 (Boring cycle) is basically same as the G85, a dwell is applied at the bottom of a hole (dwell time is specified by P, if it is not specified, the dwell is not applied, the command operation is same to the G85)
2. The command Q is disabled in this cycle, but its value is reserved as canned cycle modal value.

3.15.11 Groove rough milling inside the round G110 / G111

Format:

G110

G98 / G99 X_ Y_ R_ Z_ I _ W _ Q_ K_ V_ D_ F_

G111

Function: From the beginning of the center point, arc interpolations are performed helically till the round groove of programming dimension has been machined.

Explanation: For command explanation of the canned cycle, see the table 13.1.1.

G110: Groove rough-milling inside the round in CCW;
G111: Groove rough-milling inside the round in CW;
I: I is radius inside the round groove, it should be more than the radius of current tool.
W: The firstly cutting depth is from the R reference level to the undersurface along the Z axis direction, it should be more than 0 (The first cutting position is over the bottom of the groove, then bottom position is regarded as machining position);
Q: The cutting incremental value each time along Z axis direction;
$: The width increment of cut inside XY plane, it should be less than the tool radius, and more than 0;
$: The distance to the end machining plane at the rapid traverse, it should be more than 0 when cutting;
$: Tool radius serial number, the value range is 0~32, 0 is the default of $0$. The current
tool radius is determined by the specified serial number.

**Cycle Process:**

1. Positioning to the XY plane level at the rapid traverse rate;
2. Down to the point R plane at the rapid traverse rate;
3. Cut W depth downwards in cutting feedrate
4. Mill a round face with radius I helically by K increment each time from center point to outside.
5. The Z axis is retracted to the R reference surface at the rapid traverse rate;
6. X and Y axes are positioned to the center at the rapid traverse rate;
7. Down to distance V to the end machining surface along Z axis at the rapid traverse rate;
8. Cut along Z axis for (Q+V) depth;
9. Cycling the operations from (4) to (8) till the round surface of total depth is finished.
10. Return to the initial plane or point R plane according to G98 or G99.

**Command Path:**

![Diagram of the process](image-url)
Related Explanation:

The P and Q are disabled in this cycle, but the P value will be reserved as canned cycle modal value.

For example: A round inside groove rough-milling is specified in canned cycle G111, see the following

Figure

G:0 G00 X50 Y50 Z50;  (G00 positioning at the rapid traverse rate)
G: G111 X25 Y25 R5 Z-50 150 W20 Q10 D10 10 F800 1;  (Rough-milling cycle inside the round groove D1=5)
G80 X50 Y50 Z50;    (Canceling canned cycle, returning from the point R plane)
M30;
3.15.2.1 Fine-milling cycle inside full circle G112 G113

<table>
<thead>
<tr>
<th>Format:</th>
<th>G112</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>G113</td>
</tr>
</tbody>
</table>

**Function:** A fine-milling inside the full circle is finished with the specified radius value I and direction, the tool is retracted after the fine-milling.

**Explanation:** For command explanation of canned cycle, see the table 13.1.1.

**G112:** Fine-milling cycle inside the full circle in CCW.

**G113:** Fine-milling cycle inside the full circle in CW

**I:** Fine-milling circle radius, the value range is indicated as 0~9999.999 mm, the absolute value is taken when it is negative.

**J:** Fine-milling distance from start point to the center point, the value range is indicated as 0~9999.999 mm, the absolute value is taken when it is negative.

**D:** Sequence number of tool radius, the value range is indicated as 0~32, the 0 is default of D0. The current tool radius value is taken according to the specified sequence number.

**Cycle process:**

1. Positioning to the XY plane level at the rapid traverse rate;
2. Down to the point P level at the rapid traverse rate;
3. Feed to the bottom of a hole;
4. Perform the circle interpolation by the path of transit arc 1;
(5) Perform the full circle interpolation by the path of arc 2 and arc 3;
(6) Perform circular interpolation by the path of transit arc and return to the start point;
(7) Return to the initial point level or point R plane according to G88 or G89.

Command Path:

Related Explanation:
The commands Q, P and L are disabled in this cycle, but the Q and P value will be reserved as the canned cycle modal value.

For example: Fine-mill a finished rough-milling round groove by the canned cycle G112 command, see the following figure:

G0 G00 X50 Y50 Z50;  (G00 rapid positioning)
G G112 X25 Y25 R5 Z-50 150 10 F800 1;  (Start canned cycle, fine-milling cycle)
inside the circle at the bottom of a hole

G80 X50 Y50 Z50;  (The canned cycle is cancelled, returning from the point P level)

3.15.2.13 Fine-milling cycle outside circle G114 / G115

Format:

```
G114
G98 / G99 X_ Y_ R_ Z_ I_ J_ D_ F_;

G115
```

Function: A fine-milling outside the full circle is performed by the specified radius value and the
direction, and the tool is retracted after the fine-milling is finished.

Explanation: For command explanation of canned cycle, see the table 13.1.

G114: Finish-milling cycle for outside circle in CCW.
G115: Finish-milling cycle for outside circle in CW.
I: A fine-milling circle radius, the value range is indicated as 0~9999.999 mm, the
absolute value is taken when it is negative.
J: Distance of fine-milling between the start point and the circle, the value range is
indicated as 0~9999.999 mm; the absolute value is taken when it is negative.
D: The sequence number of tool radius, the value range is 0~32, 0 is the default of D0.
The current tool radius value is taken according to the specified sequence number.

Cycle Process:

1. Positioning to the XY plane level at the rapid traverse rate;
2. Down to the point R plane at the rapid traverse rate;
3. Cutting feed to the bottom of a hole;
4. Perform the circle interpolation by the path of transit arc 1;
5. Perform the full circle interpolation by the path of arc 2 and arc 3;
6. Perform circular interpolation by the path of transit arc 4 and return to the start point;
7. Return to the initial point level or point R plane according to G98 or G99.

Command Path:

Related Explanation:

1. The interpolation direction of between transit arc and fine-milling arc are different when the
fine-milling outside circle is performed, the interpolation direction in command explanation is
the interpolation direction of fine-milling arc.

(2) The command Q, P and L are disabled in this cycle, but the Q and P value are reserved as canned cycle modal value.

**Example:** A finished rough-milling round groove is performed by fine-milling with the canned cycle G114 command, see the following figure:

```
G0 G00 X50 Y50 Z50;    (G00 rapid positioning)
G99 G114 X25 Y25 R5 Z-50 150 J 60 F800 D1;    (Start canned cycle, the fine-milling cycle is performed outside the circle at the bottom of a hole D1= 5)
G80 X50 Y50 Z50;     (The canned cycle is cancelled, returning from the point R plane)
M30
```

### 3.15.1.1 Rectangle groove rough milling  
**G134/G135**

**Format:**  
\[
G134 \quad G98 / G99 \quad X_\_ \quad Y_\_ \quad Z_\_ \quad R_\_ \quad I_\_ \quad J_\_ \quad K_\_ \quad W_\_ \quad Q_\_ \quad V_\_ \quad U_\_ \quad D_\_ \quad F_\_
\]

**Function:** From the center of the rectangle, the linear cutting cycle is applied by the specified parameter data, till the rectangle groove with programmed dimension is made out.

**Explanation:** For command explanation of canned cycle, see the table 13.1.

- **G134:** Rectangle groove rough-milling in CCW
- **G135:** Rectangle groove rough-milling in CW

- **I:** The width of rectangle groove along the X axis direction
- **J:** The width of rectangle groove along the Y axis direction.
- **K:** The cut width increment inside XY plane, it is less than the tool radius, but, more than 0.
- **W:** For the first cutting along the Z axis direction, the distance is downward to the R reference surface, it is more than 0 (if the first cutting is over the position of the bottom of the groove, then the bottom of the groove is taken as the machining position)
- **Q:** The cutting incremental value each time along Z axis.
- **V:** Distance to the end machining surface, which is more than 0, when the rapid traverse
is executed.

- \( \text{D}: \) Corner arc radius, if it is omitted, that is no corner arc transition is not shown.
- \( \text{D}: \) Sequence number of tool radius, its value range is indicated as 0 ~ 32, thereunto, the 0 is default of \( \text{D} \). The current tool radius value is taken out according to the specified sequence number.

**Cycle Process:**

1. Positioning to the XY plane at the rapid traverse rate;
2. Down to the point R plane at the rapid traverse rate;
3. W distance depth is cut downwards by cutting feedrate
4. Fill a rectangle face helically by increment each time from center point to outside.
5. R reference surface is retracted along the Z axis at the rapid traverse rate.
6. The center of rectangle is positioned along the X and Y axes at the rapid traverse rate.
7. Down to distance V to the end machining surface along Z axis at the rapid traverse rate;
8. Cut along Z axis for (Q + V) depth;
9. Cycling the operation from (3) to (8) till the surface of total cutting is performed.
10. Return to the initial plane or point R plane according to G98 or G99.

**Command Path:**
Related Explanation:
The commands P and L are disabled in this cycle, but the P value is reserved as canned cycle modal value.

For example: An inside rectangle groove rough-milling is specified by G134 in canned cycle, see the following figure:
Chapter 3 G Command

G0 G00 X50 Y50 Z50;  (G00 rapid positioning)
G134 G1 X25 Y25 R5 Z-50 I7 0 J 50 W20 Q10 P10 Q10 F800 Z1; (Groove rough-milling cycle inside rectangle is performed D1= 5)
G80 X50 Y50 Z50;   (The canned cycle is cancelled, returning from the point R plane)
M30;

Note: If the parameter value of 97 is set for more than 10, the helical cutting feed along the Z axis will be performed by G110 and G111. So, the workpiece without groove can be machined by rough milling directly.

The helical feeding path is as follows:
3.15.15 Rectangle groove inner fine milling cycle  G136 G137

Format:

\[
\begin{align*}
G136 & : \quad G98 / G99 \quad X_\_ \quad Y_\_ \quad R_\_ \quad Z_\_ \quad I_\_ \quad J_\_ \quad D_\_ \quad K_\_ \quad U_\_ \quad F_\_ ; \\
G137 & : \\
\end{align*}
\]

Function: The tool performs fine-milling inside the rectangle with the specified width and direction, it is returned after finishing the fine-milling.

Explanation: For command explanation of canned cycle, see the table 13.1.1.

- G136: Finish-milling cycle inside groove of rectangle in CCW.
- G137: Finish-milling cycle inside groove of rectangle in CW.

I: The rectangle width along the X axis, the value range is indicated as 0~9999.999 mm.
J: The rectangle width along the Y axis, the value range is indicated as 0~9999.999 mm.
D: Sequence number of tool radius, the value range is 0~32, the 0 is default value of D0.
The current tool radius value is taken out according to the specified sequence number.
K: The distance between the finish-milling start point and the rectangle side in X axis direction, the value range is indicated as 0~9999.999 mm.
U: Corner arc radius; no corner arc transition if it is omitted. When the U is omitted or it is equal to 0 and the tool radius is more than 0, the alarm is generated.

Cycle Process:
1. Positioning to XY plane at the rapid traverse rate;
2. Down to point R plane at the rapid traverse rate;
3. Cutting feed to the bottom of a hole;
4. Perform the circle interpolation by the path of transit arc 1;
5. Perform the circular and linear interpolation by the path of 2-3-4-5-6;
6. Perform circular interpolation by the path of transit arc 7 and return to the start point;
7. Returning to the initial plane or point R plane according to G98 or G99.

Command Path:

<table>
<thead>
<tr>
<th>G136</th>
<th>G137</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Diagram 1" /></td>
<td><img src="image2.png" alt="Diagram 2" /></td>
</tr>
</tbody>
</table>

Related Explanation:

The commands Q, P and L are disabled in this cycle, but the Q and P values are reserved as the canned cycle modal value.
**Example:** To perform a fine-milling for the finished rough-milling rectangle groove with the canned cycle G136 command, see the following figure:

![Diagram of fine-milling process]

G00 G00 X50 Y50 Z50;  (G00 rapid positioning)
G136 X25 Y25 R5 Z-50 I80 J 50 K30 U10 F800 D1;  (Perform finish-milling inside the rectangle groove at the bottom of a hole in the canned cycle D1= 5)
G80 X50 Y50 Z50;  (The canned cycle is cancelled, returning from the point R plane)
M30;

### 3.15.16 Finish-milling cycle outside the rectangle G138 / G139

**Format:**

\[
G138 \quad G98 / G99 \quad X_0 \quad Y_0 \quad R_0 \quad Z_0 \quad I_0 \quad J_0 \quad D_0 \quad K_0 \quad U_0 \quad F_0
\]

\[
G139
\]

**Function:** The tool performs fine-milling outside the rectangle by the specified width and direction, it is returned after finishing the fine-milling.

**Explanation:**

- **G138:** Finish-milling cycle outside the rectangle in CCW.
- **G139:** Finish-milling cycle outside the rectangle in CW.
- I: The width of rectangle along the X axis, the value range is indicated as 0~9999.999 mm.
- J: The width of the rectangle along the Y axis, the value range is indicated as 0~9999.999 mm.
- D: Sequence number of tool radius, its value range is indicated as 0~32, thereinto, the 0 is default of D0. The current tool radius value is taken out according to the specified sequence number.
- K: The distance between the finish-milling start point and the side of rectangle along the X axis, the value range is indicated as 0~9999.999 mm.
- U: Corner arc radius, if it is omitted, no corner arc transition.

**Cycle process:**

1. Positioning to the XY plane at the rapid traverse rate;
2. Down to the point R plane at the rapid traverse rate;
3. Cutting feed to the bottom of a hole;
4. Perform the circle interpolation by the path of transit arc 1;
(5) Perform the circular and linear interpolation by the path of 2-3-4-5-6;
(6) Perform circular interpolation by the path of transit arc and return to the start point;
(7) Returning to the initial plane or point R plane according to G80 or G90.

**Command Path:**

![Diagram](image.png)

**Related Explanation:**

1. The interpolation direction of transition arc is inconsistent to that of the fine-milling arc when a fine-milling is performed outside the rectangle. The interpolation direction is the one for the fine-milling arc in the command explanation.
2. The commands Q, P and L are disabled in this cycle, but, the value of Q and P are reserved as canned cycle modal value.

**Example:** A finished rough-milling rectangle groove is performed by the fine-milling by the command G138 in canned cycle. See the following figure.

G0 G00 X50 Y50 Z50;  
G90 G138 X25 Y25 R5 Z-50 180 J 50 K30 U5 F800 D1;  
G80 X50 Y50 Z50;  
M30;

### 3.15.3 Continuous Drilling

Continuous equal interval drilling cycle is performed in the way that canned cycle is called according to the specified linear, rectangular or arc path.

Parameters related to continuous drilling
3.15.3.1 Line series punch L function

Holes machining cycle should be performed from current plane position to end point specified by X and Y are indicated if the L word is specified in canned cycle, so the current position (block start and end) will not be drilled, the end point position is regarded as the last hole, holes are equal-spaced, as follows:

<table>
<thead>
<tr>
<th>Value setting</th>
<th>System execution result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value is negative</td>
<td>Ineffective, the value should be positive</td>
</tr>
<tr>
<td>The value is unspecified or equal to 1</td>
<td>Normal drilling cycle 1 time</td>
</tr>
<tr>
<td>The value is 0</td>
<td>No change of axes, the system reserves relevant cycle modal data</td>
</tr>
<tr>
<td>The value is decimal</td>
<td>When $\text{L} &gt; 1$, using round number</td>
</tr>
<tr>
<td></td>
<td>When $\text{L} &lt; 1$, it is processed as $\text{L} = 0$, not moving but reserving its modal data and relevant cycle parameter values.</td>
</tr>
</tbody>
</table>

Note 1: The maximum input value of command L is $-9999.999999$; Decimals are ignored and absolute value is used instead of negative value. L code is effective only in current block.

Note 2: In continuous drilling, the return planes are R point plan. After the last hole is processed, the return plane is specified by G98/G99.

Note 3: When there is no axis position command in the specified L block, it means drilling cycle is performed L times in the original place.

Note 4: Canned cycle command G110, G111, G112, G113, G114, G115, G134, G135, G136, G137, G138, G139 have no continuous drilling function.

Note 5: When L is specified, no drilling will be performed.
3.15.3.2 Rectangle series punch (G140 / G141)

Format:

G140
GxxGxx
G x X_ Y_ R_ Z_ F_

G141

Function: Performing series punch on each side of the rectangle according to the punch number specified.

Explanation:

G140: Punching in CW
G141: Punching in CCW
Gxx: Punching type (G73, G74, G81, G83, G84, G85, G86, G88, G89)
X, Y: End coordinate of the first rectangle side
R: R plane position
Z: Hole depth
A: The punching number on the 1st and 3rd side
B: The punching number on the 2nd and 4th side
J: The length of the 2nd side
F: Cutting feedrate

Related Parameter:

Bit 7 of the parameter 014
1: Hole positioning of serial punching is performed by cutting path (G01~G03).
0: Hole positioning of serial punching is performed by the rapid traverse path (G00).

For example:
The end point coordinate of the rectangle first side is X:90, Y:40; the length of the 2nd side is 20mm as for the rectangle path punching. The punching holes are machined by G81, to punch 3 holes at 1st and 3rd side each other; punch 2 holes at 2nd and 4th side each other, the hole depth is 25mm;

its programming is as follows:

G90 G17 G0 X0 Y0 Z25
M03
G140 G81 X90 Y40 R5 Z-25 A3 B2 J20 F800
G0 X100 Y100
M05
M30

There are 10 holes such as A1~A3, B4, B5, A6~A8, B9 and B10 to be machined as in above figure.

Note 1: If the G140 or G141 is specified in the canned cycle, it is indicated that the rectangle serial punching will be performed. The rectangle data are defined according to specified X, Y coordinates and F value in a program and the serial punching cycle is performed.
according to the punching mode canned cycle command.

Note 1: The command value of maximum punching number A and B at each side is 9999; the command is disabled when it is negative. The decimal part will be rounded off if the command is decimal; if the A or B is not specified then it is a default.

Note 3: The rectangle is defined by the current start point, the end of the 1st side and the length of the 2nd side; the default is current start point if the end of 1st side is not specified; the alarm will be generated if the length (namely the B is not specified) of 2nd side is not specified.

Note 4: The returned levels are all R point plane in serial punching, the corresponding plane will be retracted according to G98/G99 specified in a block when the last hole is performed.

Note 5: Canned cycles such as G110, G111, G112, G113, G114, G115, G134, G136, G137, G138 and G139 have no serial punching functions.

Note 6: The command words G140, G141, A, B and J are only effective in current block. The alarm will be generated if the G140 and G141 are specified without the canned cycle punching. The A, B and J will be ignored if A, B and K are specified instead of the G140 or G141.

3.15.3.3 Arc serial punching G142/G143

Format:

G142

G98 / G99 X_ Y_ R_ Z_ B_ (I_ J_) C_ F_

G143

Function: Serial punching is performed according to the specified punching number on specified arc.

Explanation:
- G142 - Punching in CW
- G143 - Punching in CCW
- G_x - Punching type (G3, G4, G81, G82, G83, G84, G88, G89)
- X Y Z - End point coordinate for the arc it is filled for G142 plane.
- R - R plane position
- Z - Hole depth
- B - Radius of arc when a negative value is specified it is major arc.
- I J - The circle center and radius are calculated by I or J when the R value is not specified.
- C - Number of punching
- F - Cutting feedrate

Related Parameter:
Bit 7 of the parameter 014
1: Hole positioning for serial punching is performed by cutting path (G01~G03).
0: Hole positioning for serial punching is performed by the rapid traverse path (G00).

For example:

G1 G12 G81 X100 R50 Z-50

Diagram:弧形串孔
Example 2: when drilling 7 holes in full circle, the start points and end points are coordinate origins, and the radius is 50, hole depth is 50.

O0001:
G00 G00 X0 Y0 Z0 G17:
G81 G12 G82 I50 J0 R-10 Z-50 F3000:
X30;
M30

Note 1: in continuous drilling, when the start point is identical to end point, no drilling will be performed.


Note 3: The maximum drilling number is 9999; the negative value is processed as absolute value; the decimals are rounded.

Note 4: when C is not specified or equals to 0, it reaches the end point directly and no drilling will be performed.

3.15. Precautions for canned cycle

(1) The spindle should be rotated (The M code should be correctly specified, or, the alarm will be generated, the G74 by M04, G84 by M03) by using the miscellaneous function (M code) before the canned cycle is executed.

(2) Specifying any command of the X, Y, Z and R data, the hole machining can be performed in the canned cycle of G3 to G8. If neither data is contained in the block, the hole machining is not performed (G110, G111, G112, G113, G114, G115, G134, G135, G136, G137, G138 and G139 are still needed to specify the corresponding address I, J and K, or the alarm occurs). But the hole machining is not performed when the G04 X_ is specified in the circumstance of X, because the X indicates for time when the G04 is specified.

G00 X_;
G81 X_ Y_ Z_ R_ F_ L_ ;     (Hole machining performs);
;  (Without hole machining)
F_ ;                          (F value is refreshed without the hole machining)
M_ ;                          (Performing the miscellaneous function only)

(3) When the canned cycle (G74 or G84) is employed in spindle rotation consolation, if the hole
position (X, Y) or distance from initial point level to the point R plane is short, and it is necessary to machine serially, or sometimes the spindle can not reach the specified speed before the hole machining operation, for delaying the time, the dwell block by G04 is inserted into each hole machining, which is shown as follows:

![Insert the dwell; wait for the spindle speed reaches to the normal value](image)

G86 X_ Y_ Z_ R_ F_;
G04 P_; (For dwell time P, without hole machining)
X_ Y_; (The next hole is machined)
G04 P_; (For dwell time P, without hole machining)
X_ Y_; (The next hole is machined)
G04 P_; (For dwell time P, without hole machining)

Sometimes, this issue will not be considered according to different machine tool, refer to the manual supplied by the machine tool builder.

(4) As stated above, the canned cycle can also be cancelled only when G00-G03 codes are read. So, there are two cases (expresses for 0-3, for canned cycle code) will be shown when they share the same block with the canned cycle G code.

G# GƑƑ X- Y- Z- R- Q- P- F- K-; (For canned cycle)
GƑƑ G# X- Y- Z- R- Q- P- F- K-; The X, Y and Z axes are moved by G#, the R, P, Q and K are disabled, the F is stored. The principle, which the last G code is effective when G codes of same group share the same block, is met by cases above.

(5) When the canned cycle and miscellaneous function are specified at the same block, The M and MF codes are delivered at the beginning of positioning (see the Fig.13.1 (A) for the operation 1). The next hole machining can be performed till the ending signal (FIN) occurs.

(6) When the canned cycle is applied, if the tool compensation C is current state, the tool compensation information C is then temporarily cancelled and saved; the tool compensation C status is restored when the canned cycle is cancelled.

(7) If the tool length offset commands (G43, G44 and G49) are specified in a canned cycle block. Then, the offset is performed when the point R plane is positioned (operation 2). The tool length offset commands are disabled after the canned cycle is entered till it is cancelled.
(8) The cautions for the operation of canned cycle:

a. Single block

When the canned cycle operation is performed by using the single block mode, normally, it is separately stopped at the terminal of the movements 1, 2, 3, 4, 5 and 6 in the Fig. 13.1 (A). And the single block is somewhat different according to corresponding canned cycle action at the bottom of a hole. For example, the single block is stopped when the dwell is applied. The operation at the bottom of the hole for fine-milling and rough-milling are divided into multiple single stop. So, it is necessary to startup for several times to machine a hole in a single block.

b. Feed hold

The feed hold is disabled between the movement 3 ~ 5 in commands G74 and G84, but the indicator of feed hold will light up. But the control stops till the operation 6. If the feed hold is performed again in operation 6, then it is stopped immediately.

c. Override

The feedrate override is considered for 100 percent in the operation G74 and G84, the override change is disabled.

(9) When the bit 1 of parameter 3 (D_R) is set to 1, the D value in tool compensation page indicates diameter value.

3.15.5 Examples for modal data specified in canned cycle

<table>
<thead>
<tr>
<th>No.</th>
<th>Data Specification</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>N0010</td>
<td>G00 X_ M3 ;</td>
<td>G00 positioning at the rapid traverse, and rotating the spindle;</td>
</tr>
<tr>
<td>N0020</td>
<td>G81 X_ Y_ Z_ R_ F_;</td>
<td>Because it is the beginning for the canned cycle, so the value needs to be specified for Z, R and F.</td>
</tr>
<tr>
<td>N0030</td>
<td>Y_;</td>
<td>The corresponding hole machining data is same to the previous hole, only the position Y is different, so G81Z_R_F_ can be omitted. As for the hole position is shifted for Y, hole machining is performed further by using the G81;</td>
</tr>
<tr>
<td>N0040</td>
<td>G82 X_ P_;</td>
<td>The hole position needs to be moved along the X axis as for the pervious one. The Z, R and F of previous hole and the P specified by this hole are taken as hole machining data by the G82;</td>
</tr>
<tr>
<td>N0050</td>
<td>G80 X_ Y_ M5 ;</td>
<td>The hole machining is not executed, all of the hole machining data are cancelled (except for the F); The GO positioning is performed with XY;</td>
</tr>
<tr>
<td>N0060</td>
<td>G85 X_ Z_ R_ P_;</td>
<td>The Z and R are needed to be specified newly because all of the data in previous block are cancelled, the above value specified is applied when the F is omitted. Although the P value is commanded, but it is not needed for this hole machining, so the P value is saved.</td>
</tr>
<tr>
<td>N0070</td>
<td>X_ Z_;</td>
<td>The Z is different compared with the previous hole, and the hole position just moves along the X axis;</td>
</tr>
<tr>
<td>N0080</td>
<td>G89 X_ Y_ D_;</td>
<td>The Z and R, P values separately specified by N0070 and N0060, the F value specified in N0020 are taken as hole machining data, which are used for G89 hole machining.</td>
</tr>
<tr>
<td>N0090</td>
<td>G112 I_ J_ F_ D_;</td>
<td>The fine-milling hole machined by G89 is performed by G112.</td>
</tr>
<tr>
<td>N0100</td>
<td>G0 X_ Y_ Z_;</td>
<td>positioning for a rectangle machining</td>
</tr>
<tr>
<td>N0110</td>
<td>G134 Z_R_I_J_K_U_D_:</td>
<td>Start machining the rectangle;</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>N0120</td>
<td>Y_I_J_K_U_D_:</td>
<td>Begins machining the second rectangle;</td>
</tr>
<tr>
<td>N0130</td>
<td>X_Y_I_J_K_U_D_:</td>
<td>Begins machining the 3rd rectangle;</td>
</tr>
<tr>
<td>N0140</td>
<td>G138 X_Y_R Z_I_</td>
<td>The fine-milling inside the machined rectangle groove is to be performed, the corresponding data are needed;</td>
</tr>
<tr>
<td></td>
<td>J_K_U_D_F_:</td>
<td></td>
</tr>
<tr>
<td>N0150</td>
<td>G01 X_Y_</td>
<td>Cancel the hole machining mode and data (except for F); the G01 cutting feed is performed by XY.</td>
</tr>
</tbody>
</table>

Note: Adress I, J, K and U of canned cycle G10, G111, G112, G113, G114, G115, G134, G135, G136, G137, G138 and G139 are not saved as canned cycle modal data, so the I, J and K values need to be specified in each block, or the alarm will be generated.

3.15. Examples for canned cycle and tool length compensation
The values of offset numbers H11, H15 and H31 are separately set to 200.0, 190.0 and 150.0, the program is as following:

<table>
<thead>
<tr>
<th>Line</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>N001</td>
<td>G92 X0 Y0 Z0 ;</td>
<td>The coordinate system is set at the reference point</td>
</tr>
<tr>
<td>N002</td>
<td>G90 G00 Z250.0 ;</td>
<td>Plane tool length compensation is performed at the initial plane.</td>
</tr>
<tr>
<td>N003</td>
<td>G43 Z0 H11 ;</td>
<td>The spindle starts.</td>
</tr>
<tr>
<td>N004</td>
<td>S30 M3</td>
<td>The spindle starts.</td>
</tr>
<tr>
<td>N005</td>
<td>G99 G81 X400.0 Y-350.0 Z-153.0 R-97.0 F120.0 ;</td>
<td>Hole 1 is machined after positioning.</td>
</tr>
<tr>
<td>N006</td>
<td>Y-550.0 ;</td>
<td>Hole 2 is machined after positioning, point R plane returned.</td>
</tr>
<tr>
<td>N007</td>
<td>G98 Y-750.0 ;</td>
<td>Hole 3 is machined after positioning, initial plane returned.</td>
</tr>
<tr>
<td>N008</td>
<td>G99 X1200.0 ;</td>
<td>Hole 4 is machined after positioning, point R plane returned.</td>
</tr>
<tr>
<td>N009</td>
<td>Y-550.0 ;</td>
<td>Hole 5 is machined after positioning, point R plane returned.</td>
</tr>
<tr>
<td>N010</td>
<td>G98 Y-350.0 ;</td>
<td>Hole 6 is machined after positioning, initial plane returned.</td>
</tr>
<tr>
<td>N011</td>
<td>G00 X0 Y0 M5</td>
<td>Reference point return, the spindle stops.</td>
</tr>
<tr>
<td>N012</td>
<td>G49 Z250.0 ;</td>
<td>Tool length compensation cancellation.</td>
</tr>
<tr>
<td>N013</td>
<td>G43 Z0 H15 ;</td>
<td>Initial plane, tool length compensation.</td>
</tr>
<tr>
<td>N014</td>
<td>S20 M3</td>
<td>Spindle starts.</td>
</tr>
<tr>
<td>N015</td>
<td>G99 G82 X550.0 Y-450.0 Z-130.0 R-97.0 P30 F70 ;</td>
<td>Hole 7 is machined after positioning, point R plane returned.</td>
</tr>
<tr>
<td>N016</td>
<td>G98 Y-650.0 ;</td>
<td>Hole 8 is machined after positioning, initial plane returned.</td>
</tr>
<tr>
<td>N017</td>
<td>G99 X1050.0 ;</td>
<td>Hole 9 is machined after positioning, point R plane returned.</td>
</tr>
<tr>
<td>N018</td>
<td>G98 Y-450.0 ;</td>
<td>Hole 10 is machined after positioning, initial plane returned.</td>
</tr>
<tr>
<td>N019</td>
<td>G00 X0 Y0 M5</td>
<td>Reference point return, the spindle stops.</td>
</tr>
<tr>
<td>N020</td>
<td>G49 Z250.0 ;</td>
<td>Tool length compensation cancellation.</td>
</tr>
<tr>
<td>N021</td>
<td>G43 Z0 H31 ;</td>
<td>Tool length compensation at initial plane.</td>
</tr>
<tr>
<td>N022</td>
<td>S10 M3</td>
<td>Spindle starts.</td>
</tr>
<tr>
<td>N023</td>
<td>G85 G99 X800.0 Y-350.0 Z-153.0 R47.0 F50 ;</td>
<td>Hole 11 is machined after positioning, point R plane returned.</td>
</tr>
<tr>
<td>N024</td>
<td>G91 Y-200.0 ;</td>
<td>Hole 12 and 13 are machined after positioning, point R plane returned.</td>
</tr>
<tr>
<td>N025</td>
<td>Y-200.0 ;</td>
<td></td>
</tr>
<tr>
<td>N026</td>
<td>G00 G90 X0 Y0 M5</td>
<td>Reference point return, the spindle stops.</td>
</tr>
<tr>
<td>N027</td>
<td>G49 Z0 ;</td>
<td>Tool length compensation cancellation.</td>
</tr>
<tr>
<td>N028</td>
<td>M30 ;</td>
<td>Program stops.</td>
</tr>
</tbody>
</table>
3.16 Absolute and Incremental Commands G90 and G91

**Format:**

- G90; Absolute command
- G91; Incremental command

**Function:**

There are two kinds of modes for commanding axis offset, one is absolute command the other is incremental command. The absolute command is programmed by coordinate value of the terminal position by the axis movement. The incremental command is directly programmed by the movement value of the axis. They are separately specified by G90 and G91 commands.

**Example:**

```
G90 X40.0 Y70.0;  or  G91 X-60.0 Y40.0;
```

3.17 Workpiece Coordinate System Setting G92

**Function:**

The workpiece coordinate system is set by setting the absolute coordinate in current position in the system (It is also called floating coordinate system). After the workpiece coordinate is set, the coordinate value is input in absolute programming in this coordinate system till the new workpiece coordinate system is set by G92.

**Command Explanation:**

- G92, which is a non-modal G-command;
- X: The new X axis absolute coordinate of current position;
- Y: The new Y axis absolute coordinate of current position;
- Z: The new Z axis absolute coordinate of current position;

**Note:** In the command, current coordinate value will not be changed if the X, Y and Z are not input. The program zero is set by the current coordinate value. If the X, Y or Z is not input, the coordinate axis not input keeps on the original set value.

3.18 Feed per min. G94, Feed per rev. G95

**Format:**

- G94 Fxxxx; (F0001 ~ F8000, the leading zero can be omitted, mm/min.)
### 3.19 G98, G99

#### Function:

- **G98**: Tool returns to the initial plane when the hole machining is returning. 
- **G99**: Tool returns to the point R plane when the hole machining is returning.

#### Explanation:

The feedrate value is set by system data parameter No.029. The feedrate value per rev is specified by the G95 Fxxxx command. The actual feedrate, which is controlled by the system, is less than the feedrate value per rev when the spindle speed is less than 1 r/min, and the feedrate value per min is regarded as the feedrate when the spindle speed is more than 1 r/min. The spindle feed value per rev can be specified by system parameter No.031, and it changes the feed value in one command. The conversion formula for feed rate per rev and per min is as following:

\[
F_{m} = F_{r} \times S
\]

Where:
- \(F_{m}\): Feed rate per min (mm/ min) 
- \(F_{r}\): Feed rate per rev (mm/ r) 
- \(S\): Spindle speed (r/min)

Note: The cutting feedrate becomes uneven when the spindle speed is less than 1 r/min, and the feedrate fluctuates in order to guarantee the machining quality. It is recommended to use a lower spindle speed when the spindle encoder is not installed.

The feedrate value is invariable after the F command is performed. The feedrate is 0 after the F0 command is executed. The F value is invariable when the system is reset or emergency stop.

The feed override is memorized when the power is turned off.

#### Related parameters:

- System data parameter No.029: exponential acceleration or deceleration time constant for cutting and manual feed.
- System data parameter No.030: lower limit value of exponential acceleration or deceleration on cutting feed.
- System data parameter No.031: upper limit value for cutting feedrate (X, Y and Z axes)

Note: The cutting feedrate becomes uneven when the spindle speed is less than 1 r/min. The actual feedrate varies when the spindle speed fluctuates. In order to guarantee the machining quality, it is recommended that the spindle speed be higher than the spindle servo or the lowest speed of effective torque introduced by the inverter during machining.
Modal G command

<table>
<thead>
<tr>
<th>G98</th>
<th>G99</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Return to initial plane)</td>
<td>(Return to point R plane)</td>
</tr>
</tbody>
</table>

Refer to the explanation for canned cycle command.

3.20 Chamfering Function

A straight line or an arc is inserted into two figures; this is called Chamfering function. The tool can be smoothly transferred from one figure to another. GSK980MD owns two chamfering functions, one is linear chamfering, and the other is arc chamfering.

3.20.1 Linear chamfering

The linear chamfering is that a straight line is inserted between figures of the straight lines, the arcs, as well as the straight line and arc. The command address for linear chamfering is \( L \). The data followed by command address \( L \) is the length of chamfering straight line. The linear chamfering should be employed in the G01, G02 or G03 command.

- \( \text{linear to linear} \)
  - Format: \( \text{G01 } \text{IP}_- \text{L}_-; \) (IP is axis movement command)
  - Function: A straight line is inserted into interpolation between 2 straight lines.

- \( \text{arc to linear} \)
  - Format: \( \text{G01 } \text{IP}_- \text{A}_-; \) (IP is axis movement command)
  - Function: A straight line is inserted into interpolation between an arc and a straight line.
**Linear to Circular**

**Format:**

G01 IP"

G02/G03 IP_ R_(I_ J_ K_);"

**Function:** A straight line is inserted between straight line and arc interpolation.

**Circular to Circular**

**Format:**

G02/G03 IP_ R_(I_ J_ K_) L_"

G02/G03 IP_ R_(I_ J_ K_);"

**Function:** A straight line is inserted between two arc interpolations.

**Circular to Linear**

**Format:**

G02/G03 IP_ R_(I_ J_ K_);"

G01 IP_;

**Function:** A straight line is inserted between the arc and linear interpolation.
3. Circular Chamfering

An arc is inserted between the two linear figures, arc figures or linear and arc figures, this is called circular chamfering. Tangent transition is performed between arc and figure line. The command address is C for the arc chamfering, the data followed by command address C is the radius of chamfering arc. The arc chamfering should be employed in command G01, G02 or G03.

- 1. Linear to Linear

  **Format:**
  
  G01  IP_  C_;
  
  **Function:** An arc is inserted between two linear interpolations, which it is tangential with two linear lines, the data followed by command address C is radius.

- 2. Linear to Circular

  **Format:**
  
  G01  IP_  C_;
  
  G02/G03  IP_  R_(I_  J_  K_); 
  
  **Function:** An arc is inserted at the intersection of straight line and arc, this arc is tangential with both the straight line and arc, the data followed by command address C is radius.
• 3. Circular to Circular

Format:

\[G02/G03 \ IP_ \ R_\(l_ \ J_ \ K_\) \ C_\;\]

Function: An arc is inserted between two arc interpolations which it is tangential with two circulars, the data followed by the command address C is radius.

• 4. Circular to Linear

Format:

\[G02/G03 \ IP_ \ R_\(l_ \ J_ \ K_\) \ C_\;\]
\[G01 \ IP_\;\]

Function: An arc is inserted at the intersection of arc and straight line, which is tangential with the arc and straight line; the data following the command address C is radius.
3.2.3 Exceptional Cases

The chamfering function is ineffective or alarm is issued in the following circumstances:

1. Linear Chamfering
   - The chamfering function is ineffective when two interpolation lines is shown on the same line.
   - If the chamfering linear length is too long, and the CNC alarm occurs.

   ![Linear Chamfering Diagram]

   - If some line (arc) is too short, the alarm occurs.

   ![Short Line Diagram]

2. Arc Chamfering
   - The arc chamfering function is disabled when two interpolation lines are shown on the same line.
   - If the chamfering radius is excessive, the CNC alarm occurs.
C. The arc chamfering function is disabled when the line is tangential with arc or the arc is tangential with line.

D. The arc chamfering function is disabled when the arcs are tangent.

Note 1: The chamfering function can be performed only in the plane specified by G17, G18, or G19; these functions cannot be performed in parallel axes.

Note 2: Changing the coordinate system by G92 or G54 to G59, or the block followed by performing the reference point return from G28 to G30 cannot specify the chamfering.

Note 3: Chamfering function cannot be employed in the DNC mode.

3.21 Rigid Tapping

The right-handed tapping cycle (G84) and left-handed tapping cycle (G74) may be performed in standard mode or rigid tapping mode. In standard mode, the spindle is rotated and stopped along with a movement along the tapping axis using miscellaneous functions M03 (rotating the spindle clockwise), M04 (rotating the spindle counterclockwise), and M05 (stopping the spindle) to perform tapping.

In rigid mode, tapping is performed by controlling the spindle motor as if it were a servo motor and by interpolating between the tapping axis and spindle. When tapping is performed in rigid mode, the spindle rotates one turn every time a certain feed (thread lead) which takes place along the tapping axis. This operation does not vary even during acceleration or deceleration.

3.21.1 Rigid Tapping

Code format:
- Left-handed rigid tapping: G74 X_ Y_ Z_ R_ P_ F (1) _ C_
Right-handed rigid tapping: G84 X_ Y_ Z_ R_ P_ F (1) _ □ _ C_

**Code function**: In rigid mode, tapping is performed by controlling the spindle motor as if it were a servo motor and by interpolating between the tapping axis and spindle. When tapping is performed in rigid mode, the spindle rotates one turn every time a certain feed (thread lead) which takes place along the tapping axis. This operation does not vary even during acceleration or deceleration.

**Cycle process**: (1) Position to the XY plane at the rapid traverse rate;
(2) Reducing to the point R plane rapidly, then to the position where the C is specified at the rapid traverse rate;
(3) Tapping is performed to the bottom of the hole, then the spindle stops;
(4) Dwell time P is performed if the P is specified;
(5) Spindle rotates reversely returns to the point R plane, the spindle then stops; dwell time P is performed if the P is specified;
(6) Return to the origin plane if the command is G98;

**Code path**: (G74 shows a sample)

<table>
<thead>
<tr>
<th>G74 (G98)</th>
<th>G74 (G99)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Diagram" /></td>
<td><img src="image2" alt="Diagram" /></td>
</tr>
</tbody>
</table>

**Explanations**: When the tapping operation 3 is being performed, the feedrate override can not be adjusted; when the operation 5 is performing, the speed override value is set by the data parameter 084, when the data parameter 084 is set to 0, the override value is fixed as 100%.

When the tapping operation 3 is being performed, the linear acceleration or deceleration constant value is set by the data parameter 082; when the tapping operation 5 is performed, the linear acceleration constant value is set by data parameter 083, if the data parameter 083 is set to 0, the linear acceleration/deceleration time constant in operation 5 is set by the data parameter 082.

**3.2.1. Peck Rigid Tapping**

**Code format**: (High-speed/standard) peck left-handed rigid tapping: G74 X_ Y_ Z_ R_ P_ F (1) _ □ _ □ _ C_
(High-speed/standard) peck right-handed rigid tapping: G84 X_ Y_ Z_ R_ P_ F (1) _ □ _ □ _ C_

**Code function**: When the peck tapping is performed in rigid tapping, due to chips sticking to the tool or increased cutting resistance, in such cases, the preferable tapping can be performed by the peck rigid tapping.
High-speed peck rigid tapping:

When the RTPCP of state parameter No.025 is set to 1, the high-speed peck rigid tapping cycle is selected.

After positioning along the X- and Y-axes, rapid traverse is performed to point R, then position to the place where specifies by C. From point R, cutting is performed with depth \( Q \) (depth of cut for each cutting feed), then the tool is retracted by distance \( d \), the retraction speed can be overridden. When point Z has been reached, the spindle is stopped, and then rotated in the reverse direction for retraction. The tool retracts to the point R, the spindle stops. If it is G98 state, rapidly move to the initial position, the Figure is shown below:

<table>
<thead>
<tr>
<th>G74, G84 ( (G98) )</th>
<th>G74, G84 ( (G99) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d=回退距離 )</td>
<td>( d=回退距離 )</td>
</tr>
<tr>
<td>主軸定位</td>
<td>主軸定位</td>
</tr>
<tr>
<td>R点</td>
<td>R点</td>
</tr>
<tr>
<td>(1)</td>
<td>(1)</td>
</tr>
<tr>
<td>(2)</td>
<td>(2)</td>
</tr>
<tr>
<td>(3)</td>
<td>(3)</td>
</tr>
<tr>
<td>(4)</td>
<td>(4)</td>
</tr>
<tr>
<td>Z点</td>
<td>Z点</td>
</tr>
</tbody>
</table>

Standard peck rigid tapping:

When the RTPCP of state parameter No.025 is set to 1, the standard peck rigid tapping cycle is selected.

After positioning along the X- and Y-axes, rapid traverse is performed to point R, then position to the place where specifies by C. From point R, cutting is performed with depth \( Q \) (depth of cut for each cutting feed), then the tool is retracted by distance \( d \), the retraction speed can be overridden. The position is performed from point R to a distance \( d \) from the end of the last cutting, which is where cutting is restarted, and the cutting feed is performed. When point Z has been reached, the spindle is stopped, then rotated in the reverse direction for retraction. The tool retracts to the point R, the spindle stops. If it is G98 state, rapidly move to the initial position, the Figure is shown below:

<table>
<thead>
<tr>
<th>G74, G84 ( (G98) )</th>
<th>G74, G84 ( (G99) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d=切削開始距離 )</td>
<td>( d=切削開始距離 )</td>
</tr>
<tr>
<td>主軸定位</td>
<td>主軸定位</td>
</tr>
<tr>
<td>Z点</td>
<td>Z点</td>
</tr>
<tr>
<td>(1)</td>
<td>(1)</td>
</tr>
<tr>
<td>(2)</td>
<td>(2)</td>
</tr>
<tr>
<td>(3)</td>
<td>(3)</td>
</tr>
<tr>
<td>(4)</td>
<td>(4)</td>
</tr>
<tr>
<td>R点</td>
<td>R点</td>
</tr>
</tbody>
</table>

Explanations:

When tapping feed is performing, the speed override cannot be adjusted; when the retraction is
performed, the speed override value is set by data parameter 084, when the data parameter 084 is set to 0, the override value is fixed as 100.

The linear acceleration or deceleration constant value in tapping feed is set by data parameter 082, the linear acceleration or deceleration constant in retraction is set by data parameter 083, if the 083 is set to 0, the acceleration or deceleration constant in retraction is then set by data parameter 082. The start speed both tapping feed and retraction are set by data parameter 081, and the retraction distance d is set by data parameter 085.

### 3.2.1.3 Address Explanation

<table>
<thead>
<tr>
<th>Specified content</th>
<th>Address</th>
<th>Command address explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hole position</td>
<td>X, Y</td>
<td>Specify the hole position by the absolute value or incremental</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>From the initial plane to the point distance</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>Depth of a hole, the distance from point R to the bottom of the hole</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>Specify the dwell time at the bottom of the hole or at point R when a return is made. The dwell does not perform when it is not input or the value is 0.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tool infeed value of peck tapping</td>
</tr>
<tr>
<td></td>
<td></td>
<td>It indicates that the consecutive machining cycle of holes are performed on this line segment from start (the start position of block) to XY coordinate position. The continued drilling may not perform if it is not input or the value is 0.</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>Metric thread leading, the solution range: 0.001~500mm. The alarm 201 may alarm if it is not input.</td>
</tr>
<tr>
<td></td>
<td>I</td>
<td>The number of the thread head per inch, the solution range is 0.06~25400 gear/inch</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>Start angle</td>
</tr>
</tbody>
</table>

### 3.2.1.4 Technical Specification

- **Acceleration/deceleration**
  Rigid tapping adopts the acceleration or deceleration before a straight line to control.

- **Override**
  The override regulation is invalid for rigid tapping infeed, but the override value can be adjusted or not which is determined by data parameter.

- **Dry run**
  G84:G74 can be used a dry run, the dry run equals to the feedrate along Z axis. The override adjustment is invalid in dry run.

- **Machine lock**
  G84:G74 can be used a machine lock, the tapping axis and spindle axis are not moved when the machine lock is enabled.

- **Resetting**
  The resetting can be reset the tapping when the rigid tapping is performed, but the G74:G84 can be not be reset.

- **Dwell**
  The dwell is disabled.

- **Working**
  G84:G74 is only valid in Auto or MDI mode.
3.1.5 Specify a Rigid Tapping Mode

- Specify M29 before G74/G84
  G84 shows a sample for the following time-sequence
Specify M29 and G74 G84 at the same block.

G84 shows a sample for the following time-sequence:

The explanation of time sequence:
The spindle rotation operation means that the rotation axis is shifted to the position control mode (namely, the servo spindle is needed to send a switch signal in position mode), and check the position mode arrival signal of servo spindle.

3.1. **Cancellation of rigid tapping mode**

- The rigid tapping mode is canceled by G80
- Specify other canned cycles by G codes
- The other G codes of group 1.
- CNC resetting
The signal descending of F76.3 along the signal with canceling the rigid tapping of PLC, if the state RTCRG of parameter 025 is equal to 1, the system is then performed the next block without waiting for the rigid tapping mode signal which G61.0 is set to 0;

When the state parameter 025.2 (CRG) = 0, the time sequence is as follows:

- G80 or 01 group G code
- RTAP (F76.3)
- S 指令值输出
- RGTAP (G61.0)
- 主轴旋转动作
- 主轴转动信号

When the state parameter 025.2 (CRG) = 1, the time sequence is as follows:

- G80 or 01 group G code
- RTAP (F76.3)
- S 指令值输出
- RGTAP (G61.0)
- 主轴旋转动作
- 主轴转动信号

### 3.2 1.7 F and G Signals

**RGTAP (G61.0):** Rigid tapping signal

When the M29 is commanded, PMC enters the rigid tapping mode, and the signal is then set to 1 to inform the CNC

- 1: PMC enters the rigid tapping mode
- 0: PMC does not enter the rigid tapping mode

If this signal does not set to 1, after the M29 has been commanded, the alarm may occur in the block of G74/G84.

**RGSPM, RGSPP (F65.1, 0) spindle turning signal**

When the rigid tapping is performed, the signal is informed to the PMC whether the current spindle is CCW (positive) or CW (negative).

- RGSPM: 1 spindle CW (negative)  
- RGSPP: 1 spindle CCW (positive)

In rigid tapping, these signals are output when the spindle is rotated. In the mode of rigid tapping, when the spindle is positioned at the hole or stopped at the bottom of the hole or R position, these signals are not output.

In the mode of rigid tapping, when the spindle is positioned at the inter-locked stop, machine lock or Z axis ignorance states, the spindle does not regard as a stop state, in this case, these signals are output. These signals are only enabled in rigid tapping, and they are all set to 0 in the normal spindle control mode.
RTAP (F76.3): Rigid tapping process signal

This signal informs PMC which has been in the mode of rigid tapping or not. The CNC is in the mode of rigid tapping currently when the signal is set to 1.

This signal can be locked M29, P:\C has been commanded the rigid tapping mode, the PMC is then treated with the correspinding logic, and this signal can be replaced the lock of M29, even so, the FIN singl of M29 is not ignored still.

### 3.2.1. Alarm Message

<table>
<thead>
<tr>
<th>Alarm No.</th>
<th>Display Content</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>218</td>
<td>Fail to specify the tool pitch F value in G74 or G84</td>
<td>Fail to specify F value</td>
</tr>
<tr>
<td>230</td>
<td>The spindle feed can not be performed due to the S value is 0.</td>
<td>S value is 0, or S code does not specify.</td>
</tr>
<tr>
<td>231</td>
<td>S value exceeds the maximum spindle speed allowed with rigid tapping</td>
<td>S value exceeds the setting value of data parameter 086</td>
</tr>
<tr>
<td>232</td>
<td>Other axis movement codes are specified between M29 and G74:G84.</td>
<td>Specify a axis movement between M29 and G74:G84.</td>
</tr>
<tr>
<td>233</td>
<td>G61.0 signal is abnormal in rigid tapping mode</td>
<td>Rigid tapping signal G61.0 is not 1 during performing in G74:G84.</td>
</tr>
<tr>
<td>234</td>
<td>Specify M29 repeatedly</td>
<td>Specify M29 or it is consecutively specified more than twice in rigid tapping.</td>
</tr>
</tbody>
</table>

### 3.2.1. Program Example

G84 shows an example for the following program

```
O1000 (Rigid tapping example);
G0 X0 Y0 Z0;
M29 S200;
G84 X10 Y10 Z-10 R-5 P2000 F2 C20;
X20 C40
G80;
M30;
```
CHAPTER 4 CONTROL FUNCTION of ADDITIONAL AXIS

4.1 General

The additional axis is determined by the struction design of the machine, sometimes, an additional axis is required, for example, the cycle working table, rotation working table. This axis can be designed as both a linear axis and rotation axis. The basis controllable number of 980MDa is three axes, the maximum axis is 5-axis (Cs axis included). Namely, two additional axes are added based upon the original one — the 4th and the 5th axes, in this case, the relative functions of additional linear axis and rotation axis can be performed.

4.2 Axis Name

The names of three basis axes are always X, Y or Z. The axis name of additional axis can be set to A, B or C using data parameter No.202 and No.203.

- Default axis name
  When the axis name does not set, the axis name of the 4th one is an additional axis by default; the axis name of the 5th one is C.

- Repeated axis name
  When the axis name is same between the added 4th axis and the 5th axis, P/S alarm may issue.

4.3 Axis Display

When the additional axis is treated as rotation axis, the least incremental of the rotation axis is 0.01° (degree), so the 3rd digit of the decimal is displayed in unit. If it is set to a linear axis, the display is same as the basis three axes (X, Y or Z). When the 4th axis is set to a linear axis, the 5th is set to a rotation axis, the axis is displayed at the interface of related coordinate and coordinate & program.

<table>
<thead>
<tr>
<th>相对坐标</th>
<th>O0000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.000</td>
</tr>
<tr>
<td>Y</td>
<td>0.000</td>
</tr>
<tr>
<td>Z</td>
<td>0.000</td>
</tr>
<tr>
<td>A</td>
<td>0.000</td>
</tr>
<tr>
<td>C</td>
<td>0.00°</td>
</tr>
<tr>
<td></td>
<td>G00 G17 G90 G54</td>
</tr>
<tr>
<td></td>
<td>G21 G40 G49 G94 G98</td>
</tr>
<tr>
<td></td>
<td>P0100 S 00 M30</td>
</tr>
</tbody>
</table>

编程速率: 100
实际速率: 0
进给倍率: 150%
快速倍率: 100%
主轴倍率: 100%
加工件数: 0
切削时间: 0:00:00

S0000 T00 H00
4.4 Axis Startup

The Bit 1 (ROSx) of data parameter No.026 and Bit0 (ROTx) of data parameter No.028 are separately set to use whether the 4th axis and the 5th axis is either the linear axis or rotation axis. The parameter settings are shown below:

<table>
<thead>
<tr>
<th>ROS</th>
<th>ROT</th>
<th>Content</th>
</tr>
</thead>
</table>
| 0   | 0   | Linear axis  
1. It can be switched between metric and inch;  
2. All of the coordinate values are linear axis;  
3. The stored pitch error compensation is linear axis. |
| 1   |     | Rotation axis (Type A)  
1. It can not be switched between metric and inch;  
2. The machine coordinates are cycled based on the setting value of data parameter No.189 No.190. Whether the absolute coordinate and relative coordinate are cycled which based upon the data parameter No.027 No.029;  
3. The stored pitch error compensation is rotation axis;  
4. The movement amount is less than one turn when the reference position (G28, G30) is returned. |
| 0   | 1   | Ineffective setting (forbidden) |
| 1   | 0   | Rotation axis (Type B)  
1. It can not be switched between metric and inch;  
2. The machine coordinate is linear axis; whether the absolute coordinate and relative coordinate are cycled which based on the data parameter No.027 No.029.  
3. The stored pitch error compensation is linear axis. |

Note: the start of the function of the Cs axis the Bit 5 digits RCSx of the state parameter No.026 or No.028 can be set whether the function of Cs axis is enabled when the rotation axis is enabled ROSx 1.

4.5 The Additional Axis is Linear Axis

When the additional axes (the 4th and the 5th axes) are set to linear axes, its functions are same as the basis three axes.

- Realizable operation
  1. Rapid traverse (Positioning): G90/91 G00 X_ Y_ Z_ A_;  
  2. Cutting feed: G90/91 G01 X_ Y_ Z_ A_ F_;  
  3. Skip function: G90/91 G31 X_ Y_ Z_ A_ F_;
4. Reference position return: G28 29 30 X_ Y_ Z_ A_ F_; 
5. G92 coordinate setting: G92 X_ Y_ Z_ A_; 

Note: When there is no special explanation in the subsequent narration, the axis names of additional linear axes are expressed as "A".

- **Explanations**

  1. When the additional linear axis rapidly moves or performs, it can be simultaneously specified with any axes of X, Y and Z. Each axis may rapidly move at its customizable speed.

  2. When the additional linear axis is performed the cutting feed (G01) or used a skip function (G31), it can be simultaneously specified with any axes of X, Y and Z. In this case, the linear axis does not have an individual feedrate F but depend on each axis specified at the same time, which it is started or ended together with the specified each axis; namely, the additional axis is shared with the basis three-axis linkage.

  3. The additional linear axis cannot be perform a circular arc cutting (G02/03), otherwise, the P/S alarm may occur.

  4. The pitch error of additional linear axis and the compensation function of inverse interval are same as the basis three-axis.

4.6 The additional axis is rotation axis

- **Input unit**

  The pulse equivalence (namally, the least input unit) of 980MDa rotation axis is 0.01° (degree); the maximum value of output pulse frequency is 500K.

  When the selection is output based on the direction of pulse adding, it can be inputted a maximum speed n = 60 * f / 3600 = 833.33 (rev./min.).

- **Rotation axis speed**

  The feedrate of rotation axis is regarded the degree/min. as a unit. When the linear axis X, Y and Z is performed a linear interpolation with the rotation axis, the speed specified with F (mm/min) is the compound feedrate both X, Y and Z and the rotation axis.

  Feedrate calculation: Calculate the required time when the feedrate is performed to the end; then, the feedrate unit of rotation axis is changed into degree/min.

  For example: G91 G01 X20.0 C40.0 F300.0;

  The unit of C axis is switched into 40mm from the 40.0 degree. The required time to the end is:

  \[
  \frac{\sqrt{10^2 + 40^2}}{300} = 22.5032 (\text{min.})
  \]

  The speed of C axis is:

  \[
  \frac{40}{22.5032} = 1.7673 (\text{degree/min.})
  \]

  Note: When there is no special explanation in the subsequent narration, the axis names of additional linear axes are expressed as "C".

- **Cycle function of rotation axis**

  The coordinate cycle function of the additional rotation axis setting is enabled, which can be
avoided the coordinate value is overflowed from the rotation axis; the coordinate value will be cycled based on the setting value of data parameter No.189/No.190 (the movement amount of each axis for the rotation axis).

When the coordinate cycle function of the additional rotation axis setting is disabled, the coordinate value may change based on the linear axis, the programming command is also same to the one of the linear axis;

Two kinds of coordinates change are shown below:
(1) When the coordinate cycle is disabled:

![Coordinate cycle disabled diagram]

The above-mentioned may occur: 1. The machine coordinate value of rotation axis (Type B)
2. The absolute coordinate value in data parameter No.027 ROAx= 0 (absolute coordinate cycle function is disabled)
3. The relative coordinate value in data parameter No.027 RRL x= 0 (relative coordinate cycle function is disabled)

(2) When the coordinate cycle is enabled:

![Coordinate cycle enabled diagram]

The above-mentioned may occur: 1. The machine coordinate value of rotation axis (Type A)
2. The absolute coordinate value in data parameter No.027 ROAx= 1 (absolute coordinate cycle function is enabled)
3. The relative coordinate value in data parameter No.027 RRL x= 1 (relative coordinate cycle function is enabled)

Note 1: Refer to the Section of “Installation and connection” of the Parameter Explanation of Chapter Three for the parameter setting of additional rotation axis.

Note 2: When there is no special explanation in the subsequent narration, the movement amount of each revolution of the additional rotation axis is expressed with 360°.

- **The pitch error compensation function of rotation axis**
  When the additional axis is a linear axis or rotation axis (Type B), the pitch error compensation mode is same as the common linear axis. The pitch error compensation function is performed when the additional axis is regarded as rotation axis (Type A), refer to the following examples:

  - Movement amount per revolution: 360°
  - Pitch error position interval: 45°
  - The compensation position number of reference position: 60

  After the above parameters are set, the farthest compensation position number along the negative rotation axis which equals to the compensation position number of reference position;

  The farthest compensation number along positive direction is shown below:

  The compensation position number of reference point + (movement amount per revolution/compensation position interval) = 60 + 360/45 = 68;

  The corresponding relationships between machine coordinate and compensation position number are as follows:
The position error may occur if the total of compensation value from position 60 to 68 is not 0; there is no alternative other than to set the same value at the compensation position both 60 and 68. (Because the 60 and 68 are shared at the same position in the circle);

The compensation sample is shown below:

<table>
<thead>
<tr>
<th>NO.</th>
<th>60</th>
<th>61</th>
<th>62</th>
<th>63</th>
<th>64</th>
<th>65</th>
<th>66</th>
<th>67</th>
<th>68</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compensation value</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>3</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
</tbody>
</table>

- The reverse interval compensation function of rotation axis
  The reverse interval compensation never changes regardless of the linear axis or rotation axis; however, the compensation unit of the rotation axis is 0.01° (deg), and the linear axis is 0.001 (mm);

4. The zero return of rotation axis

The selection axis has four zero return methods: zero return method A, B, C and D. Wherein, the zero return methods A, B and C are the same as the one of the linear axis. Only the D is a special zero return method for the rotation axis.

- Setting of the zero return method D
The method is only valid to the rotation axis. 
Zero return can be performed for this rotation axis using the mode after the 4th and the 5th axes are set to rotation axes based on the Bit6 of data parameter 0.00 and 0.00 are set to 0 if the 4th and 5th axes are disabled or linear axes, then the Bit6 of state parameter 0.00 and 0.00 are invalid.

027 RRT4

RRT4 = 1: The zero return mode of the 4th rotation axis is used the mode D;
0: The zero return mode of the 4th rotation axis is used the mode A, B, and C.

029 RRT5

RRT5 = 1: The zero return mode of the 5th rotation axis is used the mode D;
0: The zero return mode of the 5th rotation axis is used the mode A, B and C.

The time sequence and process of the zero return mode

1. Select the machine zero return mode and press the manual positive feed key, the corresponding axis moves toward the zero point at the rapid traverse rate.
2. When the one-turn signal (PC) of servo axis is carried out, the system is decelerated to the zero return low speed, in this case, check the trailing edge of PC signal.
3. The system continuously and forward operates in the zero return low speed.
4. When the system meets one-turn signal (PC) of servo axis again, the movement stops, simultaneously, the corresponding indicator of zero return end on operator panel goes on. The machine zero return operation ends. In this case, check the rising edge of PC signal.

4.8 The Function of Cs Axis

General

The spindle is treated as the servo feed axis to rotate and position by the position movement command. Run speed is: degree/min., it can be interpolated together with other feed axes to machine a contour curve.

Increment system: the least input increment: 0.00 deg

The least command increment: 0.00 deg

Explanation: C has two control modes for the spindle.
- Spindle speed control mode. The spindle speed can be controlled by the speed command (namely, analog voltage).
- Spindle contour control mode (it is also called Cs contour control). The spindle position can be controlled by the position command (namely, position pulse).

0. Cs is required the spindle servo control unit has two control modes for the control of the spindle motor

- When Cs is at the speed control mode for the control of the spindle, the spindle servo control unit can receive a speed command issued from Cs to control the rotation speed of spindle motor.
- When Cs is at the contour control mode for the control of the spindle, the spindle servo drive unit also can receive a position command issued from Cs to control the motor operates to a specified position.

**Set Cs contour control axis**

In the 80MDa system, only the additional axis (the 4th or the 5th axis) can be set to a Cs contour control axis. But, two Cs axes cannot be set at the same time. Before the Cs axis setting is valid, this axis must be set to a rotation axis. Otherwise, Cs axis setting is invalid.

<table>
<thead>
<tr>
<th>Rotation axis</th>
<th>Type A rotation axis</th>
<th>Type B rotation axis</th>
<th>Invalid</th>
</tr>
</thead>
<tbody>
<tr>
<td>R  S  T  0</td>
<td>0  0  0</td>
<td>0  0  0</td>
<td>0</td>
</tr>
</tbody>
</table>

- C s4 =: The Cs axis function of the 4th axis is enabled;
  - 0: The Cs axis function of the 4th axis is disabled.
- C s4, C sT4: Set the type of the 4th axis;

<table>
<thead>
<tr>
<th>Rotation axis</th>
<th>Type A rotation axis</th>
<th>Type B rotation axis</th>
<th>Invalid</th>
</tr>
</thead>
<tbody>
<tr>
<td>R  S  T  0</td>
<td>0  0  0</td>
<td>0  0  0</td>
<td>0</td>
</tr>
</tbody>
</table>

- C s5 =: The Cs axis function of the 5th axis is enabled.
=0: The C轴 axis function of the 5th axis is disabled.

RO5, ROT5: Set the type of the 5th axis;

<table>
<thead>
<tr>
<th>Linear axis</th>
<th>Type A rotation axis</th>
<th>Type B rotation axis</th>
<th>Invalid</th>
</tr>
</thead>
<tbody>
<tr>
<td>R::T</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>R::S</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

The switch between spindle speed control and Cs contour control

The Cs C switching of spindle control mode is performed by the Cs C signal of P C.

in the Cs C contour control mode of Cs C, the Cs C contour control axis, as the common servo axis, can be performed manually or automatically.

- From spindle speed control shifts to the Cs contour control
  - Set the Cs C (000000) to 1, then the spindle can be set in the Cs contour control mode. If the switch is performed during the spindle rotation, the spindle is immediately stopped and then shifts.

- From Cs contour control shifts to the spindle speed control
  - Set the Cs C (000000) to 0, the spindle is then set in the spindle speed control mode. Confirm the spindle movement command has been ended before shifting, if the shift is performed when the spindle is being moved, the system will alarm.

The reference position return of Cs contour control axis

After the spindle is shifted to the Cs contour control mode from the speed control mode, the current position is not confirmed, the spindle should be returned to the reference position.

The reference position return of Cs contour control axis is as follows:

- Manual reference position return
  - After the spindle enters the Cs contour control mode, shift to the machine zero return mode. The zero return of Cs axis is performed opening the feed axis and the direction selection signal +Jn (0000) or -Jn (0000).

- Automatic
  - Specify 8 after the spindle enters the Cs contour control mode, and the spindle moves to the intermediate point and then return to the reference position.
  - Pn (0404) becomes 1 after the reference position return is executed.

The operation of Cs contour control axis

- Manual/ Automatic
  - If the Cs contour control axis has been returned to the reference position, the operation of Cs axis is same as the common C axis.
  - In the spindle speed control, the Cs contour control axis can not be performed.
  - Otherwise, the system alarms.
  - In the spindle speed control mode, it is not permitted the manual operation of Cs by the P C ladder diagram.

The signal shift of spindle contour control

N021

- Type signal input
- Function This signal is used for shifting between spindle speed control mode
and Cs contour control mode.
When this signal is set to 1, the spindle is shifted to the Cs contour
control mode from speed control mode.
When this signal is set to 0, the Cs contour control mode
comes back to the speed control mode.

### The signal shift end of spindle contour control

<table>
<thead>
<tr>
<th>$S_{SCSL} (F044#1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
</tr>
<tr>
<td><strong>Function</strong></td>
</tr>
</tbody>
</table>
| **Output condition** | Spindle speed control mode $\rightarrow 0$
Cs contour control mode $\rightarrow 1$ |

**N** and spindle servo control unit

**The signal shift relationship of the spindle working**

**Time sequence figure**

- Input shift by the user
- The signal input of spindle servo working
- The signal output of spindle servo working
- The spindle servo shifts in working mode
- The spindle servo shifts in working mode
- The spindle servo at the position mode
- The spindle servo working

**CNC system**

**Spindle servo controller**

**User shifts and inputs**

**for the spindle working**
Relative parameter

- The start speed of acceleration/deceleration of CS axis
  - Resolution range: 0 ～ 5000 (unit: deg/min)

- The acceleration/deceleration time constant of CS axis
  - Resolution range: 0 ～ 4000 (unit: ms)

- The explanation of “two points same”
  - Radius compensation mode is pre-read two blocks. Calculate the transit point and perform a path movement taking 3 position points (the start of the 1st block, the intersection of the 1st and the 2nd blocks, the end of the 3rd block). In this case, two same points may occur in the following items:
    - (a) The first two points are same when starting.
    - (b) The last two points are same when starting.
    - (c) The first two points are same during the compensation.
    - (d) The last two points are same during the compensation.
    - (e) The first two points are same during the retraction.
    - (f) The last two points are same during the retraction.
  - The two same points is regarded the point as a linear of which approximates to zero, when the two same points occurs, the transit point calculation can be performed based on the straight line (point) to straight line (point), straight line (point) to circular arc (point), circular arc (point) to straight line (point) and circular arc (point) to circular arc (point).
GSK9 80MDa provides macro programs which is similar to high level language. Variable assignment, arithmetic operation, logical judgment and conditional branch can be realized through custom macro program. It is in favor of the programming for special parts, lessens the complex operation and simplifies the custom program.

Custom macro programs are similar to subprograms. However, macro program allows variable assignment, arithmetic operation, logical judgment and conditional branch, which makes it easier to program the same machining process.

It is easy to machine the screw holes distributed in circles (shown in the figure above).
After a macro program used in circular holes is programmed and edited, it can be performed if the NC system has circular hole machining function.
By the following command, programming personnel can use circular holes function.

```
G65  P p  R r  A a  B b  K k
```

- `p`: Macro program number of circular holes
- `r`: Radius
- `a`: Start angle of the hole
- `b`: Angle of holes intervals
- `k`: Holes number

In this way, users can improve the NC performance on their own. Macro programs can be either provided by machine tool builder or defined by users.
5. Macro Call

Macro call (G65, G66) differs from subprogram call (M98) as described below:

1. With G65 or G66, an argument (data passed to a macro) can be specified. M98 does not have this capability.
2. When an M98 block contains another NC command (for example, 00.000.0 M98 P), the macro program P is called after the command 00.0 is executed. In the other hand G65 unconditionally calls a macro P.
3. When an M98 block contains another NC command (for example, 00.000.0 M98 P), the machine stops in the single block mode. In the other hand, G65 does not stop the machine.
4. With G65 or G66, the level of local variables changes. With M98, the level of local variables does not change.

- **Non-modal call (G65)**

When G65 is specified, the macro program specified at address P is called. Argument (data) can be passed to the custom macro program.

- **Format:** G65 P L <argument>
- **Explanation:**
  P — number of the program to be called
  L — repetition count (by default, 1 to 9999 can be specified)
  <Argument> — Data passed to the macro. Its value is assigned to the corresponding local variables.

```
O0001
G90 G0 X50 Y50
…
G65 P2010 A50 B20 L3
…
M30
```

- **Argument specification:** two types of argument specification are available.

Argument specification it uses letter other than O, L, G, I, N and P once each. In repeated specification, the last one prevails.

<table>
<thead>
<tr>
<th>Address</th>
<th>Variable number</th>
<th>Address</th>
<th>Variable number</th>
<th>Address</th>
<th>Variable number</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>#1</td>
<td>I</td>
<td>#4</td>
<td>T</td>
<td>#20</td>
</tr>
<tr>
<td>B</td>
<td>#2</td>
<td>J</td>
<td>#5</td>
<td>U</td>
<td>#21</td>
</tr>
<tr>
<td>C</td>
<td>#3</td>
<td>K</td>
<td>#6</td>
<td>V</td>
<td>#22</td>
</tr>
<tr>
<td>D</td>
<td>#7</td>
<td>M</td>
<td>#13</td>
<td>W</td>
<td>#23</td>
</tr>
<tr>
<td>E</td>
<td>#8</td>
<td>Q</td>
<td>#17</td>
<td>X</td>
<td>#24</td>
</tr>
<tr>
<td>F</td>
<td>#9</td>
<td>R</td>
<td>#16</td>
<td>Y</td>
<td>#25</td>
</tr>
<tr>
<td>H</td>
<td>#11</td>
<td>S</td>
<td>#19</td>
<td>Z</td>
<td>#26</td>
</tr>
</tbody>
</table>
Note: Addresses that need not to be specified can be omitted. Local variables corresponding to an omitted address are set to null.

Argument specification uses A, B, C and \( i, j, k \) (i is \( \leq 10 \)) and automatically decides the argument specification type according to the letters and the sequence. Uses A, B, C once each and uses \( i, j, k \) up to ten times.

### Argument specification II

<table>
<thead>
<tr>
<th>Address</th>
<th>Variable number</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>#1</td>
</tr>
<tr>
<td>B</td>
<td>#2</td>
</tr>
<tr>
<td>C</td>
<td>#3</td>
</tr>
<tr>
<td>( I_1 )</td>
<td>#4</td>
</tr>
<tr>
<td>( J_1 )</td>
<td>#5</td>
</tr>
<tr>
<td>( K_1 )</td>
<td>#6</td>
</tr>
<tr>
<td>( I_2 )</td>
<td>#7</td>
</tr>
<tr>
<td>( J_2 )</td>
<td>#8</td>
</tr>
<tr>
<td>( K_2 )</td>
<td>#9</td>
</tr>
<tr>
<td>( I_3 )</td>
<td>#10</td>
</tr>
<tr>
<td>( J_3 )</td>
<td>#11</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address</th>
<th>Variable number</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K_3 )</td>
<td>#12</td>
</tr>
<tr>
<td>( I_4 )</td>
<td>#13</td>
</tr>
<tr>
<td>( J_4 )</td>
<td>#14</td>
</tr>
<tr>
<td>( K_4 )</td>
<td>#15</td>
</tr>
<tr>
<td>( I_5 )</td>
<td>#16</td>
</tr>
<tr>
<td>( J_5 )</td>
<td>#17</td>
</tr>
<tr>
<td>( K_5 )</td>
<td>#18</td>
</tr>
<tr>
<td>( I_6 )</td>
<td>#19</td>
</tr>
<tr>
<td>( J_6 )</td>
<td>#20</td>
</tr>
<tr>
<td>( K_6 )</td>
<td>#21</td>
</tr>
<tr>
<td>( I_7 )</td>
<td>#22</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address</th>
<th>Variable number</th>
</tr>
</thead>
<tbody>
<tr>
<td>( J_7 )</td>
<td>#23</td>
</tr>
<tr>
<td>( K_7 )</td>
<td>#24</td>
</tr>
<tr>
<td>( I_8 )</td>
<td>#25</td>
</tr>
<tr>
<td>( J_8 )</td>
<td>#26</td>
</tr>
<tr>
<td>( K_8 )</td>
<td>#27</td>
</tr>
<tr>
<td>( I_9 )</td>
<td>#28</td>
</tr>
<tr>
<td>( J_9 )</td>
<td>#29</td>
</tr>
<tr>
<td>( K_9 )</td>
<td>#30</td>
</tr>
<tr>
<td>( I_{10} )</td>
<td>#31</td>
</tr>
<tr>
<td>( J_{10} )</td>
<td>#32</td>
</tr>
<tr>
<td>( K_{10} )</td>
<td>#33</td>
</tr>
</tbody>
</table>

Note 1: Subscripts of \( i, j, \) and \( k \) for indicating the order of argument specification are not written in the actual program.

Note 2: Arguments \( i, j, k \) do not need to be written in orders. They will be identified according to the present sequence. For example: \( \#6 \#0 \#10 \#1 \#2 \#3 \#1 \#1 \#6 \#11 \#12 \#30 \). The variables are passed as follows:

\( I1 \rightarrow \#1, \#2 \rightarrow I6 \rightarrow \#3, \#4 \rightarrow I11 \rightarrow \#5, \#6 \rightarrow I12 \rightarrow \#7, \#8 \rightarrow 30 \rightarrow \#11 \);

Format: \( \#65 \) must be specified before any argument.

Mixture of argument specifications I and II: The CNC internally identifies argument specification I and II. If a mixture of argument specification I and II is specified, the type of argument specification specified later takes precedence.

Example

\( \#65 \#9001 \#1 \#1 \#0 \#33 \#1 \#1 \#1 \)

- **Modal call (\#66)**
  
  Once \#66 is issued to specify a modal call, a macro is called after a block specifying movement along axes is executed. This continues until \#67 is issued to cancel a modal call.

  Note: The format, functions and argument specification of \#65 are identical with that of the \#65 (non-modal call). Refer to the introduction of \#65 for detailed description.

- **Modal call nesting:** Modal calls can be nested by specifying another \#66 code during
a modal call.

Explanation:
1. In the specified block, only argument is passed, and macro modal call will not be executed.
2. Macro modal call can only be executed in the block with G00, G01, G02, and G03.
3. A macro program can be called in a block which contains a code such as miscellaneous function that does not involve movement along an axis.
4. G65 and G66 should not be specified at the same time.
5. Multiple macro programs cannot be called in G65 block. vars with G65, G66 should be specified prior to arguments and P.

- Sample program
  ➢ G65 call (bolt hole circle)
  Create a macro program for machining holes on a circle. The radius is I, start angle is A, holes interval is B, holes number is H, the center of the circle is (X,Y), commands can be specified in either the absolute or incremental mode. To drill in the clockwise direction, specify a negative value for B.

Format:
G65 P9 1 0 0 Xx Yy Z z Rr Ii Aa Bb Hh;
- X: coordinate of center point (absolute or incremental #24)
- Y: coordinate of center point (absolute or incremental #25)
- Z: hole depth #26
- R: coordinates of an rapid approaching point #18
- F: cutting feedrate #9
- I: circle radius #4
- A: Drilling start angle #1
- B: Incremental angle (clockwise when negative value is specified) #2
- H: Number of holes #11

Macro call:
O0002
G90 G00 X0 Y0 Z100;
G65 P9 1 0 0 X100 Y50 R30 Z-50 F500 I100 A45 B30 H5;
M30;

Macro program (the called program):
#3 = #4003 ... Stores G codes of 03 group
IF [ #3 EQ 90 ] GOT O1 ; ... Branches to N1 in the G90 mode
#24 = #5001 + #24 ; ... Calculates the X coordinate of the center point
#25 = #5002 + #25 ; ... Calculates the Y coordinate of the center point
N1 WHILE [ #11 GT 0 ] DO 1 ; ... Until the number of remaining holes reaches 0
#5 = #24 + #4 * COS[ #1 ] ; ... Calculates the hole position on X axis
#6 = #25 + #4 * SIN[ #1 ] ; ... Calculates the hole position on X axis
G90 G81 X#5 Y#6 Z#26 R#18 F#9 ; ... Drilling after moving to the target position
#1 = #1 + #2 ; ... Updates the angles
#11 = #11 - 1 ; ... Decrements the number of holes
END1;
GSK9 80 MDa Milling CNC System User Manual

Volume I  Programming

G#3  G80 ; …̶̶̶̶̶̶̶̶̶ Returns the G codes to the original state.

Argument meanings:
- #3: store G codes of #3 group
- #5: coordinate of the next hole to drill
- #6: coordinate of the next hole to drill

G66 modal call
Shown as follows: machine # holes h1, h2, h3

Call format: G66 P9201 Aa Bb Cc; (the argument in this example is assumed)

Macro program:

G90 G17 G00 X0 Y0 Z0
G00 X150 Y20
G66 P9201 A-10 B-40 C2000
G00 X100 Y20

M09
G00 X0 Y23.5
G67
G00 X150 Y20

M30;

Called macro program:
    G81 G98 R#1 Z#2 F#3 (machining process)
    M99

5. Variables

An ordinary machining program specifies a G code and the travel distance directly with a numeric value, for example, G01 and X100.0. With a custom macro program, numerical value can be specified directly or using variables, for example, G#101 X#120. When variables are used, the variable value can be changed by programs or using operation on the MDI panel.
• **Representation and using methods of variables**

  Differ from argument data, variables are considered as the carrier of data, for example, \#1, \#0101… are variables, \#0100, \#0100 are arguments. Data of arguments \#0100, \#0100 should be transferred to variable \#1 and \#2. When using or programming macro programs, numerical value can be specified directly such as \#0100, \#0100 or using variables such as \#0100, \#0100. When variables are used, the variable value can be changed by programs or using operation on the panel.

  The address value of a macro body can be specified by variables. The variable value can be set in the main program or be assigned the calculated value when executing the macro body. Multiple variables can be identified by numbers.

  **Variable representation**

  A number sign # followed by a variable number is shown as follows:

  \#i (i = 1, 2, 3, 4 …)

  For example: \#5, \#109, \#1005

  **Omission of decimal point**

  When a variable value is defined in a program, the decimal point can be omitted. For example when defining \#1 = 123, the actual value of variable \#1 is 123.000.

  **Referencing variables**

  To reference the value of a variable in a program, specify a word address followed by the variable number. A program with an expression <address> \#i or \#i indicates that the variable value or negative value is used as address value.

  For example:
  
  \#5 \#1 = 5

  \#5 \#1 = -5

  When replace variable numbers with variables, \#9100 rather than \#9100 is used, the 9 followed by \# means the replacement. For example when \#100 = 105, \#105 = 105, \#9100 = 105, \#9100 = 105.

  \#9100 and \#105 are equal. i.e. \#9100 \#9100, \#105 \#105

  **Note:** Program number \#, sequence number \# and optional block skip number \# cannot be followed with variables. For example:\#1, \#1, \#1.
Variable display

In macro variable page, "null" indicates the variable is null, i.e., undefined. The mark "* * * * * * * * * *" indicates the variable value overflows of the range but the internal stored data may not overflow.

The value of common variables #1 #20 ~ #1 #99, #2 #500 ~ #2 #999 can be displayed on macro variable page, or be assigned directly by inputting data on the page.

The value of local variables #1 #2 #2 ~ #1 #33 and system variables do not have display screen. A value of local variable or system variable can be displayed by assigning the value to common variables.

Variable data range: Integral type: -2147483648 ~ 2147483647
Real number type: -1047 ~ -10-29, 0, or 10-29 ~ 1047.

Integration type: -4744.44474447447447447
Real number type: -4744.44474447447447447

Types of variables

Variables are classified into four types by variable number:

<table>
<thead>
<tr>
<th>Variable number</th>
<th>Type of variable</th>
<th>Function</th>
<th>Range</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>#0</td>
<td>Null variable</td>
<td>This variable is always null. No value can be assigned to this variable.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#1 #20 ~ #1 #33</td>
<td>Local variable</td>
<td>Local variable can only be used within a macro to hold data such as the results of operations. When the power is turned off, local variables are initialized to null. When a macro is called, arguments are assigned to local variables.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>#2 #500 ~ #2 #999</td>
<td>Common variable</td>
<td>Common variables can be shared among different macro programs. When the power is turned off, variables</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. In macro variable page, "null" indicates the variable is null, i.e., undefined. The mark "* * * * * * * * * *" indicates the variable value overflows of the range but the internal stored data may not overflow.
2. The value of common variables #1 #20 ~ #1 #99, #2 #500 ~ #2 #999 can be displayed on macro variable page, or be assigned directly by inputting data on the page.
3. The value of local variables #1 #2 #2 ~ #1 #33 and system variables do not have display screen. A value of local variable or system variable can be displayed by assigning the value to common variables.
4. Variable data range: Integral type: -2147483648 ~ 2147483647, Real number type: -1047 ~ -10-29, 0, or 10-29 ~ 1047.
<table>
<thead>
<tr>
<th>#5000 ~ #9999</th>
<th>When the power is turned off, data is stored</th>
<th>display</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1000 ~ #1015</td>
<td>G54, G55 input</td>
<td>read only</td>
</tr>
<tr>
<td>#1032</td>
<td>Store G54, G55, read all 16 bits of a signal at one time</td>
<td>read/write</td>
</tr>
<tr>
<td>#1033</td>
<td>Store G54, G55, write all 16 bits of a signal at one time</td>
<td>read/write</td>
</tr>
<tr>
<td>#1100 ~ #1115</td>
<td>G54, G55 output</td>
<td>read only</td>
</tr>
<tr>
<td>#1132</td>
<td>Store G54, G55, write all 16 bits of a signal at one time</td>
<td>read/write</td>
</tr>
<tr>
<td>#1133</td>
<td>Store G54, G55, write all 16 bits of a signal at one time</td>
<td>read/write</td>
</tr>
<tr>
<td>#2001 ~ #2032</td>
<td>Tool length compensation</td>
<td>read/write</td>
</tr>
<tr>
<td>#2201 ~ #2232</td>
<td>Tool length compensation</td>
<td>read/write</td>
</tr>
<tr>
<td>#2401 ~ #2432</td>
<td>Cutter compensation</td>
<td>read/write</td>
</tr>
<tr>
<td>#2601 ~ #2632</td>
<td>Cutter compensation</td>
<td>read/write</td>
</tr>
<tr>
<td>#3003</td>
<td>Automatic operation</td>
<td>read/write</td>
</tr>
<tr>
<td>#3004</td>
<td>Automatic operation</td>
<td>read/write</td>
</tr>
<tr>
<td>#3901</td>
<td>The number of machined parts</td>
<td>read/write</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 1</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 2</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 3</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 4</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 5</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 6</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 7</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 8</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 9</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 10</td>
<td>read only</td>
</tr>
<tr>
<td>#4001</td>
<td>Modal G code group 11</td>
<td>read only</td>
</tr>
<tr>
<td>#</td>
<td>Description</td>
<td>Ranges</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>#4</td>
<td>D code</td>
<td>0-99990</td>
</tr>
<tr>
<td>#4</td>
<td>E code</td>
<td>0-99990</td>
</tr>
<tr>
<td>#4</td>
<td>H code</td>
<td>0-99990</td>
</tr>
<tr>
<td>#4</td>
<td>M code—#41 13-#41 15</td>
<td>0-99990</td>
</tr>
<tr>
<td>#4</td>
<td>Program number—#41 15</td>
<td>0-99999</td>
</tr>
<tr>
<td>#4</td>
<td>S code—#41 19-#41 20</td>
<td>0-99999</td>
</tr>
<tr>
<td>#4</td>
<td>Sequence number—#41 14-#41 16</td>
<td>0-99999</td>
</tr>
<tr>
<td>#5</td>
<td>Block end point, work piece coordinate system, tool compensation value not included</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Current position, machine coordinate system, tool compensation value included</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Current position, work piece coordinate system, tool compensation value included</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Skip signal position, work piece coordinate system, tool compensation value included</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Tool length compensation value, current execution value.</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>External work piece zero point offset value</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Work piece zero point offset value</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Work piece zero point offset value</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Work piece zero point offset value</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Work piece zero point offset value</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Work piece zero point offset value</td>
<td>-9999-9999.9999</td>
</tr>
<tr>
<td>#5</td>
<td>Work piece zero point offset value</td>
<td>-9999-9999.9999</td>
</tr>
</tbody>
</table>
5.2.1 Null Variables

When the variable value is undefined, the variable is null. Variable #0 is always null, and can be read only.

a, referencing
- The address itself is ignored when an undefined variable (null variable) is quoted.

\[ \text{hen} \ #0 \ \Rightarrow \ \text{null} \]

b, Arithmetic operation
- A full equals to in any case except when assigned by "null".

\[
\begin{array}{|c|c|}
\hline
\text{hen} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{assignment} & \text{null} \\
\hline
\text{the arithmetic operation result} & \text{null} \\
\hline
\text{equals to null} & \text{null} \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\text{#} & \text{#} \\
\hline
\text{null} & \text{null} \\
\hline
\end{array}
\]

5.2.2 Local Variables

Local variables are the variables internally defined in a program. They are effective only within the program, i.e., it is only used within the program.

- A local variable # that calls macro programs at a certain moment is different from the # at another moment. No matter the macro programs are identical or not. Therefore, when macro program # is called from macro program #, like nesting, the local variables used in macro # will not be misused in macro #, and will not disable the value in macro #.
Usually, the local variables are used to accept the value passed from argument. Please refer to "Argument Specification" for the relationship between arguments and addresses. Pay attention that, the initial state of local variable is Null, before the local variable is defined (assigned).

- **Custom macro program nesting and local variable**

  When calling a macro program, its nesting level increases by one, and correspondingly, the level of local variable increases by one as well.

  The relationship between macro program call and local variable is shown as follows:

<table>
<thead>
<tr>
<th>Main Program</th>
<th>macro program 1st level</th>
<th>macro program 2nd level</th>
<th>macro program 3rd level</th>
<th>macro program 4th level</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1~#33 local variables (0 level) are provided in the main program.</td>
<td>O--- G65P- M99</td>
<td>O--- G65P- M99</td>
<td>O--- G65P- M99</td>
<td>O--- M99</td>
</tr>
</tbody>
</table>

  1. #1~#33 local variables (0 level) are provided in the main program.
  2. When a macro program (1 level) is called by G65, the local variable (0 level) is stored, and local variables #1~#33 of the new macro program is prepared. The argument replacement is possible.
  3. Each time a macro program (2, 3, 4 levels) are called, local variables (1, 2, 3 levels) in each group are stored, and new local variables (2, 3, 4 levels) are prepared.
  4. When M99 (return from macro programs) is commanded, the local variables (0, 1, 2, 3 levels) stored in (2, 3) are recovered in the state as they are stored.

**5.2.3 Common Variable**

Common variable is the global variable defined within the system. It can be used in any program. That is to say, #1~#99 used in a macro program is the same as the one used in another macro program. Therefore, the arithmetic operation result of common variable #1~#99 in a program can be used in another program.

In the system, there is no special regulation for using common variables. #100~#199 is the variable group without power-off memory function; #500~#999 is the variable group with power-off memory function, i.e. data are stored after power-off.
5.2. System Variables

System variables are used to read and write CNC internal data, such as tool length compensation value, tool nose radius compensation value. Some system variables can only be read. System variables are the basis of automatic control and general-purpose machining program development.

- **Interface Signal** The macro variable corresponding to interface signal is the exchange signal between CNC and custom macro program.

<table>
<thead>
<tr>
<th>Variable No.</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1000~#101 5</td>
<td>1-bit signal can be sent from the CNC to a custom macro. Used to read signal bit by bit.</td>
</tr>
<tr>
<td>#1032</td>
<td>1-bit signal can be sent from the CNC to a custom macro. Used to read all 16 bits of a signal at one time.</td>
</tr>
<tr>
<td>#1100~#111 5</td>
<td>1-bit signal can be sent from the CNC to a custom macro. Used to read and write signal bit by bit.</td>
</tr>
<tr>
<td>#1132</td>
<td>1-bit signal can be sent from the CNC to a custom macro. Used to read and write all 16 bits of a signal at one time.</td>
</tr>
<tr>
<td>#1133</td>
<td>32-bit signal can be sent from the CNC to a custom macro. Used to read all 32 bits of a signal at one time.</td>
</tr>
</tbody>
</table>

Note: Please refer to the GSK980TD PLC User Manual for the relations between variables and interface signals.

- **Tool Compensation Value** Tool compensation value can be read and written

<table>
<thead>
<tr>
<th>Compensation No.</th>
<th>Tool Length Compensation</th>
<th>Cutter Compensation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>eome</td>
<td>ear</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>#2201</td>
<td>#2001</td>
</tr>
<tr>
<td>02</td>
<td>#2202</td>
<td>#2002</td>
</tr>
<tr>
<td>03</td>
<td>#2203</td>
<td>#2003</td>
</tr>
<tr>
<td>31</td>
<td>#2231</td>
<td>#2031</td>
</tr>
<tr>
<td>32</td>
<td>#2232</td>
<td>#2032</td>
</tr>
</tbody>
</table>
### Automatic Operation Control

The control state of automatic operation can be changed.

<table>
<thead>
<tr>
<th>Variable No.</th>
<th>Variable Value</th>
<th>Single Block Completion of an Auxiliary Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Enabled</td>
<td>To be awaited</td>
</tr>
<tr>
<td>1</td>
<td>Disabled</td>
<td>To be awaited</td>
</tr>
<tr>
<td>2</td>
<td>Enabled</td>
<td>Not to be awaited</td>
</tr>
<tr>
<td>3</td>
<td>Disabled</td>
<td>Not to be awaited</td>
</tr>
</tbody>
</table>

Note 1: When the power is turned on, the value of this variable is 0.

Note 2: When single block stop is enabled (G46.1 is 1), the state of #3003 can change the execution of single block stop.

Note 3: When single block stop is disabled (G46.1 is 0), single block stop operation is not performed even if the single block switch is set to ON.

Note 4: When a wait for completion of auxiliary function (M, S and T functions) is not specified, program execution proceeds to the next block before completion of auxiliary functions. Also, distribution completion signal DEN is not output.

<table>
<thead>
<tr>
<th>Variable No.</th>
<th>Variable Value</th>
<th>Feed Hold</th>
<th>Feed Rate Override</th>
<th>Exact Stop</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Enabled</td>
<td>Enabled</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
<tr>
<td>1</td>
<td>Disabled</td>
<td>Enabled</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
<tr>
<td>2</td>
<td>Enabled</td>
<td>Disabled</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
<tr>
<td>3</td>
<td>Disabled</td>
<td>Disabled</td>
<td>Enabled</td>
<td>Enabled</td>
</tr>
<tr>
<td>4</td>
<td>Enabled</td>
<td>Enabled</td>
<td>Disabled</td>
<td>Disabled</td>
</tr>
<tr>
<td>5</td>
<td>Disabled</td>
<td>Enabled</td>
<td>Disabled</td>
<td>Disabled</td>
</tr>
<tr>
<td>6</td>
<td>Enabled</td>
<td>Disabled</td>
<td>Disabled</td>
<td>Disabled</td>
</tr>
<tr>
<td>7</td>
<td>Disabled</td>
<td>Disabled</td>
<td>Disabled</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

Note 1: When the power is turned on, the value of this variable is 0.

Note 2: When feed hold is disabled, if the feed hold button is held down, the machine stops in the single block stop mode. However, single block stop operation is not performed when the single block mode is disabled with variable #3003.

Note 3: When the feed hold is disabled, if the feed hold button is pressed then released, the machine does not stop; program execution continues and the machine stops at the first block where feed hold is enabled; the feed hold lamp is ON.

Note 4: When feed rate override is disabled, an override of 100% is always applied regardless of the setting of the feed rate override.

Note 5: When exact stop check is disabled, no exact stop check is
Number of machined parts

The number of machined parts can be read and written.

<table>
<thead>
<tr>
<th>Variable No.</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>#3901</td>
<td>Number of machined parts</td>
</tr>
</tbody>
</table>

Modal information

Modal information specified in blocks up to the immediately preceding block can be read.

<table>
<thead>
<tr>
<th>Variable No.</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>#4001</td>
<td>Group 1 (G00, G01, G02, G03, G04, G05, G06, G07, G08, G09, G10, G11, G12, G13, G14, G15, G16, G17, G18, G19)</td>
</tr>
<tr>
<td>#4002</td>
<td>Group 2 (G17, G18, G19)</td>
</tr>
<tr>
<td>#4003</td>
<td>Group 3 (G90, G91)</td>
</tr>
<tr>
<td>#4005</td>
<td>Group 5 (G94, G95)</td>
</tr>
<tr>
<td>#4006</td>
<td>Group 6 (G20, G21)</td>
</tr>
<tr>
<td>#4007</td>
<td>Group 7 (G40, G41, G42)</td>
</tr>
<tr>
<td>#4008</td>
<td>Group 8 (G43, G44, G49)</td>
</tr>
<tr>
<td>#4010</td>
<td>Group 10 (G98, G99)</td>
</tr>
<tr>
<td>#4014</td>
<td>Group 14 (G54, G55, G56, G57, G58, G59)</td>
</tr>
<tr>
<td>#4107</td>
<td>D code</td>
</tr>
<tr>
<td>#4109</td>
<td>F code</td>
</tr>
<tr>
<td>#4111</td>
<td>H code</td>
</tr>
<tr>
<td>#4113</td>
<td>M code</td>
</tr>
<tr>
<td>#4114</td>
<td>Block sequence number</td>
</tr>
<tr>
<td>#4115</td>
<td>Program name</td>
</tr>
<tr>
<td>#4119</td>
<td>S code</td>
</tr>
<tr>
<td>#4120</td>
<td>T code</td>
</tr>
</tbody>
</table>

Current position

Position information can be read.

<table>
<thead>
<tr>
<th>Variable No.</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>#5001~#5005</td>
<td>Workpiece coordinate system block end point (tool compensation value not included)</td>
</tr>
<tr>
<td>#5021~#5025</td>
<td>Machine coordinate system current position (tool compensation value)</td>
</tr>
</tbody>
</table>
### Volume I   Programming

#### #5041~#5045 Work piece coordinate system current position (tool compensation value included)
- Disabled

#### #5041~#5045 Work piece coordinate system skip signal position (tool compensation value included)
- Enabled

#### #5061~#5065 Tool length compensation value
- Disabled

**Note 1:** The first digit (from 1 to 5) represents an axis number.

**Note 2:** The tool length compensation value currently used for execution rather than the immediately preceding tool compensation value is held in variables #5061~#5065.

---

#### Work piece coordinate system compensation value

Work piece coordinate system compensation value can be read and written.

<table>
<thead>
<tr>
<th>Variable No.</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>#5201~#5205</td>
<td>The first to the fifth axes external work piece zero point offset value</td>
</tr>
<tr>
<td>#5221~#5225</td>
<td>The first to the fifth axes G54 work piece zero point offset value</td>
</tr>
<tr>
<td>#5241~#5245</td>
<td>The first to the fifth axes G55 work piece zero point offset value</td>
</tr>
<tr>
<td>#5261~#5265</td>
<td>The first to the fifth axes G56 work piece zero point offset value</td>
</tr>
<tr>
<td>#5281~#5285</td>
<td>The first to the fifth axes G57 work piece zero point offset value</td>
</tr>
<tr>
<td>#5301~#5305</td>
<td>The first to the fifth axes G58 work piece zero point offset value</td>
</tr>
<tr>
<td>#5321~#5325</td>
<td>The first to the fifth axes G59 work piece zero point offset value</td>
</tr>
</tbody>
</table>

---

#### 5.3 Arithmetic and Logic Operation

- Macro programs in both traditional G65 H format and statement format are compatible with GSK980MDa.
- Users can alternatively select one of them for programming. This makes programming more convenient and flexible.

- Please strictly observe the formats and specifications in the following arithmetic and logic operation table.

### Arithmetic and Logic Operation

<table>
<thead>
<tr>
<th>Operation</th>
<th>Traditional G65H format</th>
<th>Statement format</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Definition, assignment</td>
<td>#i = #j</td>
<td>G65 H1 P#i Q#j</td>
<td></td>
</tr>
<tr>
<td>Addition</td>
<td>#i = #j + #k</td>
<td>G65 H2 P#i Q#j R#k</td>
<td>Logic operation is performed on binary</td>
</tr>
<tr>
<td>Subtraction</td>
<td>#i = #j - #k</td>
<td>G65 H3 P#i Q#j R#k</td>
<td></td>
</tr>
</tbody>
</table>
### Multiplication

\[
#i = #j \times #k
\]

### Division

\[
#i = #j \div #k
\]

### Square root

\[
#i = \sqrt{#j}
\]

### Absolute value

\[
#i = |#j|
\]

### Rounding off

- **Rounding up**
  \[
  #i = \text{UP}(#j)
  \]

- **Rounding down**
  \[
  #i = \text{DOWN}(#j)
  \]

### Nature logarithm

\[
#i = \ln(#j)
\]

### Exponential function

\[
#i = e^{#j}
\]

### Sine

\[
#i = \sin(#j)
\]

### Arcsine

\[
#i = \arcsin(#j)
\]

### Cosine

\[
#i = \cos(#j)
\]

### Arccosine

\[
#i = \arccos(#j)
\]

### Tangent

\[
#i = \tan(#j)
\]

### Arctangent

\[
#i = \arctan(#j)
\]

### Unconditional branch

\[
\text{GOTO} #i
\]

### Equals to branch

\[
\text{IF } (#i = #j) \text{ GOTO} #k
\]

### Not equals to branch

\[
\text{IF } (#i \neq #j) \text{ GOTO} #k
\]

### Greater than branch

\[
\text{IF } (#i > #j) \text{ GOTO} #k
\]

### Smaller than branch

\[
\text{IF } (#i < #j) \text{ GOTO} #k
\]

### Greater than or equals to

\[
\text{IF } (#i \geq #j) \text{ GOTO} #k
\]

### Smaller than or equals to

\[
\text{IF } (#i \leq #j) \text{ GOTO} #k
\]

### Conversion from BCD to BIN

\[
#i = \text{BIN}(#j)
\]

### Conversion from BIN to BCD

\[
#i = \text{BCD}(#j)
\]

### User alarm

\[
\text{User alarm} = \text{None}
\]

### 5.3.1 Traditional Format

If traditional G65 H format is used for programming, only limited operations and jump command can be specified by it. The currently used H operation needs at most 3 operands, so the corresponding operation can be completed when the needed variables (or constants) are obtained in a block.

- **General format**
  \[
  \text{G65 Hm P#i Q#j R#k ;}
  \]

  - \(m\): 01~99 means operation command or jump command function
  - \(#i\): the name of variable that stored the operation result
  - \(#j\): operand 1; it can be constant.
#k: operand 2; it can be constant.

## Meaning:

\[ #i = #j \]

Operational sign, designated by Hm

<table>
<thead>
<tr>
<th>Example</th>
<th>Code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>G65</td>
<td>Hm</td>
<td>P#100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>#101</td>
</tr>
<tr>
<td></td>
<td></td>
<td>#102</td>
</tr>
<tr>
<td></td>
<td></td>
<td>#100 \ o #10\</td>
</tr>
<tr>
<td>G65</td>
<td>Hm</td>
<td>#100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P#101</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15 \ o #10\</td>
</tr>
<tr>
<td></td>
<td></td>
<td>#100 \ o #10\</td>
</tr>
</tbody>
</table>

**Note 1:** G65 Hm should be commanded prior to operation or jump command.

**Note 2:** When code is commanded in G65 block, G65 P means macro program call. H means argument. No operation or jump command is performed.

**Note 3:** At most 4 decimal numbers of the constant decimal part can be obtained for rounding. 3 digit numbers can be displayed in the window.

### Code function explanation

1. **Variable value assignment, \#I = \#J**

   G65  H0 1  P\#I  Q\#J;

   (example) G65  H01  P#101  Q#102  (#101 = 102)

2. **Addition operation \#I = \#J + \#K**

   G65  H0 2  P\#I  Q\#J  R\#K;

   (example) G65  H02  P#101  Q#102  R#103  (#101 = 102 + 103)

3. **Subtraction operation \#I = \#J - \#K**

   G65  H0 3  P\#I  Q\#J  R\#K;

   (example) G65  H03  P#101  Q#102  R#103  (#101 = 102 - 103)

4. **Multiplication operation \#I = \#J \times \#K**

   G65  H0 4  P\#I  Q\#J  R\#K;

   (example) G65  H04  P#101  Q#102  R#103  (#101 = 102 \times 103)

5. **Division operation \#I = \#J \div \#K**

   G65  H0 5  P\#I  Q\#J  R\#K;

   (example) G65  H05  P#101  Q#102  R#103  (#101 = 102 \div 103)

**Note:** The divisor \#K cannot be 0; otherwise an alarm occurs.

6. **OR operation \#I = \#J \ OR \#K**

   G65  H1 1  P\#I  Q\#J  R\#K;

   (example) G65  H11  P#101  Q#102  R#103  (#101 = #102 \ OR #103)

7. **AND operation \#I = \#J \ AND \#K**

   G65  H1 2  P\#I  Q\#J  R\#K;

   (example) G65  H12  P#101  Q#102  R#103  (#101 = #102 \ AND #103)
(8) XOR operation \( \#I \oplus \#J \oplus \#K \)

\[
G65 \ H13 \ P\#I \ Q\#J \ R\#K
\]

(example) \( G65 \ H13 \ P\#101 \ Q\#102 \ R\#103; \) \((#101 = \#102 \oplus \#103)\)

(9) Square root \( \sqrt{\#I} \)

\[
G65 \ H1 \ P\#I \ Q\#J
\]

(example) \( G65 \ H1 \ P\#101 \ Q\#102; \) \((#101 = \sqrt{#102})\)

*Note: the radicand \#I cannot be negative; otherwise, an alarm occurs.*

(10) Absolute value \( |\#I| \)

\[
G65 \ H2 \ P\#I \ Q\#J
\]

(example) \( G65 \ H2 \ P\#101 \ Q\#102; \) \((#101 = |#102|); \ #101\#102\)

(11) Rounding off \( \#I = \#I\#.# I\#\#\# \) (round off the first decimal)

\[
G65 \ H3 \ P\#I \ Q\#J
\]

(example) \( G65 \ H3 \ P\#101 \ Q\#101; \) \((#101 = 1.359; \ #101\#1)\)

(12) Rounding up \( \#I = \#I\#.# I\#\#\# \)

\[
G65 \ H4 \ P\#I \ Q\#J
\]

(13) Rounding down \( \#I = \#I\#.# I\#\#\# \)

\[
G65 \ H5 \ P\#I \ Q\#J
\]

With CNC, when the absolute value of the integer produced by an operation on a number is greater than the absolute value of the original number, such an operation is referred to as rounding up to an integer. Conversely, when the absolute value of the integer produced by an operation on a number is less than the absolute value of the original number, such an operation is referred to as rounding down to an integer. Be particularly careful when handling negative numbers.

(examle) suppose that \#1= 1.2, \#2 = -1.2

- when \#3= \text{FUP}[\#1] is executed, 2.0 is assigned to \#3
- when \#3= \text{FIX}[\#1] is executed, 1.0 is assigned to \#3
- when \#3= \text{FUP}[\#2] is executed, -2.0 is assigned to \#3
- when \#3= \text{FIX}[\#2] is executed, -1.0 is assigned to \#3

(14) Natural logarithm \( \#I = \text{LN}[\#J] \)

\[
G65 \ H6 \ P\#I \ Q\#J
\]

(example) \( G65 \ H6 \ P\#101 \ Q\#102; \) \((#101 = \text{LN}[#102])\)

*Note: when the antilogarithm \#J is zero or smaller, otherwise, an alarm is issued.*

(15) Exponential function \( \#I = \text{EXP}[\#J] \)

\[
G65 \ H7 \ P\#I \ Q\#J
\]

(example) \( G65 \ H7 \ P\#101 \ Q\#102; \) \((#101 = \text{EXP}[#102])\)
(16) Sine \#/i = \#i sin (unit: deg)
G65 H31 P\#i Q\#j
(example) G65 H31 P#101 Q#103; (#101=#103)

(17) Arcsine #J = \#J arcsin
G65 H32 P\#i Q\#j
(example) G65 H32 P#101 Q#103; (#101=#103)

Note 1: When the NAT bit of parameter No.015 is set to 0, the output range is 270° ~ 90°
When the NAT bit of parameter No.015 is set to 1, the output range is -90° ~ 90°

Note 2: The arcsine operand \#J cannot exceed the range -1~1, otherwise an alarm is issued.

(18) Arccosine #J = \#J arccos (deg)
G65 H33 P\#i Q\#j
(example) G65 H33 P#101 Q#103; (#101=#103)

Note 1: The arccosine operand \#J cannot exceed the range -1~1, otherwise an alarm is issued.

(19) Arctangent #J = \#J arctan (deg)
G65 H35 P\#i Q\#j
(example) G65 H35 P#101 Q#103; (#101=#103)

Note: #J cannot equal to \(K+\pi/2\) (\(K=0, \pm1, \pm2, \pm3 \ldots\)), otherwise the result is wrong.

(20) Tangent #J = \#J tan (deg)
G65 H34 P\#i Q\#j
(example) G65 H34 P#101 Q#103; (#101=#103)

(21) Arctangent #J = ATAN [\#J] / [\#K] (unit: deg)
G65 H36 P\#i Q\#j R\#k
(example) G65 H36 P#101 Q#103 R3; (#101=ATAN[#103]/[3])

Note 1: When the NAT bit of parameter No.015 is set to 0, the output range is 0° ~ 360°
When the NAT bit of parameter No.015 is set to 1, the output range is -180° ~ 180°

(22) Conversion from BCD to BIN \#J = BINC[\#J]
G65 H41 P\#i Q\#j
(example) G65 H41 P#101 Q#102; (#101=BIN[#102])

(23) Conversion from BIN to BCD \#J = BCDC[\#J]
G65 H42 P\#i Q\#j
(example) G65 H42 P#101 Q#102; (#101=BCD[#102])

(24) Unconditional branch
G65 H80 Pn;
Pn: sequence number
(example) G65 H80 P120; (Go to N120 block)

(25) Equal to conditional branch
### 5.3.2 Macro Statement

The operations listed in the "Arithmetic and Logic Operation" table can be executed in program. The expressions right to the operator contain constants and (or) variables that consisting of functions and operators. The variables #i and #k in the expression can be assigned as constants. The left variable (the first variable) can be assigned by expression. The macro statement is more intuitive, convenient and flexible. It can perform compound operation and multinesting. Sometimes, a macro statement is equal to several traditional G65H macro programs.

- **General format**

  Please refer the statement format in the "Arithmetic and Logic Operation" table for editing macro statement.
Macro program editing

In program editing mode or MID mode, by pressing macro editing state can be switched or inserted.

<table>
<thead>
<tr>
<th>Differences of two states</th>
<th>Automatic space</th>
<th>Processing of letters</th>
<th>Input of special signs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert state</td>
<td>spaces are added to identify words</td>
<td>Press to switch</td>
<td>special signs cannot be input</td>
</tr>
<tr>
<td>Macro editing state</td>
<td>space are not added</td>
<td>Input as a letter</td>
<td>special signs can be input</td>
</tr>
</tbody>
</table>

Explanations

1. Angular unit

The angular units of function SIN, COS, ASIN, ACOS, TAN and ATAN are degree. For example, 90° 30′ means 90.5 degrees.

2. ARCSIN

   i. the solution ranges are as indicated below
   when the NAT bit of parameter No.015 is set to 0: 270°~ 90°
   when the NAT bit of parameter No.015 is set to 1: -90°~ 90°
   ii. when the #j is beyond the range of -1 to 1, P/S alarm is issued.
   iii. a constant can be used instead of the #j variable.

3. ARCCOS

   i. the solution ranges from 180°~ 0°
   ii. when the #j is beyond the range of -1 to 1, P/S alarm is issued.
   iii. a constant can be used instead of the #j variable.

4. ARCTAN

   Specify the lengths of two sides and separate them by a slash /.
   The solution ranges are as follows:
   When the NAT bit of parameter No.015 is set to 0: 0°~ 360°
   [Example] when #1=ATAN[-1]/[-1] is specified, #1=225°
   When the NAT bit of parameter No.015 is set to 1: -180°~ 180°
   [Example] when #1=ATAN[-1]/[-1] is specified, #1=-135°
5. Natural logarithm \( \#i = \ln(\#j) \)
   i. Note that the relative error may be greater than \( 10^{-8} \).
   ii. When the antilogarithm \( \#j \) is zero or smaller, P/S alarm is issued.
   iii. A constant can be used instead of the \( \#j \) variable.

6. Exponential function \( \#i = \exp(\#j) \)
   i. Note that the relative error may be greater than \( 10^{-8} \).
   ii. When the result of the operation exceeds \( 3.65 \times 10^{47} \) \( \#j \) is about 110, an overflow occurs and P/S alarm is issued.
   iii. A constant can be used instead of the \( \#j \) variable.

7. ROUND function
   When the ROUND function is included in an arithmetic or logic operation command, IF statement, or WHILE statement, the ROUND function rounds off at the first decimal place.

   **Example:**
   When \#1 = \text{ROUND}(\#2) \) is executed where \#2 = 1.2345 the value of variable \#1 is 1.0.
   When the ROUND function is used in NC statement address, the ROUND function rounds off the specified value according to the least input increment of the address.

8. Rounding up and down to an integer
   With CNC, when the absolute value of the integer produced by an operation on a number is greater than the absolute value of the original number, such an operation is referred to as rounding up to an integer. Conversely, when the absolute value of the integer produced by an operation on a number is less than the absolute value of the original number, such an operation is referred to as rounding down to an integer. Be particularly careful when handling negative numbers.

   **Example:**
   Suppose that \#1 = 1.2, \#2 = -1.2
   When \#3 = \text{UP}(\#1) \) is executed, 2.0 is assigned to \#3.
   When \#3 = \text{FIX}(\#1) \) is executed, 1.0 is assigned to \#3.
   When \#3 = \text{UP}(\#2) \) is executed, -2.0 is assigned to \#3.
   When \#3 = \text{FIX}(\#2) \) is executed, -1.0 is assigned to \#3.
3. Operation such as addition and subtraction (⁺, -, OR, XOR)

<table>
<thead>
<tr>
<th>Example</th>
<th>#1 = #2 + #3 * SIN[#4];</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>▶</td>
</tr>
<tr>
<td></td>
<td>▼</td>
</tr>
<tr>
<td>▶ , ▼ and ▼ indicate the order of operations.</td>
<td></td>
</tr>
</tbody>
</table>

Brackets are used to change the order of operations. Brackets can be used to multinesting.

Note that the square bracket [ ] is used to enclose an expression; the round bracket ( ) is used in comments. When the priority is not defined, it is advised to use square bracket to enclose.

5.4 Branch and Repetition

In a program, the flow of control can be changed using the GOTO statement and IF statement. Three types of branch and repetition operations are used:

1. GOTO statement (unconditional branch)
2. IF statement (conditional branch: IF … THEN…)
3. WHILE statement (repetition WHILE…)

5.4.1 Unconditional Branch (GOTO statement)

Go to the block with sequence number n. When a sequence number out the range of 1~99999 is specified, an alarm is raised. A sequence number can also be specified using an expression.

Format:

\[ \text{GOTO } n \]

Where:

- \( n \): sequence number
  - \( 1~99999 \)

Example:

\[ \text{GOTO 1} \]
\[ \text{GOTO #101} \]

5.4.2 Conditional Branch (IF statement)

Specify a conditional expression after IF.

Format:

\[ \text{IF [conditional expression] GOTO } n \]

If the specified conditional expression is satisfied, a branch to sequence number n occurs. If the specified condition is not satisfied, the next block is executed.

If the value of variable \#1 is greater than 10, a branch to sequence number N2 occurs.

\[ \text{IF [#1 GT 10] GOTO 2} ; \]

If the condition is not satisfied

- Processing

- N2 G00 G91 X10.0 ;

If the condition is satisfied

- :
If the specified conditional expression is satisfied, a predetermined macro statement is executed. Only a single macro statement is executed.

```
#1 EQ #2] T-EN #3=0;
```

If the value of #1 and #2 are the same, 0 is assigned to #3; if not, no execution will be performed.

### Example

```
IF [3< > 2] GOTO 2; it means if 3 is not equal to 2, branch to N2 block
```

```
IF [#101> =7.22] THEN #101=SIN30; it means, if #101 is greater than 7.22, the expression after THEN is executed, i.e., assign Sin 30° to #101.
```

### Sample Program

The sample program below finds the sum of number 1 to 10.

```
O9500
#101=0            Initial value of the variable to hold the sum
#102=1             Initial value of the variable as an addend
N1 IF [#102 GT 10]GOTO 2 … Branch to N2 when the addend is greater than 10
#101= #101+ #102  … calculation to find the sum
#102= #102+ 1  … Next addend
GOTO 1  … Branch to N1
N2 M30  … End of program; Sum of number 1 to 10
```

In 980Da, operators in the following table are used to compare two values to determine whether they are equal or one value is smaller or greater than the other value.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>EQ or  = =</td>
<td>Equal to ( =)</td>
</tr>
<tr>
<td>NE or  &lt; &gt;</td>
<td>Not equal to ( ≠)</td>
</tr>
<tr>
<td>GT or &gt;</td>
<td>Greater than ( &gt;)</td>
</tr>
<tr>
<td>GE or &gt; =</td>
<td>Greater than or equal to ( ≥)</td>
</tr>
<tr>
<td>LT or &lt;</td>
<td>Less than ( &lt;)</td>
</tr>
<tr>
<td>LE or &lt; =</td>
<td>Less than or equal to ( ≤)</td>
</tr>
</tbody>
</table>

Example:

```
IF [3< > 2] GOTO 2; it means if 3 is not equal to 2, branch to N2 block
```

```
IF [#101> =7.22] THEN #101=SIN30; it means, if #101 is greater than 7.22, the expression after T-EN is executed, i.e., assign Sin 30° to #101.
```

The sample program below finds the sum of number 1 to 10.

```
O9500
#101=0            Initial value of the variable to hold the sum
#102=1             Initial value of the variable as an addend
N1 IF [#102 GT 10]GOTO 2 … Branch to N2 when the addend is greater than 10
#101= #101+ #102  … calculation to find the sum
#102= #102+ 1  … Next addend
GOTO 1  … Branch to N1
N2 M30  … End of program; Sum of number 1 to 10
```
Specify a conditional expression after `WHILE`. While the specified condition is satisfied, the program from `DO` to `END` is executed. If not, program execution proceeds to the block after `END`.

**Example**

```
WHILE [Conditional expression] DO
  [Program]
END m;
```

- If the condition is not fulfilled, program execution proceeds to the block after `END`.
- If the condition is fulfilled, the program from `DO` to `END` after `WHILE` is executed.

**Explanations:**
- While the specified condition is fulfilled, the program from `DO` to `END` after `WHILE` is executed. If the specified condition is not fulfilled, program execution proceeds to the block after `END`. The same format as the `IF` statement applies. A number after `DO` and a number after `END` are identification numbers for specifying the range of execution. The number 1, 2, and 3 can be used. When a number other than 1, 2, and 3 is used, P/S alarm occurs.

**Nesting:**
- The identification number 1 to 3 in a `DO, END` loop can be used as many times as desired. Note, however, when a program includes crossing repetition loops (overlapped `DO` ranges), P/S alarm occurs.

1. The identification numbers (1 to 3) can be used as many times as required.
   - `WHILE [...] DO 1;
     Processing
     END 1;
   - `WHILE [...] DO 2;
     Processing
     END 2;
   - `WHILE [...] DO 3;
     Processing
     END 3;

2. DO ranges cannot overlap.
   - `WHILE [...] DO 1;
     Processing
     WHILE [...] DO 2;
     Processing
     END 1;
     END 2;

3. DO loops can be nested to a maximum depth of three levels.
   - `WHILE [...] DO 1;
     WHILE [...] DO 2;
     WHILE [...] DO 3;
     Processing
     END 3;
     END 2;
     END 1;

4. Control can be transferred to the outside of a loop.
   - `WHILE [...] DO 1;
     IF [...] GOTO n;
     END 1;
     Nn

5. Branches cannot be made to a location within a loop.
   - `IF [...] GOTO n;
     WHILE [...] DO 1;
     Nn ...;
     END 1;

...
5.5 Macro Statement and NC statement

The following blocks are referred to as macro statements:

- Blocks containing arithmetic or logic operation (such as =).
- Blocks containing a controlling statement (such as GOTO, DO, END...).
- Blocks containing a macro call command (such as G65, G66).
- Blocks other than macro statements are referred to as NC statement.

Custom macro programs are similar to subprograms. They can be edited, registered and used in the same way as subprograms. M98 can call a custom macro program, but cannot pass arguments.

Usually, the macro program is provided by tool builders, but it can also be programmed by customers. It is not necessary for the customers to remember all related commands in macro programs besides codes that call macro programs.

5.5.1 Macro Programming and Registering

Custom macro programs are similar to subprograms. They can be edited, registered and used in the same way as subprograms. M98 can call a custom macro program, but cannot pass arguments.

Usually, the macro program is provided by tool builders, but it can also be programmed by customers. It is not necessary for the customers to remember all related commands in macro programs besides codes that call macro programs.

5.5.2 Limitation

- Macro statement processing in cutter compensation C mode

In cutter compensation C mode (G41, G42), in order to calculate the transmission point, NC prereads the next block. The processing way is not the same as general NC statement.

When a macro statement is executed as a single block, it is the block that does not involve movement. And, in some cases, it cannot correctly execute compensation (strictly speaking, such block involves 0 distance of movement).

- Jump (GOTO, DO, END)

In cutter compensation C mode, when jump command (GOTO, DO, EN) is specified, P/S alarm occurs.

- When the move command adopts variables

In cutter compensation C, when the move command (such as G01, X#101) adopts variables, P/S alarm occurs. Because cutter compensation C mode is block preread mode, the end point of the next block is essential for calculating the current transmission point position. Specifying X#101 (an unknown data) does not enable a correct calculation of the current transmission point.

- Single block operation (MDI)

In MDI mode, macro programs can be specified, but macro program call cannot be executed.

- \%/ appearing in the middle of an expression enclosed in brackets [ ] on the right-hand side of an arithmetic expression is regarded as a division operator; it is not regarded as the specified for an optional block skip code.

- Reset

A reset operation clears any called states of custom macro programs and subprograms, and cursor returns to the first block of the main program.
6.1 Application for Cutter Radius Compensation

Generally, the parts machining process is programmed according to parts drawing in one point on a tool. As for the tool used actually, because of the processing or other requirement, the tool is not an ideal point, but an arc only. The position offset exists between actual cutting point and ideal point when the cutting feed is performed. It may cause over cut or undercut, so the part accuracy will be affected. So, the cutter radius compensation can be used to improve the part accuracy in machining.

The path of part figure can be shifted by a cutter radius, which this method is called type tool compensation—this is a simply method but the movement path of next block can be processed only after a block is performed, so the phenomenon as over cutting will be generated at the intersection point of two blocks.

In order to settle the above issues and eliminate the error, the Tool compensation C should be setup. When a block is read in, the tool compensation C is not performed immediately but the next block is read in again. Corresponding movement path is calculated according to the point of intersection of two blocks conjunction vector. The tool compensation C performs more accurate compensation in figure because two blocks are read for processing in advance. See the Fig. 6-1

![Fig. 6-1 C type cutter radius compensation](image)

The radius value of each tool should be set before tool compensation C is applied. Tool radius compensation value is set in the OFFSET page (table 6-1), this page contains tool geometric radius and tool radius wear. There into, is the tool compensation value, when the bit 1 of bit parameter No.003 is 1, the is compensation value input by diameter. If the bit 1 of bit parameter No.003 is 0, the is compensation value input by radius. The following explanations are all indicated in radius compensation value if not especially pointed out.

Table 6-1 Display page for CNC cutter radius compensation value
### 6.1.3 Command format

<table>
<thead>
<tr>
<th>No</th>
<th>Command</th>
<th>Explanation</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>G17</td>
<td>Offset plane selection command (XY plane)</td>
<td>See the Fig.6-2</td>
</tr>
<tr>
<td>002</td>
<td>G18</td>
<td>Offset plane selection command (XZ plane)</td>
<td>See the Fig.6-2</td>
</tr>
<tr>
<td>003</td>
<td>G19</td>
<td>Offset plane selection command (YZ plane)</td>
<td>See the Fig.6-2</td>
</tr>
<tr>
<td>004</td>
<td>G40</td>
<td>Cutter radius compensation cancellation</td>
<td></td>
</tr>
<tr>
<td>005</td>
<td>G41</td>
<td>Cutter radius compensation left along advancing direction</td>
<td></td>
</tr>
<tr>
<td>006</td>
<td>G42</td>
<td>Cutter radius compensation right along advancing direction</td>
<td></td>
</tr>
</tbody>
</table>

### 6.1.4 Compensation direction

Tool compensation direction is determined according to the relative position of tool with work piece, when the cutter radius compensation is applied. See the Fig.6-2.
In initial status, CNC is in cutter radius compensation cancellation mode. CNC sets cutter radius compensation offset mode when the G41 or G42 command is executed. At the beginning of the compensation, the CNC reads two blocks in advance, the next block is stored in the cutter radius compensation buffer memory when a block is performed. When in Single mode, two blocks are read, after the end point of the 1st block is performed, it is stopped. Two blocks are read in advance in successive performance. So, there are a block being performed and two blocks behind it in CNC.

Neither setup nor cancellation of the Tool compensation C can be performed in the MDI mode.

The cutter radius compensation value can not be a negative, normally, the wearing value is negative (negative value indicates for wearing).

Instead of G02 or G03, the setting or cancellation of cutter radius compensation can be commanded only by using G00 or G01, or the alarm occurs.

CNC will cancel Tool compensation C mode when you press RESET key.

Corresponding offset should be specified while the G40, G41, or G42 is specified in the block, or the alarm occurs.

When cutter radius compensation is employed in main program and subprogram, the CNC should cancel compensation mode before calling or exiting sub-program (namely, before M98 or M99 is performed) or the alarm occurs.

Cancel the compensation mode temporarily when G54-99, G28-31 and canned cycle command are executed. Restore the cutter radius compensation mode when the above commands are finished.

The parts are machined in the coordinate system in Fig. 6-3. The tool compensation number D07 is employed, tool geometric radius is 2mm and the tool radius wearing is 0.
Perform tool setting in the mode of offset cancellation, after finishing the tool setting, and set the tool radius in the OFFSET page.

Table 4.2

<table>
<thead>
<tr>
<th>NO.</th>
<th>Geometric(H)</th>
<th>Wearing(H)</th>
<th>Geometric(D)</th>
<th>Wearing(D)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>07</td>
<td></td>
<td></td>
<td>2.000</td>
<td>0.000</td>
</tr>
<tr>
<td>08</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Programs:

N0 G92 X0 Y0 Z0; Tool are positioned at start position X0, Y0 and Z0 when the absolute coordinate system is specified.

N1 G90 G17 G00 G41 D07 X250.0 Y550.0; Start-up cutter, the tool is shifted to the tool path by the distance specified in D07, geometric radius of D07 is set to 2.0mm, tool wearing 0, then the tool radius is 2mm.

N2 G01 Y900.0 F150; Specifies machining from P1 to P2

N3 G03 X500.0 Y1150.0 R650.0; Specifies machining from P3 to P4

N4 G02 X900.0 R-250.0; Specifies machining from P4 to P5

N5 G03 X950.0 Y900.0 R650.0; Specifies machining from P5 to P6

N6 G01 X1150.0; Specifies machining from P6 to P7

N7 G00 G40 X0 Y0; Cancels the offset mode, the tool is returned to the start position X0, Y0

6.2 Offset Path Explanation for Cutter Radius Compensation

"Inner side" and "outer side" will be employed in the following explanations. When an angle of intersection created by tool paths specified by move commands for two blocks is over or equal to 180°, it is referred to as "inner side". When the angle is between 0° and 180°, it is referred to as "outer side".
There are 3 steps should be performed for cutter radius compensation: establishment, performing and cancellation.

The tool movement performed from offset cancellation mode to G41 or G42 command establishment is called tool compensation establishment (also called start-up).

(Note: For S, L and C labeled in the following figures, if not especially described, they should be regarded as the following meaning:

- S - - - - Single block stop point;
- L - - - - Linear;
- C - - - Circular arc.

(a) Tool movement along an inner side of a corner

1) Linear to linear
2) Linear to circular)
6.2.3 Tool movement in offset mode

The mode after setting the cutter radius compensation and before canceling the cutter radius compensation is called offset mode.

- **Offset path of invariable compensation direction in compensation mode**
  - Linear to linear
  - Linear to circular
3) Circular to linear

4) Circular to circular

5) Inner side machining less than 1 degree and compensation vector amplification

(a) Linear to linear

(b) Move along the outer of obtuse angle corner (α > 90°)

1) Linear to linear

2) Linear to circular
Chapter 6  Cutter Compensation

Volume I   Programming

1) Linear to linear

2) Linear to circular

3) Circular to linear

4) Circular to circular

(c) Move along the outer of acute angle corner ($\alpha > 90°$)

1) Linear to linear

2) Linear to circular

Fig. 6-9a Linear to linear
(obtuse angle, outside movement)

Fig. 6-9b Linear to circular
(obtuse, outside movement)

Fig. 6-9c Circular to linear
(obtuse angle, outside movement)

Fig. 6-9d Circular to circular
(obtuse angle, outside movement)

Fig. 6-10a Linear to linear
(acute, movement outside)

Fig. 6-10b Linear to circular
(acute, movement outside)
When it is exceptional

1. There is no intersection

When the compensation value is large

Alarm occurs and tool stops

When the tool radius value is small, there is an intersection for the arc compensation, when the radius is bigger, the intersection may not exist, the tool stops at the end of previous block, and then the alarm occurs.

Fig. 6-11 Exceptional --- There is no intersection after the path offset

2) The arc center is consistent to the start point or end point

Stop
Tool nose center path

Programmed path

The alarm will be generated by the following issues (G41)

N5 G91 G01 Z20;
N6 G02 Z10 K0;
N7 G03 X-10 I-10;

Fig. 6-12 Center of arc is consistent to the start point or end point

• Offset path with compensation direction changed in compensation mode

The compensation direction can be changed in special occasion but it cannot be changed at the beginning and the following block. There are no inner side and outer side for the full compensation.
1) Linear to linear

2) Linear to circular

When there is no intersection if the compensation is normally performed when changing the offset direction from block A to block B using G41 and G42, if the intersection of the offset path is not required, create the vector vertical to block B at the start point of block B.

i) Linear to linear

ii) Linear to circular
6.2. Tool operation in offset cancellation mode

When the G40 command is employed in block in compensation mode, the CNC enters the compensation cancellation mode. This is called compensation cancellation.

The circular arc command G02 and G03 can not be employed when the cutter radius compensation is cancelled. If they are commanded, alarm is generated and the operation is stopped.

It controls and performs this block and the blocks in the cutter radius compensation buffer memory in the compensation cancellation mode. If the single block switch is turned on, it stops after executing a block. The next block is executed instead of reading it when the start key is pressed again.

(a) Tool movement along an inner side of a corner (α≥90°)

1) Linear to linear  
2) Circular to linear
**Chapter 6  Cutter Compensation**

**Volume I  Programming**

### Tool movement along the outside of a corner at an obtuse angle $\left(180^\circ > \alpha > 90^\circ\right)$

1. Linear to linear

2. Circular to linear

### Tool movement along the outside of a corner at an acute angle $\left(90^\circ > \alpha > 0^\circ\right)$

1. Linear to linear

2. Circular to linear

### Tool movement along the corner outside at an acute angle less than 1 degree linear to linear $\left(\alpha < 1^\circ\right)$
6.2.5 Interference check

Tool over cutting is called “interference”. The interference check function can check tool over cutting in advance. This interference check is performed even if the over cutting does not occur. However, all interference can not be checked by this function.

(1) Conditions for the interference
1) The direction of the tool path is different from that of the programmed path. (90 degrees to 270 degrees between these paths)
2) In addition to the condition above, the angle between the start point and end point of the tool center path is quite different from that between the start point and end point of the programmed path in circular machining (more than 180 degrees).

Example: Linear machining
1) The groove depth less than the compensation value

There is no interference actually, but program direction in block B is opposite to the cutter radius compensation path. The cutter stops, and the alarm occurs.

2) The groove depth less than compensation value

There is no interference actually, but program direction in block A is opposite to the cutter radius compensation path. The cutter stops, and the alarm occurs.
6.2.6 Command of compensation vector cancel temporarily

If the following commands G92, G28, G29, coordinate command selection G54~G59 and canned cycle are specified in compensation mode, the compensation vector is temporarily cancelled and then automatically restored after these commands are executed. Now, the temporary compensation vector cancellation is different to the compensation cancellation mode, tool is moved to the specified point by compensation vector cancellation from the intersection. And the tool moves to the intersection directly when the compensation mode restores.

- **Coordinate system setting command G92 and coordinate system selection command G54~G59**

![Diagram](image)

Note: SS is indicated as the point stopped for twice in Single block mode.

- **Automatic return to the reference point G28**

If G28 is specified in compensation mode, the compensation will be cancelled at an intermediate position. The compensation mode is automatically restored after the reference point is returned.

![Diagram](image)
• **Canned cycle**
  If the canned cycle command is specified in compensation mode, the compensation will be temporarily cancelled in the canned cycle operation. The compensation mode is automatically restored after the canned cycle is terminated.

6.2. **Exceptional case**

• **When the inner corner machining is less than tool radius**
  When the inner corner machining is less than tool radius, the inner offset of a tool will cause over cut. The tool stops and alarm occurs after moving at the beginning or at the corner in previous block. But if the switch of “Single block” is ON, the tool will be stopped at the end of the previous block.

• **When a groove less than the tool diameter is machined**
  When the tool center moves opposite to the direction of programmed path, the over cutting will be generated by the cutter radius compensation. Tool stops and alarm appears after moving at the beginning of previous block or at the corner.

• **When a step less than the tool radius is machined**
  When a program contains a step which is an arc and less than tool radius, tool center path may form a opposite movement direction to the programmed path. So the first vector is ignored and it moves to the end of the second vector along a straight line. The program will be stopped for Single block mode, the cycle continues if it is not single block mode. The compensation will be executed correctly and no alarm will be generated if the step is a straight line. (But the uncut part is reserved.)

• **When the sub-program is contained in G code**
  CNC should be in compensation cancellation mode before calling the sub-program (namely, before the G98 is performed). Offset can be applied after entering the sub-program, but the compensation cancellation should be applied before returning to the main program (before M99), or the alarm occurs.

• **When compensation value is changed**
  (a) Usually, the compensation value is changed when the tool change is performed in compensation cancellation mode. If the compensation value is changed in compensation mode, the
new one is ineffective which is effective till the program is executed again.

(b) If different compensation values are commanded in different blocks of a program, different compensation value will be compensated to the corresponding block. But if it is an arc, the alarm will be generated. For details, refer to the following explanation.

(c) about “arc data error in C type cutter radius compensation”.

- \textbf{When the end point for the programming arc is not on the arc}

When the end point for the programming arc is not on the arc, the tool stops and the alarm information shows “end point is not on the arc”.

Two same points in the starting is shown an example:

\begin{verbatim}
N0 G90 G00 X-50 Y-50
N1 G91 G1 G41 X0 Y0 D1 F800 … without moving
N2 G90 X0 Y0
N3 X50
\end{verbatim}

The above-mentioned program may occur the “two same points” when starting, and the compensation may not perform. The transit point \texttt{N1} between \texttt{N0} and \texttt{N1} and the transit point \texttt{N2} between \texttt{N1} and \texttt{N2} are shared a same point.

\begin{verbatim}
N0 G90 G00 X-50 Y-50
N1 G1 G41 X0 Y0 D1 F800
N2 G91 X0 Y0 D1 800 without moving
N3 X50
\end{verbatim}

The “last two same points” may occur when starting at the last program, in the case of the compensation has been performed. The section without moving which is regarded as the movement
approximates to the zero, so it is necessary to maintain the compensation amount. The transit point between N1 and N2 is P1, and the transit point between N2 and N3 is P2, P1 and P2 are shared a same point.

In the same way, in the compensation mode, if the “two same points” may occur, the compensation value will be maintained. In the retraction mode, the similar start mode is divided into “the previous two same points” and “the last two same points”

● The alarm and corresponding explanation of Circular arc data error in cutter compensation C

(a) The example of this alarm may occur in a circle

Program example: N0 G90 G00 X-50 Y-50 Z50
N1 G01 G2 X0 Y0 D1 F800
N2 G02 I50
N3 G91 G01 X-50 Y-50

The transit point between straight line N1 and circular arc N2 is P1, the transit point between circular N2 and straight line N3 is P2, and the compensation radius is r, in this case, the circular after tool compensation is more than 360°.

After a block (N9 G91 G0 X0 Y0) (without moving) is inserted between N1 and N2 in the above mentioned program, the “circular data error in cutter compensation C” may alarm.

Because the point after N9 inserted which is equal to the one of N1, namely, they are regarded as “two same points”. The transit point P1 is performed treating the “two same points”, the position of P1 is obviously differ from the above one which does not insert the N9 block. So the cut circular arc path by this transit point is absolutely differing from the path to be machined, so the alarm is then generated: “circular arc data error in cutter compensation C”

(b) The example for a non-circle may occur:
Program example:
N0 G90 G00 X0 Y0 Z0
N1 G01 G41 X0 Y0 D1 F800
N2 G02 X5 Y0 R25

The P1 and P2 are the transit point of tool compensation as the left figure shown, wherein the “r” is compensation radius. This is a normal treatment mode for the straight line to circular arc.

The alarm may occur in terms of the following program:
N0 G90 G00 X0 Y0 Z0
N1 G01 G41 X0 Y0 D1 F800 without moving, originally start
N2 G02 X5 Y0

Because the N1 block does not a movement, namely, it equals to the “two same points”. The transit points P1 and P2 are performed based on the treatment of two same points (The path of two same points), so the circular arc path cut by this transit point obviously differs from the actual path to be machined, in this case, the “circular arc data error in cutter compensation C” may alarm.

(c) In the calculation of arc cutter compensation C, this alarm may issue if the compensation radius D is modified.

Program example:
N0 G90 G00 X0 Y0 Z25
N1 G01 G41 X0 Y0 D1 F800
N2 G02 X5 Y0 R25
N3 G02 X100 Y0 R25

The left figure is shown the programmed path and the tool center path.

If the compensation radius D is changed in N3, for example, the D2 is specified in N3 block (the value of D2 is not equal to the one of D1), in this case, it is similar as (b), an alarm of the “circular arc data error in cutter compensation C” may occur.
VOLUME II  OPERATION
CHAPTER 1  OPERATION MODE AND DISPLAY

This GSK980MDa system employs an aluminum alloy solid operator panel, which exterior is as follows.

1.1 Panel Division

This GSK980MDa adopts an integrated panel, which division is as follows:
## 1.1.1 State indication

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Machine Zero Return Indicator" /></td>
<td>Machine zero return finish indicator</td>
</tr>
<tr>
<td><img src="image" alt="Rapid Indicator" /></td>
<td>Rapid indicator</td>
</tr>
<tr>
<td><img src="image" alt="Single Block Indicator" /></td>
<td>Single block indicator</td>
</tr>
<tr>
<td><img src="image" alt="Block Skip Indicator" /></td>
<td>Block Skip indicator</td>
</tr>
<tr>
<td><img src="image" alt="Machine Lock Indicator" /></td>
<td>Machine Lock indicator</td>
</tr>
<tr>
<td><img src="image" alt="MST Lock Indicator" /></td>
<td>MST Lock indicator</td>
</tr>
<tr>
<td><img src="image" alt="Dry Run Indicator" /></td>
<td>Dry Run indicator</td>
</tr>
</tbody>
</table>

## 1.1.2 Edit keypad

<table>
<thead>
<tr>
<th>Key</th>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="RESET Key" /></td>
<td>RESET key</td>
<td>For CNC reset, feed, output stop etc.</td>
</tr>
<tr>
<td><img src="image" alt="Address Key" /></td>
<td>Address key</td>
<td>Address input</td>
</tr>
<tr>
<td><img src="image" alt="Double Address Key" /></td>
<td></td>
<td>Double address key, switching between two sides by pressing repeatedly</td>
</tr>
<tr>
<td><img src="image" alt="Sign Key" /></td>
<td>Sign key</td>
<td>Double address key, switching between two characters by pressing repeatedly</td>
</tr>
<tr>
<td>Key</td>
<td>Name</td>
<td>Function</td>
</tr>
<tr>
<td>----------</td>
<td>--------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>7 8 9</td>
<td>Numerical key</td>
<td>For digit input</td>
</tr>
<tr>
<td>4 5 6</td>
<td>Decimal point</td>
<td>For decimal point input</td>
</tr>
<tr>
<td>1 2 3</td>
<td>Input key</td>
<td>For confirmation of parameters, offset values input</td>
</tr>
<tr>
<td></td>
<td>Output key</td>
<td>For start communication output</td>
</tr>
<tr>
<td></td>
<td>Change key</td>
<td>For switching of message, display</td>
</tr>
<tr>
<td></td>
<td>Edit key</td>
<td>For insertion, alteration, deletion of programs, words in editing</td>
</tr>
<tr>
<td></td>
<td>EOB key</td>
<td>For block end sign input</td>
</tr>
<tr>
<td></td>
<td>Cursor moving keys</td>
<td>For cursor moving control</td>
</tr>
<tr>
<td></td>
<td>Page key</td>
<td>Page switching in a same interface</td>
</tr>
</tbody>
</table>

### 1.1.3 Menu display

<table>
<thead>
<tr>
<th>Menu key</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To enter position interface. There are RELATIVE POS, ABSOLUTE POS, INTEGRATED POS, POS&amp;PRG pages in this interface.</td>
</tr>
</tbody>
</table>
To enter program interface. There are PRG CONTENT, PRG STATE, PRG LIST, PRG PREVIEW, 4 pages in this interface.

To enter TOOL OFFSET interface. There are TOOL OFFSET, MARRO variables and Tool Life Management (modifying Bit0 of state parameter №002). OFFSET interface displays offset values; MARRO for CNC macro variables.

To enter alarm interface. There are CNC, PLC ALARM and ALARM Log pages in this interface.

To enter Setting interface. There are SWITCH, PASSWORD SETTING, DATE & TIME, SETTING (G54~G59), GRAGH SET and TRACK pages in this interface.

To enter BIT PARAMETER, DATA PARAMETER, PITCH COMP interfaces (switching between each interface by pressing repeatedly).

To enter DIAGNOSIS interface. There are CNC DIAGNOSIS, PLC STATE, PLC VALUE, VERSION MESSAGE interfaces (switching between each interfaces by pressing the key repeatedly). CNC DIAGNOSIS, PLC STATE, PLC VALUE interfaces display CNC internal signal state, PLC addresses, data state message; the VERSION MESSAGE interface displays CNC software, hardware and PLC version No.

1.1.4 Machine panel

The keys function in GSK980MDa machine panel is defined by PLC program (ladder), see their function significance in the machine builder’s manual.

The functions of the machine panel keys defined by standard PLC program are as follows:

<table>
<thead>
<tr>
<th>Key</th>
<th>Name</th>
<th>Function explanation</th>
<th>Function mode</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Feed Hold" /></td>
<td>Feed Hold key</td>
<td>Dwell commanded by program, MDI</td>
<td>Auto mode, DNC, MDI mode</td>
</tr>
<tr>
<td><img src="image" alt="Cycle Start" /></td>
<td>Cycle Start key</td>
<td>Cycle start commanded by program, MDI</td>
<td>Auto mode, DNC, MDI mode</td>
</tr>
<tr>
<td><img src="image" alt="Feedrate Override" /></td>
<td>Feedrate Override keys</td>
<td>For adjustment of the feedrate</td>
<td>Auto mode, DNC, MDI mode, Edit mode, Machine zero mode, MPG mode, Single Step mode, MANUAL mode</td>
</tr>
<tr>
<td>Key</td>
<td>Name</td>
<td>Function explanation</td>
<td>Function mode</td>
</tr>
<tr>
<td>-----</td>
<td>------</td>
<td>----------------------</td>
<td>---------------</td>
</tr>
<tr>
<td><img src="image1.png" alt="Rapid override keys" /></td>
<td>Rapid override keys</td>
<td>For adjustment of rapid traverse</td>
<td>Auto mode, DNC, MDI mode, Machine zero mode, MANUAL mode</td>
</tr>
<tr>
<td><img src="image2.png" alt="Spindle override keys" /></td>
<td>Spindle override keys</td>
<td>For spindle speed adjustment (spindle analog control valid)</td>
<td>Auto mode, DNC, MDI mode, edit mode, Machine zero mode, MPG mode, Step mode, MANUAL mode</td>
</tr>
<tr>
<td><img src="image3.png" alt="JOG key" /></td>
<td>JOG key</td>
<td>For spindle Jog ON/OFF</td>
<td>Machine zero mode, MPG mode, Single Step mode, MANUAL mode</td>
</tr>
<tr>
<td><img src="image4.png" alt="Lubricating key" /></td>
<td>Lubricating key</td>
<td>For machine lubrication ON/OFF</td>
<td>Machine zero mode, MPG mode, Single Step mode, MANUAL mode</td>
</tr>
<tr>
<td><img src="image5.png" alt="Cooling key" /></td>
<td>Cooling key</td>
<td>For coolant ON/OFF</td>
<td>Auto mode, MDI mode, Edit mode, Machine zero mode, MPG mode Step mode, MANUAL mode</td>
</tr>
<tr>
<td><img src="image6.png" alt="Spindle control keys" /></td>
<td>Spindle control keys</td>
<td>Spindle CCW, Spindle stop, Spindle CW</td>
<td>Machine zero mode, MPG mode, Single Step mode, MANUAL mode</td>
</tr>
<tr>
<td><img src="image7.png" alt="Rapid traverse key" /></td>
<td>Rapid traverse key</td>
<td>For rapid traverse /feedrate switching</td>
<td>Auto mode, DNC, MDI mode, Machine zero mode, MANUAL mode</td>
</tr>
<tr>
<td><img src="image8.png" alt="Manual feed key" /></td>
<td>Manual feed key</td>
<td>For positive/negative moving of X, Y, Z axis in Manual, Step mode</td>
<td>Machine zero mode, Step mode, MANUAL mode</td>
</tr>
<tr>
<td>Key</td>
<td>Name</td>
<td>Function explanation</td>
<td>Function mode</td>
</tr>
<tr>
<td>-----</td>
<td>------</td>
<td>----------------------</td>
<td>---------------</td>
</tr>
<tr>
<td><img src="image" alt="Handwheel axis selection key" /></td>
<td>Handwheel axis selection key</td>
<td>For X, Y, Z axis selection in MPG mode</td>
<td>MPG mode</td>
</tr>
<tr>
<td><img src="image" alt="MPG/Step increment and Rapid override selection key" /></td>
<td>MPG/Step increment and Rapid override selection key</td>
<td>Move amount per handwheel scale 0.001/0.01/0.1 mm Move amount per step 0.001/0.01/0.1 mm</td>
<td>Auto mode, MDI mode, Machine zero mode, MPG mode, Step mode, MANUAL mode,</td>
</tr>
<tr>
<td><img src="image" alt="Single Block key" /></td>
<td>Single Block key</td>
<td>For switching of block/blocks execution, Single block lamp lights up if Single mode is valid</td>
<td>Auto mode, DNC, MDI mode</td>
</tr>
<tr>
<td><img src="image" alt="Block Skip key" /></td>
<td>Block Skip key</td>
<td>For skipping of block headed with “/” sign, if its switch is set for ON, the Block Skip indicator lights up</td>
<td>Auto mode, DNC, MDI mode</td>
</tr>
<tr>
<td><img src="image" alt="Machine Lock key" /></td>
<td>Machine Lock key</td>
<td>If the machine is locked, its lamp lights up, and X, Z axis output is invalid.</td>
<td>Auto mode, DNC, MDI mode, Edit mode, Machine zero mode, MPG mode, Step mode, MANUAL mode,</td>
</tr>
<tr>
<td><img src="image" alt="M.S.T. Lock key" /></td>
<td>M.S.T. Lock key</td>
<td>If the miscellaneous function is locked, its lamp lights up and M, S, T function output is invalid.</td>
<td>Auto mode, DNC, MDI mode</td>
</tr>
<tr>
<td><img src="image" alt="Dry Run key" /></td>
<td>Dry Run key</td>
<td>If dry run is valid, the Dry run lamp lights up. Dry run for program/MDI blocks command</td>
<td>Auto mode, DNC, MDI mode</td>
</tr>
</tbody>
</table>
### 1.2 Summary of Operation Mode

There are 7 modes that include Edit, Auto, DNC, MDI, Machine zero, Step/MPG, Manual, modes in this GSK980MDa.

- **Edit mode**
  
  In this mode, the operation of part program setting-up, deletion and modification can be performed.

- **Auto mode**
  
  In this mode, the program is executed automatically.

- **MDI mode**
In this mode, the operation of parameter input, command blocks input and execution can be performed.

- **Machine zero mode**
  In this mode, the operation of X, Y, Z, 4\textsuperscript{th}, 5\textsuperscript{th} axis machine zero return can be performed separately.

- **MPG / Step mode**
  In the Step/MPG feed mode, the moving is performed by an increment selected by CNC system.

- **Manual mode**
  In this mode, the operation of Manual feed, Manual Rapid, feedrate override adjustment, Rapid override adjustment and spindle ON/OFF, cooling ON/OFF, Lubrication ON/OFF, spindle jog, manual tool change can be performed.

- **DNC mode**
  In this mode, the program is run by DNC mode.

### 1.3 Display Interface

There are 7 interfaces for GSK980MDa such as Position, Program etc., and there are multiple pages in each interface. Each interface (page) is separated from the operation mode. See the following figures for the display menu, display interface and page layers:

<table>
<thead>
<tr>
<th>Menu key</th>
<th>Display interface</th>
<th>Display page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>POSITION</strong></td>
<td>Position interface</td>
<td><img src="image" alt="Display Interface Diagram" /></td>
</tr>
<tr>
<td><strong>PROGRAM</strong></td>
<td>Pro. content</td>
<td>PRG CONTENT</td>
</tr>
<tr>
<td></td>
<td>Pro. state</td>
<td>PRG STATE</td>
</tr>
<tr>
<td></td>
<td>Pro.previe w</td>
<td>PRG PREVIEW</td>
</tr>
<tr>
<td></td>
<td>Program list</td>
<td>PRG LIST</td>
</tr>
<tr>
<td>Menu key</td>
<td>Display interface</td>
<td>Display page</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------</td>
<td>--------------</td>
</tr>
<tr>
<td></td>
<td>TOOL OFFSET interface</td>
<td><img src="https://via.placeholder.com/150" alt="Tool Offset Diagram" /></td>
</tr>
<tr>
<td></td>
<td>MACRO interface</td>
<td><img src="https://via.placeholder.com/150" alt="Macro Diagram" /></td>
</tr>
<tr>
<td></td>
<td>Tool life interface</td>
<td><img src="https://via.placeholder.com/150" alt="Tool Life Diagram" /></td>
</tr>
<tr>
<td></td>
<td>CNC alarm</td>
<td><img src="https://via.placeholder.com/150" alt="CNC Alarm" /></td>
</tr>
<tr>
<td></td>
<td>PLC alarm/warn</td>
<td><img src="https://via.placeholder.com/150" alt="PLC Alarm/Warn" /></td>
</tr>
<tr>
<td></td>
<td>Alarm log</td>
<td><img src="https://via.placeholder.com/150" alt="Alarm Log" /></td>
</tr>
<tr>
<td></td>
<td>Setting interface</td>
<td><img src="https://via.placeholder.com/150" alt="Setting Diagram" /></td>
</tr>
<tr>
<td></td>
<td>G54 setting</td>
<td><img src="https://via.placeholder.com/150" alt="G54 Setting" /></td>
</tr>
<tr>
<td>Menu key</td>
<td>Display interface</td>
<td>Display page</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td>Graph interface</td>
<td><img src="" alt="Graph Interface Diagram" /></td>
</tr>
<tr>
<td></td>
<td>Bit parameter</td>
<td><img src="" alt="Bit Parameter Diagram" /></td>
</tr>
<tr>
<td></td>
<td>Data parameter</td>
<td><img src="" alt="Data Parameter Diagram" /></td>
</tr>
<tr>
<td></td>
<td>Pitch parameter</td>
<td><img src="" alt="Pitch Parameter Diagram" /></td>
</tr>
<tr>
<td></td>
<td>CNC diagnosis</td>
<td><img src="" alt="CNC Diagnosis Diagram" /></td>
</tr>
<tr>
<td></td>
<td>PLC state</td>
<td><img src="" alt="PLC State Diagram" /></td>
</tr>
<tr>
<td></td>
<td>PLC data</td>
<td><img src="" alt="PLC Data Diagram" /></td>
</tr>
<tr>
<td>Version message</td>
<td>Version Message</td>
<td><img src="" alt="Version Message" /></td>
</tr>
</tbody>
</table>
1.3.1 Position interface

Press to enter Position interface, which has four interfaces such as ABSOLUTE POS, RELATIVE POS, INTEGRATED POS and POS&PRG, and they can be viewed by or key.

1) ABSOLUTE POS display interface

The X, Y, Z coordinates displayed are the absolute position of the tool in current workpiece coordinate system, as CNC power on, these coordinates are held on and the workpiece coordinate system is specified by G92.

<table>
<thead>
<tr>
<th>ABSOLUTE POS</th>
<th>G00 G17 G90 G54</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.000</td>
</tr>
<tr>
<td>Y</td>
<td>0.000</td>
</tr>
<tr>
<td>Z</td>
<td>0.000</td>
</tr>
<tr>
<td>MDI</td>
<td>00000 N00000</td>
</tr>
<tr>
<td>F0100 S 00 M30</td>
<td></td>
</tr>
<tr>
<td>PRG. F:</td>
<td>100</td>
</tr>
<tr>
<td>ACT. F:</td>
<td>0</td>
</tr>
<tr>
<td>FED OVRI:</td>
<td>150%</td>
</tr>
<tr>
<td>RAP OVRI:</td>
<td>100%</td>
</tr>
<tr>
<td>SPI OVRI:</td>
<td>100%</td>
</tr>
<tr>
<td>PART CNT:</td>
<td>0</td>
</tr>
<tr>
<td>CUT TIME:</td>
<td>0:00:00</td>
</tr>
<tr>
<td>S0000 T00 H00</td>
<td></td>
</tr>
</tbody>
</table>

PRG. F: a rate specified by F code in program

Note: It displays “PRG. F” in Auto, MDI mode; “MAN. F” in Machine zero, Manual mode; “HNDL INC” in MPG mode; “STEP INC” in Step mode.

ACT. F: Actual speed after feedrate override calculated.

FED OVRI: An override that is selected by feedrate override switch.

SPI OVRI: Adjust the spindle rotational speed by altering spindle override.

PART CNT: Part number plus 1 when M30 (or M99 in the main program) is executed

CUT TIME: Time counting starts if Auto run starts, time units are hour, minute and second

The parts counting and the cut time are memorized at power-down and the clearing ways for them are as follows:

PART CNT clearing: press key then press key.

CUT TIME clearing: press key then press key.

S0000: Feedback spindle speed of spindle encoder, and spindle encoder must be fixed to display actual spindle speed.

T01: Current tool No. and tool offset No.
2) RELATIVE POS display page

The X, Y, Z axis coordinates displayed are the current position relative to the relative reference point, and they are held on at CNC power on. They can be cleared at any time. If X, Y, Z axis relative coordinates are cleared, the current position will be the relative reference point. When CNC parameter No.005 Bit1=1, as the absolute coordinates are set by G92 code, X, Y, Z axis relative coordinates are identical with the set absolute coordinates.

<table>
<thead>
<tr>
<th>RELATIVE POS</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.000</td>
</tr>
<tr>
<td>Y</td>
<td>0.000</td>
</tr>
<tr>
<td>Z</td>
<td>0.000</td>
</tr>
</tbody>
</table>

In RELATIVE POS page, press and hold cross key till the “X” in the page blinks, press key to clear X coordinate;

In RELATIVE POS page, press and hold y key till the “Y” in the page blinks, press key to clear Y coordinate;

In RELATIVE POS page, press and hold z key till the “Z” in the page blinks, press key to clear Z coordinate;

The clearing steps of X, Y, Z axis relative coordinates:

The method for X, Y, Z axis relative coordinates divided by 2:

In RELATIVE POS page, press and hold key till the “X” in the page blinks, press key, X coordinate will be divided by 2;

In RELATIVE POS page, press and hold key till the “Y” in the page blinks, press key, Y coordinate will be divided by 2;

In RELATIVE POS page, press and hold key till the “Z” in the page blinks, press key, Z coordinate will be divided by 2;
3) INTEGRATED POS display page

In INTEGRATED POS page, the RELATIVE, ABSOLUTE, MACHINE coordinate, DIST TO GO (only in Auto and MDI mode) are displayed together.

The displayed value of MACHINE coordinate is the current position in the machine coordinate system which is set up according to the machine zero.

DIST TO GO is the difference between the target position of block or MDI and the current position.

The display page is as follows:

<table>
<thead>
<tr>
<th>INTEGRATED POS</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>(RELATIVE)</td>
<td>(ABSOLUTE)</td>
</tr>
<tr>
<td>X 0.000</td>
<td>X 0.000</td>
</tr>
<tr>
<td>Y 0.000</td>
<td>Y 0.000</td>
</tr>
<tr>
<td>Z 0.000</td>
<td>Z 0.000</td>
</tr>
<tr>
<td>(MACHINE)</td>
<td>(DIST TO GO)</td>
</tr>
<tr>
<td>X 0.000</td>
<td>X 0.000</td>
</tr>
<tr>
<td>Y 0.000</td>
<td>Y 0.000</td>
</tr>
<tr>
<td>Z 0.000</td>
<td>Z 0.000</td>
</tr>
<tr>
<td>MDI</td>
<td>S0000 T00 H00</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4) POS&PRG display page

In this page, it displays ABSOLUTE, RELATIVE of the current position (ABSOLUTE, DIST TO GO of current position will be displayed if BIT0 of bit parameter No.180 is set to 1) and 5 blocks of current program together. During the program execution, the blocks displayed are refreshed dynamically and the cursor is located in the block being executed.

<table>
<thead>
<tr>
<th>POS &amp; PRG</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>(RELATIVE)</td>
<td>(ABSOLUTE)</td>
</tr>
<tr>
<td>X 0.000</td>
<td>X 0.000</td>
</tr>
<tr>
<td>Y 0.000</td>
<td>Y 0.000</td>
</tr>
<tr>
<td>Z 0.000</td>
<td>Z 0.000</td>
</tr>
<tr>
<td>(MACHINE)</td>
<td>(MACHINE)</td>
</tr>
<tr>
<td>X 0.000</td>
<td>X 0.000</td>
</tr>
<tr>
<td>Y 0.000</td>
<td>Y 0.000</td>
</tr>
<tr>
<td>Z 0.000</td>
<td>Z 0.000</td>
</tr>
<tr>
<td>00000 (00000);</td>
<td></td>
</tr>
</tbody>
</table>

1.3.2 Program interface

1) PROGRAM CONTENT page

PROGRAM is a compound key. Press PROGRAM key once to enter the program content interface.
all blocks will be displayed by pressing \[ \text{ } \] and \[ \text{ } \] keys in MDI mode.

<table>
<thead>
<tr>
<th>PRG CONTENT</th>
<th>SEG1</th>
<th>COL:1</th>
<th>C:/0000.CNC</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000 (00000);</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2) PROGRAM STATE page

Press \[ \text{ } \] key to enter program state interface in program content interface. Current G,M,S,T,F commands and related commands are displayed in program state interface and a single block (MDI) can be executed in this interface.

<table>
<thead>
<tr>
<th>PRG STATE</th>
<th>00000</th>
<th>N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>(ABSOLUTE)</td>
<td>Mode of fixed cycle</td>
<td>G00 G17 G90 G54: G21 G40 G49 G94 G98</td>
</tr>
<tr>
<td>X 0.000</td>
<td>X V</td>
<td></td>
</tr>
<tr>
<td>Y 0.000</td>
<td>Y W</td>
<td></td>
</tr>
<tr>
<td>Z 0.000</td>
<td>Z P</td>
<td></td>
</tr>
<tr>
<td>R 0.000</td>
<td>R Q</td>
<td></td>
</tr>
<tr>
<td>INPUT PRG SEGMENT:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>F0100 S 00 M00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRG. F: 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ACT. F: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FED OVRI: 150%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RAP OVRI: 100%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SPI OVRI: 100%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PART CNT: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CUT TIME: 0:00:00</td>
</tr>
<tr>
<td>MDI</td>
<td>S0000 T00 H00</td>
<td></td>
</tr>
</tbody>
</table>

3) PROGRAM PREVIEW page

In program content interface, press \[ \text{ } \] key to enter program preview page. In this page, all part programs are listed. To make it easier for user to select a program, the system displays 5 blocks before the program with cursor at the bottom of the page. User can press EOB directly to select a program and process automatically, or press DEL key to delete the program in this page. It displays the following contents:

(a) Memory capacity: Display the maximum capacity of CNC memory unit.
(b) Used capacity: The space occupied by the saved programs
(c) Program NO.: Display the total number of programs in the CNC (including subprograms)
(d) Size of the program: The size of the program which the cursor is in, unit: byte (B)
(e) Program list: Display numbers of saved programs (arranged by name).

<table>
<thead>
<tr>
<th>PRG PREVIEW</th>
<th>MEM SIZE: 40.0MB</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000 00001 00002 00003</td>
<td>MEM USED: 100KB</td>
</tr>
<tr>
<td></td>
<td>PRG AMOT: 4</td>
</tr>
<tr>
<td></td>
<td>PRG SIZE: 17B</td>
</tr>
<tr>
<td>00000 (0000);</td>
<td></td>
</tr>
<tr>
<td>:</td>
<td></td>
</tr>
<tr>
<td>EDIT</td>
<td>S0000 T00 H00</td>
</tr>
</tbody>
</table>

4) FILE LIST page
GSK980MDa supports USB interface, CNC→USB and USB→CNC mutual transmission operation are provided in this interface. In this page, it is easy to see the file list and file of CNC and USB (when USB is connected). At the same time, opening, duplication and deletion can be done here.

<table>
<thead>
<tr>
<th>FILE LIST</th>
<th>O0003 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:/</td>
<td>U:/</td>
</tr>
<tr>
<td>[00000.CNC</td>
<td>G50G51 Design-new</td>
</tr>
<tr>
<td>[00001.CNC</td>
<td>MZDataProc</td>
</tr>
<tr>
<td>[00002.CNC</td>
<td>2008-4'1</td>
</tr>
<tr>
<td>[00003.CNC</td>
<td>2y</td>
</tr>
</tbody>
</table>

INPUT: FILE INFO 17B 2009-12-28 10:10:31
NOTE: [CH]:C/U SHIFT [EOB]:OPEN [OUT]:COPY TO U FLASH
EDIT | S0000 T00 H00 |

1.3.3 Tool offset, macro variable and tool life management interface

OFFSET is a compound key, press OFFSET key once in other page to enter the TOOL OFFSET page, press OFFSET key again to enter the MACRO interface.
1. OFFSET interface

There are 4 tool offset pages in this interface, and 32 offset numbers (No.001 ~ No.032) available for user, which can be shown as the following figure by pressing or keys.

<table>
<thead>
<tr>
<th>NO.</th>
<th>Geo(H)</th>
<th>Wear(H)</th>
<th>Geo(D)</th>
<th>Wear(D)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>02</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>03</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>04</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>05</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>06</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>07</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>08</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NO.</th>
<th>001</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDIT</td>
<td>50000 T00 H00</td>
</tr>
</tbody>
</table>

2. MACRRO interface

There are 25 pages in this interface, which can be shown by pressing or keys. In Macro page there are 600 (No.100 ~ No.199 and No.500 ~ No.999) macro variables which can be specified by macro command or set by keypad. Please refer to “macro, chapter 5, program” for related information.

<table>
<thead>
<tr>
<th>NO.</th>
<th>DATA</th>
<th>NO.</th>
<th>DATA</th>
<th>NO.</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>_100</td>
<td>Null</td>
<td>108</td>
<td>Null</td>
<td>116</td>
<td>Null</td>
</tr>
<tr>
<td>101</td>
<td>Null</td>
<td>109</td>
<td>Null</td>
<td>117</td>
<td>Null</td>
</tr>
<tr>
<td>102</td>
<td>Null</td>
<td>110</td>
<td>Null</td>
<td>118</td>
<td>Null</td>
</tr>
<tr>
<td>103</td>
<td>Null</td>
<td>111</td>
<td>Null</td>
<td>119</td>
<td>Null</td>
</tr>
<tr>
<td>104</td>
<td>Null</td>
<td>112</td>
<td>Null</td>
<td>120</td>
<td>Null</td>
</tr>
<tr>
<td>105</td>
<td>Null</td>
<td>113</td>
<td>Null</td>
<td>121</td>
<td>Null</td>
</tr>
<tr>
<td>106</td>
<td>Null</td>
<td>114</td>
<td>Null</td>
<td>122</td>
<td>Null</td>
</tr>
<tr>
<td>107</td>
<td>Null</td>
<td>115</td>
<td>Null</td>
<td>123</td>
<td>Null</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NO.</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDIT</td>
<td>50000 T00 H00</td>
</tr>
</tbody>
</table>

3. Tool life management

Note: The tool change signal TLCH: F064#0 should be added for PLC when using this function.
Ladder example:

```
<table>
<thead>
<tr>
<th>F064.0</th>
<th>A5.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Tool life alarm</td>
</tr>
</tbody>
</table>
```

- **Using of tool life management function**

  Parameter (No.002#0) is used as the symbol for tool life management function (0—unused, 1—used); if the function is not used, the relevant tool life management page is not shown.

- **Tool life management display interface**

  The tool life management is controlled by key, which is displayed in the third sub-interface, and it is composed by 2 pages (paging by page keys). Interface is shown by pressing key repeatedly.

  **Tool life management display (the 1st page)**

  The 1st page for tool life management interface displays the life data of the current tool and the tool group list that has been defined. This page is mainly used for monitoring the tool life data by group units. The data monitoring of each tool in a group, group number setting and tool life management data are displayed in the following page.
i. Display explanation

<Current Tool State>: It displays the life data of the current tool which is being used.

Mode: It displays the counting unit of life data. (0: minute/1: times)

State: It displays the tool status. (0—Unused, 1—Using, 2—Over, 3—Skip)

<Defined Group No.>: It only displays the group numbers which have been defined, and the undefined are not shown. The group number with the backlight means that all the tool life in that group has expired.

ii. Deletion of all defined data

In this page, press  +  keys, it may delete all the data which have been defined (including group number, group tool numbers and life values, etc.)

Tool life management interface (the 2nd page)

The 2nd page is used to set and display the life data of a group which are displayed by order 1—8.

There are 3 display types for tool group selection:

i. Directly input the group number in the “Tool Group P” of the 2nd page, it displays the tool life
data. If the group does not exist, the number input will be taken as a new group number. The new group number: 05, and the 1\textsuperscript{st} tool will be defined by system automatically:

ii. Move the cursor to select the group number in the “Defined Group No.” of the 1\textsuperscript{st} page, and it displays the group content as turning to the 2\textsuperscript{nd} page.

iii. As the current group number content is displayed in the 2\textsuperscript{nd} page, it continues to display the following group number content by turning to the next page.

1.3.4 Alarm interface

Press \textbf{ALARM} key to enter Alarm interface, there are CNC ALARM, PLC ALARM, ALARM LOG pages in this interface, which can be viewed by or key.

1) PLC ALARM: It displays the numbers of CNC alarm, PLC alarm and the current PLC alarm No., as well as PLC warning and warning No.. It may display 24 PLC alarm or warning No. together. The details for the respective alarm No. can be viewed by moving the cursor. The page is as follows:

\begin{verbatim}
<table>
<thead>
<tr>
<th>PLC ALARM /WARN</th>
<th>00003 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNC ALM: 0.</td>
<td>PLC ALM: 1.</td>
</tr>
<tr>
<td></td>
<td>PLC WARN: 0.</td>
</tr>
<tr>
<td>ALM NO:1000</td>
<td>BIT ADDR: A0000.0</td>
</tr>
<tr>
<td>Illegal M code</td>
<td></td>
</tr>
<tr>
<td>MDI</td>
<td>S0000 T00 H00</td>
</tr>
</tbody>
</table>
\end{verbatim}

Page as the cursor locates at the alarm No.1000

2) CNC ALARM: It displays the numbers of CNC alarm, PLC alarm and the current CNC alarm No.. It can display 24 CNC alarm No. together. The details for the respective alarm No. can be viewed by moving the cursor. The page is as follows:
3) WARN LOG: Press key to enter Alarm interface, then press it again to enter the WARN LOG page, which records the latest alarm message including alarm date, alarm time, alarm No. and alarm content. 200 pieces warn log messages can be viewed by or key. See the following figure:

Sequence of warn log: the latest alarm log message is shown on the forefront of the 1st page, and the others queue in sequence. If the alarm log messages are over 200, the last one will be cleared.

Manual clearing of warn log: under the 2 level authority, press key, it may clear all the warn log messages.

Alarm clearing: If multiple alarms are issued, only one alarm where the cursor locates could be cleared by pressing key each time (In alarm interface, it clears all alarms and warnings...
by pressing \( \text{RESET} \) and \( \text{CANCEL} \) keys).

5) The current alarm page is as follows:

<table>
<thead>
<tr>
<th>CNC ALARM</th>
<th>00003 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNC ALM : 3.</td>
<td>PLC ALM: 0 .</td>
</tr>
<tr>
<td>000 432</td>
<td></td>
</tr>
<tr>
<td>CTR WARN:431</td>
<td></td>
</tr>
<tr>
<td>X axis driver is not ready</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MDI</th>
<th>ESP. ALM</th>
<th>S0000 T00 H00</th>
</tr>
</thead>
</table>

Current page

<table>
<thead>
<tr>
<th>CNC ALARM</th>
<th>00003 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNC ALM : 2.</td>
<td>PLC ALM: 0 .</td>
</tr>
<tr>
<td></td>
<td>431</td>
</tr>
<tr>
<td>CTR WARN:432</td>
<td></td>
</tr>
<tr>
<td>Y axis driver is not ready</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MDI</th>
<th>S0000 T00 H00</th>
</tr>
</thead>
</table>

Page after pressing RESET key

6) Clearing PLC warning: If multiple warnings are issued, only one warning where the cursor locates could be cleared by pressing \( \text{RESET} \) or \( \text{CANCEL} \) key each time (In Alarm interface, it clears all alarms and warnings by pressing \( \text{RESET} \) and \( \text{CANCEL} \) keys).

1.3.5 Setting interface

\( \text{SETTING} \) is a compound key, press \( \text{SETTING} \) key in other page, it enters setting interface, press it again, it enters the G54~G59 interface, press it three times, it enters Graphic interface. Press key repeatedly, it switches among the above mentioned interfaces.
1. Setting interface

There are 3 pages in this interface, which can be viewed by and keys.

1) SWITCH SETTING: It is used for displaying the parameter, program, auto sequence No. on/off state.

   PARM SWT: when it is turned ON, the parameters are allowed to be modified; it is turned OFF, the parameters are unallowed to be modified.

   PROG SWT: when it is turned ON, the programs are allowed to be edited; it is turned OFF, the programs are unallowed to be edited.

   AUTO SEG: when it is turned ON, the block No. is created automatically; it is turned OFF, the block No. is not created automatically, input manually if it is needed.

   In this page, the state of on/off can be switched by ‘left / right’ key or ‘U’ and ‘D’ key on the MDI panel.

<table>
<thead>
<tr>
<th>SWITCH SETTING</th>
<th>00003 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>➤ PARM SWT: OFF *ON</td>
<td></td>
</tr>
<tr>
<td>PROG SWT: OFF *ON</td>
<td></td>
</tr>
<tr>
<td>AUTO SEG: *OFF ON</td>
<td></td>
</tr>
</tbody>
</table>

   | MDI                     | S0000 T00 H00 |

2) Data backup: In this page, the CNC data (bit parameter, data parameter, pitch parameter, tool offset) can be saved and restored.

   Data backup (user): For CNC data backup by user (save)

   Recover backup data (user): For backup data recover by user (read)

   Recover standard parameter 1 (test): For reading original parameter data of CNC test by user

   Recover standard parameter 2 (step): For reading original parameter data of suited step drive unit by user

   Recover standard parameter 3 (servo): For reading original parameter data of suited servo drive unit by user.
3) Password setting: Display and set user operation level.

The password of GSK980MDa is composed of 4 levels, including machine builder (level 2), equipment management (level 3), technician (level 4) and machining operation (level 5).

Machine builder (level 2): It allows to modify CNC bit parameter, data parameter, screw- pitch parameter, tool offset parameter, edit part program (including macro program), edit and alter PLC ladder diagram, upload and download ladder diagram.

Equipment management (level 3): Initial password is 12345. The CNC bit parameter, data parameter, screw- pitch parameter, tool offset parameter, part program editing operations are allowed.

Technician (level 4): Initial password is 1234. Tool offset data (for tool setting), macro variables, part program editing operations are allowed. However, CNC bit parameter, data parameter and pitch parameter editing operations are not allowed.

Machining operation (level 5): No password. Only the machine panel operation is allowed. The alteration of tool offset data, CNC bit parameter, data parameter, pitch parameter, and the operations of part program selection, program editing are not allowed.
1. Setting page of G54～G59 Page location

Press the SETTING key twice, this page is displayed.

<table>
<thead>
<tr>
<th>SET (G54～G59)</th>
<th>00003 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>(EXT OFFSET)</td>
<td>(G54 COORDINATE)</td>
</tr>
<tr>
<td>X 0.000</td>
<td>X 0.000</td>
</tr>
<tr>
<td>Y 0.000</td>
<td>Y 0.000</td>
</tr>
<tr>
<td>Z 0.000</td>
<td>Z 0.000</td>
</tr>
<tr>
<td>(G55 COORDINATE)</td>
<td>(G56 COORDINATE)</td>
</tr>
<tr>
<td>X 0.000</td>
<td>X 0.000</td>
</tr>
<tr>
<td>Y 0.000</td>
<td>Y 0.000</td>
</tr>
<tr>
<td>Z 0.000</td>
<td>Z 0.000</td>
</tr>
</tbody>
</table>

The zero of the coordinate system: workpiece coordinate system zero offset, G54, G55, G56, G57, G58, G59.

- Moving of the cursor
  The cursor moves at the data of each coordinate system axis. And the data where the cursor...
locates are highlighted.

The cursor supports up and down, left and right moving, and the corresponding data are backlit.

By pressing Page key, the 1st group X axis data on the corresponding interface where the cursor locates is backlit.

- Absolute data input

After “data+” key is keyed in by user, the data where the cursor locates is changed to the “data” input by user.

The validity judgement of user input data is the same as that of 980TD coordinate data input in MDI mode.

- Relative data input

After “data+” key is keyed in by user, the original data where the cursor locates is changed by the sum of “data” newly input by user and original data.

- Auto measurement input

After “” (or ˈ) key” is keyed in by user, the original data where the cursor locates is changed by the system current “X (or Z, Y) axis machine coordinate”.

3. Graphic interface

There are GRAPH SET, GRAPH TRACK pages in this interface, which can be viewed by keys.

1) GRAPH SET page

In this page, the coordinate system, scaling and scope for graphic display can be selected.

<table>
<thead>
<tr>
<th>GRAPH SET</th>
<th>00003 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>COOR OPT=</td>
<td>0 0XY 1YX 2XX 3XYZ 4YZ 5ZY 6XZ 7XYZ)</td>
</tr>
<tr>
<td>SCALE</td>
<td>100%</td>
</tr>
<tr>
<td>CENTER</td>
<td>0.000 (X axis value)</td>
</tr>
<tr>
<td>CENTER</td>
<td>0.000 (Y axis value)</td>
</tr>
<tr>
<td>CENTER</td>
<td>0.000 (Z axis value)</td>
</tr>
<tr>
<td>X MAX.</td>
<td>120.000</td>
</tr>
<tr>
<td>Y MAX.</td>
<td>120.000</td>
</tr>
<tr>
<td>Z MAX.</td>
<td>120.000</td>
</tr>
<tr>
<td>X MIN.</td>
<td>-120.000</td>
</tr>
<tr>
<td>Y MIN.</td>
<td>-120.000</td>
</tr>
<tr>
<td>Z MIN.</td>
<td>-120.000</td>
</tr>
</tbody>
</table>

MDI | S0000 T00 H00 |
2) GRAPH TRACK page

In this page, it displays the path within the parameters range (refer to absolute coordinate) of GRAPH SET page.

<table>
<thead>
<tr>
<th>GRAPH TRACK</th>
<th>00003 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>(ABSOLUTE)</td>
<td></td>
</tr>
<tr>
<td>X 62.827</td>
<td></td>
</tr>
<tr>
<td>Y 10.133</td>
<td></td>
</tr>
<tr>
<td>Z -573.547</td>
<td></td>
</tr>
<tr>
<td>S: STARTT</td>
<td></td>
</tr>
<tr>
<td>T: STOP</td>
<td></td>
</tr>
<tr>
<td>R: CLEAR</td>
<td></td>
</tr>
<tr>
<td>K: SFT VIEW</td>
<td></td>
</tr>
<tr>
<td>J: MEDIACY</td>
<td></td>
</tr>
<tr>
<td>IM: IN/OUT</td>
<td></td>
</tr>
<tr>
<td>S0000 T00 H00</td>
<td></td>
</tr>
</tbody>
</table>

1.3.6 BIT PARAMETER, DATA PARAMETER, PITCH COMP interface

![BIT PARAMETER interface](image)

is a compound key, it enters BIT PARAMETER, DATA PARAMETER and PITCH COMP interfaces by pressing this key repeatedly.

1. BIT PARAMETER interface

Press ![BIT PARAMETER interface](image) key, it enters BIT PARAMETER interface, there are 48 bit parameters which are displayed by 2 pages in this interface, and they can be viewed or modified by pressing ![DATA PARAMETER interface](image) or ![PARAMETER interface](image) key to enter the corresponding page. It is as follows:

As is shown in this page, there are 2 parameter rows at the bottom of the page, the 1\textsuperscript{st} row shows the meaning of a bit of a parameter where the cursor locates, the bit to be displayed can be positioned by pressing ![BIT PARAMETER interface](image) or ![BIT PARAMETER interface](image) key. The 2nd row shows the abbreviation of all the bits of a parameter where the cursor locates.
2. DATA PARAMETER interface

Press \textbf{PARAMETER} key repeatedly (\textbf{FUNCTION} key if in BIT PARAMETER interface), it enters DATA PARAMETER interface, there are 110 data parameters which are displayed by 7 pages in this interface, and they can be viewed or modified by pressing \textbf{PARAMETER} or \textbf{FUNCTION} key to enter the corresponding page. It is as follows:

As is shown in this page, there is a cue line at the page bottom, it displays the meaning of the parameter where the cursor locates.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|}
\hline
\textbf{NO.} & \textbf{DATA} & \textbf{NO.} & \textbf{DATA} & \textbf{NO.} & \textbf{DATA} \\
\hline
001 & 00000000 & 009 & 00000000 & 017 & 00101000 \\
002 & 00000011 & 010 & 00011111 & 018 & 00000000 \\
003 & 00000000 & 011 & 00000000 & 019 & 10000000 \\
004 & 00000000 & 012 & 00010011 & 020 & 00000000 \\
005 & 00010001 & 013 & 10000011 & 021 & 00000000 \\
006 & 00000000 & 014 & 00011111 & 022 & 00000000 \\
007 & 00000000 & 015 & 10000000 & 023 & 00000000 \\
008 & 00011111 & 016 & 00000000 & 024 & 00000000 \\
\hline
\end{tabular}
\end{table}

\textbf{PITCH COMP} interface

Press \textbf{PARAMETER} key repeatedly, it enters PITCH COMP interface, there are 256 pitch parameters which are displayed by 16 pages in this interface, and they can be viewed by pressing \textbf{PARAMETER} or \textbf{FUNCTION} key.
1.3.7 CNC DIAGNOSIS, PLC STATE, PLC VALUE, machine soft panel, VERSION MESSAGE interface

is a compound key, it enters CNC DIAGNOSIS, PLC STATE, PLC VALUE, machine soft panel, VERSION MESSAGE interfaces by pressing this key repeatedly.

1. CNC DIAGNOSIS interface CNC

The input/output signal state between CNC and machine, the transmission signal state between CNC and PLC, PLC internal data and CNC internal state can all be displayed via diagnosis. Press key it enters CNC DIAGNOSIS interface, the keypad diagnosis, state diagnosis and miscellaneous function parameters etc. can be shown in this interface, which can be viewed by pressing or key.

In CNC DIAGNOSIS page, there are 2 diagnosis No. rows at the page bottom, the 1st row shows the meaning of a diagnosis No. bit where the cursor locates, the bit to be displayed can be positioned by pressing or key. The 2nd row shows the abbreviation of all the diagnosis No. bits where the cursor locates.
2. PLC STATE interface

In the page of this interface, it orderly displays the state of address X0000~X0029, Y0000~Y0019, F0000~F0255, G0000~G0255, A0000~A0024, K0000~K0039, R0000~R0999 etc..

And it enters PLC STATE interface by pressing key repeatedly. The signal state of PLC addresses can be viewed by pressing or key.

In PLC STATE page, there are 2 rows at the page bottom; the 1st row shows the meaning of a bit of an address where the cursor locates, the bit to be displayed can be positioned by pressing or key. The 2nd row shows the abbreviation of all the bits of an address where the cursor locates.

3. PLC VALUE interface
In the page of this interface, it orderly displays the values in the registers of T0000～T0099, D0000～D0999, C0000～C0099, DT000～DT099, DC000～DC099 etc.. By pressing key repeatedly it enters PLC VALUE interface. The data values of PLC can be viewed by pressing key.

In this PLC VALUE page, there is a cue line at the page bottom, it displays the meaning of the parameter where the cursor locates. As is shown in the following figure:

<table>
<thead>
<tr>
<th>PLC DATA</th>
<th>00003</th>
<th>N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO. DATA</td>
<td>NO. DATA</td>
<td>NO. DATA</td>
</tr>
<tr>
<td>DT000 0</td>
<td>DT008 0</td>
<td>DT016 0</td>
</tr>
<tr>
<td>DT001 0</td>
<td>DT009 0</td>
<td>DT017 0</td>
</tr>
<tr>
<td>DT002 0</td>
<td>DT010 0</td>
<td>DT018 0</td>
</tr>
<tr>
<td>DT003 0</td>
<td>DT011 0</td>
<td>DT019 100</td>
</tr>
<tr>
<td>DT004 0</td>
<td>DT012 0</td>
<td>DT020 500</td>
</tr>
<tr>
<td>DT005 0</td>
<td>DT013 0</td>
<td>DT021 500</td>
</tr>
<tr>
<td>DT006 0</td>
<td>DT014 0</td>
<td>DT022 100</td>
</tr>
<tr>
<td>DT007 0</td>
<td>DT015 0</td>
<td>DT023 500</td>
</tr>
</tbody>
</table>

Reserved
NO. DT000

JOG S0000 T00 H00

4. VERSION MESSAGE interface

It enters VERSION MESSAGE interface by pressing key repeatedly. The software, hardware, and PLC version message can be shown in this interface. The figure is as follows:

<table>
<thead>
<tr>
<th>VERSION MESSAGE</th>
<th>00000</th>
<th>N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT TYPE</td>
<td>GSK980MDa</td>
<td></td>
</tr>
<tr>
<td>SOFTWARE VER.</td>
<td>V2.00-manu</td>
<td></td>
</tr>
<tr>
<td>HARDWARE VER.</td>
<td>3.01.002--08.07.21</td>
<td></td>
</tr>
<tr>
<td>SYSTEM ID</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>LADDER DESIGN</td>
<td>GSK</td>
<td></td>
</tr>
<tr>
<td>LADDER VER.</td>
<td>09.01.15-839C</td>
<td></td>
</tr>
<tr>
<td>LADDER VERIFY</td>
<td>839C</td>
<td></td>
</tr>
<tr>
<td>LADDER NOTE</td>
<td>GSK980MDa</td>
<td></td>
</tr>
</tbody>
</table>

MDI S0000 T00 H00
### 1.4 List of general operations

<table>
<thead>
<tr>
<th>Item</th>
<th>Function</th>
<th>Operation key</th>
<th>Operation mode</th>
<th>Display page</th>
<th>Passsword level</th>
<th>Program on/off</th>
<th>Parameter switch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clearing</td>
<td>Relative coordinate of X axis clearing</td>
<td></td>
<td>Relative coordinate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Relative coordinate of Y axis clearing</td>
<td></td>
<td>Relative coordinate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Relative coordinate of Z axis clearing</td>
<td></td>
<td>Relative coordinate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Part No. clearing</td>
<td>CANCEL</td>
<td>Relative coordinate or absolute coordinate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cutting time clearing</td>
<td>CANCEL</td>
<td>Relative coordinate or absolute coordinate</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tool radius offset D clearing</td>
<td>DATA INPUT</td>
<td>Tool offset</td>
<td>Level 2,3,4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tool length offset H clearing</td>
<td>DATA INPUT</td>
<td>Tool offset</td>
<td>Level 2,3,4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data input</td>
<td>Bit parameter parameter</td>
<td>DATA INPUT</td>
<td>MDI mode</td>
<td>Bit parameter</td>
<td>Level 2,3</td>
<td></td>
<td>On</td>
</tr>
<tr>
<td></td>
<td>Data parameter</td>
<td>DATA INPUT</td>
<td>MDI mode</td>
<td>Bit parameter</td>
<td>Level 2,3</td>
<td></td>
<td>On</td>
</tr>
<tr>
<td>Item</td>
<td>Function</td>
<td>Operation key</td>
<td>Operation mode</td>
<td>Display page</td>
<td>Password level</td>
<td>Program on/off</td>
<td>Parameter switch</td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
<td>---------------</td>
<td>----------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Input pitch parameter of X axis</td>
<td></td>
<td>Compensation value.</td>
<td>MDI mode</td>
<td>Pitch parameter</td>
<td>Level 2</td>
<td></td>
<td>On</td>
</tr>
<tr>
<td>Input pitch parameter of Y axis</td>
<td></td>
<td>Compensation value.</td>
<td>MDI mode</td>
<td>Pitch parameter</td>
<td>Level 2</td>
<td></td>
<td>On</td>
</tr>
<tr>
<td>Input pitch parameter of Z axis</td>
<td></td>
<td>Compensation value.</td>
<td>MDI mode</td>
<td>Pitch compensation parameter</td>
<td>Level 2</td>
<td></td>
<td>On</td>
</tr>
<tr>
<td>Macro variables</td>
<td></td>
<td>Macro variables.</td>
<td></td>
<td>Macro variables</td>
<td>Level 2,3,4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input tool radius offset D</td>
<td></td>
<td>Data value.</td>
<td></td>
<td>Tool offset</td>
<td>Level 2,3,4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input tool length offset H</td>
<td></td>
<td>Data value.</td>
<td></td>
<td>Tool offset</td>
<td>Level 2,3,4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Search down from where the cursor locates</td>
<td></td>
<td>Character.</td>
<td>Edit mode</td>
<td>Program content</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>Search up from where the cursor locates</td>
<td></td>
<td>Character.</td>
<td>Edit mode</td>
<td>Program content</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>Search down from current program</td>
<td></td>
<td></td>
<td>Edit mode or auto mode</td>
<td>Program content, program list or program state</td>
<td>Level 2,3,4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Item</td>
<td>Function</td>
<td>Operation key</td>
<td>Operation key</td>
<td>Operation mode</td>
<td>Display page</td>
<td>Password level</td>
<td>Program on/off</td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
<td>---------------</td>
<td>---------------</td>
<td>----------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td></td>
<td>Search defined program</td>
<td>-</td>
<td>-</td>
<td>2 级, 3 级, 4 级 Level 2,3,4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Search for bit parameter, data parameter or pitch parameter</td>
<td>-</td>
<td>-</td>
<td>Corresponding page of the data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PLC state, PLC data searching</td>
<td>-</td>
<td>-</td>
<td>PLC state, PLC data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deletion</td>
<td>Delete the character where the cursor is in</td>
<td>DELETE</td>
<td>Edit mode</td>
<td>Program content Level 2,3,4</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CANCEL</td>
<td>Edit mode</td>
<td>Program content Level 2,3,4</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Single block deletion</td>
<td>DELETE</td>
<td>Edit mode</td>
<td>Program content Level 2,3,4</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Multi-block deletion</td>
<td>CHANGE</td>
<td>Edit mode</td>
<td>Program content Level 2,3,4</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Segment deletion</td>
<td>CHANGE</td>
<td>Edit mode</td>
<td>Program content Level 2,3,4</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Item</td>
<td>Function</td>
<td>Operation key</td>
<td>Operation mode</td>
<td>Display page</td>
<td>Passwor d level</td>
<td>Program on/off</td>
<td>Parameter switch</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------</td>
<td>---------------</td>
<td>----------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Delete one program</td>
<td>program name.</td>
<td>DELETE</td>
<td>Edit mode</td>
<td>Program</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>Delete all programs</td>
<td>999.</td>
<td>DELETE</td>
<td>Edit mode</td>
<td>Program</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>Change name</td>
<td>program name.</td>
<td>ALTER</td>
<td>Edit mode</td>
<td>Program</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>Duplication</td>
<td>program name.</td>
<td>CHANGE</td>
<td>Edit mode</td>
<td>Program</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>CNC (send)</td>
<td>Tool offset</td>
<td>DATA OUTPUT</td>
<td>Edit mode</td>
<td>Tool offset</td>
<td>Level 2,3</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bit parameter</td>
<td>DATA OUTPUT</td>
<td>Edit mode</td>
<td>Bit parameter</td>
<td>Level 2,3</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data parameter</td>
<td>DATA OUTPUT</td>
<td>Edit mode</td>
<td>Data parameter</td>
<td>Level 2,3</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pitch parameter</td>
<td>DATA OUTPUT</td>
<td>Edit mode</td>
<td>Pitch parameter</td>
<td>Level 2</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>Send a part program</td>
<td>program name,</td>
<td>DATA OUTPUT</td>
<td>Edit mode</td>
<td>Program</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>Item</td>
<td>Function</td>
<td>Operation key</td>
<td>Operation mode</td>
<td>Display page</td>
<td>Password level</td>
<td>Program on/off</td>
<td>Parameter switch</td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
<td>---------------</td>
<td>----------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>------------------</td>
</tr>
<tr>
<td></td>
<td>Send all part programs</td>
<td>[DATA OUTPUT]</td>
<td>Edit mode</td>
<td>Program content</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tool offset</td>
<td>[DATA OUTPUT]</td>
<td>Edit mode</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CNC</td>
<td>Tool offset</td>
<td>[DATA OUTPUT]</td>
<td>Edit mode</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PC (receive)</td>
<td>Bit parameter</td>
<td>[DATA OUTPUT]</td>
<td>Edit mode</td>
<td>Level 2,3</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CNC</td>
<td>Bit parameter</td>
<td>[DATA OUTPUT]</td>
<td>Edit mode</td>
<td>Data parameter</td>
<td>Level 2,3</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td>PC (upload)</td>
<td>Bit parameter</td>
<td>[DATA OUTPUT]</td>
<td>Edit mode</td>
<td>Pitch compensation parameter</td>
<td>Level 2</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Send a program</td>
<td>[DATA OUTPUT]</td>
<td>Edit mode</td>
<td>Program content</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Send all programs</td>
<td>[DATA OUTPUT]</td>
<td>Edit mode</td>
<td>Level 2,3,4</td>
<td>On</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Volume II Operation

<table>
<thead>
<tr>
<th>Item</th>
<th>Function</th>
<th>Operation key</th>
<th>Operation mode</th>
<th>Display page</th>
<th>Password level</th>
<th>Program on/off</th>
<th>Parameter switch</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Tool offset</td>
<td>Edit mode</td>
<td>Level</td>
<td>2,3,4</td>
<td></td>
<td></td>
<td>On</td>
</tr>
<tr>
<td></td>
<td>Bit parameter</td>
<td>Edit mode</td>
<td>Level</td>
<td>2,3,4</td>
<td></td>
<td></td>
<td>On</td>
</tr>
<tr>
<td></td>
<td>Data parameter</td>
<td>Edit mode</td>
<td>Level</td>
<td>2,3,4</td>
<td></td>
<td></td>
<td>On</td>
</tr>
<tr>
<td></td>
<td>Pitch parameter</td>
<td>Edit mode</td>
<td>Level</td>
<td>2,3,4</td>
<td></td>
<td></td>
<td>On</td>
</tr>
<tr>
<td></td>
<td>Part program</td>
<td>Edit mode</td>
<td>Level</td>
<td>2,3,4</td>
<td></td>
<td></td>
<td>On</td>
</tr>
</tbody>
</table>

#### Switch setting

<table>
<thead>
<tr>
<th>Action</th>
<th>Key(s)</th>
<th>Setting</th>
<th>Level(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turn on parameter switch</td>
<td></td>
<td>Switch setting</td>
<td>2,3,4</td>
</tr>
<tr>
<td>Turn on program switch</td>
<td></td>
<td>Switch setting</td>
<td>2,3,4</td>
</tr>
<tr>
<td>Turn on auto sequence No.</td>
<td></td>
<td>Switch setting</td>
<td></td>
</tr>
<tr>
<td>Turn off parameter switch</td>
<td></td>
<td>Switch setting</td>
<td>2,3,4</td>
</tr>
<tr>
<td>Turn off program switch</td>
<td></td>
<td>Switch setting</td>
<td>2,3,4</td>
</tr>
<tr>
<td>Turn off auto sequence No.</td>
<td></td>
<td>Switch setting</td>
<td></td>
</tr>
</tbody>
</table>

#### Explanations

- “.” in the column “operation” indicates operate two keys successively, “+” indicates operate two keys simultaneously.

**Example:**

- Indicates that press  key first, and then press  key;  +  indicates that press two keys simultaneously.
2.1 System Power On

Before this GSK980MDa is powered on, the following should be confirmed:

1. The machine is in a normal state.
2. The power voltage conforms to the requirement of the machine.
3. The connection is correct and secure.

The following page is displayed after GSK980MDa is powered on:

![Image of GSK logo]

The current position (RELATIVE POS) page is displayed after system auto detection and initiation are finished.

<table>
<thead>
<tr>
<th>RELATIVE POS</th>
<th>O0000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.000</td>
</tr>
<tr>
<td>Y</td>
<td>13.776</td>
</tr>
<tr>
<td>Z</td>
<td>-1.344</td>
</tr>
</tbody>
</table>

![Image of GSK logo with numbers]

2.2 System Power Off

Before power is off, ensure that:

1. The axes of the CNC are at halt;
2. Miscellaneous functions (spindle, pump etc.) are off;
3. Cut off CNC power prior to machine power cutting off.

Note: Please see the machine builder’s manual for the machine power cut-off operation.
2.3 Overtravel Protection

Overtravel protection should be employed to prevent the damage to the machine due to the overtravel of the axes.

2.3.1 Hardware overtravel protection

The stroke switches are fixed at the positive and negative maximum travel of the machine axes X, Y, Z, 4th, 5th respectively, they are connected by the following figure. And the “MESP”of bit parameter No.017 must be setted to 0. If the overtravel occurs, the stroke switch acts to make the machine stop, and the emergency alarm issues.

When the hardware overtravel occurs, there will be an “emergency stop” alarm. The steps to eliminate this alarm is press the OVERTRAVEL button to reversely move the table to detach the stroke switch (for positive overtravel, move negatively; vice versa).

2.3.2 Software overtravel protection

When the “MOT” of bit parameter No.17 is set to 0, the software limit is valid.

The software travel stroke is set by data parameter NO.135~NO.144, they refer to machine coordinate. No.135~No.139 are for axes (X, Y, Z, 4th, 5th) positive max.overtravel, No.140~No.144 are for negative max.overtravel.

If the machine position (coordinate) exceeds the setting range, overtravel alarm will occur. The steps to eliminate this alarm is press RESET key to clear the alarm, then moves reversely (for positive overtravel, move out negatively; vice versa)

2.4 Emergency Operation

During the machining, some unexpected incidents may occur because of the user programming, operation and product fault. So this GSK980MDa should stopped immediately for these incidents. This section mainly describes the resolutions that this GSK980MDa are capable of under the emergency situation. Please see the relative explanation for these resolutions under the emergency by machine builder.
2.4.1 Reset

Press the [RESET] key to reset this GSK980MDa system if there are abnormal outputs and axis actions in it:

1. All axes movement stops;
2. M, S function output is invalid (PLC ladder defines whether automatically cut off signals such as spindle CCW/CW, lubrication, cooling by pressing the [RESET] key);
3. Auto run ends, modal function and state held on.

2.4.2 Emergency stop

During machine running, if the emergency button is pressed under the dangerous or emergent situation, the CNC system enters into emergency status and the machine movement is stopped immediately. If the emergency button is released, the emergency alarm is cancelled and the CNC resets. Its circuit wiring is shown in section 2.2.1 of this chapter.

Note 1 Ensure the fault is eliminated before the emergency alarm is cancelled.

Note 2 pressing down the Emergency button prior to power on or off may alleviate the electric shock to the machine system.

Note 3 Reperform the machine zero return operation to ensure the correct position coordinate after the emergency alarm is cancelled (machine zero return operation is unallowed if there is no machine zero on the machine.).

Note 4 Only the MESP of the bit parameter No.017 is set to 0, is the external emergency stop valid.

2.4.3 Feed hold

Key can be pressed during the machine running to make the running pause. However, in thread cutting, cycle running, this function can not stop the running immediately.

2.4.4 Power off

Under the dangerous or emergency situations during the machine running, the machine power should be cut off immediately to avoid the accidents. However, it should be noted that there may be a big error between the CNC displayed coordinate and the actual position. So the tool setting operation should be performed again.
CHAPTER 3 MANUAL OPERATION

Press the **Manual** key, it enters Manual mode. In this mode, the manual feed, spindle control, override adjustment operations can be performed.

Note!
The keys functions of this 980MDa machine panel are defined by Ladder Diagram; please refer to the respective materials by the machine builder for the function significance.

Please note that the following function introduction is described based on the 980MDa standard PLC programs!

3.1 Coordinate axis moving

In Manual mode, the coordinate axis can be moved manually for feeding and rapid traverse.

3.1.1 Manual feed

Press feed axis and axis direction key in the direction selection area.

The corresponding axis may be moved positively or negatively, and the axis stops moving if releasing these two keys; and the direction selection keys of X, Y, Z, 4th, 5th axes can be hold on at a time to make the 5 axes to move simultaneously.

3.1.2 Manual rapid traverse

First press the **RAP** key in the feed axis and direction selection area.

Till the rapid traverse indicator in the State area lights.
The corresponding axis can be rapidly moved positively or negatively by pressing direction selection key, and the axis stops moving if releasing the key; and the direction selection keys of X, Y, Z, 4th, 5th axes can be hold on at a time to make the 5 axes to move simultaneously.

In Manual rapid mode, press key to make the indicator go out, and the rapid traverse is invalid, it enters the Manual feed mode.

**Note 1:** Before machine zero return, the validity of manual rapid traverse is set by the “ISOT” of the bit parameter No.012.

**Note 2:** In Edit or MPG mode, key is invalid.

### 3.1.3 Manual feedrate override adjustment

In Manual mode, the or key in can be pressed to modify the Manual feedrate override, and the override has 16 levels. The relation of the feedrate override and the feedrate is as the following table:

<table>
<thead>
<tr>
<th>Feedrate override (%)</th>
<th>Feedrate (mm/min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>2.0</td>
</tr>
<tr>
<td>20</td>
<td>3.2</td>
</tr>
<tr>
<td>30</td>
<td>5.0</td>
</tr>
<tr>
<td>40</td>
<td>7.9</td>
</tr>
<tr>
<td>50</td>
<td>12.6</td>
</tr>
<tr>
<td>60</td>
<td>20</td>
</tr>
<tr>
<td>70</td>
<td>32</td>
</tr>
<tr>
<td>80</td>
<td>50</td>
</tr>
<tr>
<td>90</td>
<td>79</td>
</tr>
<tr>
<td>100</td>
<td>126</td>
</tr>
<tr>
<td>110</td>
<td>200</td>
</tr>
<tr>
<td>120</td>
<td>320</td>
</tr>
<tr>
<td>130</td>
<td>500</td>
</tr>
<tr>
<td>140</td>
<td>790</td>
</tr>
<tr>
<td>150</td>
<td>1260</td>
</tr>
</tbody>
</table>

**Note:** There is about 2% fluctuating error for the data in the table.
3.1.4 Manual rapid override adjustment

In the manual rapid traverse, or key in can be pressed (also by key with the respective override F0, 25%, 50%, 100%) to modify the Manual rapid override, and there are 4 gears of F0, 25%, 50%, 100% for the override. (F0 is set by data parameter No.069)

3.1.5 Relative coordinate clearing

1) Press key to enter Position interface, then press or key to select the RELATIVE POS page;

<table>
<thead>
<tr>
<th>RELATIVE POS</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>1.680</td>
</tr>
<tr>
<td>Y</td>
<td>13.776</td>
</tr>
<tr>
<td>Z</td>
<td>-1.344</td>
</tr>
<tr>
<td>JOG</td>
<td>S0000 T00 H00</td>
</tr>
</tbody>
</table>

2) Press key to make the “X” in the page to blink, then press key;
The clearing operations of other coordinates are the same as above.

### 3.2 Other Manual operations

**Note:** The following operations are also valid in Machine zero, MPG/Step mode.

#### 3.2.1 Spindle CCW, CW, stop control

- **CW:** In Manual mode, the spindle rotates counter-clockwise if pressing this key;
- **STOP:** In Manual mode, the spindle stops if pressing this key;
- **CCW:** In Manual mode, the spindle rotates clockwise if pressing this key;

#### 3.2.2 Spindle Jog

Press and hold the Jog key, the spindle rotates counter-clockwise, release it, the spindle stops.

#### 3.2.3 Cooling control

In Manual mode, press this key, the coolant is switched on/off.

#### 3.2.4 Lubrication control

See details in Appendix for its function.
3.2.5 Spindle override adjustment

In Manual mode, if the spindle speed is controlled by analog voltage output, the spindle speed may be adjusted.

By pressing the \( \downarrow \) or \( \uparrow \) key in Spindle Override keys, the spindle speed can be changed by real-time adjusting of the spindle override that has 8 levels of 50\%~120\%.
CHAPTER 4  MPG/STEP OPERATION

In MPG/Step mode, the machine moves by a specified increment.

4.1 Step Feed

Set the BIT3 of the bit parameter No.001 to 0, and press key to enter the Step mode, it displays as follows:

<table>
<thead>
<tr>
<th>RELATIVE POS</th>
<th>O0000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.000</td>
</tr>
<tr>
<td>Y</td>
<td>0.000</td>
</tr>
<tr>
<td>Z</td>
<td>0.000</td>
</tr>
<tr>
<td>STEP</td>
<td>$0000 T01 H00</td>
</tr>
</tbody>
</table>

4.1.1 Increment selection

Press key to select the move increment, the increment will be shown in the page.

Note: In the EDIT or REF modes, keys are invalid. In the AUTO or MDI modes, rapid override will be changed by pressing the above-mentioned keys. In the MANUAL mode, press rapid move key and keys together, these keys are valid, otherwise, they are invalid.
4.1.2 Moving direction selection

Press \[ \text{ or } \text{ key once, X axis can be moved negatively or positively by a step increment, other axises are the same.} \]

4.2 MPG (Handwheel) Feed

Set the BIT3 of the bit parameter No.001 to 1, and press \( \text{ key to enter the MPG mode, it displays as following:} \)

<table>
<thead>
<tr>
<th>RELATIVE POS</th>
<th>O0000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>0.000</td>
</tr>
<tr>
<td>Y</td>
<td>0.000</td>
</tr>
<tr>
<td>Z</td>
<td>0.000</td>
</tr>
<tr>
<td>HNDL</td>
<td>S0000 T01 H00</td>
</tr>
</tbody>
</table>

The handwheel figure is as follows:

![Handwheel Figure]

4.2.1 Increment selection

Press \( \text{ key to select the move increment, the increment will be shown in the page:} \)
4.2.2 Moving axis and direction selection

In MPG mode, press the key to select the corresponding axis. The page is as follows (Other axises are the same):

The handwheel feed direction is defined by its rotation direction. Generally, the handwheel CW is for positive feed, and CCW is for negative feed. In case of that handwheel CW is for negative feed, CCW for positive feed, it may exchange the A, B signals of the handwheel terminals, also you can modify the HNGX, HNGY, HNGZ, HNG4, HNG5 of the bit parameter N°019.

4.2.3 Explanation items

1. The correspondence between the handwheel scale and the machine moving amount is as following table:

<table>
<thead>
<tr>
<th>Handwheel increment</th>
<th>0.001</th>
<th>0.0100</th>
<th>0.100</th>
<th>1.000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specified coordinate value</td>
<td>0.001mm</td>
<td>0.010mm</td>
<td>0.100mm</td>
<td>1.000mm</td>
</tr>
</tbody>
</table>

2. The rotation speed of the handwheel should be less than 5 r/s, if it is over that, the scale may not coincide with the moving amount.

3. The handwheel axis selection key is valid only in the MPG mode.
CHAPTER 5 MDI OPERATION

In MDI mode, the operations of parameter setting, words input and execution can be performed.

**Note!**

The keys functions of this 980MDa machine panel are defined by Ladder; please refer to the respective materials by the machine builder for the function significance.

Please note that the following function introduction is described based on the 980MDa standard PLC programs!

5.1 Code Words Input

Select MDI mode to enter the PRG STATE page, to input an block “G00 X50 Z100”, the steps are as follows:

1. Press **MDI** key to enter MDI mode;

2. Press **PROGRAM** key to enter PRG STATE page:

<table>
<thead>
<tr>
<th>PRG STATE</th>
<th>D0000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>(ABSOLUTE)</td>
<td>(Mode of fixed cycle)</td>
</tr>
<tr>
<td>X 0.000</td>
<td>X V</td>
</tr>
<tr>
<td>Y 0.000</td>
<td>Y W</td>
</tr>
<tr>
<td>Z 0.000</td>
<td>Z P</td>
</tr>
<tr>
<td>INPUT PRG SEGMENT:</td>
<td></td>
</tr>
</tbody>
</table>

| | G00 G17 G90 G54 |
| | G21 G40 G49 G94 G98 |
| | F0100 S 00 M30 |
| PRG. F: | 100 |
| ACT. F: | 0 |
| FED OVRI: | 150%
| RAP OVRI: | 100%
| SPI OVRI: | 100%
| PAKI CNT: | Z |
| CUT TIME: | 0:00:02 |

3. Input 9. 0 0, X 5 0, Y 6 0, Z 1 0 by sequence, the page is as follows:
Chapter 5  MDI Operation

5.2 Code Words Execution

After the words are input, and press , the background color of program segment becomes white, these MDI words are executed after the key is pressed. During the execution, Press and Emergency Stop button may be pressed to terminate the MDI words execution. If key is pressed, the background color of program segment will becomes black, then words can be input again.

Note: The subprogram call command (M98 P ; etc.) is invalid in MDI mode.
5.3 Parameter Setting

In MDI mode, the parameter value can be modified after entering the parameter interface. See details in Chapter 9 of this part.

5.4 Data Modification

In the PRG STATE page, before the inputted words will be executed, if there is an error in inputted words, press **CANCEL** to cancel highlight state, then program segment can be modified. It may press **RESET** key to clear all the words, then input the correct words; for example, "Z1000" will be inputted to replace Z100 in Section 5.1 of this chapter, the steps are as follow.

1. press **CANCEL** key, the page is as follows:

```
PRG STATE
(ABSOLUTE) (Mode of fixed cycle) G00 G17 G90 G54 G21 G40 G49 G94 G98
X 0.000 X
Y 0.000 Y
Z 0.000 Z
P
Q
F0100 S 00 M30
FRG. F: 100
ACT. F: 0
FED OVRI: 150%
RAP OVRI: 100%
SPI OVRI: 100%
PART CNT: 2
CUT TIME: 0:00:02
G00 X50 Y50 Z100 _
```

2. press **CANCEL** key, the page is as follows:
### 5.5 OUT Key Start

When the “OUTR” of the K parameter K0010 is set to 1, the current words inputted
may be executed by pressing key in MDI mode. It is the same as.
CHAPTER 6  PROGRAM EDIT AND MANAGEMENT

In Edit mode, the programs can be created, selected, modified, copied and deleted, and the bidirectional communication between CNC and CNC, or CNC and PC can also be achieved. To prevent the program to be modified or deleted accidentally, a program switch is set for this GSK980MD system. And it must be turned on before program editing. Also 3 level user authority is set in this GSK980MD system to facilitate the management. Only the operation authority is above 4 level (4 or 3 level etc.) can open the program switch for program editing.

6.1 Program Creation

6.1.1 Creation of the block number

The program can be with or without a block No. The program is executed by the block numbered sequence (except the calling). When the “AUTO SEG” switch in setting page is OFF, the CNC doesn’t generate the block number automatically, but the blocks may be edited manually.

When “AUTO SEG” switch in switch setting page is on, the CNC generates the block number automatically. In editing, press \[ \text{key to generate block number of the next block automatically.} \]

The increment of this block number is set by \#216.

6.1.2 Input of the program content

1. Press \[ \text{key to enter the Edit mode;} \]

2. Press \[ \text{key to enter the Program interface, select the PRG CONTENT page} \]
3 Key in address key, numerical key, and key by sequence (e.g. Program O0001 creation):

<table>
<thead>
<tr>
<th>PRG CONTENT</th>
<th>SEG1</th>
<th>COL:1</th>
<th>C:/O0000.CNC</th>
</tr>
</thead>
<tbody>
<tr>
<td>O0000 (O0000);</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G0 G54 G90 X0 Y0 Z0;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X10 Y10;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X-10 Y-10;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M99;</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4 Press key to setup the new program;
5 Input the edited part program one by one, the character will be displayed on the screen immediately as it is input (as for compound key, press this key repeatedly for alternate input), after a block is finished, press \textbf{EOB} to terminate it.

6 Other blocks can be input by step 5 above.

### 6.1.3 Search of the character

1 **Scanning: To scan the character one by one by cursor**

Press \textbf{EXIT} key to enter the Edit mode, then press \textbf{PROGRAM} key to enter the PRG CONTENT page;

1) Press \textbf{↑} key, the cursor shifts a line upward; if the number of the column where the cursor locates is over the total columns of the previous line, the cursor moves to the previous block end (at";"sign) after \textbf{↑} key is pressed;

2) Press \textbf{↓} key, the cursor shifts a line downward; if the number of the column where the cursor locates is over the total columns of the next line, the cursor moves to the next block end (at";"sign) after the \textbf{↓} key is pressed;

3) Press \textbf{→} key, the cursor shifts a column to the right; if the cursor locates at the line end, it moves to the head of the next block;
4) Press the key, the cursor shifts a column to the left; if the cursor locates at the line head, it moves to the end of the next block;

5) Press the key to page upward, the cursor moves to the 1st line and the 1st column of the previous page, if it pages to the head of the program, the cursor moves to the 2nd line and 1st column;

6) Press the key to page downward, the cursor moves to the 1st line and 1st column of the next page, if it pages to the end of the program, the cursor moves to the last line and 1st column of the program;

2 Searching: To search for the specified character upward or downward from the cursor current location

The steps of searching are as follows:

1) Press the key to enter Edit mode;

2) Press the key to enter the PRG CONTENT page;

3) Press the key to enter Search mode, Max. 50 bytes can be input, but only 10 of them can be searched. If the characters are over 10 bytes, searching will fail. E.g. to search command ——G2, press the key, then input G2, and operate as step 4.

<table>
<thead>
<tr>
<th>PRG CONTENT</th>
<th>ITOR SEG8</th>
<th>COL:1</th>
<th>C:/00008.CNC</th>
</tr>
</thead>
<tbody>
<tr>
<td>00008 (CNC PROGRAM);</td>
<td>G40 G49 G30;</td>
<td>G0 G90 G54 X0 Y0 Z0;</td>
<td>Z50;</td>
</tr>
<tr>
<td>G1 X20 Z20 F1500;</td>
<td>G2 I-20;</td>
<td>G3 I-20;</td>
<td>G4 X5;</td>
</tr>
<tr>
<td>G1 X0 Y20 Z0 F1000;</td>
<td>X-20 Y0;</td>
<td>FIND G2</td>
<td>EDIT</td>
</tr>
</tbody>
</table>

S0000 T00 H00
4) Press key (  or  ) by the location relation between the character to be searched and the character where the cursor locates, it displays as follows:

<table>
<thead>
<tr>
<th>PRG CONTENT</th>
<th>ITOR SEG8</th>
<th>COL:1</th>
<th>C:/00008.CNC</th>
</tr>
</thead>
<tbody>
<tr>
<td>00008 (CNC PROGRAM);</td>
<td>G40 G49 G80;</td>
<td>G0 G30 G54 X0 Y0 Z0;</td>
<td>Z50;</td>
</tr>
<tr>
<td>G1 X20 Z20 F1500;</td>
<td>C2 I-20;</td>
<td>C3 I-20;</td>
<td>G4 X5;</td>
</tr>
<tr>
<td>G1 X0 Y20 Z0 F1000;</td>
<td>X-20 Y0;</td>
<td>FIND G2.</td>
<td>EDIT</td>
</tr>
</tbody>
</table>

5) After the searching, the CNC system is still in searching state, press  key or  key again, the next character can be searched. Or press  key to exit the searching state.

6) If the character is not found, the prompt of “Srch fail” will be displayed.

Note: During the searching, it doesn’t search the characters in the called subprogram

3) Method to return to the program head

1) In the Program Display page of the Edit mode, press  key, the cursor returns to the program head

2) Search the program head character by the methods in Section 6.1.3 of this part.

6.1.4 Insertion of the character

Steps:

1) Select the PRG CONTENT page in Edit mode, the page is as follows:
2) Input the character to be inserted (to insert G98 code before G2 in the above figure, input \[ \text{insertion symbol} \], the page is as follows:

<table>
<thead>
<tr>
<th>PRG CONTENT</th>
<th>SEG5</th>
<th>COL:5</th>
<th>C:/00008.CNC</th>
</tr>
</thead>
<tbody>
<tr>
<td>00008 (CNC PROGRAM); G40 G49 G30; G0 G90 G54 X0 Y0 Z0; Z50; G1 X20 Z20 F1500; G2 I-20; G3 I-20; G4 X5; G1 X0 Y20 Z0 F1000; X-20 Y0;</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

EDIT \[ \text{insertion symbol} \] S0000 T00 H00

Note 1: In the Insert mode, if the cursor is not located at the line head, a space will be automatically generated when inserting the command address; if the cursor is located at the line head, the space will not be generated, and it should be inserted manually.

Note 2: In program content edit mode or MDI mode of program state page, press \[ \text{macro edit} \] key to enter insertion or macro edit state.

In macro editing mode, special symbols can be input are: \[ [ ] '; '=' '+' '>' '<' '/' '& ' '|' \].

Above symbols are frequently used for macro edit.
### Difference between two states

<table>
<thead>
<tr>
<th>Insertion state</th>
<th>Automatic space</th>
<th>Process of character ‘O’</th>
<th>Input special symbols</th>
</tr>
</thead>
<tbody>
<tr>
<td>In program editing, insert blank automatically to separate words.</td>
<td>Program switch, duplication and deletion can be done by pressing ‘O’.</td>
<td>Special symbols can not be inputted.</td>
<td></td>
</tr>
<tr>
<td>Macro edit state</td>
<td>Blank can not be inserted automatically.</td>
<td>Only input character ‘O’.</td>
<td>Special symbols can be inputted.</td>
</tr>
</tbody>
</table>

#### 6.1.5 Deletion of the character

Steps:

1) Select the PRG CONTENT page in Edit mode;

2) Press **CANCEL** key to delete the character before the cursor; press **DELETE** key to delete the character where the cursor locates.

#### 6.1.6 Modification of the character

Cancel or delete the character and re-enter new ones.

#### 6.1.7 Deletion of a single block

This function is only applied to the block with a block No.(N command), which is at the head of a line and followed by blocks which are divided by space.

Steps:

1) Select the PRG CONTENT page in Edit mode;

2) Move the cursor to the head of the block to be deleted (column 1— where N locates), then press **DELETE** key.

Note: If the block has no block No.N, key in “N” at the head of the block, and move the cursor to “N”, then press **DELETE** key.

#### 6.1.8 Deletion of the blocks

It deletes all the content (including the specified block) from the current character where the cursor locates to the block with the specified No.(searching downward), and the
specified block must have a block No..

<table>
<thead>
<tr>
<th>PRG CONTENT</th>
<th>SEG5</th>
<th>COL:5</th>
<th>C:/00008.CNC</th>
</tr>
</thead>
<tbody>
<tr>
<td>00008 (CNC PROGRAM);</td>
<td>G40 G49 G30;</td>
<td>G0 G90 G54 X0 Y0 Z0;</td>
<td>Z50;</td>
</tr>
</tbody>
</table>

**Steps**

1) Select the PRG CONTENT page in Edit mode;

2) Press key to enter the FIND state, and key in the block No.

3) Press key to delete blocks from G0 (block 2) to N10 (including block N10). It displays as follows:
6.1.9 Segment deletion

It deletes the content downward from the current character where the cursor locates to the word specified.

```
PRG CONTENT       SEG2  COL:4  C:/00008.CNC
00008 (CNC PROGRAM);
G40 G49 G30;
G3 I-20;
G4 X5;
G1 X0 Y20 Z0 F1000;
X-20 Y0;
X0 Y-20 Z-10;
X20 Y0 Z-20;
X5 Y5 Z-50;
M99;
EDIT             S0000 T00 H00
```

Steps

1) Select the PRG CONTENT page in Edit mode

2) Press key to enter the FIND state, and key in the characters (see the following figure: input F1000)

```
PRG CONTENT       SEG2  COL:4  C:/00008.CNC
00008 (CNC PROGRAM);
G40 G49 G30;
G3 I-20;
G4 X5;
G1 X0 Y20 Z0 F1000;
X-20 Y0;
X0 Y-20 Z-10;
X20 Y0 Z-20;
X5 Y5 Z-50;
M99;
FIND F1000;
EDIT             S0000 T00 H00
```

3) Press key, and all programs from I-20 where the cursor locates to F1000. It displays as follows:
6.2 Program annotation

To facilitate the user to search, manage and edit program, the system provides program name annotation and block annotation functions.

6.2.1 Annotation for program name

The program annotation can be added in the brackets behind it. For example: program O0005 is used for machining bolt holes, the annotation can be added in program contents as follows:

1) Select edit mode, and then enter program content display page.

2) Press key, search is displayed at the left bottom of the screen, the displayed figure is as follows:
3) Input annotation behind search (input max. 50 characters except for brackets). If BOLT PROC is inputted (bolt holes machining), the page displayed is as follows:

```
PRG CONTENT  SEG1  COL:1  C:/00005.CNC
D0005 (00005);
G90 G00 X0 Y0 Z0;
(I:cir r,A:first hole angle,B:angle inc,H:hole number);
G65 P9020 X100 Y50 R30 Z-50 F1800 I100 A45 B30 H5;
M30;
%
```

```
FIND BOLT PROC
EDIT  S0000 T00 H00
```

4) Press key. program annotation setting up is finished. the displayed page is as follows:

```
PRG CONTENT  SEG1  COL:1  C:/00005.CNC
D0005 (BOLT PROC);
G90 G00 X0 Y0 Z0;
(I:cir r,A:first hole angle,B:angle inc,H:hole number);
G65 P9020 X100 Y50 R30 Z-50 F1800 I100 A45 B30 H5;
M30;
%
```

```
EDIT  S0000 T00 H00
```
6.2.2 Block annotation

Take contents in brackets ‘˄’ and ‘˅’ as program annotation, which can be put at any position of a block and displayed with green characters. The page is as follows:

```
PRG CONTENT    SEG1    COL:1    C:/00005.CNC

00005 (BOLT PROC());
G90 G00 X0 Y0 Z0;
(I: cir r, A: first hole angle, B: angle inc, H: hole number);
G65 P9020 X100 Y50 Z-50 F1800 I100 A45 B30 H5;
G04 X3 (pause 3 sec.);
M30;
%
```

Related explanations:

1) Because symbols ‘˄’ and ‘˅’ are not provided in the system, block annotation cannot be inputted by edit mode in the system. If block annotation is needed to added, edit annotation on the PC and download it to the CNC by software.

2) The system is not support Chinese characters. If Chinese characters are edited on PC, which will be displayed as blanks in the system after it is saved in the CNC.

Note 1: After a program is set up, if the program name annotation is not added, CNC defaults program name as program name annotation

Note 2: Program annotation in the CNC must be English, but the CNC supports Chinese annotation display (except for Chinese decimal points). The way of adding Chinese annotation is as follows: Edit Chinese annotation in the PC machine, and then download it to the CNC by communication software.

6.2.3 Alter program annotation

Operation steps are the same as program annotation setting steps on section 6.2.1 of this chapter.

6.3 Deletion of the Program

6.3.1 Deletion a single program

Steps:

1) Select the PRG DISPLAY page in Edit mode;
2) Key in address key ①, numerical key ② ③ ④ by sequence (take program O0001 for an example);
3) Press key, program O0001 will be deleted

Note: Press ‘DELETE’ key in page ‘program preview’ or ‘file list’ to delete program.

### 6.3.2 Deletion of all programs

**Steps**
1) Select the PRG DISPLAY page in Edit mode
2) Key in address key ①, symbol key ② numerical key ③ ④ by sequence
3) Press ⑤ key, all the programs will be deleted.

Note: Press ‘delete key’ in page ‘file list’ to delete all programs.

### 6.4 Selection of the Program

When there are multiple programs in CNC system, a program can be selected by the following 4 methods:

#### 6.4.1 Search method

1) Select Edit mode;
2) Press ① key to enter the PRG CONTENT page;
3) Press address key ② and key in the program No.;
4) Press ③ or ④ key, the searched program will be displayed.

Note: In step 4, if the program does not exist, a new program will be created by CNC system after ⑤ key is pressed.
6.4.2 Scanning method

1) Select Edit or Auto mode;

2) Press the PRG DISPLAY page;

3) Press address key

4) Press or key to display the next or previous program;

5) Repeat step 3 and 4 to display the saved programs one by one.

6.4.3 Cursor method

1) In Program Preview mode (must be in non-running state);

2) Press or key to move the cursor to the program name to be selected (change “PRG SIZE”, “NOTE” content as the cursor moves);

3) Press to open the program.

6.4.4 Select file by using file list

1) On file list page (Edit mode is operation mode)
### 6.5 Execution of the Program

After the program to be executed is selected by the method in Section 6.4 of this part, select the Auto mode, then press key (or press external cycle start key), the program will be executed automatically.

### 6.6 Rename of the Program

1) Select the PRG CONTENT page in Edit mode;

2) Press address key and key in the new program name;

3) Press key.

Note: No matter whether the program is altered or not, program annotation is changed into new program name automatically after program is renamed.

### 6.7 Copy of the Program

To save the current program to a location:

1) Select the PRG CONTENT page in Edit mode;

2) Press address key and key in the new program No.
6.8 Program positioning

- To the position where the program stops last time by TO
  Search for the point where the program execution stops by TO. Select edit mode to enter program content page and press conversion key, input TO to search which is displayed at the left bottom. Then press up or down key, searching and positioning are displayed at this time, the cursor will move to the position where program stops last time.

- Position to specified block by TO+num (num is the block number specified by user. For example: TO10000 means position to the 10000th block)
  On program content page, locate to specified block by inputing TO block number. Press conversion key after entering program content page, input TO to search which is displayed at the left bottom and then press up or down key, the cursor will move to the specified program.

6.9 Program preview

In non-edit mode, press key to enter program preview page. In this page, program names saved in CNC are displayed in the form of list. Max. 36 program names can be displayed in one page, if programs saved are over 36, press key to display programs in other page.

<table>
<thead>
<tr>
<th>PRG PREVIEW</th>
<th>O0214 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000 00001 00005 00005 00008 00020 00125 00214 00254 01212 01234 02036 02539 03654</td>
<td></td>
</tr>
<tr>
<td>MEM SIZE: 40.0MB</td>
<td></td>
</tr>
<tr>
<td>MEM USED: 222KB</td>
<td></td>
</tr>
<tr>
<td>PRG AMOT: 14</td>
<td></td>
</tr>
<tr>
<td>PRG SIZE: 117B</td>
<td></td>
</tr>
<tr>
<td>00003 (00003); G0 G90 X0 Y0 Z0; G1 X50 Y50; X100 Y0; X50 Y-50;</td>
<td></td>
</tr>
<tr>
<td>EDIT</td>
<td>S0000 T00 H00</td>
</tr>
</tbody>
</table>

- Program capacity display:
  On top right window, “storage capacity”displays the max. capacity of program which can be saved in CNC. “Used capacity”displays the capacity of saved program in CNC system. “Program
number" displays the program number saved in the CNC system. "Program size" displays the size of the currently opened program.

- Program preview selection:
  On top left of the window, the name of currently previewed program will be displayed in blue characters on white ground. Program size on top left window is the size of currently previewed program. The following window displays currently previewed program, display 5-line program.

- Usage of cursor key and conversion key:
  When select program in a program list, select the program to be previewed by cursor moving key on MDI panel. If the size is very big, max. 36 program names can be displayed in program list. Select program by pressing right moving key or pressing conversion key directly, turn pages to display the program list, and then select it by cursor moving key on MDI panel.

- Open a program:
  In edit, auto, MDI modes, when open the program on program preview window, this program can be opened by pressing EOB key on MDI panel. At the same time, the name of currently opened program is displayed on top right page.

- Deletion of program
  Move cursor to the program will be deleted, press delete key and then press Y key or N key on multiple select manue to select wether delete it or not
CHAPTER 7 AUTO OPERATION

Note!
The keys functions of this 980MDa machine panel are defined by Ladder; please refer to the respective materials by the machine builder for the function significance.

Please note that the following function introduction is described based on the...

7.1 Auto Run

7.1.1 Selection of the program to be run

1. Search method
   1) Select the Edit or Auto mode;

   2) Press key to enter the PRG CONTENT page;

   3) Press the address key and key in the program No.

   4) Press or key, the program retrieved will be shown on the screen, if the program doesn’t exist an alarm will be issued

   Note In step 4, if the program to be retrieved does not exist, a new program will be setup by CNC system after pressing key.

2 Scanning method

1) Select the Edit or Auto mode

   2) Press key to enter the PRG display page

   3) Press the address key

   4) Press the or key to display the next or previous program;

   5) Repeat the step 3, 4 above to display the saved program one by one.
3. Cursor method
   a) Select the Auto mode (must in non-run state)

   b) Press \textit{PROGRAM} key to enter the PRG LIST page;

   c) Press \textit{ } , \textit{ } , \textit{ } , \textit{ } key to move the cursor to the name of the program to be selected;

   d) Press \textit{ } key.

4. File open method
   Select the edit or operation mode:

   1) Press \textit{PROGRAM} key twice to enter the page of file list:

   2) Press \textit{ } , \textit{ } keys to move the cursor to the file will be selected.

   3) Press \textit{DATA INPUT} key to select a file.

   4) Press \textit{ } key to open the selected file.

   Note: The file can not be opened if the expanded name is not ".CNC".

7.1.2 Program start

1. Press \textit{AUTO} key to select the Auto mode

2. Press \textit{CYCLE START} key to start the program, and the program execution begins

   Note Since the program execution begins from the block where the cursor locates, before pressing the \textit{CYCLE START} key, make a check whether the cursor is located at the block to be executed. If begins from the start line, but the cursor is not in this line, move the cursor to the line.

7.1.3 Stop of the auto run
• **Stop by command (M00)**

  the block containing M00 is executed, the auto run is stopped. So the modal function and state are all reserved. Press the key or the external Run key, the program execution continues.

• **Stop by a relevant key**

  1. In Auto run, by pressing key or external dwell key, the machine remains at the following state:
     - (1) The machine feed decelerate to stop;
     - (2) During the execution of the dwell command (G04), it pauses after G04 command execution is finished.
     - (3) The modal function and state are saved;
     - (4) The program execution continues after pressing the key

  2. **Stop by Reset key**

     - (1) All axes movement is stopped.
     - (2) M, S function output is invalid (the automatic cut-off of signals such as spindle CCW/CW, lubrication, cooling by pressing key can be set by the parameters)
     - (3) Modal function and state is held on after the auto run.

  3. **Stop by Emergency stop button**

     If the external emergency button (external emergency signal valid) is pressed under the dangerous or emergent situation during the machine running, the CNC system enters into emergency state, and the machine moving is stopped immediately, all the output (such as spindle rotation, coolant) are cut off. If the Emergency button is released, the alarm is cancelled and CNC system enters into reset mode.

  4. **By Mode switching**

     When the Auto mode is switched to the Machine zero, MPG/Step, the current block "dwell" immediately; when the Auto mode is switched to the Edit, MDI mode, the “dwell” is not displayed till the current block is executed.

**Note 1**  
Ensure that the fault has been resolved before cancelling the emergency alarm.
Note 2  The electric shock to the device may be decreased by pressing the Emergency button before power on and off.

Note 3  The Machine zero return operation should be performed again after the emergency alarm is cancelled to ensure the coordinate correctness (but this operation is unallowed if there is no machine zero in the machine)

Note 4  Only the BIT3 (ESP) of the bit parameter No.017 is set to 0, could the external emergency stop be valid.

7.1.4 Auto run from an arbitrary block

1. Press key to enter the Edit mode, press key to enter the Program interface, or press key several times to select the PRG CONTENT page:

2. Move the cursor to the block to be executed (for example, move the cursor to the 3th line head if it executes from the 3th line):

```
PRG CONTENT  SEG3  COL:1  C:\00000.CNC
00000 (00000);
G0 G54 G90 X0 Y0 Z0 G49;
G01 X100 Y100 F500;
G02 I20;
G01 X52 Z01;
G91 X2 Z-6.3;
G00 X0 Y0 Z0;
M30;
%
```

3. If the mode (G, M, T, F command) of the current block where the cursor locates is defaulted and inconsistent with the running mode of this block, the corresponding modal function should be executed to continue the next step.

4. Press key to enter the Auto mode, then press key to start the program.
7.1.5 Adjustment of the feedrate override, rapid override

In Auto mode, the running speed can be altered by adjusting the feedrate override, rapid override with no need to change the settings of the program and parameter.

- Adjustment of the feedrate override

Press the or key in , it can realize 16-level real time feedrate adjustment.

Press the key each time, the feedrate override ascends a gear level till 150%.

Press the key each time, the feedrate override decends a gear level till 0;

Note 1 The actual feedrate value is specified by F in program feedrate override adjustment;

Note 2 Actual feedrate= value specified by F× feedrate override

- Adjustment of rapid override

It can realize the 4-level real time rapid override F0, 25%, 50%, 100% adjustment by pressing the

Press the key each time, the rapid override ascends a level till 100%;

Press the key each time, the rapid override decends a level till F0

Note 1 The max. rapid traverse speeds of X, Y, Z axis are set by bit parameter No.059, No.060, No.061 respectively;

X axis actual rapid traverse rate = value set by parameter No.059×rapid override

Y axis actual rapid traverse rate = value set by parameter No.060×rapid override
Z axis actual rapid traverse rate = value set by parameter No.061×rapid override

Note 2 When the rapid override is F0, the rapid traverse rate is set by bit parameter No.069.

7.1.6 Spindle override adjustment

While the spindle speed is controlled by the analog voltage output in Auto mode, it can be adjusted by spindle override.

Press the up or down key in to adjust the spindle override for the spindle speed, it can realize 8-level real-time override adjustment between 50%～120%.

Press the up key each time, the feedrate override ascends a level till 120%

Press the down key each time, the rapid override decends a level till 50%.

Note 1 The actual output analog voltage=analog voltage by parameter×spindle override

Example: When the bit parameter No.101 is set to 9999, No.100 to 645, execute S9999 command to select the spindle override 70%, the actual output analog voltage=10×70%=7V

7.2 DNC running

This CNC system has a DNC function, by the connection of the DNC communication software with this system, the high speed, high capacity program can be performed in this system.

In Auto mode, press the key, it enters the DNC mode. Then press the key to start the program DNC machining under the condition that the PC is get ready Please refer to the DNC communication software for details.

7.3 Running state

7.3.1 Single block execution

When the program is to be executed for the 1st time, to avoid the programming errors, it may select Single block mode to execute the program.

In Auto mode, the methods for turning on single are as follows.
Press the \textbf{SINGLE} key to make the single block indicator in State area to light up, it means that the single block function has been selected.

In Single block mode, when the current block execution is finished, the CNC system stops; if next block is to be executed, it needs to press the \textbf{CYCLE START} key.

\textbf{Note} Even at the mid point, the single block stops in G28, G29, G30 commands.

### 7.3.2 Dry run

Before the program is to be executed, in order to avoid the programming errors, it may select the Dry run mode to check the program. And the machine runs by a constant speed other than the speed specified by the program.

In Auto mode, the method for turning on the Dry run switch are as follows.

Press the \textbf{DRY} key to make the dry run indicator in State area to light up, it means that the dry run function is selected.

The speed specified by the program is invalid in Dry run, and actual feedrate is set by the DATA parameter No.174.

### 7.3.3 Machine lock

In Auto mode, the ways to make machine lock function valid are as follows.

Press the \textbf{MACHINE LOCK} key to make the machine lock indicator in State area to light up, it means that it has entered the machine lock state.

While in the machine lock mode:

1. The machine carriage doesn't move, the “MACHINE” in the INTEGRATED POS page of the POSITION interface doesn’t vary too. The RELATIVE POS and ABSOLUTE POS, DIST TO GO are refreshed normally.
2. M, S, T commands can be executed normally.

### 7.3.4 MST lock

In Auto mode, the ways to make MST lock function valid are as follows.

Press the \textbf{M.S.T. LOCK} key to make the MST lock indicator in State area to light up, it means that it has entered the MST lock state. And the carriage move is not performed by M, S, T.
commands

Note: When the MST lock is valid, it has no effect on the execution of M00, M30, M98, M99.

7.3.5 Block skip

If a block in program is not needed to be executed and not to be deleted, this block skip function can be used. When the block is headed with “/” sign and Block skip function is valid, this block is skipped without execution in Auto mode.

In Auto mode, the way to make block skip function valid is as follows.

Press the key to make the block skip indicator in State area to light up, it means that the block skip function is valid.

Note While the block skip function is invalid, the blocks headed with “/” signs are executed normally in Auto mode.

7.3.6 Optional stop

In AUTO mode, the valid optional stop function is as follows:

Press key to enter optional stop and the indicator lights up.

The program will be “stopped” at command M01. Press key again to continue program execution.

7.4 Memorizing at power-down

7.4.1 Program interruption in non-DNC auto operation

Operation method 1 (Manual)

1. After power on, press conversion key + press letter “T” + letter “O” + up, down moving keys on pages “program content, edit” to the block where the execution stops last time.
2. Switch to the pages “coordinate & program, machine zero”.
3. Enter the next step after machine zero is performed.
4. Switch to manual or MDI mode. Locate to the block where it stops last time. (At this moment, it is necessary to confirm whether it is at state G40, G49, G54. Ensure that tools are in a safe range during positioning.)
5. Switch to manual mode, press conversion key. It prompts “Locate to the block where it stops last time. It will recover the mode before power-down (Y/N)”.
6. Press Y to recover the mode before power-down.
7. Switch to auto mode, press cycle start key to execute the block continuously from where it stops last time.
Operation method 2 (Auto)

1. After power on, press conversion key \( \rightarrow \) press letter “T”+“O” \( \rightarrow \) up, down moving keys on pages “program content, edit” to the block where the execution stops last time.

2. Switch to the pages “coordinate & program, machine zero”.

3. Perform machine zero operation.

4. After machine zero is performed, press conversion key. It prompts at the bottom of the screen: “Locate to the block automatically where it stops last time. It will recover the mode before power-down (Y/N)”. Input Y (Ensure that tools moving path is in a safe range at this moment.). Coordinates start move, it locates to the block where it stops last time, and recovers the mode before power-down.

5. Switch to auto mode, press cycle start key to execute the block continuously where it stops last time.

7.4.2 Interruption at power-down on DNC auto operation

Operation method (Auto)

1. Switch to “coordinate program, machine zero return” after power on.

2. Execute machine zero return.

3. After machine zero return is finished, press conversion key. It prompts at the bottom of the screen: “Locate to the block automatically where it stops last time. It will recover the mode before power-down (Y/N)”. Input Y (Make sure tools moving path is in a safe range at this moment.). Coordinates start move, it locates to the block where it stops last time, and recovers the mode before power-down.

4. Switch to the highlighted block when DNC, CNC power down.

5. Search for the interrupted block in DNC transmission software, then press RESET key on panel to continue PC software transmission. Press cycle start key to continue execution.
CHAPTER 8  MACHINE ZERO RETURN OPERATION

8.1 Machine Zero

The machine coordinate system is a basic coordinate system for CNC coordinate calculation. It is an inherent coordinate system of the machine. The origin of the machine coordinate system is called machine zero (or mechanical reference point). It is defined by the zero return switches fixed on the machine. Usually the switch is fixed on the positive max. Strokes of X, Y, Z axes.

8.2 Machine Zero Return Steps

1. Press key, it enters the Machine zero mode, the bottom line of the screen page shows “REF”, the figure is as follows:

<table>
<thead>
<tr>
<th>RELATIVE POS</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>O0000 N00000</td>
<td>G00 G17 G90 G54 G21 G40 G49 G94 G98</td>
</tr>
<tr>
<td>X 0.000</td>
<td>F0100 S00 M30</td>
</tr>
<tr>
<td>Y 0.000</td>
<td>JOG. F: 1260</td>
</tr>
<tr>
<td>Z 0.000</td>
<td>ACT. F: 0</td>
</tr>
<tr>
<td></td>
<td>FED OVRI: 150%</td>
</tr>
<tr>
<td></td>
<td>RAP OVRI: 100%</td>
</tr>
<tr>
<td></td>
<td>SPI OVRI: 100%</td>
</tr>
<tr>
<td></td>
<td>PART CNT: 0</td>
</tr>
<tr>
<td></td>
<td>CUT TIME: 0:00:00</td>
</tr>
</tbody>
</table>

2. Press or or key to select the machine zero of X, Y or Z axis

3. The machine moves along the machine zero direction, and returns to the machine zero via the deceleration signal, zero signal detection. And the axis stops with the machine zero finish indicator lighting up.

   Machine zero finish indicators

Note1: If the machine zero is not fixed on the machine, machine zero operation B/C/D is unallowed.

Note2: While the coordinate is moved out from the machine zero, the machine zero finish indicators go out.

Note3: After the machine zero operation, the cancellation of the tool length offset for the
CNC is set by the BIT7 of the bit parameter No.22

Note 4: See details in the 3rd part INSTALLATION AND CONNECTION for the parameters concerning with the machine zero.

Note 5: When machine zero return, bit parameter №011 ZNIK determines whether axis movement is locked automatically.

Note 6: Only machine zero D mode can be used for rotary axis.
CHAPTER 9 DATA SETTING, BACKUP and RESTORE

9.1 Data Setting

9.1.1 Switch setting

In SWITCH SETTING page, the ON-OFF state of PARM SWT (parameter switch), PROG SWT (program switch), AUTO SEG (auto sequence No.) can be displayed and set, the figure is as follows:

<table>
<thead>
<tr>
<th>SWITCH SETTING</th>
<th>00000 N0000</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{PARM SWT: } \ast \text{OFF ON} )</td>
<td></td>
</tr>
<tr>
<td>( \text{PROG SWT: OFF } \ast \text{ON} )</td>
<td></td>
</tr>
<tr>
<td>( \text{AUTO SEG: } \ast \text{OFF ON} )</td>
<td></td>
</tr>
</tbody>
</table>

1. Press key to enter the Setting interface, then press or key to enter SWITCH SETTING page.
2. Press or key to move the cursor to the item to be set.
3. Press , and key to shift the ON-OFF state, press or key, \( \ast \) moves to the left to set the switch for OFF, Press or key, \( \ast \) moves to the right to set the switch for ON. Only the PARM SWT is set to ON, could the parameter be altered; so are PROG SWT and AUTO SEG.

Note 1: When parameter switch is shifted from “off” to “on” for the first time, CNC alarm occurs. Press keys together to eliminate the alarm. Alarm will not occur when parameter switch is shifted again. For security, set parameter switch to “off” after parameter alteration is finished.

Note 2: When parameter switch is shifted from “off” to “on”, CNC alarm occurs. Alarm will occur again when parameter switch is shifted from “on” to “off” for the first time. Press keys together to eliminate the alarm.

9.1.2 Graphic setting
Press key to enter graphic interface. Press or key to access the following graphic parameter page.

A: The way of setting graphic parameter

1. In MDI mode, press or key to move the cursor to the parameter to be set,
2. Input corresponding values,
3. Press key, and the setting is finished.

B: Significance of graphic parameter

Coordinate selection: Display view angle of the graphic path can be selected by setting different values. Corresponding coordinate for 0~7 is as follows.

Scaling: Display the scaling of current graphic path.

Graphic center: Display the center of each axis.

Maximum, minimum: Set the maximum and minimum scope can be displayed by each axis.

C: Graphic track operation

Graphic track is as follows:

```
<table>
<thead>
<tr>
<th>GRAPH SET</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>COOR OPT=</td>
<td>0 0XY 1YX 22X 3XYZ 4YZ 5ZY 6XZ 7XZY)</td>
</tr>
<tr>
<td>SCALE =</td>
<td>100%</td>
</tr>
<tr>
<td>CENTER =</td>
<td>0.000 (X axis value)</td>
</tr>
<tr>
<td>CENTER =</td>
<td>0.000 (Y axis value)</td>
</tr>
<tr>
<td>CENTER =</td>
<td>0.000 (Z axis value)</td>
</tr>
<tr>
<td>X MAX. =</td>
<td>120.000</td>
</tr>
<tr>
<td>Y MAX. =</td>
<td>120.000</td>
</tr>
<tr>
<td>Z MAX. =</td>
<td>120.000</td>
</tr>
<tr>
<td>X MIN. =</td>
<td>-120.000</td>
</tr>
<tr>
<td>Y MIN. =</td>
<td>-120.000</td>
</tr>
<tr>
<td>Z MIN. =</td>
<td>-120.000</td>
</tr>
</tbody>
</table>
```
Vertical move: Display upper and lower part of the graphic.
Horizontal move: Display right and left part of the graphic.
Scaling: Display scaling of current graphic.
Absolute coordinate: Display the absolute coordinate of the program.
S: Start drawing, S is highlighted by pressing S key. Display drawing track.
T: Stop drawing, T is highlighted by pressing S key. It stops drawing.
R: Clear graphic track, clear graphic track displayed before.
K: Switch view angle, coordinate value can be switched between 0~7 by pressing K key each time.
J: Display graphic in the center, that is, vertical move and horizontal move are 0.
I: Scale up the track, the graphic is scaled up 2 fold by pressing I key once.
M: Scale down the track, the graphic is scaled down 2 fold by pressing M key once.

9.1.3 Parameter setting

By the parameter setting, the characteristics of the drive unit and machine can be adjusted. See Appendix 1 for their significance
Press \[ \text{PARAMETER} \] key to enter the Parameter interface, then press \[ \text{or} \] key to switch the parameter page, the figure is as follows:

### BIT PARAMETER

<table>
<thead>
<tr>
<th>NO.</th>
<th>DATA</th>
<th>NO.</th>
<th>DATA</th>
<th>NO.</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>00000000</td>
<td>009</td>
<td>00011111</td>
<td>017</td>
<td>00101000</td>
</tr>
<tr>
<td>002</td>
<td>00000010</td>
<td>010</td>
<td>00011111</td>
<td>018</td>
<td>00000000</td>
</tr>
<tr>
<td>003</td>
<td>00000000</td>
<td>011</td>
<td>00000000</td>
<td>019</td>
<td>10000000</td>
</tr>
<tr>
<td>004</td>
<td>01000000</td>
<td>012</td>
<td>00010011</td>
<td>020</td>
<td>00000000</td>
</tr>
<tr>
<td>005</td>
<td>00010001</td>
<td>013</td>
<td>10000011</td>
<td>021</td>
<td>00000000</td>
</tr>
<tr>
<td>006</td>
<td>00000000</td>
<td>014</td>
<td>00011111</td>
<td>022</td>
<td>00000000</td>
</tr>
<tr>
<td>007</td>
<td>00000000</td>
<td>015</td>
<td>10000000</td>
<td>023</td>
<td>00000000</td>
</tr>
<tr>
<td>008</td>
<td>00011111</td>
<td>016</td>
<td>00000000</td>
<td>024</td>
<td>00000000</td>
</tr>
</tbody>
</table>

### Example:

Set the BIT5 (DECI) of the bit parameter No.004 to 1, and the other bits unchanged.

Move the cursor to No.004, key in 01100000 by sequence in the prompt line, the figure is as follows:
Chapter 9  Data Setting , Backup And Restore

Press key to finish the parameter alteration. The page is as follows:

<table>
<thead>
<tr>
<th>NO.</th>
<th>DATA</th>
<th>NO.</th>
<th>DATA</th>
<th>NO.</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>00000000</td>
<td>009</td>
<td>00011111</td>
<td>017</td>
<td>00101000</td>
</tr>
<tr>
<td>002</td>
<td>00000010</td>
<td>010</td>
<td>00011111</td>
<td>018</td>
<td>00000000</td>
</tr>
<tr>
<td>003</td>
<td>00000000</td>
<td>011</td>
<td>00000000</td>
<td>019</td>
<td>10000000</td>
</tr>
<tr>
<td>004</td>
<td>00100000</td>
<td>012</td>
<td>00010011</td>
<td>020</td>
<td>00000000</td>
</tr>
<tr>
<td>005</td>
<td>00010001</td>
<td>013</td>
<td>10000011</td>
<td>021</td>
<td>00000000</td>
</tr>
<tr>
<td>006</td>
<td>00000000</td>
<td>014</td>
<td>00011111</td>
<td>022</td>
<td>00000000</td>
</tr>
<tr>
<td>007</td>
<td>00000000</td>
<td>015</td>
<td>10000000</td>
<td>023</td>
<td>00000000</td>
</tr>
<tr>
<td>008</td>
<td>00011111</td>
<td>016</td>
<td>00000000</td>
<td>024</td>
<td>00000000</td>
</tr>
</tbody>
</table>

*** RDRN DECI *** PROD *** *** SCW
bit5:1/0:DEC signal is low/high level
NO. 004 = 61100000

2 Bit alteration

1) Turn on the parameter switch

2) Enter the MDI mode

3) Move the cursor to the No. of the parameter to be set

Method 1: Press or key to enter the page of the parameter to be set,

press or key to move the cursor to the No. of the parameter to be set

Method 2: Press address key key in parameter No., then press key

4) Press and hold key for 2 seconds or press key to skip to a bit of the parameter, and the bit is backlighted. Press or key to move the cursor to the bit to be altered, then key in 0 or 1

5) After all parameters setting is finished, the PARM SWT needs to be set for OFF for security
Note: After entering a bit of the parameter, press and hold key for 2 seconds or press key, it may skip out of the bit and back to the parameter No.

Example:
Set the BIT5 (DECI) of the bit parameter No.004 to 1, and the other bits unchanged. Move the cursor to “No.004” by the steps above, press and hold key for 2 seconds or press key to skip to a bit of the parameter, the figure is as follows:

<table>
<thead>
<tr>
<th>BIT PARAMETER</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO.</td>
<td>DATA</td>
</tr>
<tr>
<td>001</td>
<td>00000000</td>
</tr>
<tr>
<td>002</td>
<td>00000010</td>
</tr>
<tr>
<td>003</td>
<td>00000000</td>
</tr>
<tr>
<td>004</td>
<td>11000000</td>
</tr>
<tr>
<td>005</td>
<td>00010001</td>
</tr>
<tr>
<td>006</td>
<td>00000000</td>
</tr>
<tr>
<td>007</td>
<td>00000000</td>
</tr>
<tr>
<td>008</td>
<td>00011111</td>
</tr>
</tbody>
</table>

*** RDRN DECI *** PROD *** *** SCW
bit7:1/0:Unuscd
NO. 004

MDI  S0000 T00 H00

Move the cursor to “BIT5” by pressing or key, the figure is as follows:

<table>
<thead>
<tr>
<th>BIT PARAMETER</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO.</td>
<td>DATA</td>
</tr>
<tr>
<td>001</td>
<td>00000000</td>
</tr>
<tr>
<td>002</td>
<td>00000010</td>
</tr>
<tr>
<td>003</td>
<td>00000000</td>
</tr>
<tr>
<td>004</td>
<td>01000000</td>
</tr>
<tr>
<td>005</td>
<td>00010001</td>
</tr>
<tr>
<td>006</td>
<td>00000000</td>
</tr>
<tr>
<td>007</td>
<td>00000000</td>
</tr>
<tr>
<td>008</td>
<td>00011111</td>
</tr>
</tbody>
</table>

*** RDRN DECI *** PROD *** *** SCW
bit5:1/0:DEC signal is low/high level
NO. 004

MDI  S0000 T00 H00

Key in “1” to finish the alteration
Chapter 9  Data Setting , Backup And Restore

**B** Alteration of the data parameter, pitch data

1. Data parameter alteration
   1) Turn on the parameter switch;
   2) Enter the MDI mode
   3) Move the cursor to the No. of the parameter to be set
   4) Key in the new parameter value
   5) Press \[ \text{key} \] key, the value is entered and displayed
   6) After all parameters setting is finished, the PARM SWT needs to be set to OFF for security

Example 1: Set the data parameter \( \#059 \) to 4000.

Move the cursor to \( \#059 \) by the steps above, key in \( 4000 \) by sequence in the prompt line, the figure is as follows:

```
DATA PARAMETER

<table>
<thead>
<tr>
<th>NO.</th>
<th>DATA</th>
<th>NO.</th>
<th>DATA</th>
<th>NO.</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>049</td>
<td>1</td>
<td>057</td>
<td>1</td>
<td>065</td>
<td>100</td>
</tr>
<tr>
<td>050</td>
<td>1</td>
<td>058</td>
<td>1</td>
<td>066</td>
<td>100</td>
</tr>
<tr>
<td>051</td>
<td>1</td>
<td>059</td>
<td>7500</td>
<td>067</td>
<td>100</td>
</tr>
<tr>
<td>052</td>
<td>1</td>
<td>060</td>
<td>7500</td>
<td>068</td>
<td>100</td>
</tr>
<tr>
<td>053</td>
<td>1</td>
<td>061</td>
<td>7500</td>
<td>069</td>
<td>400</td>
</tr>
<tr>
<td>054</td>
<td>1</td>
<td>062</td>
<td>7500</td>
<td>070</td>
<td>8000</td>
</tr>
<tr>
<td>055</td>
<td>1</td>
<td>063</td>
<td>7500</td>
<td>071</td>
<td>50</td>
</tr>
<tr>
<td>056</td>
<td>1</td>
<td>064</td>
<td>100</td>
<td>072</td>
<td>100</td>
</tr>
</tbody>
</table>
```

Max. speed of rapid locating in X(mm/min)

| NO. | 059 | 4000 |

Press \[ \text{key} \] key to finish the alteration. The page is as follows
Example 2: Set the X axis value of the pitch data No.000 to 12, set the value of Z axis to 30

Move the cursor to pitch data No.000 by the steps above, key in “X12” by sequence in the cue line, the figure is as follows:

```
000 X 12
```

Press key to finish the alteration. The page is as follows:

```
000 12 0 0 0 0 0 0 0 0 0
```

The same as above, key in “Z30” by sequence in the prompt line, press key to finish the alteration. The page is as follows:
### 9.2 The Password Setting and Alteration

To prevent the part programs, CNC parameters from malicious alteration, this GSK980MD provides an authority setting function that is graded for 4 levels. By descending sequence, they are machine builder (2\textsuperscript{nd}) level, equipment management (3\textsuperscript{rd}) level, technician (4\textsuperscript{th}) level, machining operation (5\textsuperscript{th}) level.

- **The 2\textsuperscript{nd} level**: Modification of the CNC bit parameter, data parameter, pitch data, tool offset data, part program edit, PLC ladder transmission etc. are allowed.
- **The 3\textsuperscript{rd} level**: Initial password 2345, the CNC bit parameter, data parameter, tool offset data, part program edit operations are allowed.
- **The 4\textsuperscript{th} level**: Initial password 1234, tool offset data (for tool setting), macro variables, part program edit operations are allowed; but the CNC bit parameter, data parameter, pitch data operations are unallowed.
- **The 5\textsuperscript{th} level**: No password. Only the machine panel operation is allowed, and the operations of part program edit and selection, the alteration operations of CNC bit parameter, data parameter, pitch data, tool offset data are unallowed.

<table>
<thead>
<tr>
<th>AUTH. OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td>Backup PAR.  (User)</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR.  (User)</td>
</tr>
<tr>
<td>➤ INPUT PASSWORD:________</td>
<td>Resume PAR.1 (Test)</td>
</tr>
<tr>
<td>UPDATE PASS. :________</td>
<td>Resume PAR.2 (Step)</td>
</tr>
<tr>
<td>Modify parameter and edit program</td>
<td>Resume PAR.3 (Servo)</td>
</tr>
<tr>
<td>MDI S0000 T00 H00</td>
<td></td>
</tr>
</tbody>
</table>

After entering the authority setting page, the cursor locates at the “INPUT PASSWORD:” line. It
may press the \( \uparrow \) or \( \downarrow \) key to move the cursor to the corresponding item.

- Press \( \uparrow \) key once, the cursor shifts a line upward. If the current cursor locates at the “SET LOWER LEVEL” line (1st line), press \( \uparrow \) key, the cursor shifts to the “UPDATE PASS:” line (end line).
- Press \( \downarrow \) key once, the cursor shifts a line upward. If the current cursor locates at the end line, by pressing \( \downarrow \) key once, the cursor moves to the 1st line.

9.2.1 Entry of the operation level

1. After entering the PASSWORD SETTING page, move the cursor to the “INPUT PASSWORD:” line;
2. Key in the password (an “*” sign added each time inputting a character);
3. Press \( \text{DATA INPUT} \) key to finish the inputting, and it will enter the corresponding password level.

Note: The length of this GSK980MD system password corresponds to the operation level, which can’t be added or decreased by user at will.

<table>
<thead>
<tr>
<th>Operation level</th>
<th>Password length</th>
<th>Initial password</th>
</tr>
</thead>
<tbody>
<tr>
<td>3rd</td>
<td>5 bits</td>
<td>12345</td>
</tr>
<tr>
<td>4th</td>
<td>4 bits</td>
<td>1234</td>
</tr>
<tr>
<td>5th</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Example: The current CNC level is the 4th level, as the following page shows. The 3rd level password of CNC is 12345, please alter the current level to the 3rd level.
Move the cursor to the “INPUT PASSWORD:” line, key in 12345, then press the [INPUT PASSWORD: key, the CNC prompts “Modify parameter and edit program”, “Password passed”, and the current level is the 3rd level. The page is as follows:

<table>
<thead>
<tr>
<th>AUTH. OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td>Backup PAR. (User))</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR. (User))</td>
</tr>
<tr>
<td>► INPUT PASSWORD:</td>
<td>Resume PAR. 1 (Test)</td>
</tr>
<tr>
<td>UPDATE PASS.:</td>
<td>Resume PAR. 2 (Step)</td>
</tr>
<tr>
<td></td>
<td>Resume PAR. 3 (Servo)</td>
</tr>
</tbody>
</table>

Modify parameter and edit program

MDI IMAGE STORED S0000 T00 H00

Note: When current operation authority is lower than or equal to the 3rd level (3rd, 4th, 5th level), the password level is not changed if repower the CNC system. If previous level is higher than the 3rd level (0, 1st, or 2nd level), it defaults the 3rd level.

9.2.2 Alteration of the password

Steps for password alteration:
1. After entering the PASSWORD SETTING page, enter the password by the methods in Section10.3.2;
2. Move the cursor to the "ALTER PASSWORD:” line;
3. Key in the new password, and press [INPUT PASSWORD: key
4. The CNC system prompts “PLEASE INPUT USER PASSWORD AGAIN”, the page is as follows:

<table>
<thead>
<tr>
<th>AUTH. OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td>Backup PAR. (User))</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR. (User))</td>
</tr>
<tr>
<td>► INPUT PASSWORD:</td>
<td>Resume PAR. 1 (Test)</td>
</tr>
<tr>
<td>UPDATE PASS.:</td>
<td>Resume PAR. 2 (Step)</td>
</tr>
<tr>
<td></td>
<td>Resume PAR. 3 (Servo)</td>
</tr>
</tbody>
</table>

Modify parameter and edit program

MDI IMAGE STORED S0000 T00 H00
5. After reinputting the password, press the key, if the two passwords input are identical, CNC prompts “PASSWORD UPDATED”. So the password alteration is successful.

<table>
<thead>
<tr>
<th>AUTH. OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td>Backup PAR. (User)</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR. (User)</td>
</tr>
<tr>
<td>INPUT PASSWORD:</td>
<td>Resume PAR.1 (Test)</td>
</tr>
<tr>
<td>UPDATE PASS. :</td>
<td>Resume PAR.2 (Step)</td>
</tr>
<tr>
<td>PASSWORD UPDATED.</td>
<td>Resume PAR.3 (Servo)</td>
</tr>
</tbody>
</table>

Modify parameter and edit program

MDI S0000 T00 H00

6. If the two passwords input are not identical, CNC prompts “PASSWORD CHECKOUT ERROR.”, the page is as follows:

<table>
<thead>
<tr>
<th>AUTH. OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td>Backup PAR. (User)</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR. (User)</td>
</tr>
<tr>
<td>INPUT PASSWORD:</td>
<td>Resume PAR.1 (Test)</td>
</tr>
<tr>
<td>UPDATE PASS. :</td>
<td>Resume PAR.2 (Step)</td>
</tr>
<tr>
<td>PASSWORD CHECKOUT ERROR.</td>
<td>Resume PAR.3 (Servo)</td>
</tr>
</tbody>
</table>

Modify parameter and edit program

MDI S0000 T00 H00

9.2.3 Lower level set

The demotion of the operation level is used to enter a lower level from a higher level, the steps are as follows:

1. After entering the PASSWORD SETTING page, key in the password by the method in Section 10.3.2

2. Move the cursor to the “SET LOWER LEVEL” line, if the current CNC operation is the 3rd level, the page is as follows:
### Chapter 9  Data Setting, Backup And Restore

<table>
<thead>
<tr>
<th>AUTH. OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td>Backup PAR. (User)</td>
</tr>
<tr>
<td></td>
<td>Resume PAR. (User)</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR.1 (Test)</td>
</tr>
<tr>
<td>INPUT PASSWORD:</td>
<td>Resume PAR.2 (Step)</td>
</tr>
<tr>
<td>UPDATE PASS. :</td>
<td>Resume PAR.3 (Servo)</td>
</tr>
</tbody>
</table>

3. Press the DATA INPUT key, the CNC prompts "CURRENT LEVEL TO 4, OK? "; the page is as follows:

<table>
<thead>
<tr>
<th>AUTH. OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td>Backup PAR. (User)</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR. (User)</td>
</tr>
<tr>
<td>INPUT PASSWORD:</td>
<td>Resume PAR.1 (Test)</td>
</tr>
<tr>
<td>UPDATE PASS. :</td>
<td>Resume PAR.2 (Step)</td>
</tr>
</tbody>
</table>

4. Press the DATA INPUT key again, if the demotion is successful, the page is as follows:

<table>
<thead>
<tr>
<th>AUTH. OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 4</td>
<td>Backup PAR. (User)</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR. (User)</td>
</tr>
<tr>
<td>INPUT PASSWORD:</td>
<td>Resume PAR.1 (Test)</td>
</tr>
<tr>
<td>UPDATE PASS. :</td>
<td>Resume PAR.2 (Step)</td>
</tr>
</tbody>
</table>

- Can edit prog, input macro var&offset

- **Note**: If the current level is the 5th level, the demotion operation is unallowed.
9.3 Data Restore and Backup

The user data (such as bit parameter and pitch data) can be backup (saved) and restored (read) in this GSK980MD system. It doesn't affect the part programs stored in the CNC system while backuping and restoring these data. The backup page is as follows:

Press \( \text{CHANGE} \) key repeatedly, “PASSWORD SETTING” and “DATA BACKUP” pages can be switched.

<table>
<thead>
<tr>
<th>DATA BACKUP</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td></td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td></td>
</tr>
<tr>
<td>INPUT PASSWORD:</td>
<td></td>
</tr>
<tr>
<td>UPDATE PASS. :</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Press \( \text{MDI} \) key to enter the MDI mode, then press \( \text{SETTING} \) key (or \( \text{CHANGE} \) key if necessary) to enter PASSWORD SETTING page;

Press \( \text{CHANGE} \), and switch to the Data Backup page.

Move the cursor to the desired item;

Press \( \text{DATA INPUT} \) and \( \text{P} \) keys together.

**Note**  Don’t cut off the power in the backup and restore operation of the data, and no other operation is suggested to be performed before the aforesaid operation is prompted to be finished.

Example: to restore the CNC parameter to 1μ level servo standard parameter, the steps are as follows:

Turn on the parameter switch, and enter the Backup PAR. page of MDI mode, move the cursor to “Recover Default PAR. (1μ level)”, as the following figure shows:
Chapter 9  Data Setting, Backup And Restore

<table>
<thead>
<tr>
<th>DATA BACKUP</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT LEVEL: 3</td>
<td>Backup PAR. (User)</td>
</tr>
<tr>
<td>SET LOWER LEVEL</td>
<td>Resume PAR. (User)</td>
</tr>
<tr>
<td>INPUT PASSWORD:_______</td>
<td>Resume PAR.1 (Test)</td>
</tr>
<tr>
<td>UPDATE PASS. :_______</td>
<td>Resume PAR.2 (Step)</td>
</tr>
<tr>
<td></td>
<td>▶ Resume PAR.3 (Servo)</td>
</tr>
<tr>
<td>SUCCEEDING IN RECOVERING SERVO PAR (POWER ON)</td>
<td></td>
</tr>
<tr>
<td>MDI</td>
<td>S0000 T00 H00</td>
</tr>
</tbody>
</table>

Press [DATA] [INPUT] keys together, the CNC system prompts “SERVO PAR BACKUP RECOVERED (POWER ON)”.
CHAPTER 10 ADVANCE OPERATION

Advance operation interface of GSK980MDa, which is as follows, is started by connecting CNC to USB. In this interface, communication between CNC & USB and system update operations can be done. Its transmission speed is much faster than traditional serial communication speed, greatly increases the efficiency of file transmission. More over, USB is easy to carry, to use and it supports hot plugging, plug and play at once.

<table>
<thead>
<tr>
<th>ADVANCED OPERATION</th>
<th>00000 N00000</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKUP</td>
<td></td>
</tr>
<tr>
<td>□ ALL</td>
<td>□ PAR</td>
</tr>
<tr>
<td>RECOVER</td>
<td></td>
</tr>
<tr>
<td>□ ALL</td>
<td>□ PAR</td>
</tr>
<tr>
<td>SOFTWARE UPGRADE</td>
<td></td>
</tr>
<tr>
<td>□ UPGRADE CNC SOFT.</td>
<td>□ resUPGRADE</td>
</tr>
<tr>
<td>FORMATTING SOFTWARE</td>
<td>FORMATTING SOFTWARE</td>
</tr>
<tr>
<td>NOTE: BACKUP PAR, PROGRAM, PLC TO S.</td>
<td></td>
</tr>
<tr>
<td>EDIT</td>
<td></td>
</tr>
<tr>
<td>□ S0000 T00 H00</td>
<td></td>
</tr>
</tbody>
</table>

10.1 Operation path

USB operation in 980MDa is searching and setting up destination list on U disk with its number. Therefore, the system with different number is corresponding to different U disk list in advance operation.

Example: If the number of system A is CT1010MDa, the list of advance operation on U disk is as follows:

![U disk with CT1010MDa list]

If the number of system B is CT2138MDa, the list of advance operation on U disk is as follows:

![U disk with CT2138MDa list]
If the system has no number, the list of advance operation on U disk is as follows:

Note: The number of the system can be found in version information page of diagnosis. The following contents are described by list of gsk980mda_backup.

- **Path explanations**

<table>
<thead>
<tr>
<th>Path file folder</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>user\</td>
<td>Target position for parameter and PLC file backup and restore</td>
</tr>
<tr>
<td>prog\</td>
<td>Target position for part program file backup and restore</td>
</tr>
</tbody>
</table>

- **File specification**

<table>
<thead>
<tr>
<th>File name</th>
<th>Expended name</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter file</td>
<td>Para1, Para2, Para3</td>
<td>.par</td>
</tr>
<tr>
<td>Part program</td>
<td>O0000 ~ O9999</td>
<td>.CNC</td>
</tr>
<tr>
<td>PLC file</td>
<td>plc ~ plc7</td>
<td>.ldx</td>
</tr>
</tbody>
</table>

- **Operation authority**

<table>
<thead>
<tr>
<th>Backup operation</th>
<th>Parameter</th>
<th>Authority level 3 (including level 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part program</td>
<td>Authority level 3 (including level 3)</td>
<td></td>
</tr>
<tr>
<td>Ladder diagram</td>
<td>Authority level 3 (including level 3)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Restore operation</th>
<th>Parameter</th>
<th>Authority level 3 (including level 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part program</td>
<td>Authority level 3 (including level 3)</td>
<td></td>
</tr>
<tr>
<td>Ladder diagram</td>
<td>Authority level 2 (including level 2)</td>
<td></td>
</tr>
</tbody>
</table>
10.2 Operation instructions

➢ Key descriptions

Cursor moving: Press direction keys to move the cursor.

Menu selection: Press key to select the operation item which cursor is in.

Menu cancellation: Press key to cancel the operation item which cursor is in.

Operation execution: Press key to execute all operation items selected in current column.

Operation confirmation: Execution needs to be confirmed, please press key to confirm or press key to cancel the execution.

➢ Parameter restore and backup

Backup the parameter: Copy all parameter states and values to U:\gsk980MDa_backup\user\ of USB memory unit in the form of file Para1.par, Para2.par, Para3.par. If the above-mentioned file does not exist, set up a new one: If the file exists, this file will be overwritten by the new one.

Restore the parameter: Copy parameter files from USB memory unit U:\gsk980MDa_backup\user\ back to the CNC system to restore the system parameter. Restore operation cannot be done if the above-mentioned path is moved or altered or irregular file name is renamed.

Note: Repower the CNC system after parameter load is successful.

➢ Part program restore and backup

Backup the part parameter: Copy all part programs of current system to U:\gsk980MDa_backup\user\prog\ of USB memory unit in the form of file .CNC. If the above-mentioned file does not exist, set up a new one: If the file exists, this file will be overwritten by the new one.

Restore the part program: Copy all part programs from USB memory unit U:\gsk980MDa_backup\user\prog\ back to the CNC system to restore the part program. Restore operation cannot be done if the above-mentioned path is moved or altered or irregular file name is renamed.
Chapter 10 Advance Operation

- **Ladder diagram (PLC) restore and backup**

  The ladder diagram backup: Copy all ladder diagrams (.ldx file) of the current system to U:\gsk980MDa_backup\user\ of USB memory unit. If the above-mentioned file does not exist, set up a new one: If the file exists, this file will be overwritten by the new one.

  Restore the ladder diagram: Copy parameter files from USB memory unit U:\gsk980MDa_backup\user\ back to the CNC system to restore the ladder diagram. Restore operation cannot be done if the above-mentioned path is moved or altered or irregular file name is renamed.

  **Note:** Repower the CNC system after the ladder diagram restore is successful.

### 10.3 Attentions

- **Notice:** If a file or list on target path has the same name as the one will be copied, it will be overwritten and replaced by the system automatically. Therefore, to prevent the file or list from overwriting or replacing, please copy and save it separately.
- It forbids doing any other operation in advance operation. Once operation is performed, it cannot be interrupted until it is finished.
- If the file to be saved or restored is large, operation time will be long. Please wait.
- Pull out USB if abnormal conditions occur, then connect it again.
11.1. File list

Press \( \text{MDI} \) or \( \text{EDIT} \) key to select [MDI] or [EDIT] mode, press \( \text{PROGRAM} \) key to enter [file list] interface, the page is as follows:

```
FILE LIST
00000 N00000
C:/user
├── 00000.CNC
├── 00001.CNC
├── 00002.CNC
├── 00003.CNC
├── 00004.CNC
├── 00005.CNC
├── 00006.CNC
├── 00007.CNC
└── 00008.CNC
```

In edit or MDI mode, press \( \text{CHANGE} \) key to identify U disk.

If identification is unsuccessful, it prompts: “Fail to connect U disk”. If identification is successful, the following file list will be displayed.

```
FILE LIST
00000 N00000
C:/user
├── 00000.CNC
├── 00001.CNC
├── 00002.CNC
├── 00003.CNC
├── 00004.CNC
├── 00005.CNC
├── 00006.CNC
├── 00007.CNC
└── 00008.CNC
```

Special explanation:

The list information of disk CNC is displayed at the page left and list information of disk USB is displayed at the page right. The display column will not display any information if U disk is not detected. Character entry box, file attributes information and user operation prompts are displayed at the bottom of the page.

1. Current list page only display the list information of the currently opened folder.
2. U disk can be identified in edit or MDI mode.
3. It not support Chinese complex characters.
4. It not support Chinese long file name, only the first three characters .+“¬1”of this file name can be displayed.
5. Non-CNC file of C disk and U disk is displayed.

Note: The file name, which consists of “O”+“4 digits”+“.CNC ”, is considered to be CNC format file.

11.2. Introduction of general file operation function

11.2.1 Open and close file folder

Move the cursor to the folder will be opened.

Press  key to open the folder. The list which the file locates is displayed in the first line (long list is scrolling display)

Press  key to close the folder and return to the next higher level of the list.
11.2.2 Copy the file by one key (current list in C disk → current list in U disk)

In "edit" mode, select the CNC format file, press key to copy it. See the following figure:

① Select CNC file, press :

② After duplication is successful, the cursor moves to the next file in current list. The list on the other side is refreshed at once.
11.2.3 CNC file search

In “EDIT” and “AUTO” mode, input target program number in input column, and press up or down to search this program.

If program search is successful after input “O5”, the cursor moves to target program. If this program cannot be searched, “the file does not exist” will be prompted at message column.
11.2.4 Open CNC file

1. In "EDIT" and "AUTO" mode, select the CNC format file when there is no program execution.

   In "EDIT" and "AUTO" mode, select the CNC format file when there is no program execution.

   Special explanations:
   1. The program above number 9000 cannot be opened with authority level 3 or under.

   Special explanations:
   1. The program above number 9000 cannot be opened with authority level 3 or under.
level 3.

2. The program file can not be opened with authority level 5.

Attention:

1. In “program content”, it is not allowed to do any operation on U disk. These operations are: setting-up, duplication, rename, deletion, editing, save, etc. Process and check operations can be done for programs on U disk in page “program content”.

2. The called subprogram in auto-run should in a same level of list with main program.

3. Pull out U disk when it is open, system alarm occurs “U disk is not connected”.

At this time, plug in U disk again, press key to detect U disk in MDI mode, or press keys to clear the alarm.
VOLUME III INSTALLATION
CHAPTER 1 INSTALLATION LAYOUT

1.1 GSK980MDa Connection

1.1.1 GSK980MDa back cover interface layout

Fig 1-1 GSK980MDa back cover interface layout

1.1.2 Interface explanation

- Power box: GSK-PB2, for +5V, +24V, +12V, -12V, GND power supply
- CN11: X axis, 15-core DB female socket, for connecting X axis drive unit
- CN12: Y axis, 15-core DB female socket, for connecting Y axis drive unit
- CN13: Z axis, 15-core DB female socket, for connecting Z axis drive unit
- CN14: 4th axis, 15-core DB female socket, for connecting 4th axis drive unit
- CN21: coder, 15-core DB female socket, for connecting Encoder
- CN51: inverter, 9-core DB male socket, for connecting pc RS232 interface
- CN15: 5th axis & spindle port, 25-core DB male socket, for connecting inverter & 5th axis
- CN31: handwheel, 26-core 3 line female socket, for connecting handwheel
- CN62: output, 44-core 3 lines female socket, for sending the signal of CNC to machine
- CN61: input, 44-core 3 line male socket, for sending the signal of machine to CNC
1.2 GSK980MDa Installation

1.2.1 GSK980MDa external dimensions

Fig. 1-2 GSK980MDa external dimensions

1.2.2 Installation conditions of the cabinet

- The dust, cooling liquid and organic resolution should be effectively prevented from entering the cabinet;
- The designed distance between the CNC back cover and the cabinet should be not less than 20cm, the inside and outside temperature difference of the cabinet should be no less than 10°C temperature rises when the cabinet inside temperature rises;
- Fans should be fixed in the cabinet to ventilate it;
- The panel should be installed in a place where the coolant can’t splash;
- The external electrical interference should be taken into consideration in cabinet design to prevent it from transferring to CNC system.

1.2.3 Protection methods against interference

In order to ensure the CNC stable working, the anti-interference technology such as space electromagnetic radiation shielding, impact current absorbing, power mixed wave filtering are employed in CNC design. And the following measures are necessary during CNC connection:

1. Make CNC far from the interference devices (inverter, AC contactor, static generator, high-pressure generator and powered sectional devices etc.);
2. To supply the CNC via an isolation transformer, the machine with the CNC
should be grounded, the CNC and drive unit should be connected with independent grounding wires at the grounding point;

3. To suppress interference: connect parallel RC circuit at both ends of AC coil (Fig. 1-4), RC circuit should approach to inductive loading as close as possible; reversely connect parallel freewheeling diode at both ends of DC coil (Fig. 1-5); connect parallel surge absorber at the ends of AC motor coil (Fig. 1-6);

![Diagram 1-4](image1.png)

![Diagram 1-5](image2.png)

4. To employ with twisted shield cable or shield cable for the leadout cable of CNC, the cable shield tier is grounded by single end at CNC side, signal cable should be as short as possible;

5. In order to decrease the mutual interference between CNC cables or CNC cables with strong-power cables, the wiring should comply to the following principles:
<table>
<thead>
<tr>
<th>Group</th>
<th>Cable type</th>
<th>Wiring requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>AC power line</td>
<td>Tie up A group cables with a clearance at least 10cm from that of B, C groups, or shield A group cables from electromagnetism</td>
</tr>
<tr>
<td></td>
<td>AC coil</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AC contactor</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>DC coil (24VDC)</td>
<td>Tie up B and A group cables separately or shield B group cables; and the further B group cables are from that of C group, the better it is</td>
</tr>
<tr>
<td></td>
<td>DC relay (24VDC)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cables between CNC and strong-power cabinet</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cables between CNC and machine</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Cables between CNC and servo drive unit</td>
<td>Tie up C and A group cables separately, or shield C group cables; and the cable distance between C group and B group is at least 10cm with twisted pair cable applied.</td>
</tr>
<tr>
<td></td>
<td>Position feedback cable</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Position encoder cable</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MPG cable</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Other cables for shield</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 2 DEFINITION & CONNECTION OF INTERFACE SIGNALS

2.1 Connection to Drive unit

2.1.1 Drive interface definition

![Diagram of CN11, CN12, CN13 interface (DB15 female)]

2.1.2 Command pulse and direction signals

nCP+, nCP- are command pulse signals, nDIR+, nDIR- are command direction signals. These two group signals are both difference output (AM26LS31), the interior circuit for them is shown in Fig. 2-2.

![Diagram of interior circuit of command pulse and direction signals]

2.1.3 Drive unit alarm signal

The low or high level of the drive unit alarm level is set by the CNC bit parameter No.009 BIT0~BIT4, whose interior circuit is shown in Fig. 2-3:

![Diagram of interior circuit of drive unit alarm signal]
This input circuit requires that the drive unit transmits signal by the following types in Fig. 2-4:

![Signal types of drive unit](image)

**Fig.2-4** Signal types of drive unit

**2.1.4 Axis enable signal ENn**

nEN signal output is valid as CNC works normally (nEN signal to 0V); when the drive unit alarm or emergency alarm occurs, CNC cuts off nEN signal output (nEN signal to 0V off). The interior interface circuit is shown in Fig.2-5:

![Interior interface circuit for axis enable signal](image)

**Fig.2-5** Interior interface circuit for axis enable signal

**2.1.5 Pulse disable signal SETn**

nSET signal is used to control servo input disable which can enhance the anti-disturbance capability between CNC and drive unit. This signal is at low level if there is pulse output from CNC, high resistance if not. The interior interface circuit of it is shown in Fig. 2-6:

![Interior interface circuit for pulse disable signal](image)

**Fig.2-6** Interior interface circuit for pulse disable signal

**2.1.6 Zero signal nPC**

The one-rotation or approach switch signal is taken as zero signal for machine zero return. Its interior connection circuit is shown in Fig.2-7.
Chapter 2 Definition & Connection of Interface Signals

Fig. 2-7 Zero signal circuit

Note: nPC signal uses +24V level.

a) The connection for NPN Hall elements taken as both deceleration signal and zero signal is shown in Fig. 2-8:

Fig. 2-8 Connection using NPN Hall elements

b) The connection for PNP Hall elements taken as both deceleration signal and zero signal is shown in Fig. 2-9:

Fig 2-9 Connection using PNP Hall elements

2.1.7 Connection to drive unit

The connection of GSK 980MDa to GSK drive unit is shown in Fig. 2-10:
2.2 Connection of 4th axis

2.2.1 4th axis interface definition

<table>
<thead>
<tr>
<th>Signal</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP4+, CP4-</td>
<td>Command pulse signal</td>
</tr>
<tr>
<td>DIR4+, DIR4-</td>
<td>Command direction signal</td>
</tr>
<tr>
<td>PC4</td>
<td>Zero signal</td>
</tr>
<tr>
<td>ALM4</td>
<td>Drive alarm signal</td>
</tr>
<tr>
<td>EN4</td>
<td>Axis enable signal</td>
</tr>
<tr>
<td>SET4</td>
<td>Pulse disable signal</td>
</tr>
</tbody>
</table>

Fig.2-10 Connection of 4th axis interface to drive unit

Fig.2-11 Interface CN14 (DB15 female)
2.2.2 Connection of 4th axis interface as linear axis

![Diagram of connection of 4th axis interface to drive unit]

Fig.2-12 Connection of 4th axis interface to drive unit
2.2.3 Connection of 4th axis interface as rotary axis

![Connection Diagram](image)

Fig. 2-13 Connection of 4th axis interface to spindle drive unit

2.3 Connection of spindle port

2.3.1 Definition of signal

<table>
<thead>
<tr>
<th>Pin</th>
<th>Signal</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CP5+</td>
<td>Spindle pulse signal</td>
</tr>
<tr>
<td>2</td>
<td>DIR5+</td>
<td>Spindle direction signal</td>
</tr>
<tr>
<td>3</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>ALM5</td>
<td>Spindle alarm signal</td>
</tr>
<tr>
<td>5</td>
<td>X5.0</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>X5.2</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>RDY5</td>
<td>Spindle is ready</td>
</tr>
<tr>
<td>8</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>PC5</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>+24V</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>SVC</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Y5.0~Y5.3</td>
<td>PLC Address, only for these, Lower voltage is valid</td>
</tr>
<tr>
<td>14</td>
<td>CP5-</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>DIR5-</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>+24V</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>SET5</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>EN5</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 2-14 CN15 Spindle Prot

2.3.2 Spindle zero signal

Except for the PC5 signal, other fixed signals of the spindle interface are the same as that of the X, Y, Z, 4th axes. The PC5 interface circuit is shown as follows:
2.3.3 Linear axis

![Spindle zero signal interface circuit](image)

**Fig. 2-15** Spindle zero signal interface circuit

![Connection of spindle interface to drive unit](image)

**Fig. 2-16** Connection of spindle interface to drive unit

2.3.4 Connected with inverter

The connection of GSK980MDa with convertor is shown in Fig. 2-17:

![Connection of GSK980MDa to inverter](image)

**Fig. 2-17** Connection of GSK980MDa to inverter
2.3.5 Connection of spindle interface as rotary axis

![Connection of spindle to DAP03](image)

Fig.2-18 Connection of spindle to DAP03

2.3.6 Connection of spindle interface as “CS” axis

![Connection of spindle to DAP03](image)

Fig.2-19 Connection of spindle to DAP03

2.3.7 SVC Signal explanation

The analog spindle interface SVC can output 0~10V voltage, its interior signal circuit is shown in Fig. 2-20:
Chapter 2 Definition & Connection of Interface Signals

2.4 Connection to Spindle Encoder

2.4.1 Spindle encoder interface definition

2.4.2 Signal Explanation

MPZ-/MPZ+, MPB-/MPB+, MPA-/MPA+ are the encoder Z, B, A phase differential input signals respectively, which are received by 26LS32; MPB-/MPB+, MPA-/MPA+ are normal square wave of phase shift 90° with the maximum signal frequency less than 1MHz; the encoder pulses for GSK980MDa are set by data parameter No.109, whose range is from 100 to 5000.

Its interior connection circuit is shown in Fig. 2-22: ( n=A, B, C )

2.4.3 Connection of spindle encoder interface

The connection of GSK980MDa to spindle encoder is shown in Fig. 2-23, twisted pair cables are used to connection.
2.5 Connection to Handwheel

2.5.1 Handwheel interface definition

<table>
<thead>
<tr>
<th>Signal</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>HA+, HA-</td>
<td>Handwheel A phase signal</td>
</tr>
<tr>
<td>HB+, HB-</td>
<td>Handwheel B phase signal</td>
</tr>
<tr>
<td>X6.0~X6.5</td>
<td>PLC adress</td>
</tr>
<tr>
<td>+24V</td>
<td>Direct current</td>
</tr>
</tbody>
</table>

Fig.2-23 Connection of GSK980MDa to encoder

2.5.2 Signal explanation

"HA+", "HA-", "HB+", "HB-" are the input signals of handwheel A and B phases. Its interior connection circuit is shown in Fig. 2-25:
Chapter 2 Definition & Connection of Interface Signals

2.6 Connection of GSK980MDa to PC

2.6.1 Communication interface definition

<table>
<thead>
<tr>
<th>Signal</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>RXD</td>
<td>For date reception</td>
</tr>
<tr>
<td>TXD</td>
<td>For date transmitting</td>
</tr>
<tr>
<td>GND</td>
<td>For signal grounding</td>
</tr>
</tbody>
</table>

![Fig.2-27 CN51 communication interface (DB9 female socket)](image)

2.6.2 Communication interface connection

The communication between GSK980MDa and PC can be done via RS232 interface (GSK980MDa communication software needed), The connection of them is shown in Fig.2-28.
The communication of a GSK980MDa to another GSK980MDa can be made via their CN51 interfaces, and the connection of them is shown in Fig.2-29:

![Connection Diagram](image)

Fig.2-29 Communication connection of GSK980MDa to GSK980MDa

### 2.7 Connection of Power Interface

GSK-PB2 power box is applied in this GSK980MDa, which involves 4 groups of voltage: +5V (3A), +12V(1A), -12V(0.5A), +24V(0.5A), and its common terminal is COM(0V). The connection of GSK-PB2 power box to GSK980MDa CN1 interface has been done for its delivery from factory, and the user only need to connect it to a 220V AC power in using:

The interface definition of GSK980MDa CN1 is shown below:

![Power Supply Interface](image)

Fig.2-30
2.8 I/O Interface Definition:

CN61: 44-core (3-line) male socket

<table>
<thead>
<tr>
<th>NO.</th>
<th>Address</th>
<th>NO.</th>
<th>Address</th>
<th>NO.</th>
<th>Address</th>
<th>NO.</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X0.0</td>
<td>12</td>
<td>X1.3 (DECZ)</td>
<td>23</td>
<td>GND</td>
<td>34</td>
<td>X2.5 (DEC5)</td>
</tr>
<tr>
<td>2</td>
<td>X0.1</td>
<td>13</td>
<td>X1.4</td>
<td>24</td>
<td>GND</td>
<td>35</td>
<td>X2.6</td>
</tr>
<tr>
<td>3</td>
<td>X0.2</td>
<td>14</td>
<td>X1.5</td>
<td>25</td>
<td>GND</td>
<td>36</td>
<td>X2.7</td>
</tr>
<tr>
<td>4</td>
<td>X0.3 (DECX)</td>
<td>15</td>
<td>X1.6</td>
<td>26</td>
<td>GND</td>
<td>37</td>
<td>X3.0</td>
</tr>
<tr>
<td>5</td>
<td>X0.4</td>
<td>16</td>
<td>X1.7</td>
<td>27</td>
<td>GND</td>
<td>38</td>
<td>X3.1</td>
</tr>
<tr>
<td>6</td>
<td>X0.5 (ESP)</td>
<td>17</td>
<td>GND</td>
<td>28</td>
<td>GND</td>
<td>39</td>
<td>X3.2</td>
</tr>
<tr>
<td>7</td>
<td>X0.6</td>
<td>18</td>
<td>GND</td>
<td>29</td>
<td>X2.0</td>
<td>40</td>
<td>X3.3</td>
</tr>
<tr>
<td>8</td>
<td>X0.7</td>
<td>19</td>
<td>GND</td>
<td>30</td>
<td>X2.1</td>
<td>41</td>
<td>X3.4</td>
</tr>
<tr>
<td>9</td>
<td>X1.0</td>
<td>20</td>
<td>GND</td>
<td>31</td>
<td>X2.2</td>
<td>42</td>
<td>X3.5 (SKIP)</td>
</tr>
<tr>
<td>10</td>
<td>X1.1</td>
<td>21</td>
<td>GND</td>
<td>32</td>
<td>X2.3 (DECY)</td>
<td>43</td>
<td>X3.6</td>
</tr>
<tr>
<td>11</td>
<td>X1.2</td>
<td>22</td>
<td>GND</td>
<td>33</td>
<td>X2.4 (DEC4)</td>
<td>44</td>
<td>X3.7</td>
</tr>
</tbody>
</table>

CN62: 44-core (3-line) female socket

<table>
<thead>
<tr>
<th>NO.</th>
<th>Address</th>
<th>NO.</th>
<th>Address</th>
<th>NO.</th>
<th>Address</th>
<th>NO.</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Y0.0</td>
<td>12</td>
<td>Y1.3</td>
<td>23</td>
<td>+24V</td>
<td>34</td>
<td>Y2.5</td>
</tr>
<tr>
<td>2</td>
<td>Y0.1</td>
<td>13</td>
<td>Y1.4</td>
<td>24</td>
<td>+24V</td>
<td>35</td>
<td>Y2.6</td>
</tr>
<tr>
<td>3</td>
<td>Y0.2</td>
<td>14</td>
<td>Y1.5</td>
<td>25</td>
<td>+24V</td>
<td>36</td>
<td>Y2.7</td>
</tr>
<tr>
<td>4</td>
<td>Y0.3</td>
<td>15</td>
<td>Y1.6</td>
<td>26</td>
<td>GND</td>
<td>37</td>
<td>Y3.0</td>
</tr>
<tr>
<td>5</td>
<td>Y0.4</td>
<td>16</td>
<td>Y1.7</td>
<td>27</td>
<td>GND</td>
<td>38</td>
<td>Y3.1</td>
</tr>
<tr>
<td>6</td>
<td>Y0.5</td>
<td>17</td>
<td>GND</td>
<td>28</td>
<td>GND</td>
<td>39</td>
<td>Y3.2</td>
</tr>
<tr>
<td>7</td>
<td>Y0.6</td>
<td>18</td>
<td>GND</td>
<td>29</td>
<td>Y2.0</td>
<td>40</td>
<td>Y3.3</td>
</tr>
<tr>
<td>8</td>
<td>Y0.7</td>
<td>19</td>
<td>GND</td>
<td>30</td>
<td>Y2.1</td>
<td>41</td>
<td>Y3.4</td>
</tr>
<tr>
<td>9</td>
<td>Y1.0</td>
<td>20</td>
<td>+24V</td>
<td>31</td>
<td>Y2.2</td>
<td>42</td>
<td>Y3.5</td>
</tr>
<tr>
<td>10</td>
<td>Y1.1</td>
<td>21</td>
<td>+24V</td>
<td>32</td>
<td>Y2.3</td>
<td>43</td>
<td>Y3.6</td>
</tr>
<tr>
<td>11</td>
<td>Y1.2</td>
<td>22</td>
<td>+24V</td>
<td>33</td>
<td>Y2.4</td>
<td>44</td>
<td>Y3.7</td>
</tr>
</tbody>
</table>

Note 1: The I/O function of GSK980MDa drilling and milling CNC is defined by ladder diagram;
Note 2: If output function is valid, the output signal is on to 0V. If output function is invalid, the output signal is cut off by high impedance;
Note 3: If input function is valid, the input signal is on to 24V. If input function is invalid, the input signal is cut off with it;
Note 4: The effectiveness of +24V, 0V is equal to GSK980MD power box terminals that have the same name;
Note 5: XDEC, YDEC, ZDEC, DEC4, DEC5, ESP, SKIP are fixed signals that can't be altered.

2.8.1 Input Signal

Input signal means the signal from machine to CNC, when this signal is on with +24V, the input is valid; when it is off with +24V, the input is invalid. The contact point of input signal at machine side should meet the following conditions:
The capacity of the contact point: DC30V, 16mA above
Leakage current between contact points in open circuit: 1mA below
Voltage drop between contact points in closed circuit: 2V below (current 8.5mA, including cable voltage drop)

There are two external input types for input signals: one type is input by trigger point switch whose signals are from keys, stroke switch and contacts of relay at machine side, as is shown in Fig 2-31:

![Fig.2-31](image1)

The other type is input by switch with no contacts (transistor), as is shown in Fig. 2-32, 2-33

![Fig.2-32 Connection of NPN](image2)

![Fig.2-33 Connection of PNP](image3)
2.8.2 Output signal

The output signal is used for the machine relay and indicator, if it is on with 0V, the output function is valid; if it is off with 0V, the output function is invalid. There are total 36 digital volume outputs in I/O interface that they all have the same structure as is shown in Fig.2-34:

![Fig.2-34 Circuit for digital volume output module](image)

The logic signal OUTx output from the main board is sent to the input terminal of inverter (ULN2803) via a connector. And there are 2 output types for nOUTx: output with 0V, or high impedance. Its typical application is shown in follows:

- **To drive LED**
  
  A serial resistance is needed to limit the current (usually 10mA) that goes through the LED by using ULN2803 output to drive LED, which is shown in Fig.2-35

![Fig.2-35](image)

- **To drive filament indicator**
  
  An external preheat resistance is needed to decrease the current impact at power on by using ULN2803 output to drive filament indicator, and this resistance value should be within a range that the indicator can’t light up. It is shown in Fig.2-36:

![Fig.2-36](image)
To drive inductive load (relay etc.)
To use ULN2803 output to drive an inductive load, it requires to connect a freewheeling diode near the coil to protect output circuit and deduce interference. It is shown in Fig.2-37:

![Diagram](image)

**Fig.2-37**

### 2.9 Machine Zero

#### Relative signal

<table>
<thead>
<tr>
<th>DECX</th>
<th>X axis deceleration signal</th>
<th>PCX</th>
<th>X axis zero signal</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECY</td>
<td>Y axis deceleration signal</td>
<td>PCY</td>
<td>Y axis zero signal</td>
</tr>
<tr>
<td>DECZ</td>
<td>Z axis deceleration signal</td>
<td>PCZ</td>
<td>Z axis zero signal</td>
</tr>
<tr>
<td>DEC4</td>
<td>4th axis deceleration signal</td>
<td>PC4</td>
<td>4th axis zero signal</td>
</tr>
<tr>
<td>DEC5</td>
<td>5th axis deceleration signal</td>
<td>PC5</td>
<td>5th axis zero signal</td>
</tr>
</tbody>
</table>

#### CNC diagnosis

<table>
<thead>
<tr>
<th>0 0 0 0</th>
<th>DEC5</th>
<th>DEC4</th>
<th>DECZ</th>
<th>DECY</th>
<th>DECX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corresponding pin-out</td>
<td>CN61.34</td>
<td>CN61.33</td>
<td>CN61.12</td>
<td>CN61.32</td>
<td>CN61.4</td>
</tr>
<tr>
<td>PLC address</td>
<td>X2.5</td>
<td>X2.4</td>
<td>X1.3</td>
<td>X2.3</td>
<td>X0.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>0 0 8</th>
<th>PC5</th>
<th>PC4</th>
<th>PCZ</th>
<th>PCY</th>
<th>PCX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corresponding pin-out</td>
<td>CN15.1 0</td>
<td>CN14.3</td>
<td>CN13.3</td>
<td>CN12.3</td>
<td>CN11.3</td>
</tr>
</tbody>
</table>

#### Bit parameter

<table>
<thead>
<tr>
<th>0 0 4</th>
<th>DECI</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEC1</td>
<td>=1: Deceleration signal is on with 24V for deceleration when machine zero return is performed</td>
</tr>
<tr>
<td></td>
<td>=0: Deceleration signal is off 24V for deceleration when machine zero return is performed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>0 0 6</th>
<th>ZM5</th>
<th>ZM4</th>
<th>ZM2</th>
<th>ZMY</th>
<th>ZMX</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZMX</td>
<td>=1: X axis machine zero return type C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>=0: X axis machine zero return type B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZMY</td>
<td>=1: Y axis machine zero return type C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter 2 Definition & Connection of Interface Signals

ZMZ = 1: Z axis machine zero return type C;
= 0: Z axis machine zero return type B.

ZM4 = 1: 4th axis machine zero return type C;
= 0: 4th axis machine zero return type B.

ZM5 = 1: 5th axis machine zero return type C;
= 0: 5th axis machine zero return type B.

ZCX = 1: The deceleration signal \( \text{DECX} \) and one-rotation signal \( \text{PCX} \) of X axis are in parallel connection during machine zero return (a proximity switch acting as both the deceleration signal and zero signal);
= 0: The deceleration signal \( \text{DECX} \) and one-rotation signal \( \text{PCX} \) of X axis are connected independently during machine zero return (the independent deceleration signal and zero signal are required).

ZCY = 1: The deceleration signal \( \text{DECY} \) and one-rotation signal \( \text{PCY} \) of Y axis are in parallel connection during machine zero return (a proximity switch acting as both the deceleration signal and zero signal);
= 0: The deceleration signal \( \text{DECY} \) and one-rotation signal \( \text{PCY} \) of Y axis are connected independently during machine zero return (the independent deceleration signal and zero signal are required).

ZCZ = 1: The deceleration signal \( \text{DECZ} \) and one-rotation signal \( \text{PCZ} \) of Z axis are in parallel connection during machine zero return (a proximity switch acting as both the deceleration signal and zero signal);
= 0: The deceleration signal \( \text{DECZ} \) and one-rotation signal \( \text{PCZ} \) of Z axis are connected independently during machine zero return (the independent deceleration signal and zero signal are required).

ZC4 = 1: The deceleration signal \( \text{DEC4} \) and one-rotation signal \( \text{PC4} \) of 4th axis are in parallel connection during machine zero return (a proximity switch acting as both the deceleration signal and zero signal);
= 0: The deceleration signal \( \text{DEC4} \) and one-rotation signal \( \text{PC4} \) of 4th axis are connected independently during machine zero return (the independent deceleration signal and zero signal are required).

ZC5 = 1: The deceleration signal \( \text{DEC5} \) and one-rotation signal \( \text{PC5} \) of 5th axis are in parallel connection during machine zero return (a proximity switch acting as both the deceleration signal and zero signal);
= 0: The deceleration signal \( \text{DEC5} \) and one-rotation signal \( \text{PC5} \) of 5th axis are connected independently during machine zero return (the independent deceleration signal and zero signal are required).
independently during machine zero return (the independent deceleration signal and zero signal are required).

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>1</th>
<th></th>
<th></th>
<th>ZNIK</th>
</tr>
</thead>
</table>

ZNLK =1: The direction keys are locked as machine zero return is performed, by pressing the direction key once, it moves to the machine zero automatically and stops. By pressing the [RESET] key at the machine zero return, the motion stops immediately;

=0: The direction keys are not locked as machine zero return is performed, but the direction keys should be pressed and held on

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th></th>
<th></th>
<th>ISOT</th>
</tr>
</thead>
</table>

ISOT =1: Manual rapid traverse valid prior to machine zero return;

=0: Manual rapid traverse invalid prior to machine zero return.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>4</th>
<th></th>
<th></th>
<th>ZRS5</th>
<th>ZRS4</th>
<th>ZRSZ</th>
<th>ZRSY</th>
<th>ZRSX</th>
</tr>
</thead>
</table>

ZRSZ, ZRSX, ZRSY, ZRS4, ZRS5 =1: To select machine zero return type B, C, which have machine zero, it needs to detect deceleration and zero signals in machine zero return;

=0: To select machine zero return type A, which has no machine zero, it does not detect deceleration and zero signals in machine zero return.

<table>
<thead>
<tr>
<th>0</th>
<th>2</th>
<th>2</th>
<th></th>
<th></th>
<th>MZR5</th>
<th>MZR4</th>
<th>MZRZ</th>
<th>MZRY</th>
<th>MZRX</th>
</tr>
</thead>
</table>

MZRX, MZRZ, MZRY, MZR4, MZR5 =1: The direction of zero return is negative for X, Z, Y, 4th, 5th axes;

=0: The direction of zero return is positive for X, Z, Y, 4th, 5th axes

### Date parameter

| 089 | Low speed of machine zero return of X axis |
| 090 | Low speed of machine zero return of Y axis |
| 091 | Low speed of machine zero return of Z axis |
| 092 | Low speed of machine zero return of 4th axis |
| 093 | Low speed of machine zero return of 5th axis |

| 094 | High speed of machine zero return of X axis |
| 095 | High speed of machine zero return of Y axis |
| 096 | High speed of machine zero return of Z axis |
| 097 | High speed of machine zero return of 4th axis |
| 098 | High speed of machine zero return of 5th axis |

| 130 | X axis machine zero offset (0.001) |
| 131 | Y axis machine zero offset (0.001) |
| 132 | Z axis machine zero offset (0.001) |
| 133 | The 4th axis machine zero offset (0.001) |
| 134 | The 5th axis machine zero offset (0.001) |
Chapter 2 Definition & Connection of Interface Signals

### Signal connection

The interior wiring circuit of deceleration signal is shown in Fig.2-37

![Fig.2-37](image)

- **Machine zero return type B** by regarding servo motor one-rotation signal as zero signal
  - Its sketch map is shown in follows:
② The circuit of deceleration signal (for three axes)

![Circuit Diagram]

When ZMn(n is X, Y, Z, 4th, 5th axis) of the bit parameter No.006, ZCn(n=X, Y, Z, 4th, 5th) of bit parameter No.007 and the BIT5(DEC1) of the bit parameter No.004 are all set to 0, the deceleration signal low level is valid. The action time sequence of machine zero return is shown in follows

![Action Time Sequence Diagram]

③ Action time sequence of machine zero return

When ZMn(n is X, Y, Z, 4th, 5th axis) of the bit parameter No.006, ZCn(n=X, Y, Z, 4th, 5th) of bit parameter No.007 and the BIT5(DEC1) of the bit parameter No.004 are all set to 0, the deceleration signal low level is valid. The action time sequence of machine zero return is shown in follows
Machine zero return process

A: Select machine zero return mode, press the manual positive or negative feed key (machine zero return direction is set by bit parameter No.022), the corresponding axis moves to the machine zero by a rapid traverse speed. As the axis press down the deceleration switch to cut off deceleration signal, the feed slows down immediately, and it continues to run in a fixed low speed.

B: When the deceleration switch is released, the deceleration signal contact point is closed again. And CNC begins to detect the encoder one-rotation signal, if the signal level changes, the motion will be stopped. And the corresponding zero indicator on the operator panel lights up for machine zero return completion.

- Machine zero return type B as an proximity switch is taken as both deceleration and zero signals
  ① Its sketch map is shown in follows:

  ![Sketch map](image)

  In figure: usually L1≥(1.5 – 2) folds width of the approach switch, L2=the width of the approach switch

Fig.2-42

② Wiring of the deceleration signal
   See details in Section 2.1.6 of this chapter

③ Action time sequence of machine zero return
   When ZMn (n is X,Y,Z,4th,5th axis) of the bit parameter No.006 and the BIT5 (DECI) of the bit parameter No.004 are all set to 0, ZCn (n is X,Y,Z,4th,5th axis) of the bit parameter No.007 is set to 1, the deceleration signal low level is valid. The action time sequence of zero return is shown as follows:
Machine zero return process

A: Select the Machine Zero mode, press manual positive or negative (zero return direction set by bit parameter No.183) feed key, the corresponding axis will move to the zero at a traverse speed.

B: As the approach switch touches the tongue for the first time, the deceleration signal is valid and it slows down immediately to run in a low speed.

C: As the approach switch detaches the tongue, the deceleration signal is invalid, it moves at a fixed low speed after deceleration and starts to detect zero signal (PC).

D: As the approach switch touches the tongue for the second time, the zero signal is valid and the movement stops. The indicator for zero return on the panel lights up.

- Machine zero return type C as servo motor one-rotation signal taken as zero signal
  
  ① Its sketch map is shown below:

  ② Circuit of the deceleration signal
③ Action time sequence of machine zero return

When ZMn (n is X,Y,Z,4th, 5th axis) of the bit parameter No.006 are all set for 1, ZCn (n is X,Y,Z,4th, 5th axis) of the bit parameter No.007 are all set for 0, the BIT5 (DECI) of the bit parameter No.004 is set for 0, and the deceleration signal low level is valid. The action time sequence of machine zero return is shown in follows.

![Diagram of machine zero return]

Fig.2-46

④ Machine zero returns process

A: Select the Machine Zero mode, press manual positive or negative (zero return direction set by bit parameter No.022) feed key, the corresponding axis will move to the machine zero at a traverse speed. Then it touches the tongue and presses down the deceleration switch, and moves forward. When the tongue detaches the deceleration switch, the axis slows down to zero, then moves reversely and accelerates to a fixed low speed for continuous moving.

B: As the tongue touches the deceleration switch for the second time, it moves on till the tongue detaches the deceleration switch. And it begins to detect the zero signals. If the zero signal level changes, the movement stops. Then zero return indicator of the corresponding axis on the panel lights up and machine zero operation is finished.

- Machine zero return type C as an proximity switch is taken as both deceleration and zero signals

① Its sketch map is shown below:
② Circuit of the deceleration signal
See details in Section 2.1.6 of this chapter

③ Action time sequence of machine zero return
When ZMn (n is X,Y,Z,4th,5th axis) of the bit parameter No.006 and ZCn (n is X,Y,Z,4th,5th axis) of the bit parameter No.007 are all set to 1, the BIT5 (DECI) of the bit parameter No.004 is set to 0, the deceleration signal low level is valid. The action time sequence of machine zero return is shown in follows:

④ Machine zero returns process
A: Select the Machine Zero mode, press manual positive or negative (zero return direction is set by bit parameter No.183) feed key, the corresponding axis will move to the machine zero at a traverse speed. Then it touches the tongue and presses down the deceleration switch, and moves forward. When the tongue detaches the deceleration switch, the axis slows down to zero speed, then moves reversely and accelerates to a fixed low speed for continuous moving.
B: As the tongue touches the deceleration switch for the second time, it begins to detect the zero signal. It moves on till the tongue detaches the deceleration switch, the movement stops immediately. Then zero return indicator of the corresponding axis on the panel lights up and machine zero return operation is finished.
CHAPTER 3 PARAMETER

In this chapter the CNC bit and data parameters are introduced. Various functions can be set by these parameters.

3.1 Parameter Description (by sequence)

3.1.1 Bit parameter

The expression of bit parameter is shown in follows:

<table>
<thead>
<tr>
<th>Parameter NO.</th>
<th>BIT7</th>
<th>BIT6</th>
<th>BIT5</th>
<th>BIT4</th>
<th>BIT3</th>
<th>BIT2</th>
<th>BIT1</th>
<th>BIT0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 1</td>
<td>***</td>
<td>***</td>
<td>ACS</td>
<td>HWL</td>
<td>***</td>
<td>***</td>
<td>***</td>
<td>***</td>
</tr>
</tbody>
</table>

ACS = 1: Analog voltage control of spindle speed;
   = 0: Switching control of spindle speed.
HWL = 1: MPG mode;
   = 0: Step mode.

<table>
<thead>
<tr>
<th>Parameter NO.</th>
<th>BIT7</th>
<th>BIT6</th>
<th>BIT5</th>
<th>BIT4</th>
<th>BIT3</th>
<th>BIT2</th>
<th>BIT1</th>
<th>BIT0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 2</td>
<td>***</td>
<td>***</td>
<td>LIFJ</td>
<td>MDITL</td>
<td>LIFC</td>
<td>NRC</td>
<td>TLIF</td>
<td></td>
</tr>
</tbody>
</table>

LIFJ = 1: Tool life management group skip valid;
   = 0: Tool life management group skip invalid.
MDITL = 1: Tool life management valid in MDI mode;
   = 0: Tool life management invalid in MDI mode.
LIFC = 1: Tool life counting type 2, by times;
   = 0: Tool life counting type 1, by times.
NRC = 1: Tool nose radius compensation valid;
   = 0: Tool nose radius compensation invalid.
TLIF = 1: Tool life management valid;
   = 0: Tool life management invalid.

<table>
<thead>
<tr>
<th>Parameter NO.</th>
<th>BIT7</th>
<th>BIT6</th>
<th>BIT5</th>
<th>BIT4</th>
<th>BIT3</th>
<th>BIT2</th>
<th>BIT1</th>
<th>BIT0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 3</td>
<td>PCOM</td>
<td>***</td>
<td>***</td>
<td>***</td>
<td>***</td>
<td>D/R</td>
<td>***</td>
<td></td>
</tr>
</tbody>
</table>

PCOMP = 1: Screw-pitch error compensation valid;
   = 0: Screw-pitch error compensation invalid.
D/R = 1: Tool offset D is diameter value;
   = 0: Tool offset D is radius value.

<table>
<thead>
<tr>
<th>Parameter NO.</th>
<th>BIT7</th>
<th>BIT6</th>
<th>BIT5</th>
<th>BIT4</th>
<th>BIT3</th>
<th>BIT2</th>
<th>BIT1</th>
<th>BIT0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 4</td>
<td>RDRN</td>
<td>DECI</td>
<td>***</td>
<td>PROD</td>
<td>***</td>
<td>***</td>
<td>SCW</td>
<td></td>
</tr>
</tbody>
</table>

RDRN = 1: In G00 dry run mode, speed = feedrate × speed of dry run;
   = 0: G00 speed = rapid override × rapid tranverse speed.
DECI = 1: Deceleration signal high level for machine zero return;
   = 0: Deceleration signal low level for machine zero return.
PROD  =1: Relative coordinate displayed in POSITION page is programming position;
        =0: Relative coordinate displayed in POSITION page involving tool compensation.
SCW  =1: Inch output(inch system)valid after repower;
        =0: Metric output(metric system)valid after repower

The functions of metric and inch system

There are two kinds of input and output units for CNC numerical control system: metric unit, millimeter (mm) and English unit (inch).

Output increment unit is set by Bit0 (SCW) of bit parameter №004 in GSK980MDa system. SCW=0 indicates that minimum command increment, parameter and screw–pitch values are in metric units; SCW=1 indicates that minimum command increment, parameter and screw–pitch values are in inches units. The setting of this parameter depends on machine tool.

G code: By selecting G20/G21 code, it is able to set whether minimum input increment values are in inch or in metric. Executing G21 indicates that minimum input increment values are in metric; and executing G20 indicates that values are in inch.

<table>
<thead>
<tr>
<th>0 0 5</th>
<th>***</th>
<th>***</th>
<th>SMAL</th>
<th>M30</th>
<th>***</th>
<th>***</th>
<th>PPD</th>
<th>PCMD</th>
</tr>
</thead>
</table>
SMAL  =1: Spindle manual gear shift for S command;
        =0: Spindle auto gear shift for S command.
M30   =1: Cursor returns to beginning after M30 execution;
        =0: Cursor not to beginning after M30 execution.
PPD   =1: Relative coordinate set by G92;
        =0: Relative coordinate not set by G92.
PCMD  =1: Axial output wave form is pulse;
        =0: Axial output wave form is square.

Square output, max. output frequency 266KPPS
Pulse output, max. output frequency 266KPPS,
Pulse width 1 μs.

<table>
<thead>
<tr>
<th>0 0 6</th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>ZM5</th>
<th>ZM4</th>
<th>ZMZ</th>
<th>ZMY</th>
<th>ZMX</th>
</tr>
</thead>
</table>
ZM5   =1: 5th zero return type C;
        =0: 5th zero return type B.
ZM4   =1: 4th zero return type C;
        =0: 4th zero return type B.
ZMZ   =1: Z zero return type C;
        =0: Z zero return type B.
ZMY   =1: Y zero return type C;
        =0: Y zero return type B.
ZMX   =1: X zero return type C;
        =0: X zero return type B.
On the condition that blocks smoothing transition is valid, more smooth velocity link and better machining quality will be obtained during the path transition from line to line or from line to arc by properly changing the linear feedrate.

So the actual output speed may be different to the programming speed when using this function. And it may also differ as regard to the linear segment with the same programming speed. The deviation is not more than 15mm/min between the actual output speed and the programming speed on the condition that the programming speed F is less than 1200mm/min.

AVGL  =1: When SMZ=0 linear smoothing is valid, i.e. smoothing transition function is valid;
       =0: Linear smoothing transition function is invalid.

SMZ   =1: To execute next block till all moving blocks executed;
       =0: For smooth transition between blocks.

ZC5   =1: Deceleration signal (DEC5) and one-rotation signal (PC5) of 5th axis are in parallel connection (a proximity switch taken as both deceleration signal and zero signal) during machine zero return;
       =0: Deceleration signal (DEC5) and one-rotation signal (PC5) of 5th axis are connected independently (independent deceleration signal and zero signal are required) during machine zero return.

ZC4   =1: Deceleration signal (DEC4) and one-rotation signal (PC4) of 4th axis are in parallel connection (a proximity switch taken as both deceleration signal and zero signal) during machine zero return;
       =0: Deceleration signal (DEC4) and one-rotation signal (PC4) of 4th axis are connected independently (independent deceleration signal and zero signal are required) during machine zero return.

ZCZ   =1: Deceleration signal (DECZ) and one-rotation signal (PCZ) of Z axis are in parallel connection (a proximity switch taken as both deceleration signal and zero signal) during machine zero return;
       =0: Deceleration signal (DECZ) and one-rotation signal (PCZ) of Z axis are connected independently (independent deceleration signal and zero signal are required) during machine zero return.

ZCY   =1: Deceleration signal (DECY) and one-rotation signal (PCY) of Y axis are in parallel connection (a proximity switch taken as both deceleration signal and zero signal) during machine zero return;
       =0: Deceleration signal (DECY) and one-rotation signal (PCY) of Y axis are connected independently (independent deceleration signal and zero signal are required) during machine zero return.

ZCX   =1: Deceleration signal (DECX) and one-rotation signal (PCX) of X axis are in parallel connection (a proximity switch taken as both deceleration signal and zero signal) during machine zero return;
machine zero return;
=0: Deceleration signal (DECX) and one-rotation signal (PCX) of X axis are connected independently (independent deceleration signal and zero signal are required) during machine zero return.

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>8</th>
<th>DISP</th>
<th>***</th>
<th>***</th>
<th>DIR5</th>
<th>DIR4</th>
<th>DIRZ</th>
<th>DIRY</th>
<th>DIRX</th>
</tr>
</thead>
</table>
| DISP | =1: Enter absolute page after power on;  
=0: Enter relative page after power on. |
| DIR5 | =1: Direction signal (DIR)is high level as 5th axis moves positively;  
=0: Direction signal (DIR)is low level as 5th axis moves negatively. |
| DIR4 | =1: Direction signal (DIR)is high level as 4th axis moves positively;  
=0: Direction signal (DIR)is low level as 4th axis moves negatively. |
| DIRZ | =1: Direction signal (DIR)is high level as Z axis moves positively;  
=0: Direction signal (DIR)is low level as Z axis moves negatively. |
| DIRY | =1: Direction signal (DIR)is high level as Y axis moves positively;  
=0: Direction signal (DIR)is low level as Y axis moves negatively. |
| DIRX | =1: Direction signal (DIR)is high level as X axis moves positively;  
=0: Direction signal (DIR)is low level as X axis moves negatively. |

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>9</th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>ALM5</th>
<th>ALM4</th>
<th>ALMZ</th>
<th>ALMY</th>
<th>ALMX</th>
</tr>
</thead>
</table>
| ALM5 | =1: 5th axis low level alarm signal (ALM5);  
=0: 5th axis high level alarm signal (ALM5). |
| ALM4 | =1: 4th axis low level alarm signal (ALM4);  
=0: 4th axis high level alarm signal (ALM4). |
| ALMZ | =1: Z axis low level alarm signal (ALMZ);  
=0: Z axis high level alarm signal (ALMZ). |
| ALMY | =1: Y axis low level alarm signal (ALMY);  
=0: Y axis high level alarm signal (ALMY). |
| ALMX | =1: X axis low level alarm signal (ALMX);  
=0: X axis high level alarm signal (ALMX). |

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>0</th>
<th>CPF7</th>
<th>CPF6</th>
<th>CPF5</th>
<th>CPF4</th>
<th>CPF3</th>
<th>CPF2</th>
<th>CPF1</th>
<th>CPF0</th>
</tr>
</thead>
</table>
| CPF0~CPF7: Setting values of backlash compensation pulse frequency.  
Set frequency = \( 2^7 \times CPF7 + 2^6 \times CPF6 + 2^5 \times CPF5 + 2^4 \times CPF4 + 2^3 \times CPF3 + 2^2 \times CPF2 + 2^1 \times CPF1 + CPF0 \) Kpps |

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>1</th>
<th>BDEC</th>
<th>BD8</th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>ZNIK</th>
<th>***</th>
<th>***</th>
</tr>
</thead>
</table>
| BDEC | =1: Backlash compensation type B, the compensation data are output by ascending type and the set frequency is invalid.;  
=0: Backlash compensation type A, the compensation data are output by the set frequency (by bit parameter No.010) or 1/8 of it. |
| BD8 | =1: Backlash compensation is done by the 1/8 of the set frequency;  
=0: Backlash compensation is done by the set frequency. |
ZNIK  =1: Direction keys locked during zero return, homing continues to end by pressing direction key once;
    =0: Direction keys unlocked but should be held on during zero return.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>TMANL</th>
<th>***</th>
<th>***</th>
<th>EBCL</th>
<th>ISOT</th>
</tr>
</thead>
</table>

TMANL =1: Manual tool change for T code;
    =0: Auto tool change for T code.

EBCL  =1: Program end sign EOB displays “;”(semicolon);
    =0: Program end sign EOB displays “*”(asterisk).

ISOT  =1: Prior to machine zero return after power on, manual rapid traverse valid;
    =0: Prior to machine zero return after power on, manual rapid traverse invalid.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>3</th>
<th>SCRD</th>
<th>G01</th>
<th>RSCD</th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>SKPI</th>
<th>G31P</th>
</tr>
</thead>
</table>

SCRD  =1: Coordinate system holding on at power down;
    =0: Coordinate system not holding on at power down, G54 coordinate system is set after power on.

G01   =1: G01 status when power on;
    =0: G00 status when power on.

RSCD  =1: G54 coordinate system when reset 4;
    =0: Coordinate system not changed when reset.

SKPI  =1: High level valid for skip signal;
    =0: Low level valid for skip signal.

G31P  =1: G31 immediately stops when skip signal is valid;
    =0: G31 slows down to stop when skip signal is valid.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>4</th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>ZRS5</th>
<th>ZRS4</th>
<th>ZRSZ</th>
<th>ZRSY</th>
<th>ZRX</th>
</tr>
</thead>
</table>

ZRS5  =1: There are machine zero point in 5th axis, it detects deceleration signal and zero signal when performing machine zero return;
    =0: There are no machine zero point in 5th axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

ZRS4  =1: There are machine zero point in 4th axis, it detects deceleration signal and zero signal when performing machine zero return;
    =0: There are no machine zero point in 4th axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

ZRSZ  =1: There are machine zero point in Z axis, it detects deceleration signal and zero signal when performing machine zero return;
    =0: There are no machine zero point in Z axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

ZRSY  =1: There are machine zero point in Y axis, it detects deceleration signal and zero signal when performing machine zero return;
=0: There are no machine zero point in Y axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

ZRSX  =1: There are machine zero point in X axis, it detects deceleration signal and zero signal when performing machine zero return;
=0: There are no machine zero point in X axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPTK</td>
<td>RPTK</td>
<td>NAT</td>
</tr>
</tbody>
</table>

LPTK  =1: Hole locating is done by cutting feed on line continuous drilling;
=0: Hole locating is done by rapid feed on line continuous drilling;

RPTH =1: Hole locating is cutting path in circle and rectangle continuous drilling;
=0: Hole locating is rapid path in circle and rectangle continuous drilling;

NAT   =1 Define the range of user macro program asin, atan;
=0: Not define the range of user macro program asin, atan;

BRCH  =1: Plane returning is selected by G98 and G99 in continuous drilling;
=0: Plane returning is selected by G99 in continuous drilling

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>***</td>
<td>MST</td>
<td>MSP</td>
</tr>
</tbody>
</table>

MST  =1: External cycle start signal (ST) invalid;
=0: External cycle start signal (ST) valid.

MSP  =1: External stop signal (SP) invalid;
=0: External stop signal (SP) valid with external stop switch connected, otherwise CNC shows "stop".

MOT  =1: Not detect software stroke limit;
=0: Detect software stroke limit.

MESP =1: Emergency stop invalid;
=0: Emergency stop valid.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>***</td>
<td>***</td>
<td>***</td>
</tr>
</tbody>
</table>

ESCD =1: S code off at emergency stop;
=0: S code not off at emergency stop.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>KEY1</td>
<td>***</td>
<td>***</td>
</tr>
</tbody>
</table>

KEY1 =1: Prog. switch ON after power on;
=0: Prog. switch OFF after power on.

HNG5 =1: 5th MPG:ccw:+,cw:-;
=0: 5th MPG:ccw:-,cw:+.

HNG4 =1: 4th MPG:ccw:+,cw:-;
=0: 4th MPG:ccw:-,cw:+.

HNGZ =1: Z MPG:ccw:+,cw:-;
=0: Z MPG:ccw:-,cw:+.
**Chapter 3 Parameter**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>2</th>
<th>0</th>
<th>SPFD</th>
<th>SAR</th>
<th>THDA</th>
<th>VAL5</th>
<th>VAL4</th>
<th>VALZ</th>
<th>VALY</th>
<th>VALX</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SPFD = 1</strong></td>
<td>Cutting feed stops if spindle stops;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SPFD = 0</strong></td>
<td>Cutting feed not stop after spindle stop.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SAR = 1</strong></td>
<td>Detect spindle SAR signal prior to cutting;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SAR = 0</strong></td>
<td>Not detect spindle SAR signal prior to cutting.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THDA = 1</strong></td>
<td>Thread machining adopts exponential acceleration and deceleration;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>THDA = 0</strong></td>
<td>Thread machining adopts linear acceleration and deceleration.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VAL5 = 1</strong></td>
<td>For 5th axis move key, ↑ is positive, ↓ is negative;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VAL5 = 0</strong></td>
<td>For 5th axis move key, ↓ is positive, ↑ is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VAL4 = 1</strong></td>
<td>For 4th axis move key, ↑ is positive, ↓ is negative;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VAL4 = 0</strong></td>
<td>For 4th axis move key, ↓ is positive, ↑ is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VALZ = 1</strong></td>
<td>For Z axis move key, ↑ is positive, ↓ is negative;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VALZ = 0</strong></td>
<td>For Z axis move key, ↓ is positive, ↑ is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VALY = 1</strong></td>
<td>For Y axis move key, ↑ is positive, ↓ is negative;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VALY = 0</strong></td>
<td>For Y axis move key, ↓ is positive, ↑ is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VALX = 1</strong></td>
<td>For X axis move key, ← is positive, → is negative;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>VALX = 0</strong></td>
<td>For X axis move key, → is positive, ← is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>2</th>
<th>2</th>
<th>CALH</th>
<th>SOT</th>
<th>***</th>
<th>MZR5</th>
<th>MZR4</th>
<th>MZRZ</th>
<th>MZRY</th>
<th>MZRX</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CALH = 1</strong></td>
<td>Length offset not cancelled in reference point return;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>CALH = 0</strong></td>
<td>Length offset cancelled in reference point return.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SOT = 1</strong></td>
<td>Software limit is valid after zero return at power on;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SOT = 0</strong></td>
<td>Software limit is valid once power on.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZR5 = 1</strong></td>
<td>Machine zero return in negative 5th axis;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZR5 = 0</strong></td>
<td>Machine zero return in positive 5th axis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZR4 = 1</strong></td>
<td>Machine zero return in negative 4th axis;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZR4 = 0</strong></td>
<td>Machine zero return in positive 4th axis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZRZ = 1</strong></td>
<td>Machine zero return in negative Z axis;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZRZ = 0</strong></td>
<td>Machine zero return in positive Z axis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZRY = 1</strong></td>
<td>Machine zero return in negative Y axis;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZRY = 0</strong></td>
<td>Machine zero return in positive Y axis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZRX = 1</strong></td>
<td>Machine zero return in positive X axis;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>MZRX = 0</strong></td>
<td>Machine zero return in negative X axis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>2</th>
<th>5</th>
<th>RTORI</th>
<th>***</th>
<th>RTPCP</th>
<th>***</th>
<th>***</th>
<th>RTCRG</th>
<th>***</th>
<th>***</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RTORI = 1</strong></td>
<td>Spindle performs zero return when M29 is executed;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>RTORI = 0</strong></td>
<td>Spindle does not perform zero return when M29 is executed.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
RTPCP=1:  Rigid tapping is the high-speed deep hole cycle (G73 mode);
  =0:  Rigid tapping is the high-speed deep hole cycle (G83 mode).
RTCRG=1: Do not wait for G61.0 to be 1 as executing next program block after rigid tapping cancelled;
  =0:  Do wait for G61.0 to be 1 as executing next program block after rigid tapping cancelled.

<table>
<thead>
<tr>
<th>0</th>
<th>2</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>A4IS1</td>
<td>A4IS0</td>
<td>RCS4</td>
</tr>
</tbody>
</table>

RCS4 =1:  4th Cs function is valid (power on);
  =0:  4th Cs function is invalid (power on).
Note: Only when the rotary axis function is valid (ROT4=1), can the RCS4 be set valid.

ROS4, ROT4:  Set the type of 4th;

<table>
<thead>
<tr>
<th>Linear</th>
<th>Rotary A</th>
<th>Rotary B</th>
<th>invalid</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROT4</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>ROS4</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

A4IS1, A4IS0: Select increment system of 4th.

<table>
<thead>
<tr>
<th>0</th>
<th>2</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>***</td>
<td>RRT4</td>
<td>***</td>
</tr>
</tbody>
</table>

RRT4 =1:  Zero mode D is used on 4th rotary axis (power on);
  =0:  Zero mode A, B, C are used on 4th rotary axis (power on).

RRL4 =1:  4th rel. coor. cycle func. is valid (power on);
  =0:  4th rel. coor. cycle func. is invalid (power on).

RAB4 =1:  4th rotates according to symbol direction;
  =0:  4th rotates according to nearby rotation.

ROA4 =1:  4th abs. coor. cycle func. is valid (power on);
  =0:  4th abs. coor. cycle func. is invalid (power on).
Note 1: Parameter ROA4 is valid for only rotary axis (ROT4=1),
Note 2: Only parameter ROA4 =1, is RAB4 valid
Note 3: Only parameter ROA4 =1, is RRL4 valid

<table>
<thead>
<tr>
<th>0</th>
<th>2</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>A5IS1</td>
<td>A5IS0</td>
<td>RCS5</td>
</tr>
</tbody>
</table>

RCS5 =1:  5th Cs function is valid (power on);
  =0:  5th Cs function is invalid (power on).
Note: Only rotary axis function is valid (ROT5=1), is RCS5 valid.

ROS5, ROT5:  Set the type of 5th;
A5IS1, A5IS0: Selecte increment system of 5th.

<table>
<thead>
<tr>
<th>A5IS1</th>
<th>A5IS0</th>
<th>Increment System of 5TH</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Same to the X, Y, Z</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>IS-A</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>IS-B</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>IS-C</td>
</tr>
</tbody>
</table>

RRT5 = 1: Zero mode D is used on 5th rotary axis (power on);
= 0: Zero mode A, B, C are used on 5th rotary axis (power on).

RRL5 = 1: 5th rel. coor. cycle func. is valid (power on);
= 0: 5th rel. coor. cycle func. is invalid (power on).

RAB5 = 1: 5th rotates according to symbol direction;
= 0: 5th rotates according to nearby rotation.

ROA5 = 1: 5th abs. coor. cycle func. is valid (power on);
= 0: 5th abs. coor. cycle func. is invalid (power on).

Note 1: ROA5 is valid to only rotary axis (ROT5 = 1);
Note 2: Only when parameter ROA4 = 1, is RAB4 valid;
Note 3: Only when parameter ROA4 = 1, is RRL4 valid;

ISC = 1: Minimum increment system is IS-C (need restart);
= 0: Minimum increment system is IS-B (do not need restart).

ABPx = 1: Output axis pulse by two right-angle intersection phases (need restart);
= 0: Output axis pulse by pulse and direction (do not need restart).

L2, L1, L0: Interface language selection:
### 3.1.2 Data parameter

<table>
<thead>
<tr>
<th>Language</th>
<th>L2</th>
<th>L1</th>
<th>L0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chinese</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>English</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>French</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Spanish</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>German</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Italian</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Russian</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Korean</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting range: 1~32767</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMRX</td>
<td>X axis multiplier coefficient</td>
</tr>
<tr>
<td>CMRY</td>
<td>Y axis multiplier coefficient</td>
</tr>
<tr>
<td>CMRZ</td>
<td>Z axis multiplier coefficient</td>
</tr>
<tr>
<td>CMR4</td>
<td>4th axis multiplier coefficient</td>
</tr>
<tr>
<td>CMR5</td>
<td>5th axis multiplier coefficient</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting range: 1~32767</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMDX</td>
<td>X axis frequency division coefficient</td>
</tr>
<tr>
<td>CMDY</td>
<td>Y axis frequency division coefficient</td>
</tr>
<tr>
<td>CMDZ</td>
<td>Z axis frequency division coefficient</td>
</tr>
<tr>
<td>CMD4</td>
<td>4th axis frequency division coefficient</td>
</tr>
<tr>
<td>CMD5</td>
<td>5th axis frequency division coefficient</td>
</tr>
</tbody>
</table>

Electronic gear ratio formula:

\[
\frac{CMR}{CMD} = \frac{S \times 360 \times Z_M}{\alpha \times L \times Z_D}
\]

- \(S\): min. command output unit
- \(\alpha\): motor rotation angle for a pulse
- \(Z_M\): belt wheel teeth of lead screw
- \(Z_D\): Wheel teeth of motor belt
- \(L\): Screw lead

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting range: 10~99999999 (Unit: mm/min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 5 9</td>
<td>X axis max. rapid traverse speed</td>
</tr>
<tr>
<td>0 6 0</td>
<td>Y axis max. rapid traverse speed</td>
</tr>
<tr>
<td>0 6 1</td>
<td>Z axis max. rapid traverse speed</td>
</tr>
<tr>
<td>0 6 2</td>
<td>4th axis max. rapid traverse speed</td>
</tr>
<tr>
<td>0 6 3</td>
<td>5th axis max. rapid traverse speed</td>
</tr>
</tbody>
</table>
### Chapter 3 Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting Range</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 6 4</td>
<td>Acceleration &amp; deceleration time constant of X axis rapid traverse (ms)</td>
<td>10～4000 (Unit: ms)</td>
</tr>
<tr>
<td>0 6 5</td>
<td>Acceleration &amp; deceleration time constant of Y axis rapid traverse (ms)</td>
<td></td>
</tr>
<tr>
<td>0 6 6</td>
<td>Acceleration &amp; deceleration time constant of Z axis rapid traverse (ms)</td>
<td></td>
</tr>
<tr>
<td>0 6 7</td>
<td>Acceleration &amp; deceleration time constant of 4th axis rapid traverse (ms)</td>
<td></td>
</tr>
<tr>
<td>0 6 8</td>
<td>Acceleration &amp; deceleration time constant of 5th axis rapid traverse (ms)</td>
<td></td>
</tr>
<tr>
<td>0 6 9</td>
<td>Rapid traverse speed when rapid override is F0</td>
<td>6～4000 (Unit: mm/min)</td>
</tr>
<tr>
<td>0 7 0</td>
<td>Axes top feedrate of cutting</td>
<td>10～4000 (Unit: mm/min)</td>
</tr>
<tr>
<td>0 7 1</td>
<td>Exponential acceleration start speed and deceleration end speed in cutting feed</td>
<td>0～8000 (Unit: mm/min)</td>
</tr>
<tr>
<td>0 7 2</td>
<td>Exponential acceleration &amp; deceleration time constant of cutting</td>
<td>10～4000 (Unit: ms)</td>
</tr>
<tr>
<td>0 7 3</td>
<td>Start speed in manual feed</td>
<td>0～8000 (Unit: mm/min)</td>
</tr>
<tr>
<td>0 7 4</td>
<td>Exponential acceleration &amp; deceleration time constant of manual feed</td>
<td>10～4000 (Unit: ms)</td>
</tr>
<tr>
<td>0 7 5</td>
<td>Threading axes start speed</td>
<td>6～8000 (Unit: mm/min)</td>
</tr>
<tr>
<td>0 7 7</td>
<td>Initial speed of acc. &amp; dec. speed of CS axis</td>
<td>0～5000 (Unit: deg/min)</td>
</tr>
<tr>
<td>0 7 8</td>
<td>Acc. &amp; dec. time constant of CS axis</td>
<td>10～10000 (Unit: ms)</td>
</tr>
<tr>
<td>Parameter</td>
<td>Setting Range</td>
<td>Unit</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>0 8 1</td>
<td>Initial speed of linear acceleration/deceleration in rigid tapping</td>
<td>0~5000 (Unit: mm/min)</td>
</tr>
<tr>
<td>0 8 2</td>
<td>Linear acc.&amp;dec. time constant in rigid tapping tool infeed</td>
<td>10~10000 (Unit: ms)</td>
</tr>
<tr>
<td>0 8 3</td>
<td>Linear acc.&amp;dec. time constant in rigid tapping tool retract</td>
<td>0~4000 (Unit: ms), 082 setting value is used when it is set to 0.</td>
</tr>
<tr>
<td>0 8 4</td>
<td>Override value in rigid tapping tool retract</td>
<td>0~200, 0: override is set to 100%</td>
</tr>
<tr>
<td>0 8 5</td>
<td>Tool retract amount in deep hole rigid tapping (high-speed, standard)</td>
<td>0~32767000 (Unit: 0.001 mm)</td>
</tr>
<tr>
<td>0 8 9, 0 9 0, 0 9 1, 0 9 2, 0 9 3</td>
<td>Low speed of X axis machine zero return, Low speed of Y axis machine zero return, Low speed of Z axis machine zero return, Low speed of 4th axis machine zero return, Low speed of 5th axis machine zero return</td>
<td>10~1000 (Unit: mm/min)</td>
</tr>
<tr>
<td>0 9 4, 0 9 5, 0 9 6, 0 9 7, 0 9 8</td>
<td>High speed of X axis machine zero return, High speed of Y axis machine zero return, High speed of Z axis machine zero return, High speed of 4th axis machine zero return, High speed of 5th axis machine zero return</td>
<td>10~921571875 (Unit: mm/min)</td>
</tr>
<tr>
<td>0 9 9</td>
<td>Voltage compensation for 0V analog voltage output</td>
<td>-1000~1000 (Unit: mV)</td>
</tr>
<tr>
<td>1 0 0</td>
<td>Voltage offset value when spindle max. speed analog voltage 10V output</td>
<td>-2000~2000 (Unit: mV)</td>
</tr>
<tr>
<td>1 0 1, 1 0 2, 1 0 3, 1 0 4</td>
<td>Max spindle speed of 1st gear when analog voltage output is 10V, Max spindle speed of 2nd gear when analog voltage output is 10V, Max spindle speed of 3rd gear when analog voltage output is 10V, Max spindle speed of 4th gear when analog voltage output is 10V</td>
<td>10~9999 (Unit: r/min)</td>
</tr>
</tbody>
</table>
### Chapter 3 Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>107</td>
<td>0~4080 (Unit: ms)</td>
<td>Spindle speed reaches to signal detection delay time</td>
</tr>
<tr>
<td>108</td>
<td>50~1000 (Unit: r/min)</td>
<td>Max. spindle speed fluctuation allowed by system</td>
</tr>
<tr>
<td>109</td>
<td>0~5000 (Unit: p/r)</td>
<td>Spindle encoder pulses</td>
</tr>
<tr>
<td>110</td>
<td></td>
<td>It is drilling holes when 0 indicates G74 and G84 cycle.</td>
</tr>
<tr>
<td>111</td>
<td>1~255</td>
<td>Transmission ratio of encoder and spindle- spindle gear teeth</td>
</tr>
<tr>
<td>112</td>
<td>1~255</td>
<td>Transmission ratio of encoder and spindle- encoder gear teeth</td>
</tr>
<tr>
<td>113</td>
<td>0~2000 (Unit: 0.001mm)</td>
<td>X axis backlash offset</td>
</tr>
<tr>
<td>114</td>
<td>0~2000 (Unit: 0.001mm)</td>
<td>Y axis backlash offset</td>
</tr>
<tr>
<td>115</td>
<td>0~2000 (Unit: 0.001mm)</td>
<td>Z axis backlash offset</td>
</tr>
<tr>
<td>116</td>
<td>0~2000 (Unit: 0.001mm)</td>
<td>4th axis backlash offset</td>
</tr>
<tr>
<td>117</td>
<td>0~2000 (Unit: 0.001mm)</td>
<td>5th axis backlash offset</td>
</tr>
<tr>
<td>118</td>
<td>10000~999999 (Unit: 0.001mm)</td>
<td>Interval of X axis screw-pitch error compensation</td>
</tr>
<tr>
<td>119</td>
<td>10000~999999 (Unit: 0.001mm)</td>
<td>Interval of Y axis screw-pitch error compensation</td>
</tr>
<tr>
<td>120</td>
<td>10000~999999 (Unit: 0.001mm)</td>
<td>Interval of Z axis screw-pitch error compensation</td>
</tr>
<tr>
<td>121</td>
<td>10000~999999 (Unit: 0.001mm)</td>
<td>Interval of 4th axis screw-pitch error compensation</td>
</tr>
<tr>
<td>122</td>
<td>10000~999999 (Unit: 0.001mm)</td>
<td>Interval of 5th axis screw-pitch error compensation</td>
</tr>
<tr>
<td>123</td>
<td>0~255</td>
<td>Screw-pitch error compensation position number of X axis machine zero</td>
</tr>
<tr>
<td>124</td>
<td>0~255</td>
<td>Screw-pitch error compensation position number of Y axis machine zero</td>
</tr>
<tr>
<td>125</td>
<td>0~255</td>
<td>Screw-pitch error compensation position number of Z axis machine zero</td>
</tr>
<tr>
<td>126</td>
<td>0~255</td>
<td>Screw-pitch error compensation position number of 4th axis machine zero</td>
</tr>
<tr>
<td>127</td>
<td>0~255</td>
<td>Screw-pitch error compensation position number of 5th axis machine zero</td>
</tr>
<tr>
<td>128</td>
<td>-99999~99999 (Unit: 0.001mm)</td>
<td>X axis machine zero offset</td>
</tr>
<tr>
<td>129</td>
<td>-99999~99999 (Unit: 0.001mm)</td>
<td>Y axis machine zero offset</td>
</tr>
<tr>
<td>130</td>
<td>-99999~99999 (Unit: 0.001mm)</td>
<td>Z axis machine zero offset</td>
</tr>
<tr>
<td>131</td>
<td>-99999~99999 (Unit: 0.001mm)</td>
<td>4th axis machine zero offset</td>
</tr>
<tr>
<td>132</td>
<td>-99999~99999 (Unit: 0.001mm)</td>
<td>5th axis machine zero offset</td>
</tr>
</tbody>
</table>
### Installation

| 1 3 5 | Max. X coordinate value of software limit |
| 1 3 6 | Max. Y coordinate value of software limit |
| 1 3 7 | Max. Z coordinate value of software limit |
| 1 3 8 | Max. 4th coordinate value of software limit |
| 1 3 9 | Max. 5th coordinate value of software limit |
| 1 4 0 | Min. X coordinate value of software limit |
| 1 4 1 | Min. Y coordinate value of software limit |
| 1 4 2 | Min. Z coordinate value of software limit |
| 1 4 3 | Min. 4th coordinate value of software limit |
| 1 4 4 | Min. 5th coordinate value of software limit |

Setting range: -9999999~+9999999 (Unit:0.001mm)

| 1 4 5 | X machine coordinate of 1st reference point |
| 1 4 6 | Y machine coordinate of 1st reference point |
| 1 4 7 | Z machine coordinate of 1st reference point |
| 1 4 8 | 4th machine coordinate of 1st reference point |
| 1 4 9 | 5th machine coordinate of 1st reference point |
| 1 5 0 | X machine coordinate of 2nd reference point |
| 1 5 1 | Y machine coordinate of 2nd reference point |
| 1 5 2 | Z machine coordinate of 2nd reference point |
| 1 5 3 | 4th machine coordinate of 2nd reference point |
| 1 5 4 | 5th machine coordinate of 2nd reference point |
| 1 5 5 | X machine coordinate of 3rd reference point |
| 1 5 6 | Y machine coordinate of 3rd reference point |
| 1 5 7 | Z machine coordinate of 3rd reference point |
| 1 5 8 | 4th machine coordinate of 3rd reference point |
| 1 5 9 | 5th machine coordinate of 3rd reference point |
| 1 6 0 | X machine coordinate of 4th reference point |
| 1 6 1 | Y machine coordinate of 4th reference point |
| 1 6 2 | Z machine coordinate of 4th reference point |
| 1 6 3 | 4th machine coordinate of 4th reference point |
| 1 6 4 | 5th machine coordinate of 4th reference point |

Setting range: -9999999~+9999999 (Unit:0.001mm)

| 1 7 2 | Initial value of cutting feedrate when power on |

Setting range: 10~15000 (Unit:mm/min)

| 1 7 4 | Feedrate of dry run |

Setting range: 10~99999999 (Unit:mm/min)
### Arc radius error limit

Setting range: \(0 \sim 1000\) (Unit:0.001mm), On arc code \((G02,G03)\), if error exceeds the difference executing limit between initial point radius and end point radius, alarm will be issued.

### Retraction amount of G73 high deep hole drilling cycle

Setting range: \(0 \sim 32767000\) (Unit:0.001mm),

### Cutting initial point of G83 high deep hole drilling cycle

Setting range: \(0 \sim 32767000\) (Unit:0.001mm),

### G110,G111,G134,G135  Lead of helical tool infeed

Setting range: \(0 \sim 999999\) (unit 0.001mm)

If setting value is less than 10, helical feeding is invalid for rough milling command G110, G111, G134, G135, and it feeds by linear type.

If setting value is more than or equal to 10, it feeds by helical type for rough milling command G110, G111, G134, G135.

Rough milling command \((G110,G111,134,G135)\) helical feed function:

Namely, for Z axis depth cutting of rough milling command G110, G111, 134, G135, the tool feeds not by linear type, but by helical type. So the workpiece with no groove may be rough milled directly.

**Note 1** when the Z axis cutting depth is less than \(10\mu m\) each time, the helical feeding is invalid.

**Note 2** when the tool radius is less than \(1mm\), the helical feeding is also invalid.

The helical feeding path is shown in follows:
### 1.8.9
Movement per rotation of the 4th axis
Movement per rotation of the 5th axis

Setting range: 1~9999999 (unit: 0.001deg)

### 2.0.1
Allowed valid ey number at the same time

Setting range: 2~5

### 2.0.2
Define the name of the 4th axis (A:65, B:66, C:67)
Define the name of the 5th axis (A:65, B:66, C:67)

Setting range: 65~67 65-A, 66-B, 67-C

### 2.1.3
Total tool number selection

Setting range: 1~32

### 2.1.4
Reset output time

Setting range: 16~4080 (unit: ms)

### 2.1.5
Serial communication baudrate

Setting range: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 (unit: bit/s)

### 2.1.6
Block No. increment for block No. auto insertion

Setting range: 1~100

### 3.2 Parameter description (by function sequence)

#### 3.2.1 Axis control logic

<table>
<thead>
<tr>
<th>0 0 0 8</th>
<th>DISP</th>
<th>***</th>
<th>***</th>
<th>DIR5</th>
<th>DIR4</th>
<th>DIRZ</th>
<th>DIRY</th>
<th>DIRX</th>
</tr>
</thead>
</table>
| DIR5    | =1:  | Direction signal (DIR) is high level as the 5th axis moves positively;  
               | =0:  | Direction signal (DIR) is low level as the 5th axis moves negatively.  
| DIR4    | =1:  | Direction signal (DIR) is high level as the 4th axis moves positively;  
               | =0:  | Direction signal (DIR) is low level as the 4th axis moves negatively.  
| DIRZ    | =1:  | Direction signal (DIR) is high level as Z axis moves positively;  
               | =0:  | Direction signal (DIR) is low level as Z axis moves negatively.  
| DIRY    | =1:  | Direction signal (DIR) is high level as Y axis moves positively;  
               | =0:  | Direction signal (DIR) is low level as Y axis moves negatively.  
| DIRX    | =1:  | Direction signal (DIR) is high level as X axis moves positively;  
               | =0:  | Direction signal (DIR) is low level as X axis moves negatively.  

---

**Volume III Installation**
<table>
<thead>
<tr>
<th>0 0 9</th>
<th>***</th>
<th>***</th>
<th>ALM5</th>
<th>ALM4</th>
<th>ALMZ</th>
<th>ALMY</th>
<th>ALMX</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALM5</td>
<td>=1:</td>
<td>the 5th axis low level alarm signal (ALM5);</td>
<td>=0:</td>
<td>the 5th axis high level alarm signal (ALM5).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALM4</td>
<td>=1:</td>
<td>the 4th axis low level alarm signal (ALM4);</td>
<td>=0:</td>
<td>the 4th axis high level alarm signal (ALM4).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALMZ</td>
<td>=1:</td>
<td>Z axis low level alarm signal (ALMZ);</td>
<td>=0:</td>
<td>Z axis high level alarm signal (ALMZ).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALMY</td>
<td>=1:</td>
<td>Y axis low level alarm signal (ALMY);</td>
<td>=0:</td>
<td>Y axis high level alarm signal (ALMY).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALMX</td>
<td>=1:</td>
<td>X axis low level alarm signal (ALMX);</td>
<td>=0:</td>
<td>X axis high level alarm signal (ALM).</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>0 1 9</th>
<th>KEY1</th>
<th>***</th>
<th>HNG5</th>
<th>HNG4</th>
<th>HNGZ</th>
<th>HNGY</th>
<th>HNGX</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>0 2 0</th>
<th>SPFD</th>
<th>SAR</th>
<th>THDA</th>
<th>VAL5</th>
<th>VAL4</th>
<th>VALZ</th>
<th>VALY</th>
<th>VALX</th>
</tr>
</thead>
<tbody>
<tr>
<td>VAL5</td>
<td>=1:</td>
<td>For the 5th axis move key, ↑ is positive. ↓ is negative;</td>
<td>=0:</td>
<td>For the 5th axis move key, ↓ is positive. ↑ is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAL4</td>
<td>=1:</td>
<td>For the 4th axis move key, ↑ is positive. ↓ is negative;</td>
<td>=0:</td>
<td>For the 4th axis move key, ↓ is positive. ↑ is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VALZ</td>
<td>=1:</td>
<td>For Z axis move key, ↑ is positive. ↓ is negative;</td>
<td>=0:</td>
<td>For Z axis move key, ↓ is positive. ↑ is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VALY</td>
<td>=1:</td>
<td>For Y axis move key, ↑ is positive. ↓ is negative;</td>
<td>=0:</td>
<td>For Y axis move key, ↓ is positive. ↑ is negative.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VALX</td>
<td>=1:</td>
<td>For X axis move key, → is positive. ← is negative;</td>
<td>=0:</td>
<td>For X axis move key, ← is positive. → is negative</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### 3.2.2 Acceleration & deceleration control

RDRN  
=1:  G00 rapid traverse, speed = federate × dry run speed;  
=0:  G00 speed = rapid override × rapid traverse speed .  

ISOT  
=1:  Prior to machine zero return after power on, manual rapid traverse valid;  
=0:  Prior to machine zero return after power on, manual rapid traverse invalid.

### Electronic gear ratio formula:

\[
\frac{CMR}{CMD} = \frac{S \times 360}{\alpha \times L} \times \frac{Z_M}{Z_D}
\]

- **S**: Min. command output unit  
- **Z_M**: belt wheel teeth of lead screw  
- **Z_D**: Wheel teeth of motor belt  
- **L**: Screw lead

### Setup for axis and frequency division coefficients:

<table>
<thead>
<tr>
<th>Setting range: 1~32767</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMRX: X axis multiplier coefficient</td>
</tr>
<tr>
<td>CMRY: Y axis multiplier coefficient</td>
</tr>
<tr>
<td>CMRZ: Z axis multiplier coefficient</td>
</tr>
<tr>
<td>CMR4: 4th axis multiplier coefficient</td>
</tr>
<tr>
<td>CMR5: 5th axis multiplier coefficient</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Setting range: 1~32767</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMDX: X axis frequency division coefficient</td>
</tr>
<tr>
<td>CMDY: Y axis frequency division coefficient</td>
</tr>
<tr>
<td>CMDZ: Z axis frequency division coefficient</td>
</tr>
<tr>
<td>CMD4: 4th axis frequency division coefficient</td>
</tr>
<tr>
<td>CMD5: 5th axis frequency division coefficient</td>
</tr>
</tbody>
</table>

### Max. rapid traverse speeds:

<table>
<thead>
<tr>
<th>Setting range: 10~1843143750 (unit: mm/min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X axis max. rapid traverse speed</td>
</tr>
<tr>
<td>Y axis max. rapid traverse speed</td>
</tr>
<tr>
<td>Z axis max. rapid traverse speed</td>
</tr>
<tr>
<td>4th axis max. rapid traverse speed</td>
</tr>
<tr>
<td>5th axis max. rapid traverse speed</td>
</tr>
</tbody>
</table>
### Chapter 3 Parameter

#### 3.2.3 Machine protection

<table>
<thead>
<tr>
<th>0 1 7</th>
<th>***</th>
<th>MST</th>
<th>MSP</th>
<th>MOT</th>
<th>MESP</th>
<th>***</th>
<th>***</th>
<th>***</th>
</tr>
</thead>
</table>
| MST  | =1: External cycle start signal (ST) invalid.  
      | =0: External cycle start signal (ST) valid.  
| MSP  | =1: External stop signal (SP) invalid.  
      | =0: External stop signal (SP) valid with external stop switch connected, otherwise CNC shows “stop”.  
| MOT  | =1: Not detect software stroke limit;  
      | =0: Detect software stroke limit.  
| MESP | =1: Emergency stop invalid;  
      | =0: Emergency stop valid |

<table>
<thead>
<tr>
<th>0 1 8</th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>ESCD</th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>***</th>
</tr>
</thead>
</table>
| ESCD | =1: S code off at emergency stop;  
      | =0: S code not off at emergency stop |
3.2.4 Thread function

<table>
<thead>
<tr>
<th>0 2 0</th>
<th>SPFD SAR THDA VAL5 VAL4 VALZ VALY VALX</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>THDA =1: Threading machining adopts exponential acceleration and deceleration;</td>
</tr>
<tr>
<td></td>
<td>=0: Threading machining adopts linear acceleration and deceleration.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>0 7 5</th>
<th>Threading axes start speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting range: 6~8000 (unit:mm/min)</td>
<td></td>
</tr>
</tbody>
</table>

3.2.5 Spindle control

<table>
<thead>
<tr>
<th>0 0 1</th>
<th>*** *** ACS HWL *** *** ***</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACS =1: Analog voltage control of spindle speed;</td>
<td></td>
</tr>
<tr>
<td>=0: Switching control of spindle speed.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>0 9 9</th>
<th>Voltage compensation for 0V analog voltage output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting range: -1000~1000 (unit:mV)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1 0 0</th>
<th>Voltage offset value when spindle max. speed analog voltage 10V output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting range: -2000~2000 (unit: mV)</td>
<td></td>
</tr>
</tbody>
</table>
### 3.2.6 Tool function

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 2</td>
<td>LIFJ, MDITL, LIFC, NRC, TLIF</td>
</tr>
</tbody>
</table>

**LIFJ**
- 1: Tool life management group skip valid;
- 0: Tool life management group skip invalid.

**MDITL**
- 1: Tool life management valid in MDI mode;
- 0: Tool life management invalid in MDI mode.

**LIFC**
- 1: Tool life counting type 2 by times;
- 0: Tool life counting type 1 by times.

**NRC**
- 1: Tool nose radius compensation valid;
- 0: Tool nose radius compensation invalid.

**TLIF**
- 1: Tool life management valid;
- 0: Tool life management invalid

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 2</td>
<td>TMANL, TMAN</td>
</tr>
</tbody>
</table>

**TMANL**
- 1: Manual tool change for T code;
- 0: Auto tool change for T code.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 1 3</td>
<td>Total tool number selection</td>
</tr>
</tbody>
</table>

Setting range: 10  9999  (unit: r/min)

Setting range: 0  4080  (unit: ms)

Setting range: 50  1000  (unit: r/min)

Setting range: 0 5000  (unit: p/r) 0: Not detect spindle encoder in G74, G84 tapping.

Setting range: 1  255
3.2.7 Edit and Display

<table>
<thead>
<tr>
<th>PROD</th>
<th>RDRN</th>
<th>DEC1</th>
<th>PROD</th>
<th>SCW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

PROD =1: Relative coordinate displayed in POSITION page is programming position;

=0: Relative coordinate displayed in POSITION page is position involving tool offset.

<table>
<thead>
<tr>
<th>DISP</th>
<th>DIR5</th>
<th>DIR4</th>
<th>DIRZ</th>
<th>DIRY</th>
<th>DIRX</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DISP =1: Enter absolute page after power on;

=0: Enter relative page after power on.

<table>
<thead>
<tr>
<th>EBCL</th>
<th>TMANL</th>
<th>ISOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

EBCL =1: Program end sign EOB displays “;” (semicolon);

=0: Program end sign EOB displays “*” (asterisk).

<table>
<thead>
<tr>
<th>L2</th>
<th>L1</th>
<th>L0</th>
</tr>
</thead>
</table>

L2, L1, L0: Interface language selection;

<table>
<thead>
<tr>
<th>Language</th>
<th>L2</th>
<th>L1</th>
<th>L0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chinese</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>English</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>French</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Spanish</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>German</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Italian</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Russian</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Korean</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

3.2.8 Precision compensation

<table>
<thead>
<tr>
<th>PCOMP</th>
<th>D/R</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

PCOMP =1: Screw-pitch error compensation valid;

=0: Screw-pitch error compensation invalid.

D/R =1: Tool offset D value is diameter input;

=0: Tool offset D value is radius input.

<table>
<thead>
<tr>
<th>CPF7</th>
<th>CPF6</th>
<th>CPF5</th>
<th>CPF4</th>
<th>CPF3</th>
<th>CPF2</th>
<th>CPF1</th>
<th>CPF0</th>
</tr>
</thead>
</table>

CPF0~CPF7: Setting values of backlash compensation pulse frequency.

The set frequency = 

\[ (2^7 \times CPF7 + 2^5 \times CPF6 + 2^5 \times CPF5 + 2^4 \times CPF4 + 2^3 \times CPF3 + 2^2 \times CPF2 + 2^1 \times CPF1 + CPF0) \text{ Kpps} \]
<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>1</th>
<th><strong>BDEC</strong></th>
<th><strong>BD8</strong></th>
<th>***</th>
<th>***</th>
<th>***</th>
<th>ZNIK</th>
<th>***</th>
<th>***</th>
</tr>
</thead>
</table>

**BDEC** = 1: Backlash compensation type B, the compensation data are output by ascending or descending type and the set frequency is invalid.;

= 0: Backlash compensation type A, the compensation data are output by the set frequency (set by bit parameter No.010) or 1/8 of it.

**BD8** = 1: Backlash compensation is done by the 1/8 of the set frequency;

= 0: Backlash compensation is done by the set frequency.

<table>
<thead>
<tr>
<th>0</th>
<th>2</th>
<th>2</th>
<th><strong>CALH</strong></th>
<th><strong>SOT</strong></th>
<th>***</th>
<th><strong>MZR5</strong></th>
<th><strong>MZR4</strong></th>
<th><strong>MZR2</strong></th>
<th><strong>MZRY</strong></th>
<th><strong>MZRX</strong></th>
</tr>
</thead>
</table>

**CALH** = 1: Length offset not cancel in reference point return;

= 0: Length offset cancel in reference point return.

| 1 | 1 | 5 | X axis backlash offset |
| 1 | 1 | 6 | Y axis backlash offset |
| 1 | 1 | 7 | Z axis backlash offset |
| 1 | 1 | 8 | 4th axis backlash offset |
| 1 | 1 | 9 | 5th axis backlash offset |

Setting range: 0 ～ 2000 (unit: 0.001 mm)

| 1 | 2 | 0 | Interval of X axis screw-pitch error compensation |
| 1 | 2 | 1 | Interval of Y axis screw-pitch error compensation |
| 1 | 2 | 2 | Interval of Z axis screw-pitch error compensation |
| 1 | 2 | 3 | Interval of 4th axis screw-pitch error compensation |
| 1 | 2 | 4 | Interval of 5th axis screw-pitch error compensation |

Setting range: 1000 ～ 999999 (unit: 0.001 mm)

| 1 | 2 | 5 | Screw-pitch error compensation number of X axis machine zero |
| 1 | 2 | 6 | Screw-pitch error compensation number of Y axis machine zero |
| 1 | 2 | 7 | Screw-pitch error compensation number of Z axis machine zero |
| 1 | 2 | 8 | Screw-pitch error compensation number of the 4th axis machine zero |
| 1 | 2 | 9 | Screw-pitch error compensation number of the 5th axis machine zero |

Setting range: 0 ～ 255

### 3.2.9 Communication setting

<table>
<thead>
<tr>
<th>2</th>
<th>1</th>
<th>5</th>
<th>Serial communication baudrate</th>
</tr>
</thead>
</table>

Setting range: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 (unit: bit/s)
3.2.10 Machine zero return

```
0 0 4
*** RDRN DECI *** PROD *** *** SCW
```

DECI = 1: Deceleration signal high level for machine zero return;
= 0: Deceleration signal low level for machine zero return.

```
0 1 1
BDEC BD8 *** *** *** ZNIK *** ***
```

ZNIK = 1: Direction keys locked during zero return, homing continues to end by pressing direction key once;
= 0: Direction keys unlocked but should be held on during zero return.

```
0 0 6
*** *** *** ZM5 ZM4 ZMZ ZMY ZMX
```

ZM5 = 1: 5th zero return type C;
= 0: 5th zero return type B.

ZM4 = 1: 4th zero return type C;
= 0: 4th zero return type B.

ZMZ = 1: Z zero return type C;
= 0: Z zero return type B.

ZMY = 1: Y zero return type C;
= 0: Y zero return type B.

ZMX = 1: X zero return type C;
= 0: X zero return type B.

```
0 0 7
AVGL *** SMZ ZC5 ZC4 ZCZ ZCY ZCX
```

ZC5 = 1: The deceleration signal (DEC5) and one-rotation signal (PC5) of 5th axis in parallel connection (a proximity switch acting as both the deceleration signal and zero signal) during machine zero return;
= 0: The deceleration signal (DEC5) and one-rotation signal (PC5) of 5th axis are connected independently (the indepent deceleration signal and zero signal are required) during machine zero return.

ZC4 = 1: The deceleration signal (DEC4) and one-rotation signal (PC4) of 4th axis in parallel connection (a proximity switch acting as both the deceleration signal and zero signal) during machine zero return;
= 0: The deceleration signal (DEC4) and one-rotation signal (PC4) of 4th axis are connected independently (the indepent deceleration signal and zero signal are required) during machine zero return.

ZCZ = 1: The deceleration signal (DECZ) and one-rotation signal (PCZ) of Z axis in parallel connection (a proximity switch acting as both the deceleration signal and zero signal)
during machine zero return;
=0: The deceleration signal (DECZ) and one-rotation signal (PCZ) of Z axis are connected independently (the independent deceleration signal and zero signal are required) during machine zero return.

\[ ZCY = 1 \]: The deceleration signal (DECY) and one-rotation signal (PCY) of Y axis in parallel connection (a proximity switch acting as both the deceleration signal and zero signal) during machine zero return;
=0: The deceleration signal (DECY) and one-rotation signal (PCY) of Y axis are connected independently (the independent deceleration signal and zero signal are required) during machine zero return.

\[ ZCX = 1 \]: The deceleration signal (DECX) and one-rotation signal (PCX) of X axis in parallel connection (a proximity switch acting as both the deceleration signal and zero signal) during machine zero return;
=0: The deceleration signal (DECX) and one-rotation signal (PCX) of X axis are connected independently (the independent deceleration signal and zero signal are required) during machine zero return.

\[ ZRS5 \]
\[ ZRS4 \]
\[ ZRSZ \]
\[ ZRSY \]
\[ ZRSX \]

\[ ZRS5 = 1 \]: There are machine zero point in the 5th axis, it detects deceleration signal and zero signal when performing machine zero return;
=0: There are no machine zero point in the 5th axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

\[ ZRS4 = 1 \]: There are machine zero point in the 4th axis, it detects deceleration signal and zero signal when performing machine zero return;
=0: There are no machine zero point in the 4th axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

\[ ZRSZ = 1 \]: There are machine zero point in Z axis, it detects deceleration signal and zero signal when performing machine zero return;
=0: There are no machine zero point in Z axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

\[ ZRSY = 1 \]: There are machine zero point in Y axis, it detects deceleration signal and zero signal when performing machine zero return;
=0: There are no machine zero point in Y axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.

\[ ZRSX = 1 \]: There are machine zero point in X axis, it detects deceleration signal and zero signal when performing machine zero return;
=0: There are no machine zero point in X axis, it returns to machine zero without detecting deceleration signal and zero signal when performing machine zero return.
<table>
<thead>
<tr>
<th>CALH</th>
<th>SOT</th>
<th>MZR5</th>
<th>MZR4</th>
<th>MZRZ</th>
<th>MZRY</th>
<th>MZRX</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 2 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**CALH =1:** Length offset not cancel in reference point return;
- 0: Length offset cancel in reference point return.

**MZR5 =1:** Machine zero return in negative the 5th axis;
- 0: Machine zero return in positive the 5th axis.

**MZR4 =1:** Machine zero return in negative the 4th axis;
- 0: Machine zero return in positive the 4th axis.

**MZRZ =1:** Machine zero return in negative Z axis;
- 0: Machine zero return in positive Z axis.

**MZRY =1:** Machine zero return in negative Y axis;
- 0: Machine zero return in positive Y axis.

**MZRX =1:** Machine zero return in positive X axis;
- 0: Machine zero return in negative X axis.

### Low Speeds

<table>
<thead>
<tr>
<th>0 8 9</th>
<th>0 9 0</th>
<th>0 9 1</th>
<th>0 9 2</th>
<th>0 9 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low speed of X axis machine zero return</td>
<td>Low speed of Y axis machine zero return</td>
<td>Low speed of Z axis machine zero return</td>
<td>Low speed of the 4th axis machine zero return</td>
<td>Low speed of the 5th axis machine zero return</td>
</tr>
</tbody>
</table>

Setting range: 10~1000 (unit: mm/min)

### High Speeds

<table>
<thead>
<tr>
<th>0 9 4</th>
<th>0 9 5</th>
<th>0 9 6</th>
<th>0 9 7</th>
<th>0 9 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>High speed of X axis machine zero return</td>
<td>High speed of Y axis machine zero return</td>
<td>High speed of Z axis machine zero return</td>
<td>High speed of the 4th axis machine zero return</td>
<td>High speed of the 5th axis machine zero return</td>
</tr>
</tbody>
</table>

Setting range: 10~921571875 (unit:mm/min)

### Machine Zero Offsets

<table>
<thead>
<tr>
<th>1 3 0</th>
<th>1 3 1</th>
<th>1 3 2</th>
<th>1 3 3</th>
<th>1 3 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>X axis machine zero offset</td>
<td>Y axis machine zero offset</td>
<td>Z axis machine zero offset</td>
<td>The 4th axis machine zero offset</td>
<td>The 5th axis machine zero offset</td>
</tr>
</tbody>
</table>

Setting range: -99999~99999 (unit: 0.001 mm)
| 1 4 5 | X machine coordinate of the 1st reference point |
| 1 4 6 | Y machine coordinate of the 1st reference point |
| 1 4 7 | Z machine coordinate of the 1st reference point |
| 1 4 8 | The 4th machine coordinate of the 1st reference point |
| 1 4 9 | The 5th machine coordinate of the 1st reference point |
| 1 5 0 | X machine coordinate of the 2nd reference point |
| 1 5 1 | Y machine coordinate of the 2nd reference point |
| 1 5 2 | Z machine coordinate of the 2nd reference point |
| 1 5 3 | The 4th machine coordinate of the 2nd reference point |
| 1 5 4 | The 5th machine coordinate of the 2nd reference point |
| 1 5 5 | X machine coordinate of the 3rd reference point |
| 1 5 6 | Y machine coordinate of the 3rd reference point |
| 1 5 7 | Z machine coordinate of the 3rd reference point |
| 1 5 8 | The 4th machine coordinate of the 3rd reference point |
| 1 5 9 | The 5th machine coordinate of the 3rd reference point |
| 1 6 0 | X machine coordinate of the 4th reference point |
| 1 6 1 | Y machine coordinate of the 4th reference point |
| 1 6 2 | Z machine coordinate of the 4th reference point |
| 1 6 3 | The 4th machine coordinate of the 4th reference point |
| 1 6 4 | The 5th machine coordinate of the 4th reference point |

Setting range: -99999999 — 99999999 (unit: 0.001mm)

3.2.11 Rotary axis function

**RTORI**

- **RTORI = 1**: M29 is executed, Spindle need to return zero;
  - **= 0**: M29 is executed, Spindle need not to return zero.

**RTPCP**

- **RTPCP = 1**: Rigid tapping is the high-speed deep hole cycle (G73);
  - **= 0**: Rigid tapping is the high-speed deep hole cycle (G83).

**RTCRG**

- **RTCRG = 1**: Do not wait for G61.0 to be 1 as executing next program block after rigid tapping cancelled;
  - **= 0**: Do wait for G61.0 to be 1 as executing next program block after rigid tapping cancelled.

**RCS4, ROT4**

- **RCS4 = 1**: Cs function of 4th axis is valid (power on);
  - **= 0**: Cs function of 4th axis is invalid (power on).
- **ROS4, ROT4**: Set the type of 4th axis;
<table>
<thead>
<tr>
<th>Linear</th>
<th>Rotary A</th>
<th>Rotary B</th>
<th>Invalid</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ROT4</strong></td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><strong>ROS4</strong></td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

**RRT4** =1: Zero mode D is used on the 4th rotary axis (power on);
=0: Zero mode A, B, C are used on the 4th rotary axis (power on).

**RRL4** =1: the 4th rel.coor.cycle func.is valid (power on);
=0: the 4th rel.coor.cycle func.is invalid (power on).

**RAB4** =1: the 4th rotates according to symbol direction;
=0: the 4th rotates according to nearby rotation.

**ROA4** =1: the 4th abs.coor.cycle func.is valid (power on);
=0: the 4th abs.coor.cycle func.is invalid (power on).

<table>
<thead>
<tr>
<th>Linear</th>
<th>Rotary A</th>
<th>Rotary B</th>
<th>Invalid</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ROT5</strong></td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><strong>ROS5</strong></td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

**RRT5** =1: Zero mode D of the 5th axis (power on);
=0: Zero mode A, B, C of the 5th axis (power on).

**RRL5** =1: the 5th rel.coor.cycle func.is valid (power on);
=0: the 5th rel.coor.cycle func.is invalid (power on).

**RAB5** =1: the 5th rotation according to symbol direction;
=0: the 5th rotation according to nearby direction.

**ROA5** =1: the 5th abs.coor.cycle func.is valid (power on);
=0: the 5th abs.coor.cycle func.is invalid (power on).

**RRT4** =1: Zero mode D is used on the 5th rotary axis (power on);
=0: Zero mode A, B, C are used on the 5th rotary axis (power on).

**RRL4** =1: the 5th rel.coor.cycle func.is valid (power on);
=0: the 5th rel.coor.cycle func.is invalid (power on).

**RAB4** =1: 5th rotates according to symbol direction;
=0: 5th rotates according to nearby rotation.

**ROA4** =1: the 5th abs.coor.cycle func.is valid (power on);
=0: the 5th abs.coor.cycle func.is invalid(power on).

<table>
<thead>
<tr>
<th>Setting</th>
<th>Parameter Description</th>
<th>Setting Range</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>077</td>
<td>Initial speed of acc. &amp; dec in using CS function</td>
<td>0 ～ 5000 (Unit: deg/min)</td>
<td></td>
</tr>
<tr>
<td>078</td>
<td>Acc. &amp; dec. time constant in using CS function</td>
<td>10 ～ 10000 (Unit: ms)</td>
<td></td>
</tr>
<tr>
<td>081</td>
<td>Initial speed of linear acceleration/deceleration in rigid tapping</td>
<td>0 ～ 5000 (Unit: mm/min)</td>
<td></td>
</tr>
<tr>
<td>082</td>
<td>Linear time constant in rigid tapping tool infeed</td>
<td>10 ～ 10000 (Unit: ms)</td>
<td></td>
</tr>
<tr>
<td>083</td>
<td>Time constant in rigid tapping tool retract</td>
<td>0 ～ 4000 (Unit: ms)</td>
<td></td>
</tr>
<tr>
<td>084</td>
<td>Override value in rigid tapping tool retract(0: override is set to 100%)</td>
<td>0 ～ 200, 0: override is set to 100%</td>
<td></td>
</tr>
<tr>
<td>085</td>
<td>Tool retract amount in deep hole rigid tapping (high-speed, standard)</td>
<td>0 ～ 32767000 (Unit: 0.001mm)</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>One-rotation increment of the 4th axis</td>
<td>1 ～ 9999999 (Unit: 0.001deg)</td>
<td></td>
</tr>
<tr>
<td>190</td>
<td>One-rotation increment of the 5th axis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>Amount of valid keys pressed simultaneously</td>
<td>2 ～ 5</td>
<td></td>
</tr>
<tr>
<td>202</td>
<td>Define the name of the 4th axis (A: 65, B: 66, C: 67)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>203</td>
<td>Define the name of the 5th axis (A: 65, B: 66, C: 67)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 4  MACHINE DEBUGGING METHODS AND STEPS

The trial run methods and steps at initial power on for this GSK980MDa are described in this chapter. The corresponding operation can be performed after the debugging by the following steps.

4.1 Emergency Stop and Stroke Limit

This GSK980MDa system has software limit function, it is suggested that the stroke limit switches are fixed in the positive or negative axes for hardware limit. The connection is shown in follows: (The chart is designed for X, Y, Z axes)

![Diagram of Emergency Stop and Stroke Limit](image)

So the MESP of bit parameter No.17 should be set to 0.

And the CNC diagnostic message ESP can monitor the state of emergency stop input signal.

In Manual or MPG mode, slowly move the axes to test the validity of stroke limit switch, correctness of alarm display, validity of overtravel release button. When the overtravel occurs or Emergency Stop button is pressed, “emergency stop” alarm will be issued by CNC system. The alarm can be cancelled by pressing down the Overtravel button and moving reversely.

4.2 Drive unit Unit Setting

Set BIT4~BIT0 of bit parameter No.009 according to alarm logic level of drive unit. The BIT4~BIT0 of bit parameter No.009 for our drive unit are all set for 1.

If the machine moving direction is not consistent with the moving command, modify the BIT4~BIT0 of bit parameter No.008, BIT4~BIT0 of bit parameter No.019, BIT4~BIT0 of bit parameter No.20.
4.3 Gear Ratio Adjustment

The data parameter No.049 ~ No.058 can be modified for electronic gear ratio adjustment to meet the different mechanical transmission ratio if the machine travel distance is not consistent with the displacement distance displayed by the CNC coordinate.

Calculation formula:
\[
\frac{CMR}{CMD} = \frac{\delta \times 360}{\alpha \times L} \times \frac{Z_m}{Z_d}
\]

CMR: command multiplier coefficient (data parameter No049, No050, No051, No052, No053)
CMD: command frequency division coefficient (data parameter No054, No055, No056, No057, No058)
\[\alpha\] :: pulse volume, motor rotation angle for a pulse
L: lead
\[\delta\] :: min. input command unit of CNC (0.0001 for all axes of GSK980MDa)
ZM: gear teeth of lead screw
ZD: gear teeth of motor

If the electronic gear ratio numerator is greater than the denominator, the allowed CNC max. speed will decrease. For example: the data parameter No.051 (CMRZ) =2, No.056 (CMDZ) =1, so the allowed Z axis max. speed is 8000mm/min.

If the electronic gear ratio numerator is not equal to the denominator, the allowed CNC positioning precision will decrease. For example: when the data parameter No.051 (CMRZ) =1 and No.056 (CMDZ) =5, the pulse is not output as the input increment is 0.004, but a pulse is output if the input increment is up to 0.005.

In order to ensure the CNC positioning precision, speed index and match with digit servo with electronic gear ratio function, it is suggested that the CNC electronic gear ratio is set for 1:1 or the electronic gear ratio calculated is set to the digital servo.

When matching with the step drive, choose the drive unit with step division function as far as possible, and properly select mechanical transmission ratio. The 1:1 electronic gear ratio should be ensured to avoid the too large difference between the numerator and the denominator of this CNC gear ratio.

Example:

Match GSK980MDa with DA98B, take X axis for example: set command multiplier coefficient and command frequency division coefficient to 1. Calculation formula is shown below.

CNC:
\[
\frac{CMR}{CMD} = \frac{\delta \times 360}{\alpha \times L} \times \frac{Z_m}{Z_d} = 1
\]

The following conclusions can be reached:
\[
\alpha = \frac{\delta \times 360}{L} \times \frac{Z_m}{Z_d} \text{(deg/pulse)}
\]

Drive unit:
Parameters 12, 13 of drive unit correspond to position command pulse frequency division
molecule and denominator. Calculation formula of drive unit gear ratio is shown as follows:

\[ P \times G = 4 \times N \times C \]

\[ P: \text{Correspondence between required pulse volume for motor rotates 3600 and CNC end:} \]

\[ P = \frac{360}{\alpha} \]

\[ G: \text{Electronic gear ratio of drive unit, G= position command pulse frequency division molecule / position command pulse frequency division denominator} \]

\[ N: \text{Set motor rev number to 1} \]

\[ C: \text{Wire number of feedback encoder: DA98B is 2500p/r.} \]

The following conclusions can be reached:

\[ G = \frac{4 \times N \times C}{P} = \frac{4 \times N \times C \times \alpha}{360} = \frac{4 \times N \times C}{360} \times \frac{\delta \times 360}{L} \times \frac{Z_M}{Z_D} = \]

\[ = \frac{10 \times Z_M}{L \times Z_D} \]

Set molecule and denominator of calculated ratio to drive unit 12, 13 separately.

4.4 Acceleration & Deceleration Characteristic Adjustment

Adjust the relative CNC parameters according to the factors such as the drive unit, motor characteristics and machine load:

- Data parameter №059 – №063: X, Y, Z, 4th, 5th axis rapid traverse rate;
- Data parameter №064 – №068: Linear acceleration & deceleration time constant of X, Y, Z, 4th, 5th axis rapid traverse rate;
- Data parameter №069: Rapid traverse speed when rapid override is F0;
- Data parameter №070: Upper limit of axes cutting feedrate;
- Data parameter №071: Start/end speed of exponential acceleration & deceleration in cutting feeding;
- Data parameter №072: Exponential acceleration & deceleration time constant of cutting feeding;
- Data parameter №073: Start/end speed of exponential acceleration & deceleration in MPG/Step feedrate;
- Data parameter №074: Exponential acceleration & deceleration time constant of MPG/STEP/manual feed;
- Data parameter №075: Start/end speed in thread cutting of each ax;
- Data parameter №077: Initial feedrate of acc.&dec in CS axis;
- Data parameter №078: Acc.&dec.time constant in CS axis;
- Data parameter №081: Initial speed of linear acceleration/deceleration in rigid tapping;
- Data parameter №082: Linear acceleration/deceleration time constant in rigid tapping tool infeed;
- Data parameter №083: Linear acceleration/deceleration time constant in rigid tapping tool retraction;
- Data parameter №084: Override value in rigid tapping tool retract;
- Data parameter №172: Initial feedrate when power on;
- Data parameter №174: Feedrate of DRY run;
- SMZ of bit parameter №007: for validity of smoothing transition between blocks.
The larger the acceleration&deceleration time constant is, the slower the acceleration&deceleration is, the smaller the machine movement impact and the lower the machining efficiency is. And vice versa.

If acceleration&deceleration time constants are equal, the higher the acceleration & deceleration start/end speed is, the faster the acceleration & deceleration is, the bigger the machine movement impact and the higher the machining efficiency is. And vice versa.

The principle for acceleration&deceleration characteristic adjustment is to properly reduce the acceleration & deceleration time constant and increase the acceleration&deceleration start/end speed to improve the machining efficiency on the condition that there is no alarm, motor out-of-step and obvious machine impact. If the acceleration&deceleration time constant is set too small, and the start/end speed is set too large, it is easily to cause drive unit alarm, motor out-of-step or machine vibration.

When the bit parameter \( \text{N007 BIT3 (SMZ)} \) =1, the feedrate drops to the start speed of the acceleration&deceleration at the cutting path intersection, then it accelerates to the specified speed of the adjacent block to obtain an accurate positioning at the path intersection, but this will reduce the machining efficiency. When \( \text{SMZ}=0 \), the adjacent cutting path transits smoothly by the acceleration&deceleration. The feedrate does not always drop to the start speed when the previous path is finished and a circular transition (non-accurate positioning) will be formed at the path intersection. The machining surface by this path transition has a good finish and a higher machining efficiency. When the stepper motor drive unit is applied, the SMZ of the bit parameter \( \text{N007} \) should be set to 1 to avoid the out-of-step.

When the stepper motor drive unit is applied to this system, the out-of-step may occur if rapid traverse speed is too large, acceleration&deceleration time constant is too small, acceleration&deceleration start/end speed is too large. The suggested parameter setting is shown in follows (the electronic gear ratio is 1:1):

- Data parameter N059~N063≤5000
- Data parameter N064~N068≥350
- Data parameter N071≤50
- Data parameter N072≥150
- Data parameter N073≤50
- Data parameter N074≥150
- Data parameter N075≤100

When AC servo motor drive unit is applied to this system, the machining efficiency can be improved by a larger start speed and smaller ACC&DEC time constant setting. If optimum ACC&DEC characteristics are required, the ACC&DEC time constant may be set to 0, which can be got by adjusting the AC servo ACC&DEC parameters. The suggested parameter settings are as follows (electronic gear ratio is 1:1):

- Data parameter N059~N063 set higher properly
- Data parameter N064~N068≤60
- Data parameter N071≥50
- Data parameter N072≤50
- Data parameter N073≥50
- Data parameter N074≤50
- Data parameter N075≤500

The parameter settings above are recommended for use, refer to the actual conditions of the drive unit, motor characteristic and machine load for its proper setting.
4.5 Machine Zero Adjustment

Adjust the relevant parameters based on the valid level of the connection signal, zero return type or direction applied:

(DEC1) of the bit parameter №004: valid level of deceleration signal as machine zero return
(ZM5~ZMX) of the bit parameter №006: return and initial backlash direction of X, Y, Z, 4th, 5th axes machine zeroes at deceleration.

(ZC5~ZCX) of the bit parameter №007: it is able to set whether an approach switch taken as both deceleration and zero signals when X, Y, Z, 4th, 5th axes return to machine zero point.

(ZNLK) of the bit parameter №011: for direction keys lock when performing zero return
(ZRS5~ZRSX) of the bit parameter №014: for deceleration and zero signals detection of X, Y, Z axes in machine zero return.

(MZR5~MZRX) of the bit parameter №22: for positive or negative zero turn of X, Y, Z, 4th, 5th axes

Data parameter №089~№093: low speed of X, Y, Z, 4th, 5th axes in machine zero return
Data parameter №094~№098: high speed of X, Y, Z, 4th, 5th axes in machine zero return
RRT4 of bit parameter №027 and RRT5 of №029 set the machine zero return type of the 4th and the 5th axis separately.

Machine zero return can be done after the validity of overtravel limit switch is confirmed. Machine zero return types A, B, C can be selected for basic axes (X, Y, Z). Machine zero return types A, B, C, D can be selected for additional axes (4th, 5th).

The machine zero is usually fixed at the max. travel point, and the effective stoke of the zero return touch block should be more than 25mm to ensure a sufficient deceleration distance for accurate zero return. The more rapid the machine zero return is, the longer the zero return touch block should be. Or the moving carriage will rush through the block which may influence the zero return precision because of the insufficient deceleration distance.

Usually there are 2 types of machine zero return connection:
1. The connection to AC servo motor: schematic diagram of using a travel switch and a servo motor one-rotation signal separately

Fig. 4-2
By this connection type, when the deceleration switch is released in machine zero return, the one-rotation signal of encoder should be avoided to be at a critical point after the travel switch is released. In order to improve the zero return precision, it should be ensured the motor reaches the one-rotation signal of encoder after it rotates for half circle. And the moving distance for motor half circle rotation is the motor gear teeth/(2×lead screw gear teeth)

2 The connection to stepper motor: the schematic diagram of using a proximity switch taken as both deceleration signal and zero signal

4.6 Spindle Adjustment

4.6.1 Spindle encoder

Encoder with the linear number 100~5000p/r is needed to be installed on the machine for threading. The linear number is set by data parameter No.109. The transmission ratio(spindle gear teeth/encoder gear teeth) between encoder and spindle is 1/255~255. The spindle gear teeth are set by CNC data parameter No.110, and the encoder gear teeth are set by data parameter No.111. Synchronous belt transmission should be applied for it (no sliding transmission).

The DGN.011 and DNG.012 of CNC diagnosis messages are used to check the validity of threading signal from the spindle encoder.

4.6.2 Spindle brake

After spindle stop is executed, proper spindle brake time should be set to stop the spindle promptly in order to enhance the machining efficiency. If the brake is employed with energy consumption type, too long braking time may damage the motor. So the brake time is set by PLC.