

GE Fanuc Automation

Computer Numerical Control Products

Series 0-MD / 0-GSD

Operator's Manual

GFZ-62574EN/02

February 1997

SAFETY PRECAUTIONS

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

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

Contents

| 1. | DEFINITION OF WARNING, CAUTION, AND NOTE |
|----|---|
| 2. | GENERAL WARNINGS AND CAUTIONS |
| 3. | WARNINGS AND CAUTIONS RELATED TO PROGRAMMING |
| 4. | WARNINGS AND CAUTIONS RELATED TO HANDLING s–7 |
| 5. | WARNINGS RELATED TO DAILY MAINTENANCE |

DEFINITION OF WARNING, CAUTION, AND NOTE

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

WARNING

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

CAUTION

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

NOTE

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

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

2

GENERAL WARNINGS AND CAUTIONS

WARNING

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

WARNING

8. Some functions may have been implemented at the request of the machine–tool builder. When using such functions, refer to the manual supplied by the machine–tool builder for details of their use and any related cautions.

NOTE

Programs, parameters, and macro variables are stored in nonvolatile memory in the CNC unit. Usually, they are retained even if the power is turned off. Such data may be deleted inadvertently, however, or it may prove necessary to delete all data from nonvolatile memory as part of error recovery.

To guard against the occurrence of the above, and assure quick restoration of deleted data, backup all vital data, and keep the backup copy in a safe place.

3

WARNINGS AND CAUTIONS RELATED TO PROGRAMMING

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

WARNING

1. Coordinate system setting

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

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

2. Positioning by nonlinear interpolation

When performing positioning by nonlinear interpolation (positioning by nonlinear movement between the start and end points), the tool path must be carefully confirmed before performing programming.

Positioning involves rapid traverse. If the tool collides with the workpiece, it may damage the tool, the machine itself, the workpiece, or cause injury to the user.

3. Function involving a rotation axis

When programming polar coordinate interpolation or normal-direction (perpendicular) control, pay careful attention to the speed of the rotation axis. Incorrect programming may result in the rotation axis speed becoming excessively high, such that centrifugal force causes the chuck to lose its grip on the workpiece if the latter is not mounted securely.

Such mishap is likely to damage the tool, the machine itself, the workpiece, or cause injury to the user.

4. Inch/metric conversion

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

5. Constant surface speed control

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

WARNING

6. Stroke check

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

7. Tool post interference check

A tool post interference check is performed based on the tool data specified during automatic operation. If the tool specification does not match the tool actually being used, the interference check cannot be made correctly, possibly damaging the tool or the machine itself, or causing injury to the user.

After switching on the power, or after selecting a tool post manually, always start automatic operation and specify the tool number of the tool to be used.

8. Absolute/incremental mode

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

9. Plane selection

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

10. Torque limit skip

Before attempting a torque limit skip, apply the torque limit. If a torque limit skip is specified without the torque limit actually being applied, a move command will be executed without performing a skip.

11. Programmable mirror image

Note that programmed operations vary considerably when a programmable mirror image is enabled.

12. Compensation function

If a command based on the machine coordinate system or a reference position return command is issued in compensation function mode, compensation is temporarily canceled, resulting in the unexpected behavior of the machine.

Before issuing any of the above commands, therefore, always cancel compensation function mode.



WARNINGS AND CAUTIONS RELATED TO HANDLING

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

WARNING

1. Manual operation

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

2. Manual reference position return

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

3. Manual numeric command

When issuing a manual numeric command, determine the current position of the tool and workpiece, and ensure that the movement axis, direction, and command have been specified correctly, and that the entered values are valid.

Attempting to operate the machine with an invalid command specified may damage the tool, the machine itself, the workpiece, or cause injury to the operator.

4. Manual handle feed

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

5. Disabled override

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

6. Origin/preset operation

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

WARNING

7. Workpiece coordinate system shift

Manual intervention, machine lock, or mirror imaging may shift the workpiece coordinate system. Before attempting to operate the machine under the control of a program, confirm the coordinate system carefully.

If the machine is operated under the control of a program without making allowances for any shift in the workpiece coordinate system, the machine may behave unexpectedly, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the operator.

8. Software operator's panel and menu switches

Using the software operator's panel and menu switches, in combination with the MDI panel, it is possible to specify operations not supported by the machine operator's panel, such as mode change, override value change, and jog feed commands.

Note, however, that if the MDI panel keys are operated inadvertently, the machine may behave unexpectedly, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the user.

9. Manual intervention

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

10. Feed hold, override, and single block

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

11. Dry run

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

12. Cutter and tool nose radius compensation in MDI mode

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

13. Program editing

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

5

WARNINGS RELATED TO DAILY MAINTENANCE

WARNING

1. Memory backup battery replacement

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

When replacing the batteries, be careful not to touch the high–voltage circuits (marked \triangle and fitted with an insulating cover).

Touching the uncovered high-voltage circuits presents an extremely dangerous electric shock hazard.

NOTE

The CNC uses batteries to preserve the contents of its memory, because it must retain data such as programs, offsets, and parameters even while external power is not applied.

If the battery voltage drops, a low battery voltage alarm is displayed on the machine operator's panel or CRT screen.

When a low battery voltage alarm is displayed, replace the batteries within a week. Otherwise, the contents of the CNC's memory will be lost.

Refer to the maintenance section of this manual for details of the battery replacement procedure.

WARNING

2. Absolute pulse coder battery replacement

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

When replacing the batteries, be careful not to touch the high–voltage circuits (marked ▲ and fitted with an insulating cover).

Touching the uncovered high-voltage circuits presents an extremely dangerous electric shock hazard.

NOTE

The absolute pulse coder uses batteries to preserve its absolute position.

If the battery voltage drops, a low battery voltage alarm is displayed on the machine operator's panel or CRT screen.

When a low battery voltage alarm is displayed, replace the batteries within a week. Otherwise, the absolute position data held by the pulse coder will be lost.

Refer to the maintenance section of this manual for details of the battery replacement procedure.

WARNING

3. Fuse replacement

Before replacing a blown fuse, however, it is necessary to locate and remove the cause of the blown fuse.

For this reason, only those personnel who have received approved safety and maintenance training may perform this work.

When replacing a fuse with the cabinet open, be careful not to touch the high–voltage circuits (marked \blacktriangle and fitted with an insulating cover).

Touching an uncovered high-voltage circuit presents an extremely dangerous electric shock hazard.

| S | AF | ET | PRECAUTIONS | s–1 |
|----|----|-----|---|-----|
| I | GE | NE | RAL | |
| | 1. | GE | NERAL | 3 |
| | | 1.1 | GENERAL FLOW OF OPERATION OF CNC MACHINE TOOL | 5 |
| | | 1.2 | NOTES ON READING THIS MANUAL | 7 |
| II | PR | OG | RAMMING | |
| | 1. | GE | NERAL | 11 |
| | | 1.1 | TOOL MOVEMENT ALONG WORKPIECE PARTS FIGURE-INTERPOLATION | 12 |
| | | 1.2 | FEED-FEED FUNCTION | 14 |
| | | 1.3 | PART DRAWING AND TOOL MOVEMENT | 15 |
| | | | 1.3.1 Reference Position (Machine–Specific Position) 1.3.2 Coordinate System on Part Drawing and Coordinate System Specified by CNC Coordinate System | 15 |
| | | | 1.3.3 How to Indicate Command Dimensions for Moving the Tool – Absolute, Incremental Commands | 19 |
| | | 1.4 | CUTTING SPEED – SPINDLE SPEED FUNCTION | 21 |
| | | 1.5 | SELECTION OF TOOL USED FOR VARIOUS MACHINING – TOOL FUNCTION | 22 |
| | | 1.6 | COMMAND FOR MACHINE OPERATIONS – MISCELLANEOUS FUNCTION | 23 |
| | | 1.7 | PROGRAM CONFIGURATION | 24 |
| | | 1.8 | TOOL FIGURE AND TOOL MOTION BY PROGRAM | 27 |
| | | 1.9 | TOOL MOVEMENT RANGE – STROKE | 28 |
| | 2. | СС | NTROLLED AXES | 29 |
| | | 2.1 | CONTROLLED AXES | 30 |
| | | 2.2 | NAME OF AXES | 30 |
| | | 2.3 | INCREMENT SYSTEM | 30 |
| | | 2.4 | MAXIMUM STROKE | 30 |
| | 3. | RE | PARATORY FUNCTION (G FUNCTION) | 31 |
| | 4. | IN | ERPOLATION FUNCTIONS | 34 |
| | | 4.1 | POSITIONING (G00) | 35 |
| | | 4.2 | SINGLE DIRECTION POSITIONING (G60) | 36 |
| | | 4.3 | LINEAR INTERPOLATION (G01) | 37 |
| | | 4.4 | CIRCULAR INTERPOLATION (G02.G03) | 39 |
| | | 4.5 | SKIP FUNCTION (G31) (FOR 0–GSD ONLY) | 43 |
| | 5. | FE | | 45 |
| | | 51 | GENERAL | 46 |
| | | 5.2 | RAPID TRAVERSE | 48 |
| | | 53 | CUTTING FEED | 49 |
| | | 5.4 | CUTTING FEEDRATE CONTROL | 51 |
| | | | 5.4.1 Exact Stop (G09, G61) Cutting Mode (G64) Tapping Mode (G63) | 52 |

| | 5.5 | 5.4.2 Internal Circular Cutting Feedrate Change DWELL (G04) | 53 54 |
|----|------|---|------------|
| 6. | RE | | 55 |
| 7. | со | ORDINATE SYSTEM | 59 |
| | 7 1 | | 60 |
| | 7.1 | | 00 |
| | 7.2 | WORKPIECE COORDINATE SYSTEM | 61 |
| | | 7.2.1 Setting a Workpiece Coordinate System | 61 |
| | | 7.2.2 Selecting a Workpiece Coordinate System | 62 63 |
| | 73 | | 65 |
| | 7.5 | | 67 |
| | /.+ | I LANE SELECTION | 07 |
| 8. | CO | ORDINATE VALUE AND DIMENSION | 68 |
| | 8.1 | ABSOLUTE AND INCREMENTAL PROGRAMMING (G90, G91) | 69 |
| | 8.2 | INCH/METRIC CONVERSION (G20,G21) | 70 |
| | 8.3 | DECIMAL POINT PROGRAMMING | 71 |
| • | | | 70 |
| 9. | 5P | INDLE SPEED FUNCTION (S FUNCTION) | 13 |
| | 9.1 | SPECIFYING THE SPINDLE SPEED WITH A BINARY CODE | 74 |
| | 9.2 | SPECIFYING THE SPINDLE SPEED VALUE DIRECTLY (S5–DIGIT COMMAND) | 74 |
| 10 |).TO | OL FUNCTION (T FUNCTION) | 75 |
| | 10.1 | TOOL SELECTION FUNCTION | 76 |
| | | | |
| 11 | .AU | | 77 |
| | 11.1 | AUXILIARY FUNCTION (M FUNCTION) | 78 |
| | 11.2 | MULTIPLE M COMMANDS IN A SINGLE BLOCK | 79 |
| | 11.3 | THE SECOND AUXILIARY FUNCTIONS (B CODES) (FOR 0-MD ONLY) | 80 |
| | | | |
| 12 | 2.PR | | 81 |
| | 12.1 | PROGRAM COMPONENTS OTHER THAN PROGRAM SECTIONS | 83 |
| | 12.2 | PROGRAM SECTION CONFIGURATION | 86 |
| | 12.3 | SUBPROGRAM | 92 |
| 13 | | NCTIONS TO SIMPLIFY PROGRAMMING | 96 |
| 10 | 12.1 | | 07 |
| | 15.1 | 13.1.1 High speed Pack Drilling Cycle (G73) | 97 101 |
| | | 13.1.2 Left-handed Tapping Cycle (G74) | 101 |
| | | 13.1.3 Fine Boring Cycle (G76) | 105 |
| | | 13.1.4 Drilling Cycle, Spot Drilling (G81) | 107 |
| | | 13.1.5 Drilling Cycle Counter Boring Cycle (G82) | 109 |
| | | 13.1.6 Peck Drilling Cycle (G83) | 111 |
| | | 13.1.7 Tapping Cycle (G84) | 113 |
| | | 13.1.8 Boring Cycle (G85) | 115 117 |
| | | 13.1.10 Boring Cycle Back Boring Cycle (G87) | 119 |

| | 13.1.11 Boring Cycle (G88) | 121 |
|-------|--|-----|
| | 13.1.12 Boring Cycle (G89) | 123 |
| | 13.1.13 Canned Cycle Cancel (G80) | 124 |
| 13.2 | RIGID TAPPING | 127 |
| | 13.2.1 Rigid Tapping (G84) | 127 |
| | 13.2.2 Left-handed Rigid Tapping Cycle (G74) | 129 |
| | 13.2.3 Peck Rigid Tapping Cycle (G84 or G74) | 131 |
| | 13.2.4 Canned Cycle Cancel (G80) | 132 |
| 13.3 | CANNED GRINDING CYCLE (FOR 0-GSD ONLY) | 133 |
| | 13.3.1 Plunge Grinding Cycle (G75) | 134 |
| | 13.3.2 Direct Constant–dimension Plunge Grinding Cycle (G77) | 136 |
| | 13.3.3 Continuous–feed Surface Grinding Cycle (G78) | 138 |
| | 13.3.4 Intermittent–feed Surface Grinding Cycle (G79) | 140 |
| 13.4 | AUTOMATIC GRINDING WHEEL DIAMETER COMPENSATION AFTER DRESSING | 142 |
| 1011 | 13.4.1 Checking the Minimum Grinding Wheel Diameter (for 0–GSD only) | 142 |
| 12.5 | IN FEED CRINDING ALONG THE VAND ZAYES AT THE END OF TABLE SWING | 1.2 |
| 15.5 | (FOR 0–GSD ONLY) | 143 |
| 13.6 | EXTERNAL MOTION FUNCTION (G81) | 144 |
| 44.00 | | |
| 14.00 | | 43 |
| 14.1 | TOOL LENGTH OFFSET (G43,G44,G49) | 146 |
| 14.2 | OVERVIEW OF CUTTER COMPENSATION C (G40 – G42) | 150 |
| 14.3 | DETAILS OF CUTTER COMPENSATION C | 156 |
| | 14.3.1 General | 156 |
| | 14.3.2 Tool Movement in Start-up | 157 |
| | 14.3.3 Tool Movement in Offset Mode | 161 |
| | 14.3.4 Tool Movement in Offset Mode Cancel | 175 |
| | 14.3.5 Interference Check | 181 |
| | 14.3.6 Overcutting by Cutter Compensation | 186 |
| | 14.3.7 Input Command from MDI | 189 |
| 14.4 | TOOL COMPENSATION VALUES, NUMBER OF COMPENSATION VALUES, AND ENTERING VALUES FROM THE PROGRAM (G10) | 190 |
| 14.5 | NORMAL DIRECTION CONTROL ($C150$, $C151$, $C152$) (EOR 0, CSD ONLY) | 102 |
| 14.3 | NORMAL DIRECTION CONTROL (G150, G151, G152) (FOR 0–GSD ONLE) | 192 |
| 15.CU | STOM MACRO A | 196 |
| 15.1 | CUSTOM MACRO COMMAND | 197 |
| 10.1 | 15.1.1 M98 (Single call) | 107 |
| | 15.1.1 W/8 (Single call) | 197 |
| | 15.1.3 Subprogram call using T code . | 198 |
| | 15.1.4 G66 (Modal call) | 199 |
| | 15.1.5 Argument specification | 200 |
| 15.2 | CUSTOM MACRO BODY | 202 |
| 10.2 | 15.2.1 Variables | 202 |
| | 15.2.1 variables 15.2.2 Kind of Variables | 202 |
| | 15.2.2 Macro Instructions (G65) | 205 |
| | 15.2.4 Warning and Notes on Custom Macro | 213 |
| | | |
| 16.CU | SIOM MACRO B | 214 |
| 16.1 | VARIABLES | 215 |
| 16.2 | SYSTEM VARIABLES | 218 |
| 16.3 | ARITHMETIC AND LOGIC OPERATION | 224 |

| 16.4 | MACR | O STATEMENTS AND NC STATEMENTS | 228 |
|-------|--------------|---------------------------------------|-----|
| 16.5 | BRAN | CH AND REPETITION | 229 |
| | 16.5.1 | Unconditional Branch (GOTO Statement) | 229 |
| | 16.5.2 | Conditional Branch (IF Statement) | 229 |
| | 16.5.3 | Repetition (While Statement) | 230 |
| 16.6 | MACR | O CALL | 233 |
| | 16.6.1 | Simple Call (G65) | 233 |
| | 16.6.2 | Modal Call (G66) | 238 |
| | 16.6.3 | Macro Call Using G Code | 240 |
| | 16.6.4 | Macro Call Using an M Code | 241 |
| | 16.6.5 | Subprogram Call Using an M Code | 242 |
| | 16.6.6 | Subprogram Calls Using a T Code | 243 |
| | 16.6.7 | Sample Program | 244 |
| 16.7 | PROCE | ESSING MACRO STATEMENTS | 246 |
| 16.8 | REGIS | TERING CUSTOM MACRO PROGRAMS | 248 |
| 16.9 | LIMITA | ATIONS | 249 |
| 16.10 |) EXTER | NAL OUTPUT COMMANDS | 250 |
| 17.PR | OGRAN | IMABLE PARAMETER ENTRY(G10) | 254 |
| 18.RO | TARY A | XIS ROLL-OVER | 256 |

III OPERATION

| 1. | GE | NERAL | | 259 |
|----|-----|-------|---|-----|
| | 1.1 | MANU | JAL OPERATION | 260 |
| | 1.2 | TOOL | MOVEMENT BY PROGRAMING – AUTOMATIC OPERATION | 262 |
| | 1.3 | AUTO | MATIC OPERATION | 263 |
| | 1.4 | TESTI | NG A PROGRAM | 264 |
| | | 1.4.1 | Check by Running the Machine | 264 |
| | | 1.4.2 | How to View the Position Display Change without Running the Machine | 265 |
| | 1.5 | EDITI | NG A PART PROGRAM | 266 |
| | 1.6 | DISPL | AYING AND SETTING DATA | 267 |
| | 1.7 | DISPL | ΑΥ | 270 |
| | | 1.7.1 | Program Display (See III–11.2.1 and III–11.3.1) | 270 |
| | | 1.7.2 | Current Position Display (See III-11.1.1 to 11.1.3) | 271 |
| | | 1.7.3 | Alarm Display (See III–7.1) | 271 |
| | | 1.7.4 | Parts Count Display, Run Time Display (See III-11.1.5) | 272 |
| | 1.8 | DATA | OUTPUT (See III-8) | 273 |
| 2. | OP | ERATI | ONAL DEVICES | 274 |
| | 2.1 | CRT/N | IDI PANELS AND LCD/MDI PANELS | 275 |
| | 2.2 | FUNC | TION KEYS AND SOFT KEYS | 278 |
| | | 2.2.1 | General Screen Operations | 278 |
| | | 2.2.2 | Function Keys | 279 |
| | | 2.2.3 | Key Input and Input Buffer | 280 |
| | 2.3 | EXTE | RNAL I/O DEVICES | 282 |
| | | 2.3.1 | FANUC Handy File | 284 |
| | | 2.3.2 | FANUC Floppy Cassette | 284 |

| | | 2.3.3 FANUC FA Card 2.3.4 FANUC PPR | |
|--------------------------|---|--|---|
| | 2.4 | 2.3.5 Portable Tape Reader | 286 |
| | 2.4 | 2.4 POWER ON/OFF | |
| | | 2.4.1 Display of Software Configuration | 288 |
| | | 2.4.3 Power Disconnection | |
| 3. | MA | MANUAL OPERATION | 289 |
| | 3.1 | 3.1 MANUAL REFERENCE POSITION RETURN | |
| | 3.2 | 3.2 JOG FEED | |
| | 3.3 | 3.3 INCREMENTAL FEED | |
| | 3.4 | 3.4 MANUAL HANDLE FEED | |
| | 3.5 | 3.5 MANUAL ABSOLUTE ON AND OFF | |
| 4. | AU | | 302 |
| | 4.1 | A.1 MEMORY OPERATION | |
| | 4.2 | A.2 MDI OPERATION | |
| | 4.3 | A.3 DNC OPERATION | |
| | 4.4 | A.4 MIRROR IMAGE | |
| | 4.5 | 1.5 SEQUENCE NUMBER SEARCH | |
| 5. | TE | TEST OPERATION | 312 |
| | | | |
| | 5.1 | 5.1 MACHINE LOCK AND AUXILIARY FUNCTION LOCK | |
| | 5.1 5.2 | 5.1MACHINE LOCK AND AUXILIARY FUNCTION LOCK5.2FEEDRATE OVERRIDE | |
| | 5.1 5.2 5.3 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE | |
| | 5.1 5.2 5.3 5.4 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN | |
| | 5.1 5.2 5.3 5.4 5.5 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK | 313 314 314 315 316 316 317 |
| 6. | 5.1 5.2 5.3 5.4 5.5 SA | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS | 313 314 314 315 316 316 317 319 |
| 6. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS | 313 314 314 315 316 317 317 319 320 |
| 6. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS EMERGENCY STOP OVERTRAVEL | 313 314 314 315 316 316 317 319 320 321 |
| 6. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS 5.1 EMERGENCY STOP 5.2 OVERTRAVEL 5.3 STROKE CHECK | 313 314 314 315 316 317 319 320 321 322 |
| 6. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 AL | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS 5.1 EMERGENCY STOP 5.2 OVERTRAVEL 5.3 STROKE CHECK ALARM AND SELF-DIAGNOSIS FUNCTIONS | 313 314 314 315 316 317 319 320 321 322 324 |
| 6 . 7 . | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 ALL 7.1 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS 5.1 EMERGENCY STOP 5.2 OVERTRAVEL 5.3 STROKE CHECK ALARM AND SELF-DIAGNOSIS FUNCTIONS 7.1 ALARM DISPLAY | 313 314 314 315 316 317 319 320 321 322 324 325 |
| 6. 7. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 ALL 7.1 7.2 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS EMERGENCY STOP OVERTRAVEL STROKE CHECK ALARM AND SELF-DIAGNOSIS FUNCTIONS ALARM DISPLAY CHECKING BY SELF-DIAGNOSTIC SCREEN | 313 314 314 315 316 317 319 320 321 322 324 325 327 |
| 6. 7. 8. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 AL 7.1 7.2 DA | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE APID TRAVERSE OVERRIDE ADRY RUN SINGLE BLOCK SINGLE BLOCK SAFETY FUNCTIONS EMERGENCY STOP OVERTRAVEL OVERTRAVEL STROKE CHECK ALARM AND SELF-DIAGNOSIS FUNCTIONS ALARM DISPLAY CHECKING BY SELF-DIAGNOSTIC SCREEN | 313 314 314 315 316 317 319 320 321 322 324 325 327 329 |
| 6. 7. 8. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 ALL 7.1 7.2 DA 8.1 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE A DRY RUN SINGLE BLOCK SAFETY FUNCTIONS EMERGENCY STOP OVERTRAVEL OVERTRAVEL STROKE CHECK ALARM DISPLAY CHECKING BY SELF-DIAGNOSTIC SCREEN FILES | 313 314 314 315 316 317 319 320 321 322 324 322 324 325 327 329 330 |
| 6. 7. 8. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 ALL 7.1 7.2 DA 8.1 8.2 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS EMERGENCY STOP OVERTRAVEL STROKE CHECK ALARM AND SELF-DIAGNOSIS FUNCTIONS ALARM DISPLAY CHECKING BY SELF-DIAGNOSTIC SCREEN FILES FILES FILE SEARCH | 313 314 314 315 316 317 319 320 321 322 324 324 325 327 329 330 330 332 |
| 6. 7. 8. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 ALL 7.1 7.2 DA 8.1 8.2 8.3 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS EMERGENCY STOP OVERTRAVEL STROKE CHECK ALARM AND SELF-DIAGNOSIS FUNCTIONS ALARM DISPLAY CHECKING BY SELF-DIAGNOSTIC SCREEN FILES FILE SEARCH FILE SEARCH FILE DELETION | 313 314 315 316 317 319 320 321 322 324 325 327 329 329 329 320 321 320 321 322 324 325 327 329 330 330 332 330 332 330 332 330 331 |
| 6. 7. 8. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 ALL 7.1 7.2 DA 8.1 8.2 8.3 8.4 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN ADRY RUN SINGLE BLOCK SAFETY FUNCTIONS SAFETY FUNCTIONS< | 313 314 314 315 316 317 319 320 321 322 324 325 327 329 330 330 332 333 334 |
| 6. 7. 8. | 5.1 5.2 5.3 5.4 5.5 SA 6.1 6.2 6.3 ALL 7.1 7.2 DA 8.1 8.2 8.3 8.4 | MACHINE LOCK AND AUXILIARY FUNCTION LOCK FEEDRATE OVERRIDE RAPID TRAVERSE OVERRIDE DRY RUN SINGLE BLOCK SAFETY FUNCTIONS EMERGENCY STOP OVERTRAVEL STROKE CHECK ALARM AND SELF-DIAGNOSIS FUNCTIONS ALARM DISPLAY CHECKING BY SELF-DIAGNOSTIC SCREEN FILES FILE SEARCH FILE SEARCH FILE DELETION PROGRAM INPUT/OUTPUT 8.4.1 Inputting a Program 8.4.2 Outputting a Program | 313 314 314 315 316 317 319 320 321 322 324 325 327 324 325 327 329 330 332 332 333 334 334 334 |

| | | 8.5.1 | Inputting Offset Data | 339 |
|------------|-----|----------------|---|------------|
| | | 8.5.2 | Outputting Offset Data | 340 |
| 8. | 6 | INPUT AND P | TING AND OUTPUTTING PARAMETERS ITCH ERROR COMPENSATION DATA | 341 |
| | | 8.6.1 | Inputting Parameters | 341 |
| | | 8.6.2 | Outputting Parameters | 342 |
| 8. | 7 | INPUT | TING/OUTPUTTING CUSTOM MACRO B COMMON VARIABLES | 343 |
| | | 8.7.1 | Inputting Custom Macro B Common Variables | 343 344 |
| о г | וחי | | | 3 A E |
| 9. E | וט | TING F | 2RUGRAM5 | 345 |
| 9. | 1 | INSER | TING, ALTERING AND DELETING A WORD | 346 |
| | | 9.1.1 | Word Search | 348 |
| | | 9.1.2 | Heading a Program | 350 |
| | | 9.1.3 | Inserting a Word | 351 |
| | | 9.1.4 | Altering a Word | 352 |
| | | 9.1.5 | Deleting a Word | 353 |
| 9. | 2 | DELET | TNG BLOCKS | 354 |
| | | 9.2.1 | Deleting a Block | 354 |
| | | 9.2.2 | Deleting Multiple Blocks | 355 |
| 9. | 3 | PROGE | RAM NUMBER SEARCH | 356 |
| 9. | 4 | DELET | TING PROGRAMS | 357 |
| | • | 0/1 | Delating One Program | 357 |
| | | 942 | Deleting All Programs | 357 |
| 0 | 5 | D.T.Z | | 259 |
| 9. | 5 | EATER | | 250 |
| | | 9.5.1 | Copying an Entire Program | 359 |
| | | 9.5.2 | Copying Part of a Program | 360 |
| | | 9.5.5 | Moving Part of a Program | 262 |
| | | 9.5.4 | Supplementary Explanation for Copying Moving and Marging | 362 |
| | | 9.5.5 | Paplacement of Words and Addresses | 364 |
| 0 | 6 | DACV | | 266 |
| 9. | 0 | DACK | | 300 |
| 9. | 7 | REORO | GANIGING MEMORY | 368 |
| 10.C | R | EATING | G PROGRAMS | 369 |
| 10 |).1 | CREAT | TNG PROGRAMS USING THE MDI PANEL | 370 |
| 10 | 12 | | MATIC INSERTION OF SEQUENCE NUMBERS | 371 |
| П | J.2 | AUTO | WATE INSERTION OF SEQUENCE NOMBERS | 571 |
| 11.S | E1 | | AND DISPLAYING DATA | 373 |
| 11 | 1.1 | SCREE | ENS DISPLAYED BY FUNCTION KEY POS | 381 |
| | | 11.1.1 | Position Display in the Work Coordinate System | 381 |
| | | 11.1.2 | Position Display in the Relative Coordinate System | 382 |
| | | 11.1.3 | Overall Position Display | 384 |
| | | 11.1.4 | Actual Feedrate Display | 385 |
| | | 11.1.5 | Display of Run Time and Parts Count | 386 |
| | | 11.1.6 | Operating Monitor Display | 387 |
| 11 | .2 | SCREE | $ \text{ENS DISPLAYED BY FUNCTION KEY} PRGRM (IN AUTO MODE OR MDI MODE) \dots $ | 389 |
| | | 11.2.1 | Program Contents Display | 389 |
| | | 11.2.2 | Current Block Display Screen | 390 |
| | | 11.2.3 | Next Block Display Screen | 391 |

| | | 11.2.4 11.2.5 | Program Check Screen Program Screen for MDI Operation | 392 393 |
|--|---|---|---|---|
| | 11.3 | SCRE | ENS DISPLAYED BY FUNCTION KEY RGRM (IN THE EDIT MODE) | 394 |
| | | 11.3.1 | Displaying Memory Used and a List of Programs | 394 |
| | 11.4 | SCRE | ENS DISPLAYED BY FUNCTION KEY | 397 |
| | | 11.4.1 11.4.2 11.4.3 | Setting and Displaying the Tool Offset Value Displaying and Setting the Workpiece Origin Offset Value Displaying and Setting Custom Macro Common Variables | 397 399 400 |
| | 11.5 | SCRE | ENS DISPLAYED BY FUNCTION KEY | 401 |
| | | 11.5.1 11.5.2 11.5.3 11.5.4 | Displaying and Setting Parameters Displaying and Setting Pitch Error Compensation Data Displaying and Entering Setting Data Displaying and Setting Run Time,Parts Count | 402 404 409 411 |
| | 11.6 | SCRE | ENS DISPLAYED BY FUNCTION KEY | 413 |
| | 11.7 | 11.6.1 11.6.2 DISPL | Displaying Operator Message Displaying and Setting the Software Operator's Panel AYING THE PROGRAM NUMBER. SEOUENCE NUMBER. | 413 414 |
| | | AND S | STATUS, AND WARNING MESSAGES FOR DATA SETTING | 416 |
| | | 11.7.1 11.7.2 | Displaying the Program Number and Sequence Number | 416 417 |
| | | | | |
| IV M | AIN | TENA | NCE | 419 |
| IV M | AIN MF | TENA | NCE | 419 421 |
| IV M 1. | AIN ME | TENA THOD REPLA | NCE OF REPLACING BATTERY ACING CNC BATTERY FOR MEMORY BACK–UP | 419 421 422 |
| IV M 1. | AIN ME 1.1 1.2 | TENA THOD REPLA REPLA | NCE | 419 421 422 423 |
| IV M 1. APP | AIN ME 1.1 1.2 ENI | TENA THOD REPLA REPLA | NCE OF REPLACING BATTERY ACING CNC BATTERY FOR MEMORY BACK–UP ACING BATTERIES FOR ABSOLUTE PULSE CODER | 419 421 422 423 |
| IV M 1. APP A | AIN ME 1.1 1.2 ENI . TAI | TENA THOD REPLA REPLA DIX PE COI | NCE OF REPLACING BATTERY ACING CNC BATTERY FOR MEMORY BACK–UP ACING BATTERIES FOR ABSOLUTE PULSE CODER DE LIST | 419 421 422 423 427 |
| IV M 1. APP A B | AIN ME 1.1 1.2 ENI ENI . TAI | TENA THOD REPLA REPLA DIX PE COI | NCE OF REPLACING BATTERY ACING CNC BATTERY FOR MEMORY BACK-UP ACING BATTERIES FOR ABSOLUTE PULSE CODER DE LIST FUNCTIONS AND TAPE FORMAT | 419 421 422 423 427 430 |
| IV M 1. APP A B C | AIN ME 1.1 1.2 ENI ENI . TAI . LIS . RA | TENA THOD REPLA REPLA DIX PE COI ST OF F | NCE OF REPLACING BATTERY ACING CNC BATTERY FOR MEMORY BACK-UP ACING BATTERIES FOR ABSOLUTE PULSE CODER DE LIST FUNCTIONS AND TAPE FORMAT F COMMAND VALUE | 419 421 422 423 427 430 434 |
| IV M 1. APP A B C D | AIN ME 1.1 1.2 ENI ENI . TAI . LIS . RA . NO D.1 D.2 | TENA THOD REPLA REPLA DIX PE COI ST OF F NGE O MOGR TOOL RADIU | NCE OF REPLACING BATTERY ACING CNC BATTERY FOR MEMORY BACK-UP ACING BATTERIES FOR ABSOLUTE PULSE CODER DE LIST CUNCTIONS AND TAPE FORMAT F COMMAND VALUE APHS PATH AT CORNER JS DIRECTION ERROR AT CIRCLE CUTTING | 419 421 422 423 427 430 434 437 438 441 |
| IV M 1. APP A C D | AIN ME 1.1 1.2 ENI ENI AIS A A A A A A A A A A A A A A A A A | TENA THOD REPLA REPLA DIX PE COI ST OF F NGE O MOGR TOOL RADIU | NCEOF REPLACING BATTERY | 419 421 422 423 427 430 434 437 438 441 442 |
| IV M 1. APP A B C D E F. | AIN ME 1.1 1.2 ENI CH | TENA THOD REPLA REPLA DIX PE COI ST OF F NGE O MOGR TOOL RADIU ATUS V ARACT | NCEOF REPLACING BATTERY | 419 421 422 423 427 430 434 437 438 441 442 444 |

H. OPERATION OF PORTABLE TAPE READER 461

I. GENERAL

GENERAL

About this manual

I. GENERAL

Describes chapter organization, applicable models, related manuals, and notes for reading this manual.

II. PROGRAMMING

Describes each function: Format used to program functions in the NC language, characteristics, and restrictions.

III. OPERATION

Describes the manual operation and automatic operation of a machine, procedures for inputting and outputting data, and procedures for editing a program.

IV. MAINTENANCE

Describes procedure for batteries.

This manual consists of the following parts:

APPENDIX

Lists tape codes, valid data ranges, and error codes.

This manual does not describe parameters in detail. For details on parameters mentioned in this manual, refer to the manual for parameters (B–62580EN).

This manual describes all optional functions. Look up the options incorporated into your system in the manual written by the machine tool builder.

The models covered by this manual, and their abbreviations are:

Applicable models

| Product name | Ab | obreviations | |
|--------------------|-------|--------------|--|
| FANUC Series 0-MD | 0-MD | 0-D | |
| FANUC Series 0–GSD | 0–GSD | | |

Special symbols

This manual uses the following symbols:

- ; : Indicates the end of a block. It actually corresponds to the ISO code LF or EIA code CR.

— 3 —

Related manuals

The table below lists manuals related to the FANUC Series 0–D. In the table, this manual is marked with an asterisk (*).

| Table 1 | Manuals | Related to | the FANUC | Series 0–D |
|---------|---------|------------|-----------|------------|

| Manual name | Specification number | |
|--|-------------------------|---|
| FANUC Series 0–TD/MD/GCD/GSD CONNECTION MANUAL (HARDWARE) | B–62543EN | |
| FANUC Series 0–TD/MD/GCD/GSD CONNECTION MANUAL (FUNCTION) | B-62543EN-1 | |
| FANUC Series 0-TD/GCD OPERATOR'S MANUAL | B–62544EN | |
| FANUC Series 0-MD/GSD OPERATOR'S MANUAL | B–62574EN | * |
| FANUC Series 0–TD/MD/GCD/GSD MAINTENANCE MANUAL | B–62545EN | |
| FANUC Series 0-TD/GCD PARAMETER MANUAL | B-62550EN | |
| FANUC Series 0-MD/GSD PARAMETER MANUAL | B-62580EN | |

— 4 —

1.1 GENERAL FLOW OF OPERATION OF CNC MACHINE TOOL

When machining the part using the CNC machine tool, first prepare the program, then operate the CNC machine by using the program.

- First, prepare the program from a part drawing to operate the CNC machine tool. Store the program to a media appropriate for the CNC. How to prepare the program is described in the Chapter II. PROGRAMMING.
- The program is to be read into the CNC system. Then, mount the workpieces and tools on the machine, and operate the tools according to the programming. Finally, execute the machining actually. How to operate the CNC system is described in the Chapter III.



Before the actual programming, make the machining plan for how to machine the part.

Machining plan

- 1. Determination of workpieces machining range
- 2. Method of mounting workpieces on the machine tool
- **3.** Machining sequence in every cutting process
- 4. Cutting tools and cutting conditions

Decide the cutting method in every cutting process.

| Cutting process | 1 | 2 | 3 |
|--|--------------|--------------|---------------------|
| Cutting procedure | Feed cutting | Side cutting | Hole machin- ing |
| 1. Cutting method : Rough Semi Finish | | | |
| 2. Cutting tools | | | |
| 3. Cutting conditions : Feedrate Cutting depth | | | |
| 4. Tool path | | | |



Prepare the program of the tool path and machining condition according to the workpiece figure, for each machining.

1.2 NOTES ON READING THIS MANUAL

NOTE

- 1 The function of an CNC machine tool system depends not only on the CNC, but on the combination of the machine tool, its magnetic cabinet, the servo system, the CNC, the operator's panels, etc. It is too difficult to describe the function, programming, and operation relating to all combinations. This manual generally describes these from the stand-point of the CNC. So, for details on a particular CNC machine tool, refer to the manual issued by the machine tool builder, which should take precedence over this manual.
- 2 Headings are placed in the left margin so that the reader can easily access necessary information. When locating the necessary information, the reader can save time by searching though these headings.
- 3 Machining programs, parameters, variables, etc. are stored in the CNC unit internal non-volatile memory. In general, these contents are not lost by the switching ON/OFF of the power. However, it is possible that a state can occur where precious data stored in the non-volatile memory has to be deleted, because of deletions from a maloperation, or by a failure restoration. In order to restore rapidly when this kind of mishap occurs, it is recommended that you create a copy of the various kinds of data beforehand.
- 4 This manual describes as many reasonable variations in equipment usage as possible. It cannot address every combination of features, options and commands that should not be attempted.
 - If a particular combination of operations is not described, it should not be attempted.
- 5 This manual describes the functions supported by the software of the following versions and editions of FANUC Series 0–D. These functions may not be supported by other versions or editions of the software.

| Product name | Versions | Editions |
|--------------------|----------|------------|
| FANUC Series 0–MD | 0471 | 04 or more |
| | 0472 | 01 or more |
| FANUC Series 0–GSD | 0891 | 01 or more |

6 This manual provides a general description of the FANUC Series 0–D. Some functions described in this manual may not, therefore, be available depending on the software series or type.

II PROGRAMMING



1.1 TOOL MOVEMENT ALONG WORKPIECE PARTS FIGURE– INTERPOLATION

Explanations

• Tool movement along a straight line

The tool moves along straight lines and arcs constituting the workpiece parts figure (See II–4).





• Tool movement along an arc



Fig. 1.1 (b) Tool movement along an arc

Symbols of the programmed commands G01, G02, ... are called the preparatory function and specify the type of interpolation conducted in the control unit.



Fig. 1.1 (c) Interpolation function

NOTE

Some machines move tables instead of tools but this manual assumes that tools are moved against workpieces.

1.2 FEED– FEED FUNCTION

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



Fig. 1.2 (a) Feed function

Feedrates can be specified by using actual numerics. For example, to feed the tool at a rate of 150 mm/min, specify the following in the program:

F150.0

The function of deciding the feed rate is called the feed function (See II–5).

1.3 PART DRAWING AND TOOL MOVEMENT

1.3.1 Reference Position (Machine–Specific Position)

A CNC machine tool is provided with a fixed position. Normally, tool change and programming of absolute zero point as described later are performed at this position. This position is called the reference position.



Fig. 1.3.1 (a) Reference position

Explanations

The tool can be moved to the reference position in two ways:

- Manual reference position return (See III–3.1) Reference position return is performed by manual button operation.
- (2) Automatic reference position return (See II–6) In general, manual reference position return is performed first after the power is turned on. In order to move the tool to the reference position for tool change thereafter, the function of automatic reference position return is used.

— 15 —

1.3.2 Coordinate System on Part Drawing and Coordinate System Specified by CNC – Coordinate System



Fig. 1.3.2 (a) Coordinate system

Explanations

• Coordinate system

The following two coordinate systems are specified at different locations: (See II–7)

- (1) Coordinate system on part drawing The coordinate system is written on the part drawing. As the program data, the coordinate values on this coordinate system are used.
- (2) Coordinate system specified by the CNC The coordinate system is prepared on the actual machine tool table. This can be achieved by programming the distance from the current position of the tool to the zero point of the coordinate system to be set.





The positional relation between these two coordinate systems is determined when a workpiece is set on the table.



Fig. 1.3.2 (c) Coordinate system specified by CNC and coordinate systemon part drawing

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

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

 Methods of setting the two coordinate systems in the same position To set the two coordinate systems at the same position, simple methods shall be used according to workpiece shape, the number of machinings.

(1) Using a standard plane and point of the workpiece.



(2) Mounting a workpiece directly against the jig



(3) Mounting a workpiece on a pallet, then mounting the workpiece and pallet on the jig



1.3.3 How to Indicate Command Dimensions for Moving the Tool – Absolute, Incremental Commands

Explanations

• Absolute coordinates

Coordinate values of command for moving the tool can be indicated by absolute or incremental designation (See II–8.1).

The tool moves to a point at "the distance from zero point of the coordinate system" that is to the position of the coordinate values.



Incremental coordinates

Specify the distance from the previous tool position to the next tool position.


1.4 CUTTING SPEED – SPINDLE SPEED FUNCTION

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

As for the CNC, the cutting speed can be specified by the spindle speed in rpm unit.



Examples

<When a workpiece should be machined with a tool 100 mm in diameter at a cutting speed of 80 mm/min. >

The spindle speed is approximately 250 rpm, which is obtained from N=1000v/ π D. Hence the following command is required:

S250;

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

(See II–9).

1.5 SELECTION OF TOOL USED FOR VARIOUS MACHINING – TOOL FUNCTION

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



Examples

<When No.01 is assigned to drilling tool>

When the tool is stored at location 01 in the ATC magazine, the tool can be selected by specifying T01. This is called the tool function (See II–10).

1.6 COMMAND FOR MACHINE OPERATIONS – MISCELLANEOUS FUNCTION

When machining is actually started, it is necessary to rotate the spindle, and feed coolant. For this purpose, on–off operations of spindle motor and coolant valve should be controlled (See II–11).



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

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

1.7 PROGRAM CONFIGURATION

A group of commands given to the CNC for operating the machine is called the program. By specifying the commands, the tool is moved along a straight line or an arc, or the spindle motor is turned on and off.

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



Fig. 1.7 (a) Program configuration

A group of commands at each step of the sequence is called the block. The program consists of a group of blocks for a series of machining. The number for discriminating each block is called the sequence number, and the number for discriminating each program is called the program number (See II–12).

Explanations

Block

The block and the program have the following configurations.



Fig. 1.7 (b) Block configuration

A block has a sequence number at its head, which identifies the block, and an end–of–block code at the end, indicating the end of the block. This manual indicates the end–of–block code by ; (LF in the ISO code and CR in the EIA code).

• Program



Fig. 1.7 (c) Program configuration

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

Main program and subprogram

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





1.8 TOOL FIGURE AND TOOL MOTION BY PROGRAM

Explanations

 Machining using the end of cutter – Tool length compensation function (See II–14.1) Usually, several tools are used for machining one workpiece. The tools have different tool length. It is very troublesome to change the program in accordance with the tools.

Therefore, the length of each tool used should be measured in advance. By setting the difference between the length of the standard tool and the length of each tool in the CNC (data display and setting : see III–11), machining can be performed without altering the program even when the tool is changed. This function is called tool length compensation.



 Machining using the side of cutter – Cutter compensation function (See II–14.2) Because a cutter has a radius, the center of the cutter path goes around the workpiece with the cutter radius deviated.



If radius of cutters are stored in the CNC (Data Display and Setting : see III–11), the tool can be moved by cutter radius apart from the machining part figure. This function is called cutter compensation.

1.9 TOOL MOVEMENT RANGE – STROKE

Limit switches are installed at the ends of each axis on the machine to prevent tools from moving beyond the ends. The range in which tools can move is called the stroke.



Besides strokes defined with limit switches, the operator can define an area which the tool cannot enter using a program or data in memory (see III–11). This function is called stroke check.



2.1 CONTROLLED AXES

| | 0-MD | 0-GSD |
|---|-------------|-------------|
| No. of basic controlled axes | 3 axes | 3 axes |
| Controlled axes expansion (PMC axis is not included.) | Max. 4 axes | Max. 4 axis |
| Basic simultaneously con- trolled axes | 4 axes | 3 axes |
| Simultaneously controlled axes expansion | Max. 4 axes | Max. 3 axes |

2.2 NAME OF AXES

Names of the three basic axes are fixed as X, Y, and Z. Names of additional axes can be optionally selected from A, B, C, U, V, and W. They can be set by parameter No. 008 #2, #3, #4.

2.3 INCREMENT SYSTEM

| Name of incre- ment system | Least input increment | Least command increment | Maximum stroke |
|-------------------------------|-----------------------|----------------------------|----------------|
| IS-B | 0.001mm | 0.001mm | 99999.999mm |
| | 0.0001inch | 0.0001inch | 9999.9999inch |
| | 0.001deg | 0.001deg | 99999.999deg |

| Name of incre- ment system | Least input increment | Least command increment | Maximum stroke |
|-------------------------------|-----------------------|----------------------------|----------------|
| IS-C | 0.0001mm | 0.0001mm | 9999.9999mm |
| | 0.00001inch | 0.00001inch | 999.99999inch |
| | 0.00001deg | 0.00001deg | 9999.9999deg |

Combined use of the inch system and the metric system is not allowed. There are functions that cannot be used between axes with different unit systems (circular interpolation, cutter compensation, etc.). For the increment system, see the machine tool builder's manual.

2.4 MAXIMUM STROKE

Maximum stroke = Least command increment × 99999999 See 2.3 Incremen System.



REPARATORY FUNCTION (G FUNCTION)

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

G codes are divided into the following two types.

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

(Example)

G01 and G00 are modal G codes in group 01.

$$\begin{array}{c} \text{G01X-;} \\ \text{Z-;} \\ \text{X-;} \end{array} \end{array} \begin{array}{c} \text{G01 is effective in this range.} \\ \text{G00Z-;} \end{array}$$

Explanations

- 1. Modal G codes have the following initial conditions when the power is turned on or the system is reset to the clear state (bit 6 of parameter No. 045).
 - 1) Those G codes marked **r** in Table 3 are specified automatically.
 - 2) G20 and G21 retain their original conditions.
 - 3) G00 or G01 is automatically selected depending on the setting of bit 6 of parameter No. 011.
 - 4) G90 or G91 is automatically selected depending on the setting of bit 7 of parameter No. 030.
- 2. The G codes of group 00, except G10 and G11, are one-shot G codes.
- 3. If a G code that does not appear in the G code list is specified, or a G code whose options are not supported is specified, alarm No. 010 is displayed.
- 4. Multiple G codes of different groups can be specified in a single block. When multiple G codes of one group are specified in a block, the G code specified last is effective.
- 5. If any G code of group 01 is specified in a canned cycle mode, the canned cycle is automatically cancelled and the G80 condition is entered. However, a G code of group 01 is not affected by any of the canned cycle G codes.
- 6. A G code is displayed from each group.

| G code | Group | Function | |
|--------|-------|---|---------------------------------|
| G00 | | Positioning | |
| G01 | 01 | Linear interpolation | |
| G02 | | Circular interpolation CW | |
| G03 | | Circular interpolation CCW | |
| G04 | | Dwell, Exact stop | |
| G09 | | Exact stop | |
| G10 | | Data setting | |
| G11 | | Data setting mode cancel | |
| G17 | 02 | XpYp plane selection | Xp: X axis or its parallel axis |
| G18 | 02 | ZpXp plane selection | Yp: Y axis or its parallel axis |
| G19 | 02 | YpZp plane selection | Zp: Z axis or its parallel axis |
| G20 | 06 | Input in inch | • |
| G21 | 00 | Input in mm | |
| G27 | | Reference position return | check |
| G28 | | Return to reference position | n |
| G29 | 00 | Return from reference pos | sition |
| G30 | | 2nd reference position ret | urn |
| G31 | | Skip function (0–GSD) | |
| G40 | | Cutter compensation cand | cel |
| G41 | 07 | Cutter compensation left | |
| G42 | | Cutter compensation right | |
| G43 | | Tool length compensation | + direction |
| G44 | 08 | Tool length compensation | - direction |
| G49 | | Tool length compensation | cancel |
| G52 | 00 | Local coordinate system s | setting |
| G53 | | Machine coordinate system selection | |
| G54 | | Workpiece coordinate system 1 selection | |
| G55 | | Workpiece coordinate sys | tem 2 selection |
| G56 | 14 | Workpiece coordinate sys | tem 3 selection |
| G57 | 1 14 | Workpiece coordinate sys | tem 4 selection |
| G58 | | Workpiece coordinate sys | tem 5 selection |
| G59 | | Workpiece coordinate system 6 selection | |
| G60 | 00 | Single direction positioning | |
| G61 | | Exact stop mode | |
| G62 | | Automatic corner override | |
| G63 | 15 | Tapping mode | |
| G64 | 1 | Cutting mode | |
| G65 | 00 | Macro call | |

Table 3 G code list (1/2)

| G code | Group | Function |
|--------|-------|--|
| G66 | | Macro modal call |
| G67 | 12 | Macro modal call cancel |
| G73 | 00 | Peck drilling cycle |
| G74 | - 09 | Counter tapping cycle |
| G75 | 01 | Plunge grinding cycle (0–GSD) |
| G76 | 09 | Fine boring cycle |
| G77 | | Direct constant-dimension plunge grinding cycle (0-GSD) |
| G78 | 01 | Continuous-feed surface grinding cycle (0-GSD) |
| G79 | | Intermittent-feed surface grinding cycle (0-GSD) |
| G80 | | Canned cycle cancel/external operation function cancel |
| G81 | | Drilling cycle, spot boring cycle or external operation function |
| G82 | | Drilling cycle or counter boring cycle |
| G83 | | Peck drilling cycle |
| G84 | 09 | Tapping cycle |
| G85 | | Boring cycle |
| G86 | 1 | Boring cycle |
| G87 | 1 | Back boring cycle |
| G88 | | Boring cycle |
| G89 | | Boring cycle |
| G90 | 03 | Absolute command |
| G91 | 00 | Increment command |
| G92 | 00 | Setting for work coordinate system or clamp at maximum spindle speed |
| G94 | 05 | Feed per minute |
| G98 | 10 | Return to initial point in canned cycle |
| G99 | | Return to R point in canned cycle |
| G150 | | Normal direction control cancel mode (0–GSD) |
| G151 | 19 | Normal direction control left side on (0–GSD) |
| G152 | | Normal direction control right side on (0–GSD) |
| G160 | 20 | In-feed control function cancel (0-GSD) |
| G161 | | In–feed control function (0–GSD) |

Table 3 G code list (2/2)



INTERPOLATION FUNCTIONS

| 4.1 Positioning (G00) | The G00 command moves a tool to the position in the workpiece system specified with an absolute or an incremental command at a rapid traverse rate. In the absolute command, coordinate value of the end point is programmed. In the incremental command the distance the tool moves is programmed. |
|-----------------------------|--|
| Format | |

G00 I₽_;

IP_: For an absolute command, the coordinates of an end position, and for an incremental command, the distance the tool moves.

Tool path generally does not become a straight line.



The rapid traverse rate in the G00 command is set to the parameter No. 518 to 521 for each axis independently by the machine tool builder. In the positioning mode actuated by G00, the tool is accelerated to a predetermined speed at the start of a block and is decelerated at the end of a block. Execution proceeds to the next block after confirming the in-position.

"In-position" means that the feed motor is within the specified range. This range is determined by the machine tool builder by setting to parameter No. 500 to 503.

Restrictions

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

Explanations

4.2 SINGLE DIRECTION POSITIONING (G60)

For accurate positioning without play of the machine (backlash), final positioning from one direction is available.



Format

G60 I₽_;

₱_: For an absolute command, the coordinates of an end position, and for an incremental command, the distance the tool moves.

Explanations

Restrictions

- An overrun and a positioning direction are set by the parameter (No. 029 #0 to #3, No. 204 to 207). Even when a commanded positioning direction coincides with that set by the parameter, the tool stops once before the end point.
- During drilling canned cycle, no single direction positioning is effected in Z axis.
- No single direction positioning is effected in an axis for which no overrun has been set by the parameter.
- When the move distance 0 is commanded, the single direction positioning is not performed.
- The direction set to the parameter is not effected by mirror image.
- The single direction positioning does not apply to the shift motion in the canned cycles of G76 and G87.

4.3 LINEAR INTERPOLATION (G01)

Format

Tools can move along a line

G01 IP_F_;

- IP_: For an absolute command, the coordinates of an end point, and for an incremental command, the distance the tool moves.
- F_:Speed of tool feed (Feedrate)

Explanations

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

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

The feedrate commanded by the F code is measured along the tool path. If the F code is not commanded, the feedrate is regarded as zero. The feedrate of each axis direction is as follows.

```
G01 \alpha \alpha \beta \beta \gamma \gamma \zeta \zeta Ff_{\pm};

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

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

Feed rate of \gamma axis direction : F\gamma = \frac{\gamma}{L} \times f

Feed rate of \zeta axis direction : F_{\zeta} = \frac{\zeta}{L} \times f

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

The feed rate of the rotary axis is commanded in the unit of deg/min (the unit is decimal point position).

When the straight line axis α (such as X, Y, or Z) and the rotating axis β (such as A, B, or C) are linearly interpolated, the feed rate is that in which the tangential feed rate in the α and β cartesian coordinate system is commanded by F(mm/min).

 β -axis feedrate is obtained ; at first, the time required for distribution is calculated by using the above formula, then the β -axis feedrate unit is changed to deg 1min.

A calculation example is as follows. G91 G01 X20.0B40.0 F300.0 ;

This changes the unit of the C axis from 40.0 deg to 40mm with metric input. The time required for distribution is calculated as follows:

| $\frac{\sqrt{20^2 + 40^2}}{300}$ | ≒ 0.14907 (min) |
|----------------------------------|-------------------|
| The feed rate | for the C axis is |
| <u>40</u> 0.14907 | ≒ 268.3 deg/min |

In simultaneous 3 axes control, the feed rate is calculated the same way as in 2 axes control.

Examples

• Linear interpolation



• Feedrate for the rotation axis



4.4 CIRCULAR INTERPOLATION (G02,G03)

Format

Arc in the XpYp plane

$$G17 \quad \begin{cases} G02\\G03 \end{cases} Xp_Yp_- \quad \begin{cases} I_J_-\\R_- \end{cases} F_-;$$
Arc in the ZpXp plane

$$G18 \quad \begin{cases} G02\\G03 \end{cases} Xp_Zp_- \quad \begin{cases} I_K_-\\R_- \end{cases} F_-;$$
Arc in the YpZp plane

$$G19 \quad \begin{cases} G02\\G03 \end{cases} Yp_Zp_- \quad \begin{cases} J_K_-\\R_- \end{cases} F_-;$$

The command below will move a tool along a circular arc.

Table.4.4 Description of the Command Format

| Command | Description |
|-----------------|---|
| G17 | Specification of arc on XpYp plane |
| G18 | Specification of arc on ZpXp plane |
| G19 | Specification of arc on YpZp plane |
| G02 | Circular Interpolation Clockwise direction (CW) |
| G03 | Circular Interpolation Counterclockwise direction (CCW) |
| X _{p_} | Command values of X axis or its parallel axis |
| Y _{p_} | Command values of Y axis or its parallel axis |
| Z _{p_} | Command values of Z axis or its parallel axis |
| I_ | $X_{p} \text{axis}$ distance from the start point to the center of an arc with sign |
| J_ | Y_{p} axis distance from the start point to the center of an arc with sign |
| k | $Z_{\mbox{\scriptsize p}}$ axis distance from the start point to the center of an arc with sign |
| R_ | Arc radius with sign fixed to radius designation. |
| F_ | Feedrate along the arc |

Explanations

 Direction of the circular interpolation

"Clockwise"(G02) and "counterclockwise"(G03) on the X_pY_p plane (Z_pX_p plane or Y_pZ_p plane) are defined when the X_pY_p plane is viewed in the positive-to-negative direction of the Z_p axis (Y_p axis or X_p axis, respectively) in the Cartesian coordinate system. See the figure below.



- Distance moved on an arc
- Distance from the start point to the center of arc

The end point of an arc is specified by address Xp, Yp or Zp, and is expressed as an absolute or incremental value according to G90 or G91. For the incremental value, the distance of the end point which is viewed from the start point of the arc is specified.

The arc center is specified by addresses I, J, and K for the Xp, Yp, and Zp axes, respectively. The numerical value following I, J, or K, however, is a vector component in which the arc center is seen from the start point, and is always specified as an incremental value irrespective of G90 and G91, as shown below.

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



I0,J0, and K0 can be omitted. When X_p , Y_p , and Z_p are omitted (the end point is the same as the start point) and the center is specified with I, J, and K, a 360° arc (circle) is specified.

G02I; Command for a circle

If the difference between the radius at the start point and that at the end point exceeds the value in a parameter (No.876), an alarm (No.020) occurs. (Valid only when bit 6 of parameter No. 393 is set to 1.)

Arc radius

The distance between an arc and the center of a circle that contains the arc can be specified using the radius, R, of the circle instead of I, J, and K. In this case, one arc is less than 180° , and the other is more than 180° are considered. When an arc exceeding 180° is commanded, the radius must be specified with a negative value. If Xp, Yp, and Zp are all omitted, if the end point is located at the same position as the start point and when R is used, an arc of 0° is programmed G02R ; (The cutter does not move.)



The feedrate in circular interpolation is equal to the feed rate specified by the F code, and the feedrate along the arc (the tangential feedrate of the arc) is controlled to be the specified feedrate.

The error between the specified feedrate and the actual tool feedrate is $\pm 2\%$ or less. However, this feed rate is measured along the arc after the cutter compensation is applied

Restrictions

Feedrate

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

If an axis not comprising the specified plane is commanded, an alarm is displayed.

For example, if axis U is specified as a parallel axis to X axis when plane XY is specified, an alarm (No.028)is displayed.

When an arc having a center angle close to 180ø is specified using its radius R, the system may fail to calculate the center of the arc correctly. Therefore, specify the arc with I, J, and K.

Examples



The above tool path can be programmed as follows ;

(1) In absolute programming

G92X200.0 Y40.0 Z0 ; G90 G03 X140.0 Y100.0R60.0 F300.; G02 X120.0 Y60.0R50.0 ; or G92X200.0 Y40.0Z0 ; G90 G03 X140.0 Y100.0I-60.0 F300.; G02 X120.0 Y60.0I-50.0 ;

(2) In incremental programming

G91 G03 X-60.0 Y60.0 R60.0 F300.; G02 X-20.0 Y-40.0 R50.0 ; or G91 G03 X-60.0 Y60.0 I-60.0 F300. ; G02 X-20.0 Y-40.0 I-50.0 ;

| 4.5 SKIP FUNCTION (G31) (FOR 0–GSD ONLY) | Linear interpolation can be commanded by specifying axial move following the G31 command, like G01. If an external skip signal is input during the execution of this command, execution of the command is interrupted and the next block is executed. The skip function is used when the end of machining is not programmed but specified with a signal from the machine, for example, in grinding. It is used also for measuring the dimensions of a workpiece. | |
|--|--|--|
| Format | | |
| | G31 ₽_; | |
| | G31: One-shot G code (If is effective only in the block in which it is specified) | |
| Explanations | The coordinate values when the skip signal is turned on can be used in a | |
| | custom macro because they are stored in the custom macro system variable #5061 to #5064, as follows: | |

- #5061 X axis coordinate value#5062 Y axis coordinate value
- #5063 Z axis coordinate value

WARNING

Disable feedrate override, dry run, and automatic acceleration/deceleration (These operations can be validated by setting bit 3 of parameter No. 015 to 1.) when the feedrate per minute is specified, allowing for an error in the position of the tool when a skip signal is input. These functions are enabled when the feedrate per rotation is specified.

NOTE

If G31 command is issued while cutter compensation C is applied, an P/S alarm of No.035 is displayed. Cancel the cutter compensation with the G40 command before the G31 command is specified.

— 43 —

Examples

 The next block to G31 is an incremental command



Fig.4.5 (a) The next block is an incremental command

• The next block to G31 is an absolute command for 1 axis





• The next block to G31 is an absolute command for 2 axes



Fig 4.5 (c) The next block is an absolute command for 2 axes



| 5.1 GENERAL | The feed functions control the feedrate of the tool. The following two feed functions are available: |
|--|---|
| Feed functions | Rapid traverse When the positioning command (G00) is specified, the tool moves at a rapid traverse feedrate set in the CNC (parameter No. 518 to 521). Cutting feed The tool moves at a programmed cutting feedrate. |
| Override | Override can be applied to a rapid traverse rate or cutting feedrate using the switch on the machine operator's panel. |
| Automatic acceleration/ deceleration | To prevent a mechanical shock, acceleration/deceleration is automatically applied when the tool starts and ends its movement (Fig. 5.1 (a)). |



Fig. 5.1 (a) Automatic acceleration/deceleration (example)

— 46 —

Tool path in a cutting feed

If the direction of movement changes between specified blocks during cutting feed, a rounded–corner path may result (Fig. 5.1 (b)).



Fig. 5.1 (b) Example of Tool Path between Two Blocks

In circular interpolation, a radial error occurs (Fig. 5.1(c)).



Fig. 5.1 (c) Example of Radial Error in Circular Interpolation

The rounded–corner path shown in Fig. 5.1(b) and the error shown in Fig. 5.1(c) depend on the feedrate. So, the feedrate needs to be controlled for the tool to move as programmed.

5.2 RAPID TRAVERSE

Format

G00 IP_;

G00 : G code (group 01) for positioning (rapid traverse) ${\rm I\!P}_-$; Dimension word for the end point

Explanations

The positioning command (G00) positions the tool by rapid traverse. In rapid traverse, the next block is executed after the specified feedrate becomes 0 and the servo motor reaches a certain range set by the machine tool builder (in–position check). (In–position check can be disabled for each block by setting bit 5 of parameter No. 020 to 1.)

A rapid traverse rate is set for each axis by parameter No. 518 to 521, so no rapid traverse feedrate need be programmed.

The following overrides can be applied to a rapid traverse rate with the switch on the machine operator's panel:F0, 25, 50, 100%

F0: Allows a fixed feedrate to be set for each axis by parameter No. 533. For detailed information, refer to the appropriate manual of the machine tool builder.

Command value range of rapid traverse rate

| | Increment system | |
|---------------|---|--|
| | IS–B | IS-C |
| Metric output | 30 to 100,000 mm/min 30 to 100,000 deg/min | 30 to 24,000 mm/min 30 to 24.000 deg/min |
| Inch output | 3.0 to 4,000.0 inch/min 3.0 to 100,000 deg/min | 3.0 to 960.0 inch/min 3.0 to 24,000 deg/min |

5.3 Feedrate of linear interpolation (Getc. are commanded with numbers In cutting feed, the next block is ext the previous block is minimized.

Format

Feedrate of linear interpolation (G01), circular interpolation (G02, G03), etc. are commanded with numbers after the F code. In cutting feed, the next block is executed so that the feedrate change from

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

Feed per minute G94; G code (group 05) for feed per minute

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

Explanations

 Tangential speed constant control Cutting feed is controlled so that the tangential feedrate is always set at a specified feedrate.



Fig. 5.3 (a) Tangential feedrate (F)

• Feed per minute (G94)

After specifying G94 (in the feed per minute mode), the amount of feed of the tool per minute is to be directly specified by setting a number after F. G94 is a modal code. At power–on, the feed per minute mode is set. An override from 0% to 150% (in 10% steps) can be applied to feed per minute with the switch on the machine operator's panel. For detailed information, see the appropriate manual of the machine tool builder.



Fig. 5.3 (b) Feed per minute

WARNING

No override can be used for some commands such as for threading.

— 49 —

< Command value range of feed per minute >

| | Increment system | | |
|--------------|---|---|--|
| | IS-B | IS-C | |
| Metric input | 1 to 100,000 mm/min 1 to 100,000 deg/min | 1 to 12,000 mm/min 1 to 12,000 deg/min | |
| Inch input | 0.01 to 4,000.0 inch/min 0.01 to 6,000.0 deg/min | 0.01 to 480.0 inch/min 0.01 to 600.0 deg/min | |

<Feedrate command value with a decimal fraction for feed per minute> When a value with a decimal fraction is specified in a feedrate command for feed per minute, the value is valid within the following range. To specify a value that falls outside this range, specify an integer.

| | Increment system | | |
|--------------|---|---|--|
| | IS–B | IS-C | |
| Metric input | 0.001 to 99,999.999 mm/min 0.001 to 99,999.999 deg/min | 0.001 to 12,000.000 mm/min 0.001 to 12,000.000 deg/min | |
| Inch input | 0.00001 to 999.99999 inch/min 0.00001 to 999.99999 deg/min | 0.00001 to 480.00000 inch/min 0.00001 to 600.00000 deg/min | |

• Cutting feedrate clamp

A common upper limit can be set on the cutting feedrate along each axis with parameter No. 527. If an actual cutting feedrate (with an override applied) exceeds a specified upper limit, it is clamped to the upper limit.

NOTE

An upper limit is set in mm/min or inch/min. CNC calculation may involve a feedrate error of $\pm 2\%$ with respect to a specified value. However, this is not true for acceleration/deceleration. To be more specific, this error is calculated with respect to a measurement on the time the tool takes to move 500 mm or more during the steady state:

5.4 CUTTING FEEDRATE CONTROL

G code **Function name** Validity of G code Description This function is valid for specified The tool is decelerated at the end point blocks only. of a block, then an in-position check is G09 Exact stop made. Then the next block is executed. Once specified, this function is The tool is decelerated at the end point G61 valid until G63 or G64 is specified. of a block, then an in-position check is Exact stop made. Then the next block is executed. The tool is not decelerated at the end Once specified, this function is valid until G61 or G63 is specified. point of a block, but the next block is G64 Cutting mode executed. The tool is not decelerated at the end Once specified, this function is valid until G61 or G64 is specified. point of a block, but the next block is G63 executed. Tapping mode When G63 is specified, feedrate override and feed hold are invalid.

Table 5.4(a) Cutting Feedrate Control

Cutting feedrate can be controlled, as indicated in Table 5.4(a).

NOTE

- The purpose of in-position check is to check that the servo motor has reached within a specified range (specified with a parameter by the machine tool builder). In-position check can be disabled by setting bit 5 of parameter No. 020 to 1.
- 2. Inner corner angle θ : $2^{\circ} < \theta \le \alpha \le 178^{\circ}$ (α is a set value)

| | λ_{1} | |
|--------|------------------------|--|
| θ | | |
|) Tool | $\mathbf{O}\mathbf{A}$ | |

| Exact stop | G09 IP_; G61; |
|--------------|------------------|
| Cutting mode | G64 ; |
| Tapping mode | G63 ; |

5.4.1 Exact Stop (G09, G61) Cutting Mode (G64) Tapping Mode (G63)

Explanations

The inter–block paths followed by the tool in the exact stop mode, cutting mode, and tapping mode are different (Fig. 5.4.1 (a)).





CAUTION

The cutting mode (G64 mode) is set at power–on or system clear.

5.4.2 Internal Circular Cutting Feedrate Change

For internally offset circular cutting, the feedrate on a programmed path is set to a specified feedrate (F) by specifying the circular cutting feedrate with respect to F, as indicated below (Fig. 5.4.2(a)). This function is valid in the cutter compensation mode, regardless of the G62 code.

$$F \times \frac{RC}{Rp}$$

- Rc : Cutter center path radius
- Rp : Programmed radius

It is also valid for the dry run and the one-digit F command.



Fig. 5.4.2(a) Internal circular cutting feedrate change

If Rc is much smaller than Rp, $Rc/Rp \doteq 0$; the tool stops. A minimum deceleration ratio (MDR) is to be specified with parameter No. 213. When $Rc/Rp \le MDR$, the feedrate of the tool is (F×MDR).

CAUTION

When internal circular cutting must be performed together with automatic override for inner corners, the feedrate of the tool is as follows:

 $F \times \frac{Rc}{Rp} \times$ (automatic override for the inner corners) ×(feedrate override)

5.5 DWELL (G04)

Format

Dwell

G04 X_; or G04 P_; X_: Specify a time (decimal point permitted)

P_: Specify a time (decimal point not permitted)

Explanations

By specifying a dwell, the execution of the next block is delayed by the specified time. In addition, a dwell can be specified to make an exact check in the cutting mode (G62 mode).

When neither P nor X is specified, exact stop is performed.

 Table 5.5 (a)
 Command value range of the dwell time (Command by X)

| Increment system | Command value range | Dwell time unit |
|------------------|---------------------|-----------------|
| IS–B | 0.001 to 99999.999 | s |
| IS-C | 0.0001 to 9999.9999 | - |

Table 5.5 (b) Command value range of the dwell time (Command by P)

| Increment system | Command value range | Dwell time unit |
|------------------|---------------------|-----------------|
| IS–B | 1 to 99999999 | 0.001 s |
| IS–C | 1 to 99999999 | 0.0001 s |

6 REFERENCE POSITION

General

• Reference position

The reference position is a fixed position on a machine tool to which the tool can easily be moved by the reference position return function. For example, the reference position is used as a position at which tools are automatically changed. Up to two reference positions can be specified by setting coordinates in the machine coordinate system in parameters. The first reference position must be the machine zero point.



Fig. 6 (a) Machine zero point and reference positions

 Reference position return and movement from the reference position Tools are automatically moved to the reference position via an intermediate position along a specified axis. Or, tools are automatically moved from the reference position to a specified position via an intermediate position along a specified axis. When reference position return is completed, the lamp for indicating the completion of return goes on.



Fig. 6 (b) Reference position return and return form the reference position

position along a specified axis, the lamp for the axis goes on.

• **Reference position** return check The reference position return check (G27) is the function which checks whether the tool has correctly returned to the reference position as specified in the program. If the tool has correctly returned to the reference

Format

Reference position
 return

G28 IP_; Reference position return

G30 P2 IP ;2nd reference position return

(P2 can be omitted.)

- IP : Command specifying the intermediate position (Absolute/incremental command)
- Return from reference position

G29 ₽_;

IP : Command specifying the destination of return from reference position (Absolute/incremental command)

• Reference position return check

G27 I₽_;

IP : Command specifying the reference position (Absolute/incremental command)

— 56 —
Explanations

| Reference position return (G28) | Positioning to the intermediate or reference positions are performed at the rapid traverse rate of each axis. Therefore, for safety, the cutter compensation, and tool length compensation should be cancelled before executing this command. The coordinates for the intermediate position are stored in the CNC only for the axes for which a value is specified in a G28 block. For the other axes, the previously specified coordinates are used. Example N1 G28 X40.0 ; Intermediate position (X40.0) N2 G28 Y60.0 ; Intermediate position (X40.0, Y60.0) |
|---|---|
| 2nd reference position return (G30) | In a system without an absolute–position detector, the second reference position return functions can be used only after the reference position return (G28) or manual reference position return (see III–3.1) is made. The G30 command is generally used when the automatic tool changer (ATC) position differs from the reference position. |
| • Return from the reference position (G29) | In general, it is commanded immediately following the G28 command or G30. For incremental programming, the command value specifies the incremental value from the intermediate point. Positioning to the intermediate or reference points are performed at the rapid traverse rate of each axis. When the workpiece coordinate system is changed after the tool reaches the reference position through the intermediate point by the G28 command, the intermediate point also shifts to a new coordinate system. If G29 is then commanded, the tool moves to to the commanded position through the intermediate point which has been shifted to the new coordinate system. The same operations are performed also for G30 commands. |
| Reference position return check (G27) | G27 command positions the tool at rapid traverse rate. If the tool reaches the reference position, the reference position return lamp lights up. However, if the position reached by the tool is not the reference position, an alarm (No. 092) is displayed. |
| Restrictions | |
| Status the machine lock being turned on | The lamp for indicating the completion of return does not go on when the machine lock is turned on, even when the tool has automatically returned to the reference position. In this case, it is not checked whether the tool has returned to the reference position even when a G27 command is specified. |
| • First return to the reference position after the power has been turned on (without an absolute position detector) | When the G28 command is specified when manual return to the reference position has not been performed after the power has been turned on, the movement from the intermediate point is the same as in manual return to the reference position. In this case, the tool moves in the direction for reference position return specified in parameter (No. 003 #0 to #3). Therefore the specified intermediate position must be a position to which reference position return is possible. |

| Reference position return check in an offset mode | In an offset mode, the position to be reached by the tool with the G27 command is the position obtained by adding the offset value. Therefore, if the position with the offset value added is not the reference position, the lamp does not light up, but an alarm is displayed instead. Usually, cancel offsets before G27 is commanded. | |
|--|---|--|
| Lighting the lamp when the programmed position does not coincide with the reference position | When the machine tool system is an inch system with metric input, the reference position return lamp may also light up even if the programmed position is shifted from the reference position by 1μ . This occurs because the least input increment of the machine tool system is smaller than its | |

least command increment.

Reference

Manual reference
 position return

Examples

See III–3.1.

G28G90X1000.0Y500.0;(Programs movement from A to B)T1111;(Changing the tool at the reference position)G29X1300.0Y200.0;(Programs movement from B to C)



Fig. 6 (c) Reference position return and return from the reference position

— 58 —

COORDINATE SYSTEM

By teaching the CNC a desired tool position, the tool can be moved to the position. Such a tool position is represented by coordinates in a coordinate system. Coordinates are specified using program axes. When three program axes, the X-axis, Y-axis, and Z-axis, are used, coordinates are specified as follows:

X_Y_Z

This command is referred to as a dimension word.



Fig. 7 Tool Position Specified by X40.0Y50.0Z25.0

Coordinates are specified in one of following three coordinate systems:

- (1)Machine coordinate system
- (2)Workpiece coordinate system
- (3)Local coordinate system

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

| 7.1 MACHINE COORDINATE SYSTEM | The point that is specific to a machine and serves as the reference of the machine is referred to as the machine zero point. A machine tool builder sets a machine zero point for each machine. The machine zero point matches the first reference position. A coordinate system with a machine zero point set as its origin is referred to as a machine coordinate system. A machine coordinate system is set by performing manual reference position return after power–on (see III–3.1). A machine coordinate system, once set, remains unchanged until the power is turned off. | |
|---|---|--|
| Format | G53 IP_; IP_; Absolute dimension word | |
| Explanations | | |
| Selecting a machine coordinate system (G53) | When a command is specified based on a machine coordinate system, the tool moves by rapid traverse. G53, which is used to select a machine coordinate system, is a one-shot G code; that is, it is valid only in the block in which it is specified. The absolute command G90 is valid, but the incremental command G91 is ignored. When the tool is to be moved to a machine-specific position such as a tool change position, program the movement in a machine coordinate system based on G53. | |

Restrictions

• Cancel of the compensation function

When the G53 command is specified, cancel the cutter compensation, tool length offset, and tool offset.

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

- 60 -

ΖĄ

23.0

0

| 7.2 WORKPIECE COORDINATE SYSTEM | A coordinate system used for machining a workpiece is referred to as a workpiece coordinate system. A workpiece coordinate system is to be se with the NC beforehand (setting a workpiece coordinate system). A machining program sets a workpiece coordinate system (selecting a workpiece coordinate system). A set workpiece coordinate system can be changed by shifting its origin (changing a workpiece coordinate system). | | |
|---|---|--|--|
| 7.2.1 | A workpiece coordinate system can be set using one of three methods: | | |
| Setting a Workpiece Coordinate System | (1) Method using G92 A workpiece coordinate system is set by specifying a value after G92 in the program. | | |
| | (2) Automatic settingIf bit 0 of parameter No. 010 #7 is set beforehand, a workpiece coordinate system is automatically set when manual reference position return is performed (see III–3.1.). | | |
| | (3) Input using the CRT/MDI panel Six workpiece coordinate systems can be set beforehand using the CRT/MDI panel (see III–11.4.3.). | | |
| | To use absolute programming, establish a workpiece coordinate system by applying one of the methods described above. | | |
| Format | | | |
| Setting a workpiece coordinate system by G92 | G92 IP_ | | |
| Explanations | A workpiece coordinate system is set so that a point on the tool, such as the tool tip, is at specified coordinates. If a coordinate system is set using G92 during tool length offset, a coordinate system in which the position before offset matches the position specified in G92 is set. Cutter compensation is cancelled temporarily with G92. | | |
| Examples | | | |
| Example 1 | Example 2 | | |
| Setting the coordinate system by the G92X25.2Z23.0; command | Setting the coordinate system by the G92X600.0Z1200.0; command (The base point on the tool holder is the start point for the program.) | | |

Ρ (The tool tip is the start point for the program.) Ζ Base point 1200.0 25.2 X 600.0 X 0

If an absolute command is issued, the base point moves to the commanded position. In order to move the tool tip to the commanded position, the difference from the tool tip to the base point is compensated by tool length offset.

| 7.2.2 Selecting a Workpiece | The user can choose from set workpiece coordinate systems as described below. (For information about the methods of setting, see 7.2.1.) |
|--------------------------------|--|
| Coordinate System | (1) Selecting a workpiece coordinate system set by G92 or automatic workpiece coordinate system setting Once a workpiece coordinate system is selected, absolute commands work with the workpiece coordinate system |
| | (2) Choosing from six workpiece coordinate systems set using the |
| | CRT/MDI panel |
| | By specifying a G code from G54 to G59, one of the workpiece |
| | coordinate systems 1 to 6 can be selected. |
| | G54 Workpiece coordinate system 1 |
| | G55 Workpiece coordinate system 2 |
| | G56 Workpiece coordinate system 3 |
| | G57 Workpiece coordinate system 4 |
| | G58 Workpiece coordinate system 5 |
| | G59 Workpiece coordinate system 6 |
| | Workpiece coordinate system 1 to 6 are established after reference position return after the power is turned on. When the power is turned on, G54 coordinate system is selected. |

Examples



Fig. 7.2.2 (a)

7.2.3 Changing Workpiece Coordinate System

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

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

- (1) Inputting from the CRT/MDI panel (see III–11.4.2)
- (2) Programming by G10 or G92
- (3) Changing an external workpiece zero point offset value (refer to machine tool builder's manual)
- (4) System variables by Custom Macro B



Fig. 7.2.3 (a) Changing an external workpiece zero point offset value or workpiece zero point offset value

| Format | | | |
|-------------------------------------|--|--|--|
| Changing by G10 | G10 L2 Pp IP_; | | |
| | p=0 : External workpiece zero point offset value p=1 to 6 : Workpiece zero point offset value correspond to workpiece coordinate system 1 to 6 IP : Workpiece zero point offset value of each axis | | |
| Changing by G92 | G92 IP_; | | |
| Explanations | | | |
| Changing by G10 | With the G10 command, each workpiece coordinate system can be | | |

changed separately. When an absolute workpiece zero point offset value is specified, the

specified value becomes a new offset value. When an incremental workpiece zero point offset value is specified, the specified value is added to the current offset value to produce a new offset value.

Changing by G92

By specifying G92IP_;, a workpiece coordinate system (selected with a code from G54 to G59) is shifted to set a new workpiece coordinate system so that the current tool position matches the specified coordinates (IP_).

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

WARNING

When a coordinate system is set with G92 after an external workpiece zero point offset value is set, the coordinate system is not affected by the external workpiece zero point offset value. When G92X100.0Z80.0; is specified, for example, the coordinate system having its current tool reference position at X = 100.0 and Z = 80.0 is set.

Examples





nate system is specified. Then, a G55 workpiece coordinate system where the black circle on the tool (figure at the left) is at (600.0,12000.0) can be set with the following command if the relative relationship between the G54 workpiece coordinate system and G55 workpiece coordinate system is set correctly:G92X600.0Z1200.0;Also, suppose that pallets are loaded at two different positions. If the relative relationship of the coordinate systems of the pallets at the two positions is correctly set by handling the coordinate systems as the G54 workpiece coordinate system and G55 workpiece coordinate system, a coordinate system shift with G92 in one pallet causes the same coordinate system shift in the other pallet. This means that workpieces on two pallets can be machined with the same program just by specifying G54 or G55.

- 64 ----

.....

7.3 LOCAL COORDINATE SYSTEM

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

Format

G52 $\mathbb{P}_{:}$; Setting the local coordinate system

G52 IP 0 ; Canceling of the local coordinate system IP_ : Origin of the local coordinate system

Explanations

By specifying G52 IP_;, a local coordinate system can be set in all the workpiece coordinate systems (G54 to G59). The origin of each local coordinate system is set at the position specified by IP_ in the workpiece coordinate system.

When a local coordinate system is set, the move commands in absolute mode (G90), which is subsequently commanded, are the coordinate values in the local coordinate system. The local coordinate system can be changed by specifying the G52 command with the zero point of a new local coordinate system in the workpiece coordinate system. To cancel the local coordinate system and specify the coordinate value in the workpiece coordinate system, match the zero point of the local coordinate system with that of the workpiece coordinate system.





— 65 —

WARNING

- 1 When an axis returns to the reference point by the manual reference point return function, the zero point of the local coordinate system of the axis matches that of the work coordi–nate system. The same is true when the following command is issued:
 - G52 α0;
 - α :Axis which returns to the reference point
- 2 The local coordinate system setting does not change the workpiece and machine coordi–nate systems.
- 3 The local coordinate system is cancelled when the reset operation is performed.
- 4 When setting a workpiece coordinate system with the G92 command, the local coordinate systems are cancelled.
- 5 G52 cancels the offset temporarily in cutter compensation.
- 6 Command a move command immediately after the G52 block in the absolute mode.

7.4 PLANE SELECTION

Explanations

Select the planes for circular interpolation, cutter compensation, and drilling by G-code.

The following table lists G–codes and the planes selected by them.

| Table 7.4 | Plane | selected | by | G code |
|-----------|-------|----------|----|--------|
|-----------|-------|----------|----|--------|

| G code | Selected plane | Хр | Үр | Zp |
|--------|----------------|---------------|---------------|---------------|
| G17 | Xp Yp plane | X–axis or an | Y–axis or an | Z–axis or an |
| G18 | Zp Xp plane | axis parallel | axis parallel | axis parallel |
| G19 | Yp Zp plane | to it | to it | to it |

Xp, Yp, Zp are determined by the axis address appeared in the block in which G17, G18 or G19 is commanded.

When an axis address is omitted in G17, G18 or G19 block, it is assumed that the addresses of basic three axes are omitted. Parameter No. 279 is used to specify that an optional axis be parallel to the each axis of the X, Y–, and Z–axes as the basic three axes. The plane is unchanged in the block in which G17, G18 or G19 is not commanded.

Examples

Plane selection when the X-axis is parallel with the U-axis.

| | A |
|---------|---|
| G17X_Y_ | XY plane, |
| G17U_Y_ | UY plane |
| G18X_Z_ | ZX plane |
| X_Y_ | Plane is unchanged (ZX plane) |
| G17 | XY plane |
| G18 | ZX plane |
| G17 U_ | UY plane |
| G18Y_; | ZX plane, Y axis moves regardless without any |
| | relation to the plane. |

NOTE

When the system is turned on or placed in the clear state by a reset (bit 6 of parameter No. 045), G17, G18, or G19 is selected according to the setting of parameter No. 212.

8

COORDINATE VALUE AND DIMENSION

This chapter contains the following topics.

8.1 ABSOLUTE AND INCREMENTAL PROGRAMMING (G90, G91)
8.2 INCH/METRIC CONVERSION (G20, G21)
8.3 DECIMAL POINT PROGRAMMING

8.1 ABSOLUTE AND INCREMENTAL PROGRAMMING

(G90, G91)

Format

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

| Absolute command | G90 I₽_; |
|---------------------|-----------|
| Incremental command | G91 I₽_ ; |



Examples

8.2 INCH/METRIC CONVERSION(G20,G21)

Format

G20; Inch input

G21; mm input

This G code must be specified in an independent block before setting the coordinate system at the beginning of the program. After the G code for inch/metric conversion is specified, the unit of input data is switched to the least inch or metric input increment of increment system IS–B or IS–C (2.3). The unit of data input for degrees remains unchanged. The unit systems for the following values are changed after inch/metric conversion:

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

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

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

WARNING

- 1 G20 and G21 must not be switched during a program.
- 2 When switching inch input (G20) to metric input (G21) and vice versa, the tool compensation value must be re-set according to the least input increment.

CAUTION

Reference position return is performed at a low speed for the first G28 command after the inch input is switched to the metric input or vice versa.

NOTE

- 1 When the least input increment and the least command increment systems are different, the maximum error is half of the least command increment. This error is not accumulated.
- 2 The inch and metric input can also be switched using settings.

— 70 —

8.3 DECIMAL POINT PROGRAMMING

Explanations

Numerical values can be entered with a decimal point. A decimal point can be used when entering a distance, time, or speed. Decimal points can be specified with the following addresses: X, Y, Z, U, V, W, A, B, C, I, J, K, Q, R, and F.

There are two types of decimal point notation: calculator-type notation and standard notation.

When calculator-type decimal notation is used, a value without decimal point is considered to be specified in mm, inch, or deg. When standard decimal notation is used, such a value is considered to be specified in least input increments.Select either calculator-type or standard decimal notation by using the parameter No. 051#7. Values can be specified both with and without decimal point in a single program.

Examples

| Program command | Pocket calculator type decimal point programming | Standard type decimal point programming |
|---|--|---|
| X1000 Command value with- out decimal point | 1000mm Unit : mm | 1mm Unit : Least input increment (0.001 mm) |
| X1000.0 Command value with decimal point | 1000mm Unit : mm | 1000mm Unit : mm |

| WARNING In a single b value. The po | lock, specify a G code before entering a osition of decimal point may depend on the |
|---|---|
| | |
| Examples: | |
| G20; | Input in inches |
| X1.0 G04; | X1.0 is considered to be a distance and processed as X10000. This command is equivalent to G04 X10000. The tool dwells for 10 seconds. |
| G04 X1.0; | Equivalent to G04 X1000. The tool dwells for one second. |

| NOT 1 | E Fractions les truncated. Examples: | s than the least input increment are |
|-----------------|---|--|
| | X1.2345; | Truncated to X1.234 when the least input |
| | | increment is 0.001 mm. |
| | | Processed as X1.2345 when the least input |
| | | increment is 0.0001 inch. |
| 2 | When more to occurs. If a v number of di converted to increment. Examples: | han eight digits are specified, an alarm value is entered with a decimal point, the gits is also checked after the value is an integer according to the least input |
| | X1.23456789; | Alarm 003 occurs because more than eight digits are specified. |
| | X123456.7; | If the least input increment is 0.001 mm, the value is converted to integer 123456700. Because the integer has more than eight digits, an alarm occurs. |



SPINDLE SPEED FUNCTION (S FUNCTION)

The spindle speed can be controlled by specifying a value following address S.

This chapter contains the following topics.

9.1 SPECIFYING THE SPINDLE SPEED WITH A CODE9.2 SPECIFYING THE SPINDLE SPEED VALUE DIRECTLY (S5-DIGIT COMMAND)

9.1 SPECIFYING THE SPINDLE SPEED WITH A BINARY CODE

A 2-digit S code can be specified in a block. For a description of the use of S codes, such as their execution sequence in a block in which a spindle speed, move command, and S code are specified, see the manual provided by the machine tool builder.

9.2 SPECIFYING THE SPINDLE SPEED VALUE DIRECTLY (S5-DIGIT COMMAND)

The spindle speed can be specified directly by address S followed by a five-digit value (rpm). The unit for specifying the spindle speed may vary depending on the machine tool builder. Refer to the appropriate manual provided by the machine tool builder for details.

10 TOOL FUNCTION (T FUNCTION)

Tool Selection function is available at tool function.

10.1 TOOL SELECTION FUNCTION

By specifying two or four-digit numerical value following address T, tools can be selected on the machine.

One T code can be commanded in a block. Refer to the machine tool builder's manual for the number of digits commandable with address T and the correspondence between the T codes and machine operations. When a move command and a T code are specified in the same block, the commands are executed in one of the following two ways:

- (i) Simultaneous execution of the move command and T function commands.
- (ii) Executing T function commands upon completion of move command execution.

The selection of either (i) or (ii) depends on the machine tool builder's specifications. Refer to the manual issued by the machine tool builder for details.

AUXILIARY FUNCTION

There are two types of auxiliary functions ; miscellaneous function (M code) for specifying spindle start, spindle stop program end, and so on, and secondary auxiliary function (B code) for specifying index table positioning.

When a move command and miscellaneous function are specified in the same block, the commands are executed in one of the following two ways:

- i) Simultaneous execution of the move command and miscellaneous function commands.
- ii) Executing miscellaneous function commands upon completion of move command execution.

The selection of either sequence depends on the machine tool builder's specification. Refer to the manual issued by the machine tool builder for details.

| 11.1 | |
|--|--|
| AUXILIARY FUNCTION (M FUNCTION) | When a three–digid numeral is specified following address M, code signal and a strobe signal are sent to the machine. The machine uses these signals to turn on or off its functions. Usually, only one M code can be specified in one block. In some cases, however, up to three M codes can be specified for some types of machine tools. Which M code corresponds to which machine function is determined by the machine tool builder. All M codes are processed in the machine except for M98, M99, M codes for calling a subprogram, and M codes for calling a custom macro. Refer to the machine tool builder's instruction manual for details. |
| Explanations | The following M codes have special meanings. |
| • M02,M03 (End of program) | This indicates the end of the main program Automatic operation is stopped and the CNC unit is reset. This differs with the machine tool builder. After a block specifying the end of the program is executed, control returns to the start of the program. Bit 5 of parameter 019 (M02) can be used to disable M02 from returning control to the start of the program. |
| M00 (Program stop) | Automatic operation is stopped after a block containing M00 is executed. When the program is stopped, all existing modal information remains unchanged. The automatic operation can be restarted by actuating the cycle operation. This differs with the machine tool builder. |
| M01 (Optional stop) | Similarly to M00, automatic operation is stopped after a block containing M01 is executed. This code is only effective when the Optional Stop switch on the machine operator's panel has been pressed. |
| M98 (Calling of sub – program) | This code is used to call a subprogram. The code and strobe signals are not sent. See the subprogram section 12.3 for details . |
| M99 (End of subprogram) | This code indicates the end of a subprogram. M99 execution returns control to the main program. See the subprogram section 12.3 for details. |
| | NOTE The block following M00, M01, M02 and M30, is not read into the input buffer register, if present. Similarly, two M codes which do not buffer can be set by parameters (Nos. 111 to 112). Refer to the machine tool builder's instruction manual for these M codes. |

11.2 So far, one block has been able to contain only one M code. However, this function allows up to three M codes to be contained in one block. **MULTIPLE M** Up to three M codes specified in a block are simultaneously output to the COMMANDS machine. This means that compared with the conventional method of a **IN A SINGLE BLOCK** single M command in a single block, a shorter cycle time can be realized in machining. To use this function, set bit 7 of parameter No. 065 to 1. **Explanations** CNC allows up to three M codes to be specified in one block. However, some M codes cannot be specified at the same time due to mechanical operation restrictions. M00, M01, M02, M30, M98 or M99 must not be specified together with another M code. Some M codes other than M00, M01, M02, M30, M98, and M99 cannot be specified together with other M codes; each of those M codes must be specified in a single block. Such M codes include these which direct the CNC to perform internal operations in addition to sending the M codes themselves to the machine. To be specified, such M codes are M codes for calling program numbers 9001 to 9009 and M codes for disabling advance reading (buffering) of subsequent blocks. Meanwhile, multiple of M codes that direct the CNC only to send the M codes themselves (without performing internal operations) can be specified in a single block.

Examples

| One M command | Multiple M commands |
|-------------------|---------------------|
| in a single block | in a single block |
| M40 ; | M40M50M60 ; |
| M50 ; | G28G91X0Y0Z0 ; |
| M60 ; | : |
| G28G91X0Y0Z0 ; | : |
| : | : |

| 11.3 THE SECOND AUXILIARY FUNCTIONS (B CODES) (FOR 0–MD ONLY) | Indexing of the table is performed by address B and a following 3 or 6–digit number. The relationship between B codes and the corresponding indexing differs between machine tool builders. Refer to the manual issued by the machine tool builder for details. |
|--|--|
| Restrictions | When this functions is used, the B address specifying an axis movement disabled. |

PROGRAM CONFIGURATION

General

 Main program and subprogram There are two program types, main program and subprogram. Normally, the CNC operates according to the main program. However, when a command calling a subprogram is encountered in the main program, control is passed to the subprogram. When a command specifying a return to the main program is encountered in a subprogram, control is returned to the main program.



Fig. 12 (a) Main program and Subprogram

The CNC memory can hold up to 200 main programs and subprograms (63 as standard). A main program can be selected from the stored main programs to operate the machine. See Chapter 10 in OPERATION for the methods of registering and selecting programs.

• Program components

A program consists of the following components:

Table 12 (a) Program components

| Components | Descriptions | | |
|-----------------|---|--|--|
| Tape start | Symbol indicating the start of a program file | | |
| Leader section | Used for the title of a program file, etc. | | |
| Program start | Symbol indicating the start of a program | | |
| Program section | Commands for machining | | |
| Comment section | Comments or directions for the operator | | |
| Tape end | Symbol indicating the end of a program file | | |



Fig. 12 (b) Program configuration

 Program section configuration

A program section consists of several blocks. A program section starts with a program number and ends with a program end code.

| Program section | Program section | | |
|-----------------|--------------------------|--|--|
| configuration | | | |
| Program number | O0001 ; | | |
| Block 1 | N1 G91 G00 X120.0 Y80.0; | | |
| Block 2 | N2 G43 Z–32.0 H01 ; | | |
| : : | | | |
| Block n | Nn M20 ; | | |
| Program end | M30 ; | | |

A block contains information necessary for machining, such as a move command or coolant on/off command. Specifying a slash (/) at the start of a block disables the execution of some blocks (see "optional block skip" in 12.2).

— 82 —

12.1 PROGRAM COMPONENTS OTHER THAN PROGRAM SECTIONS

This section describes program components other than program sections. See 12.2 for a program section.



Fig. 12.1(a) Program configuration

Explanations

Tape start

The tape start indicates the start of a file that contains NC programs. The mark is not required when programs are entered using SYSTEM P or ordinary personal computers. The mark is not displayed on the CRT display screen. However, if the file is output, the mark is automatically output at the start of the file.

Table 12.1(a) Code of a tape start

| Name | ISO code | EIA code | Notation in this manual |
|------------|-------------|-------------|-------------------------|
| Tape start | % | ER | % |

• Leader section Data entered before the programs in a file constitutes a leader section. When machining is started, the label skip state is usually set by turning on the power or resetting the system. In the label skip state, all information is ignored until the first end–of–block code is read. When a file is read into the CNC unit from an I/O device, leader sections are skipped by the label skip function.

A leader section generally contains information such as a file header. When a leader section is skipped, even a TV check is not made. So a leader section can contain any codes except the EOB code.

• **Program start** The program start code is to be entered immediately after a leader section, that is, immediately before a program section. This code indicates the start of a program, and is always required to disable the label skip function. With SYSTEM P or ordinary personal computers, this code can be entered by pressing the return key.

| [able 12.1(b) | Code of | a program start |
|---------------|---------|-----------------|
|---------------|---------|-----------------|

| Name | ISO code | EIA code | Notation in this manual |
|---------------|-------------|-------------|-------------------------|
| Program start | LF | CR | ; |

Comment section

NOTE

If one file contains multiple programs, the EOB code for label skip operation must not appear before a second or subsequent program number. However, an program start is required at the start of a program if the preceding program ends with %.

Any information enclosed by the control–out and control–in codes is regarded as a comment. The user can enter a header, comments, directions to the operator, etc. in a comment section using the EOB code or any other code. There is no limit on the length of a comment section.

| Table 12.1 | (c) Codes of a | control-in and a | control-out |
|------------|----------------|------------------|-------------|
|------------|----------------|------------------|-------------|

| Name | ISO code | EIA code | Notation in this manual | Meaning |
|-------------|-------------|-------------|-------------------------|--------------------------|
| Control-out | (| 2–4–5 | (| Start of comment section |
| Control-in |) | 2–4–7 |) | End of comment section |

When a command tape is read into memory for memory operation, comment sections, if any, are not ignored but are also read into memory. Note, however, that codes other than those listed in the code table in Appendix F are ignored, and thus are not read into memory. When data in memory is punched out on paper tape with the punch function, the comment sections are also punched out.

When a program is displayed on the screen, its comment sections are also displayed. However, those codes that were ignored when read into memory are not punched out or displayed.

During memory operation, all comment sections are ignored.

The TV check function can be used for a comment section by setting parameter (No. 0018 #6).

CAUTION

If a long comment section appears in the middle of a program section, a move along an axis may be suspended for a long time because of such a comment section. So a comment section should be placed where movement suspension may occur or no movement is involved.

NOTE

If only a control-in code is read with no matching control-out code, the read control-in code is ignored.

— 84 —

• Tape end

A tape end is to be placed at the end of a file containing NC programs. If programs are entered using the automatic programming system, the mark need not be entered. When a file is output, the mark is automatically output at the end of the file.

If an attempt is made to execute % when M02 or M03 is not placed at the end of the program, the alarm (No. 008) is occurred.

Table 12.1 (d) Code of a tape end

| Name | ISO | EIA | Notation in this |
|----------|------|------|------------------|
| | code | code | manual |
| Tape end | % | ER | % |

12.2 PROGRAM SECTION CONFIGURATION

This section describes elements of a program section. See 12.1 for program components other than program sections.



Fig. 12.2(a) Program configuration (Example of using ISO code)

A program number consisting of address O followed by a four-digit number is assigned to each program at the beginning registered in memory to identify the program.

In ISO code, the colon (:) can be used instead of O.

When no program number is specified at the start of a program, the sequence number (N...) at the start of the program is regarded as its program number. Note, however, that N0 cannot be used for a program number.

If there is no program number or sequence number at the start of a program, a program number must be specified using the CRT/MDI panel when the program is stored in memory(See 8.4 in Part III.).

NOTE

Program numbers 8000 to 9999 may be used by machine tool builders, and the user may not be able to use these numbers.

• Program number

Sequence number and block

A program consists of several commands. One command unit is called a block. One block is separated from another with an EOB of end of block code.

| Table 12.2(a) EOB code | |
|------------------------|--|
|------------------------|--|

| Name | ISO code | EIA code | Notation in this manual |
|--------------------|-------------|-------------|-------------------------|
| End of block (EOB) | LF | CR | ; |

At the head of a block, a sequence number consisting of address N followed by a number not longer than four digits (1 to 9999) can be placed. Sequence numbers can be specified in a random order, and any numbers can be skipped. Sequence numbers may be specified for all blocks or only for desired blocks of the program. In general, however, it is convenient to assign sequence numbers in ascending order in phase with the machining steps (for example, when a new tool is used by tool replacement, and machining proceeds to a new surface with table indexing.)

N300 X200.0 Z300.0; A sequence number is underlined.



NOTE

N0 must not be used for the reason of file compatibility with other CNC systems.

Program number 0 cannot be used. So 0 must not be used for a sequence number regarded as a program number.

TV check (Vertical parity check along tape)

A parity check is made for a block on input tape horizontally. If the number of characters in one block (starting with the code immediately after an EOB and ending with the next EOB) is odd, an alarm (No.002) is output. No TV check is made only for those parts that are skipped by the label skip function. A comment section enclosed in parentheses is also subject to TV check to count the number of characters. The TV check function can be enabled or disabled by setting on the MDI unit (See 11.5.3 in Part III.).

Block configuration (word and address)

A block consists of one or more words. A word consists of an address followed by a number some digits long. (The plus sign (+) or minus sign (-) may be prefixed to a number.)

Word = Address + number (Example : X-1000)

For an address, one of the letters (A to Z) is used ; an address defines the meaning of a number that follows the address. Table 12.2 (b) indicates the usable addresses and their meanings.

The same address may have different meanings, depending on the preparatory function specification.

| Table 12.2(b) | Major | functions | and | addresses |
|---------------|-------|-----------|-----|-----------|
|---------------|-------|-----------|-----|-----------|

| Function | Address | Meaning |
|----------------------------|------------------------------|---|
| Program number | O ⁽¹⁾ | Program number |
| Sequence number | Ν | Sequence number |
| Preparatory function | G | Specifies a motion mode (linear, arc, etc.) |
| Dimension word | X, Y, Z, U, V, W, A, B, C | Coordinate axis move command |
| | I, J, K | Coordinate of the arc center |
| | R | Arc radius |
| Feed function | F | Rate of feed per minute, Rate of feed per revolution |
| Spindle speed function | S | Spindle speed |
| Tool function | Т | Tool number |
| Auxiliary function | М | On/off control on the machine tool |
| | В | Table indexing, etc. |
| Offset number | D, H | Offset number |
| Dwell | P, X | Dwell time |
| Program number designation | Р | Subprogram number |
| Number of repetitions | Ρ | Number of subprogram repetitions |
| Parameter | P, Q | Canned cycle parameter |

NOTE

In ISO code, the colon (:) can also be used as the address of a program number.

| N_ | G_ > | (_ Y_ | F _ | S _ | T _ | M_ | ; |
|--------------------|-------------------------|-------------------|-------------------|------------------------------|------------------|-----------------------|-------|
| Sequence number | Preparatory function | Dimension word | Feed- function | Spindle speed function | Tool function | Miscellar function | neous |

Fig. 12.2(c) 1 block (example)

— 88 —

Major addresses and ranges of command values

Major addresses and the ranges of values specified for the addresses are shown below. Note that these figures represent limits on the CNC side, which are totally different from limits on the machine tool side. For example, the CNC allows a tool to traverse up to about 100 m (in millimeter input) along the X axis.

However, an actual stroke along the X axis may be limited to 2 m for a specific machine tool.

Similarly, the CNC may be able to control a cutting federate of up to 100 m/min, but the machine tool may not allow more than 3 m/min. When developing a program, the user should carefully read the manuals of the machine tool as well as this manual to be familiar with the restrictions on programming.

| Function | | Address | Input in mm | Input in inch |
|---------------------------------|--------------------------|-------------------------|--------------------|-------------------------|
| Program number | | O ⁽¹⁾ | 1–9999 | 1–9999 |
| Sequence | e number | N | 1–9999 | 1–9999 |
| Preparato | ory function | G | 0–255 | 0–255 |
| Dimen- sion | Increment system IS–B | X, Y, Z, U, V, W, | \pm 999999.999mm | \pm 99999.9999inch |
| wora | Increment system IS–C | а, в, с, I, J, K, R, | \pm 99999.9999mm | \pm 999.999999inch |
| Feed per | Increment system IS–B | F | 1–100000mm/min | 0.01–4000 inch/min |
| minute | Increment system IS–C | | 1–24000mm/min | 0.01–480.00 inch/min |
| Spindle speed function | | S | 0–20000 | 0–20000 |
| Tool funct | ion | Т | 0–9999 | 0–9999 |
| Auxiliary | function | М | 0–999 | 0–9999 |
| | | В | 0–999999 | 0–999999 |
| Offset nui | mber | H, D | 0–200 | 0–200 |
| Dwell Increment system IS-B | | Х, Р | 0–99999.999s | 0–99999.999s |
| | Increment system IS–C | | 0–9999.9999s | 0–9999.9999s |
| Designation of a program number | | Ρ | 1–9999 | 1–9999 |
| Number of repetitions | | Р | 1–999 | 1–999 |

Table 12.2(c) Major addresses and ranges of command values

Optional block skip

When a slash is specified at the head of a block, and optional block skip switch on the machine operator panel is set to on, the information contained in the block is ignored in tape operation or memory operation. When optional block skip switch is set to off, the information contained in the block for which /is specified is valid. This means that the operator can determine whether to skip the block containing.

This function is ignored when programs are loaded into memory. Blocks containing / are also stored in memory, regardless of how the optional block skip switch is set.

Programs held in memory can be output, regardless of how the optional block skip switches are set.

Optional block skip is effective even during sequence number search operation.

WARNING

1 Position of a slash

A slash (/) must be specified at the head of a block. If a slash is placed elsewhere, the information from the slash to immediately before the EOB code is ignored.

2 Disabling an optional block skip switch Optional block skip operation is processed when blocks are read from memory or tape into a buffer. Even if a switch is set to on after blocks are read into a buffer, the blocks already read are not ignored.

NOTE

TV and TH check

When an optional block skip switch is on. TH and TV checks are made for the skipped portions in the same way as when the optional block skip switch is off.

Program end

The end of a program is indicated by commanding one of the following codes at the end of the program:

Table 12.2(d) Code of a program end

| Code | Meaning usage |
|------|-------------------|
| M02 | For main program |
| M30 | r of main program |
| M99 | For subprogram |

If one of the program end codes is executed in program execution, the CNC terminates the execution of the program, and the reset state is set. When the subprogram end code is executed, control returns to the program that called the subprogram.

WARNING

A block containing an optional block skip code such as /M02 ; , /M30 ; , or /M99 ; is not regarded as the end of a program, if the optional block skip switch on the machine operator's panel is set to on.

(See "· Optional block skip" as previous item for optional block skip.)

12.3 SUBPROGRAM

If a program contains a fixed sequence or frequently repeated pattern, such a sequence or pattern can be stored as a subprogram in memory to simplify the program. A subprogram can be called from the main program. A called subprogram can also call another subprogram.

Format

 Subprogram configuration

| One subprogram | | | | |
|---|---|--|--|--|
| O | Subprogram number (or the colon (:) optionally in the case of ISO) | | | |
| M99; | Program end | | | |
| M99 need not constitute a separate block as indicated below. Example) X100.0 Y100.0 M99 ; | | | | |

• Subprogram call



Explanations

When the main program calls a subprogram, it is regarded as a one-level subprogram call. Thus, subprogram calls can be nested up to two levels as shown below.



A single call command can repeatedly call a subprogram up to 999 times. For compatibility with automatic programming systems, in the first block, Nxxxx can be used instead of a subprogram number that follows O (or :). A sequence number after N is registered as a subprogram number.

See Chapter 10 in Part III for the method of registering a subprogram.
NOTE

- 1 The M98 and M99 signals are not output to the machine tool.
- 2 If the subprogram number specified by address P cannot be found, an alarm (No. 078) is output.

Examples



Special Usage

 Specifying the sequence number for the return destination in the main program If P is used to specify a sequence number when a subprogram is terminated, control does not return to the block after the calling block, but returns to the block with the sequence number specified by P. Note, however, that P is ignored if the main program is operating in a mode other than memory operation mode.

This method consumes a longer time than the normal return method to return to the main program.

| Main program | S | ubprogram |
|-------------------|-----|------------------|
| N0010 ; | 10 | 1010 ; |
| N0020 ; | N | 1020 ; |
| N0030 M98 P1010 ; | / N | 1030 ; |
| N0040 ; | Ν | 1040 ; |
| N0050 ; | Ν | 1050 ; |
| N0060 ; | N | 1060 M99 P0060 ; |
| 4 | | |

• Using M99 in the main program

If M99 is executed in a main program, control returns to the start of the main program. For example, M99 can be executed by placing /M99; at an appropriate location of the main program and setting the optional block skip function to off when executing the main program. When M99 is executed, control returns to the start of the main program, then execution is repeated starting at the head of the main program.

Execution is repeated while the optional block skip function is set to off. If the optional block skip function is set to on, the /M99; block is skipped; control is passed to the next block for continued execution.

If/M99P<u>n</u>; is specified, control returns not to the start of the main program, but to sequence number n. In this case, a longer time is required to return to sequence number n.



• Using a subprogram only

A subprogram can be executed just like a main program by searching for the start of the subprogram with the MDI.

(See Sec. 9.3 in Part III for information about search operation.)

In this case, if a block containing M99 is executed, control returns to the start of the subprogram for repeated execution. If a block containing M99Pn is executed, control returns to the block with sequence number n in the subprogram for repeated execution. To terminate this program, a block containing /M02; or /M30; must be placed at an appropriate location, and the optional block skip must be set to off; this switch is to be set to on first.





FUNCTIONS TO SIMPLIFY PROGRAMMING

This chapter explains the following items:

- 13.1 CANNED CYCLE
- **13.2 RIGID TAPPING**
- 13.3 CANNED GRINDING CYCLE
- 13.4 AUTOMATIC GRINDING WHEEL DIAMETER COMPENSATION AFTER DRESSING
- 13.5 IN-FEED GRINDING ALONG THE Y AND Z AXES AT THE END OF TABLE SWING
- **13.6 EXTERNAL MOTION FUNCTION**

13.1 CANNED CYCLE

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, normally more than one block is required. In addition, the use of canned cycles can shorten the program to save memory.

Table 13.1 (a) lists canned cycles.

| G code | Drilling(-Z direction) | Operation at the bottom of a hole | Retraction(+Z direction) | Application |
|--------|------------------------|-----------------------------------|--------------------------|--------------------------------------|
| G73 | Intermittent feed | _ | Rapid traverse | High-speed peck drilling cycle |
| G74 | Feed | Dwell→Spindle CW | Feed | Left-hand tapping cycle |
| G76 | Feed | Spindle Orientation | Rapid traverse | Fine boring cycle |
| G80 | _ | _ | _ | Cancel |
| G81 | Feed | _ | Rapid traverse | Drilling cycle, spot drilling cycle |
| G82 | Feed | Dwell | Rapid traverse | Drilling cycle, counter boring cycle |
| G83 | Intermittent feed | - | Rapid traverse | Peck drilling cycle |
| G84 | Feed | Dwell→Spindle CCW | Feed | Tapping cycle |
| G85 | Feed | _ | Feed | Boring cycle |
| G86 | Feed | Spindle stop | Rapid traverse | Boring cycle |
| G87 | Feed | Spindle CW | Rapid traverse | Back boring cycle |
| G88 | Feed | Dwell→spindle stop | Manual | Boring cycle |
| G89 | Feed | Dwell | Feed | Boring cycle |

Explanations

A canned cycle consists of a sequence of six operations (Fig. 13.1 (a))

| Operation 1 | Positioning of axes X and Y |
|-------------|--|
| | (including also another axis) |
| Operation 2 | Rapid traverse up to point R level |
| Operation 3 | Hole machining |
| Operation 4 | Operation at the bottom of a hole |
| Operation 5 | Retraction to point R level |
| Operation 6 | Rapid traverse up to the initial point |
| | |



Fig. 13.1 (a) Canned cycle operation sequence

• Positioning plane

• Drilling axis

The positioning plane is determined by plane selection code G17, G18, or G19.

The positioning axis is an axis other than the drilling axis.

Although canned cycles include tapping and boring cycles as well as drilling cycles, in this chapter, only the term drilling will be used to refer to operations implemented with canned cycles.

The drilling axis is a basic axis (X, Y, or Z) not used to define the positioning plane, or any axis parallel to that basic axis.

| G code | Positioning plane | Drilling axis |
|--------|-------------------|---------------|
| G17 | Xn-Yn plane | 7 avis |
| | | |
| G18 | Zp–Xp plane | Y axis |
| G19 | Yp–Zp plane | X axis |

Table 13.1(b) Positioning plane and drilling axis

G17 to G19 may be specified in a block in which any of G73 to G89 is not specified.

CAUTION

Switch the drilling axis after canceling a canned cycle.

NOTE

A parameter (No. 057 #6) can be set to the Z axis always used as the drilling axis. When FXY=0, the Z axis is always the drilling axis.

• Travel distance along the drilling axis G90/G91

The travel distance along the drilling axis varies for G90 and G91 as follows:



Drilling mode

G73, G74, G76, and G81 to G89 are modal G codes and remain in effect until canceled. When in effect, the current state is the drilling mode. Once drilling data is specified in the drilling mode, the data is retained until modified or canceled.

Specify all necessary drilling data at the beginning of canned cycles; when canned cycles are being performed, specify data modifications only.

 Return point level G98/G99 When the tool reaches the bottom of a hole, the tool may be returned to point R or to the initial level. These operations are specified with G98 and G99. The following illustrates how the tool moves when G98 or G99 is specified. Generally, G99 is used for the first drilling operation and G98 is used for the last drilling operation.

The initial level does not change even when drilling is performed in the G99 mode.



| • Repeat | To repeat drilling for equally-spaced holes, specify the number of repeats in KK is effective only within the block where it is specified.Specify the first hole position in incremental mode (G91).If it is specified in absolute mode (G90), drilling is repeated at the same position. | | |
|--|--|--|--|
| | Number of repeats K The maximum command value = 9999 | | |
| • Concol | If K0 is specified, drilling data is stored, but drilling is not performed. | | |
| | To cancel a canned cycle, use G80 or a group 01 G code. | | |
| | <u>Group 01 G codes</u> | | |
| | G00 : Positioning (rapid traverse) G01 : Linear interpolation | | |
| | G02 : Circular interpolation (CW) | | |
| | G03 : Circular interpolation (CCW) | | |
| Symbols in figures | Subsequent sections explain the individual canned cycles. Figures in these explanations use the following symbols: | | |
| | → Positioning (rapid traverse G00) | | |
| | Cutting feed (linear interpolation G01) | | |

Manual feed

Dwell

Shift (rapid traverse G00)

Spindle orientation (The spindle stops at a fixed rotation position)

7

OSS \Rightarrow

Ρ

13.1.1 High–speed Peck Drilling Cycle (G73)

This cycle performs high–speed peck drilling. It performs intermittent cutting feed to the bottom of a hole while removing chips from the hole.

Format



| Explanations | The high–speed peck drillin the Z–axis. When this cycle easily, and a smaller value ca to be performed efficiently. The tool is retracted in rapid Before specifying G73, rotat (M code). When the G73 code and an M code is executed at the ti system then proceeds to the When K is used to specify th for the first hole only; for th is not executed. When a tool length offset (G cycle, the offset is applied at | g cycle performs intermittent feeding along is used, chips can be removed from the hole an be set for retraction. This allows, drilling Set the clearance, d, in parameter 531. traverse. e the spindle using a miscellaneous function M code are specified in the same block, the ime of the first positioning operation. The next drilling operation. e number of repeats, the M code is executed e second and subsequent holes, the M code 643, G44, or G49) is specified in the canned t the time of positioning to point R. |
|------------------------------------|--|--|
| Restrictions | | |
| Axis switching | Before the drilling axis can be changed, the canned cycle must be canceled. | |
| • Drilling | In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed. | |
| • Q/R | Specify Q and R in blocks that perform drilling. If they are specified in a block that does not perform drilling, they cannot be stored as modal data. | |
| Cancel | Do not specify a group 01 G c If they are specified together | ode (G00 to G03) and G73 in the same block. r, G73 is canceled. |
| Examples | M3 S2000 ; G90 G99 G73 X300. Y-250. Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; G80 G28 G91 X0 Y0 Z0 ; | Cause the spindle to start rotating. Z–150. R–100. Q15. F120. ; Position, drill hole 1, then return to point R. Position, drill hole 2, then return to point R. Position, drill hole 3, then return to point R. Position, drill hole 4, then return to point R. Position, drill hole 5, then return to point R. Position, drill hole 6, then return to the ini- tial level. Return to the reference position return |
| | M5; | Cause the spindle to stop rotating. |

clockwise.

13.1.2 Left–handed Tapping Cycle (G74)

Format



This cycle performs left-handed tapping. In the left-handed tapping cycle, when the bottom of the hole has been reached, the spindle rotates

Explanations

Tapping is performed by turning the spindle counterclockwise. When the bottom of the hole has been reached, the spindle is rotated clockwise for retraction. This creates a reverse thread.

Feedrate overrides are ignored during left–handed tapping. A feed hold does not stop the machine until the return operation is completed.

Before specifying G74, use a miscellaneous function (M code) to rotate the spindle counterclockwise.

When the G74 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. The system then proceeds to the next drilling operation.

When K is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

| Before the drilling axis canceled. | can be changed, the canned cycle must be |
|---|---|
| In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed. | |
| Specify R in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data. | |
| Do not specify a group 01 If they are specified toget | G code (G00 to G03) and G74 in the same block. her, G74 is canceled. |
| M4 S100; G90 G99 G74 X300. Y-2 Y-550.; Y-750.; X1000.; Y-550.; G98 Y-750.; G80 G28 G91 X0 Y0 Z0 M5; | Cause the spindle to start rotating. 50. Z–150. R–120. F120. ; Position, tapping hole 1, then return to point R. Position, tapping hole 2, then return to point R. Position, tapping hole 3, then return to point R. Position, tapping hole 4, then return to point R. Position, tapping hole 5, then return to point R. Position, tapping hole 6, then return to the initial level. ; Return to the reference position return Cause the spindle to stop rotating. |
| | Before the drilling axis canceled. In a block that does not c not performed. Specify R in blocks that p does not perform drilling. Do not specify a group 01 If they are specified toget M4 S100 ; G90 G99 G74 X300. Y–2 Y–550. ; Y–750. ; X1000. ; Y–550. ; G88 G28 G91 X0 Y0 Z0 M5 ; |

13.1.3 Fine Boring Cycle (G76)

The fine boring cycle bores a hole precisely. When the bottom of the hole has been reached, the spindle stops, and the tool is moved away from the machined surface of the workpiece and retracted.

Format



WARNING

Q (shift at the bottom of a hole) is a modal value retained within canned cycles. It must be specified carefully because it is also used as the depth of cut for G73 and G83.

| Explanations | When the bottom of the hole has been reached, the spindle is stopped at the fixed rotation position, and the tool is moved in the direction opposite to the tool tip and retracted. This ensures that the machined surface is not damaged and enables precise and efficient boring to be performed. Before specifying G76, use a miscellaneous function (M code) to rotate the spindle. When the G76 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. The system then proceeds to the next operation. When K is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed. When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R. | | |
|------------------------------------|---|---|--|
| • Restrictions | | | |
| Axis switching | Before the drilling axis can be changed, the canned cycle must be canceled. | | |
| • Boring | In a block that does not contain X, Y, Z, R, or any additional axes, boring is not performed. | | |
| ● Q/R | Be sure to specify a positive value in Q. If Q is specified with a negative value, the sign is ignored. Set the direction of shift in bits 4 and 5 of parameter 002. Specify Q and R in a block that performs boring. If they are specified in a block that does not perform boring, they are not stored as modal data. | | |
| ● Cancel | Do not specify a group 01 G code (G00 to G03) and G76 in the same block. If they are specified together, G76 is canceled. | | |
| Examples | M3 S500 ; G90 G99 G76 X300. Y-250. Z-150. R-120. Q5. P1000 F120. ; Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; | Cause the spindle to start rotating. Position, bore hole 1, then return to point R. Orient at the bottom of the hole, then shift by 5 mm. Stop at the bottom of the hole for 1 s. Position, drill hole 2, then return to point R. Position, drill hole 3, then return to point R. Position, drill hole 4, then return to point R. Position, drill hole 5, then return to point R. Position, drill hole 6, then return to the initial level. | |
| | M5; | Cause the spindle to stop rotating. | |

13.1.4 Drilling Cycle, Spot Drilling (G81)

Format

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.



Explanations

After positioning along the X– and Y–axes, rapid traverse is performed to point R.

Drilling is performed from point R to point Z.

The tool is then retracted in rapid traverse.

Before specifying G81, use a miscellaneous function (M code) to rotate the spindle.

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

When K is used to specify the number of repeats, the M code is performed for the first hole only; for the second and subsequent holes, the M code is not executed.

| | Restrictions | |
|---|--------------|--|
| • | | |

| Axis switching | Before the drilling axis canceled. | can be changed, the canned cycle must be |
|------------------------------------|---|---|
| • Drilling | In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed. | |
| • R | Specify R in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data. | |
| Cancel | Do not specify a group 01 G code (G00 to G03) and G81 in the same block. If they are specified together, G81 is canceled. | |
| Examples | M3 S2000 ; G90 G99 G81 X300. Y-25 Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; G80 G28 G91 X0 Y0 Z0 ; M5 ; | Cause the spindle to start rotating. 50. Z–150. R–100. F120. ; Position, drill hole 1, then return to point R. Position, drill hole 2, then return to point R. Position, drill hole 3, then return to point R. Position, drill hole 4, then return to point R. Position, drill hole 5, then return to point R. Position, drill hole 6, then return to the initial level. Return to the reference position return Cause the spindle to stop rotating. |

13.1.5 Drilling Cycle Counter Boring Cycle (G82)

Format

This cycle is used for normal drilling. Cutting feed is performed to the bottom of the hole. At the bottom, a dwell is performed, then the tool is retracted in rapid traverse. This cycle is used to drill holes more accurately with respect to depth.

G82 X_Y_Z_R_P_F_K_; X_Y_ : Hole position data : The distance from point R to the bottom of the hole Ζ_ R_{-} : The distance from the initial level to point R level Ρ_ : Dwell time at the bottom of a hole **F**_ : Cutting feed rate : Number of repeats K G82 (G99) G82 (G98) Initial level Point R level Point R Point R Point Z Point Z PC

Explanations

After positioning along the X– and Y–axes, rapid traverse is performed to point R.

Drilling is then performed from point R to point Z.

When the bottom of the hole has been reached, a dwell is performed. The tool is then retracted in rapid traverse.

Before specifying G82, use a miscellaneous function (M code) to rotate the spindle.

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

When K is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

| • | Restrictions | |
|---|--------------|--|

Examples

Axis switching Before the drilling axis can be changed, the canned cycle must be canceled.
 Drilling In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.
 R Specify R in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data.
 Cancel Do not specify a group 01 G code (G00 to G03) and G82 in the same block. If they are specified together, G82 is canceled.

| M3 S2000 ; | Cause the spindle to start rotating. | | |
|---------------------|---|--|--|
| G90 G99 G82 X300. Y | ′–250. Z–150. R–100. P1000 F120. ; | | |
| | Position, drill hole 2, and dwell for 1 s at the | | |
| | bottom of the hole, then return to point R. | | |
| Y–550. ; | Position, drill hole 2, then return to point R. | | |
| Y–750. ; | Position, drill hole 3, then return to point R. | | |
| X1000. ; | Position, drill hole 4, then return to point R. | | |
| Y–550. ; | Position, drill hole 5, then return to point R. | | |
| G98 Y–750. ; | Position, drill hole 6, then return to the initial | | |
| | level. | | |
| G80 G28 G91 X0 Y0 Z | 20 ; Return to the reference position return | | |
| M5; | Cause the spindle to stop rotating. | | |

13.1.6 Peck Drilling Cycle (G83)

This cycle performs peck drilling. It performs intermittent cutting feed to the bottom of a hole while removing shavings from the hole.

Format



Explanations

Q represents the depth of cut for each cutting feed. It must always be specified as an incremental value.

In the second and subsequent cutting feeds, rapid traverse is performed up to a point just before where the last drilling ended, and cutting feed is performed again. Specify the amount of retraction in parameter No. 532. Be sure to specify a positive value in Q. Negative values are ignored.

Before specifying G83, use a miscellaneous function (M code) to rotate the spindle.

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

When K is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

| Axis switching | Before the drilling axis c canceled. | Before the drilling axis can be changed, the canned cycle must be canceled. | |
|------------------------------------|---|--|--|
| • Drilling | In a block that does not con not performed. | ntain X, Y, Z, R, or any other axes, drilling is | |
| • Q/R | Specify Q and R in blocks a block that does not perform | Specify Q and R in blocks that perform drilling. If they are specified in a block that does not perform drilling, they cannot be stored as modal data. | |
| Cancel | Do not specify a group 01 G If they are specified togeth | code (G00 to G03) and G83 in the same block. er, G83 is canceled. | |
| Examples | M3 S2000 ; G90 G99 G83 X300. Y-250 Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; G80 G28 G91 X0 Y0 Z0 ; | Cause the spindle to start rotating. 0. Z–150. R–100. Q15. F120. ; Position, drill hole 1, then return to point R. Position, drill hole 2, then return to point R. Position, drill hole 3, then return to point R. Position, drill hole 4, then return to point R. Position, drill hole 5, then return to point R. Position, drill hole 6, then return to the initial level. Return to the reference position return | |
| | M5; | Cause the spindle to stop rotating. | |

This cycle performs tapping.

spindle is rotated in the reverse direction.

13.1.7 Tapping Cycle (G84)

Format



In this tapping cycle, when the bottom of the hole has been reached, the

Explanations

Tapping is performed by rotating the spindle clockwise. When the bottom of the hole has been reached, the spindle is rotated in the reverse direction for retraction. This operation creates threads.

Feedrate overrides are ignored during tapping. A feed hold does not stop the machine until the return operation is completed.

Before specifying G84, use a miscellaneous function (M code) to rotate the spindle.

When the G84 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. The system then proceeds to the next drilling operation.

When the K is used to specify number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

| Restrictions | |
|----------------------------------|--|
|----------------------------------|--|

| Before the drilling axis can be changed, the canned cycle must be canceled. | |
|---|--|
| In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed. | |
| Specify R in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data. | |
| Do not specify a group 01 G code (G00 to G03) and G84 in the same block. If they are specified together, G84 is canceled. | |
| M3 S100 ; G90 G99 G84 X300. Y-25 Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; G80 G28 G91 X0 Y0 Z0 ; M5 ; | Cause the spindle to start rotating. 50. Z–150. R–120. P300 F120. ; Position, drill hole 1, then return to point R. Position, drill hole 2, then return to point R. Position, drill hole 3, then return to point R. Position, drill hole 4, then return to point R. Position, drill hole 5, then return to point R. Position, drill hole 6, then return to the initial level. Return to the reference position return Cause the spindle to stop rotating. |
| | Before the drilling axis canceled. In a block that does not cont not performed. Specify R in blocks that per does not perform drilling, Do not specify a group 01 C If they are specified togeth M3 S100 ; G90 G99 G84 X300. Y-28 Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; G80 G28 G91 X0 Y0 Z0 ; M5 ; |

13.1.8 Boring Cycle (G85)

Format

This cycle is used to bore a hole.



Explanations

After positioning along the X– and Y– axes, rapid traverse is performed to point R.

Drilling is performed from point R to point Z.

When point Z has been reached, cutting feed is performed to return to point R.

Before specifying G85, use a miscellaneous function (M code) to rotate the spindle.

When the G85 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. The system then proceeds to the next drilling operation.

When K is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

| • | Restrictions |
|---|--------------|
| | |

| • Restrictions | | |
|------------------------------------|---|--|
| Axis switching | Before the drilling axis can be changed, the canned cycle must be canceled. | |
| • Drilling | In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed. | |
| • R | Specify R in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data. | |
| Cancel | Do not specify a group 01 G code (G00 to G03) and G85 in the same block. If they are specified together, G85 is canceled. | |
| Examples | M3 S100 ; G90 G99 G85 X300. Y-25 Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; G80 G28 G91 X0 Y0 Z0 ; | Cause the spindle to start rotating. 50. Z–150. R–120. F120. ; Position, drill hole 1, then return to point R. Position, drill hole 2, then return to point R. Position, drill hole 3, then return to point R. Position, drill hole 4, then return to point R. Position, drill hole 5, then return to point R. Position, drill hole 6, then return to the initial level. Return to the reference position return |
| | M5 ; | Cause the spindle to stop rotating. |

13.1.9 Boring Cycle (G86)

Format

This cycle is used to bore a hole.



Explanations

After positioning along the X– and Y–axes, rapid traverse is performed to point R.

Drilling is performed from point R to point Z.

When the spindle is stopped at the bottom of the hole, the tool is retracted in rapid traverse.

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

When the G86 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation.

The system then proceeds to the next drilling operation.

When K is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

| • | Restrictions | |
|---|--------------|--|

| Before the drilling axis can be changed, the canned cycle must be canceled. | |
|---|---|
| In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed. | |
| Specify R in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data. | |
| Do not specify a group 01 G code (G00 to G03) and G86 in the same block. If they are specified together, G86 is canceled. | |
| M3 S2000 ; G90 G99 G86 X300. Y-29 Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; G80 G28 G91 X0 Y0 Z0 M5 ; | Cause the spindle to start rotating. 50. Z–150. R–100. F120. ; Position, drill hole 1, then return to point R. Position, drill hole 2, then return to point R. Position, drill hole 3, then return to point R. Position, drill hole 4, then return to point R. Position, drill hole 5, then return to point R. Position, drill hole 6, then return to the initial level. ; Return to the reference position return Cause the spindle to stop rotating. |
| | Before the drilling axis canceled. In a block that does not cont not performed. Specify R in blocks that p does not perform drilling, Do not specify a group 01 C If they are specified toget M3 S2000 ; G90 G99 G86 X300. Y–29 Y–550. ; Y–750. ; X1000. ; Y–550. ; G88 G28 G91 X0 Y0 Z0 M5 ; |

This cycle performs accurate boring.

13.1.10 Boring Cycle Back Boring Cycle (G87)

Format



WARNING

Q (shift at the bottom of a hole) is a modal value retained in canned cycles. It must be specified carefully because it is also used as the depth of cut for G73 and G83.

— 119 —

| Explanations | After positioning along the X fixed rotation position. The tool tip, positioning (rapid tra (point R). The tool is then shifted in the rotated clockwise. Boring is Z-axis until point Z is reached At point Z, the spindle is stop tool is shifted in the direction returned to the initial level. The tool tip and the spindle is rote operation. Before specifying G87, use at the spindle. When the G87 command and the M code is executed at the The system then proceeds to to specify the number of reper- only; for the second and subst When a tool length offset (G cycle, the offset is applied at | K- and Y-axes, the spindle is stopped at the cool is moved in the direction opposite to the verse) is performed to the bottom of the hole e direction of the tool tip and the spindle is performed in the positive direction along the ed. oped at the fixed rotation position again, the on opposite to the tool tip, then the tool is The tool is then shifted in the direction of the ated clockwise to proceed to the next block a miscellaneous function (M code) to rotate d an M code are specified in the same block, e time of the first positioning operation. the next drilling operation. When K is used eats, the M code is executed for the first hole sequent holes, the M code is not executed. 43, G44, or G49) is specified in the canned the time of positioning to point R. |
|------------------|--|--|
| • Restrictions | | |
| • Axis switching | Before the drilling axis can canceled. | n be changed, the canned cycle must be |
| • Boring | In a block that does not conta is not performed. | in X, Y, Z, R, or any additional axes, boring |
| ● Q/R | Be sure to specify a positive value in Q. If Q is specified with a negative value, the sign is ignored. Set the direction of shift in bits 4 and 5 of parameter 002. Specify Q and R in a block that performs boring. If they are specified in a block that does not perform boring, they are not stored as modal data. | |
| • Cancel | Do not specify a group 01 G c If they are specified together | ode (G00 to G03) and G76 in the same block. c, G76 is canceled. |
| Examples | M3 S500 ; G90 G87 X300. Y-250. Z120. R-150. Q5. P1000 F120. ; Y-550. ; Y-750. ; X1000. ; Y-550. ; Y-750. ; G80 G28 G91 X0 Y0 Z0 ; M5 ; | Cause the spindle to start rotating. Position, bore hole 1. Orient at the initial level, then shift by 5 mm. Stop at point Z for 1 s. Position, drill hole 2. Position, drill hole 3. Position, drill hole 4. Position, drill hole 5. Position, drill hole 6 Return to the reference position return Cause the spindle to stop rotating. |

13.1.11 Boring Cycle (G88)

Format

This cycle is used to bore a hole.



Explanations

After positioning along the X– and Y–axes, rapid traverse is performed to point R. Boring is performed from point R to point Z. When boring is completed, a dwell is performed, then the spindle is stopped. The tool is manually retracted from the bottom of the hole (point Z) to point R. At point R, the spindle is rotated clockwise, and rapid traverse is performed to the initial level.

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

When the G88 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. The system then proceeds to the next drilling operation.

When K is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

Position, drill hole 5, then return to point R.

Cause the spindle to stop rotating.

Position, drill hole 6, then return to the initial

| Restrictions | |
|--------------|--|
|--------------|--|

| Axis switching | Before the drilling canceled. | g axis can be changed, the canned cycle must be | |
|------------------------------------|---|---|--|
| • Drilling | In a block that does not performed. | s not contain X, Y, Z, R, or any other axes, drilling is | |
| • R | Specify R in blocks does not perform d | Specify R in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data. | |
| • Cancel | Do not specify a gro If they are specified | Do not specify a group 01 G code (G00 to G03) and G88 in the same block. If they are specified together, G88 is canceled. | |
| Examples | M3 S2000 ; G90 G99 G88 X30 | Cause the spindle to start rotating. D. Y-250. Z-150. R-100. P1000 F120. : | |
| | | Position, drill hole 1, return to point R | |
| | | then stop at the bottom of the hole for 1 s. | |
| | Y–550. ; | Position, drill hole 2, then return to point R. | |
| | Y–750. ; | Position, drill hole 3, then return to point R. | |
| | X1000. ; | Position, drill hole 4, then return to point R. | |

level. G80 G28 G91 X0 Y0 Z0 ; Return to the reference position return

Y-550.;

M5;

G98 Y-750.;

13.1.12 Boring Cycle (G89)

Format

This cycle is used to bore a hole.



Explanations

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

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

When the G89 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. The system then proceeds to the next drilling operation.

When K is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

• Restrictions

| Axis switching | Before the drilling axis can be changed, the canned cycle must be canceled. | | | |
|------------------------------------|---|---|--|--|
| • Drilling | In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed. | | | |
| • R | Specify R in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data. | | | |
| • Cancel | Do not specify a group 01 G code (G00 to G03) and G73 in the same block. If they are specified together, G73 is canceled. | | | |
| Examples | M3 S100 ; G90 G99 G89 X300. Y-25 Y-550. ; Y-750. ; X1000. ; Y-550. ; G98 Y-750. ; G80 G28 G91 X0 Y0 Z0 ; M5 ; | Cause the spindle to start rotating. 50. Z–150. R–120. P1000 F120. ; Position, drill hole 1, return to point R then stop at the bottom of the hole for 1 s. Position, drill hole 2, then return to point R. Position, drill hole 3, then return to point R. Position, drill hole 4, then return to point R. Position, drill hole 5, then return to point R. Position, drill hole 6, then return to the initial level. ; Return to the reference position return Cause the spindle to stop rotating. | | |

13.1.13 Canned Cycle Cancel (G80)

G80 cancels canned cycles.

Format

| | G80 ; | G80 ; | | | |
|--------------|---|---|--|--|--|
| Explanations | All canned cycles are point Z are cleared. T Other drilling data is | All canned cycles are canceled to perform normal operation. Point R and point Z are cleared. This means that $R = 0$ and $Z = 0$ in incremental mode. Other drilling data is also canceled (cleared). | | | |
| Examples | M3 S100 ; G90 G99 G88 X300. | Cause the spindle to start rotating. . Y–250. Z–150. R–120. F120. ; | | | |
| | V 550 · | Position, drill hole 1, then return to point R. | | | |
| | Y-750. : | Position, drill hole 3 then return to point R. | | | |
| | X1000. ; | Position, drill hole 4, then return to point R. | | | |
| | Y–550. ; | Position, drill hole 5, then return to point R. | | | |
| | G98 Y–750. ; | Position, drill hole 6, then return to the initial level. | | | |
| | G80 G28 G91 X0 Y0 | D Z0 ; Return to the reference position return, canned cycle cancel | | | |
| | M5; | Cause the spindle to stop rotating. | | | |
| | | | | | |

Program example using tool length offset and canned cycles



| Offset val | ue +200.0 is set in offset No.11, | +190.0 is set in offset No.15, and +150.0 is set in offset No.31 |
|----------------|-----------------------------------|--|
| <u>Progran</u> | <u>n example</u> | |
| ; | | |
| N001 | G92X0Y0Z500.0; | Coordinate setting at reference position |
| N002 | G90 G00 Z250.0 T11 M6; | Tool change |
| N003 | G43 Z0 H11; | Initial level, tool length offset |
| N004 | S30 M3 | Spindle start |
| N005 | G99 G81X400.0 R Y-350.0 | |
| | Z–153.0R–97.0 F120; | Positioning, then #1 drilling |
| N006 | Y–550.0; | Positioning, then #2 drilling and point R level return |
| N007 | G98Y–750.0; | Positioning, then #3 drilling and initial level return |
| N008 | G99X1200.0; | Positioning, then #4 drilling and point R level return |
| N009 | Y–550.0; | Positioning, then #5 drilling and point R level return |
| N010 | G98Y–350.0; | Positioning, then #6 drilling and initial level return |
| N011 | G00X0Y0M5; | Reference position return, spindle stop |
| N012 | G49Z250.0T15M6; | Tool length offset cancel, tool change |
| N013 | G43Z0H15; | Initial level, tool length offset |
| N014 | S20M3; | Spindle start |
| N015 | G99G82X550.0Y-450.0 | Positioning, then #7 drilling, point R level return |
| | Z-130.0R-97.0P300F70; | |
| N016 | G98Y–650.0; | Positioning, then #8 drilling, initial level return |
| N017 | G99X1050.0; | Positioning, then #9 drilling, point R level return |
| N018 | G98Y–450.0; | Positioning, then #10 drilling, initial level return |
| N019 | G00X0Y0M5; | Reference position return, spindle stop |
| N020 | G49Z250.0T31M6; | Tool length offset cancel, tool change |
| N021 | G43Z0H31; | Initial level, tool length offset |
| N022 | S10M3; | Spindle start |
| N023 | G85G99X800.0Y-350.0 | Positioning, then #11 drilling, point R level return |
| | Z–153.0R47.0F50; | |
| N024 | G91Y–200.0K2; | Positioning, then #12, 13 drilling. point R level return |
| N025 | G28X0Y0M5; | Reference position return, spindle stop |
| N026 | G49Z500.0; | Tool length offset cancel |
| N027 | M0; | Program stop |
| | | |

| 13.2 PICID TAPPING | The tapping cycle (G84) and left–handed tapping cycle (G74) may performed in standard mode or rigid tapping mode. | | | | |
|-----------------------|---|--|--|--|--|
| RIGID TAFFING | 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. | | | | |
| | Rigid mode eliminates the need to use a floating tap required in the standard tapping mode, thus allowing faster and more precise tapping. | | | | |

13.2.1 Rigid Tapping (G84)

Format

When the spindle motor is controlled in rigid mode as if it were a servo motor, a tapping cycle can be sped up.



| Explanations | After positioning along the X– and Y–axes, rapid traverse is performed to point R. Tapping is performed from point R to point Z. When tapping is completed, a dwell is performed and the spindle is stopped. The spindle is then rotated in the reverse direction, the tool is retracted to point R, then the spindle is stopped. Rapid traverse to initial level is then performed. While tapping is being performed, the feedrate override and spindle override are assumed to be 100%. However, the speed for extraction (operation 5) can be overridden by up to 200% depending on the setting at bit 4 of parameter and parameter 063 and parameter 258. | | | | | |
|--|---|---|--|--|--|--|
| Rigid mode | Rigid mode can be specified using any of the following methods: Specify M29 S***** before a tapping command. Specify M29 S***** in a block which contains a tapping command. Specify G84 for rigid tapping (parameter No. 076 #3 = 1). | | | | | |
| Thread lead | In feed-per-minute mode, the thread lead is obtained from the expression, feedrate ÷ spindle speed. If a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R. | | | | | |
| Restrictions | | | | | | |
| Axis switching | Before the drilling axis can be changed, the canned cycle must be canceled. If the drilling axis is changed in rigid mode, alarm (No. 206) is issued. | | | | | |
| • S command | If a speed higher than the maximum speed for the gear being used is specified, alarm (No. 200) is issued. | | | | | |
| • F command | If a value exceeding the upper limit of cutting feedrate is specified, alarm (No. 011) is issued. | | | | | |
| • M29 | If an S command and axis movement are specified between M29 and G84, alarm (No. 203) is issued. If M29 is specified in a tapping cycle, alarm (No. 204) is issued. | | | | | |
| • R | Specify R in a block that performs drilling. If R is specified in a non-drilling block, it is not stored as modal data. | | | | | |
| Cancel | Do not specify a group 01 G code (G00 to G03) and G73 in the same block. If they are specified together, G73 is canceled. | | | | | |
| • Unit of F | | Metric input | Inch input | Remarks | | |
| | G94 | 1 mm/min | 0.01 inch/min | Decimal point programming allowed | | |
| | G95 | 0.01 mm/rev | 0.0001 inch/rev | Decimal point programming allowed | | |
| Examples | Z–axis f Spindle Thread <progra G94 ; G00 X12 M29 S10</progra | feedrate 1000 speed 1000 r lead 1.0 mm mming of fee 20.0 Y100.0 ; | mm/min pm d per minute> Specify a Positionin Rigid mo | n feed–per–minute command. ng de specification | | |
| G84 Z-100.0 R-20.0 F1000 ; Rigid tapping | | | | | | |
13.2.2 Left–handed Rigid Tapping Cycle (G74)

Format

When the spindle motor is controlled in rigid mode as if it were a servo motor, tapping cycles can be sped up.



Explanations

After positioning along the X– and Y–axes, rapid traverse is performed to point R.

Tapping is performed from point R to point Z. When tapping is completed, a dwell is performed and the spindle is stopped. The spindle is then rotated in the normal direction, the tool is retracted to point R, then the spindle is stopped. Rapid traverse to initial level is then performed. While tapping is being performed, the feedrate override and spindle override are assumed to be 100%.

However, the speed for extraction (operation 5) can be overridden by up to 200% depending on the setting at bit 4 of parameter 063 and parameter 258.

Rigid mode

Rigid mode can be specified using any of the following methods:

- Specify M29 S***** before a tapping command.
- Specify M29 S***** in a block which contains a tapping command.
- Specify G84 for rigid tapping. (parameter No. 076#3 = 1).

| 13. FUNCTIONS TO SIMPLIFY PROGRAMMING | | PROGRAMMIN | G | | B6 | 2574EN/02 |
|--|---|---|---|---|------------------------------------|------------|
| Thread lead | In feed expressi If a tool the offse | -per-minute on, feedrate ÷ s length offset (C et is applied at | node, the threa pindle speed. 643, G44, or G49 the time of posit | ad lead is) is specified ioning to poi | obtained I in the car int R. | from the |
| Restrictions | | | | | | |
| Axis switching | Before canceled is issued | Before the drilling axis can be changed, the canned cycle must be canceled. If the drilling axis is changed in rigid mode, alarm (No. 206) is issued. | | | | |
| S command | Specify used cau | Specifying a rotation speed exceeding the maximum speed for the gear used causes alarm (No. 200). | | | | |
| F command | Specifying a value that exceeds the upper limit of cutting feedrate causes alarm (No. 011). | | | | | |
| M29 | Specifying an S command or axis movement between M29 and G84 causes alarm (No. 203). Then, specifying M29 in the tapping cycle causes alarm (No. 204). | | | | | |
| R | Specify R in a block that performs drilling. If R is specified in a non-drilling block, it ss not stored as modal data. | | | | | |
| Cancel | Do not s If they a | pecify a group (are specified to |)1 G code (G00 to gether, G73 is ca | GO3) and G nceled. | 73 in the s | ame block. |
| Unit of F | | Metric input | Inch input | F | Remarks | |
| | G94 | 1 mm/min | 0.01 inch/min | Decimal poin | t programmi | ng allowed |
| | G95 | 0.01 mm/rev | 0.0001 inch/rev | Decimal poin | t programmi | ng allowed |
| Examples | Z–axis f Spindle Thread <progra G94 ; G00 X1: M29 S1 G84 Z–</progra | feedrate 1000 speed 1000 r lead 1.0 mm amming for fee 20.0 Y100.0 ; 000 ; 100 0 R-20 0 F | mm/min pm ed per minute> Specify a Positionia Rigid mo | n feed–per–n ng de specificat | ninute con tion | nmand. |

| 13.2.3 Peck Rigid Tapping Cycle (G84 or G74) | Tapping a deep hole in rigid tapping mode may be difficult due to chips sticking to the tool or increased cutting resistance. In such cases, the peck rigid tapping cycle is useful. In this cycle, cutting is performed several times until the bottom of the hole is reached. Two peck tapping cycles are available: High–speed peck tapping cycle and standard peck tapping cycle. These cycles are selected using the bit (bit 0) of parameter 388. | | | |
|--|---|--|--|--|
| Format | |] | | |
| | G84 (or G74) X_Y_Z_R_P_Q_ X_Y_ : Hole position data Z_ : The distance from point position of the bottom of R_ : The distance from the in P_ : Dwell time at the bottom return is made Q_ : Depth of cut for each cut F_ : The cutting feedrate K Number of repeats | _ F _ K _ ; R to the bottom of the hole or the the hole itial level to point R level of the hole and at point R when a tting feed | | |
| | G84, G74 (G98) | G84, G74 (G99) | | |
| High–speed peck tapping cycle (Parameter No.388#0=0) (1) The tool operates at a normal cutting feedrate. The normal time constant is used. (2) Retraction can be overridden. The retraction time constant is used. | d=retraction distance | Point R Point R level q (1) q d dq q d $dPoint Z d$ | | |
| Peck tapping cycle (Parameter No.388#0=1) (1) The tool operates at a normal auttion foodlate. The normal time | d=cutting start distance | | | |
| cutting feedrate. The normal time constant is used. (2) Retraction can be overridden. The retraction time constant is used. (3) Retraction can be overridden. The normal time constant is used. During a rigid tapping cycle, CNC checks whether the pulse distribution is performed or not and start the next operation at the end of each operation of (1) and (2) in the peck tapping cycle. | Point R Point R level | Point R Point R level | | |

Explanations

| High–speed peck tapping cycle | After positioning along the X– and Y–axes, rapid traverse is performed to point R. 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 bit (bit 4) of parameter 063 specifies whether retraction can be overridden or not. When point Z has been reached, the spindle is stopped, then rotated in the reverse direction for retraction. Set the retraction distance, d, in parameter 403. | | | |
|---|--|--------------|-----------------|-----------------------------------|
| • Peck tapping cycle | After positioning along the X– and Y–axes, rapid traverse is performed to point R level. From point R, cutting is performed with depth Q (depth of cut for each cutting feed), then a return is performed to point R. The bit (bit 4) of parameter 063 specifies whether the retraction can be overridden or not. Traverse is performed from point R to a position distance d from the end point of the last cutting, which is where cutting is restarted. For this traverse, the specification of the bit (bit 4) of parameter 063 is also valid. When point Z has been reached, the spindle is stopped, then rotated in the reverse direction for retraction. Set d (distance to the point at which cutting is started) in parameter 403. | | | |
| • Restrictions | | | | |
| • Axis switching | Before the drilling axis can be changed, the canned cycle must be canceled. If the drilling axis is changed in rigid mode, alarm (No. 206) is issued. | | | |
| • S command | Specifying a rotation speed exceeding the maximum speed for the gear used causes alarm (No. 200). | | | |
| • F command | Specifying a value that exceeds the upper limit of cutting feedrate causes alarm (No. 011). | | | |
| • M29 | Specifying an S command or axis movement between M29 and G84 causes alarm (No. 203). Then, specifying M29 in the tapping cycle causes alarm (No. 204). | | | |
| ● Q/R | Specify Q and R in a block that performs drilling. If they are specified in a block that does not perform drilling, they are not stored as modal data. When Q0 is specified, the peck rigid tapping cycle is not performed. | | | |
| Cancel | Do not specify a group 01 G code (G00 to G03) and G73 in the same block. If they are specified together, G73 is canceled. | | | |
| • Unit of F | | Metric input | Inch input | Remarks |
| | G94 | 1 mm/min | 0.01 inch/min | Decimal point programming allowed |
| | G95 | 0.01 mm/rev | 0.0001 inch/rev | Decimal point programming allowed |

The rigid tapping canned cycle is canceled. For how to cancel this cycle, see Section 13.1.13.

13.3 CANNED GRINDING CYCLE (FOR 0–GSD ONLY)

Canned grinding cycles make it easier for the programmer to create programs that include grinding. With a canned grinding cycle, repetitive operation peculiar to grinding can be specified in a single block with a G function; without canned grinding cycles, normally more than one block is required. In addition, the use of canned grinding cycles shortens the program to save memory. The following four canned grinding cycles are available:

·Plunge grinding cycle (G75)

·Direct constant-dimension plunge grinding cycle (G77)

·Continuous-feed surface grinding cycle (G78)

·Intermittent-feed surface grinding cycle (G79)

A plunge grinding cycle is performed.

13.3.1 Plunge Grinding Cycle (G75)

Format

Explanations

• Grinding wheel cutting

- Dwell
- Grinding
- Grinding wheel cutting
- Dwell
- Grinding (return direction)

The plunge grinding cycle consists of six operation sequences. Operations (1) to (6) are repeated until the depth reaches the total depth of cut specified at address K. In the single block stop mode, operations (1) to (6) are performed every cycle start.

- (1) Cutting is performed along the Y-axis in cutting feed mode for the amount specified by I (depth of cut 1). The feedrate is specified by R.
- (2) Dwell is performed for the time specified by P.
- (3) Cutting feed is performed for the amount specified by X (or Z). The feedrate is specified by F.
- (4) Cutting is performed along the Y-axis in cutting feed mode for the amount specified by J (depth of cut 2). The feedrate is specified by R.
- (5) Dwell is performed for the time specified by P.
- (6) Feeding is performed in the reverse direction for the amount specified by X (or Z) at a feedrate specified by F.

Restrictions

- X(Z), I, J, K
- Clear
- Operation performed when the total depth of cut is reached

X, (Z), I, J, and K must all be specified in incremental mode.

I, J, X, and Z in canned cycles are modal data common to G75, G77, G78, and G79. They remain valid until new data is specified. They are cleared when a group 00 G code other than G04 or a group 01 G code other than G75, G77, G78, and G79 is specified.

When the total depth of cut is reached during cutting using I or J, the subsequent operation sequences (up to 6) are executed, then the cycle terminates. In this case, no further cutting is performed after the total depth of cut is reached.

· Chart of operation in which the total depth of cut is reached by cutting specified by I and J:



 Chart of operation in which the total depth of cut is reached during cutting specified by I and J:



— 135 —

13.3.2 Direct Constant–dimension Plunge Grinding Cycle (G77)

Format

A direct constant-dimension plunge grinding cycle is performed.

| G77 I_ J_ K_ X(Z)_ R_ F_ P_ L_ ; |
|---|
| I_ : Depth-of-cut 1 (A sign in the command specifies the direction |
| of cutting.) |
| J_ : Depth-of-cut 2 (A sign in the command specifies the direction |
| of cutting.) |
| K_ : Total depth of cut |
| $X(Z)_{-}$: Range of grinding (A sign in the command specifies the direction |
| of grinding.) |
| R_ : Feedrate for I and J |
| F_ : Feedrate for X (Z) |
| P_ : Dwell time |
| L_ : Grinding–wheel wear compensation (Only for continuous |
| dressing) |
| G77 |
| |
| Y X(Z) |
| |
| <u></u> (1) (R) (3) (F) |
| $ \begin{array}{c} \downarrow \\ \downarrow \uparrow \\ \downarrow \uparrow \end{array} $ |
| (6) (F) (5) P |
| |
| <pre></pre> |
| ↓ |
| |

Explanations

- Grinding wheel cutting
- Dwell
- Grinding
- Grinding wheel cutting
- Dwell
- Grinding (return direction)

The constant-dimension plunge grinding cycle consists of six operation sequences. Operations (1)1 to (6) are repeated until the depth reaches the total depth of cut specified at address K.

- (1) Cutting is performed along the Y-axis in cutting feed mode for the amount specified by I (depth of cut 1). The feedrate is specified by R.
- (2) Dwell is performed for the time specified by P.
- (3) Cutting feed is performed for the amount specified by X (or Z). The feedrate is specified by F.
- (4) Cutting is performed along the Y-axis in cutting feed mode for the amount specified by J (depth of cut 2). The feedrate is specified by R.
- (5) Dwell is performed for the time specified by P.
- (6) Feeding is performed in the reverse direction for the amount specified by X (or Z) at a feedrate specified by F.

Skip signal

When the cycle is performed using G77, a skip signal can be input to terminate the cycle. When a skip signal is input, the current operation sequence is interrupted or completed, then the cycle is terminated. The following shows how the system operates when the skip signal is input during each operation sequence.

• When the skip signal is input during operation sequence 1 or 4 (cutting feed specified by I or J), cutting is stopped immediately and the tool returns to the X (Z) coordinate at which the cycle started.



- When the skip signal is input during operation sequence 2 or 5 (dwell), dwell is stopped immediately and the tool returns to the X (Z) coordinate at which the cycle started.
- When the skip signal is input during operation sequence 3 or 6 (movement), the tool returns to the X (Z) coordinate at which the cycle started after the movement specified by X (Z) is completed.



Restrictions

- X(Z), I, J, K
- Clear

X, (Z), I, J, and K must all be specified in incremental mode.

I, J, X, and Z in canned cycles are modal data common to G75, G77, G78, and G79. They remain valid until new data is specified. They are cleared when a group 00 G code other than G04 or a group 01 G code other than G75, G77, G78, and G79 is specified.

13.3.3 Continuous–feed Surface Grinding Cycle (G78)

Format

A continuous-feed surface grinding cycle is performed.

| G78 I_ (J_) K_ X_ F_ P_ L_ ; | | | | |
|--|---|--|--|--|
| I_ : Depth-of-cut 1 (A sign in the command specifies the direction | | | | |
| of cutting.) | | | | |
| J_ : Depth–of–cut 2 (A sign in the command specifies the direction | ۱ | | | |
| of cutting.) | | | | |
| K_ : Total depth of cut | | | | |
| X(Z)_: Range of grinding (A sign in the command specifies the | | | | |
| direction of grinding.) | | | | |
| R_ : Feedrate for I and J | | | | |
| F_ : Feed rate | | | | |
| P_ : Dwell time | | | | |
| L_ : Grinding–wheel wear compensation (Only for continuous | | | | |
| dressing) | | | | |
| G78 | | | | |
| 7 | | | | |
| | | | | |
| ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ | | | | |
| | | | | |
| (1) P(Dwell) $(2) (F)$ | | | | |
| | | | | |
| J ↓ (4) (F) (3) P(Dwell) | | | | |
| | | | | |
| | | | | |
| X | | | | |
| | | | | |

Explanations

The continuous-feed surface grinding cycle consists of four operation sequences. Operations (1) to (4) are repeated until the depth reaches the total depth of cut specified in address K. In the single block stop mode, operations (1) to (4) are performed every cycle start.

- (1) Dwell
- (2) Grinding
- (3) Dwell
- (4) Grinding (in reverse direction)

Restrictions

• J

When J is omitted, it is assumed to be 1. J is valid only in the block where it is specified.

• I, J, K, X

X, (Z), I, J, and K must all be specified in incremental mode.

- Clear
- Operation performed when the total depth of cut is reached

I, J, X, and Z in canned cycles are modal data common to G75, G77, G78, and G79. They remain valid until new data is specified. They are cleared when a group 00 G code other than G04 or a group 01 G code other than G75, G77, G78, and G79 is specified.

When the total depth of cut is reached during cutting using I or J, the subsequent operation sequences (up to 6) are executed, then the cycle terminates. In this case, no further cutting is performed after the total depth of cut is reached.

· Chart of operation in which the total depth of cut is reached by cutting specified by I and J:



• Chart of operation in which the total depth of cut is reached during cutting specified by I and J:



13.3.4 Intermittent–feed Surface Grinding Cycle (G79)

Format

An intermittent-feed surface grinding cycle is performed.

| G79 I_ J_ K_ X_ R_ F_ P_ L_ ; |
|--|
| I_ : Depth–of–cut 1 (A sign in the command specifies the direction |
| of cutting.) |
| J_ : Depth–of–cut 2 (A sign in the command specifies the direction |
| of cutting.) |
| K_ : Total depth of cut |
| X(Z)_: Range of grinding (A sign in the command specifies the |
| direction of grinding.) |
| R_ : Feedrate for I and J |
| F_ : Feedrate for X (Z) |
| P_ : Dwell time |
| L_ : Grinding–wheel wear compensation (Only for continuous |
| dressing) |
| G79 |
| |
| z X |
| <> |
| (1) (B) (C) |
| $\frac{1}{1} \frac{1}{1} \frac{1}$ |
| $ \underbrace{J \uparrow}_{(4)} (2) P \underbrace{J (4)}_{(7)} (2) P \underbrace{J (4)}$ |
| (5) P |
| ╡───┤ |
| X |
| |

Explanations

- Grinding wheel cutting
- Dwell
- Grinding
- Grinding wheel cutting
- Dwell
- Grinding (return direction)

The intermittent-feed surface grinding cycle consists of six operation sequences. Operations (1) to (6) are repeated until the depth reaches the total depth of cut specified at address K. In the single block stop mode, operations (1) to (6) are performed every cycle start.

- (1) Cutting is performed along the Z-axis in cutting feed mode for the amount specified by I (depth of cut 1). The feedrate is specified by R.
- (2) Dwell is performed for the time specified by P.
- (3) Cutting feed is performed for the amount specified by X (or Z). The feedrate is specified by F.
- (4) Cutting is performed along the Z-axis in cutting feed mode for the amount specified by J (depth of cut 2). The feedrate is specified by R.
- (5) Dwell is performed for the time specified by P.
- (6) Feeding is performed in the reverse direction for the amount specified by X at a feedrate specified by F.

Restrictions

- X, I, J, K
- Clear

X, (Z), I, J, and K must all be specified in incremental mode.

I, J, X, and Z in canned cycles are modal data common to G75, G77, G78, and G79. They remain valid until new data is specified. They are cleared when a group 00 G code other than G04 or a group 01 G code other than G75, G77, G78, and G79 is specified.

13.4 AUTOMATIC GRINDING WHEEL DIAMETER COMPENSATION AFTER DRESSING

13.4.1 Checking the Minimum Grinding Wheel Diameter (for 0–GSD only)

Compensation amounts set in offset memory can be modified by using the external tool compensation function or programming (by changing offsets using custom macro variables).

With these functions, the compensation amount for the diameter of the dressed grinding wheel can be changed.

If the compensation amount associated with the offset number specified in the H code is smaller than the minimum grinding wheel diameter specified in parameter 838 when programmed compensation (using G43 or G44) is performed, a signal is output to the PMC.



13.5 IN-FEED GRINDING ALONG THE Y AND Z AXES AT THE END OF TABLE SWING (FOR 0-GSD ONLY)

Every time an external signal is input, cutting is performed by a fixed amount according to the programmed profile in the specified Y–Z plane.

Format

| G | 161 R_ ; | | | | | | |
|---|-----------------|---|--|--|--|--|--|
| | profile program | n | | | | | |
| G | 160 ; | | | | | | |

Explanations

- G161 R_
- Profile program

• G160

Restrictions

• Profile program

Examples

Specify the start of an operation mode and profile program. Also specify the depth of cut in R.

Program a workpiece figure in the Y–Z plane using linear interpolation (G01) and/or circular interpolation (G02 or G03). One or more blocks can be specified.

Cancel the operation mode (end of the profile program).

Do not specify codes other than G01, G02, and G03 within the profile program.



In the above program, every time the in-feed cutting start signal is input, the tool is moved by 10.000 along the machining profile shown above. α = Travel distance for each in-feed control cutting start signal input

The feedrate is programmed with an F code.

| 13.6 EXTERNAL MOTION FUNCTION (G81) | Upon completion of positioning in each block in the program, an external operation function signal can be output to allow the machine to perform specific operation. Concerning this operation, refer to the manual supplied by the machine tool builder. |
|---|---|
| Format | |
| | G81 IP_; (IP_ Axis move command) |
| Explanations | Every time positioning for the IP_move command is completed, the CNC sends a external operation function signal to the machine. An external operation signal is output for each positioning operation until canceled by G80 or a group 01 G code. |
| Restrictions | |
| A block without X or Y axis | No external operation signals are output during execution of a block that contains neither X nor Y. |
| | G81 is also used as a G code for a drilling canned cycle (see Subsection 13.1.4). Use bit 4 of parameter No. 011 to specify whether G81 is to be used for the above function or for a canned cycle. |

COMPENSATION FUNCTION

General

This chapter describes the following compensation functions:

| TOOL LENGTH OFFSET (G43, G44, G49) | Sec.14.1 |
|---|----------------|
| CUTTER COMPENSATION C (G40–G42) | Sec.14.2, 14.3 |
| TOOL COMPENSATION VALUES, NUMBER OF COM | PENSATION |
| VALUES, AND ENTERING VALUES FROM THE PROG | GRAM (G10) |
| | Sec.14.4 |
| NORMAL DIRECTION CONTROL (G150, G151, G152) | Sec.14.5 |

14.1 TOOL LENGTH OFFSET (G43,G44,G49)

This function can be used by setting the difference between the tool length assumed during programming and the actual tool length of the tool used into the offset memory. It is possible to compensate the difference without changing the program.

Specify the direction of offset with G43 or G44. Select a tool length offset value from the offset memory by entering the corresponding address and number (H code).



Fig14.1(a) Tool length offset

The following three methods of tool length offset can be used, depending on the axis along which tool length offset can be made.

· Tool length offset A

Compensates for the difference in tool length along the Z-axis.

- **Tool length offset B** Compensates for the difference in tool length along the X–,Y–,or
- Z–axis.
- Tool length offset C

Compensates for the difference in tool length along a specified axis.

| Tool length offset A | G43 Z_ H_ ; G44 Z_ H_ ; | Explanation of each address G43 : Positive offset |
|---------------------------|--|---|
| Tool length offset B | G17 G43 Z_ H_ ; G17 G44 Z_ H_ ; G18 G43 Y_ H_ ; G18 G44 Y_ H_ ; G19 G43 X_ H_ ; G19 G44 X_ H_ ; | G44 : Negative offset G17 : XY plane selection G18 : ZX plane selection G19 : YZ plane selection α : Address of a specified axis H : Code for specifying the |
| Tool length offset C | G43 α_ H_ ; G44 α_ H_ ; | |
| Tool length offset cancel | G49 ; or H0 ; | |

Format

— 146 —

Explanations

Direction of the offset

| Selection of tool length | Select tool length offset A, B, or C, by setting bit 6 of parameter 003 and |
|--|---|
| offset | bit 3 of parameter No. 019. |

When G43 is specified, the tool length offset value (stored in offset memory) specified with the H code is added to the coordinates of the end position specified by a command in the program. When G44 is specified, the same value is subtracted from the coordinates of the end position. The resulting coordinates indicate the end position after compensation, regardless of whether the absolute or incremental mode is selected. If movement along an axis is not specified, the system assumes that a move command that causes no movement is specified. When a positive value is specified for tool length offset with G43, the tool is moved accordingly in the positive direction. When a positive value is specified with G44, the tool is moved accordingly in the negative direction. When a negative value is specified, the tool is moved in the opposite direction. G43 and G44 are modal G codes. They are valid until another G code belonging to the same group is used.

• Specification of the tool length offset value specified in the H code is selected from offset memory and added to or subtracted from the commands specified by a command in the program. The tool length offset value may be set in the offset memory through the CRT/MDI panel.

The range of values that can be set as the tool length offset value is as follows.

| Increment system | Metric input | Inch input |
|------------------|------------------|--------------------|
| IS–B | 0 to ±999.999mm | 0 to ±99.9999inch |
| IS-C | 0 to ±999.9999mm | 0 to ±99.99999inch |

WARNING

When the tool length offset value is changed due to a change of the offset number, the offset value changes to the new tool length offset value, the new tool length offset value is not added to the old tool length offset value.

H1 : tool length offset value 20.0

H2 : tool length offset value 30.0

G90 G43 Z100.0 H1 ; Z will move to 120.0

G90 G43 Z100.0 H2 ; Z will move to 130.0

NOTE

The tool length offset value corresponding to offset No. 0, that is, H0 always means 0. It is impossible to set any other tool length offset value to H0.

— 147 —

| Performing tool length offset along two or more axes | Tool length offset B can be executed along two or more axes when the axes are specified in two or more blocks. | | |
|--|---|--|--|
| | Offset in X and Y axes. G19 G43 H_; Offset in X axis G18 G43 H_; Offset in Y axis (Offsets in X and Y axes are performed) | | |
| | If the bit (bit 5 of parameter No. 036) is set to 1, an alarm will not occur even when tool length offset C is executed along two or more axes at the same time. | | |
| Tool length offset cancel | To cancel tool length offset, specify G49 or H0. After G49 or H0 is specified, the system immediately cancels the offset mode. | | |
| | WARNING After tool length offset B is executed along two or more axes, offset along all the axes is canceled by specifying G49. If H0 is specified, only offset along an axis perpendicular to the specified plane is canceled. | | |
| | CAUTION | | |

In the case of the offset in three axes or more, if the offset is canceled by G49 code, the P/S alarm 015 is generated. Cancel the offset by using G49 and H00.

Examples



· Program

| H1=-4.0 (Tool length offset value) |
|------------------------------------|
| N1 G91 G00 X120.0 Y80.0; (1) |
| N2 G43 Z–32.0 H1 ; (2) |
| N3 G01 Z–21.0 F1000 ; (3) |
| N4 G04 P2000 ; (4) |
| N5 G00 Z21.0; |
| N6 X30.0 Y–50.0 ; (6) |
| N7 G01 Z-41.0; |
| N8 G00 Z41.0 ; |
| N9 X50.0 Y30.0 ; |
| N10G01 Z-25.0; |
| N11 G04 P2000 ; |
| N12G00 Z57.0 H0; (12) |
| N13X–200.0 Y–60.0 ; |
| N14M2 ; |

14.2 OVERVIEW OF CUTTER COMPENSATION C (G40 – G42)

When the tool is moved, the tool path can be shifted by the radius of the tool (Fig. 14.2 (a)).

To make an offset as large as the radius of the tool, CNC first creates an offset vector with a length equal to the radius of the tool (start–up). The offset vector is perpendicular to the tool path. The tail of the vector is on the workpiece side and the head positions to the center of the tool.

If a linear interpolation or circular interpolation command is specified after start–up, the tool path can be shifted by the length of the offset vector during machining.

To return the tool to the start position at the end of machining, cancel the cutter compensation mode.



Fig. 14.2 (a) Outline of Cutter Compensation C

In cutter compensation C, an appropriately offset tool path can be obtained for both the inner and outer corners simply by specifying the offset direction. Unlike cutter compensation B, programs can easily be created without having to consider the difference between the outer and inner corners.

Format

ullet

•

| Start up (Tool compensation start) | G00(or G01)G41(o | r G42) IP_ H_ ; | |
|---|--|--|-----------------------|
| (reer compensation start) | G41 : Cutter co | mpensation left (Group07) | |
| | G42 : Cutter co | mpensation right (Group07) | |
| | IP_ : Comman | d for axis movement | |
| | H_ : Code for (H code) | specifying as the cutter compensa | tion value(1–3digits) |
| Cutter compensation cancel (offset mode cancel) | G40 <u>₽</u> _; | | |
| (| G40 ∶ Cutter co (Offset m IP_ ∶ Comman | mpensation cancel(Group 07) node cancel) d for axis movement | |
| Selection of the offset plane | Offset plane | Command for plane selection | IP_ |
| | ХрҮр | G17 ; | Xp_Yp_ |
| | ZpXp | G18 ; | Xp_Zp_ |
| | YpZp | G19 ; | Yp_Zp_ |
| | | | <u> </u> |

Explanations

| Offset cancel mode | At the beginning when power is applied the control is in the cancel mode. In the cancel mode, the vector is always 0, and the tool center path coincides with the programmed path. |
|--|---|
| • Start Up | When a cutter compensation command (G41 or G42, nonzero dimension words in the offset plane, and H code other than H0) is specified in the offset cancel mode, the CNC enters the offset mode. Moving the tool with this command is called start–up. Specify positioning (G00) or linear interpolation (G01) for start–up. If circular interpolation (G02, G03) is specified, alarm 34 occurs. When processing the start–up block and subsequent blocks, the CNC prereads two blocks. The second preread block is not indicated. |
| • Offset mode | In the offset mode, compensation is accomplished by positioning (G00), linear interpolation (G01), or circular interpolation (G02, G03). If two or more blocks that do not move the tool (miscellaneous function, dwell, etc.) are processed in the offset mode, the tool will make either an excessive or insufficient cut. If the offset plane is switched in the offset mode, alarm 37 occurs and the tool is stopped. |

Offset mode cancel

In the offset mode, when a block which satisfies any one of the following conditions is executed, the equipment enters the offset cancel mode, and the action of this block is called the offset cancel.

- 1. G40 has been commanded.
- 2. 0 has been commanded as the offset number for cutter compensation.

When performing offset cancel, circular arc commands (G02 and G03) are not available. If a circular arc is commanded, an alarm (No. 034) is generated and the tool stops.

In the offset cancel, the control executes the instructions in that block and the block in the cutter compensation buffer. In the meantime, in the case of a single block mode, after reading one block, the control executes it and stops. By pushing the cycle start button once more, one block is executed without reading the next block.

Then the control is in the cancel mode, and normally, the block to be executed next will be stored in the buffer register and the next block is not read into the buffer for cutter compensation.



Fig. 14.2 (b) Changing the offset mode

In general, the cutter compensation value shall be changed in the cancel mode, when changing tools. If the cutter compensation value is changed in offset mode, the vector at the end point of the block is calculated for the new cutter compensation value.



Fig. 14.2 (c) Changing the Cutter Compensation Value

• Change of the Cutter compensation value Positive/negative cutter compensation value and tool center path If the offset amount is negative (–), distribution is made for a figure in which G41's and G42's are all replaced with each other on the program. Consequently, if the tool center is passing around the outside of the workpiece, it will pass around the inside, and vice versa.

The figure below shows one example. Generally, the offset amount is programmed to be positive (+).

When a tool path is programmed as in ((1)), if the offset amount is made negative (–), the tool center moves as in ((2)), and vice versa. Consequently, the same tape permits cutting both male and female shapes, and any gap between them can be adjusted by the selection of the offset amount. Applicable if start–up and cancel is A type. (See 14.3.2 and 14.3.4)



Fig. 14.2 (d) Tool Center Paths when Positive and Negative Cutter Compensation Values are Specified

• Cutter compensation value setting

Assign a cutter compensation values to the H codes on the CRT/MDI panel. The table below shows the range in which cutter compensation values can be specified.

| Increment system | mm input | inch input |
|------------------|---------------|------------------|
| IS–B | 0–±999.999mm | 0–±99.99999inch |
| IS-C | 0–±999.9999mm | 0-±99.999999inch |

NOTE

- 1 The cutter compensation value corresponding to offset No. 0, that is, H0 always means 0. It is impossible to set H0 to any other offset amount.
- 2 Cutter compensation C can be specified by D code with parameter (No. 036 #6) set to 1.

Offset vector

The offset vector is the two dimensional vector that is equal to the cutter compensation value assigned by H code. It is calculated inside the control unit, and its direction is up–dated in accordance with the progress of the tool in each block.

The offset vector is deleted by reset.

| Specifying a cutter compensation value | Specify a cutter compensation value with a number assigned to it. The number consists of 1 to 3 digits after address H (H code). The H code is valid until another H code is specified. The H code is used to specify the tool offset value as well as the cutter compensation value. When an H code for cutter compensation is specified during tool length compensation, the amount of tool length compensation remains as is. When an H code for tool length compensation is specified during cutter compensation, however, the cutter compensation, as well as tool length compensation, is changed. Do not modify the tool length compensation in cutter compensation mode. The amount of tool length compensation and that of cutter compensation can be separately specified with the H and D codes by setting bit 6 of parameter No. 036 appropriately. Only the tool length compensation can be changed in cutter compensation. Even in this case, the tool length compensation is equal to the cutter compensation. That is, the setting of H10 is equal to that of D10. |
|--|--|
| Plane selection and vector | Offset calculation is carried out in the plane determined by G17, G18 and G19, (G codes for plane selection). This plane is called the offset plane. Compensation is not executed for the coordinate of a position which is not in the specified plane. The programmed values are used as they are. In simultaneous 3 axes control, the tool path projected on the offset plane is compensated. The offset plane is changed during the offset cancel mode. If it is performed during the offset mode, an alarm (No. 37) is displayed and the machine is stopped. |

Examples



G92 X0 Y0 Z0 ; Specifies absolute coordinates.

N1 G90 G17 G00 G41 H07

| | | the tool i |
|-----|-----------------------------|------------|
| N2 | G01 Y900.0 F150 ; | Specifies |
| N3 | X450.0; | Specifies |
| N4 | G03 X500.0 Y1150.0 R650.0 : | Specifies |
| N5 | G02 X900.0 R-250.0 ; | Specifies |
| N6 | G03 X950.0 Y900.0 R650.0 ; | Specifies |
| N7 | G01 X1150.0 ; | Specifies |
| N8 | Y550.0 ; | Specifies |
| N9 | X700.0 Y650.0 ; | Specifies |
| N10 | X250.0 Y550.0 ; | Specifies |
| N11 | G00 G40 X0 Y0 ; | Cancels |
| | | The tool |

The tool is positioned at the start position (X0, Y0, Z0).

| | The tool is positioned at the start position (110, 10, 20). |
|----------------|---|
| X250.0 Y550.0; | Starts cutter compensation (start-up). The tool is shifted to the |
| | left of the programmed path by the distance specified in D07. |
| | In other words the tool path is shifted by the radius of the tool |
| | (offset mode) because D07 is set to 15 beforehand (the radius of |
| | the tool is 15 mm). |
| | Specifies machining from P1 to P2. |
| | Specifies machining from P2 to P3. |
| 650.0 : | Specifies machining from P3 to P4. |
| | Specifies machining from P4 to P5. |
| 50.0 ; | Specifies machining from P5 to P6. |
| | Specifies machining from P6 to P7. |
| | Specifies machining from P7 to P8. |
| | Specifies machining from P8 to P9. |
| | Specifies machining from P9 to P1. |
| | Cancels the offset mode. |
| | The tool is returned to the start position (X0, Y0, Z0). |
| | |

14.3This sectionDETAILS OF CUTTERfor cutter
This sectionCOMPENSATION C14.3.1 G

This section provides a detailed explanation of the movement of the tool for cutter compensation C outlined in 14.5. This section consists of the following subsections:

14.3.1 General
14.3.2 Tool Movement in Start-up
14.3.3 Tool Movement in Offset Mode
14.3.4 Tool Movement in Offset Mode Cancel
14.3.5 Interference Check
14.3.6 Over cutting by Cutter Compensation
14.3.7 Input command from MDI

14.3.1 General

• Inner side and outer side

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



• Meaning of symbols

The following symbols are used in subsequent figures:

- -S indicates a position at which a single block is executed once.
- SS indicates a position at which a single block is executed twice.
- SSS indicates a position at which a single block is executed three times.
- -L indicates that the tool moves along a straight line.
- C indicates that the tool moves along an arc.
- -r indicates the cutter compensation value.
- An intersection is a position at which the programmed paths of two blocks intersect with each other after they are shifted by *r*.
- indicates \odot the center of the tool.

14.3.2 Tool Movement in Start–up

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

Explanations

 Tool movement around an inner side of a corner (180° ≦ α)



 Tool movement around the outside of a corner at an obtuse angle (90° ≤ α<180°) Tool path in start–up has two types A and B, and they are selected by parameter (No. 016#2).



— 158 —

- B-62574EN/02
- Tool movement around Tool the outside of an acute para angle (α<90°)

Tool path in start–up has two types A and B, and they are selected by parameter (No.016#2).



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



 A block without tool movement specified at start-up If the command is specified at start–up, the offset vector is not created.



NOTE

For the definition of blocks that do not move the tool, see 14.3.3.

14.3.3 Tool Movement in Offset Mode

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

Explanations

 Tool movement around the inside of a corner (180° ≦ α)



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



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

 Tool movement around the outside corner at an obtuse angle (90° ≦ α<180°)



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


• When it is exceptional

End position for the arc is not on the arc

If the end of a line leading to an arc is programmed as the end of the arc by mistake as illustrated below, the system assumes that cutter compensation has been executed with respect to an imaginary circle that has the same center as the arc and passes the specified end position. Based on this assumption, the system creates a vector and carries out compensation. The resulting tool center path is different from that created by applying cutter compensation to the programmed path in which the line leading to the arc is considered straight.



The same description applies to tool movement between two circular paths.

If the cutter compensation value is sufficiently small, the two circular tool center paths made after compensation intersect at a position (P). Intersection P may not occur if an excessively large value is specified for cutter compensation. When this is predicted, alarm 33 occurs at the end of the previous block and the tool is stopped. In the example shown below, tool center paths along arcs A and B intersect at P when a sufficiently small value is specified for cutter compensation. If an excessively large value is specified, this intersection does not occur.



There is no inner intersection

• Change in the offset direction in the offset mode

If the center of the arc is identical with the start position or end point, alarm (No. 038) is displayed, and the tool will stop at the end position of the preceding block.

PROGRAMMING



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

| Sign of offset amount Gcode | + | _ |
|--------------------------------|-------------------|-------------------|
| G41 | Left side offset | Right side offset |
| G42 | Right side offset | Left side offset |

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

Tool center path with an intersection



Tool center path without an intersection

When changing the offset direction in block A to block B using G41 and G42, if intersection with the offset path is not required, the vector normal to block B is created at the start point of block B.



— 168 —

The length of tool center path larger than the circumference of a circle Normally there is almost no possibility of generating this situation. However, when G41 and G42 are changed, or when a G40 was commanded with address I, J, and K this situation can occur.

In this case of the figure, the cutter compensation is not performed with more than one circle circumference: an arc is formed from P_1 to P_2 as shown. Depending on the circumstances, an alarm may be displayed due to the "Interference Check" described later. To execute a circle with more than one circumference, the circle must be specified in segments.



Temporary cutter compensation cancel

If the following command is specified in the offset mode, the offset mode is temporarily canceled then automatically restored. The offset mode can be canceled and started as described in Subsections 15.6.2 and 15.6.4.

Specifying G28 (automatic return to the reference position) in the offset mode If G28 is specified in the offset mode, the offset mode is canceled at an intermediate position. If the vector still remains after the tool is returned to the reference position, the components of the vector are reset to zero with respect to each axis along which reference position return has been made.



Specifying G29 (automatic return from the reference position) in the offset mode

If G29 is commanded in the offset mode, the offset will be cancelled at the intermediate point, and the offset mode will be restored automatically from the subsequent block.



Cutter compensation G code in the offset mode

The offset vector can be set to form a right angle to the moving direction in the previous block, irrespective of machining inner or outer side, by commanding the cutter compensation G code (G41, G42) in the offset mode, independently. If this code is specified in a circular command, correct circular motion will not be obtained.

When the direction of offset is expected to be changed by the command of cutter compensation G code (G41, G42), refer to 15.3.3.



• G92 command in the offset mode

During offset mode, if G92 (absolute zero point programming) is commanded,the offset vector is temporarily cancelled and thereafter offset mode is automatically restored.

In this case, without movement of offset cancel, the tool moves directly from the intersecting point to the commanded point where offset vector is canceled. Also when restored to offset mode, the tool moves directly to the intersecting point.



— 171 —

A block without tool movement

The following blocks have no tool movement. In these blocks, the tool will not move even if cutter compensation is effected.

| M05 ; M code output S21 ; S code output G04 X10.0 ; Dwell G10 L11 P01 R10.0 ; Cutter compensation value setting (G17) Z200.0 ; . Move command not included in the offset plane. G90 ; G code only G91 X0 ;Move distance is zero. | Commands (1) to (6) are of no movement. |
|---|---|
|---|---|

A block without tool movement specified in offset mode

When a single block without tool movement is commanded in the offset mode, the vector and tool center path are the same as those when the block is not commanded. This block is executed at the single block stop point.



However, when the move distance is zero, even if the block is commanded singly, tool motion becomes the same as that when more than one block of without tool movement are commanded, which will be described subsequently.



Two blocks without tool movement should not be commanded consecutively. If commanded, a vector whose length is equal to the offset value is produced in a normal direction to tool motion in earlier block, so overcutting may result.



— 172 —

Corner movement

When two or more vectors are produced at the end of a block, the tool moves linearly from one vector to another. This movement is called the corner movement.

If these vectors almost coincide with each other, the corner movement isn't performed and the latter vector is ignored.



If $\Delta Vx \leq \Delta V$ limit and $\Delta Vy \leq \Delta V$ limit, the latter vector is ignored. The ΔV limit is set in advance by parameter (No. 557).

If these vectors do not coincide, a move is generated to turn around the corner. This move belongs to the latter block.



However, if the path of the next block is semicircular or more, the above function is not performed.

The reason for this is as follows:



If the vector is not ignored, the tool path is as follows: $P_1 \rightarrow P_2 \rightarrow P_3 \rightarrow (Circle) \rightarrow P_4 \rightarrow P_5 \rightarrow P_6$

But if the distance between P2 and P3 is negligible, the point P3 is ignored. Therefore, the tool path is as follows:

$$P_2 \rightarrow P_4$$

Namely, circle cutting by the block N6 is ignored.

• Interruption of manual For manual operation during the cutter compensation, refer to Section III–3.5, "Manual Absolute ON and OFF."

14.3.4 Tool Movement in Offset Mode Cancel

Explanations

 Tool movement around an inside corner (180° ≤ α)



• Tool movement around an outside corner at an 016#2). obtuse angle Linear→Linear (90°≦α<180°) G40 α Workpiece Programmed path **—** L Š Tool center path Туре A Circular→Linear G40 α Workpiece Programmed path Tool center path Linear→Linear G4(α Workpiece Programmed path Tool center path Θ L Intersection š Type B Circular→Linear G40 α Workpiece С Lintersection 1 Programmed path Tool center path

Tool path has two types, A and B; and they are selected by parameter (No.

 Tool movement around Tool path has two types, A and B : and they are selected by parameter (No. an outside corner at an 016#2) acute angle Linear→Linear **(α<90°)** G40 Workpiece G42 Programmed path ──── Tool center path L Š Туре Circular→Linear А G40 r G42 Work-S С piece Ф Tool center path Programmed path Linear→Linear G4 Workpiece Programmed path Tool center path G L L ้ร Type B Circular→Linear Work-S piece d Tool center path Programmed path

- Tool movement around the outside linear→linear at an acute angle less than 1 degree (α<1°)
- A block without tool movement specified together with offset cancel



When a block without tool movement is commanded together with an offset cancel, a vector whose length is equal to the offset value is produced in a normal direction to tool motion in the earlier block, the vector is cancelled in the next move command.



Block containing G40 and I_J_K_

The previous block contains G41 or G42

If a G41 or G42 block precedes a block in which G40 and $I_{,J_{,K_{}}}$ are specified, the system assumes that the path is programmed as a path from the end position determined by the former block to a vector determined by (I,J), (I,K), or (J,K). The direction of compensation in the former block is inherited.



In this case, note that the CNC obtains an intersection of the tool path irrespective of whether inner or outer side machining is specified



When an intersection is not obtainable, the tool comes to the normal position to the previous block at the end of the previous block.



— 179 —

The length of the tool center path larger than the circumference of a circle In the example shown below, the tool does not trace the circle more than once. It moves along the arc from P1 to P2. The interference check function described in 14.6.5 may raise an alarm.



To make the tool trace a circle more than once, program two or more arcs.

14.3.5 Interference Check

Tool overcutting is called interference. The interference check function checks for tool overcutting in advance. However, all interference cannot be checked by this function. The interference check is performed even if overcutting does not occur.

Explanations

- Criteria for detecting interference
- (1) The direction of the tool path is different from that of the programmed path (from 90 degrees to 270 degrees between these paths).



(2) In addition to the condition (1), the angle between the start point and end point on the tool center path is quite different from that between the start point and end point on the programmed path in circular machining(more than 180 degrees).



In the above example, the arc in block N6 is placed in the one quadrant. But after cutter compensation, the arc is placed in the four quadrants.

— 182 —

 Correction of interference in advance (1) Removal of the vector causing the interference

When cutter compensation is performed for blocks A, B and C and vectors V_1 , V_2 , V_3 and V_4 between blocks A and B, and V_5 , V_6 , V_7 and V_8 between B and C are produced, the nearest vectors are checked first. If interference occurs, they are ignored. But if the vectors to be ignored due to interference are the last vectors at the corner, they cannot be ignored.

Check between vectors V_4 and V_5

Interference — V_4 and V_5 are ignored.

Check between V_3 and V_6

Interference — V_3 and V_6 are ignored

Check between V_2 and V_7

Interference — V_2 and V_7 are Ignored

Check between V_1 and V_8

Interference — V_1 and V_8 are cannot be ignored

If while checking, a vector without interference is detected, subsequent vectors are not checked. If block B is a circular movement, a linear movement is produced if the vectors are interfered.

(Example 1) The tool moves linearly from V_1 to V_8







(2) If the interference occurs after correction (1), the tool is stopped with an alarm.

If the interference occurs after correction (1) or if there are only one pair of vectors from the beginning of checking and the vectors interfere, the alarm (No.41) is displayed and the tool is stopped immediately after execution of the preceding block. If the block is executed by the single block operation, the tool is stopped at the end of the block.



After ignoring vectors V_2 and V_5 because of interference, interference also occurs between vectors V_1 and V_6 . The alarm is displayed and the tool is stopped.

— 184 —

 When interference is assumed although actual interference does not occur

(1)**D**epression which is smaller than the cutter compensation value



There is no actual interference, but since the direction programmed in block B is opposite to that of the path after cutter compensation the tool stops and an alarm is displayed.





In the same way as 1, the programmed path will be opposite to the tool path after cutter compensation has been applied. This state is determined to constitute interference. An alarm is issued and the machine stops.

14.3.6 Overcutting by Cutter Compensation

Explanations

 Machining an inside corner at a radius smaller than the cutter radius When the radius of a corner is smaller than the cutter radius, because the inner offsetting of the cutter will result in overcuttings, an alarm is displayed and the CNC stops at the start of the block. In single block operation, the overcutting is generated because the tool is stopped after the block execution.



 Machining a groove smaller than the tool radius Since the cutter compensation forces the path of the center of the tool to move in the reverse of the programmed direction, overcutting will result. In this case an alarm is displayed and the CNC stops at the start of the block.



— 186 —

Machining a step smaller than the tool radius

When machining of the step is commanded by circular machining in the case of a program containing a step smaller than the tool radius, the path of the center of tool with the ordinary offset becomes reverse to the programmed direction. In this case, the first vector is ignored, and the tool moves linearly to the second vector position. The single block operation is stopped at this point. If the machining is not in the single block mode, the cycle operation is continued. If the step is of linear, no alarm will be generated and cut correctly. However uncut part will remain.



Starting compensation and cutting along the Z-axis

It is usually used such a method that the tool is moved along the Z axis after the cutter compensation is effected at some distance from the workpiece at the start of the machining.

In the case above, if it is desired to divide the motion along the Z axis into rapid traverse and cutting feed, follow the procedure below.



In the program example above, when executing block N1, blocks N3 and N6 are also entered into the buffer storage, and by the relationship among them the correct compensation is performed as in the figure above.

Then, if the block N3 (move command in Z axis) is divided as follows: As there are two move command blocks not included in the selected plane and the block N6 cannot be entered into the buffer storage, the tool center path is calculated by the information of N1 in the figure above. That is, the offset vector is not calculated in start–up and the overcutting may result.





The move command in the same direction as that of the move command after the motion in Z axis should be programmed.



As the block with sequence No. N2 has the move command in the same direction as that of the block with sequence No. N6, the correct compensation is performed.

14.3.7 Input Command from MDI

Cutter compensation C is not performed for commands input from the MDI.

However, when automatic operation by absolute commands is temporarily stopped by the single block function, MDI operation is performed, then automatic operation starts again, the tool path is as follows :

In this case, the vectors at the start position of the next block are translated and the other vectors are produced by the next two blocks. Therefore, from next block but one, cutter compensation C is accurately performed.



When position P_A , P_B , and P_C are programmed in an absolute command, tool is stopped by the single block function after executing the block from P_A to P_B and the tool is moved by MDI operation. Vectors V_{B1} and V_{B2} are translated to V_{B1} ' and V_{B2} ' and offset vectors are recalculated for the vectors V_{C1} and V_{C2} between block P_B-P_C and P_C-P_D .

However, since vector V_{B2} is not calculated again, compensation is accurately performed from position P_C .

14.4 TOOL COMPENSATION VALUES, NUMBER OF COMPENSATION VALUES, AND ENTERING VALUES FROM THE PROGRAM (G10)



Fig14.4 (a) Tool compensation

Tool compensation values can be entered into CNC memory from the CRT/MDI panel (see III–11.4.1) or from a program.

A tool compensation value is selected from the CNC memory when the corresponding code is specified after address H or D in a program. The value is used for tool length compensation, cutter compensation, or the tool offset.

Table 14.4(a) shows the valid input range of tool compensation values.

 Valid range of tool compensation values

Explanations

Table14.4 (b) The valid input range of tool compensation value

| Incroment system | Tool compensation value | | |
|------------------|-------------------------|--------------------|--|
| increment system | Metric input | Inch input | |
| IS–B | \pm 999.999 mm | \pm 99.9999inch | |
| IS–C | \pm 999.9999 mm | ±99.99999 inch | |

 Number of tool compensation values and the addresses to be specified The memory can hold 32 or 64 tool compensation values (option). Address D or H is used in the program. The address used depends on which of the following functions is used: tool length compensation(see 14.1) or cutter compensation C (see 14.2).

The range of the number that comes after the address (D or H) depens on the number of tool compensation values : 0 to 32 (0–GSD), 0 to 64 (0–MD).

• Tool compensation value to be entered

The tool compensation memory determines the tool compensation values that are entered (set) (Table 14.4 (b)).

Table14.4 (b) Setting contents tool compensation memory and tool compensation value

| Tool compensation memory | Tool compensation value | |
|--------------------------|--|--|
| A | The tool compensation corresponding to the tool compensation number is used. | |

Format

The programming format depends on which tool compensation memory is used.

• Input of tool compensation value by programing

Table14.4(c) Setting format of Tool compensation memory and Tool compensation value

| Tool compensation memory | | Format |
|--------------------------|-------------------------|----------|
| Α | Tool compensation value | G10P_R_; |

P: Number of tool compensation

R : Tool compensation value in the absolute command(G90) mode Value to be added to the specified tool compensation value in the incremental command(G91) mode (the sum is also a tool compensation value.)

NOTE

To provide compatibility with the format of older CNC programs, the system allows L1 to be specified instead of L11.

14.5 NORMAL DIRECTION CONTROL (G150, G151, G152) (FOR 0–GSD ONLY)

When a tool with a rotation axis (fourth–axis) is moved in the XY plane during cutting, the normal direction control function can control the tool so that the fourth–axis is always perpendicular to the tool path (Fig. 14.5 (a)).



Fig14.5 (a) Sample Movement of the tool

Format

| G code | Function | Explanation |
|--------|-----------------------------------|---|
| G151 | Normal direction control left | If the workpiece is to the right of the tool path looking toward the direction in which the tool advances, the normal direction |
| G152 | Normal direction control right | control left (G151) function is specified After G151 or G152 is specified, the no mal direction control function is enable |
| G150 | Normal direction control cancel | (normal direction control mode). When G150 is specified, the normal direc- tion control mode is canceled. |







Explanations

• Angle of the fourth axis

When viewed from the center of rotation around the fourth-axis, the angular displacement about the fourth-axis is determined as shown in Fig. 14.5 (d). The positive side of the X-axis is assumed to be 0, the positive side of the Y-axis is 90, the negative side of the X-axis is 180, and the negative side of the Y-axis is 270.



Fig14.5 (d) Angle of the fourth axis

When the cancel mode is switched to the normal direction control mode, the fourth–axis becomes perpendicular to the tool path at the beginning of the block containing G151 or G152.

In the interface between blocks in the normal direction control mode, a command to move the tool is automatically inserted so that the fourth–axis becomes perpendicular to the tool path at the beginning of each block. The tool is first oriented so that the fourth–axis becomes perpendicular to the tool path specified by the move command, then it is moved along the X– and Y axes.

In the cutter compensation mode, the tool is oriented so that the fourth-axis becomes perpendicular to the tool path created after compensation.

In single–block operation, the tool is not stopped between a command for rotation of the tool and a command for movement along the X– and Y–axes. A single–block stop always occurs after the tool is moved along the X– and Y–axes.

Normal direction control



Fig14.5(e) Point at which a Single–Block Stop Occurs in the Normal Direction Control Mode

Before circular interpolation is started, the fourth–axis is rotated so that the fourth–axis becomes normal to the arc at the start point. During circular interpolation, the tool is controlled so that the fourth–axis is always perpendicular to the tool path determined by circular interpolation.



Fig14.5 (f) Normal direction control of the circular interpolation

CAUTION

During normal direction control, the fourth axis always rotates through an angle less than 180 deg. I.e., it rotates in whichever direction provides the shorter route.

— 194 —

fourth axis feedrate

Movement of the tool inserted at the beginning of each block is executed at the feedrate set in parameter 683. If dry run mode is on at that time, the dry run feedrate is applied. If the tool is to be moved along the X–and Y–axes in rapid traverse (G00) mode, the rapid traverse feedrate is applied.

The federate of the fourth axis during circular interpolation is defined by the following formula.

- F× Amount of movement of the revolution (deg) Length of arc (mm or inch) (deg/min)
- F : Federate (mm/min or inch/min) specified by the corresponding block of the arc
- Amount of movement of the revolution :

The difference in angles at the beginning and the end of the block.

NOTE

1 Do not specify any command to the fourth-axis during normal direction control.

Any command specified at this time is ignored.

- 2 Before processing starts, it is necessary to correlate the workpiece coordinate of the fourth axis with the actual position of the fourth axis on the machine using the coordinate system setting (G92) or the like.
- 3 Normal direction control cannot be performed by the G53 move command.

15 CUSTOM MACRO A

A function covering a group of instructions is stored in memory as same as a subprogram. The stored function is presented by one instruction, so that only the representative instruction need be specified to execute the function. This group of registered instructions is called a "custom macro body" and the representative instruction is called a "custom macro instruction". The custom macro body may simply be called a macro. And the custom macro instruction may be called a macro call command.



Programmers need only remember representative macro instructions without having to remember all the instructions in a custom macro body. The three most significant points on custom macros are that variables can be used in the custom macro body, operations can be performed on variables and actual values can be assigned to the variables in custom macro instructions.

NOTE

Machine tool builders are requested to attach your custom macro program tape or program list to the CNC unit without fail. If it is necessary to replace part program storage memory due to a failure, FANUC serviceman or end users operators in charge of maintenance should know the contents of custom macro for the purpose of repairing the trouble immediately.

— 196 —

body.

15.1 CUSTOM MACRO COMMAND

15.1.1 M98 (Single call)

Command format is as follows :

Format

Command format is as follows :



The custom macro command is the command to call the custom macro

With the above command, the macro body specified by P is called.

| 15.1.2 Subprogram call using M code | The subprogram can be called using M code set in parameter. $N_G_X M98P $; instead of commanding as above, the same operation can be commanded using following command : $N_G_X M < m>$; The correspondence of M code $< m>$ which calls subprogram and the program number (O9001 to O9003) of the called subprogram shall be set by parameters (No. 0240 to No. 0242). For subprogram call, a maximum of 3 among M03 to M255, except M30 and M code which does not buffer (parameter No. 111, 112) can be used. | |
|---|---|--|
| | CAUTION Similarly to M98, signal MF and M code are not output. Delivery of argument is not possible. Subprogram call M code used in the subprogram which is called by M or T code does not executes subprogram | |

call but as an ordinal M code.

15.1.3 Subprogram call using T code

When parameter (No. 0040 #5) is set beforehand, subprogram (O9000) can be called using T code.

N_G_X_ -- T <t> ;

the above command results in the same operation of command of the following 2 blocks.

#149 = <t> ;

N_G_X_ -- M98 P9000 ;

The T code t___is stored in a common variable #0149 as an argument.

CAUTION

- 1 It is not possible to command with a same block as that of subprogram call using M code.
- 2 Subprogram call T code used in the subprogram which is called by M or T code does not executes subprogram call but as an original T code.

15.1.4 G66 (Modal call)

Format



The above command selects the macro modal call mode for NC. In other words, every time each block subsequent to the above command is executed, the macro designated by P is called. Also, an argument can be designated by a block subsequent to the above command. For this argument, refer to 15.1.5. The macro modal call mode is cancelled by the command below.

G67;

CAUTION

- 1 Blocks containing the G66 and G67 commands do not cqll a macro.
- 2 In MDI mode, the G66 command specifies macro modal calling mode. The G67 command cancels macro modal calling mode. Other commands, however, do not call a macro, even when executed between G66 and G67. Instead, they are executed as ordinary commands. If MDI operation B is specified, macros can be called in MDI mode.
- 3 Commands other than O, N, and P are ignored in the G66 and G67 blocks.
- 4 A repetition count cannot be specified for macro modal calling. Only the four low–order digits of the values specified with P in a G66 block are valid.
- 5 The maximum nesting level for macro modal calling is 1. The maximum nesting level for subprogram calling is 4. However, the maximum total nesting level for macro modal calling and subprogram calling is 4.

15.1.5An argument means an actual val
called macro. An argument can be
except for O. The format of argu

An argument means an actual value given to a variable employed in a called macro. An argument can be specified at all employable addresses except for O. The format of argument specification is the same as in normal CNC command. The limitation at each address as a normal CNC command, such as decimal point, sign, maximum number of digits, etc., is also applicable to the format.

The following table indicates the correspondence between argument specification addresses and variable numbers.

| Variable number (value) | Variable number (flag) | Address | Remarks |
|----------------------------|---------------------------|---------|---------|
| #8004 | #8104 | I | |
| #8005 | #8105 | J | |
| #8006 | #8106 | к | |
| #8009 | #8109 | F | |
| #8010 | #8110 | G | |
| #8011 | #8111 | н | |
| #8013 | #8113 | М | |
| #8014 | #8114 | Ν | |
| #8016 | #8116 | Р | |
| #8017 | #8117 | Q | |
| #8018 | #8118 | R | |
| #8019 | #8119 | S | |
| #8020 | #8120 | Т | |
| #8024 | #8124 | х | |
| #8025 | #8125 | Y | |
| #8026 | #8126 | Z | |

| Table 15.1.5 (a) | Correspondence between addresses |
|------------------|----------------------------------|
| | and variable numbers |
#8100's variables are flags to indicate whether an argument has been specified or not every call. These variables are 1 if an argument is specified, and 0 if no argument is specified.

#8000's variables show a specified value, if an argument is specified. However, they become as specified below, if no argument is specified. (a) Reference in CNC command

The address itself is neglected.

(b) Reference in macro command and branch command (Refer to 15.2.3). This value is undefined. Use it after confirming #8100's numbers.

 Table 15.1.5 (b)
 Correspondence between G codes of the argument specification and variable numbers

| Variable number (value) | Variable number (flag) | G code group number | G codes of the argument specification |
|-------------------------------|------------------------------|---------------------------|---------------------------------------|
| #8030 | #8130 | 00 | One shot and others |
| #8031 | #8131 | 01 | G00, G01, G02, G03 |
| #8032 | #8132 | 02 | G17, G18, G19 |
| #8033 | #8133 | 03 | G90, G91 |
| #8035 | #8135 | 05 | G94 |
| #8036 | #8136 | 06 | G20, G21 |
| #8037 | #8137 | 07 | G40, G41, G42 |
| #8038 | #8138 | 08 | G43, G44, G49 |
| #8039 | #8139 | 09 | G73, G74, G76, G80 to G89 |
| #8040 | #8140 | 10 | G98, G99 |
| #8041 | #8141 | 11 | G50, G51 |
| #8042 | #8142 | 12 | G66, G67 |
| #8045 | #8145 | 15 | G61, G62, G63, G64 |
| #8046 | #8146 | 16 | G68, G69 |

If plural G codes are specified a arguments in the same block, a value is input to variables in each group as shown in Table 15.7.2 (b). In this case, #8010 is the smallest group number out of the numbers specified at a time. No value can institute into #8100's and 8000's variables.

15.2 CUSTOM MACRO BODY

15.2.1

Variables

In the custom macro body, the CNC command, which uses ordinary CNC command variables, calculation, and branch command can be used. The custom macro body starts from the program No. which immediately follows O and ends at M99.



Fig. 15.2 Construction of the custom macro body

A variable can be specified to make the macro flexible and versatile by applying the calculated variable when calling the macro or when executing the macro itself. Multiple variables are identified from each other by variable numbers.

(1) How to express variables

Variables are expressed by variable numbers following # as shown below.

#i (i = 1, 2, 3, 4) (Example) #5, #109, #1005

(2) How to quote variables

A numeral following an address can be replaced by a variable. Assume that $\langle Address \rangle \#1$ or $\langle Address \rangle - \#1$ is programmed, and it means that the variable value or its complement serves as the command value of the address.

(Example)

F#103 F15 was commanded when #103=15

Z-#110 Z-250 was commanded when #110=250

G#130 G3 was commanded when #103=3.

When replacing a variable number with a variable, it is not expressed as "##100", for example, but express as "#9100". That is, "9" next to "#" indicates the substitute of the variable number, while the lower number to be replaced.

(Example)

If #100=105 and #105=-500, "X#9100" indicates that X-500 was commanded, and "X-#9100" indicates that X500 was commanded.

NOTE

- 1 No variable can be quoted at address O and N. Neither O#100 nor N#120 can be programmed.
- 2 It is not possible to command a value exceeding the maximum command value set in each address.When #30=120, G#30 has exceeded the maximum

command value.

| 15.2.2 Kind of Variables | Variables are sorted into common variables and system variables according to variable numbers, and their applications and characters differ from each other. | |
|-----------------------------|---|--|
| | (1) Common variable #100 to #149 and #500 to #531 Common variables are common to main programs and each macro called from these main programs. That is, #i in a macro is equal to #i in another macro. Common variables #100 to #149 are cleared when the power is turned off, and reset to "0" just after power was turned on. Common variables #500 to #531 are not cleared, even if power is turned off, and their values remain unchanged. | |
| | (2) System variable The system variables are defined as variables whose applications remain fixed. | |
| | (a) Tool offset amount #1 to #99, #2000 to #2200 The offset amount can be known by reading system variable #1 to #99 values for tool offset amounts, and these values can be changed by substituting them into system variables #1 to #99. Among these offset numbers 1 to 99, those which are not used as offset amounts can be treated as hold type common variables (#500 to #531). The system variables #2001 to #2200 correspond to the tool offset numbers 1 to 2000. They can be read and substituted as the same as #1 to #99. #2000 always can read 0. | |

- (b) Interface input signals #1000 to #1015, #1032
 - Interface signals can be known, by reading system variables #1000 to #1015 for reading interface signals.

2¹⁵ 2¹⁴ 2¹³ 2¹² 2¹¹ 2¹⁰ 2⁹ 2⁶ 2⁵ 2⁸ 27 24 2³ 2² 2¹ 2⁰ UI15 UI14 UI13 UI12 UI11 UI10 UI9 UI8 UI7 UI6 UI5 UI4 UI3 UI2 UI1 UI0 #1004 #1014 #1012 #1010 #1008 #1006 #1002 #1000 #1015 #1013 #1011 #1009 #1007 #1005 #1003 #1001

| Input signal | Variable value |
|----------------|----------------|
| Contact closed | 1 |
| Contact opened | 0 |

By reading system variable #1032, all the input signals can be read at once.

$$\#1032 = \sum_{i=0}^{15} \#(1000 + i) \times 2^{i}$$

NOTE

- 1 No value can be substituted into system variables #1000 to #1032.
- 2 System variables #1000 to #1015 can be displayed by diagnostic function.
 DGNOS No.110 U10 to U17
 DGNOS No.111 U18 to U15
- 3 System variables #1000 to #1032 can be used only when PMC is combined.

(c) Interface output signals #1100 to #1115, #1132, #1133
 A value can be substituted into system variables #1100 to #1115
 for sending the interface signals.

2¹⁵ 2¹⁴ 2¹³ 2¹² 2¹¹ 2¹⁰ 2⁹ 2⁸ 2⁷ 2⁶ 2⁵ 2^4 2^3 2^2 2^1 2^0 UO15 UO14 UO13 UO12UO11 UO10 UO9 UO8 UO7 UO6 UO5 UO4 UO3 UO2 UO1 UO0 #1114 #1112 #1110 #1108 #1106 #1104 #1102 #1100 #1103 #1113 #1105 #1101 #1115 #1111 #1109 #1107 Output signal Variable value

| Contact closed | 1 |
|----------------|---|
| Contact open | 0 |

By substituting a value into system variable #1132, all output signals (UO0 to UO15) can be sent out at once.

$$\#1132 = \sum_{i=0}^{15} \#(1100 + i) \times 2^{i}$$

32 interface signals (UO100 to UO131) can be sent out by #1133 at once.

CAUTION

If any other number than '0' or '1' is substituted into system variables #1100 to #1115, it is treated as '1'.

NOTE

- 1 It is possible to read the values of system variables #1100 to #1133.
- 2 System variables #1100 to #1115 and #1133 can be displayed by diagnostic function.
 - DGNOS No.162 UO0 to UO7 No.163 UO8 to UO15 No.196 UO100 to UO107
 - No.197 UO108 to UO115
 - No.198 UO116 to UO123 No.199 UO124 to UO131
- 3 System variables #1100 to #1133 can be used only when PMC is combined.

(d) Number of necessary parts, number of machined parts

| Kind | System variable |
|---------------------------|-----------------|
| Number of machined parts | #3901 |
| Number of necessary parts | #3902 |

NOTE

Do not substitute a negative value.

(e) Modal information #4001 to #4120

It is possible to know the current values of modal information (modal command given till immediately preceding block) by reading values of system variables #4001 to #4120.

| Variables | Modal information |
|--|--|
| #4001 #4002 #4003 | G code (group 01) G code (group 02) G code (group 03) |
| #4022 #4102 #4107 #4109 #4111 #4113 #4114 #4115 #4119 #4120 | G code (group 22) B code D code F code H code M code Sequence No. Program No. S code T code |

NOTE

The unit will be the one being used when the command is given.

 (f) Position information #5001 to #5083
 The position information can be known by reading system variables #5001 to #5083. The unit of position information is 0.001 mm in metric input and 0.0001 inch in inch input.

| System variables | Position information | Reading while moving | Cutter and tool length compensation |
|----------------------------------|---|----------------------------|--|
| #5001 #5002 #5003 #5004 | Block end point position of X axis (ABSIO) Block end point position of Y axis Block end point position of Z axis Block end point position of 4th axis | Possible | Not considered. Position of tool nose (program command posi- tion) |
| #5021 #5022 #5023 #5024 | X axis coordinate position (ABSMT) Y axis coordinate position Z axis coordinate position 4th axis coordinate position | Impossible | Considered. Position of tool reference point (Machine coordi- nate) |
| #5041 #5042 #5043 #5044 | Present position of X axis (ABSOT) Present position of Y axis Present position of Z axis Present position of 4th axis | Impossible | Considered. Position of tool reference point (ABSOLUTE in- dication) |
| #5061 #5062 #5063 #5064 | Skip signal position of X axis (ABSKP) Skip signal position of Y axis Skip signal position of Z axis Skip signal position of 4th axis | Possible | Considered. Position of tool reference point |
| #5080 #5081 #5082 #5083 | Value of cutter compensation Value of tool length compensation (X axis) Value of tool length compensation (Y axis) Value of tool length compensation (Z axis) | Possible | |

NOTE

- 1 It is not possible to substitute any value into system variables #5001 to #5083.
- 2 When the skip signal doesn't turn on at G31, the skip signal position is the end point of that block.

15.2.3 Macro Instructions (G65)

General Form

G65HmP#i Q#j R#k ;

- m : Indicates macro functions at 01 to 99
- #i : Variable name to which arithmetic result is loaded.
- #j: Variable name 1 to be operated. A constant is also acceptable.
- #k : Variable name 2 to be operated. A constant is also acceptable.

Meaning $#1 = #j \oplus #k$

— Operator (Specified by Hm)

Example

| P#100 Q#101 R#102 | #100=#101⊕#102 |
|--------------------|-----------------|
| P#100 Q#101 R15 | #100=#101⊕15 |
| P#100 Q-100 R#102 | #100=−100⊕#102 |
| P#100 Q120 R-50 | #100=120⊕–50 |
| P#100 Q-#101 R#102 | #100=-#101⊕#012 |
| | |

CAUTION

 No decimal point can be put to variable values. Therefore, the meaning of each value is the same as that designated without decimal point when quoted in each address.
 (Example) #100 = 10

X#100 -- 0.01 mm (metric input)

2 Those indicating an angle must be expressed by degree, and the least input increment is 1/1000 degree.
 (Example) 100 ---- 0.1°

NOTE

H code specified by G65 does not affect any selection of offset amount.

| G code | H code | Function | Definition |
|--------|--------|----------------------------------|---|
| G65 | H01 | Definition, substitution | #i = #j |
| G65 | H02 | Addition | #i = #j + #k |
| G65 | H03 | Subtraction | #i = #j - #k |
| G65 | H04 | Product | $\#i = \#j \times \#k$ |
| G65 | H05 | Division | #i = #j ÷ #k |
| G65 | H11 | Logical sum | #i = #j. OR. #k |
| G65 | H12 | Logical product | #i = #j. AND. #k |
| G65 | H13 | Exclusive OR | #i = #j. XOR. #k |
| G65 | H21 | Square root | $\#i = \sqrt{\#j}$ |
| G65 | H22 | Absolute value | #i = #j |
| G65 | H23 | Remainder | $\#i = \#j - trunc (\#j / \#k) \times \#k (trunc : Discard fractions less than 1)$ |
| G65 | H24 | Conversion from BCD to binary | #i = BIN (#j) |
| G65 | H25 | Conversion from binary to BCD | #i = BCD (#j) |
| G65 | H26 | Combined multiplication/division | #i = (#i ×#j) ÷ #k |
| G65 | H27 | Combined square root 1 | $\#i = \sqrt{\#J^2 + \#K^2}$ |
| G65 | H28 | Combined square root 2 | $\#i = \sqrt{\#J^2 - \#K^2}$ |
| G65 | H31 | Sine | #i = #j ⋅ SIN (#k) |
| G65 | H32 | Cosine | #i = #j · COS (#k) |
| G65 | H33 | Tangent | #i = #j · TAN (#k) |
| G65 | H34 | Arctangent | #i = ATAN (#j / #k) |
| G65 | H80 | Unconditional divergence | GOTOn |
| G65 | H81 | Conditional divergence 1 | IF#j = #k, GOTOn |
| G65 | H82 | Conditional divergence 2 | IF#j ≠ #k, GOTOn |
| G65 | H83 | Conditional divergence 3 | IF#j > #k, GOTOn |
| G65 | H84 | Conditional divergence 4 | IF#j < #k, GOTOn |
| G65 | H85 | Conditional divergence 5 | IF#j ≧ #k, GOTOn |
| G65 | H86 | Conditional divergence 6 | IF#j ≦ #k, GOTOn |
| G65 | H99 | P/S alarm occurrence | P/S alarm number 500 +n occurrence |

Table 15.2.3

| Variable arithmetic command | |
|---------------------------------|--|
| (| (a) Definition and substitution of variable $\#i = \#j$ G65 H01 P#i Q#j; [Example] G65 H01 P#101 Q1055; (#101=1005) G65 H01 P#101 Q#110; (#101=#110) G65 H01 P#101 Q-#112; (#101=-#112) |
| (| (b) Addition #i = #j + #k G65 H02 P#i Q#j R#k; [Example] G65 H02 P#101 Q#102 R15 ; (#101=#102+15) |
| (| (c) Subtraction #i = #j - #k G65 H03 P#i Q#j R#k; [Example] G65 H03 P#101 Q#102 R#103 ; (#101=#102-#103) |
| (| (d) Product #i = #j ×#k G65 H04 P#i Q#j R#k; [Example] G65 H04 P#101 Q#102 R#103 ; (#101=#102×#103) |
| (| (e) Division #i = #j ÷ #k G65 H05 P#i Q#j R#k; [Example] G65 H05 P#101 Q102 R#103 ; (#101=#102÷#103) |
| (| (f) Logical sum #i = #j.OR.#k G65 H11 P#i Q#j R#k; [Example] G65 H11 P#101 Q102 R#103 ; (#101=#102.OR.#103) |
| (| (g) Logical product #i = #j.AND.#k G65 H12 P#i Q#j R#k ; [Example] G65 H12 P#101 Q#102 R#103 ; (#101=#102.AND.#103) |
| (| (h) Exclusive OR #i = #j.XOR.#k G65 H13 P#i Q#j R#k ; [Example] G65 H13 P#101 Q#102 R#103 ; (#101=#102.XOR.#103) |
| (| (i) Square root #i = $\sqrt{\#j}$ G65 H21 P#i Q#j ; [Example] G65 H21 P#101 Q#102 ; (#101= $\sqrt{\#102}$) |
| (| (j) Absolute value #i = #j G65 H22 P#i Q#j; [Example] G65 H22 P#101 Q#102; (#101= #102) |
| (| (k) Remainder #i = #j - trunc (#j/#k)×#k trunc : Discard fractions less than 1 G65 H23 P#i Q#j R#k ; [Example] G65 H23 P#101 Q#102 R#103 ; (#101=#102-trunc (#102/#103)×#103) |
| (| Conversion from BCD to binary #i = BIN (#j) G65 H24 P#i Q#j; [Example] G65 H24 P#101 Q#102 ; (#101=BIN (#102)) |
| (| (m) Conversion from binary to BCD #i = BCD (#j) G65 H25 P#i Q#j ; [Example] G65 H25 P#101 Q#102 ; (#101=BCD (#102)) |
| (| (n) (n) Combined multiplication/division $\#i = (\#1 \times \#j) \div \#k$ G65 H26 P#i Q#j R#k ; [Example] G65 H26 P#101 Q#102 R#103 ; $(\#101=(\#101 \times \#102)(\div \#103)$ |

- (o) Combined square root 1 $\#i = \sqrt{\#j^2 + \#k^2}$ G65 H27 P#i Q#j R#k ; [Example] G65 H27 P#101 Q#102 R#103 ; $(\#101 = \sqrt{\#102^2 + \#103^2})$
- (p) Combined square root 2 #i = $\sqrt{#j^2 #k^2}$ G65 H28 P#i Q#j R#k ; [Example] G65 H28 P#101 Q#102 R#103 ; (#101= $\sqrt{#102^2 - #103^2}$)
- (q) Sine #i = #j·SIN (#k) (degree unit) G65 H31 P#i Q#j R#k ; [Example] G65 H31 P#101 Q#102 R#103 ; (#101=#102·SIN(#103)
- (r) Cosine #i = $\#j \cdot COS$ (#k) (degree unit) G65 H32 P#i Q#j R#k ; [Example] G65 H32 P#101 Q#102 R#103 ; (#101=#102 \cdot COS (#103)
- (s) (s) Tangent #i = #j·TAN (#k) (degree unit) G65 H33 P#i Q#j R#k ; [Example] G65 H33 P#101 Q#102 R#103 ; (#101=#102·TAN (#103)
- (t) (t) Arctangent #i = ATAN (#j/#k) (degree unit) G65 H34 P#i Q#j R#k ; $(0^{\circ} \le \#i < 360^{\circ})$ [Example] G65 H34 P#101 Q#102 R#103 ; (#101=ATAN (#102 / #103))

CAUTION

Angle in (g) to (t) must be indicated by degree and the least input increment is 1/1000 degree.

NOTE

- 1 If either Q or R necessary for each arithmetic operation was not indicated, its value is calculated as '0'.
- 2 All figures below decimal point are truncated if each arithmetic result includes decimal point.

— 211 —

- Divergence command
- (a) Unconditional branch
 G65 H80 Pn ; n : Sequence number
 [Example] G65 H80 P120 ; (Diverge to N120)
- (b) Conditional divergence 1 #j. EQ. #k (=) G65 H81 Pn Q#j R#k ; n : Sequence number [Example] G65 H81 P1000 Q#101 R#102 ; #101=#102, go to N1000 #101 \neq #102, go to next
- (c) Conditional divergence 2 #j. NE. #k (\neq) G65 H82 Pn Q#j R#k ; n : Sequence number [Example] G65 H82 P1000 Q#101 R#102 ; #101 \neq #102, go to N1000 #101=#102, go to
- (d) (Conditional divergence 3 #j. GT. #k (>) G65 H83 Pn Q#j R#k ; n : Sequence number [Example] G65 H83 P1000 Q#101 R#102 ; #101 > #102, go to N1000 #101 \leq #102, go to next
- (e) Conditional divergence 4 #j. LT. #k (<) G65 H84 Pn Q#j R#k ; n : Sequence number [Example] G65 H84 P1000 Q#101 R#102 ; #101 < #102, go to N1000 #101 \ge #102, go to next
- (f) Conditional divergence 5 #j. GE. #k (\geq) G65 H85 Pn Q#j R#k ; n : Sequence number [Example] G65 H85 P1000 Q#101 R#102 ; #101 \geq #102, go to N1000 #101 < #102, go to next
- (g) Conditional divergence 6 #j. LE. #k (\leq) G65 H86 Pn Q#j R#k ; n : Sequence number [Example] G65 H86 P1000 Q#101 R#102 ; #101 \leq #102, go to N1000 #101 > #102, go to next
- (h) P/S alarm occurrence
 G65 H99 Pn ; Alarm No. : 500+n
 [Example] G65 H99 P15 ; P/S alarm 515 occurrence

NOTE

- 1 If positive numbers were designated as sequence numbers at branch designations, they are searched forward first and then, backward. If negative numbers were designated, they are searched backward first and then, forward.
- 2. Sequence number can also be designated by variables. (Example) G65 H81 P#100 Q#101 R#102 ; When conditions are satisfied, processing branches to the block having the sequence number designated with #100.

15.2.4 Warning and Notes on Custom Macro

WARNING

Since an integer only is employable as the variable value, in case the operation results with decimal numbers, the figures below decimal point truncated, if an arithmetic result contains a fraction part.

Particularly be careful with the arithmetic sequence, accordingly.

[Example]

When #100=35, #101=10, #102=5, the following results.

#110=#100 ÷ #101 (=3)

#111=#110×#102 (=15) #120=#100×#102 (=175)

#121=#120 ÷ #101 (=17)

#111=15 and #121=17

NOTE

- 1 How to input "#" For standard MDI key, when " /# EOB" key is depressed after address G, X, Y, Z, R, I, J, K, F, H, M, S, T, or P, # code is input
- 2 It is also possible to give a macro instruction in the MDI mode. However address data other than G65 are not displayed by keying operation.
- 3 Address H, P, Q and R of macro instruction must always be written after G65. Address O and N only are writable before G65.

H02 G65 P#100 Q#101 R#102 ; ---- Error N100 G65 H01 P#100 Q10 ;----- Correct

- 4 Single block Generally, the macro instruction block does not stop even if single block stop is turned on. However, by setting parameter No. 0011#5, it is possible to make single block effective. This is used for macro testing.
- 5 Variable values can be taken within a range of -2³¹ to 2³¹ -1, but they are not displayed correctly, except for -99999999 to 99999999. If they exceed the above range, they are displayed as *******.
- 6 It is possible to nest subprograms up to four times.
- 7 When a custom macro is loaded from a paper tape in the EIA code, '&' code is treated as '#', because there is no '#' code in the EIA code.

16 CUSTOM MACRO B

Although subprograms are useful for repeating the same operation, the custom macro function also allows use of variables, arithmetic and logic operations, and conditional branches for easy development of general programs such as pocketing and user–defined canned cycles. A machining program can call a custom macro with a simple command, just like a subprogram.



| 16.1 VARIABLES | An ordinary machining program specifies a G code and the travel distance directly with a numeric value; examples are G100 and X100.0. With a custom macro, numeric values can be specified directly or using a variable number. When a variable number is used, the variable value can be changed by a program or using operations on the MDI panel. | | |
|---|--|--|--|
| | #1=#2+100; | | |
| | G01 X#1 F300 ; | | |
| Explanation | | | |
| Variable representation | When specifying a variable, specify a number sign (#) followed by a variable number. Personal computers allow a name to be assigned to a variable, but this capability is not available for custom macros. Example: #1 | | |
| | An expression can be used to specify a variable number. In such a case, the expression must be enclosed in brackets. | | |
| | Example: #[#1+#2–12] | | |
| Range of variable values | Local and common variables can have value 0 or a value in the following ranges : -10^{47} to -10^{-29} 10^{-29} to 10^{47} | | |
| | If the result of calculation turns out to be invalid, an alarm No. 111 is issued. | | |
| Omission of the decimal point | When a variable value is defined in a program, the decimal point can be omitted. | | |
| | Example: When #1=123; is defined, the actual value of variable #1 is 123.000. | | |
| Undefined variable | When the value of a variable is not defined, such a variable is referred to as a "null" variable. Variable #0 is always a null variable. It cannot be written to, but it can be read. | | |

• Types of variables

Variables are classified into four types by variable number.

Table 16.1 Types of variables

| Variable number | Type of variable | Function |
|----------------------------|---------------------|--|
| #0 | Always null | This variable is always null. No value can be assigned to this variable. |
| #1 – #33 | Local variables | Local variables 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. |
| #100 – #149 #500 – #531 | Common variables | Common variables can be shared among different macro programs. When the power is turned off, variables #100 to #149 are initialized to null. Variables #500 to #531 hold data even when the power is turned off. |
| #1000 - | System variables | System variables are used to read and write a variety of NC data items such as the current position and tool compensation val- ues. |

• Referencing variables

To reference the value of a variable in a program, specify a word address followed by the variable number. When an expression is used to specify a variable, enclose the expression in brackets.

Example: G01X[#1+#2]F#3;

A referenced variable value is automatically rounded according to the least input increment of the address.

Example:

When G00X#1; is executed on a 1/1000–mm CNC with 12.3456 assigned to variable #1, the actual command is interpreted as G00X12.346;.

To reverse the sign of a referenced variable value, prefix a minus sign (–) to #.

Example: G00X-#1;

When an undefined variable is referenced, the variable is ignored up to an address word.

Example:

When the value of variable #1 is 0, and the value of variable #2 is null, execution of G00X#1Y#2; results in G00X0;.

Displaying variable values

Procedure for displaying variable values

Procedure

- 1 Press the key to display the tool compensation screen.
- 2 Press the soft key [MACRO] to display the macro variable screen.
- 3 After press the $|_{N_0}$ key, enter a variable number, then press



The cursor moves to the position of the entered number.

| | | | | · · · |
|------------|---|---|---|---|
| VARIABLE | | | 01234 N1234 | |
| NO. | DATA | NO. | DATA | |
| 100 | 123.456 | 108 | | |
| 101 | 0.000 | 109 | | |
| 102 | | 110 | | |
| 103 | | 111 | | |
| 104 | | 112 | | |
| 105 | | 113 | | |
| 106 | | 114 | | |
| 107 | | 115 | | |
| ACTUAL POS | SITION (RELATI | VE) | | |
| Х | 0.000 | Y | 0.000 | |
| Z | 0.000 | В | 0.000 | |
| [OFFSET] | [MACRO] [M | ienu] [wor | к][] | |
| | VARIABLE NO. 100 101 102 103 104 105 106 107 ACTUAL POS X Z | VARIABLE NO. DATA 100 123.456 101 0.000 102 103 104 105 106 107 ACTUAL POSITION (RELATI X 0.000 Z 0.000 | VARIABLE NO. DATA NO. 100 123.456 108 101 0.000 109 102 110 103 111 104 112 105 113 106 114 107 115 ACTUAL POSITION (RELATIVE) X 0.000 Y Z 0.000 B [OFFSET] [MACRO] [MENU] [WOR | VARIABLE 01234 N1234 NO. DATA NO. DATA 100 123.456 108 101 0.000 109 102 110 103 111 104 112 105 113 106 114 107 115 ACTUAL POSITION (RELATIVE) X 0.000 Y 0.000 Z 0.000 B 0.000 [OFFSET] [MACRO] [MENU] [WORK] [] |

- When the value of a variable is blank, the variable is null.
- The mark ****** indicates an overflow (when the absolute value of a variable is greater than 99999999) or an underflow (when the absolute value of a variable is less than 0.0000001).

Limitations

Program numbers, sequence numbers, and optional block skip numbers cannot be referenced using variables.

Example:

Variables cannot be used in the following ways: O#1; /#2G00X100.0; N#3Y200.0;

16.2 SYSTEM VARIABLES

System variables can be used to read and write internal NC data such as tool compensation values and current position data. Note, however, that some system variables can only be read. System variables are essential for automation and general–purpose program development.

Explanations

• Interface signals

Signals can be exchanged between the programmable machine controller (PMC) and custom macros.

| Variable number | Function |
|----------------------|--|
| #1000–#1015 #1032 | A 16–bit signal can be sent from the PMC to a custom macro. Variables #1000 to #1015 are used to read a signal bit by bit. Variable #1032 is used to read all 16 bits of a signal at one time. |
| #1100–#1115 #1132 | A 16–bit signal can be sent from a custom macro to the PMC. Variables #1100 to #1115 are used to write a signal bit by bit. Variable #1132 is used to write all 16 bits of a signal at one time. |
| #1133 | Variable #1133 is used to write all 32 bits of a signal at one time from a custom macro to the PMC. Note, that values from –999999999 to +999999999 can be used for #1133. |

Table 16.2(a) System variables for interface signals

For detailed information, refer to the connection manual (B-62543EN-1).

• Tool compensation values

Tool compensation values can be read and written using system variables. Usable variable numbers depend on the number of compensation pairs, and whether a distinction is made between tool length compensation and cutter compensation. When the number of compensation pairs is not greater than 200, variables #2001 to #2400 can also be used.

Table 16.2(b) System variables for tool compensation memory A

| Compensation number | System variable |
|---------------------|-----------------|
| 1 | #10001 (#2001) |
| : | : |
| 200 | #10200 (#2200) |

Macro alarms

| Table 16.2(c) | System | variable fo | r macro alarms |
|---------------|--------|-------------|----------------|
|---------------|--------|-------------|----------------|

| Variable number | Function |
|--------------------|--|
| #3000 | When a value from 0 to 99 is assigned to variable #3000, the NC stops with an alarm. After an expression, an alarm message not longer than 26 characters can be described. The CRT screen displays alarm numbers by adding 500 to the value in variable #3000 along with an alarm message. |

— 218 —

Example:

#3000=1(TOOL NOT FOUND);

 \rightarrow The alarm screen displays "501 TOOL NOT FOUND."

 Automatic operation control The control state of automatic operation can be changed.

Table 16.2(d) System variable (#3003) for automatic operation control

| #3003 | Single block | Completion of an auxiliary function |
|-------|--------------|-------------------------------------|
| 0 | Enabled | To be awaited |
| 1 | Disabled | To be awaited |
| 2 | Enabled | Not to be awaited |
| 3 | Disabled | Not to be awaited |

- When the power is turned on, the value of this variable is 0.
- When single block stop is disabled, single block stop operation is not performed even if the single block switch is set to ON.
- When a wait for the completion of auxiliary functions (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.

| #3004 | Feed hold | Feedrate Override | Exact stop |
|-------|-----------|-------------------|------------|
| 0 | Enabled | Enabled | Enabled |
| 1 | Disabled | Enabled | Enabled |
| 2 | Enabled | Disabled | Enabled |
| 3 | Disabled | Disabled | Enabled |
| 4 | Enabled | Enabled | Disabled |
| 5 | Disabled | Enabled | Disabled |
| 6 | Enabled | Disabled | Disabled |
| 7 | Disabled | Disabled | Disabled |

Table 16.2(e) System variable (#3004) for automatic operation control

- When the power is turned on, the value of this variable is 0.
- When feed hold is disabled:
 - 1. When 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.
 - 2. When the feed hold button is pressed then released, the feed hold lamp comes on, but the machine does not stop; program execution continues and the machine stops at the first block where feed hold is enabled.
- When feedrate override is disabled, an override of 100% is always applied regardless of the setting of the feedrate override switch on the machine operator's panel.

• When exact stop check is disabled, no exact stop check (position check) is made even in blocks including those which do not perform cutting.



Fig. 16.2(a) Example of using variable #3004 in a tapping cycle

Settings

Settings can be read and written. Binary values are converted to decimals.

| | | | # | #3005 | | | | |
|---|-----|-----|-----|--------------|-----|------|------|------|
| | #15 | #14 | #13 | #12 | #11 | #10 | #9 | #8 |
| Setting | | | | | | | REV4 | |
| | #7 | #6 | #5 | #4 | #3 | #2 | #1 | #0 |
| Setting | SEQ | ABS | | INCH | ISO | TVON | REVY | REVX |
| REVX : X-axis mirror image on/off REVY : Y-axis mirror image on/off TVON : TV check on/off ISO : Output code, EIA/ISO INCH : Metric input/inch input ABS : Incremental programming/absolute programming SEQ : Sequence-number automatic insertion on/off REV4 : Fourth axis mirror image on/off | | | | | | | | |

Number of machined parts

The number (target number) of parts required and the number (completion number) of machined parts can be read and written.

Table 16.2 (f) System variables for the number of parts required and the number of machined parts

| Variable number | Function |
|-----------------|--|
| #3901 | Number of machined parts (completion number) |
| #3902 | Number of required parts (target number) |

NOTE

Do not substitute a negative value.

Modal information

Modal information specified in blocks up to the immediately preceding block can be read.

| Variable number | Function | |
|--|---|---|
| #4001 #4002 #4003 #4004 #4005 #4006 #4007 #4008 #4009 #4010 #4010 #4010 #4012 #4012 #4015 #4016 : #4022 #4102 #4102 #4107 #4109 #4111 #4113 #4114 #4115 #4119 #4120 | G00, G01, G02, G03, G33 G17, G18, G19 G90, G91 G94, G95 G20, G21 G40, G41, G42 G43, G44, G49 G73, G74, G76, G80–G89 G98, G99 G50, G51 G65, G66, G67 G54–G59 G61–G64 G68, G69 : B code D code F code H code M code Sequence number Program number S code T code | (Group 01) (Group 02) (Group 03) (Group 04) (Group 05) (Group 06) (Group 07) (Group 08) (Group 09) (Group 10) (Group 11) (Group 12) (Group 14) (Group 15) (Group 15) (Group 16) : (Group 22) |

Table 16.2(g) System variables for modal information

Example:

When #1=#4001; is executed, the resulting value in #1 is 0, 1, 2, 3, or 33.

• Current position

Position information cannot be written but can be read.

 Table 16.2(h)
 System variables for position information

| Variable number | Position information | Coordinate system | Tool com- pensation value | Read operation during movement |
|--------------------|-------------------------|-----------------------------------|---------------------------------|---|
| #5001–#5004 | Block end point | Workpiece coordinate system | Not included | Enabled |
| #5021-#5024 | Current position | Machine coordinate system | Included | Disabled |
| #5041-#5044 | Current position | Workpiece | | |
| #5061-#5064 | Skip signal position | system | | Enabled |
| #5081–#5084 | Tool offset value | | | Disabled |
| #5101-#5104 | Deviated servo position | | | |

- The first digit (from 1 to 4) represents an axis number. Digit 1 corresponds to the X-axis, digit 2 to the Y-axis, digit 3 to the Z-axis, and digit 4 to the fourth axis.
- The tool offset value currently used for execution rather than the immediately preceding tool offset value is held in variables #5081 to 5088.
- The tool position where the skip signal is turned on in a G31 (skip function) block is held in variables #5061 to #5068. When the skip signal is not turned on in a G31 block, the end point of the specified block is held in these variables.
- When read during movement is "disabled," this means that expected values cannot be read due to the buffering (preread) function.

 Workpiece coordinate system compensation values (workpiece zero point offset values) Workpiece zero point offset values can be read and written.

Table 16.2 (i) System variables for workpiece zero point offset values

| Variable Number | Function |
|-----------------|--|
| #2500 | First-axis external workpiece zero point offset value |
| #2501 | First–axis G54 workpiece zero point offset value |
| #2502 | First–axis G55 workpiece zero point offset value |
| #2503 | First–axis G56 workpiece zero point offset value |
| #2504 | First–axis G57 workpiece zero point offset value |
| #2505 | First–axis G58 workpiece zero point offset value |
| #2506 | First–axis G59 workpiece zero point offset value |
| #2600 | Second-axis external workpiece zero point offset value |
| #2601 | Second–axis G54 workpiece zero point offset value |
| #2602 | Second–axis G55 workpiece zero point offset value |
| #2603 | Second–axis G56 workpiece zero point offset value |
| #2604 | Second–axis G57 workpiece zero point offset value |
| #2605 | Second–axis G58 workpiece zero point offset value |
| #2606 | Second–axis G59 workpiece zero point offset value |
| #2700 | Third-axis external workpiece zero point offset value |
| #2701 | Third–axis G54 workpiece zero point offset value |
| #2702 | Third–axis G55 workpiece zero point offset value |
| #2703 | Third–axis G56 workpiece zero point offset value |
| #2704 | Third–axis G57 workpiece zero point offset value |
| #2705 | Third–axis G58 workpiece zero point offset value |
| #2706 | Third–axis G59 workpiece zero point offset value |
| #2800 | Fourth-axis external workpiece zero point offset value |
| #2801 | Fourth–axis G54 workpiece zero point offset value |
| #2802 | Fourth–axis G55 workpiece zero point offset value |
| #2803 | Fourth-axis G56 workpiece zero point offset value |
| #2804 | Fourth-axis G57 workpiece zero point offset value |
| #2805 | Fourth-axis G58 workpiece zero point offset value |
| #2806 | Fourth–axis G59 workpiece zero point offset value |

16.3 ARITHMETIC AND LOGIC OPERATION

The operations listed in Table 16.3(a) can be performed on variables. The expression to the right of the operator can contain constants and/or variables combined by a function or operator. Variables #j and #K in an expression can be replaced with a constant. Variables on the left can also be replaced with an expression.

Table 16.3(a) Arithmetic and logic operation

| Function | Format | Remarks | | | |
|----------------------------|-------------------|--|--|--|--|
| Definition | #i=#j | | | | |
| Sum | #i=#j+#k; | | | | |
| Difference | #i=#j—#k; | | | | |
| Product | #i=#j*#k; | | | | |
| Quotient | #i=#j/#k; | | | | |
| Sine | #i=SIN[#j]; | An angle is specified in de- | | | |
| Cosine | #i=COS[#j]; | grees. 90 degrees and 30 minutes is represented as | | | |
| Tangent | #i=TAN[#j]; | 90.5 degrees. | | | |
| Arctangent | #i=ATAN[#j]/[#k]; | | | | |
| Square root | #i=SQRT[#j]; | | | | |
| Absolute value | #i=ABS[#j]; | | | | |
| Rounding off | #i=ROUND[#j]; | | | | |
| Rounding down | #i=FIX[#j]; | | | | |
| Rounding up | #i=FUP[#j]; | | | | |
| OR | #i=#j OR #k; | A logical operation is per- | | | |
| XOR | #i=#j XOR #k; | formed on binary numbers bit by bit. | | | |
| AND | #i=#j AND #k; | 2.1.2) 2.1. | | | |
| Conversion from BCD to BIN | #i=BIN[#j]; | Used for signal exchange | | | |
| Conversion from BIN to BCD | #i=BCD[#j]; | to and from the PMC | | | |

Explanations

| Angle units | The units of angles used with the SIN, COS, TAN, and ATAN functions are degrees. For example, 90 degrees and 30 minutes is represented as 90.5 degrees. |
|---------------------------------|--|
| • ATAN function | After the ATAN function, specify the lengths of two sides separated by a slash. A result is found where $0 \leq \text{result} < 360$. |
| | Example : When #1=ATAN[1]/[–1], the value of #1 is 135.0 |
| • 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=ROUND[#2]; is executed where #2 holds 1.2345, the value of variable #1 is 1.0. |

When the ROUND function is used in NC statement addresses, the ROUND function rounds off the specified value according to the least input increment of the address.

Example:

Creation of a drilling program that cuts according to the values of variables #1 and #2, then returns to the original position

Suppose that the increment system is 1/1000 mm, variable #1 holds 1.2345, and variable #2 holds 2.3456. Then,

G00 G91 X-#1; G01 X-#2 F300;

G00 X[#1+#2];

Moves 1.235 mm. Moves 2.346 mm.

Since 1.2345 + 2.3456 = 3.5801, the travel distance is 3.580, which does not return the tool to the original position.

This difference comes from whether addition is performed before or after rounding off. G00X-[ROUND[#1]+ROUND[#2]] must be specified to return the tool to the original position.

 Rounding up and down With NC, 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 and #2=–1.2. When #3=FUP[#1] is executed, 2.0 is assigned to #3. When #3=FIX[#1] is executed, 1.0 is assigned to #3. When #3=FUP[#2] is executed, -2.0 is assigned to #3. When #3=FIX[#2] is executed, -1.0 is assigned to #3.

When a function is specified in a program, the first two characters of the function name can be used to specify the function.

Example:

```
ROUND \to RO
FIX \rightarrow FI
```

Priority of operations

arithmetic and logic

operation commands

Abbreviations of

to an integer

(1) Functions

(2) Operations such as multiplication and division (*, /, AND, MOD) (3) Operations such as addition and subtraction (+, -, OR, XOR)

| Example) #1=#2+#3*SIN[#4]; |
|---|
| |
| (1) |
| |
| (2) |
| |
| (3) |
| (1), (2), and (3) indicate the order of operations. |

— 225 —

• Bracket nesting

Brackets are used to change the order of operations. Brackets can be used to a depth of five levels including the brackets used to enclose a function. When a depth of five levels is exceeded, alarm No. 118 occurs.

| - |
|--|
| Example) #1=SIN [[[#2+#3] *#4 +#5] *#6] ; |
| (1) |
| |
| (2) |
| |
| |
| (4) |
| |
| (5) |
| (1) to (5) indicate the order of operations. |

Limitations

- Brackets
- Operation error

are used for comments.

Errors may occur when operations are performed.

Table 16.3(b) Errors involved in operations

Brackets ([,]) are used to enclose an expression. Note that parentheses

| Operation | Average error | Maximum error | Type of error | | |
|-----------------------|------------------------|------------------------|--|--|--|
| a = b*c | 1.55×10 ⁻¹⁰ | 4.66×10 ⁻¹⁰ | Relative error(*1) | | |
| a = b / c | 4.66×10 ⁻¹⁰ | 1.88×10 ⁻⁹ | $\frac{\epsilon}{\epsilon}$ | | |
| $a = \sqrt{b}$ | 1.24×10 ⁻⁹ | 3.73×10 ^{−9} | ∣a ∣ | | |
| a = b + c | 2.33×10 ⁻¹⁰ | 5.32×10 ⁻¹⁰ | (*2) | | |
| a = b - c | | | $ $ win $ \overline{b} , \overline{c} $ | | |
| a = SIN [b] | 5.0×10 ⁻⁹ | 1.0×10 ⁻⁸ | Absolute error(*3) | | |
| a = COS [b] | | | | | |
| a = ATAN [b]/[c] (*4) | 1.8×10 ⁻⁶ | 3.6×10 ⁻⁶ | ϵ degrees | | |

NOTE

- 1 The relative error depends on the result of the operation.
- 2 Smaller of the two types of errors is used.
- 3 The absolute error is constant, regardless of the result of the operation.
- 4 Function TAN performs SIN/COS.

The precision of variable values is about 8 decimal digits. When very large numbers are handled in an addition or subtraction, the expected results may not be obtained.

Example:

When an attempt is made to assign the following values to variables #1 and #2:

#1=9876543210123.456 #2=9876543277777.777

the values of the variables become:

#1=9876543200000.000

#2=9876543300000.000

In this case, when #3=#2-#1; is calculated, #3=100000.000 results. (The actual result of this calculation is slightly different because it is performed in binary.)

• Also be aware of errors that can result from conditional expressions using EQ, NE, GE, GT, LE, and LT.

Example:

IF[#1 EQ #2] is effected by errors in both #1 and #2, possibly resulting in an incorrect decision.

Therefore, instead find the difference between the two variables with IF[ABS[#1–#2]LT0.001].

Then, assume that the values of the two variables are equal when the difference does not exceed an allowable limit (0.001 in this case).

· Also, be careful when rounding down a value.

Example:

When #2=#1*1000; is calculated where #1=0.002;, the resulting value of variable #2 is not exactly 2 but 1.99999997.

Here, when #3=FIX[#2]; is specified, the resulting value of variable #1 is not 2.0 but 1.0. In this case, round down the value after correcting the error so that the result is greater than the expected number, or round it off as follows:

- #3=FIX[#2+0.001]
- #3=ROUND[#2]

Divisor

When a divisor of zero is specified in a division or TAN[90], alarm No. 112 occurs.

| 16.4 | The following blocks are referred to as macro statements: | | | | |
|---|---|--|--|--|--|
| MACRO STATEMENTS | • Blocks containing an arithmetic or logic operation (=) | | | | |
| AND NC STATEMENTS | • Blocks containing a control statement (such as GOTO, DO, END) | | | | |
| | Blocks containing a macro call command (such as macro calls by G65, G66, G67, or other G codes, or by M codes) | | | | |
| | Any block other than a macro statement is referred to as an NC statement. | | | | |
| Explanations | | | | | |
| Differences from NC statements | • Even when single block mode is on, the machine does not stop. Note, however, that the machine stops in the single block mode when bit 5 of parameter 011 is 1. | | | | |
| | • Macro blocks are not regarded as blocks that involve no movement in the cutter compensation mode (see Section 16.7). | | | | |
| NC statements that have the same property as macro statements | • Subprogram call blocks (blocks in which subprogram calls using M98, M codes, or T codes are specified) which only contain addresses O, N, P, and L have the same features as a macro statement. | | | | |
| | • Blocks containing M99 and addresses O, N, P, and L have the same features as a macro statement. | | | | |

| 16.5 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: | | | |
|--|---|--|--|--|
| NEFETTION | Branch and repetitionGOTO statement (unconditional branch) | | | |
| | IF statement (conditional branch: if, then) | | | |
| | WHILE statement (repetition while) | | | |
| | | | | |
| 16.5.1 Unconditional Branch (GOTO Statement) | A branch to sequence number n occurs. When a sequence number outside of the range 1 to 9999 is specified, alarm No. 128 occurs. A sequence number can also be specified using an expression. | | | |
| | GOTO n ; n: Sequence number (1 to 9999) | | | |
| | Example: GOTO1; GOTO#10; | | | |
| 16.5.2 Conditional Branch | Specify a conditional expression after IF. 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 Statement) | If the value of variable #1 is greater than 10, a branch to sequence number N2 occurs. | | | |
| | If the condi- tion is not sa- tisfied Processing N2 G00 G91 X10.0 ; - | | | |
| Explanations | | | | |
| Conditional expression | A conditional expression must include an operator inserted between two variables or between a variable and constant, and must be enclosed in brackets ([,]). An expression can be used instead of a variable. | | | |

Operators

Operators each consist of two letters and are used to compare two values to determine whether they are equal or one value is smaller or greater than the other value. Note that the inequality sign cannot be used.

| Operator | Meaning |
|----------|------------------------------------|
| EQ | Equal to (=) |
| NE | Not equal to (≠) |
| GT | Greater than (>) |
| GE | Greater than or equal to (\geqq) |
| LT | Less than (<) |
| LE | Less than or equal to (\leq) |

Table 16.5.2 Operators

Sample program

The sample program below finds the total of numbers 1 to 10.

| O9500; | |
|-------------------------|---|
| #1=0; | Initial value of the variable to hold the sum |
| #2=1; | Initial value of the variable as an addend |
| N1 IF[#2 GT 10] GOTO 2; | Branch to N2 when the addend is greater than 10 |
| #1=#1+#2; | Calculation to find the sum |
| #2=#2+1; | Next addend |
| GOTO 1; | Branch to N1 |
| N2 M30; | End of program |

16.5.3 Repetition (While Statement)

Specify a conditional expression after WHILE. While the specified condition is satisfied, the program from DO to END is executed. If the specified condition is not satisfied, program execution proceeds to the block after END.



Explanations

While the specified condition is satisfied, the program from DO to END after WHILE is executed. If the specified condition is not satisfied, program execution proceeds to the block after END. The same format as for the IF statement applies. A number after DO and a number after END are identification numbers for specifying the range of execution. The numbers 1, 2, and 3 can be used. When a number other than 1, 2, and 3 is used, alarm No. 126 occurs.

Nesting

The identification numbers (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), alarm No. 124 occurs.



Limitations

Infinite loops

When DO m is specified without specifying the WHILE statement, an infinite loop ranging from DO to END is produced.

- **Processing time** When a branch to the sequence number specified in a GOTO statement occurs, the sequence number is searched for. For this reason, processing in the reverse direction takes a longer time than processing in the forward direction. Using the WHILE statement for repetition reduces processing time.
- Undefined variable In a conditional expressio different effects. In other

In a conditional expression that uses EQ or NE, a null value and zero have different effects. In other types of conditional expressions, a null value is regarded as zero.

Sample program

The sample program below finds the total of numbers 1 to 10.

O0001; #1=0; #2=1; WHILE[#2 LE 10]DO 1; #1=#1+#2; #2=#2+1; END 1; M30;

| 16.6 | A macro program can be called using the following methods: | | | |
|--|--|--|--|--|
| MACRO CALL | Macro call Simple call ((G65) modal call (G66, G67) Macro call with G code Macro call with M code Subprogram call with M code Subprogram call with T code | | | |
| Limitations | | | | |
| • Differences between macro calls and subprogram calls | Macro call (G65) differs from subprogram call (M98) as described below. With G65, an argument (data passed to a macro) can be specified. M98 does not have this capability. When an M98 block contains another NC command (for example, G01 X100.0 M98Pp), the subprogram is called after the command is executed. On the other hand, G65 unconditionally calls a macro. When an M98 block contains another NC command (for example, G01 X100.0 M98Pp), the machine stops in the single block mode. On the other hand, G65 does not stops the machine. With G65, the level of local variables changes. With M98, the level of local variables does not change. | | | |

16.6.1 Simple Call (G65)

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



Explanations

• Call

- After G65, specify at address P the program number of the custom macro to call.
- When a number of repetitions is required, specify a number from 1 to 9999 after address L. When L is omitted, 1 is assumed.
- By using argument specification, values are assigned to corresponding local variables.

Argument specification

Two types of argument specification are available. Argument specification I uses letters other than G, L, O, N, and P once each. Argument specification II uses A, B, and C once each and also uses I, J, and K up to ten times. The type of argument specification is determined automatically according to the letters used.

Argument specification I

| Address | Variable number | Address | Variable number | Address | Variable number |
|---------|--------------------|---------|--------------------|---------|--------------------|
| А | #1 | I | #4 | Т | #20 |
| В | #2 | J | #5 | U | #21 |
| С | #3 | K | #6 | V | #22 |
| D | #7 | М | #13 | W | #23 |
| E | #8 | Q | #17 | Х | #24 |
| F | #9 | R | #18 | Y | #25 |
| Н | #11 | S | #19 | Z | #26 |

- · Addresses G, L, N, O, and P cannot be used in arguments.
- Addresses that need not be specified can be omitted. Local variables corresponding to an omitted address are set to null.

Argument specification II

Argument specification II uses A, B, and C once each and uses I, J, and K up to ten times. Argument specification II is used to pass values such as three–dimensional coordinates as arguments.

| Address | Variable number | Address | Variable number | Address | Variable number |
|---|--|---|---|---|---|
| A B C I ₁ J ₁ K ₁ I ₂ J ₂ K ₂ I ₃ | #1 #2 #3 #4 #5 #6 #7 #8 #9 #10 #11 | K ³ I4 J4 K4 I5 J5 K5 I6 J6 K6 | #12 #13 #14 #15 #16 #17 #18 #19 #20 #21 #22 | J ₇ K I ₈ J ₈ K I ₉ J ₉ g I ₁₀ J J ₁₀ J ₁₀ | #23 #24 #25 #26 #27 #28 #29 #30 #31 #32 #32 |

• Subscripts of I, J, and K for indicating the order of argument specification are not written in the actual program.

Limitations

- Format
- Mixture of argument specifications I and II
- Position of the decimal point

G65 must be specified before any argument.

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

The units used for argument data passed without a decimal point correspond to the least input increment of each address. The value of an argument passed without a decimal point may vary according to the system configuration of the machine. It is good practice to use decimal points in macro call arguments to maintain program compatibility. • Call nesting

Local variable levels

- Calls can be nested to a depth of four levels including simple calls (G65) and modal calls (G66). This does not include subprogram calls (M98).
 - · Local variables from level 0 to 4 are provided for nesting.
 - The level of the main program is 0.
 - Each time a macro is called (with G65 or G66), the local variable level is incremented by one. The values of the local variables at the previous level are saved in the NC.
 - When M99 is executed in a macro program, control returns to the calling program. At that time, the local variable level is decremented by one; the values of the local variables saved when the macro was called are restored.



Sample program (bolt hole circle)

A macro is created which drills H holes at intervals of B degrees after a start angle of A degrees along the periphery of a circle with radius I. 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.



• Calling format

| | G | 65 P9100 Xx Yy Zz Rr Ff li Aa Bb Hh; |
|----|---|---|
| X | : | X coordinate of the center of the circle (absolute or incremental (124) |
| •• | | specification) (#24) |
| Y | : | Y coordinate of the center of the circle (absolute or incremental |
| | | specification) |
| Ζ | : | Hole depth (#26) |
| R | : | Coordinates of an approach point |
| F | : | Cutting feedrate |
| I | : | Radius of the circle |
| A | : | Drilling start angle (#1) |
| В | : | Incremental angle (clockwise when a negative value is specified) |
| | | |
| Η | : | Number of holes (#11) |

####
| Program calling a macro program | O0002; G90 G92 X0 Y0 Z100.0; G65 P9100 X100.0 Y50.0 R30.0 Z–50.0 F500 I100.0 A0 B45.0 H5; M30; |
|---|---|
| Macro program (called program) | O9100; #3=#4003; |

M99;

Meaning of variables:

#3: Stores the G code of group 3.#5: X coordinate of the next hole to drill#6: Y coordinate of the next hole to drill

16.6.2 Modal Call (G66)

Once G66 is issued to specify a modal call a macro is called after a block specifying movement along axes is executed. This continues until G67 is issued to cancel a modal call.



Explanations

| • Call | • After G66, specify at address P a program number subject to a modal call. |
|--|---|
| | • When a number of repetitions is required, a number from 1 to 9999 can be specified at address L. |
| | • As with a simple call (G65), data passed to a macro program is specified in arguments. |
| Cancellation | When a G67 code is specified, modal macro calls are no longer performed in subsequent blocks. |
| Call nesting | Calls can be nested to a depth of four levels including simple calls (G65) and modal calls (G66). This does not include subprogram calls (M98). |
| Modal call nesting | Modal calls can be nested by specifying another G66 code during a modal call. |
| Limitations | · In a G66 block, no macros can be called. |
| | · G66 needs to be specified before any arguments. |
| | • No macros can be called in a block which contains a code such as a miscellaneous function that does not involve movement along an axis. |
| | • Local variables (arguments) can only be set in G66 blocks. Note that local variables are not set each time a modal call is performed. |

Sample program

The same operation as the drilling canned cycle G81 is created using a custom macro and the machining program makes a modal macro call. For program simplicity, all drilling data is specified using absolute values.



• Calling format

| | G65 P9110 X x Y y Z z R r F f L I; |
|--|--|
| | X : X coordinate of the hole (absolute specification only) (#24) Y : Y coordinate of the hole (absolute specification only) (#25) Z : Coordinates of position Z (absolute specification only) (#26) R : Coordinates of position R (absolute specification only) (#18) F : Cutting feedrate |
| Program that calls a macro program | O0001; G28 G91 X0 Y0 Z0; G92 X0 Y0 Z50.0; G00 G90 X100.0 Y50.0; G66 P9110 Z–20.0 R5.0 F500; G90 X20.0 Y20.0; X50.0; Y50.0; X70.0 Y80.0; G67; M30; |
| Macro program (program called) | O9110;#1=#4001;Stores G00/G01.#3=#4003;Stores G90/G91.#4=#4109;Stores the cutting feedrate.#5=#5003;Stores the Z coordinate at the start of drilling.G00 G90 Z#18;Positioning at position RG01 Z#26 F#9;Cutting feed to position ZIF[#4010 EQ 98]GOTO 1;Return to position IG00 Z#18;Positioning at position RG0TO 2;N1 G00 Z#5;N1 G00 Z#5;Positioning at position IN2 G#1 G#3 F#4;Restores modal information.M99;M99; |

Γ

16.6.3 Macro Call Using G Code

By setting a G code number used to call a macro program in a parameter, the macro program can be called in the same way as for a simple call (G65).



Explanations

By setting a G code number from 1 to 255 used to call a custom macro program (9010 to 9019) in the corresponding parameter (220 to 229), the macro program can be called in the same way as with G65. For example, when a parameter is set so that macro program O9010 can be called with G81, a user–specific cycle created using a custom macro can be called without modifying the machining program.

 Correspondence between parameter numbers and program numbers

| Program number | Parameter number |
|----------------|------------------|
| O9010 | 220 |
| O9011 | 221 |
| O9012 | 222 |
| O9013 | 223 |
| O9014 | 224 |
| O9015 | 225 |
| O9016 | 226 |
| O9017 | 227 |
| O9018 | 228 |
| O9019 | 229 |

- **Repetition** As with a simple call, a number of repetitions from 1 to 9999 can be specified at address L.
- Argument specification As with a simple call, two types of argument specification are available: Argument specification I and argument specification II. The type of argument specification is determined automatically according to the addresses used.

Limitations

Nesting of calls using G codes
 In a program called with a G code, no macros can be called using a G code. A G code in such a program is treated as an ordinary G code. In a program called as a subprogram with an M or T code, no macros can be called using a G code. A G code in such a program is also treated as an ordinary G code.

16.6.4 Macro Call Using an M Code

By setting an M code number used to call a macro program in a parameter, the macro program can be called in the same way as with a simple call (G65).



Explanations

By setting an M code number from 1 to 255 used to call a custom macro program (9020 to 9029) in the corresponding parameter (230 to 239), the macro program can be called in the same way as with G65.

 Correspondence between parameter numbers and program numbers

| Program number | Parameter number |
|----------------|------------------|
| O9020 | 230 |
| O9021 | 231 |
| O9022 | 232 |
| O9023 | 233 |
| O9024 | 234 |
| O9025 | 235 |
| O9026 | 236 |
| O9027 | 237 |
| O9028 | 238 |
| O9029 | 239 |

| Repetition | As with a simple call, a number of repetitions from 1 to 9999 can be specified at address L. |
|--|--|
| Argument specification | As with a simple call, two types of argument specification are available: Argument specification I and argument specification II. The type of argument specification is determined automatically according to the addresses used. |
| Limitations | An M code used to call a macro program must be specified at the start of a block. |
| | |

 In a macro called with a G code or in a program called as a subprogram with an M or T code, no macros can be called using an M code. An M code in such a macro or program is treated as an ordinary M code.

| 16.6.5 Subprogram Call Using an M Code | By setting an M code number used to call a subprogram (macro program) in a parameter, the macro program can be called in the same way as with a subprogram call (M98). |
|--|--|
| Explanations | By setting an M code number from 1 to 255 used to call a subprogram in a parameter (240 to 242), the corresponding macro program (9001 to 9009) can be called in the same way as with M98. |
| Correspondence between parameter numbers and program numbers | Program numberParameter numberO9001240O9002241O9003242 |
| Repetition | As with a simple call, a number of repetitions from 1 to 9999 can be specified at address L. |
| Argument specification | Argument specification is not allowed. |
| • M code | An M code in a macro program that has been called is treated as an ordinary M code. |
| Limitations | In a macro called with a G code or in a program called with an M or T code, no subprograms can be called using an M code. An M code in such a macro or program is treated as an ordinary M code. |

| 16.6.6 Subprogram Calls Using a T Code | By enabling subprograms (macro program) to be called with a T code in a parameter, a macro program can be called each time the T code is specified in the machining program. $\begin{array}{c c} \hline 00001; \\ \vdots \\ T23; \\ \vdots \\ M30; \\ \hline M30; \\ \hline \\ Bit 5 of parameter 040 = 1 \end{array}$ | |
|--|---|--|
| Explanations | | |
| ● Call | By setting bit 5 of parameter 040 to 1, the macro program O9000 can be called when a T code is specified in the machining program. A T code specified in a machining program is assigned to common variable #149. | |
| Limitations | In a macro called with a G code or in a program called with an M or T code, no subprograms can be called using a T code. A T code in such a macro or program is treated as an ordinary T code. | |

| 16.6.7 Sample Program | By using the subprogram call function that uses M codes, the cumulative usage time of each tool is measured. |
|---------------------------------------|--|
| Conditions | • The cumulative usage time of each of tools T01 to T05 is measured. No measurement is made for tools with numbers greater than T05. |
| | • The following variables are used to store the tool numbers and measured times: |
| | #501 Cumulative usage time of tool number 1 |
| | #502 Cumulative usage time of tool number 2 |
| | #503 Cumulative usage time of tool number 3 |
| | #504 Cumulative usage time of tool number 4 |
| | #505 Cumulative usage time of tool number 5 |
| | • Usage time starts being counted when the M03 command is specified and stops when M05 is specified. System variable #3002 is used to measure the time during which the cycle start lamp is on. The time during which the machine is stopped by feed hold and single block stop operation is not counted, but the time used to change tools and pallets is included. |
| Operation check | |
| Parameter setting | Set 3 in parameter 240, and set 05 in parameter 241. |
| • Variable value setting | Set 0 in variables #501 to #505. |
| • Program that calls a macro program | O0001; T01 M06; M03; G04 X20.0; M05; Changes #501. T02 M06; M03; G04 X20.0; M05; Changes #502. T03 M06; M03; G04 X20.0; M03; G04 X20.0; M05; Changes #503. T04 M06; M03; G04 X20.0; M05; Changes #504. T05 M06; M03; G04 X20.0; M05; Changes #504. T05 M06; M03; G04 X20.0; M05; Changes #505. |

- 244 -

Macro program (program called)

| O9001(M03); Macro to start counting M01 . |
|---|
| IF[#4120 EQ 0]GOTO 9; No tool specified IF[#4120 GT 5]GOTO 9; Out–of–range tool number #3002=0; Clears the timer. |
| N9 M03; Rotates the spindle in the forward direction. M99; |
| O9002(M05); Macro to end counting M01; |
| IF[#4120 EQ 0]GOTO 9; No tool specified IF[#4120 GT 5]GOTO 9; Out–of–range tool number #[500+#4120]=#3002+#[500+#4120]; Calculates cumulative time. |
| N9 M05; Stops the spindle. M99; |

16.7 PROCESSING MACRO STATEMENTS

For smooth machining, the NC statement is preread to be performed next. This operation is referred to as buffering. In cutter compensation mode (G41, G42), the NC prereads NC statements two or three blocks ahead to find intersections. Macro statements for arithmetic expressions and conditional branches are processed as soon as they are read into the buffer. Blocks containing M00, M01, M02, or M30, blocks containing M codes for which buffering is suppressed by setting parameters 111 to 112, and blocks containing G31 are not preread.

Explanations

 When the next block is not buffered (M codes that are not buffered, G31, etc.)



 Buffering the next block in other than cutter compensation mode (G41, G42) (normally prereading one block)



When N1 is being executed, the next NC statement (N4) is read into the buffer. The macro statements (N2, N3) between N1 and N4 are processed during execution of N1.

 Buffering the next block in cutter compensation mode (G41, G42)



When N1 is being executed, the NC statements in the next two blocks (up to N5) are read into the buffer. The macro statements (N2, N4) between N1 and N5 are processed during execution of N1.

 When the next block involves no movement in cutter compensation C (G41, G42) mode



When the NC1 block is being executed, the NC statements in the next two blocks (up to N5) are read into the buffer. Since N5 is a block that involves no movement, an intersection cannot be calculated. In this case, the NC statements in the next three blocks (up to N7) are read. The macro statements (N2, N4, and N6) between N1 and N7 are processed during execution of N1.

16.8 REGISTERING CUSTOM MACRO PROGRAMS

Custom macro programs are similar to subprograms. They can be registered and edited in the same way as subprograms. The storage capacity is determined by the total length of tape used to store both custom macros and subprograms.

16.9 LIMITATIONS

| MDI operation | Macro call can be specified in automatic operation. During automatic operation, however, it is impossible to switch to the MDI mode for a macro program call. |
|---|---|
| Sequence number search | A custom macro program cannot be searched for a sequence number. |
| • Single block | Even while a macro program is being executed, blocks can be stopped in the single block mode (except blocks containing macro call commands, arithmetic operation commands, and control commands). A block containing a macro call command (G65, G66, or G67) does not stop even when the single block mode is on. Blocks containing arithmetic operation commands and control commands can be stopped in single block mode by setting (bit 5 of parameter 011) to 1. Single block stop operation is used for testing custom macro programs. Note that when a single block stop occurs at a macro statement in cutter compensation C mode, the statement is assumed to be a block that does not involve movement, and proper compensation cannot be performed in some cases. (Strictly speaking, the block is regarded as specifying a movement with a travel distance 0.) |
| Optional block skip | A / 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 specifier for an optional block skip code.</expression> |
| Operation in EDIT mode | Registered custom macro programs and subprograms should be protected from being destroyed by accident. By setting (bit 0 of parameter 389 #2) and (bit 4 of parameter 010) to 1, deletion and editing are disabled for custom macro programs and subprograms with program numbers 8000 to 8999 and 9000 to 9999. When the entire memory is cleared (by pressing the RESET and DELET keys at the same time to turn on the power), the contents of memory such as custom macro programs are deleted |
| • Reset | Local variables and common variables #100 to #199 are cleared to null values by a reset operation. They can be prevented from being cleared by setting, (bits 6 and 7 of parameter 040). System variables #1000 to #1133 are not cleared. A reset operation clears any called states of custom macro programs and subprograms, and any DO states, and returns control to the main program. |
| Display of the PROGRAM RESTART page | As with M98, the M and T codes used for subprogram calls are not displayed. |
| Feed hold | When a feed hold is enabled during execution of a macro statement, the machine stops after execution of the macro statement. The machine also stops when a reset or alarm occurs. |
| Constant values that can be used in <expression></expression> | +0.0000001 to +99999999 -99999999 to -0.0000001 The number of significant digits is 8 (decimal). If this range is exceeded, alarm No. 003 occurs. |

| 16.10 EXTERNAL OUTPUT COMMANDS | In addition to the custom macro commands, the following macro commands are available. They are referred to as external output commands. - BPRNT - DPRNT - POPEN - PCLOS These commands are provided to output variable values and characters through the reader/punch interface. |
|---|--|
| Explanations | Specify these commands in the following order: |
| | Open command: POPEN Before specifying a sequence of data output commands, specify this command to establish a connection to an external input/output device. |
| | Data output command: BPRNT or DPRNT Specify necessary data output. |
| | Close command: PCLOS When all data output commands have completed, specify PCLOS to release a connection to an external input/output device. |
| Open command POPEN | POPEN POPEN establishes a connection to an external input/output device. It must be specified before a sequence of data output commands. The NC outputs a DC2 control code. |
| Data output command BPRNT | BPRNT [a #b [c]] Number of significant decimal places Variable Character |
| | The BPRNT command outputs characters and variable values in binary. |
| | (i) Specified characters are converted to corresponding codes according to the setting (ISO) that is output at that time. Specifiable characters are as follows: |

- Letters (A to Z)
- Numbers
- Special characters (*, /, +, -, etc.)

An asterisk (*) is output by a space code.

- (ii) All variables are stored with a decimal point. Specify a variable followed by the number of significant decimal places enclosed in brackets. A variable value is treated as 2-word (32-bit) data, including the decimal digits. It is output as binary data starting from the highest byte.
- (iii)When specified data has been output, an EOB code is output according to the ISO code settings on the parameter screen.
- (iv) Null variables are regarded as 0.





 Data output command DPRNT



The DPRNT command outputs characters and each digit in the value of a variable according to the code set in the settings (ISO).

- (i) For an explanation of the DPRNT command, see Items (i), (iii), and (iv) for the BPRNT command.
- (ii) When outputting a variable, specify # followed by the variable number, then specify the number of digits in the integer part and the number of decimal places enclosed in brackets.

One code is output for each of the specified number of digits, starting with the highest digit. For each digit, a code is output according to the settings (ISO). The decimal point is also output using a code set in the settings (ISO).

Each variable must be a numeric value consisting of up to eight digits. When high–order digits are zeros, these zeros are not output if (bit1 of parameter 040) is 1. If it is 0, a space code is output each time a zero is encountered.

When the number of decimal places is not zero, digits in the decimal part are always output. If the number of decimal places is zero, no decimal point is output.

When (bit 1 of parameter 040) is 0, a space code is output to indicate a positive number instead of +; if it is 1, no code is output.

Example)



• Close command PCLOS

PCLOS;

The PCLOS command releases a connection to an external input/output device. Specify this command when all data output commands have terminated. DC4 control code is output from the CNC.

• Required setting

Specify the channel use for parameter. According to the specification of this parameter, set data items (such as the baud rate) for the reader/punch interface. The remote buffer cannot be specified.

Specify that the reader/punch interface is used as the output device for punching. (Never specify output to the FANUC CASSETTE or floppy disks.)

To indicate the end of a line of data in ISO code, specify whether to use only an LF (bit 4 of parameter 057 is 0) or an LF and CR (bit 4 of parameter 057 is 1).

NOTE

- 1 It is not necessary to always specify the open command (POPEN), data output command (BPRNT, DPRNT), and close command (PCLOS) together. Once an open command is specified at the beginning of a program, it does not need to be specified again except after a close command was specified.
- 2 Be sure to specify open commands and close commands in pairs. Specify the close command at the end of the program. However, do not specify a close command if no open command has been specified.
- 3 When a reset operation is performed while commands are being output by a data output command, output is stopped and subsequent data is erased. Therefore, when a reset operation is performed by a code such as M30 at the end of a program that performs data output, specify a close command at the end of the program so that processing such as M30 is not performed until all data is output.

PROGRAMMABLE PARAMETER ENTRY(G10)

The values of parameters can be entered in a lprogram. This function is used for setting pitch error compensation data when attachments are changed or the maximum cutting feedrate or cutting time constants are changed to meet changing machining conditions.

Format

| Format | | | | | | | |
|--|------------------------------|--|--|--|--|--|--|
| G10L50; N_P_; | Parameter entry mode setting | | | | | | |
| | | | | | | | |
| G11; | Parameter entry mode cancel | | | | | | |
| Meaning of command | | | | | | | |
| N_: Parameter No. (4digids) or data No. for pitch errors compensation P_: Parameter setting value (Leading zeros can be omitted.) | | | | | | | |

Explanations

Parameter setting value (P_)

Can not use a decimal point in a value set in a parameter (P_). a decimal point cannot be used in a custom macro variable for P_ either.

WARNING

- 1 Do not fail to perform reference point return manually after changing the pitch error compensation data or backlash compensation data. Without this, the machine position can deviate from the correct position.
- 2 The canned–cycle mode must be cancelled before entering of parameters.

NOTE

Other NC statements cannot be specified while in parameter input mode.

Examples

To change the length of the effective area (in–position) along
the X–axisG10L50 ;Parameter entry modeN500 P50 ;Cancel parameter entry mode



ROTARY AXIS ROLL-OVER

General

The roll–over function prevents coordinates for the rotation axis from overflowing. The roll–over function is enabled by setting bit 1 of parameter 398 to 1.

Explanations For an incremental command, the tool moves the angle specified in the command. For an absolute command, the coordinates after the tool has moved are values set in parameter No. 860, and rounded by the angle corresponding to one rotation. The tool moves in the direction in which the final coordinates are closest when bit 2 of parameter No. 398 is set to 0. Displayed values for relative coordinates are also rounded by the angle corresponding to one rotation when bit 3 of parameter No. 398 is set to 1.

Examples

Assume that axis A is the rotating axis and that the amount of movement per rotation is 360.000 (parameter No. 1260 = 360000). When the following program is executed using the roll–over function of the rotating axis, the axis moves as shown below.

| G90 A0 ; | Sequence number | Actual movement value | Absolute coordinate value after movement end |
|------------------|--------------------|-----------------------------|--|
| N1 G90 A–150.0 ; | N1 | -150 | 210 |
| N2 G90 A540.0 ; | N2 | -30 | 180 |
| N3 G90 A–620.0 ; | N3 | -80 | 100 |
| N4 G91 A380.0 ; | N4 | +380 | 120 |
| N5 G91 A–840.0 ; | N5 | -840 | 0 |



— 256 —

III OPERATION



1.1 MANUAL OPERATION

Explanations

 Manual reference position return (See Section III–3.1) The CNC machine tool has a position used to determine the machine position.

This position is called the reference position, where the tool is replaced or the coordinate are set. Ordinarily, after the power is turned on, the tool is moved to the reference position.

Manual reference position return is to move the tool to the reference position using switches and pushbuttons located on the operator's panel.



Fig.1.1 (a) Manual reference position return

The tool can be moved to the reference position also with program commands.

This operation is called automatic reference position return (See II-6).

• The tool movement by manual operation

Using machine operator's panel switches, pushbuttons, or the manual handle, the tool can be moved along each axis.



Fig.1.1 (b) The tool movement by manual operation

The tool can be moved in the following ways:

- (i) Jog feed (See III–3.2) The tool moves continuously while a pushbutton remains pressed.
- (ii) Incremental feed (See III–3.3) The tool moves by the predetermined distance each time a button is pressed.
- (iii) Manual handle feed (See III–3.4)By rotating the manual handle, the tool moves by the distance corresponding to the degree of handle rotation.

1.2 TOOL MOVEMENT BY PROGRAMING – AUTOMATIC OPERATION

Automatic operation is to operate the machine according to the created program. It includes memory, DNC, and MDI operations. (See III–4).



Fig.1.2 (a) Tool Movement by Programming

Explanations

• Memory operation

After the program is once registered in memory of CNC, the machine can be run according to the program instructions. This operation is called memory operation.



Fig.1.2 (b) Memory Operation

In this mode of operation, the program is not registered in the CNC memory. It is read from the connected I/O unit instead. This mode is

DNC operation

MDI operation
 useful when the program is too large to fit the CNC memory.
 After the program is entered, as an command group, from

After the program is entered, as an command group, from the MDI keyboard, the machine can be run according to the program. This operation is called MDI operation.



Fig.1.2 (c) MDI operation

— 262 —

1.3 AUTOMATIC OPERATION

Explanations

• Program selection

Select the program used for the workpiece. Ordinarily, one program is prepared for one workpiece. If two or more programs are in memory, select the program to be used, by searching the program number (III–9.3).



Fig.1.3 (a) Program Selection for Automatic Operation

Pressing the cycle start pushbutton causes automatic operation to start. By pressing the feed hold or reset pushbutton, automatic operation pauses or stops. By specifying the program stop or program termination command in the program, the running will stop during automatic operation. When one process machining is completed, automatic operation stops.



Fig.1.3 (b) Start and Stop for Automatic Operation

 Start and stop (See III–4)

1.4 TESTING A PROGRAM

Before machining is started, the automatic running check can be executed. It checks whether the created program can operate the machine as desired. This check can be accomplished by running the machine actually or viewing the position display change (without running the machine) (See III–5).

1.4.1 Check by Running the Machine

Explanations

• Dry run (See III–5.4)

Remove the workpiece, check only movement of the tool. Select the tool movement rate using the dial on the operator's panel.





• Feedrate override (See III–5.2)

Check the program by changing the feed rate specified by the program.



Fig1.4 (b) Feedrate Override

— 264 —

- Single block (See III–5.5)
- When the cycle start pushbutton is pressed, the tool executes one operation then stops. By pressing the cycle start again, the tool executes the next operation then stops. The program is checked in this manner.



Fig.1.4 (c) Single Block

1.4.2

How to View the Position Display Change without Running the Machine

Explanations

• Machine lock (See III–5.1)



Fig1.4 (d) Machine Lock

• Auxiliary function lock (See III-5.1)

When automatic running is placed into the auxiliary function lock mode during the machine lock mode, all auxiliary functions (spindle rotation, tool replacement, coolant on/off, etc.) are disabled.

1.5 EDITING A PART PROGRAM

After a created program is once registered in memory, it can be corrected or modified from the CRT/MDI panel (See III–9).

This operation can be executed using the part program storage/edit function.



Fig. 1.5 (a) Part Program Editing

1.6 DISPLAYING AND SETTING DATA

The operator can display or change a value stored in CNC internal memory by key operation on the CRT/MDI screen (See III–11).



Fig.1.6 (a) Displaying and Setting Data

Explanations

Offset value



Fig.1.6 (b) Displaying and Setting Offset Values

The tool has the tool dimension (length, diameter). When a workpiece is machined, the tool movement value depends on the tool dimensions.

By setting tool dimension data in CNC memory beforehand, automatically generates tool routes that permit any tool to cut the workpiece specified by the program. Tool dimension data is called the offset value (See III–11.4.1).



Fig.1.6 (c) Offset Value

• Displaying and setting operator's setting data

Apart from parameters, there is data that is set by the operator in operation. This data causes machine characteristics to change.

For example, the following data can be set:

·Inch/Metric switching

·Selection of I/O devices

 \cdot Mirror image cutting on/off

The above data is called setting data (See III-11.5.3).



Fig.1.6 (d) Displaying and Setting Operator's setting data

• Displaying and setting parameters

The CNC functions have versatility in order to take action in characteristics of various machines.

For example, CNC can specify the following:

- · Rapid traverse rate of each axis
- · Whether increment system is based on metric system or inch system.
- · How to set command multiply/detect multiply (CMR/DMR)

Data to make the above specification is called parameters (See III-11.5.1).

Parameters differ depending on machine tool.



Fig.1.6 (e) Displaying and setting parameters

• Data protection key

A key called the data protection key can be defined. It is used to prevent part programs from being registered, modified, or deleted erroneously (See III–11).



Fig.1.6 (f) Data Protection Key

1.7 DISPLAY

1.7.1 The contents of the currently active program are displayed. In addition, the programs scheduled next and the program list are **Program Display** displayed. (See III-11.2.1 and Active sequence number İII–11.3.1) Active program number -¥ O1100 N0005 PROGRAM N1 G90 G17 G00 G41 D07 X250.0 Y550.0 ; N2 G01 Y900.0 F150 ; N3 X450.0; N4 G03 X500.0 Y1150.0 R650.0; <u>N</u>5 G02 X900.0 R-250.0 ; Program ▲ N6 G03 X950.0 Y900.0 R650.0 ; content N7 G01 X1150.0; N8 Y550.0; N9 X700.0 Y650.0; N10 X250.0 Y550.0; N11 G00 G40 X0 Y0; **BUF AUTO** (PRGRM) (CURRNT) (NEXT) (CHECK) (Currently executed program

The cursor indicates the currently executed location

| / | | |
|---|---|---|
| , | PROGRAM O1100 N0003 | |
| | SYSTEM EDITION 0471 – 04 PROGRAM NO. USED ' 10 FREE ' 53 MEMORY AREA USED ' 960 FREE ' 5280 | |
| | PROGRAM LIBRARY LIST O0001 O0002 O0010 O0020 O0040 O0050 O0100 O0200 O1000 O1100 | |
| | EDIT | |
| | (PRGRM)(CONDNS)())() |) |

1.7.2 Current Position Display (See III–11.1.1 to 11.1.3)

The current position of the tool is displayed with the coordinate values. The distance from the current position to the target position can also be displayed.





1.7.3 Alarm Display (See III–7.1) When a trouble occurs during operation, error code and alarm message are displayed on CRT screen. See APPENDIX 7 for the list of error codes and their meanings.

| ALARM | MESSAGE | | | O1000 | N0003 |
|-----------|-----------|----|---------|-------|-------|
| 010 | P/S ALARM | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| 09 : 52 : | : 24 | | ALARM | | |
| | M) (OPR |)(| MSG) (|)(|)] |

1.7.4 Parts Count Display, Run Time Display (See III–11.1.5) When an option is selected, two types of run time and number of parts are displayed on the screen.



1.8 DATA OUTPUT (See III–8)

Programs, offset values, parameters, etc. input in CNC memory can be output to paper tape, cassette, or a floppy disk for saving. After once output to a medium, the data can be input into CNC memory.



Fig.1.8 (a) Data Output


OPERATIONAL DEVICES

The peripheral devices available include the CRT/MDI panel attached to the CNC, machine operator's panel and external input/output devices such as tape reader, PPR, floppy cassette, and FA card.

2.1 CRT/MDI PANELS AND LCD/MDI PANELS

Fig. 2.1 (a) show the CRT/MDI and LCD/MDI panels.

| 9" small monochrome or color CRT/MDI panel (horizontal type) | Fig.2.1(a) |
|--|------------|
|--|------------|

External view



Fig. 2.1(a) 9" small monochrome/or color CRT/MDI panel (with soft key)

Explanation of the keyboard



Fig. 2.1(b) 9" small monochrome or color CRT/MDI (with softkey)

| Number | Name | Explanation | | | |
|--------|---|---|--|--|--|
| 1 | RESET key | Press this key to reset the CNC, to cancel an alarm, etc. | | | |
| 2 | START key | This key is used to start MDI operation or automatic operation, depending on the machine. Refer to the manual provided by the machine tool builder. This key is also used to output data to an input/output unit. | | | |
| 3 | Soft keys (option) | The soft keys have various functions, according to the Applications. The soft key functions are displayed at the bottom of the CRT screen. | | | |
| 4 | Address and numeric keys $N_{()} \leftarrow 4 \cdots$ | Press these keys to input alphabetic, numeric, and other characters. | | | |
| 5 | | When an address or a numerical key is pressed, the data is input to the buffer, and it is displayed on the CRT screen. To copy the data in the key input buffer to the offset register, etc., press the <input/> key. This key is also used to input data from an input/output unit. | | | |
| 6 | Cancel key | Press this key to delete the last character or symbol input to the key input buffer. | | | |
| 7 | Program edit keys | Press these keys when editing the program. ALTER INSRT DELET ALTER : Alteration INSRT : Insertion DELET : Deletion | | | |

Table2.1 Explanation of the MDI keyboard

| Number | Name | Explanation |
|--------|----------------------------|---|
| 8 | Function keys | Press theses keys to switch display screens for each function. See 2.2 for detailas of the function keys. |
| 9 | Cursor move keys CURSOR | There are two different cursor move keys. This key is used to move the cursor in a downward or forward direction. The cursor is moved in small units in the forward direction. This key is used to move the cursor in an upward or reverse direction. The cursor is moved in small units in the reverse direction. The cursor is moved in small units in the reverse direction. |
| 10 | Page change keys PAGE | Two kinds of page change keys are described below. This key is used to changeover the page on the CRT screen in the forward direction. This key is used to changeover the page on the CRT screen in the reverse direction. |

2.2 FUNCTION KEYS AND SOFT KEYS

2.2.1 General Screen Operations



Return menu key Continuous menu key

- **1** Press a function key on the CRT/MDI panel. The chapter selection soft keys that belong to the selected function appear.
- 2 Press one of the soft keys. The screen for the selected chapter appears. If the soft key for a target chapter is not displayed, press the continuous menu key (next-menu key). In some cases, additional chapters can be selected within a chapter.
- **3** To redisplay the previous soft keys, press the return menu key.

The general screen display procedure is explained above. However, the actual display procedure varies from one screen to another. For details, see the description of individual operations.

If the soft keys are not provided, press a key having the equivalent function to select the desired chapter.

The functions of the soft keys are automatically displayed, depending on the configuration. The functions are displayed, irrespective of the function configuration, if bit 7 of parameter 048 is specified accordingly.

| 2.2.2 Function Keys | Function keys are provided to select the type of screen to be displayed. The following function keys are provided on the CRT/MDI panel: |
|------------------------|--|
| POS | Press this key to display the position screen . |
| PRGRM | Press this key to display the program screen . |
| MENU OFSET | Press this key to display the offset screen . |
| DGNOS PARAM | Press this key to display the parameter/diagnosis screen . |
| OPR ALARM | Press this key to display the alarm screen. |
| AUX GRAPH | This key is not used. |

2.2.3 Key Input and Input Buffer

Explanations

• For standard key

When an address and a numerical key are pressed, the character corresponding to that key is input once into the key input buffer. The contents of the key input buffer is displayed at the bottom of the CRT screen.

The key input buffer begins with "ADDRESS". When an address is input, the key input buffer begins with "NUMERIC".



Fig. 2.2.3 Key input buffer display

On the standard key panel, the same key is used to input both an address and a numeric value.

When the key input buffer begins with "ADDRESS", pressing that key inputs the address. When the key input buffer begins "NUMERIC", pressing that key inputs the numeric value.



Data of one word (address + numeric value) can be entered into the key input buffer at one time. The following data input keys are used to input addresses. Each time the key is pressed, the input address changes as shown below:



Pressing the CAN key deletes all the data input to the key input buffer.

When bit 7 of parameter 0394 is set to 1, each press of the $\begin{bmatrix} CAN \end{bmatrix}$ key deletes only the most recently entered character during data input using the parameter, diagnostic, or offset screen.

2.3 EXTERNAL I/O DEVICES

Five types of external input/output devices are available. This section outlines each device. For details on these devices, refer to the corresponding manuals listed below.

Table 2.3(a) External I/O device

| Device name | Usage | Max. storage capacity | Reference manual |
|-----------------------|--|-----------------------------|---------------------|
| FANUC Handy File | Easy-to-use, multi function input/output device. It is de- signed for FA equipment and uses floppy disks. | 3600m | B–61834E |
| FANUC Floppy Cassette | Input/output device. Uses floppy disks. | 2500m | B–66040E |
| FANUC FA Card | Compact input/output device. Uses FA cards. | 160m | B–61274E |
| FANUC PPR | Input/output device consist- ing of a paper tape reader, tape punch, and printer. | 275m | B–58584E |
| Portable Tape Reader | Input device for reading pa- per tape. | | Appendix H |

The following data can be input/output to or from external input/output devices:

- 1. Programs
- 2. Offset data
- 3. Parameters
- 4. Custom macro common variables

For how data is input and output, see Chapter 8.

Parameter

Before an external input/output device can be used, parameters must be set as follows.



Series 0–D has one channels of reader/punch interfaces. The input/output device to be used is specified by setting the channel connected to that device in setting parameter I/O.

The specified data, such as a baud rate and the number of stop bits, of an input/output device connected to a specific channel must be set in parameters for that channel in advance.

For channel 1, two combinations of parameters to specify the input/output device data are provided.

The following shows the interrelation between the reader/punch interface parameters for the channels.



2.3.1 FANUC Handy File

The Handy File is an easy-to-use, multi function floppy disk input/output device designed for FA equipment. By operating the Handy File directly or remotely from a unit connected to the Handy File, programs can be transferred and edited.

The Handy File uses 3.5–inch floppy disks, which do not have the problems of paper tape (i.e., noisy during input/output, easily broken, and bulky).

One or more programs (up to 1.44M bytes, which is equivalent to the memory capacity of 3600–m paper tape) can be stored on one floppy disk.



2.3.2 FANUC Floppy Cassette

When the Floppy Cassette is connected to the NC, machining programs stored in the NC can be saved on a Floppy Cassette, and machining programs saved in the Floppy Cassette can be transferred to the NC.



2.3.3 FANUC FA Card

An FA Card is a memory card used as an input medium in the FA field. It is compact, but has a large memory capacity with high reliability, and requires no special maintenance.

When an FA Card is connected to the CNC via the card adapter, NC machining programs stored in the CNC can be transferred to and saved in an FA Card. Machining programs stored on an FA Card can also be transferred to the CNC.



The FANUC PPR consists of three units: A printer, paper tape punch, and paper tape reader.

When the PPR is used alone, data can be read from the tape reader and printed or punched out. It is also possible to perform TH and TV checks on data that was read.



2.3.4 FANUC PPR

2.3.5 Portable Tape Reader

The portable tape reader is used to input data from paper tape.



)(

2.4 POWER ON/OFF

2.4.1 Turning on the Power

| | Procedure of turnin | ng on the power | | | |
|-----------|---------------------|---|--|--|--|
| Procedure | 1 | Check that the appearance of the CNC machine tool is normal. (For example, check that front door and rear door are closed.) | | | |
| | 2 | Turn on the power according to the manual issued by the machine tool builder. | | | |
| | 3 | After the power is turned on, check that the position screen is displayed. If the software configuration screen is being displayed, a system failure may have occurred. | | | |
| | | ACTUAL POSITION (ABSOLUTE) O0010 N0010 | | | |
| | | X 123.456 | | | |
| | | Y 363.233 | | | |
| | | Z 0.000 | | | |
| | | PART COUNT 16 RUN TIME 0H 57M CYCLE TIME 0H 0M38S ACT.F 3000 MM/M BUF AUTO | | | |

4 Check that the fan motor is rotating.

ABS) (REL) (ALL) (

WARNING

When pressing the <POWER ON> key, do not touch any other CRT/MDI or thin type display/MDI panel keys, until the positional or alarm screen is displayed. Some keys are used for the maintenance or special operation purpose. When they are pressed, unexpected operation may be caused.

2.4.2 Display of Software Configuration



2.4.3 Power Disconnection

Procedure

- 1 Check that the LED indicating the cycle start is off on the operator's panel.
- 2 Check that all movable parts of the CNC machine tool is stopping.
- **3** If an external input/output device such as the Handy File is connected to the CNC, turn off the external input/output device.
- 4 Continue to press the POWER OFF pushbutton for about 5 seconds.

NOTE

Refer to the machine tool builder's manual for turning off the power to the machine.

3 ∎

MANUAL OPERATION

MANUAL OPERATION are five kinds as follows :

- 1. Manual reference position return
- 2. Jog feed
- 3. Incremental feed
- 4. Manual handle feed
- 5. Manual absolute on/off

3.1 MANUAL REFERENCE POSITION RETURN

The tool is returned to the reference position as follows :

The tool is moved in the direction specified in parameter (bit 0 to #3 of No. 003) for each axis with the reference position return switch on the machine operator's panel. The tool moves to the deceleration point at the rapid traverse rate, then moves to the reference position at the FL speed. The rapid traverse rate and FL speed are specified in parameters (No. 518 to 521, 533, and 534).

Fourstep rapid traverse override is effective during rapid traverse.

When the tool has returned to the reference position, the reference position return completion LED goes on. The tool generally moves along only a single axis, but can move along three axes simultaneously when specified so in parameter (bit 4 of No.049).



Procedure for Manual Reference Position Return

| MODE EDIT MEMORI MOI HANDLE JOG CERO JOG CERO LECH |
|--|
| RAPID TRAVERSE OVERRIDE (%) |

JL

JL

| | rion ₁ |
|-----|-------------------|
| | (↓) ↓+Y |
| | |
| ✓-Y | - C |

- **1** Press the reference position return switch, one of the mode selection swithces.
- **2** To decerease the feedrate, press a rapid traverse override switch. When the tool has returned to the reference position, the reference position return completion LED goes on.
- **3** Press the feed axis and direction selection switch corresponding to the axis and direction for reference position return. Continue pressing the switch until the tool returns to the reference position. The tool can be

moved along three axes simultaneously when specified so in an appropriate parameter setting. The tool moves to the deceleration point at the rapid traverse rate, then moves to the reference position at the FL speed set in a parameter.

4 Perform the same operations for other axes, if necessary. The above is an example. Refer to the appropriate manual provided by the machine tool builder for the actual operations.

| ZEF | 20 | | | | | | | |
|--------------------------|-------------|----------------|-----|--------------|---|-------|-------|------|
| PO: | SITION | J I | | | N | MIRRR | OR IN | IAGE |
| ХҮ | Z | С | | | | Х | Y | Z |
| | | | | | | | | |
| PROGRAM M02/ STOP M30 | MANU ABS | SPINDLE ORI | TAP | ATC READY | | | NC? | MC? |
| | | | | | | | | |

Explanations

Restrictions

| Automatic coordinate system setting | If the parameter for automatic coordinate system setting (bit 7 of parameter 010) is specified, the coordinate system is determined automatically when a manual reference position return is made. If α , β , γ , and δ are specified in parameters 708 to 711, the system specifies a workpiece coordinate system such that the tip of the basic tool or a reference position on the tool holder, after reference position return, becomes $X = \alpha$, $Y = \beta$, $Z = \gamma$, and 4 th = δ . |
|---|--|
| | |

| Moving the tool again | Once the REFERENCE POSITION RETURN COMPLETION LED lights at the completion of reference position return, the tool does not move unless the REFERENCE POSITION RETURN switch is turned off. |
|--|---|
| Reference position return completion LED | The REFERENCE POSITION RETURN COMPLETION LED is extinguished by either of the following operations: Moving from the reference position. Entering an emergency stop state. |

• The distance to return to reference position For the distance (Not in the deceleration condition) to return the tool to the reference position, refer to the manual issued by the machine tool builder.

3.2 JOG FEED

| MODE | 7 |
|------|--------|
| | ן |
| | ה ר |
| | J |

In the jog mode, pressing a feed axis and direction selection switch on the machine operator's panel continuously moves the tool along the selected axis in the selected direction.

The jog feedrate is specified in Table 3.2.

| Table 3 | 3.2 、 | Jog F | eed | ate |
|---------|-------|-------|-----|-----|
|---------|-------|-------|-----|-----|

| Rotary | Feed | rate | Rotary | Feedrate | | |
|----------|--------------------------|--------------------------|----------|--------------------------|--------------------------|--|
| position | Metric input (mm/min) | Inch input (inch/min) | position | Metric input (mm/min) | Inch input (inch/min) | |
| 0 | 0 | 0 | 8 | 50 | 2.0 | |
| 1 | 2.0 | 0.08 | 9 | 79 | 3.0 | |
| 2 | 3.2 | 0.12 | 10 | 126 | 5.0 | |
| 3 | 5.0 | 0.2 | 11 | 200 | 8.0 | |
| 4 | 7.9 | 0.3 | 12 | 320 | 12 | |
| 5 | 12.6 | 0.5 | 13 | 500 | 20 | |
| 6 | 20 | 0.8 | 14 | 790 | 30 | |
| 7 | 32 | 1.2 | 15 | 1260 | 50 | |

NOTE

The feedrate error (about +3%) affects on the feedrate in the table above.

The jog feedrate can be adjusted with the jog feedrate override dial. Pressing the rapid traverse switch moves the tool at the rapid traverse feedrate regardless of the postiotion of the jog feedrate override dial. Manual operation is allowed for one axis at a time. 3 axes can be selected at a time by parameter JAX (No.049#4).



Procedure for Jog Feed





JOG FEED RATE OVERRIDE



Restrictions

- Acceleration/deceleration for rapid traverse
- Change of modes
- Rapid traverse prior to reference position return

- 1 Press the jog switch, one of the mode selection switches.
- 2 Press the feed axis and direction selection switch corresponding to the axis and direction the tool is to be moved. While the switch is pressed, the tool moves at the feedrate specified in Table 3.2. The tool stops when the switch is released.
- **3** The jog feedrate can be adjusted with the jog feedrate override dial.
- 4 Pressing the rapid traverse switch while pressing a feed axis and direction selection switch moves the tool at the rapid traverse rate while the rapid traverse switch is pressed. Rapid traverse override by the rapid traverse override switches is effective during rapid traverse.

The above is an example. Refer to the appropriate manual provided by the machine tool builder for the actual operations.

- Feedrate, time constant and method of automatic acceleration/ deceleration for manual rapid traverse are the same as G00 in programmed command.
- Changing the mode to the jog mode while pressing a feed axis and direction selection switch does not enable jog feed. To enable jog feed, enter the jog mode first, then press a feed axis and direction selection switch.

If reference position return is not performed after power–on, pushing RAPID TRAVERSE button does not actuate the rapid traverse but the remains at the JOG feedrate. This function can be disabled by setting parameter (No.010 #0).

3.3 INCREMENTAL FEED

In the incremental (STEP) mode, pressing a feed axis and direction selection switch on the machine operator's panel moves the tool one step along the selected axis in the selected direction. The minimum distance the tool is moved is the least input increment. Each step can be 10, 100, or 1000 times the least input increment.

This mode is effective when a manual pulse generator is not connected.



Procedure for Incremental Feed





- **1** Press the STEP switch, one of the mode selection switches.
- 2 Select the distance to be moved for each step with the magnification dial.
- **3** Press the feed axis and direction selection switch corresponding to the axis and direction the tool is to be moved. Each time a switch is pressed, the tool moves one step. The feedrate is the same as the jog feedrate.
- **4** If an axis direction selection switch is pressed after rapid traverse has been selected, movement is made at the rapid traverse rate. Rapid traverse override by the rapid traverse override switch is effective during rapid traverse.

The above is an example. Refer to the appropriate manual provided by the machine tool builder for the actual operations.

3.4 MANUAL HANDLE FEED

In the handle mode, the tool can be minutely moved by rotating the manual pulse generator on the machine operator's panel. Select the axis along which the tool is to be moved with the handle feed axis selection switches.

The minimum distance the tool is moved when the manual pulse generator is rotated by one graduation is equal to the least input increment. Or the distance the tool is moved when the manual pulse generator is rotated by one graduation can be magnified by 10 times or by one of the two magnifications specified by parameters (No. 121 and 699).



Procedure for Manual Handle Feed

| | — мо | DE | |
|--------|------|---------------------|------------|
| | | O REMOTE | |
| HANDLE | | O ZERO RETURN | O TEACH |



Manual pulse generator

- **1** Press the HANDLE switch, one of the mode selection switches.
- 2 Select the axis along which the tool is to be moved by pressing a handle feed axis selection switch.
- **3** Select the magnification for the distance the tool is to be moved by pressing a handle feed magnification switch. The minimum distance the tool is moved when the manual pulse generator is rotated by one graduation is equal to the least input increment.
- 4 Move the tool along the selected axis by rotating the handle. Rotating the handle 360 degrees moves the tool the distance equivalent to 100 graduations. The above is an example. Refer to the appropriate manual

provided by the machine tool builder for the actual operations.

Explanations

| Availability of manual | Parameter (bit 0 of No. 013) enables or disables the manual handle feed |
|--|---|
| pulse generator in Jog | in the JOG mode. |
| mode | When the parameter (bit 0 of No. 013) is set 1, both manual handle feed |
| | and incremental feed are enabled. |

- Availability of manual pulse generator in TEACH IN JOG mode
- A command to the MPG exceeding rapid traverse rate

Parameter (bit 6 of No. 002) enables or disables the manual handle feed in the TEACH IN JOG mode.

Parameter (bit 4 of No. 060) specifies as follows:

SET VALUE0: The feedrate is clamped at the rapid traverse rate andgenerated pulses exceeding the rapid traverse rate are ignored.(The distance the tool is moved may not match the graduations on the manual pulse generator.)

SET VALUE1: The feedrate is clamped at the rapid traverse rate and generated pulses exceeding the rapid traverse rate are not ignored but accumulated in the CNC. (No longer rotating the handle does not immediately stop the tool. The tool is moved by the pulses accumulated in the CNC before it stops.)

WARNING

Rotating the handle quickly with a large magnification such as x100 moves the tool too fast. The feedrate is clamped at the rapid traverse feedrate.

NOTE

Rotate the manual pulse generator at a rate of five rotations per second or lower. If the manual pulse generator is rotated at a rate higher than five rotations per second, the tool may not stop immediately after the handle is no longer rotated or the distance the tool moves may not match the graduations on the manual pulse generator.

3.5 MANUAL ABSOLUTE ON AND OFF

Whether the distance the tool is moved by manual operation is added to the coordinates can be selected by turning the manual absolute switch on or off on the machine operator's panel. When the switch is turned on, the distance the tool is moved by manual operation is added to the coordinates. When the switch is turned off, the distance the tool is moved by manual operation is not added to the coordinates.



Fig. 3.5(a) Coordinates with the switch ON



Fig. 3.5(b) Coordinates with the switch OFF

Explanation

The following describes the relation between manual operation and coordinates when the manual absolute switch is turned on or off, using a program example.

| G01G90 | X100.0Y100.0F | 010; | (1) |
|--------|---------------|------|-----|
| | X200.0Y150.0 | ; | (2) |
| | X300.0Y200.0 | ; | (3) |
| | | | |

The subsequent figures use the following notation:

Movement of the tool when the switch is on ➤ Movement of the tool when the switch is off

The coordinates after manual operation include the distance the tool is moved by the manual operation. When the switch is off, therefore, subtract the distance the tool is moved by the manual operation.

 Manual operation after the end of block

Coordinates when block (2) has been executed after manual operation (X-axis +20.0, Y-axis +100.0) at the end of movement of block (1).



 Manual operation after a feed hold

Coordinates when the feed hold button is pressed while block (2) is being executed, manual operation (Y-axis + 75.0) is performed, and the cycle start button is pressed and released



- 298 ----

 When reset after a manual operation following a feed hold Coordinates when the feed hold button is pressed while block (2) is being executed, manual operation (Y-axis +75.0) is performed, the control unit is reset with the RESET button, and block (2) is read again



When there is only one axis in the following command, only the commanded axis returns.



When the following commands are incremental commands, operation is the same as when the switch is OFF.

When the switch is OFF

After manual operation is performed with the switch OFF during cutter compensation, automatic operation is restarted then the tool moves parallel to the movement that would have been performed if manual movement had not been performed. The amount of separation equals to the amount that was performed manually.



 When a movement command in the next block is only one axis

- When the next move block is an incremental
- Manual operation during cutter compensation

When the switch is ON during cutter compensation

Operation of the machine upon return to automatic operation after manual intervention with the switch is ON during execution with an absolute command program in the cutter compensation mode will be described. The vector created from the remaining part of the current block and the beginning of the next block is shifted in parallel. A new vector is created based on the next block, the block following the next block and the amount of manual movement. This also applies when manual operation is performed during cornering.

Manual operation performed in other than cornering

Assume that the feed hold was applied at point P_H while moving from P_A to P_B of programmed path P_A , P_B , and P_C and that the tool was manually moved to $P_{H'}$. The block end point P_B moves to the point $P_{B'}$ by the amount of manual movement, and vectors V_{B1} and V_{B2} at P_B also move to $V_{B1'}$ and $V_{B2'}$. Vectors V_{C1} and V_{C2} between the next two blocks $P_B - P_C$ and $P_C - P_D$ are discarded and new vectors $V_{C1'}$ and $V_{C2'}$ ($V_{C2'} = V_{C2}$ in this example) are produced from the relation between $P_B' - P_C$ and $P_C - P_D$. However, since $V_{B2'}$ is not a newly calculated vector, correct offset is not performed at block $P_{B'} - P_C$. Offset is correctly performed after P_C .



Manual operation during cornering

This is an example when manual operation is performed during cornering. $V_{A2'}$, $V_{B1'}$, and $V_{B2'}$ are vectors moved in parallel with V_{A2} , V_{B1} and V_{B2} by the amount of manual movement. The new vectors are calculated from V_{C1} and V_{C2} . Then correct cutter compensation is performed for the blocks following Pc.



Manual operation after single block stop

Manual operation was performed when execution of a block was terminated by single block stop.

Vectors V_{B1} and V_{B2} are shifted by the amount of manual operation. Sub–sequent processing is the same as case a described above. An MDI operation can also be interveneted as well as manual operation. The movement is the same as that by manual operation.





AUTOMATIC OPERATION

Programmed operation of a CNC machine tool is referred to as automatic operation.

This chapter explains the following types of automatic operation:

- MEMORY OPERATION
- Operation by executing a program registered in CNC memory
- MDI OPERATION

•

- Operation by executing a program entered from the MDI panel
- **DNC OPERATION** Function for operating a machine while reading a program from an input/output unit
- **MIRROR IMAGE** Function for enabling mirror–image movement along an axis during automatic operation
- SEQUENCE NUMBER SEARCH
 - This function is usually used to search for a sequence number within a program and to start or resume the program from the block having that sequence number.

| 4.1 MEMORY OPERATION | Programs are registered in memory in advance. When one of these programs is selected and the cycle start switch on the machine operator's panel is pressed, automatic operation starts, and the cycle start lamp goes on. When the feed hold switch on the machine operator's panel is pressed during automatic operation, automatic operation is stopped temporarily. When the cycle start switch is pressed again, automatic operation is restarted. When the reset switch on the CRT/MDI panel is pressed, automatic operation terminates and the reset state is entered. The following procedure is given as an example. For actual operation, refer to the manual supplied by the machine tool builder. | | |
|----------------------------|--|--|--|
| Procedure for | Memory Operation | | |
| Procedure | Press the AUTO mode selection switch. Select a program from the registered programs. To do this, follow the steps below. 2-1 Press mean to display the program screen. 2-2 Press address O. 3 Press the cursor key J. Press the cycle start switch on the machine operator's panel. Automatic operation starts, and the cycle start lamp goes on. When automatic operation terminates, the cycle start lamp goes off. To stop or cancel memory operation midway through, follow the steps below. Stopping memory operation Press the feed hold switch on the machine operator's panel. The feed hold lamp goes on and the cycle start lamp goes off. The machine responds as follows: (i) When the machine was moving, feed operation decelerates and stops. (ii) When dwell was being performed, dwell is stopped. (iii) The current operation, instigated by executing by an M, S, or T command, is continued. When the cycle start switch on the machine operator's panel is pressed while the feed hold lamp is on, machine operation restarts. Terminating memory operation | | |

| Explanation | |
|--|---|
| Memory operation | After memory operation is started, the following are executed: A one-block command is read from the specified program. The block command is decoded. The command execution is started. The command in the next block is read. (5) Buffering is executed. That is, the command is decoded to allow immediate execution. (6) Immediately after the preceding block is executed, execution of the next block can be started. This is because buffering has been executed. (7) Hereafter, memory operation can be executed by repeating the steps (4) to.(6) |
| Stopping and terminating memory operation | Memory operation can be stopped using one of two methods: Specify a stop command, or press a key on the machine operator's panel. The stop commands include M00 (program stop), M01 (optional stop), and M02 and M30 (program end). There are two keys to stop memory operation: The feed hold key and reset key. |
| Program stop (M00) | Memory operation is stopped after a block containing M00 is executed. When the program is stopped, all existing modal information remains unchanged as in single block operation. The memory operation can be restarted by pressing the cycle start button. Operation may vary depending on the machine tool builder. Refer to the manual supplied by the machine tool builder. |
| Optional stop (M01) | Similarly to M00, memory operation is stopped after a block containing M01 is executed. This code is only effective when the Optional Stop switch on the machine operator's panel is set to ON. Operation may vary depending on the machine tool builder. Refer to the manual supplied by the machine tool builder. |
| Program end (M02, M30) | When M02 or M30 (specified at the end of the main program) is read, memory operation is terminated and the reset state is entered.In some machines, M30 returns control to the top of the program. For details, refer to the manual supplied by the machine tool builder. |
| Feed hold | When Feed Hold button on the operator's panel is pressed during memory operation, the tool decelerates to a stop at a time. |
| • Reset | Automatic operation can be stopped and the system can be made to the reset state by using \bigcirc RESET key on the CRT/MDI panel or external reset signal. When reset operation is applied to the system during a tool moving status, the motion is slowed down then stops. |
| Optional block skip | When the optional block skip switch on the machine operator's panel is turned on, blocks containing a slash (/) are ignored. |

| 4.2 | In the MDI mode, a program can be inputted in the same format as normal programs and executed from the MDI panel. |
|---------------|---|
| MDI OPERATION | MDI operation is used for simple test operations. |
| | The following procedure is given as an example. For actual operation, refer to the manual supplied by the machine tool builder. |

Procedure for MDI Operation – A

Example of X10.5 Y200.5;

One command block can be entered from the CRT/MDI for execution.

- **1** Press MDI key on the mode select switch.
- 2 Press the PRGRM button.
- **3** Press soft key **[MDI]** to display a screen with MDI at the top left.

| | 00501 20501 |
|-------------------|------------------|
| PROGRAM | 09501 N9501 |
| (MDI) | (MODAL) |
| | G67 G00 F |
| | G97 G17 R |
| | G54 G90 P |
| | G64 G22 Q |
| | G69 G94 H D |
| | G15 G21 M |
| | G25 G40 S |
| | G49 T |
| | G80 B |
| | G98 |
| | G50 |
| | |
| ADRS. | |
| | MDI |
| [PRGRM][CURRNT | '][NEXT][MDI][]] |
| ▲ 1 = 1 | |

- 4 Input "X 10.5" by address/numeric key.
- 5 Press INPUT key.

The data, X 10.5, is input and displayed. If you are aware of an error in the keyed–in number before pressing the key, press

the |CAN| key and key in X and the correct number again.

- 6 Input "Y 200.5" by address/numeric key.
- 7 Press INPUT key.

The data, Y200.5 is input and displayed. If you pressed wrong number keys, correct the operation following the instruction described above.

Cancel before pressing the

START button

| | | | | | | | | | | | _ |
|---|---------|----|--------|-----|------|-------|-----|-------|---|------|---|
| / | | | | | | | | | | | |
| | PROGRAM | | | | | | C | 09501 | N | 9501 | |
| | (MDI) | | | | (| (MOD2 | AL) | | | | |
| | | Х | : 10. | 500 | G67 | G00 | F | | | | |
| | | Y | 200. | 500 | G54 | G17 | R | | | | |
| | | | | | G64 | G90 | Ρ | | | | |
| | | | | | G69 | G22 | Q | | | | |
| | | | | | G15 | G94 | н | | | | |
| | | | | | G25 | G21 | М | | | | |
| | | | | | | G40 | s | | | | |
| | | | | | | G49 | т | | | | |
| | | | | | | G80 | в | | | | |
| | | | | | | G98 | | | | | |
| | | | | | | G50 | | | | | |
| | ADRS. | | | | | | | | | | |
| | | | | | | MD | Ľ | | | | |
| | [PRGRM |][| CURRNT |][| NEXT |] [| MDI | [][| |] | |
| | | | | | | | | | | | |

8 Press the $\begin{bmatrix} OUTPT \\ START \end{bmatrix}$ key.

Press the cycle start button on the machine operator's panel (depending on the machine tool).

To modify X10.5Y200.5 to X10.5, cancel Y200.5, by following the steps described below:

- 1 Press Y CAN INPUT keys.
- 2 Press the $\begin{bmatrix} OUTPT \\ START \end{bmatrix}$ or the cycle start button on the machine operator's panel.

WARNING

Modal G codes cannot be cancelled. Enter the correct data again.

Limitations

- A single MDI operation executes a single input block. Two or more blocks cannot be executed at one time.
- · The end–of–block symbol (;) need not be entered.
- A subprogram call or macro call cannot be specified.
- In MDI operation, the setting (absolute) determines whether the commands are absolute or incremental. The G code (G90/G91) is ignored.
- The input block is cleared when the MDI operation is completed or when a reset is specified.

| 4.3 DNC OPERATION | In DNC operation, the machine is not operated by a program registered in memory of the CNC, instead being operated by a program read directly from a connected input/output unit. This mode is used when the program is too large to be registered in the memory of the CNC. | | | |
|----------------------|---|--|--|--|
| | Procedure for DNC Operation | | | |
| | 1 Select the MDI mode and specify the channel for the connected input/output unit in the I/O field on the setting screen. | | | |
| | 2 Select AUTO mode. | | | |
| | 3 Search for the beginning of the program of the input/output unit. | | | |
| | 4 Input the DNCI signal. (For details of the actual operation, refer to the manual provided by the machine tool builder.) | | | |
| | 5 Press the cycle start key. | | | |
| | DNC operation starts. The operation can be stopped and resumed in the same way as for memory operation. | | | |
| Explanations | | | | |
| | • In DNC operation mode, the current program can call a subprogram registered in memory. | | | |
| | • In DNC operation mode, the current program can specify a custom macro. However, a repeat instruction or branch instruction cannot be programmed. | | | |
| | • In DNC operation mode, the current program can call a macro program registered in memory. | | | |
| | • In DNC operation mode, to return the system from the current subprogram or macro program to the calling program, a sequence number (M99P****) cannot be specified. | | | |
| | • The DC3 code is output and reading is stopped at the end of each block (each time the EOB is read). To enable continuous reading of the blocks, specify bit 7 of parameter 0390 accordingly. | | | |
| | • In DNC operation mode, no program can be displayed. Only the current block and subsequent block can be displayed. | | | |
| | • In DNC operation mode, F is displayed as the address of the program number at the top right corner of the CRT screen. | | | |
| | | | | |

4.4 MIRROR IMAGE

During automatic operation, the mirror image function can be used for movement along an axis. To use this function, set the mirror image switch to ON on the machine operator's panel, or set the mirror image setting to ON from the CRT/MDI panel.



Fig 4.4 (a) Mirror Image

The following procedure is given as an example. For actual operation, refer to the manual supplied by the machine tool builder.

- 1 Press the single block switch to stop automatic operation. When the mirror image function is used from the begining of operation, this step is omitted.
- **2** Press the mirror image switch for the target axis on the machine operator's panel.

Alternatively, turn on the mirror image setting by following the steps below:

- 2–1 Set the **MDI** mode.
- **2–2** Press the $\begin{bmatrix} D_{GNOS} \\ PARAM \end{bmatrix}$ function key.
- **2–3** Press the page key for chapter selection to display the setting screen.

```
PARAMETER
                                   00100 N0002
    (SETTING 1)
    _REVX
              = 0
    REVY
              = 0
    TVON
                0
                              1:ISO
    ISO
              = 1
                    (0:EIA
                                     )
    INCH
                    (0:MM
                              1:INCH)
              = 0
    I/O
              = 0
    ABS
              = 0
                    (0:INC
                              1:ABS)
              = 0
    SEQ
NO. REVX =
                              MDI
[ PARAM ][ DGNOS ][
                           ][ SV-PRM ][
                                               1
```

Procedure

| | | 2–4 | Move the cursor to the mirror image setting position, then set the target axis to 1. |
|--------------|----------------------------|--|---|
| | 3 | Enter an automatic operation mode (AUTO mode or MDI mo press the cycle start button to start automatic operation. | |
| Explanations | ·Fo | or the m machin | nirror image switches, refer to the manual supplied by the e tool builder. |
| Restrictions | Th mo au un cy | ne direct ovemen tomatic idirecti cle (G7 | tion of movement during manual operation, the direction of t from an intemidiate point to the reference position during reference position return, the direction of approach during onal positioning (G60), and the shift direction in a boring 6, G87) cannot be reserved. |
4.5 SEQUENCE NUMBER SEARCH

Sequence number search operation is usually used to search for a sequence number in the middle of a program so that execution can be started or restarted at the block of the sequence number.

Example)Sequence number 2346 in a program (O0002) is searched for.



Procedure for sequence number search

| | 1 Select AUTO mode. |
|---------------------------|---|
| | 2 Press Prgrm key. |
| | 3 If the program contains a sequence number to be searched for, perform the operations 4 to 7 below. If the program does not contain a sequence number to be searched for, select the program number of the program that contains the sequence number to be searched for. |
| | 4 Key in address \mathbb{N} . |
| | 5 Key in a sequence number to be searched for. |
| | 6 Press the cursor key. |
| | 7 Upon completion of search operation, the sequence number searched for is displayed in the upper–right corner of the CRT screen. |
| Explanations | |
| • Operation during Search | Those blocks that are skipped do not affect the CNC. This means that the data in the skipped blocks such as coordinates and M, S, and T codes does not alter the CNC coordinates and modal values. So, in the first block where execution is to be started or restarted by using a sequence number search command, be sure to enter required M, S, and T codes and coordinates. A block searched for by sequence number search usually represents a point of shifting from one process to another. When a block in the middle of a process must be searched for to restart execution at the block, specify M, S, and T codes, coordinates, and so forth as required from the MDI after closely checking the machine tool and NC states at that point. |

• Checking during search

During search operation, the following checks are made : •Optional block skip •P/S alarm (No. 003 to 010) A P/S alarm 003 to 010 occurring during the sequence number search can be ignored if bit 1 of parameter 050 is specified accordingly.

Limitations

 Searching in sub–program During sequence number search operation, M98Pxxxx (subprogram call) is not executed. So an alarm (No. 060) is raised if an attempt is made to search for a sequence number in a subprogram called by the program currently selected.



Alarm

| Number | Contents |
|--------|---|
| 60 | Command sequence number was not found in the sequence |
| | number search. |

5 TEST OPERATION

The following functions are used to check before actual machining whether the machine operates as specified by the created program.

- 1. Machine Lock and Auxiliary Function Lock
- 2. Feedrate Override
- 3. Rapid Traverse Override
- 4. Dry Run
- 5. Single Block

5.1 **MACHINE LOCK AND AUXILIARY FUNCTION** LOCK

To display the change in the position without moving the tool, use machine lock.

There are two types of machine lock: all-axis machine lock, which stops the movement along all axes, and Z-axis machine lock, which stops the movement along Z-axis only. In addition, auxiliary function lock, which disables M, S, and T commands, is available for checking a program together with machine lock.



Procedure for Machine Lock and Auxiliary Function Lock

| • Machine Lock | Press the machine lock switch on the operator's panel. The tool does not move but the position along each axis changes on the display as if the tool were moving. Some machines have a Z-axis machine lock switch. Refer to the appropriate manual provided by the machine tool builder for machine lock. The position relationship between the workpiece coordinates and machine coordinates may change after automatic operation by the machine lock function has been executed. If this occurs, reset the workpiece coordinate system by specifying the coordinate system setting command or by making a manual reference position return. |
|--|---|
| Auxiliary Function Lock | Press the auxiliary function lock switch on the operator's panel. M, S, and T codes are disabled and not executed. Refer to the appropriate manual provided by the machine tool builder for auxiliary function lock. |
| Restrictions | |
| M, S, T command by only machine lock | M, S, and T commands are executed in the machine lock state. |
| Reference position return under Machine Lock | When a G27, G28, or G30 command is issued in the machine lock state, the command is accepted but the tool does not move to the reference position and the reference position return LED does not go on. |
| M codes not locked by auxiliary function lock | M00, M01, M02, M30, M98, and M99 and commands are executed even in the auxiliary function lock state. |

5.2 FEEDRATE OVERRIDE

A programmed feedrate can be reduced or increased by a percentage (%) selected by the override dial. This feature is used to check a program. For example, when a feedrate of 100 mm/min is specified in the program, setting the override dial to 50% moves the tool at 50 mm/min.



Fig. 5.2 Feedrate override

Procedure for Feedrate Override



JOG FEED RATE OVERRIDE

Restrictions

• Override Range

Set the feedrate override dial to the desired percentage (%) on the machine operator's panel, before or during automatic operation. On some machines, the same dial is used for the feedrate override dial and jog feedrate dial. Refer to the appropriate manual provided by the machine tool builder for feedrate override.

The override that can be specified ranges from 0 to 150% (10% steps). For individual machines, the range depends on the specifications of the machine tool builder.

5.3 RAPID TRAVERSE OVERRIDE

An override of four steps (F0, 25%, 50%, and 100%) can be applied to the rapid traverse rate. F0 is set by a parameter (No. 533).



Fig. 5.3 Rapid traverse override

Rapid Traverse Override



Select one of the four feedrates with the rapid traverse override switch during rapid traverse. Refer to the appropriate manual provided by the machine tool builder for rapid traverse override.

Explanation

The following types of rapid traverse are available. Rapid traverse override can be applied for each of them.

- 1) Rapid traverse by G00
- 2) Rapid traverse during a canned cycle
- 3) Rapid traverse in G27, G28 and G30
- 4) Manual rapid traverse
- 5) Rapid traverse in manual reference position return

5.4 DRY RUN

The tool is moved at the feedrate specified by a parameter regardless of the feedrate specified in the program. This function is used for checking the movement of the tool under the state taht the workpiece is removed from the table.





Procedure for Dry Run

Press the dry run switch on the machine operator's panel during automatic operation.

The tool moves at the feedrate specified in a parameter. The rapid traverse switch can also be used for changing the feedrate. Refer to the appropriate manual provided by the machine tool builder for dry run.

Explanation

• Dry run feedrate



The dry run feedrate changes as shown in the table below according to the rapid traverse switch and parameters.

| Rapid tra- | Program command | | | | | |
|--------------|--|----------------------|--|--|--|--|
| verse button | Rapid traverse | Feed | | | | |
| ON | Rapid traverse rate | Jog maximum feedrate | | | | |
| OFF | Jog feedrate or rapid traverse rate *1) | Jog feedrate | | | | |

*1: Jog feedrate when parameter (bit 6 of No. 001) is 1. Rapid traverse rate when that parameter is 0.

— 316 —

5.5 SINGLE BLOCK

Pressing the single block switch starts the single block mode. When the cycle start button is pressed in the single block mode, the tool stops after a single block in the program is executed. Check the program in the single block mode by executing the program block by block.



Procedure for Single block

Procedure

1 Press the single block switch on the machine operator's panel. The execution of the program is stopped after the current block is executed.

2 Press the cycle start button to execute the next block. The tool stops after the block is executed.

Refer to the appropriate manual provided by the machine tool builder for single block execution.

Explanation

• Reference position If G28 to G30 are issued, the single block function is effective at the intermediate point.

• Single block during a canned cycle

In a canned cycle, the single block stop points are the end of (1)(2), and (6) shown below. When the single block stop is made after the point (1) or (2), the feed hold LED lights.



Fig.5.5(a) Single block during canned cycle

• Subprogram call and single block

Single block stop is not performed in a block containing M98P_;. M99; or G65.

However, single block stop is even performed in a block with M98P_ or M99 command, if the block contains an address other than O, N or P.



SAFETY FUNCTIONS

To immediately stop the machine for safety, press the Emergency stop button. To prevent the tool from exceeding the stroke ends, Overtravel check and Stroke check are available. This chapter describes emergency stop., overtravel check, and stroke check.

6.1 EMERGENCY STOP

If you press Emergency Stop button on the machine operator's panel, the machine movement stops in a moment.



Fig. 6.1 Emergency stop

This button is locked when it is pressed. Although it varies with the machine tool builder, the button can usually be unlocked by twisting it.

Explanation

EMERGENCY STOP interrupts the current to the motor. Causes of trouble must be removed before the button is released.

6.2 OVERTRAVEL

When the tool tries to move beyond the stroke end set by the machine tool limit switch, the tool decelerates and stops because of working the limit switch and an OVER TRAVEL is displayed.



Fig. 6.2 Overtravel

Explanation

- Overtravel during automatic operation
- Overtravel during manual operation
- Releasing overtravel
- Alarm

When the tool touches a limit switch along an axis during automatic operation, the tool is decelerated and stopped along all axes and an overtravel alarm is displayed.

In manual operation, the tool is decelerated and stopped only along the axis for which the tool has touched a limit switch. The tool still moves along the other axes.

Press the reset button to reset the alarm after moving the tool to the safety direction by manual operation. For details on operation, refer to the operator's manual of the machine tool builder.

| No. | Description |
|-----|---|
| 5n6 | The tool has exceeded the hardware–specified overtravel limit along the positive nth axis (n : 1 to 3). |
| 5n7 | The tool has exceeded the hardware–specified overtravel limit along the negative nth axis (n : 1 to 3). |

Setting bit 5 of parameter 0057 enables or disables the overtravel limit switch. For details, refer to the manual provided by the machine tool builder.

6.3 STROKE CHECK

Two areas which the tool cannot enter can be specified with stored stroke limit 1, 2 and stored stroke limit 3.



Fig. 6.3(a) Stroke check

When the tool exceeds a stored stroke limit, an alarm is displayed and the tool is decelerated and stopped.

When the tool enters a forbidden area and an alarm is generated, the tool can be moved in the reverse direction from which the tool came.

Explanation

- Stored stroke limit 1 and 2
- Checkpoint for the forbidden area

Parameters (Nos. 700 to 707 or Nos. 743 to 750) set boundary. Outside the area of the set limits is a forbidden area. The machine tool builder usually sets this area as the maximum stroke.

Confirm the checking position (the top of the tool or the tool chuck) before programming the forbidden area.

If point A (The top of the tool) is checked in Fig. 6.3 (b), the distance "a" should be set as the data for the stored stroke limit function. If point B (The tool chuck) is checked, the distance "b" must be set. When checking the tool tip (like point A), and if the tool length varies for each tool, setting the forbidden area for the longest tool requires no re–setting and results in safe operation.



Fig. 6.3(b) Setting the forbidden area

— 322 —

 Overrun amount of stored stroke limit

• Effective time for a forbidden area

If the maximum rapid traverse rate is F mm/minute, the maximum overrun amount, L mm, of the stored stroke limit is obtained from the following expression:

L mm = F/7500

The tool enters the specified inhibited area by up to L mm. The tool can be stopped at a position up to L mm in front of the inhibited area if bit 7 of parameter 0076 is specified accordingly. In this case, the tool will not enter the inhibited area.

Bit 3 of parameter 065 is used to select whether each stroke limit is enabled, either immediately after power–on or later, after a manual reference position return or reference position return by G28.

After the power is turned on, if the reference position is in the forbidden area of each limit, an alarm is generated immediately.

• **Releasing the alarms** If a stroke check alarm occurs, manually retract the tool from the inhibited area in the direction opposite to the displayed alarm direction. Press the reset key to cancel the alarm.

NOTE

In setting a forbidden area, if the two points to be set are the same, the area is as follows:

- 1 When the forbidden area is limit 1, 2, all areas are forbidden areas.
- 2 When the forbidden area is limit 3, all areas are movable areas.

Alarms

| Number | Message | Contents | | | | |
|--------|-----------------|---|--|--|--|--|
| 5n0 | OVER TRAVEL: +n | Exceeded the n-th axis $(1-8)$ + side stored stroke limit 1 or 2. | | | | |
| 5n1 | OVER TRAVEL:n | Exceeded the n-th axis (1-8) - side stored stroke limit 1 or 2. | | | | |



ALARM AND SELF-DIAGNOSIS FUNCTIONS

When an alarm occurs, the corresponding alarm screen appears to indicate the cause of the alarm. The causes of alarms are classified by error codes. The system may sometimes seem to be at a halt, although no alarm is displayed. In this case, the system may be performing some processing. The state of the system can be checked using the self-diagnostic function.

7.1 ALARM DISPLAY

Explanations

• Alarm screen

When an alarm occurs, the alarm screen appears.

| ALARM M | IESSAGE | | | 000 | 00 NOOC | 00 |
|--------------------|----------------------------------|------------------------|--------------|--------------------|----------------|----|
| 100 417 427 | P/S ALAI SERVO AI SERVO AI | RM LARM : LARM : | X AX Y AX | IS DGTL IS DGTL | PARAM PARAM | |
| | | | | | | |
| NOT REA [ALARM | DY ALARM [][OPR |][| MSG | MDI][|][|] |

• Another method for alarm displays

In some cases, the alarm screen may not be displayed. Instead, ALARM will blink at the bottom of the screen.

| PARAMETER | | | | 00100 N | 10002 |
|--------------|-------|---|--------|-------------|-------|
| (SETTING | 1) | | | | |
| REVX | = | 0 | | | |
| REVY | = | 0 | | | |
| TVON | = | 0 | | | |
| ISO | = | 1 | (0:EIA | 1:ISO) | |
| INCH | = | 0 | (0:MM | 1:INCH) | |
| I/O | = | 0 | | | |
| ABS | = | 0 | (0:INC | 1:ABS) | |
| SEQ | = | 0 | | | |
| NO TSO - | | | | | |
| NOT READY AT | .ARN | r | | MDT | |
| | | - | 11 1 | 1 MOG_DDM 1 | 1 |
| [FARAM][I | JGINC | 5 | 11 1 | | L |
| | | | | | |
| | | | | | |

In this case, display the alarm screen as follows:

- **1.** Press the function $\begin{bmatrix} OPR \\ ALARM \end{bmatrix}$ key.
- 2. Press the soft key [ALARM].

| Reset of the alarm | Error codes and messages indicate the cause of an alarm. To recover from an alarm, eliminate the cause and press the reset key. |
|--|--|
| • Error codes | The error codes are classified as follows: No. 000 to 250: Program errors(*1) No. 3n0 to 3n8: Absolute pulse coder (APC) alarms (*2) No. 3n9: Serial pulse coder (SPC) alarms (*2) No. 400 to 495: Servo alarms No. 510 to 581: Overtravel alarms No. 600 to 607: PMC alarms No. 700 to 704: Overheat alarms No. 910 to 988: System alarms |
| | *1) For an alarm (No. 000 to 250) that occurs in association with background operation, the indication "xxxBP/S alarm" is provided (where xxx is an alarm number). Only a BP/S alarm is provided for No. 140. *2) n is controlled axis number. |

See the error code list in the appendix for details of the error codes.

7.2 CHECKING BY SELF-DIAGNOSTIC SCREEN

The system may sometimes seem to be at a halt, although no alarm has occurred. In this case, the system may be performing some processing. The state of the system can be checked by displaying the self-diagnostic screen.

Procedure for Diagnois

- **1** Press the function key DGNOS PARAM
- 2 Press the soft key [DGNOS].
- **3** The diagnostic screen has more than 1 pages. Select the screen by the following operation.
 - (1) Change the page by the 1-page change key.
 - (2) \cdot Press the No. key.
 - Key input the number of the diagnostic data to be displayed.

| PARAMETE | R | | 01224 N0000 |
|----------|-------------|------|-------------|
| NO. | DATA | NO. | DATA |
| 0700 | 00000000 | 0725 | 00000000 |
| 0701 | 00000000 | 0726 | 00000000 |
| 0710 | 00000000 | 0727 | 00000000 |
| 0711 | 00000000 | 0730 | 00000000 |
| 0712 | 00000000 | 0731 | 00000000 |
| 0720 | 00000000 | 0732 | 00000000 |
| 0721 | 00000000 | 0733 | 00000000 |
| 0722 | 00000000 | 0734 | 00000000 |
| 0723 | 00000000 | 0735 | 00000000 |
| 0724 | 0000000 | 0736 | 00000000 |
| NO. 0700 | = | | |
| 04:02:0 | 2 | AU | TO |
| [PARAM |][DGNOS][|][S | V-PRM][|

| | #7 | #6 | #5 | #4 | #3 | #2 | #1 | #0 |
|------|----|------|------|------|------|------|------|------|
| 0700 | | CSCT | CITL | COVZ | CINP | CDWL | CMTN | CFIN |

When a digit is "1", the corresponding status is effective.

- CFIN: The M, S, O, or T function is being executed.
- CMTN : A move command in the cycle operation is being executed.
- CDWL : Dwell is being executed.
- CINP : An in-posiiton check is being executed.
- COVZ: Override is at 0%.
- CITL: Interlock signal (STLK) is turned on.
- CSCT: Speed arrival signal of spindle is turned on.

| | #7 | #6 | #5 | #4 | #3 | #2 | #1 | #0 |
|------|----|----|------|----|----|----|----|----|
| 0701 | | | CRST | | | | | |

CRST: One of the following : The reset button on the MDI panel, emergency stop, or remote reset is on.

| | #7 | #6 | #5 | #4 | #3 | #2 | #1 | #0 |
|------|-----|------|-----|----|------|----|----|-----|
| 0712 | STP | REST | EMS | | RSTB | | | CSU |

Indicates automatic operation stop or feed hold status. These are used for troubleshooting.

- STP : The flag which stops the automatic operation. This is set at the following condition.
 - External reset signal is turned on.
 - Emergency stop signal is turned on.
 - · Feed hold signal is turned on.
 - · Reset button on the CRT/MDI panel is turned on.
 - The mode is changed to the manual mode, such as JOG, HANDLE/STEP, TEACH INJOG, TEACH IN HANDLE.
 - \cdot Other alarm is generated.
- REST : This is set when one of the external reset, emergency stop, or reset button is set on.
- EMS : This is set when the emergency stop is set on.
- RSTB: This is set when the reset button is on.
- CSU: This is set when the emergency stop is turned on, or when the servo alarm has been generated.

8

DATA INPUT/OUTPUT

NC data is transferred between the NC and external input/output devices such as the Handy File.

The following types of data can be entered and output :

- 1.Program
- 2.Offset data
- 3.Parameter
- 4.Pitch error compensation data

5.Custom macro common variable

Before an input/output device can be used, the input/output related parameters must be set.



| 8.1 FILES | Of the external input/output devices, the FANUC Handy File and FANUC Floppy Cassette use floppy disks as their input/output medium, and the FANUC FA Card uses an FA card as its input/output medium. In this manual, an input/output medium is generally referred to as a floppy. However, when the description of one input/output medium varies from the description of another, the name of the input/output medium is used. In the text below, a floppy represents a floppy disk or FA card. Unlike an NC tape, a floppy allows the user to freely choose from several types of data stored on one medium on a file–by–file basis. Input/output is possible with data extending over more than one floppy disk. |
|----------------------------------|--|
| Explanations | |
| • What is a File | The unit of data, which is input/output between the floppy and the CNC by one input/output operation (pressing the [READ] or [PUNCH] key), is called a HfileI. When inputting CNC programs from, or outputting them to the floppy, for example, one or all programs within the CNC memory are handled as one file. Files are assigned automatically file numbers 1,2,3,4 and so on, with the lead file as 1. File 1 File 2 File 3 File n Blank |
| • Request for floppy replacement | When one file has been entered over two floppies, LEDs on the adaptor flash alternately on completion of data input/output between the first floppy and the CNC, prompting floppy replacement. In this case, take the first floppy out of the adaptor and insert a second floppy in its place. Then, data input/output will continue automatically. Floppy replacement is prompted when the second floppy and later is required during file search–out, data input/output between the CNC and the floppy, or file deletion. Floppy 1 File 1 File 2 File 3 File (k–1) File k Floppy 2 Continuation File (k+1) File n Blank Since floppy replacement is processed by the input/output device, no special operation is required. The CNC will interrupt data input/output operation until the next floppy is inserted into the adaptor. |

When reset operation is applied to the CNC during a request for floppy replacement, the CNC is not reset at once, but reset after the floppy has been replaced.

Protect switch

The floppy is provided with the write protect switch. Set the switch to the write enable state. Then, start output operation.



Writing memo

Fig. 8.1. Protect swtich

Once written in the cassette or card, data can subsequently be read out by correspondence between the data contents and file numbers. This correspondence cannot be verified, unless the data contents and file numbers are output to the CNC and displayed.

To display the contents, write the file numbers and the contents on the memo column which is the back of floppy.

| (Entry example on | MEMO) |
|-------------------|------------------|
| File 1 | NC parameters |
| File 2 | Offset data |
| File 3 | NC program O0100 |
| • | • |
| • | • |
| | • |
| File (n–1) | NC program O0500 |
| File n | NC program O0600 |
| | |

| 8.2 FILE SEARCH | When the program is input from the floppy, the file to be input first must be searched. For this purpose, proceed as follows: | |
|-------------------------|---|--|
| | File 1 File 2 File 3 S File n Blank | |
| | File searching of the file n | |
| Procedure for | File heading | |
| Procedure | Press the EDIT or AUTO switch on the machine operator's panel. Press function program key and display the program screen | |
| | 3 Enter address N key. | |
| | 4 Enter the number of the file to search for. N0 The beginning of the cassette or card is searched. One of N1 to N9999 Of the file Nos. 1 to 9999, a designated file is searched. N-9999 The file next to that accessed just before is searched. N-9998 When N-9998 is designated, N-9999 is automatically inserted each time a file is input or output. This condition is reset by the designation of N1,N1 to 9999, or N – 9999 or reset. 5 Press soft key NPUT. The specified file is searched for. | |
| Explanation | | |
| ● File search by N-9999 | The same result is obtained both by sequentially searching the files by specifying Nos. N1 to N9999 and by first searching one of N1 to N9999 and then using the N–9999 searching method. The searching time is shorter in the latter case. | |

| No. | Description |
|-----|--|
| | The ready signal (DR) of an input/output device is off. |
| 86 | An alarm is not immediately indicated in the CNC even when an alarm occurs during head searching (when a file is not found, or the like). |
| | An alarm is given when the input/output operation is performed after that. This alarm is also raised when N1 is specified for writing data to an empty floppy. (In this case, specify N0.) |

8.3 FILE DELETION

Files stored on a floppy can be deleted file by file as required.

| Procedure for | File Deletion |
|---|--|
| Procedure | Insert the floppy into the input/output device so that it is ready for writing. Press the EDIT switch on the machine operator's panel. Press function Provide the program screen. |
| | 4 Enter address N key. 5 Enter the number (from 1 to 9999) of the file to delete. 6 Press soft key OUTPT START . The specified file is deleted. |
| Explanations | |
| File number after the file is deleted | When a file is deleted, the file numbers after the deleted file are each decremented by one. Suppose that a file numbered k was deleted. In this case, files are renumbered as follows: Before deletion \ldots after deletion 1 to (k>1) \ldots 1 to (k>1) k \ldots Deleted (k+1) to n \ldots k to (n>1) |
| Protect switch | Set the write protect switch to the write enable state to delete the files. |

— 333 —

8.4 PROGRAM INPUT/OUTPUT

8.4.1 This section describes how to load a program into the CNC from a floppy or NC tape.

Procedure for Inputting a Program

| Procedure | 1 Make sure the input device is ready for reading. |
|---|--|
| | 2 Press the EDIT switch on the machine operator's panel. |
| | 3 When using a floppy, search for the required file according to the procedure in Section 8.2 . |
| | 4 Press function $PRGRM$ key and display the program screen. |
| | 5 After entering address O, specify a program number to be assigned to the program. When no program number is specified here, the program number used on the floppy or NC tape is assigned. |
| | 6 Press soft key INPUT. |
| | The program is input and the program number specified is assigned to the program. |
| | 7 To abandon the input at any point, press the \mathbb{R}^{ESET} key. |
| Explanations | |
| • Collation | If a program is input while the data protect key on the machine operator's panel turns ON, the program loaded into the memory is verified against the contents of the floppy or NC tape. If a mismatch is found during collation, the collation is terminated with an alarm (P/S No. 79). If the operation above is performed with the data protection key turns OFF, collation is not performed, but programs are registered in memory. |
| Inputting multiple programs from an NC tape | When a tape holds multiple programs, the tape is read up to ER (or %). |
| 1 | C 01111 M02; 02222 M30; 03333 M02; ER(%) |

| Program numbers on a NC tape | □ When a program is entered without specifying a program number. •The O-number of the program on the NC tape is assigned to the program. If the program has no O-number, the N-number in the first block is assigned to the program. •When the program has neither an O-number nor N-number, the previous program number is incremented by one and the result is assigned to the program. □ When a program is entered with a program number The O-number on the NC tape is ignored and the specified number is assigned to the program. When the program is followed by additional programs, the first additional program is given the program number. Additional program numbers are calculated by adding one to the last program. | |
|--|--|--|
| Input with the soft keys | The soft keys can be used to input a program. | |
| Procedure for program input with the soft keys | | |
| | Display the program screen in EDIT mode or background edit mode. Press the [I/O] soft key. Input address O, then the program number. If this step is skipped, | |

- the program number on the NC tape is automatically selected.
- 4. Press the **[READ]** soft key. To abandon input at any point, press the **[STOP]** soft key.
- 5. After input has been completed, press the **[CAN]** soft key to display the program screen again.
- Program input in background edit mode

Program input is identical to that in foreground edit mode. If the reset key is pressed to abandon input in the background while a program is being executed, however, the program execution is also halted. To input a program in background edit mode, use the soft keys.

Alarm

| No. | Description |
|-----|--|
| 70 | The size of memory is not sufficient to store the input pro- grams |
| 73 | An attempt was made to store a program with an existing pro- gram number. |
| 79 | The verification operation found a mismatch between a pro- gram loaded into memory and the contents of the program on the floppy or NC tape. |

8.4.2 Outputting a Program

A program stored in the memory of the CNC unit is output to a floppy or NC tape.

| Procedure for Outputting a Program | | |
|---|--|--|
| Procedure | 1 Make sure the output device is ready for output. | |
| | 2 To output to an NC tape, specify the punch code system (ISO or EIA) using a parameter. To output the program to a floppy disk, select ISO. | |
| | 3 Press the EDIT switch on the machine operator's panel. | |
| | 4 Press function \Pr{RGRM} key. | |
| | 5 Enter address O key. | |
| | 6 Enter a program number. If –99999 is entered, all programs stored in memory are output. | |
| | 7 Press soft key $\begin{bmatrix} OUTPT\\ START \end{bmatrix}$. | |
| | The specified program or programs are output. | |
| Explanations (Output to a floppy) | | |
| File output location | When output is conducted to the floppy, the program is output as the new file after the files existing in the floppy. New files are to be written from the beginning with making the old files invalid, use the above output operation after the N0 head searching. | |
| An alarm while a program is output | When P/S alarm 86 occurs during program output, the floppy is restored to the condition before the output. | |
| Outputting a program after file heading | When program output is conducted after N1 to N9999 head searching, the new file is output as the designated n–th position. In this case, 1 to n–1 files are effective, but the files after the old n–th one are deleted. If an alarm occurs during output, only the 1 to n–1 files are restored. | |
| Efficient use of memory | To efficiently use the memory in the cassette or card, output the program by setting parameter (No. 002#7 or No. 012#7) to 1. This parameter makes the feed is not output, utilizing the memory efficiently. | |
| On the memo record | Head searching with a file No. is necessary when a file output from the CNC to the floppy is again input to the CNC memory or compared with the content of the CNC memory. Therefore, immediately after a file is output from the CNC to the floppy, record the file No. on the memo. | |
| Output with the soft keys | The soft keys can be used to input a program. | |

Procedure for program output with the soft keys

- 1. Display the program screen in EDIT mode.
- 2. Press the **[I/O]** soft key.
- 3. Input address \bigcirc , then the program number. If –9999 is entered as

the program number, all programs in memory are output.

- 4. Press the **[READ]** soft key. To abandon input at any point, press the **[STOP]** soft key.
- 5. After input has been completed, press the **[CAN]** soft key to display the program screen again.

 Program outnput in background edit mode Program input is identical to that in foreground edit mode. If the reset key is pressed to abandon input in the background while a program is being executed, however, the program execution is also halted. A program that is currently selected in the foreground can also be output.

NOTE

Some machines also use the (START) key, used to start punching of the CNC tape in the background, as the cycle start key to start automatic operation. If such a machine is being used, punch the CNC tape in a mode other than automatic operation mode. (To check whether the key is used for both purposes, refer to the manual provided by the machine tool builder.)

| Explanations (Output to an NC tape) | | | | |
|---|---|--|--|--|
| Format | A program is output to paper tape in the following format: | | | |
| | $\begin{array}{c c} & & & & & & \\ \hline & & & & & \\ \hline & & & & &$ | | | |
| | Feed of 3 feet Feed of 3 feet | | | |
| | If three–feet feeding is too long, press the CAN key during feed punching | | | |
| | to cancel the subsequent feed punching. | | | |
| • TV check | A space code for TV check is automatically punched. | | | |
| ISO code | When a program is punched in ISO code, two CR codes are punched after an LF code. | | | |
| | LF CR CR | | | |
| | Only LF can be punched as the EOB if bit 7 of parameter 070 is specified accordingly. | | | |
| Stopping the punch | Press the RESET key to stop punch operation. | | | |
| Punching all programs | All programs are output to paper tape in the following format. | | | |
| | $\underbrace{\text{ER}}_{(\%)} \xleftarrow{\text{Program}} \underbrace{\text{Program}}_{(\%)} \underbrace{\text{Program}}_{(\%)}$ | | | |
| | Space of 1–feet Feed of 3–feet | | | |

The sequence of the programs punched is undefined.

8.5 OFFSET DATA INPUT AND OUTPUT

| 8.5.1 Inputting Offset Data | Offset data is loaded into the memory of the CNC from a floppy or NC tape. The input format is the same as for offset value output. See 8.5.2. When an offset value is loaded which has the same offset number as an offset number already registered in the memory, the loaded offset data replaces existing data. | |
|-------------------------------------|---|--|
| Procedure for Inputting Offset data | | |
| Procedure | 1 Make sure the input device is ready for reading | |
| | 2 Press the EDIT switch on the machine operator's panel. | |
| | 3 When using a floppy, search for the required file according to the procedure in 8.2. | |

- **4** Press function $\begin{bmatrix} MENU \\ OFSET \end{bmatrix}$ key.and display the offset screen.
- 5 Press INPUT key.

The input offset data will be displayed on the screen after completion of input operation.

| 8.5.2 Outputting Offset Data | All offset data is output in a output format from the memory of the CNC to a floppy or NC tape. |
|---------------------------------|---|
|---------------------------------|---|

| Procedure for Outputting Offset Data | |
|--------------------------------------|---|
| Procedure | Make sure the output device is ready for output. Specify the punch code system (ISO or EIA) using a parameter. Press the EDIT switch on the machine operator's panel. |
| | 4 Press function (MENU OFSET) key. 5 Press soft key (OUTPUT START) . Offset data is output in the output format described below. |
| Explanations | |
| Output format | Output format is as follows: |
| | Format (1) For tool compensation memory A G10 P_R_; where P_: Offset No. R_: Tool compensation amount The L1 command may be used instead of L11 for format compatibility of the conventional CNC |

— 340 —

| 8.6 INPUTTING AND OUTPUTTING PARAMETERS AND PITCH ERROR COMPENSATION DATA | Parameters and pitch error compensation data are input and output from different screens, respectively. This chapter describes how to enter them. |
|--|---|
| 8.6.1 Inputting Parameters | Parameters are loaded into the memory of the CNC unit from a floppy or NC tape. The input format is the same as the output format. See 8.6.2 . When a parameter is loaded which has the same data number as a parameter already registered in the memory, the loaded parameter replaces the existing parameter. |
| Inputting parar | neters |
| Procedure | Make sure the input device is ready for reading. When using a floppy, search for the required file according to the procedure in 8.2. Press the EMERGENCY STOP button on the machine operator's panel. Press function DENOSE have been and display the parameter screen. Enter 1 in response to the prompt for writing parameters (PWE). Alarm P/S100 (indicating that parameters can be written) appears. Press soft key DEVT. Parameters are read into memory. Upon completion of input, the "INPUT" indicator at the lower-right corner of the screen disappears. Enter 0 in response to the prompt for writing parameters. Turn the power to the NC back on. Release the EMERGENCY STOP button on the machine operator's panel. |

8.6.2 All parameters are output in the defined format from the memory of the CNC to a floppy or NC tape.

| Outputting parameters | |
|-----------------------------------|--|
| Procedure | Make sure the output device is ready for output. Specify the punch code system (ISO or EIA) using a parameter. Press the EDIT switch on the machine operator's panel. Press function press function press key. and display the parameter screen. Press soft key output are output in the defined format. |
| Explanations | |
| Output format | Output format is as follows: N P; |
| | N:Parameter No. P:Parameter setting value . |

8.7 INPUTTING/OUTPUTT ING CUSTOM MACRO B COMMON VARIABLES

8.7.1 Inputting Custom Macro B Common Variables

The value of a custom macro B common variable (#500 to #531) is loaded into the memory of the CNC from a floppy or NC tape. The same format used to output custom macro B common variables is used for input. See **8.7.2.** For a custom macro common variable to be valid, the input data must be executed by pressing the cycle start button after data is input. When the value of a common variable is loaded into memory, this value replaces the value of the same common variable already existing (if any) in memory.

Inputting custom macro common variables

Procedure 1 In

- 1 Input the program according to the procedure in **8.4.1**.
- 2 Press the **AUTO** switch on the machine operator's panel upon completing input.
- **3** Press the cycle start button to execute the loaded program.
- 4 Display the macro vriable screen to chek whether the values of the common variables have been set correctly.

Explanations

• **Common variables** The common variables (#500 to #531) can be input and output. Common variables #100 to 199 cannot be input or output.

8.7.2 Outputting Custom Macro B Common Variable

Custom macro common variables (#500 to #531) stored in the memory of the CNC can be output in the defined format to a floppy or NC tape.

Outputting custom macro common variable

Procedure

- 1 Make sure the output device is ready for output.
- 2 Specify the punch code system (ISO or EIA) using a parameter.
- **3** Press the EDIT switch on the machine operator's panel.
- 4 Press function MENU key and display the macro variables.
- 5 Press function [UTPT] start key. Common variables are output in the defined format.

Explanations

• Output format

The output format is as follows:

- (1) The precision of a variable is maintained by outputting the value of the variable as <expression>.
- (2) Undefined variable
- (3) When the value of a variable is 0
- Output file name

When the floppy disk directory display function is used, the name of the output file is [MACRO VAR].

Common variable
 The common variable

The common variables (#500 to #531) can be input and output. Common variables #100 to 199 cannot be input or output.



EDITING PROGRAMS

General

This chapter describes how to edit programs registered in the CNC. Editing includes the insertion, modification, deletion, and replacement of words. Editing also includes deletion of the entire program and automatic insertion of sequence numbers. The extended part program editing function can copy, move, and merge programs. This chapter also describes program number search, sequence number search, word search, and address search, which are performed before editing the program.


9.1 **INSERTING, ALTERING** AND DELETING A WORD

This section outlines the procedure for inserting, modifying, and deleting a word in a program registered in memory.

Procedure for inserting, altering and deleting a word

- 1 Select **EDIT** mode.
- 2 Press function key and display the program screen.
- Select a program to be edited. 3 If a program to be edited is selected, perform the operation 4. If a program to be edited is not selected, search for the program number.
- 4 Search for a word to be modified. ·Scan method ·Word search method
- 5 Perform an operation such as altering, inserting, or deleting a word.

Explanation

 Concept of word and editing unit

A word is an address followed by a number. With a custom macro B, the concept of word is ambiguous. So the editing unit is considered here. The editing unit is a unit subject to alteration or deletion in one operation. In one scan operation, the cursor indicates the start of an editing unit. An insertion is made after an editing unit. Definition of editing unit (i) Program portion from an address to immediately before the next address (ii) An address is an alphabet, IF, WHILE, GOTO, END, DO=, or; (EOB). According to this definition, a word is an editing unit. The word "word," when used in the description of editing, means an editing unit according to the precise definition. Data input during editing To insert or modify a word during editing, the following data is entered. When the standard key panel is being used One word (a single alphabetic character followed by a numeric value or symbol) is entered. • Editing B with the standard key panel Even if the standard key panel is being used, editing B can be enabled by specifying bit 7 of parameter 018 accordingly. Two or more addresses can be input at one time. If the INPUT key is pressed after a single word (a single alphabetic character followed by a numeric

value or symbol) is input, another word can be input. After all data has been entered, press the edit key to start editing. To enable editing B, note the following:

— 346 —

 \cdot The **INPUT** key is used to identify a breakpoint between words.

A program cannot be input or output while a program is displayed. Input or output a program on the program directory screen.

- Input a program number as one word containing address O.
- \cdot Up to 32 characters can be entered at one time.
- Each time the $\begin{bmatrix} CAN \end{bmatrix}$ key is pressed, only the most-recently entered character is deleted.

WARNING

The user cannot continue program execution after altering, inserting, or deleting data of the program by suspending machining in progress by means of an operation such as a single block stop or feed hold operation during program execution. If such a modification is made, the program may not be executed exactly according to the contents of the program displayed on the screen after machining is resumed. So, when the contents of memory are to be modified by part program editing, be sure to enter the reset state or reset the system upon completion of editing before executing the program.

9.1.1 Word Search

A word can be searched for by merely moving the cursor through the text (scanning), by word search, or by address search.

Procedure for scanning a program

1 Press the cursor key |

The cursor moves forward word by word on the screen; the cursor is displayed at a selected word. The cursor is positioned to the address of the selected word.

2 Press the cursor key

The cursor moves backward word by word on the screen; the cursor is displayed at a selected word.

Example) When Z1250.0 is scanned

| Program | O0050 N1234 |
|--------------------------------|-------------|
| O0050 ; | |
| N1234 X100.0 <u>Z</u> 1250.0 ; | |
| S12 ; | |
| N5678 M03 ; | |
| M02 ; | |
| % | |

- 3 Holding down the cursor key ↓ or ↑ scans words continuously.
- 4 Pressing the page key ↓ displays the next page and searches for the first word of the page.
- 5 Pressing the page key **†** displays the previous page and searches for the first word of the page.
- 6 Holding down the page key ↓ or ↑ displays one page after another.

— 348 —

Procedure for searching a word

Example) of Searching for S12



Procedure for searching an address

1

2

Example) of Searching for M03

| _ | | |
|---------------------------------------|-----------------------|----------------------|
| PROGRAM | O0050 N1234 | N1234 is being |
| N1234 X100.0 Z1250.0 ; | ٠ | searched for/scanned |
| S12 ; N5678 M03 · ◀ | | M03 is searched for |
| M02 ; | | |
| % | | |
| Key in address M. | | |
| Press the cursor key \blacksquare . | | |
| Upon completion of search o | peration, the curso | r is displayed at |
| "M" of M03. Pressing the | key rather than | the 📕 key |
| performs search operation in | the reverse direction | on. |
| | | |

Alarm

| Alarm number | Description |
|--------------|---|
| 71 | The word or address being searched for was not found. |

9.1.2 Heading a Program

The cursor can be jumped to the top of a program. This function is called heading the program pointer. This section describes the two methods for heading the program pointer.

Procedure for Heading a Program

| Method 1 | 1 | Press \bigcirc when the program screen is selected in EDIT mode. When the cursor has returned to the start of the program, the contents of the program are displayed from its start on the screen. |
|----------|---|---|
| Method 2 | 1 | Select AUTO or EDIT mode. |
| | 2 | Press function $PROG$ key and display the program. |
| | 3 | Press the address key O. |
| | 4 | Press the cursor key 1. |

9.1.3 Inserting a Word

Procedure for inserting a word

- **1** Search for or scan the word immediately before a word to be inserted.
- 2 Key in an address to be inserted.
- 3 Key in data.
- 4 Press the INSRT key.

Example of Inserting T15

Procedure

1 Search for or scan Z1250.0.

| | Program O0050 ; N1234 X100.0 <u>Z</u> 1250.0 ; ← S12 ; N5678 M03 ; M02 ; % | O0050 N1234 | Z1250.0 is searched for/scanned. |
|------------|--|-------------|-------------------------------------|
| 2] 3] | Key in T 1 5. Press the I key. | | |
| | Program O0050 ; N1234 X100.0 Z1250.0 <u>_</u> 1 5 S12 ; N5678 M03 ; M02 ; % | O0050 N1234 | — T15 is inserted. |

9.1.4 Altering a Word

| | Procedure for altering a word | |
|-----------|---|--|
| | Search for or scan a word to be altered. Key in an address to be inserted. Key in data. Press the ALTER key. | |
| | Example of changing T15 to M15 | |
| Procedure | 1 Search for or scan T15. | |

3 Press the ALTER key.

| | Program O0050 ; | O0050 N1234 | T15 is searched for/ |
|---|--|-------------|----------------------|
| | N1234 X100.0 Z1250.0 <u>T</u> 15 ; S12 ; N5678 M03 ; M02 ; % | 4 | scanned. |
| 2 | Key in M 1 5. | | |

 Program
 O0050 N1234

 O0050 ;
 N1234 X100.0 Z1250.0 M15 ;
 T15 is changed to

 S12 ;
 N5678 M03 ;
 M15.

 M02 ;
 %
 T15 is changed to

9.1.5 Deleting a Word

| Proc | cedure for deleting a word |
|-----------|--|
| | Search for or scan a word to be deleted. Press the DELET key. |
| Exa | ample of deleting X100.0 |
| Procedure | 1 Search for or scan X100.0. Program 00050 N1234 00050 ; N1234 X100.0 Z1250.0 M15 ; S12 ; N5678 M03 ; M02 ; % |
| | 2 Press the DELET key. Program 00050 N1234 00050 ; N1234 Z1250.0 M15 ; |

| 9.2 DELETING BLOCKS | A block or blocks can be deleted in a program. | |
|---------------------------|--|--|
| 9.2.1 Deleting a Block | The procedure below deletes a block up to its EOB code; the cursor advances to the address of the next word. | |
| Procedure | for deleting a block | |
| | Search for or scan address N for a block to be deleted. Key in EOB. Press the DELET key. | |
| Example o | of deleting a block of No.1234 | |
| Procedure | 1 Search for or scan N1234. Program 00050 N1234 00050 ; N1234 Z1250.0 M15 ; | |



9.2.2 Deleting Multiple Blocks

The blocks from the currently displayed word to the block with a specified sequence number can be deleted.

Procedure for deleting multiple blocks

- 1 Search for or scan a word in the first block of a portion to be deleted.
- 2 Key in address N
- **3** Key in the sequence number for the last block of the portion to be deleted.
- 4 Press the DELET key.

Example of deleting blocks from a block containing N01234 to a block containing N56789

Procedure

1 Search for or scan N1234.

| | Program O0050 ; <u>N</u> 1234 Z1250.0 M15 ; S12 ; N56789 M03 ; M02 ; % | O0050 N1234 | N1234 is searched for/scanned. |
|---|--|-------------|---|
| 2 | Key in N 5 6 | 78. | |
| 2 | Program O0050 ; <u>N1234 Z1250.0 M15 ;</u> S12 ; N5678 M03 ; M02 ; % | 00050 N1234 | Underlined part is deleted. |
| 5 | Program | O0050 N1234 | |
| | O0050 ; <u>M</u> 02 ; % | | Blocks from block containing N1234 to block containing N5678 have been deleted. |

9.3 PROGRAM NUMBER SEARCH

When memory holds multiple programs, a program can be searched for. There are two methods as follows.

| Procedure for program number search | | | |
|-------------------------------------|----------|--|--|
| | Method 1 | Select ED Press PRGPM Key in add Key in a p Press the o Upon comfor is disput fithe proof | IT or AUTO mode. (a) key to display the program screen. (dress \bigcirc . (b) corogram number to be searched for. (cursor key \bigcirc . (cursor key \bigcirc . (cursor key \bigcirc . (cursor key \bigcirc .) (cursor key \bigcirc . (cursor key \bigcirc .) (cursor key \bigcirc |
| | Method 2 | Select ED Press PRGEN Key in add Press the output this case | IT or AUTO mode. (It or AUTO mode.) (It or |
| Alarm | | No. | Contents The program with the selected number cannot be searched |

| 59 | The program with the selected number cannot be searched during external program number search. |
|----|--|
| 71 | The specified program number was not found during program number search. |

9.4 DELETING PROGRAMS

Programs registered in memory can be deleted, either one program by one program or all at once. Also, More than one program can be deleted by specifying a range.

9.4.1 **Deleting One Program**

A program registered in memory can be deleted.

Procedure for deleting one program 1 Select the **EDIT** mode. Press PRGRM to display the program screen. 2 **3** Key in address **O** . Key in a desired program number. 4 Press the DELET key. 5 The program with the entered program number is deleted.

9.4.2

All programs registered in memory can be deleted.

Deleting All Programs

Procedure for deleting all programs

- Select the **EDIT** mode. 1
- 2 Press PRGRM to display the program screen.
- 3 Key in address Ο
- 4 Key in –9999.
- 5 Press edit key DELET to delete all programs.

9.5 EXTENDED PART PROGRAM EDITING FUNCTION

With the extended part program editing function, the operations described below can be performed using soft keys for programs that have been registered in memory.

Following editing operations are available :

- All or part of a program can be copied or moved to another program.
- One program can be merged at free position into other programs.
- A specified word or address in a program can be replaced with another word or address.

9.5.1 Copying an Entire Program

A new program can be created by copying a program.



Fig. 9.5.1 Copying an Entire Program

In Fig. 9.5.1, the program with program number xxxx is copied to a newly created program with program number yyyy. The program created by copy operation is the same as the original program except the program number.

Procedure of copying an entire program

- 1 Enter the **EDIT** mode.
- 2 Press function **PRGRM** key to display the program screen .
- **3** Press the continuous menu key \triangleright .
- 4 Press soft key [EX–EDT].
- 5 Check that the screen for the program to be copied is selected and press soft key **[COPY]**.
- 6 Press soft key [ALL].
- 7 Enter the number of the new program (with only numeric keys) and press the key.
- 8 Press soft key [EXEC].



(EXEC) (

)(

9.5.2 Copying Part of a Program

A new program can be created by copying part of a program.



Fig. 9.5.2 Copying Part of a Program

In Fig. 9.5.2, part B of the program with program number xxxx is copied to a newly created program with program number yyyy. The program which No.is xxxx remains unchanged after copy operation.

Procedure for copying part of a program

Numeric keys 0 to 9



- 1 Perform steps 1 to 5 in subsection 9.5.1.
- 2 Move the cursor to the start of the range to be copied and press soft key [CRSR ~].
- 3 Move the cursor to the end of the range to be copied and press soft key [~ CRSR] or [~ BTTM] (in the latter case, the range to the end of the program is copied regardless of the position of the cursor).
- 4 Enter the number of the new program (with only numeric keys) and press the $\boxed{|NPUT|}$ key.
- 5 Press soft key [EXEC].

9.5.3 Moving Part of a Program

A new program can be created by moving part of a program.



Fig. 9.5.3 Moving Part of a Program

In **Fig. 9.5.3**, part B of the program with program number xxxx is moved to a newly created program with program number yyyy; part B is deleted from the program with program number xxxx.

2 Check that the screen for the program to be moved is selected and

3 Move the cursor to the start of the range to be moved and press

Procedure for moving part of a program

1 Perform steps **1** to **4** in subsection 9.5.1.

press soft key [MOVE].

soft key [CRSR ~].



0 to 9

)(

Numeric keys

(EXEC) () () (

- 4 Move the cursor to the end of the range to be moved and press soft key [~ CRSR] or [~ BTTM] (in the latter case, the range to the end of the program is copied regardless of the position of the cursor).
 - 5 Enter the number of the new program (with only numeric keys) and press the input key.
 - 6 Press soft key [EXEC].

9.5.4 Merging a Program

Another program can be inserted at an arbitrary position in the current program.





In **Fig. 9.5.4**, the program with program number XXXX is merged with the program with program number YYYY. The OYYYY program remains unchanged after merge operation.

Procedure for merging a program



Numeric keys O to 9



- 1 Perform steps 1 to 4 in subsection 9.5.1.
- 2 Check that the screen for the program to be edited is selected and press soft key [MERGE].
- 3 Move the cursor to the position at which another program is to be inserted and press soft key [~ 'CRSR] or [~ BTTM'] (in the latter case, the end of the current program is displayed).

4 Enter the number of the program to be inserted (with only numeric keys) and press the key.

5 Press soft key [EXEC]. The program with the number specified in step 4 is inserted before the cursor positioned in step 3.

| 9.5.5 Supplementary Explanation for Copying, Moving and Merging Explanations | |
|--|--|
| • Setting an editing range | The setting of an editing range start point with [CRSR ~] can be changed freely until an editing range end point is set with [~ CRSR] or [~ BTTM] . If an editing range start point is set after an editing range end point, the end point must be reset. The setting of an editing range start point and end point remains valid until an operation is performed to invalidate the setting. One of the following operations invalidates a setting: An edit operation other than address search, word search/scan, and search for the start of a program is performed after a start point or end point is set. •Processing is returned to operation selection after a start point or end point is set. |
| Without specifying a program number | In copying program and moving program, if [EXEC] is pressed without specifying a program number after an editing range end point is set, a program with program number O0000 is registered as a work program. This O0000 program has the following features: The program can be edited in the same way as a general program. (Do not run the program.) If a copy or move operation is newly performed, the previous information is deleted at execution time, and newly set information (all or part of the program) is reregistered. (In merge operation, the previous information is not deleted.) When the program becomes unnecessary, delete the program by a normal editing operation. |
| Editing when the system waiting for a program number to be entered | When the system is waiting for a program number to be entered, no edit operation can be performed. |
| Restrictions | |

• Number of digits for program number

If a program number is specified by 5 or more digits, a format error is generated.

| Alarm no. | Contents |
|-----------|---|
| 70 | Memory became insufficient while copying or inserting a program. Copy or insertion is terminated. |
| 101 | The power was interrupted during copying, moving, or inserting a program and memory used for editing must be cleared. |

Alarm

9.5.6 Replacement of Words and Addresses

Replace one or more specified words.

Replacement can be applied to all occurrences or just one occurrence of specified words or addresses in the program.

Procedure for hange of words or addresses









Examples

- Replace X100 with Y200
- Replace X100Y200 with X30
- Replace IF with WHILE
- Replace X with ,C10

Explanation

 Replacing custom macros

- **1** Perform steps **1** to **4** in 9.5.1.
- 2 Press soft key [CHANGE].
- 3 Enter the word or address to be replaced.
- 4 Press soft key [BEFORE].
- 5 Enter the new word or address.
- 6 Press soft key [AFTER].
- 7 Press soft key [EXEC] to replace all the specified words or addresses after the cursor.
 Press soft key [1–EXEC] to search for and replace the first occurrence of the specified word or adress after the cursor.
 Press soft key [SKIP] to only search for the first occurrence of the specified word or address after the cursor.

| [CHANGE] X 1 0 0 [BEFORE] Y 2 0 0 [AFTER][EXEC] |
|--|
| [CHANGE] X 1 0 0 Y 2 0 0 [BEFORE] X 3 0 [AFTER][EXEC] |
| [CHANGE] I F [BEFORE] W H I L E [AFTER] [EXEC] |
| [CHANGE] X [BEFOR] , C 1 0 [AFTER][EXEC] |

The following custom macro B words are replaceable: IF, WHILE, GOTO, END, DO, BPRNT, DPRINT, POPEN, PCLOS The abbreviations of custom macro B words can be specified. When abbreviations are used, however, the screen displays the abbreviations as they are key input, even after soft key **[BEFORE]** and **[AFTER]** are pressed.

Restrictions

- The number of Up to 15 characters can be specified for words before or after replacement. (Sixteen or more characters cannot be specified.)
- The characters for replacement

Words before or after replacement must start with a character representing an address.(A format error occurs.)

9.6 BACKGROUND EDITING

Editing a program while executing another program is called background editing. The method of editing is the same as for ordinary editing (foreground editing).

During background editing, all programs cannot be deleted at once.

Procedure for background editing

- **1** Press function |PRGRM key to display the program screen.
- 2 Press the rightmost soft key (continuous menu key), then press soft key [BG–EDT].The background editing screen is displayed (PROGRAM

(BG-EDIT) is displayed at the top left of the screen).

- **3** Edit a program on the background editing screen in the same way as for ordinary program editing.
- 4 After editing is completed, press soft key [BG–END].

Explanation

 Alarms during background editing Alarms that may occur during background editing do not affect foreground operation. Conversely, alarms that may occur during foreground operation do not affect background editing. In background editing, if an attempt is made to edit a program selected for foreground operation, a BP/S alarm (No. 140) is raised. On the other hand, if an attempt is made to select a program subjected to background editing during foreground operation (by means of subprogram calling or program number search operation using an external signal), a P/S alarm (Nos. 059, 078) is raised in foreground operation. As with foreground program editing, P/S alarms occur in background editing. However, to distinguish these alarms from foreground alarms, BP/S is displayed in the data input line on the background editing screen.

CAUTION

- 1 If the available part program storage is 80 m or less, free space in memory is used for background editing. A program to be subjected to background editing is copied into the free area in memory, then the original program is deleted. Subsequently, editing starts. Background editing can be executed if the part program storage has sufficient free area to which the target program can be copied and if program registration is allowed in terms of number. If background editing is repeated, the number of deleted areas will increase. To use these deleted areas efficiently, memory must be reorganized.
- 2 If the reset key is pressed to abandon program input or output in background editing, the machining in the foreground will also be halted. To input or output a program in the background, therefore, use the soft keys. To halt the input or output, press the **[STOP]** soft key.
- 3 If a reset by M02/M30 of the machining program in the foreground is executed during program input or output in background editing, program input or output is halted. Program input or output can be prevented from being halted by the reset in the foreground if bit 2 of parameter 076 is specified accordingly.
- 4 In background editing, program input or output by the external activation signal (MINP) or input/output unit external control is inhibited.

9.7 REORGANIGING MEMORY

If the available part program storage is 120 m or more, or if the background editing function is supported, repeated program editing will create many small, unused areas in memory. Reorganizing memory arranges these unused areas into a single, contiguous area that can be used by programs.

| Procedure | e for Reorganiging Memory |
|----------------------------|---|
| Procedure 1 (reset key) | Press the emergency stop, external reset, or reset key. The procedure for reorganizing memory is automatically started. |
| Procedure 2 (soft key) | Select EDIT mode. Press the PRGRM key to display the program. Press the [LIB] soft key. Press the [REORGANIZE] soft key. |
| | NOTE 1 One memory is reorganized, the system searches for the beginning of the selected program and the cursor is returned to that point. 2 If the power is turned off during the memory reorganization, alarm 101 occurs when the power is subsequently turned on. Before turning the power off after resetting an alarm, first check whether memory reorganization has been completed. While memory reorganization is being performed, EDIT blinks at the bottom right corner of the screen. 3 As described in procedure 1, above, the memory reorganization procedure is automatically started when a reset is performed. Memory reorganization can be prevented from being started by a reset if bit 0 of parameter 056 is specified accordingly. 4 Memory reorganization cannot be executed during background editing. |

10 CREATING PROGRAMS

Programs can be created using any of the following methods: \cdot MDI keyboard

This chapter describes creating programs using the MDI panel. This chapter also describes the automatic insertion of sequence numbers.

10.1 CREATING PROGRAMS USING THE MDI PANEL

Programs can be created in the EDIT mode using the program editing functions described in Chapter 9.

Procedure for Creating Programs Using the MDI Panel

Procedure

- 1 Enter the **EDIT** mode.
- **2** Press the |PRGRM| key.
- **3** Press address key $| \mathbf{O} |$ and enter the program number.
- 4 Press the **NSRT** key.
- **5** Create a program using the program editing functions described in Chapter 9.

10.2 AUTOMATIC INSERTION OF SEQUENCE NUMBERS

Sequence numbers can be automatically inserted in each block when a program is created using the MDI keys in the EDIT mode. Set the increment for sequence numbers in parameter 550.

Procedure for automatic insertion of sequence numbers **Procedure 1** Set 1 for SEQUENCE (see subjection 11.4.3). 2 Enter the **EDIT** mode. **3** Press PRGRM to display the program screen. 4 Search for or register the number of a program to be edited and move the cursor to the EOB (;) of the block after which automatic insertion of sequence numbers is started. When a program number is registered and an EOB (;) is entered with the **INSRT** key, sequence numbers are automatically inserted starting with 0. Change the initial value, if required, according to step 10, then skip to step 7. N and enter the initial value of N. 5 Press address key 6 Press INSRT key. 7 Enter each word of a block. 8 Press EOB key.

9 Press key. The EOB is registered in memory and sequence numbers are automatically inserted. For example, if the initial value of N is 10 and the parameter for the increment is set to 2, N12 inserted and displayed below the line where a new block is specified.

| _ | | | | | | | | | ~ |
|------|------------------|----|------|----|-----|------|-------|-------|---|
| | PROGRAM | | | | | | 00040 | N0012 | |
| | 00040 • | | | | | | 00010 | 10012 | |
| | N10 G92 | x0 | YO Z | 0; | | | | | |
| | <u>N</u> 12 % | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | (| | | | | EDIT | | | |
| | [PRGRM |][| LIB |][| I/O |][|][| 1 | |
| < l> | | | | | | | | | 1 |

- 10 · In the example above, if N12 is not necessary in the next block, pressing the p_{ELET} key after N12 is displayed deletes N12.
 - To insert N100 in the next block instead of N12, enter N100 and press

ALTER after N12 is displayed. N100 is registered and initial

value is changed to 100.

11

SETTING AND DISPLAYING DATA

General

To operate a CNC machine tool, various data must be set on the CRT/MDI panel. The operator can monitor the state of operation with data displayed during operation.

This chapter describes how to display and set data for each function. This chapter describes the procedure, assuming that the soft keys are used to select a desired chapter.

Explanations

• Screen transition chart



MDI function keys (Shaded keys (_____) are described in this chapter.)

Data protection key

The screen transition for when each function key on the MDI panel is pressed is shown below. The subsections referenced for each screen are also shown. See the appropriate subsection for details of each screen and the setting procedure on the screen. See other chapters for screens not described in this chapter.

Function key GRAPH is not used

The machine may have a data protection key to protect part programs. Refer to the manual issued by the machine tool builder for where the data protection key is located and how to use it.



— 374 —











• Setting screens

The table below lists the data set on each screen.

Table.11. Setting screens and data on them

| No. | Setting screen | Contents of setting | Reference item |
|-----|--------------------------------|---|-------------------|
| 1 | Tool offset value | Tool length offset value Cutter compensation value | 11.4.1 |
| 2 | Setting data (handy) | Parameter write TV check Punch code Input unit (mm/inch) I/O channel Automatic insert of Sequence No. Conversion of tape format Incremental/absolute MDI travel distance | 11.5.3 |
| | | Mirror image | 11.5.3 |
| | | Parts required | 11.5.4 |
| 3 | Macro variables | Custom macro common vari- ables (#100 to #149) or (#500 to #531) | 11.4.3 |
| 4 | Parameter | Parameter | 11.5.1 |
| | | Pitch error compensation data | 11.5.2 |
| 5 | software operator's panel | Mode selection Jog feed axis selection Jog rapid traverse Axis selection for Manual pulse generator Multiplication for manual pulse generator Jog feedrate Feedrate override Rapid traverse override Optional block skip Single block Machine lock Dry run Protect key Feed hold | 11.6.2 |
| 6 | Work coordinate system setting | Work origin offset value | 11.4.2 |

| 11.1 SCREENS DISPLAYED BY FUNCTION KEY POS | Press function key POS to display the current position of the tool. The following three screens are used to display the current position of the tool: | | | | |
|--|--|--|--|--|--|
| | Position display screen for the work coordinate system. Position display screen for the relative coordinate system. | | | | |
| | · Overall position display screen. | | | | |
| | The above screens can also display the feedrate, run time, and the number of parts. | | | | |
| | | | | | |
| 11.1.1 | Displays the current position of the tool in the workpiece coordinate | | | | |

11.1.1 Position Display in the Work Coordinate System

Displays the current position of the tool in the workpiece coordinate system. The current position changes as the tool moves. The least input increment is used as the unit for numeric values. The title at the top of the screen indicates that absolute coordinates are used.

Display procedure for the current position screen in the workpiece coordinate system

- **1** Press function key POS
- 2 Press soft key [ABS].

| ACTUAL POS | ITION (ABSOLUT | E) C | 0010 | N0000 |
|------------|-----------------|----------|----------|-------|
| x | 123. | 456 | 5 | |
| | | | | |
| Ŷ | 363. | 233 | 3 | |
| 7 | 0 | 000 | ` | |
| L | υ. | | J | |
| | | | | |
| | PA | RT COUNT | | 1 |
| RUN TIME | 0H 1M CY | CLE TIME | 0H 1 | M335 |
| ACT.F 3 | 000 MM/M BUI | F AUTO | | |
| | | | | |

Explanations

• Display including compensation values

Bits 5 of parameter 018 can be used to select whether the displayed values include tool length offset.
11.1.2 Position Display in the Relative Coordinate System

Displays the current position of the tool in a relative coordinate system based on the coordinates set by the operator. The current position changes as the tool moves. The increment system is used as the unit for numeric values. The title at the top of the screen indicates that relative coordinates are used.

Display procedure for the current position screen with the relative coordinate system

- **1** Press function key Pos
- 2 Press soft key [REL].

| ACTUAL POSITI | ION (ABSOLUTE) | 00010 N0000 |
|------------------------|----------------------------------|-----------------------|
| X | 123.4 | 56 |
| Y | 363.2 | 33 |
| Z | 0.0 | 00 |
| | | |
| RUN TIME ACT.F 3000 | PART CO 0H 1M CYCLE T MM/M | UNT 1 IME OH 1M33S |
| | BUF AUT | 0 |
| [ABS] [RE | L] [ALL] [|][]] |

Explanations

• Setting the relative coordinates The current position of the tool in the relative coordinate system can be reset to 0 or preset to a specified value as follows:

Procedure to reset the axis coordinate to a specified value

- 1 Key in the address of the axis name (X, Y, etc.) on the relative coordinate screen. The entered axis address blinks. Two or more axis names can be input.
- 2 Press the CAN key. The relative coordinates of the axis having the blinking address are reset to 0.

Procedure topreset a value for a specified axis

- 1 Key in the desired axis name and value on the relative coordinate screen. The entered axis address blinks.
- 2 Press the Key. The relative coordinate of the axis with the blinking address is preset to the specified value.

To enable this operation, specify bit 0 of parameter 0064 accordingly. In this mode, a reset cannot be performed for the specified axis. To reset the coordinate, key in 0 as the preset value.

• Display including Bit 1 of parameter 0001 can be used to select whether the displayed values include tool length offset and cutter compensation.

• Presetting by setting a coordinate system Bit 1 of parameter 0002 is used to specify whether the displayed positions in the relative coordinate system are preset to the same values as in the workpiece coordinate system when a coordinate system is set by a G92 command or when the manual reference position return is made.

11.1.3 Overall Position Display

Displays the following positions on a screen : Current positions of the tool in the workpiece coordinate system, relative coordinate system, and machine coordinate system, and the remaining distance. The relative coordinates can also be set on this screen. See 11.1.2 for the procedure.

Procedure for displaying overall position display screen

Press function key POS
 Press soft key [ALL].

| ACTUAI | POSITION | | 01 | 000 1 | N00010 |
|--------|-----------|---------|-----------|-------|--------|
| (| RELATIVE) | (A | BSOLUTE) | | |
| х | 246.912 | x | 123.456 | | |
| Y | 913.780 | Y | 456.890 | | |
| Z | 1578.246 | Z | 789.123 | | |
| (| MACHINE) | (DI | STANCE TO |) GO) | |
| х | 0.000 | x | 0.000 | | |
| Y | 0.000 | Y | 0.000 | | |
| Z | 0.000 | Z | 0.000 | | |
| | | PAR | T COUNT | | 1 |
| RUN TI | МЕ ОН 4М | CYC | LE TIME | ОН | 1M38s |
| ACT.F | 0 MM/M | | | | |
| | | | MDI | | |
| ADC | ו זיז כו | AT.T. 1 | г | 1 г | 1 |

Explanations

• Coordinate display

The current positions of the tool in the following coordinate systems are displayed at the same time:

- Current position in the relative coordinate system (relative coordinate)
- Current position in the work coordinate system (absolute coordinate)
- Current position in the machine coordinate system (machine coordinate)
- Distance to go (distance to go)
- Distance to go
- Machine coordinate system

The distance remaining is displayed in the AUTO or MDI mode. The distance the tool is yet to be moved in the current block is displayed.

The least command increment is used as the unit for values displayed in the machine coordinate system. However, the least input increment can be used by setting bit 0 of parameter 063.

11.1.4 Actual Feedrate Display

The actual feedrate on the machine (per minute) can be displayed on a current position display screen or program check screen by setting bit 2 of parameter 028. On a 14–inch CRT, the actual feedrate is always displayed.

Display procedure for the actual feedrate on the current position display screen

| ACTUAL POS | ITION (ABSOLUTE) 00010 N0000 |
|------------|------------------------------|
| x | 123 456 |
| Z X | 123.130 |
| Y | 363.233 |
| - | 0 000 |
| Ľ | 0.000 |
| | |
| | PART COUNT 1 |
| | |

The actual feedrate is displayed in units of milimeter/min or inch/min (depending on the specified least input increment) under the display of the current position.

Explanations

• Actual feedrate value

The actual rate is calculated by the following expression:

$$Fact = \sqrt{\sum_{i=1}^{n} (fi)^2}$$

where

n: Number of axes

- fi : Cutting feed rate in the tangential direction of each axis or rapid traverse rate
- Fact : Actual feedrate displayed

The display unit: mm/min (metric input). inch/min (Inch input)

- In the case of movement of rotary axis, the speed is displayed in units of deg/min but is displayed on the screen in units of input system at that time. For example, when the rotary axis moves at 50 deg/min, the following is displayed: 50 INCH/M
- Actual feedrate display of rotary axis
- Actual feedrate display on the other screen

The program check screen also displays the actual feedrate.

11.1.5 Display of Run Time and Parts Count

The run time, cycle time, and the number of machined parts are displayed on the current position display screens.

Procedure for displaying run time and parts count on the current position display screen

| ACTUAL POS | ITION (RELATIVE) | 01000 N00000 |
|------------|------------------|--------------|
| | | |
| X | 123.4 | 56 |
| v | 363 2 | 22 |
| – | JUJ • Z | 55 |
| Ζ | 0.0 | 00 |
| | | |
| | PART CO | DUNT 5 |

The number of machined parts (PART COUNT), run time (RUN TIME), and cycle time (CYCLE TIME) are displayed under the current position.

Explanations

PART COUNT Indicates the number of machined parts. The number is incremented each time M02, M30, or an M code specified by parameter 6710 is executed. P, then press the |CAN| key. The value is reset to 0. Key in address • RUN TIME Indicates the total run time during automatic operation, excluding the stop and feed hold time. CYCLE TIME Indicates the run time of one automatic operation, excluding the stop and feed hold time. This is automatically preset to 0 when a cycle start is performed at reset state. It is preset to 0 even when power is removed. • Display on the other Details of the run time and the number of machined parts are displayed screen on the setting screen. See 11.4.5. Setting of Machine parts The number of machined parts and run time cannot be set on current and run time position display screens. They can be set by parameters 6711, 6751, and 219 or on the setting screen.

• Incrementing the number of machined parts Bit 0 (PCM) of parameter 219 is used to specify whether the number of machined parts is incremented each time M02, M30, or an M code specified by parameter is executed, or only each time an M code specified by parameter 6710 is executed.

11.1.6 Operating Monitor Display

This function displays the load of basic feed axes and 1st spindle with serial interface. And also, it is possible to display the speed of 1st spindle with serial interface. This function is basic.

Explanations

Operation

- 1 The position screen is selected by pressing the function key POS
- 2 The rightmost key [>] on the soft-keys is pressed.
- **3** The soft-key **[MONI]** is pressed.

By the above operation, the operating monitor screen is displayed. Pressing the page keys can display the screen instead of the above operation (2) and (3).

 Operating Monitor Screen

| OPERATING MONITC (LOAD METER) | R | 00001 10000 |
|----------------------------------|----|-----------------------|
| X:******** | 0% | S1:******** 0% |
| Y:******** | 0% | (כספידים אדייניס RPM) |
| Z:******** | 0% | S1:********** 0% |
| | | |
| | | |
| [MONI][|][| 11 11 1 |

(1) LOAD METER (X to Z)

The load of the axis is displayed by percentages to the rated torque of the axis.

One "*" marks on the bar graph denotes 10%.

(2) LOAD METER (S1)

The load of 1st spindle is displayed by percentages to the rated power of the spindle.

One "*" marks on the bar graph denotes 20%.

(3) SPEED METER (S1)

The speed of 1st spindle is displayed by RPM. One "*" marks on the bar graph denotes 10% of the maximum speed of the spindle.

NOTE

- 1 The loads of only the basic axes are displayed.
- 2 The load and speed of only 1st spindle with serial interface are displayed. Those for the 2nd serial spindle and analog interface spindle are not displayed.
- 3 This function is enabled by setting bit 5 of parameter No. 060 to 1.

11.2 SCREENS DISPLAYED BY FUNCTION KEY PROM (IN AUTO MODE OR MDI MODE)

This section describes the screens displayed by pressing function key

in AUTO or MDI mode. The first four of the following screens

display the execution state for the program currently being executed in AUTO or MDI mode and the last screen displays the command values for MDI operation in the MDI mode:

- 1. Program contents display screen
- 2. Current block display screen
- 3. Next block display screen
- 4. Program check screen
- 5. Program screen for MDI operation

11.2.1 Program Contents Display

Displays the program currently being executed in AUTO mode.

Procedure for displaying the program contents

- **1** Press function **PRGRM** key to display the program.
- 2 Press soft key [PRGRM]. The cursor is positioned at the block currently being executed.

```
PROGRAM
                                 O2000 N0130
02000 ;
N100 G92 X0 Y0 Z50. ;
N110 G91 G00 Y50. ;
N120 Z-50. ;
N130 G41 G17 H1 G01 X20. F3000 ;
N140 G02 J-25.5 ;
N150 X20. ;
N160 G02 X12.5 Y12.5 R12.5 ;
N170 G01 Y40. ;
N180 X30. Y30. ;
N190 G40 X50. ;
 16:59:40
                       BUF AUTO
[ PRGRM ][CURRNT][ NEXT ][ CHECK ][
                                         1
```

11.2.2 Current Block Display Screen

Displays the block currently being executed and modal data in the AUTO or MDI mode.

Procedure for displaying the current block display screen

- 1 Press function key PRGRM
- 2 Press soft key **[CURRNT]**. The block currently being executed and modal data are displayed.

| PROGRA | м | | | | | 02000 | N00130 |
|---------|------|----------|-------|---------|-----|-------|--------|
| | (CU | RRNT) | | (MODA | L) | | |
| | x | 20.000 | G67 | G01 | F | | 3000 |
| G01 | F | 3000 | G54 | G17 | R | | |
| G17 | н | 1 | G64 | G91 | Р | | |
| | | | G69 | G22 | Q | | |
| | | | G15 | G94 | н | | |
| | | | G25 | G21 | М | | |
| | | | | G41 | s | | |
| G41 | | | | G49 | т | | |
| | | | | G80 | | | |
| G80 | | | | G98 | s | | |
| | | | | G50 | | | |
| | | | DUB | | ~ | | |
| | | | BOL | AUT | 0 | | |
| [PRGRM |][| CURRNT] | [NEX | XT] | [(| CHECK |][] |
| | | | | | | | |

— 390 —

11.2.3 Next Block Display Screen

Displays the block currently being executed and the block to be executed next in the AUTO or MDI mode.

Procedure for displaying the next block display screen

- 1 Press function key PRGRM
- 2 Press soft key [NEXT].

The block currently being executed and the block to be executed next are displayed.

.

| · | | | | |
|---------|----------|-----------------|------------|-------------|
| PROG | RAM | | | 02000 N0130 |
| | (CURRNT) | | | |
| | х | 20.000 | J | -25.500 |
| G | 01 F | 3000 | G02 | |
| G | 17 н | 1 | | |
| G | 41 | | | |
| G | 80 | | G80 | |
| | | ו 11 מיזארוררוז | BUF AUTO | |
| L PRGRM | 1110 | OKKNI][] | NEAL JI CH | |

11.2.4 Program Check Screen

Displays the program currently being executed, current position of the tool, and modal data in the AUTO mode.

Procedure for displaying the program check screen

- **1** Press function key PRGRM
- 2 Press soft key [CHECK].

The program currently being executed, current position of the tool, and modal data are displayed.

| PROGRAM | CHECK | | | C | 2000 | N0130 | |
|---------|------------|---------|---------|--------|--------|-------|---|
| N130 C | 341 G17 H1 | L G01 2 | K20. F3 | 3000 ; | | | |
| N140 C | €02 J-25.5 | 5; | | | | | |
| N150 X | (20.0 ; | | | | | | |
| N160 C | 302 X12.5 | ¥12.5 | R12.5 | ; | | | |
| (RE | LATIVE) | (DIST | TO GO) | ((| G) | | |
| х | 17.600 | х | 2.400 | G01 G | 21 G5 | 0 | |
| Y | 50.000 | Y | 0.000 | G17 G | 41 G6 | 7 | |
| Z | 0.000 | z | 0.000 | G91 G | 49 G54 | 4 | |
| | | | | G22 G | 80 G64 | 4 | |
| | | | | G94 G | 98 G69 | 9 | |
| F | 3000 | P | | н 1 | S | | |
| R | | Q | | м | т | | |
| | | | | | | | |
| ACT.F | 3000 |) MM/1 | vr | | | | |
| | | | | | | | |
| | | | BOF | AUTO | | | |
| [PRGRM |][CURR | NT][| NEXT] | [CHE | CK][| |] |
| | | | | | | | |

Explanations

• Program display

For the program currently being executed, the block currently being executed is displayed first.

• **Current position display** The position in the workpiece coordinate system or relative coordinate system and the remaining distance are displayed. The absolute positions and relative positions are switched by parameter No. 028 #0.

11.2.5 Program Screen for MDI Operation

Displays the program input from the MDI and modal data in the MDI mode.

Procedure for displaying the program screen for MDI operation

Procedure

- **1** Press function key PRGRM
- 2 Press soft key [MDI]. The program input from the MDI and modal data are displayed.
 - (1) MDI operation A

| | | | | | ١ |
|------------|-----------|---------|-------|-------|---|
| PROGRAM | | | 02000 | N0130 | , |
| (MDI) | | (MOD | AL) | | |
| х | 10.500 | | F | 2000 | |
| Y | 200.500 | G00 | R | | |
| | | G17 | Р | | |
| | | G90 | Q | | |
| | | G94 | н | 2 | |
| | | G21 | М | | |
| | | G40 | S | 15 | |
| | | G49 | т | 02 | |
| | | G80 | | | |
| | | G98 | | | |
| | | G67 | | | |
| ADDRESS | | | | | |
| | | MDI | : | | |
| [PRGRM][| CURRNT][| NEXT][| MDI] | [] | |
| | | | | | Ϊ |

Explanations

• MDI operation

See 4.2 for MDI operation.

11.3 SCREENS DISPLAYED BY FUNCTION KEY (IN THE EDIT MODE)

This section describes the screens displayed by pressing function key $\[PRGRM]$ in the EDIT mode. Function key $\[PRGRM]$ in the EDIT mode can display the program editing screen and the library screen (displays memory used and a list of programs).

11.3.1 Displaying Memory Used and a List of Programs Displays the number of registered programs, memory used, and a list of registered programs.

Procedure for displaying memory used and a list of programs

- 1 Select the **EDIT** mode.
- 2 Press function key PRGRM
- **3** Press soft key **[LIB]**.

| | |) |
|----------------------|-------------------------------------|---|
| | PROGRAM 01224 N0000 | |
| | SYSTEM EDITION 0471 - 04 | |
| | PROGRAM NO. USED : 14 FREE : 49 | |
| | MEMORY AREA USED : 275 FREE : 3820 | |
| | PROGRAM LIBRARY LIST | |
| | 00010 02000 00020 00030 00200 00300 | |
| | 00555 01200 00777 01234 00040 00050 | |
| | 01969 01224 | |
| | 01909 01221 | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | > | |
| | | |
| | [PRGRM][CONDNS][][][] | J |
| | | |

•

Explanations

| Details of memory used | PROGRAM NO. USE PROGRAM NO. USED FREE | D : : | The number of the programs registered (including the subprograms) The number of programs which can be registered additionally. |
|------------------------|---|--------------------|---|
| | MEMORY AREA US | ED | |
| | MEMORY AREA USED | : | The capacity of the program memory in which data is registered (indicated by the number of characters). |
| | FREE | : | The capacity of the program memory which can be used additionally (indicated by the number of characters). |

Program library list

Program Nos. registered are indicated.

Also, the program name can be displayed in the program table by setting parameter No. 040#0.

| | | ١. |
|---|------------------------------------|----|
| | PROGRAM 01224 N0000 | |
| | SYSTEM EDITION 0471 - 04 | |
| | PROGRAM NO. USED : 14 FREE : 49 | |
| | MEMORY AREA USED : 275 FREE : 3820 | |
| | PROGRAM LIBRARY LIST | |
| | O0010 (MACRO-GCODE.MAIN) | |
| | O2000 (MACRO GCODE.SUB1) | |
| | O0020 (TEST-PROGRAM.NO-1) | |
| | O0030 (TEST-PROGRAM.F10-MACRO) | I |
| | O0200 (TEST-PROGRAM.OFFSET) | |
| | 00300 | |
| | O0555 (INCH/MM CONVERT CHECK NO-1) | |
| | O1200 (MACRO-MCODE.MAIN) | |
| | | |
| | > | |
| | EDIT | I |
| | [PRGRM] [CONDNS] [] [] [] | J |
| • | | |

• Program name

Always enter a program name between the control out and control in codes immediately after the program number.

Up to 31 characters can be used for naming a program within the parentheses. If 31 characters are exceeded, the exceeded characters are not displayed.

Only program number is displayed for the program without any program name.



Program number Program name (up to 31 characters)

Software series

 Order in which programs are displayed in the program library list Software series of the system is displayed.

It is used for maintenance ; user is not required this information.

Programs are displayed in the same order that they are registered in the program library list. However, if bit 4 of parameter 040 is set to 1, programs are displayed in the order of program number starting from the smallest one.

• Free area

Any unused areas created by background editing are counted as used program and used memory areas. When memory is reorganized (refer to 9.7), these unused areas are counted as free areas.

the Tool Offset Value

| 11.4 SCREENS DISPLAYED BY FUNCTION KEY | Press function key $\begin{bmatrix} MENU \\ OFSET \end{bmatrix}$ to display or set tool compensation values and other data. This section describes how to display or set the following data: | | |
|--|---|--|--|
| | Tool offset value Workpiece origin offset value Custom macro common variables | | |
| 11.4.1 Setting and Displaying | Tool length offset values, and cutter compensation values are specified by D codes or H codes in a program. Compensation values corresponding to D codes or H codes are displayed or set on the screen. | | |

Procedure for setting and displaying the cutter compensation value

- 1 Press function key
- 2 Press soft key **[OFFSET]**. The screen varies according to the type of tool offset memory.

| OFFSET | | | 01224 N0000 | |
|------------|----------------|-----|-------------|--|
| NO. | DATA | NO. | DATA | |
| 001 | | 009 | 0.000 | |
| 002 | 0.000 | 010 | 12.269 | |
| 003 | 5.000 | 011 | 10.230 | |
| 004 | 0.000 | 012 | -11.265 | |
| 005 | 12.580 | 013 | -8.562 | |
| 006 | 0.000 | 014 | 0.000 | |
| 007 | 0.000 | 015 | 0.000 | |
| 008 | 0.000 | 016 | 0.000 | |
| ACTUAL POS | SITION (RELATI | VE) | | |
| х | 0.000 | Y | 0.000 | |
| Z | 0.000 | | | |
| | | | | |
| NO. 013 = | | | | |
| | | MDI | | |
| [OFFSET |][MACRO][|][| WORK][| |
| | | | | |

Tool offset memory A

3 Move the cursor to the compensation value to be set or changed using page keys and cursor keys. Press NO. key and enter the compensa-

tion number for the compensation value to be set or changed and then press key.

4 Enter a compensation value, and Key.

| Decimal point input | A decimal point can be used when entering a compensation value. |
|--|--|
| Other method | An external input/output device can be used to input or output a cutter compensation value. See Chapter 9. A tool length offset value can be set by measuring the tool length as described in the next subsection. |
| Tool offset memory | There are tool offset memories A and C, which are classified as follows: |
| | Tool offset memory A Tool geometry compensation and tool wear compensation are treated the same. |
| | Tool offset memory C Offset memory is divided into tool length offset memory (H) and cutter offset memory (D). It is not divided into a geometry offset area and wear offset area. |
| Disabling entry of compensation values | The entry of compensation values may be disabled by setting bit 0 and 1 of parameter 078 (not applied to tool offset memory A). |
| Incremental input | The current compensation value can be incremented or decremented if bit 4 of parameter 001 is specified accordingly. If this is selected, key in a desired increment or decrement. |

11.4.2 Displaying and Setting the Workpiece Origin Offset Value

Displays the workpiece origin offset for each workpiece coordinate system (G54 to G59) and external workpiece origin offset. The workpiece origin offset and external workpiece origin offset can be set on this screen.

Procedure for Displaying and Setting the Workpiece Origin Offset Value

- 1 Press function key
- Press soft key [WORK].
 The workpiece coordinate system setting screen is displayed.

| WORK CO | ORDINATES | 00001 N00000 |
|----------|----------------|--------------|
| NO. | (SHIFT) | NO. (G55) |
| _00 | х | 02 X 0.000 |
| | Y 0.000 | Y 0.000 |
| | z 0.000 | z 0.000 |
| NO. | (G54) | NO. (G56) |
| 01 | x 0.000 | 03 X 0.000 |
| | Y 0.000 | Y 0.000 |
| | Z 0.000 | z 0.000 |
| ADRS. | | |
| | | MDI |
| [OFFSET | [] [MACRO] [|][WORK][|
| | | |

3 The screen for displaying the workpiece origin offset values consists of two or more pages. Display a desired page in either of the following two ways:

| Press the page up | | or page down | ₽ | key. | |
|-------------------|--|--------------|---|------|--|
|-------------------|--|--------------|---|------|--|

Press NO. key and nter the workpiece coordinate system number (0: external workpiece origin offset, 1 to 6: workpiece coordinate systems G54 to G59, and then press NPUT key.

- 4 Move the cursor to the workpiece origin offset to be changed.
- 5 Press the address key corresponding to the desired axis.
- 6 Enter a desired value by pressing numeric keys, then press key . The entered value is specified in the the workpiece origin offset value.
- 7 Repeat 4 and 6 to change other offset values.

11.4.3 **Displaying and Setting Custom Macro Common Variables**

Displays common variables (#100 to #149 and #500 to #531). When the absolute value for a common variable exceeds 99999999, ******* is displayed. The values for variables can be set on this screen. Relative coordinates can also be set to valiables.

Procedure for displaying and setting custom macro common variables

- 1 Press function key MENU OFSET
- Press the soft key [MACRO]. The following screen is displayed: 2

| VARIABI | E | | 01224 | N0000 |
|---------|-----------|------------|------------|-------|
| NO. | DATA | NO. | DATA | |
| 100 | 00010.000 | 108 | | |
| 101 | 00020.000 | 109 | 02564.146 | |
| 102 | 00156.870 | 110 | | |
| 103 | 0.000 | 111 | 25000.000 | |
| 104 | 0.000 | 112 | 0.000 | |
| 105 | 01523.579 | 113 | | |
| 106 | 0.000 | _114 | 30000.000 | |
| 107 | 00458.256 | 115 | | |
| ACTUAL | POSITION | (RELATIVE) | | |
| х | 0.000 | Y | 0.000 | |
| Z | 0.000 | | | |
| NO. 114 | = | | | |
| | | | MDI | |
| [OFFSE | T][MACF | RO][|][WORK][|] |

- 3 Move the cursor to the variable number to set using either of the following methods:
- \blacksquare Press the $\mid NO \mid$ key and enter the variable number. And then press INPUT key. ■ Move the cursor to the variable number to set by pressing page keys 1 and/or and cursor keys and/or **†** | Enter data with numeric keys and press INPUT key. To set a relative coordinate in a variable, press address key X Z, while holding down the EOB key. Then press Y , or INPUT key. **6** To set a blank in a variable, just press |INPUT| key after |CAN| key. The value field for the variable becomes blank. (Only for custom

macro B)

4

5

11.5 SCREENS DISPLAYED

Parameters must be set to determine the specifications and functions of the machine in order to fully utilize the characteristics of the servo motor or other parts.

This chapter describes how to set parameters on the MDI panel. Parameters can also be set with external input/output devices such as the Handy File (see Chapter 9).

If the DGNOS function key is pressed, the following data can be displayed and set:

·Pitch error compensation data

•Setting data

Run time and parts count

·Self-diagnostic data

See Chapter 7 for the diagnostic screens.

| 11.5.1 Displaying and Setting Parameters | Parameters are set to determine the specifications and functions of the machine in order to fully utilize the characteristics of the servo motor. The setting of parameters depends on the machine. Refer to the parameter list prepared by the machine tool builder. Normally, the user need not change parameter setting. |
|--|--|
|--|--|

Procedure for displaying and setting parameters

- 1 When set the paameter, set 1 for **PARAMETER WRITE (PWE)** to enable writing. See the procedure for enabling/disabling parameter writing described below.
- 2 Press function key DGNOS ARAM .
- **3** Press soft key **[PARAM]** to display the parameter screen.

| / | | | | | |
|---|------------|------------|------|-----------|------|
| | PARAMETER | | | 01224 N | 0000 |
| | NO. | DATA | NO. | DATA | |
| | 0001 | 00000000 | 0011 | 00000000 | |
| | 0002 | 00000011 | 0012 | 0000000 | |
| | 0003 | 00000000 | 0013 | 00000000 | |
| | 0004 | 01110111 | 0014 | 00000100 | |
| | 0005 | 01110111 | 0015 | 0000000 | |
| | 0006 | 01110111 | 0016 | 0000000 | |
| | 0007 | 00000000 | 0017 | 01111111 | |
| | 0008 | 00000000 | 0018 | 0000000 | |
| | 0009 | 00000000 | 0019 | 1000000 | |
| | 0010 | 10000000 | 0020 | 00000000 | |
| | | | | | |
| | NO. 0001 = | | | | |
| | | | MDI | | |
| | [PARAM] | [DGNOS][|][| SV-PRM][| 1 |
| | | | | | |

4 Move the cursor to the parameter number to be set or displayed in either of the following ways:

| Press the $NO.$ key and enter the variable number. And then pres | s |
|---|---|
| INPUT key. | |
| Move the cursor to the parameter number using the page keys, \uparrow |] |
| and $[]$, and cursor keys, \uparrow and $[]$. | |

5 To set the parameter, enter a new value with numeric keys and press key. The parameter is set to the entered value and the value is displayed.

6 When the parameter is set, set 0 for **PARAMETER WRITE (PWE)** to disable writing.

Procedure for enabling/displaying parameter writing

- 1 Select the **MDI** mode or enter state emergency stop.
- 2 Press function key DGNOS PARAM
- 3 Press soft key [PARAM] to display the setting screen.

| PARAMETER | 01224 N0000 |
|----------------------|----------------|
| (SETTING 2) | |
| PWE = 1 (0:DISABLE | 1:ENABLE) |
| REV4 = 0 | |
| TAPEF = 0 | |
| (SEQUENCE STOP) | |
| PRGNO = 0 | |
| SEQNO = 0 | |
| PART TOTAL = 17 | |
| PART REQUIRED = 50 | |
| PART COUNT = 17 | |
| RUN TIME 0H 4M CYCLE | TIME OH OM 2S |
| NO. PWE = = | |
| | |
| [PARAM][DGNOS][|][SV-PRM][] |

- 4 Move the cursor to **PARAMETER WRITE (PWE)**using cursor keys.
- **5** Press 1, then press ||| key to enable parameter writing. At this time, the CNC enters the P/S alarm state (No. 100).
- 6 After setting parameters, move the cursor to PARAMETER WRITE
 (PWE) and press 0 key, then press key.
- Depress the RESET key to release the alarm condition. If alarm No.
 000 has occurred, however, turn off the power supply and then turn it on, otherwise the alarm is not released.

Explanations

- Setting parameters with external input/output devices
- Parameters that require turning off the power

See Chapter 8 for setting parameters with external input/output devices such as the Handy File.

Parameters are not effective until the power is turned off and on again after they are set. Setting such parameters causes alarm 000. In this case, turn off the power, then turn it on again.

11.5.2 Displaying and Setting Pitch Error Compensation Data

If pitch error compensation data is specified, pitch errors of each axis can be compensated per axis.

Pitch error compensation data is set for each compensation point at the intervals specified for each axis. The origin of compensation is the reference position to which the tool is returned.

The pitch error compensation data is set according to the characteristics of the machine. The content of this data varies according to the machine model. If it is changed, the machine accuracy is reduced.

In principle, the end user must not alter this data.

Pitch error compensation data can be set with external devices such as the Handy File (see Chapter 9). Compensation data can also be written directly with the MDI panel.

The following parameters must be set for pitch error compensation. Set the pitch error compensation value by these parameters.

In the following example, 33 is set for the pitch error compensation point at the reference position.



- Number of the pitch error compensation point at the reference position $(\Box : axis number) : Parameter No. \Box 000$
- Pitch error compensation magnification : Parameter No. 011#0, #1
- Interval of the pitch error compensation points : Parameter No. 0712 to 0715

·Setting compensation value : Parameter No. $\Box 001 + compensation points$

Explanations

| Compensation point number | 128 compensation points from No. 0 to 127 are available for each axis. Specify the compensation number for the reference position of each axis in the corresponding parameter (Parameter n000, n: axis number). | |
|---|--|--|
| Compensation value | Specify the compensation value in the corresponding parameter (Parameter n001 + compensation point number, n: axis number). | |
| Restrictions | | |
| Compensation value range | Compensation values can be set within the range from $-7 x$ compensation magnification (output unit) to $+7 x$ compensation magnification (detection unit). The compensation magnification can be set 1, 2, 4, 8 in parameter No. 011#0, #1. The units of the compensation value can be changed to the detection units if bit 7 of parameter 035 is specified accordingly. | |
| Intervals of compensation points | The pitch error compensation points are arranged with equally spaced. Set the space between two adjacent positions for each axis to the parameter (No. 712 to 715). Valid data range is 0 to 99999999. The minimum interval between pitch error compensation points is limited and obtained from the following equation: Minimum interval of pitch error compensation points = maximum feedrate (rapid traverse rate) / 1875 Unit: mm, inches, deg, and mm/min, inches/min, deg/min | |
| Pitch error compensation of the rotary axis | For the rotating axis, the interval between the pitch error compensation points shall be set to one per integer of the amount of movement (normal 360°) per rotation. The sum of all pitch error compensation amounts p rotation must be made to 0. | |
| • Conditions where pitch | Note that the pitch error is not compensated in the following cases: | |
| error compensation is not performed | • When the machine is not returned to the reference position after turning on the power. If an absolute–position detector is provided and if the reference position has been determined, pitch error compensation is carried out. | |
| | It the interval between the interval encodered term α | |

· If the interval between the pitch error compensation points is 0.

Examples

- For linear axis (X axis)
- \cdot Machine stroke: -400 mm to +800 mm
- · Interval between the pitch error compensation points: 50 mm
- \cdot No. of the compensation point of the reference position: 40

If the above is specified, the No. of the farthest compensation point in the negative direction is as follows:

No. of the compensation point of the reference position – (Machine stroke on the negative side/Interval between the compensation points) + 1 = 40 - 400/50 + 1 = 33

No. of the farthest compensation point in the positive direction is as follows:

No. of the compensation point of the reference position + (Machine stroke on the positive side/Interval between the compensation points) = 40 + 800/50 = 56

The correspondence between the machine coordinate and the compensation point No. is as follows:



Therefore, set the parameters as follows:

| Parameter | Setting value |
|---|---------------|
| 1000 : Compensation number for the reference position | 40 |
| 011#0, #1 : Compensation magnification | #0=0, #1=0 |
| 0712 : Interval between pitch error compensation points | 50000 |

The compensation amount is output at the compensation point No. corresponding to each section between the coordinates.

| Compensation position number | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 4 | 1 |
|------------------------------|------|------|------|------|------|------|------|------|--------|------|
| Parameter | 1034 | 1035 | 1036 | 1037 | 1038 | 1039 | 1040 | 1041 | 10 | 42 |
| Compensation value | +2 | +1 | +1 | -2 | 0 | -1 | 0 | -1 | + | -2 |
| | | | | | | | | r | | 1 |
| Compensation position number | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | \sum | 56 |
| Parameter | 1043 | 1044 | 1045 | 1046 | 1047 | 1048 | 1049 | 1050 | \sum | 1057 |
| Compensation value | +1 | 0 | -1 | -1 | -2 | 0 | +1 | +2 | \sum | 1 |

The following is an example of the compensation amounts.



For rotary axis (C axis)

 \cdot Amount of movement per rotation: 360°

• Interval between pitch error compensation points: 45°

 \cdot No. of the compensation point of the reference position: 60

If the above is specified, the No. of the farthest compensation point in the negative direction for the rotating axis is always equal to the compensation point No. of the reference position.

The No. of the farthest compensation point in the positive direction is as follows:

No. of the compensation point of the reference position + (Move amount per rotation/Interval between the compensation points)

= 60 + 360/45 = 68

0.0 Reference position 45.0 315.0 (61) (68) (60) (67) (62) (+) 90.0 270.0 (63) (66) (64) (65) 225.0 135.0 180.0 Compensation values are output at the positions indicated by \bigcirc .

The correspondence between the machine coordinate and the compensation point No. is as follows:

If the sum of the compensation values for positions 61 to 68 is not 0, pitch error compensation values are accumulated for each rotation, causing positional deviation.

The same value must be set for compensation points 60 and 68. Therefore, set the parameters as follows:

| Parameter | Setting value |
|--|---------------|
| 4000 : Compensation number for the reference position | 60 |
| 011#0, #1 : Compensation magnification | #0=0, #1=0 |
| 715 : Interval between pitch error compensation points | 45000 |

The following is an example of compensation amounts.

| Compensation position number | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 |
|------------------------------|------|------|------|------|------|------|------|------|------|
| Parameter | 4061 | 4062 | 4063 | 4064 | 4065 | 4066 | 4067 | 4068 | 4069 |
| Compensation value | +1 | -2 | +1 | +3 | -1 | -1 | -3 | +2 | +1 |



— 408 —

| 11.5.3 | Data such as the TV check flag and punch code is set on the setting data |
|---|--|
| Displaying and Entering Setting Data | screen. On this screen, the operator can also enable/disable parameter writing, enable/disable the automatic insertion of sequence numbers in program editing. |
| | See Chapter 9 for automatic insertion of sequence numbers. This subsection describes how to set data. |

Procedure for setting the setting data

Procedure

- 1 Select the **MDI** mode.
- 2 Press function key DGNOS PARAM
- **3** Press soft key **[PARAM]** to display the parameter screen. This screen consists of several pages.

```
Press page key \uparrow or \downarrow until the desired screen is displayed.
```

An example of the setting data screen is shown below.

| PARAMETER | | 0122 | 24 N0000 |
|------------------------|---------------|-----------|----------|
| (SETTING 1) | | | |
| $\underline{REVX} = 0$ | | | |
| REVY = 0 | | | |
| TVON = 0 | | | |
| ISO = 0 | (0:EIA 1:ISO) | | |
| INCH = 0 | (0:MM 1:INCH |) | |
| I/O = 0 | | | |
| ABS = 0 | (0:INC 1:ABS) | | |
| SEQ = 0 | | | |
| | | | |
| | | | |
| NO. REVX = | | | |
| | | MDI | |
| [PARAM][I | GNOS][|][SV-PRM | 1[] |

- 5 Enter a new value and press INPUT key.

Contents of settings

- PARAMETER WRITE (PWE)
- TV CHECK (TVON)

Setting whether parameter writing is enabled or disabled.

- 0: Disabled
- 1 : Enabled

Setting to perform TV check.

- 0: No TV check
- 1: Perform TV check

— 409 —

| • PUNCH CODE (ISO) | Setting code when data is output through reader puncher interface.0: EIA code output1: ISO code output |
|--|--|
| • INPUT UNIT (INCH) | Setting a program input unit, inch or metric system 0 : Metric 1 : Inch |
| • I/O CHANNEL (I/O) | Using channel of reader/puncher interface. 0 : Channel 0 1 : Channel 1 |
| • SEQUENCE NUMBER (SEQ) | Setting of whether to perform automatic insertion of the sequence number or not at program edit in the EDIT mode. 0 : Does not perform automatic sequence number insertion. 1 : Perform automatic sequence number insertion. |
| • MIRROR IMAGE (REVX, REVY, REV4) | Setting of mirror image ON/OFF for each axes.0 : Mirror image off1 : Mirror image on |
| MDI travel command (ABS) | Specifies whether a travel command executed in MDI operation is absolute or incremental. 0: Incremental command 1: Absolute command |

11.5.4 Displaying and Setting Run Time,Parts Count

Various run times, the total number of machined parts, number of parts required, and number of machined parts can be displayed. The data except for the total number of machined parts can be set on this screen .

Procedure for Displaying and Setting Run Time and Parts Count

- 1 Select the MDI mode.
- 2 Press function key DGNOS
- **3** Press chapter selection soft key **[PARAM]**.
- 4 Press page key $\left[\stackrel{\bullet}{\underset{PAGE}{\clubsuit}} \right]$ or $\left[\stackrel{PAGE}{\underset{\bullet}{\clubsuit}} \right]$ several times until the following screen is displayed.

| PARAME | TE | R | | | | 0122 | 24 | N0000 |
|--------|----|------|--------|--------|----|--------|----|-------|
| (SETT | IN | G 1) |) | | | | | |
| REVX | = | 0 | | | | | | |
| REVY | = | 0 | | | | | | |
| TVON | = | 0 | | | | | | |
| ISO | = | 0 | (0:EIA | 1:ISO) | | | | |
| INCH | = | 0 | (0:MM | 1:INCH |) | | | |
| I/O | = | 0 | | | | | | |
| ABS | = | 0 | (0:INC | 1:ABS) | | | | |
| SEQ | = | 0 | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| NO. RE | vx | = | | | | | | |
| | | | | | ME | I | | |
| [PAR | м |][| DGNOS |][|][| SV-PRM |] | []] |

| PARAMETER | 012 | 24 N0000 |
|---|---|------------|
| (SETTING 2) PWE = 1 (0:1 REV4 = 0 TAPEF = 0 | DISABLE 1:ENABLE) | |
| (SEQUENCE STOP) PRGNO = 0 SEQNO = 0 | | |
| PART TOTAL PART REQUIRED PART COUNT RUN TIME | = 17 = 50 = 17 OH 4M CYCLE TIM | e oh om 25 |
| NO. PWE = | | |
| [PARAM][DGNOS | JL JL SV-PRM | 11 1 |

5 To set the number of parts required, move the cursor to PARTS REQUIRED and enter the number of parts to be machined.

| | Item | Maximum value | ltem | Maximum value |
|--|---|--|--|--|
| Time settings | Neither negative v table can be set. | alue nor the value | exceeding the valu | e in the following |
| Run time and part count settings | Negative value car is valid from 0 to 3 Negative value ma | nnot be set. Also, t 59. ay not be set to the | he setting of "M" a e total number of r | and "S" of run time nachined parts. |
| Restrictions | | | | |
| Explanations • Usage | When the comma machined parts and Therefore, create t the processing of to the parameter (I manner. Also, it i executed (paramet issued by machine | nd of M02 or M3 d the number of ma he program so that one part is comple NO. 219) is execu s possible to disal er (No. 040#3) is a tool builders. | 30 is executed, the achined parts are in t M02 or M30 is ex- eted. Furthermore ted, counting is m tole counting even set to 1). For detail | e total number of cremented by one. accuted every time of an M code set nade in the similar if M02 or M30 is ils, see the manual |
| | feed hold time. T performed at reset | This is automatical state. It is preset | lly preset to 0 wh to 0 even when po | en a cycle start is ower is removed. |
| • CUTTING TIME | Indicates the run ti | me of one automa | tic operation, excl | uding the stop and |
| • FREE PURPOSE | This value can be u flows Refer to the | used, for example, a | as the total time dur | ring which coolant builder for details |
| • OPERATING TIME | Indicates the total stop and feed hold | run time during a time. | automatic operatio | on, excluding the |
| • PARTS COUNT | This value is increated by parameter 219 reaches the number machine tool build | mented by one when is executed. In er of parts required ler for details. | en M02, M30, or ar general, this valu l. Refer to the ma | n M code specified e is reset when it nual issued by the |
| • PARTS REQUIRED | It is used for settir When the "0" is Also, its setting ca | ng the number of r set to it, there is r in be made by the | nachined parts required in the parts required in the parameter (NO. 6 | uired. e number of parts. 00). |
| PARTS TOTAL | This value is increated by parameter 219 Set the value in pa | mented by one who is executed. This trameter 779. | en M02, M30, or an s value cannot be s | n M code specified set on this screen. |
| Dianlay itoma | | | | |
| | date or time, t | hen press INPUT ke | ey. | ie, enter a new |
| | 6 To set the cloc | ck, move the curso | or to DATE or TIM | IE, enter a new |

| ltem | Maximum value | Item | Maximum value |
|-------|---------------|--------|---------------|
| Year | 99 | Hour | 23 |
| Month | 12 | Minute | 59 |
| Day | 31 | Second | 59 |

11.6 SCREENS DISPLAYED BY FUNCTION KEY

The alarm message and operator message can be displayed by pressing the $\begin{bmatrix} OPR \\ ALARM \end{bmatrix}$ key. The software operator's panel can also be displayed and specified. For details of how to display the alarm message, see Chapter 7.

11.6.1 Displaying Operator Message

The operator message function displays a message on the PMC screen.

Procedure for Displaying Operator Message

- **1** Press function key OPR ALARM
- 2 press soft key [MSG].

| - | | | | | | | | | - |
|---|--------|--------|------|----|-----|-----|-------|-------|---|
| | OPERA | TOR ME | SSAG | | | | 00000 | N0000 | |
| | NO. | 2001 | | | | | | | |
| | COOLAN | IT ON | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | 1 | MDI | | | |
| | [ALA | RM][| OPR |][| MSG |][|][|] | |
| | | | | | | | | | |

Explanations

- If the operator message display function is enabled, the screen is automatically switched to the operator message screen.
- A PMC command can also be used to clear the operator message.
- For details of the contents of the operator message, and how to clear the message, refer to the manual provided by the machine tool builder.

11.6.2 Displaying and Setting the Software Operator's Panel

With this function, functions of the switches on the machine operator's panel can be controlled from the CRT/MDI panel. Jog feed can be performed using numeric keys.

Procedure for displaying and setting the software operator's panel

1 Press function key OPR ALARM

)()(OPR)()(

Continuous menu key

Press soft key [OPR].
 The screen consists of several pages.

Press page key \uparrow or \downarrow until the desired screen is displayed.

| MODE : MDI | AUTO ED | IT HNDI | JOG | ZRN | |
|----------------|-----------|-------------|---------|-----|---|
| HANDLE AXIS | : нх н | Y HZ | | | |
| HANDLE MULT. | : 1 * | 10 *100 | | | |
| RAPID OVRD. | : 100% | 50% 2 | 25% 📕 F | 0 | |
| JOG FEED | : 500 | MM/MIN | | | |
| | | , | * * | | |
| FEED OVRD. | : 100% | | | | |
| | | **** | *** | | |
| ACTUAL POSITIO | ON (ABSOL | JTE) | | | |
| x 0.0 | 00 | Y C | 0.000 | | |
| z 0.0 | 00 | | | | |
| | | | | | |
| | | MD | I | | |
| | ו אפר | MSG 1 | г | 11 | 1 |

| OPERATOR'S PAN | EL (| D1224 N0000 |
|-----------------|------------|-------------|
| BLOCK SKIP : | OFF | ON |
| SINGLE BLOCK : | OFF | ON |
| MACHINE LOCK : | OFF | ON |
| DRY RUN : | OFF | ON |
| PROTECT KEY : | PROTECT | RELEASE |
| FEED HOLD : | OFF | ON |
| ACTUAL POSITION | (ABSOLUTE) | |
| x 0.000 | Y 0.0 | 000 |
| z 0.000 | | |
| | MDI | |
| [ALARM][OPR |][MSG][|][] |

4 Move the cursor to the desired switch by pressing cursor key \uparrow

or 📕

- 5 Push the cursor move key $(\leftarrow /)$ or $(\rightarrow = 0^{-1})$ to match the mark \blacksquare to an arbitrary position and set the desired condition.
- 6 Press one of the following arrow keys to perform jog feed. Press the function key w5 together with an arrow key to perform jog rapid traverse.

| ~ 7 | t 8 | ₹9 |
|------------|------------|------------|
| -4 | ••5 | → 6 |
| 1 | 42 | ß |

these switches, refer to the manual issued by machine tool builder.

Explanations

| • Valid operations | The valid operations on the software operator's panel are shown below. Whether to use the MDI key or machine operator's panel for each group of operations can be selected by parameter 017#0 to #6. Group1 : Mode selection Group2 : Selection of jog feed axis, jog rapid traverse Group3 : Selection of manual pulse generator feed axis, selection of manual pulse magnification x1, x10, x100 Group4 : Jog federate, federate override, rapid traverse override Group5 : Optional block skip, single block, machine lock, dry run Group6 : Protect key Group7 : Feed hold |
|--|--|
| ● Display | The groups for which the machine operator's panel is selected by parameter 017#0 to #6 are not displayed on the software operator's panel. |
| Screens on which jog feed is valid | When the screen indicates other than the software operator's panel screen and diagnostic screen, jog feed is not conducted even if the arrow key is pushed. |
| Jog feed and arrow keys | The feed axis and direction corresponding to the arrow keys can be set with parameters (Nos. 130 to 137). |
| General purpose switches | Eight optionally definable switches are added as an extended function of the software operator's panel. The name of these switches can be set by parameters as character strings of max. 8 characters. For the meanings of |

11.7 DISPLAYING THE PROGRAM NUMBER, SEQUENCE NUMBER, AND STATUS, AND WARNING MESSAGES FOR DATA SETTING

11.7.1 Displaying the Program Number and Sequence Number The program number, sequence number, and current CNC status are always displayed on the screen except when the power is turned on or a system alarm occurs.

This section describes the display of the program number, sequence number, and status.

The program number and sequence number are displayed at the top right on the screen as shown below.

| PROGRAM | 02000 N0130 | Sequence |
|----------------------------------|-------------|------------|
| 02000 ; | | No. |
| N100 G92 X0 Y0 Z50. ; | | Program No |
| N110 G91 G00 Y50. ; | | |
| <u>N</u> 120 Z-50. ; | | |
| N130 G41 G17 H1 G01 X20. F3000 ; | | |
| N140 G02 J-25.5 ; | | |
| N150 X20. ; | | |
| N160 G02 X12.5 Y12.5 R12.5 | | |
| N170 G01 Y40. ; | | |
| N180 X30. Y30 ; | | |
| N190 G40 X50.; | | |
| < | | |
| 02:55:13 EDIT | | |
| [PRGRM][LIB][I/O][|][C.A.P.] | |
| | | 1 |

The program number and sequence number displayed depend on the screen and are given below:

On the program screen in the EDIT mode on Background edit screen : The program No. being edited and the sequence number just prior to the cursor are indicated.

Other than above screens :

The program No. and the sequence No. executed last are indicated.

Immediately after program number search or sequence number search :

Immediately after the program No. search and sequence No. search, the program No. and the sequence No. searched are indicated.

11.7.2 **Displaying the Status** and Warning for Data Setting

The current mode, automatic operation state, alarm state, and program editing state are displayed on the next to last line on the CRT screen allowing the operator to readily understand the operation condition of the system.

Explanations

• Description of each display

| display | NOT READY |
|--|---|
| | ALARM BAT BUF EDIT INPUT |
| | (Display soft keys) |
| | |
| Current mode | MDI : Manual data input |
| | AUTO : Automatic operation |
| | RMT : Automatic operation (Tape operation, or such like) |
| | EDIT : Memory editing |
| | HNDL : Manual handle feed |
| | JOG : Jog feed |
| | TJOG : (TEACH IN JOG) |
| | THND : (TEACH IN HANDLE) |
| | STEP : Manual incremental feed |
| | ZRN : Manual reference position return |
| Alarm status | ALARM : Indicates that an alarm is issued. |
| | BAT : Indicates that the battery is low. |
| Other status display | INPUT : Indicates that data is being input. |
| | OUTPUT : Indicates that data is being output. |
| | SRCH : Indicates that a search is being performed. |
| | EDIT : Indicates that another editing operation is being performed |
| | (insertion, modification, etc.) |
| | COMPARE : Indicates that the program is being collated. |
| | LSK : Indicates that labels are skipped when data is input. |
| | BUF : Indicates that the block to be executed next is being read. |
| | NOT READY : Indicates that the system is in the emergency stop state. |
IV MAINTENANCE

METHOD OF REPLACING BATTERY

This chapter describes the method of replacing batteries as follows.

1.1 REPLACING CNC BATTERY FOR MEMORY BACK-UP1.2 REPLACING BATTERIES FOR ABSOLUTE PULSE CODER

1.1 REPLACING CNC BATTERY FOR MEMORY BACK-UP

When the message "BAT" appears at the bottom of the screen, replace the backup batteries for the CNC memory according to the procedure described below.

Procedure for replacing CNC battery for memory back-up

- **1** Have three commercially available fresh alkaline R20 (D) batteries handy.
- 2 Switch on the machine (CNC). (When replacing the batteries, keep the CNC switched on. If you replace the batteries with the CNC switched off, the memory contents will be lost.)
- **3** Loosen the screws for the battery case cover and remove it. For the location of the battery case, refer to the relevant operating manual issued by the machine builder.
- **4** Replace the batteries in the battery case with the fresh ones you have handy. Be careful for the orientation of the batteries. Of the three batteries, the middle one is in the orientation opposite to the other two.
- 5 After replacing the batteries, put the battery case cover back in place.
- **6** Press the |RESET| key and ensure the message "BAT" disappears.



1.2 REPLACING BATTERIES FOR ABSOLUTE PULSE CODER

If absolute pulse coder alarm 3n7 (where n is an axis number) occurs, replace the batteries (alkaline) for the absolute pulse coder according to the procedure described below.

Procedure for replacing batteries for absolute pulse coder

Procedure

- **1** Have three commercially available fresh alkaline R20 (D) batteries handy.
- 2 Switch on the machine (CNC). (When replacing the batteries, keep the CNC switched on. If you replace the batteries with the CNC switched off, the memory contents will be lost.)
- **3** Loosen the screws for the battery case cover and remove it. For the location of the battery case, refer to the relevant operating manual issued by the machine builder.
- 4 Replace the batteries in the battery case with the fresh ones you have handy. Be careful for the orientation of the batteries. (Place the two batteries in the orientation opposite to the other two as shown below.)
- 5 After replacing the batteries, put the battery case cover back in place.
- 6 Switch the machine (CNC) off and on again, and ensure that alarm 3n7 does not occur.



APPENDIX



| | | ISC |) C | ode | | | | | | EIA code | | | | Meaning | | | | | | | | |
|-----------|------------|------------|-----|------------|------------|---|------------|------------|------------|-----------|---|---|------------|------------|---|---|------------|------------|------------|-----------|------------------------------|---------------------------|
| Character | 8 | 7 | 6 | 5 | 4 | | 3 | 2 | 1 | Character | 8 | 7 | 6 | 5 | 4 | | 3 | 2 | 1 | | Without custom macro B | With custom macro B |
| 0 | | | 0 | \bigcirc | | 0 | | | | 0 | | | 0 | | | 0 | | | | Number 0 | | |
| 1 | 0 | | 0 | 0 | | 0 | | | 0 | 1 | | | | | | 0 | | | 0 | Number 1 | | |
| 2 | 0 | | 0 | 0 | | 0 | | 0 | | 2 | | | | | | 0 | | 0 | | Number 2 | | |
| 3 | | | 0 | \bigcirc | | 0 | | 0 | \bigcirc | 3 | | | | 0 | | 0 | | \bigcirc | 0 | Number 3 | | |
| 4 | \bigcirc | | 0 | \bigcirc | | 0 | 0 | | | 4 | | | | | | 0 | 0 | | | Number 4 | | |
| 5 | | | 0 | \bigcirc | | 0 | 0 | | \bigcirc | 5 | | | | 0 | | 0 | \bigcirc | | 0 | Number 5 | | |
| 6 | | | 0 | 0 | | 0 | 0 | 0 | | 6 | | | | 0 | | 0 | 0 | \bigcirc | | Number 6 | | |
| 7 | \bigcirc | | 0 | \bigcirc | | 0 | 0 | 0 | \bigcirc | 7 | | | | | | 0 | 0 | \bigcirc | 0 | Number 7 | | |
| 8 | \bigcirc | | 0 | \bigcirc | \bigcirc | 0 | | | | 8 | | | | | 0 | 0 | | | | Number 8 | | |
| 9 | | | 0 | \bigcirc | \bigcirc | 0 | | | \bigcirc | 9 | | | | 0 | 0 | 0 | | | 0 | Number 9 | | |
| А | | \bigcirc | | | | 0 | | | \bigcirc | а | | 0 | 0 | | | 0 | | | 0 | Address A | | |
| В | | 0 | | | | 0 | | 0 | | b | | 0 | 0 | | | 0 | | \bigcirc | | Address B | | |
| С | \bigcirc | 0 | | | | 0 | | 0 | \bigcirc | с | | 0 | 0 | 0 | | 0 | | \bigcirc | 0 | Address C | | |
| D | | \bigcirc | | | | 0 | 0 | | | d | | 0 | 0 | | | 0 | \bigcirc | | | Address D | | |
| E | \bigcirc | 0 | | | | 0 | 0 | | \bigcirc | е | | 0 | 0 | 0 | | 0 | 0 | | 0 | Address E | | |
| F | \bigcirc | 0 | | | | 0 | 0 | 0 | | f | | 0 | 0 | 0 | | 0 | 0 | \bigcirc | | Address F | | |
| G | | \bigcirc | | | | 0 | 0 | 0 | \bigcirc | g | | 0 | 0 | | | 0 | 0 | \bigcirc | 0 | Address G | | |
| н | | \bigcirc | | | \bigcirc | 0 | | | | h | | 0 | 0 | | 0 | 0 | | | | Address H | | |
| I | 0 | 0 | | | 0 | 0 | | | \bigcirc | i | | 0 | 0 | 0 | 0 | 0 | | | 0 | Address I | | |
| J | \bigcirc | 0 | | | \bigcirc | 0 | | 0 | | j | | 0 | | 0 | | 0 | | \bigcirc | 0 | Address J | | |
| К | | \bigcirc | | | \bigcirc | 0 | | 0 | \bigcirc | k | | 0 | | 0 | | 0 | | \bigcirc | | Address K | | |
| L | \bigcirc | 0 | | | \bigcirc | 0 | 0 | | | I | | 0 | | | | 0 | | \bigcirc | 0 | Address L | | |
| М | | 0 | | | 0 | 0 | 0 | | \bigcirc | m | | 0 | | 0 | | 0 | \bigcirc | | | Address M | | |
| N | | 0 | | | \bigcirc | 0 | 0 | 0 | | n | | 0 | | | | 0 | \bigcirc | | 0 | Address N | | |
| 0 | \bigcirc | \bigcirc | | | \bigcirc | 0 | 0 | 0 | \bigcirc | о | | 0 | | | | 0 | $^{\circ}$ | \bigcirc | | Address O | | |
| Р | | 0 | | 0 | | 0 | | | | р | | 0 | | 0 | | 0 | 0 | \bigcirc | 0 | Address P | | |
| Q | \bigcirc | 0 | | \bigcirc | | 0 | | | \bigcirc | q | | 0 | | 0 | 0 | 0 | | | | Address Q | | |
| R | \bigcirc | 0 | | \bigcirc | | 0 | | 0 | | r | | 0 | | | 0 | 0 | | | 0 | Address R | | |
| S | | \bigcirc | | \bigcirc | | 0 | | 0 | \bigcirc | s | | | 0 | \bigcirc | | 0 | | \bigcirc | | Address S | | |
| Т | \bigcirc | 0 | | 0 | | 0 | 0 | | | t | | | 0 | | | 0 | | \bigcirc | 0 | Address T | | |
| U | | 0 | | 0 | | 0 | 0 | | \bigcirc | u | | | 0 | 0 | | 0 | \bigcirc | | | Address U | | |
| V | | \bigcirc | | 0 | | 0 | 0 | 0 | | V | | | 0 | | | 0 | \bigcirc | | 0 | Address V | | |
| W | \bigcirc | \bigcirc | | \bigcirc | | 0 | \bigcirc | \bigcirc | \bigcirc | w | | | \bigcirc | | | 0 | \bigcirc | \bigcirc | | Address W | | |
| Х | \bigcirc | \bigcirc | | \bigcirc | \bigcirc | 0 | | | | х | | | \bigcirc | \bigcirc | | 0 | \bigcirc | \bigcirc | \bigcirc | Address X | | |
| Y | | \bigcirc | | 0 | \bigcirc | 0 | | | \bigcirc | у | | | 0 | 0 | 0 | 0 | | | | Address Y | | |
| Z | | 0 | | 0 | \bigcirc | 0 | | 0 | | z | | | 0 | | 0 | 0 | | | 0 | Address Z | | |

— 427 —

| | I | ISC |) cc | ode | | | | | | | | EIA | со | de | | | | | | Meaning | |
|-----------|---|-----|------------|------------|---|---|------------|------------|---|-----------|---|-----|------------|------------|---|---|---|------------|---|------------------------------|---------------------------|
| Character | 8 | 7 | 6 | 5 | 4 | | 3 | 2 | 1 | Character | 8 | 7 | 6 | 5 | 4 | | 3 | 2 | 1 | Without custom macro B | With custom macro B |
| DEL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Del | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | × | × |
| NUL | | | | | | 0 | | | | Blank | | | | | | 0 | | | | × | × |
| BS | 0 | | | | 0 | 0 | | | | BS | | | \bigcirc | | 0 | 0 | | \bigcirc | | × | × |
| HT | | | | | 0 | 0 | | | 0 | Tab | | | \bigcirc | \bigcirc | 0 | 0 | 0 | 0 | | × | × |
| LF or NL | | | | | 0 | 0 | | 0 | | CR or EOB | 0 | | | | | 0 | | | | | |
| CR | 0 | | | | 0 | 0 | \bigcirc | | 0 | | | | | | | | | | | × | × |
| SP | 0 | | \bigcirc | | | 0 | | | | SP | | | | \bigcirc | | 0 | | | | | |
| % | 0 | | \bigcirc | | | 0 | \bigcirc | | 0 | ER | | | | | 0 | 0 | | 0 | 0 | | |
| (| | | 0 | | 0 | 0 | | | | (2–4–5) | | | | 0 | 0 | 0 | | 0 | | | |
|) | 0 | | 0 | | 0 | 0 | | | 0 | (2–4–7) | | 0 | | | 0 | 0 | | 0 | | | |
| + | | | \bigcirc | | 0 | 0 | | \bigcirc | 0 | + | | 0 | \bigcirc | \bigcirc | | 0 | | | | × | |
| - | | | 0 | | 0 | 0 | \bigcirc | | 0 | - | | 0 | | | | 0 | | | | | |
| : | | | \bigcirc | \bigcirc | 0 | 0 | | 0 | | | | | | | | | | | | | |
| / | 0 | | \bigcirc | | 0 | 0 | \bigcirc | 0 | 0 | / | | | \bigcirc | \bigcirc | | 0 | | | 0 | | |
| | | | \bigcirc | | 0 | 0 | \bigcirc | 0 | | | | 0 | 0 | | 0 | 0 | | 0 | 0 | | |
| # | 0 | | 0 | | | 0 | | 0 | 0 | Parameter | | | | | | | | | | | |
| | | | | | | | | | | (No.0044) | | | | | | | | | | | |
| \$ | | | 0 | | | 0 | 0 | | | | | | | | | | | | | Δ | 0 |
| & | 0 | | 0 | | | 0 | 0 | 0 | | & | | | | | 0 | 0 | 0 | 0 | | Δ | 0 |
| • | | | 0 | | | 0 | 0 | 0 | 0 | | | | | | | | | | | Δ | 0 |
| * | 0 | | 0 | | 0 | 0 | | 0 | | Parameter | | | | | | | | | | Δ | |
| | | | | | | | | | | (No.0042) | | | | | | | | | | | |
| , | 0 | | 0 | | 0 | 0 | \bigcirc | | | , | | | 0 | 0 | 0 | 0 | | 0 | 0 | | |
| ; | 0 | | \bigcirc | 0 | 0 | 0 | | 0 | 0 | | | | | | | | | | | Δ | Δ |
| < | | | 0 | 0 | 0 | 0 | \bigcirc | | | | | | | | | | | | | Δ | Δ |
| = | 0 | | 0 | 0 | 0 | 0 | 0 | | 0 | Parameter | | | | | | | | | | Δ | |
| | | | | | | | | | | (No.0043) | | | | | | | | | | | |
| > | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | Δ | Δ |
| ? | | | 0 | 0 | 0 | 0 | \bigcirc | 0 | 0 | | | | | | | | | | | Δ | 0 |
| @ | 0 | 0 | | | | 0 | | | | | | | | | | | | | | Δ | 0 |
| " | | | 0 | | | | | 0 | | | | | | | | | | | | Δ | Δ |
| [| 0 | 0 | | 0 | 0 | 0 | | 0 | 0 | Parameter | | | | | | | | | | Δ | |
| - | | | | | | | | | | (No.0053) | | | | | | | | | | | |
| 1 | 0 | 0 | | 0 | 0 | 0 | 0 | | 0 | Parameter | | | | | | | | | | Δ | |
| | | | | | | | | | | (No.0054) | | | | | | | | | | | |

NOTE

- 1 The symbols used in the remark column have the following meanings.
 - (Space) : The character will be registered in memory and has a specific meaning. If it is used incorrectly in a statement other than a comment, an alarm occurs.
 - \times : The character will not be registered in memory and will be ignored.
 - Δ : The character will be registered in memory, but will be ignored during program execution.
 - : The character will be registered in memory. If it is used in a statement other than a comment, an alarm occurs.
 - □ : If it is used in a statement other than a comment, the character will not be registered in memory. If it is used in a comment, it will be registered in memory.
 - 2 Codes not in this table are ignored if their parity is correct.
- 3 Codes with incorrect parity cause the TH alarm. But they are ignored without generating the TH alarm when they are in the comment section.
- 4 A character with all eight holes punched is ignored and does not generate TH alarm in EIA code.

B

LIST OF FUNCTIONS AND TAPE FORMAT

Some functions cannot be added as options depending on the model. In the tables below, \mathbb{P}_{-} presents a combination of arbitrary axis addresses using X,Y,Z,A,B and C (such as X_Y_Z_A_).

- x = 1st basic axis (X usually)
- y = 2nd basic axis (Y usually)
- z = 3rd basic axis (Z usually)

| Functions | Illustration | Tape format |
|--------------------------------------|---|--|
| Positioning (G00) | Start point | G00 _; ₽ |
| Linear interpolation (G01) | Start point | G01 _ F_; IP |
| Circular interpolation (G02, G03) | Start point R J G02 (x, y) (x, y) R J (x, y) R J Start point | $ \begin{array}{c} G17 \left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\} X_{-} Y_{-} \left\{ \begin{matrix} R_{-} \\ I_{-} J_{-} \end{matrix} \right\} F_{-}; \\ G18 \left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\} X_{-} Z_{-} \left\{ \begin{matrix} R_{-} \\ I_{-} K_{-} \end{matrix} \right\} F_{-}; \\ G19 \left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\} Y_{-} Z_{-} \left\{ \begin{matrix} R_{-} \\ J_{-} K_{-} \end{matrix} \right\} F_{-}; \\ \end{array} $ |
| Dwell (G04) | | $G04 \left\{ egin{array}{c} X_{-} \ P_{-} \end{array} ight\};$ |
| Exact stop (G09) | Velocity Time Inposition check | $G09 \left\{ \begin{matrix} G01_\\ G02_\\ G03_ \end{matrix} \right\} _;$ |

— 430 —

| Functions | Illustration | Tape format |
|---|--|---|
| Change of offset value by program (G10) | | Tool offset value G10 P_ R_ ; Workpiece origin offset value G10 L2 P_ IP_ ; G10 L20 P_ IP_ ; |
| Cutter compensation (G40 – G42) | G41 G40 G42 | $ \left\{ \begin{array}{c} G17\\G18\\G19 \end{array} \right\} \left\{ \begin{array}{c} G40\\G41\\G42 \end{array} \right\} I\!\!P_{-} \left\{ \begin{array}{c} H_{-} \ ;\\ D_{-} \ ; \end{array} \right\} \\ H, D: Cutter \ compensation \ number \\ G40: Cancel \end{array} \right. $ |
| Tool length offset (G43, G44, G49) | Z Offset | $ \left\{ \begin{array}{c} G43\\ G44 \end{array} \right\} \alpha_{-} H_{-}; \\ \left\{ \begin{array}{c} G43\\ G44 \end{array} \right\} H_{-}; \\ H: \text{ Tool offset number} \\ G49: \text{ Cancel} \end{array} \right. $ |
| Local coordinate system setting (G52) | x Local coordinate system IP y Workpiece coordinate system | G52 ₽ _ ; |
| Machine coordinate system selection (G53) | | G53 IP_ ; |
| Workpiece coordinate system selection (G54 – G59) | Workpiece zero point offset Workpiece coordinate system Machine coordinate system | $ \left\{ \begin{array}{c} G54 \\ \vdots \\ G59 \end{array} \right\} \mathbf{IP}_{-}; $ |
| Single direction posiitoning (G50) | ₽ • ~ | G60 ₽ _; |
| Inch/millimeter conversion (G20, G21) | | G20 ; Inch input G21 ; Millimeter input |

— 431 —

B. LIST OF FUNCTIONS AND TAPE FORMAT

| Functions | Illustration | Tape format |
|---|--------------------------------|--|
| Reference position return check (G27) | Start point | G27 ₽_; |
| Reference position return (G28) 2nd reference position re- turn (G30) | Reference position (G28) | G28 IP_; G30 P2 IP_; |
| Return from reference position to start point (G29) | Reference position | G29 IP _; |
| Skip function (G31) | Start point Skip signal | G31 IP_ F_; |
| Cutting mode, Exact stop mode, Tapping mode, Automatic corner override mode | v G64 t v G60 t | G64_ ; Cutting mode G60_ ; Exact stop mode G62_ ; Automatic corner override mode G63_ ; Tapping mode |
| Custom macro (G65, G66, G67) | G65 P_; | Varialde arithmetic command and divergence commnad G65H_P_Q_R_; Modal call G66 P_ G67; cancel |

— 432 —

| Functions | Illustration | Tape format |
|---|--|--|
| Canned cycles (G73, G74, G80 – G89) | Refer to II.13. FUNCTIONS TO SIMPLIFY PROGRAMMING | $ \begin{array}{c} G80 \ ; Cancel \\ G73 \\ G74 \\ G76 \\ G81 \\ \vdots \\ G89 \end{array} \right\} X_Y_Z_P_Q_R_F_K_; $ |
| Absolute/incremental programming (G90/G91) | | G90_; Absolute command G91_; Incremental command G90_G91_; Combined use |
| Change of workpiece coordinate system (G92) | | G92 IP_; |
| Initial point return / R point return (G98, G99) | G98 I point G99 R point Z point | G98 ; G99 ; |



Linear axis

• In case of millimeter input, feed screw is millimeter

| | Incremer | nt system |
|----------------------------------|---------------------------------|-------------------------------------|
| | IS–B | IS–C |
| Least input incre- ment | 0.001 mm | 0.0001 mm |
| Least command in- crement | 0.001 mm | 0.0001 mm |
| Max. program- mable dimension | ±99999.999 mm | ±9999.9999 mm |
| Max. rapid traverse | 100000 mm/min | 24000 mm/min |
| Feedrate range NOTE | 1 to 100000 mm/min | 1 to 12000 mm/min |
| Incremental feed | 0.001, 0.01, 0.1, 1 mm/ step | 0.0001, 0.001, 0.01, 0.1 mm/step |
| Tool compensation | 0 to ±999.999 mm | 0 to ±999.9999 mm |
| Dwell time | 0 to 99999.999 sec | 0 to 9999.9999 sec |

• In case of inch input, feed screw is millimeter

| | Incremer | it system |
|----------------------------------|---------------------------------------|--|
| | IS–B | IS-C |
| Least input incre- ment | 0.0001 inch | 0.00001 inch |
| Least command in- crement | 0.001 mm | 0.0001 mm |
| Max. program- mable dimension | ±9999.9999 inch | ±393.70078 inch |
| Max. rapid traverse NOTE | 100000 mm/min | 24000 mm/min |
| Feedrate range NOTE | 0.01 to 4000 inch/min | 0.01 to 12000 inch/min |
| Incremental feed | 0.0001, 0.001, 0.01, 0.1 inch/step | 0.00001, 0.0001, 0.001, 0.01 inch/step |
| Tool compensation | 0 to ±99.9999 inch | 0 to ±99.99999 inch |
| Dwell time | 0 to 99999.999 sec | 0 to 9999.9999 sec |

• In case of inch input, feed screw is inch

| | Incremen | it system |
|----------------------------------|---------------------------------------|--|
| | IS–B | IS-C |
| Least input incre- ment | 0.0001 inch | 0.00001 inch |
| Least command in- crement | 0.0001 inch | 0.00001 inch |
| Max. program- mable dimension | ±9999.9999 inch | ±999.99999 inch |
| Max. rapid traverse | 4000 inch/min | 960 inch/min |
| Feedrate range NOTE | 0.01 to 4000 inch/min | 0.01 to 480 inch/min |
| Incremental feed | 0.0001, 0.001, 0.01, 0.1 inch/step | 0.00001, 0.0001, 0.001, 0.01 inch/step |
| Tool compensation | 0 to ±99.9999 inch | 0 to ±99.99999 inch |
| Dwell time | 0 to 99999.999 sec | 0 to 9999.9999 sec |

— 435 —

In case of millimeter input, feed screw is inch

| | Incremer | nt system |
|----------------------------------|---------------------------------|-------------------------------------|
| | IS–B | IS-C |
| Least input incre- ment | 0.001 mm | 0.0001 mm |
| Least command in- crement | 0.0001 inch | 0.00001 inch |
| Max. program- mable dimension | ±99999.999 mm | ±9999.9999 mm |
| Max. rapid traverse | 4000 inch/min | 960 inch/min |
| Feedrate range NOTE | 1 to 100000 mm/min | 1 to 12000 mm/min |
| Incremental feed | 0.001, 0.01, 0.1, 1 mm/ step | 0.0001, 0.001, 0.01, 0.1 mm/step |
| Tool compensation | 0 to ±999.999 mm | 0 to ±999.9999 mm |
| Dwell time | 0 to 99999.999 sec | 0 to 9999.9999 sec |

Rotation axis

| | Increm | ent system |
|---|----------------------------------|---------------------------------------|
| | IS–B | IS-C |
| Least input incre- ment | 0.001 deg | 0.0001 deg |
| Least command in- crement | 0.001 deg | 0.0001 deg |
| Max. program- mable dimension | ±99999.999 deg | ±9999.9999 deg |
| Max. rapid traverse NOTE | 240000 deg/min | 100000 deg/min |
| Feedrate range (metric input) NOTE | 1 to 100000 deg/min | 1 to 24000 deg/min |
| Feedrate range (inch input) NOTE | 1 to 6000 deg/min | 1 to 600 deg/min |
| Incremental feed | 0.001, 0.01, 0.1, 1 deg/ step | 0.0001, 0.001, 0.01, 0.1 deg/ step |

NOTE

The feedrate range shown above are limitations depending on CNC interpolation capacity. As a whole system, limitations depending on servo system must also be considered.

— 436 —

NOMOGRAPHS

D.1 TOOL PATH AT CORNER

When servo system delay (by exponential acceleration/deceleration at cutting or caused by the positioning system when a servo motor is used) is accompanied by cornering, a slight deviation is produced between the tool path (tool center path) and the programmed path as shown in Fig. 4.3 (a).

Time constant T_1 of the exponential acceleration/deceleration is fixed to 0.



Fig. D.1 (a) Slight deviation between the tool path and the programmed path

This tool path is determined by the following parameters:

- · Feedrate (V_1, V_2)
- · Corner angle (θ)
- Exponential acceleration / deceleration time constant (T₁) at cutting $(T_1 = 0)$
- · Presence or absence of buffer register.

The above parameters are used to theoretically analyze the tool path. When actually programming, the above items must be considered and programming must be performed carefully so that the shape of the workpiece is within the desired precision.

In other words, when the shape of the workpiece is not within the theoretical precision, the commands of the next block must not be read until the specified feedrate becomes zero. The dwell function is then used to stop the machine for the appropriate period.

• Analysis

The tool path shown in Fig. 4.3 (b) is analyzed based on the following conditions:

Feedrate is constant at both blocks before and after cornering. The controller has a buffer register. (The error differs with the reading speed of the tape reader, number of characters of the next block, etc.)



Fig. D.1(b) Example of tool path

• Description of conditions and symbols

| $V_{X1} = V \cos \phi_1$ |
|--|
| $V_{Y1} = V \sin \phi_1$ |
| $V_{X2} = V \cos \phi_2$ |
| $V_{Y2} = V \sin \phi_2$ |
| V : Feedrate at both blocks before and after cornering V_{X1} : X-axis component of feedrate of preceding block V_{Y1} : Y-axis component of feedrate of preceding block V_{X2} : X-axis component of feedrate of following block V_{Y2} : Y-axis component of feedrate of following block 0 /ul> |

Initial value calculation



The initial value when cornering begins, that is, the X and Y coordinates at the end of command distribution by the controller, is determined by the feedrate and the positioning system time constant of the servo motor.

$$X_{0} = V_{X1}(T_{1} + T_{2})$$
$$Y_{0} = V_{Y1}(T_{1} + T_{2})$$

 T_1 :Exponential acceleration / deceleration time constant. (T=0) T_2 :Time constant of positioning system (Inverse of position loop gain)

 Analysis of corner tool path The equations below represent the feedrate for the corner section in X-axis direction and Y-axis direction.

$$V_X(t) = \frac{V_{x1} V_{x2}}{T_1 - T_2} \{ T_1 \exp(-\frac{t}{T_1}) - T_2 \exp(-\frac{t}{T_2}) \} + V_{x2}$$
$$V_Y(t) = \frac{V_{Y1} - V_{Y2}}{T_1 - T_2} \{ T_1 \exp(-\frac{t}{T_1}) - T_2 \exp(-\frac{t}{T_2}) \} + V_{Y2}$$

Therefore, the coordinates of the tool path at time *t* are calculated from the following equations:

$$\begin{aligned} X(t) &= \int_{0}^{t} V_{X}(t) dt - X_{0} \\ &= \frac{V_{X2} - V_{X1}}{T_{1} - T_{2}} \{T_{1}^{2} \exp(-\frac{t}{T_{1}}) - T_{2}^{2} \exp(-\frac{t}{T_{2}})\} - V_{X2}(T_{1} + T_{2} - t) \\ Y(t) &= \int_{0}^{t} V_{Y}(t) dt - Y_{0} \\ &= \frac{V_{Y2} - V_{Y1}}{T_{1} - T_{2}} \{T_{1}^{2} \exp(-\frac{t}{T_{1}}) - T_{2}^{2} \exp(-\frac{t}{T_{2}})\} - V_{Y2}(T_{1} + T_{2} - t) \end{aligned}$$

— 440 —

D.2 RADIUS DIRECTION ERROR AT CIRCLE CUTTING

When a servo motor is used, the positioning system causes an error between input commands and output results. Since the tool advances along the specified segment, an error is not produced in linear interpolation. In circular interpolation, however, radial errors may be produced, sepecially for circular cutting at high speeds.

This error can be obtained as follows:



Since the machining radius r (mm) and allowable error Δr (mm) of the workpiece is given in actual machining, the allowable limit feedrate v (mm/sec) is determined by equation (1).

Since the acceleration/deceleration time constant at cutting which is set by this equipment varies with the machine tool, refer to the manual issued by the machine tool builder.

Е

STATUS WHEN TURNING POWER ON, WHEN CLEAR AND WHEN RESET

Parameter No. 046 bit6 is used to select whether resetting the CNC places it in the cleared state or in the reset state (0: reset state/1: cleared state). The symbols in the tables below mean the following :

- \bigcirc : The status is not changed or the movement is continued.
- \times : The status is cancelled or the movement is interrupted.

| | Item | When turning power | Cleared | Reset |
|------------|---------------------------------------|--|---|--|
| | | on | | |
| Setting | Offset value | 0 | 0 | 0 |
| uala | Data set by the MDI setting operation | 0 | 0 | 0 |
| | Parameter | 0 | 0 | 0 |
| Various | Programs in memory | 0 | 0 | 0 |
| uala | Contents in the buffer storage | × | × | ○ : MDI mode× : Other mode |
| | Display of sequence number | 0 | ○ (Note 1) | ○ (Note 1) |
| | One shot G code | × | × | × |
| | Modal G code | Initial G codes. (The G20 and G21 codes return to the same state they were in when the power was last turned off.) | Initial G codes. (G20/G21 are not changed.) | 0 |
| | F | Zero | Zero | 0 |
| S, T, M | | × | 0 | 0 |
| | K (Number of re- peats) | × | × | × |
| Work coord | dinate value | Zero | 0 | 0 |
| Action in | Movement | × | × | × |
| operation | Dwell | × | × | × |
| | Issuance of M, S and T codes | × | × | × |
| | Tool length com- pensation | × | Depending on parameter No.001#3 | ○ : MDI mode Other modes depend on parameter No.001#3. |
| | Cutter compensation | × | × | ○ : MDI mode× : Other modes |
| | Storing called subpro- gram number | × | × (Note 2) | ○ : MDI mode × : Other modes (Note 2) |

| | ltem | When turning power on | Cleared | Reset |
|-------------------|--|---|---|---|
| Output signals | CNC alarm signal AL | Extinguish if there is no cause for the alarm | Extinguish if there is no cause for the alarm | Extinguish if there is no cause for the alarm |
| | Reference position return completion LED | × | ○ (× : Emergency stop) | ○ (× : Emergency stop) |
| | S, T and B codes | × | 0 | 0 |
| | M code | × | × | × |
| | M, S and T strobe signals | × | × | × |
| | Spindle revolution sig- nal (S analog signal) | × | 0 | 0 |
| | CNC ready signal MA | ON | 0 | 0 |
| | Servo ready signal SA | ON (When other than servo alarm) | ON (When other than servo alarm) | ON (When other than servo alarm) |
| | Cycle start LED (STL) | × | × | × |
| | Feed hold LED (SPL) | × | × | × |

APPENDIX

NOTE

- 1 When heading is performed, the main program number is displayed as the sequence number.
- 2 When a reset is performed during execution of a subprogram, control returns the main program. Execution cannot be started from the middle of the subprogram.



CHARACTER-TO-CODES CORRESPONDENCE TABLE

APPENDIX

| Char- acter | Code | Comment | Char- acter | Code | Comment |
|----------------|------|---------|----------------|------|----------------------|
| А | 065 | | 6 | 054 | |
| В | 066 | | 7 | 055 | |
| С | 067 | | 8 | 056 | |
| D | 068 | | 9 | 057 | |
| Е | 069 | | | 032 | Space |
| F | 070 | | ! | 033 | Exclamation mark |
| G | 071 | | " | 034 | Quotation mark |
| Н | 072 | | # | 035 | Hash sign |
| I | 073 | | \$ | 036 | Dollar sign |
| J | 074 | | % | 037 | Percent |
| К | 075 | | & | 038 | Ampersand |
| L | 076 | | , | 039 | Apostrophe |
| М | 077 | | (| 040 | Left parenthesis |
| N | 078 | |) | 041 | Right parenthesis |
| 0 | 079 | | * | 042 | Asterisk |
| Р | 080 | | + | 043 | Plus sign |
| Q | 081 | | , | 044 | Comma |
| R | 082 | | - | 045 | Minus sign |
| S | 083 | | | 046 | Period |
| Т | 084 | | / | 047 | Slash |
| U | 085 | | : | 058 | Colon |
| V | 086 | | ; | 059 | Semicolon |
| W | 087 | | < | 060 | Left angle bracket |
| Х | 088 | | = | 061 | Equal sign |
| Y | 089 | | > | 062 | Right angle bracket |
| Z | 090 | | ? | 063 | Question mark |
| 0 | 048 | | @ | 064 | HAtI mark |
| 1 | 049 | | [| 091 | Left square bracket |
| 2 | 050 | | ^ | 092 | |
| 3 | 051 | | 0 | 093 | Yen sign |
| 4 | 052 | |] | 094 | Right square bracket |
| 5 | 053 | | _ | 095 | Underscore |

1) Program errors (P/S alarm)

| Number | Meaning | Contents and remedy |
|--------|-----------------------------------|--|
| 000 | PLEASE TURN OFF POWER | A parameter which requires the power off was input, turn off power. |
| 001 | TH PARITY ALARM | TH alarm (A character with incorrect parity was input). Correct the tape. |
| 002 | TV PARITY ALARM | TV alarm (The number of characters in a block is odd). This alarm will be generated only when the TV check is effective. |
| 003 | TOO MANY DIGITS | Data exceeding the maximum allowable number of digits was input. (Refer to the item of max. programmable dimensions.) |
| 004 | ADDRESS NOT FOUND | A numeral or the sign " $-$ " was input without an address at the beginning of a block. Modify the program . |
| 005 | NO DATA AFTER ADDRESS | The address was not followed by the appropriate data but was followed by another address or EOB code. Modify the program. |
| 006 | ILLEGAL USE OF NEGATIVE SIGN | Sign " ." input error (Sign " – " was input after an address with which it cannot be used. Or two or more " – " signs were input.) Modify the program. |
| 007 | ILLEGAL USE OF DECIMAL POINT | Decimal point "–" input error (A decimal point was input after an address with which it can not be used. Or two decimal points were input.) Modify the program. |
| 008 | ILLEGAL USE OF PROGRAM END | An attempt was made to execute EOR (%) because there was not M02, M30, or M99 at the end of the program. Correct the program. |
| 009 | ILLEGAL ADDRESS INPUT | Unusable character was input in significant area. Modify the program. |
| 010 | IMPROPER G-CODE | An unusable G code or G code corresponding to the function not pro- vided is specified. Modify the program. |
| 011 | NO FEEDRATE COMMANDED | Feedrate was not commanded to a cutting feed or the feedrate was in- adequate. Modify the program. |
| 014 | CAN NOT COMMAND G95 | A synchronous feed is specified without the option for threading / syn- chronous feed. |
| 015 | TOO MANY AXES COMMANDED | The number of the commanded axes exceeded that of simultaneously controlled axes. |
| 020 | OVER TOLERANCE OF RADIUS | In circular interpolation (G02 or G03), difference of the distance between the start point and the center of an arc and that between the end point and the center of the arc exceeded the value specified in parameter No. 876. |
| 021 | ILLEGAL PLANE AXIS COMMAN- DED | An axis not included in the selected plane (by using G17, G18, G19) was commanded in circular interpolation. Modify the program. |
| 025 | CANNOT COMMAND F0 IN G02/G03 | F0 (fast feed) was instructed by F1 –digit column feed in circular interpolation. Modify the program. |
| 027 | NO AXES COMMANDED IN G43/G44 | No axis is specified in G43 and G44 blocks for the tool length offset type C. Offset is not canceled but another axis is offset for the tool length offset type C. Modify the program. |

— 445 —

| Number | Meaning | Contents and remedy |
|--------|--------------------------------------|--|
| 028 | ILLEGAL PLANE SELECT | In the plane selection command, two or more axes in the same direction are commanded. Modify the program. |
| 029 | ILLEGAL OFFSET VALUE | The offset values specified by H code is too large. Modify the program. |
| 030 | ILLEGAL OFFSET NUMBER | The offset number specified by D/H code for tool length offset or cutter compensation is too large. Modify the program. |
| 031 | ILLEGAL P COMMAND IN G10 | In setting an offset amount by G10, the offset number following address P was excessive or it was not specified. Modify the program. |
| 032 | ILLEGAL OFFSET VALUE IN G10 | In setting an offset amount by G10 or in writing an offset amount by system variables, the offset amount was excessive. |
| 033 | NO SOLUTION AT CRC | A point of intersection cannot be determined for cutter compensation C. Modify the program. |
| 034 | NO CIRC ALLOWED IN ST-UP /EXT BLK | The start up or cancel was going to be performed in the G02 or G03 mode in cutter compensation. Modify the program. |
| 035 | CAN NOT COMMANDED G39 | G39 is commanded in cutter compensation B cancel mode or on the plane other than offset plane. Modify the program. |
| 036 | CAN NOT COMMANDED G31 | Skip cutting (G31) was specified in cutter compensation mode. Modify the program. |
| 037 | CAN NOT CHANGE PLANE IN CRC | G40 is commanded on the plane other than offset plane in cutter com- pensation B. The plane selected by using G17, G18 or G19 is changed in cutter compensation C mode. Modify the program. |
| 038 | INTERFERENCE IN CIRCULAR BLOCK | Overcutting will occur in cutter compensation C because the arc start point or end point coincides with the arc center. Modify the program. |
| 041 | INTERFERENCE IN CRC | Overcutting will occur in cutter compensation C. Two or more blocks are consecutively specified in which functions such as the auxiliary function and dwell functions are performed without movement in the cut- ter compensation mode. Modify the program. |
| 042 | G45/G48 NOT ALLOWED IN CRC | Tool offset (G45 to G48) is commanded in cutter compensation. Modify the program. |
| 043 | ILLEGAL T-CODE COMMAND | In the DRILL–MATE, a T code was not specified together with the M06 code in a block. Alternatively, the Tcode was out of range. |
| 044 | G27–G30 NOT ALLOWED IN FIXED CYC | One of G27 to G30 is commanded in canned cycle mode. Modify the program. |
| 046 | ILLEGAL REFERENCE RETURN COMMAND | Other than P2, P3 and P4 are commanded for 2nd, 3rd and 4th reference position return command. |
| 050 | CHF/CNR NOT ALLOWED IN THRD BLK | Chamfering or corner R is commanded in the thread cutting block. Modify the program. |
| 051 | MISSING MOVE AFTER CHF/CNR | Improper movement or the move distance was specified in the block next to the chamfering or corner R block. Modify the program. |
| 052 | CODE IS NOT G01 AFTER CHF/ CNR | The block next to the chamfering or corner R block is not G01. Modify the program. |
| 053 | TOO MANY ADDRESS COM- MANDS | For systems without the arbitary angle chamfering or corner R cutting, a comma was specified. For systems with this feature, a comma was followed by something other than R or C Correct the program. |

| Number | Meaning | Contents and remedy |
|--------|--------------------------------------|---|
| 055 | MISSING MOVE VALUE IN CHF/ CNR | In the arbitrary angle chamfering or corner R block, the move distance is less than chamfer or corner R amount. |
| 058 | END POINT NOT FOUND | In a arbitrary angle chamfering or corner R cutting block, a specified axis is not in the selected plane. Correct the program. |
| 059 | PROGRAM NUMBER NOT FOUND | In an external program number search, a specified program number was not found. Otherwise, a program specified for searching is being edited in background processing. Check the program number and ex- ternal signal. Or discontinue the background eiting. |
| 060 | SEQUENCE NUMBER NOT FOUND | Commanded sequence number was not found in the sequence number search. Check the sequence number. |
| 070 | NO PROGRAM SPACE IN MEMORY | The memory area is insufficient. Delete any unnecessary programs, then retry. |
| 071 | DATA NOT FOUND | The address to be searched was not found. Or the program with speci- fied program number was not found in program number search. Check the data. |
| 072 | TOO MANY PROGRAMS | The number of programs to be stored exceeded 63 (basic), 125 (option) or, 200 (option). Delete unnecessary programs and execute program registeration again. |
| 073 | PROGRAM NUMBER ALREADY IN USE | The commanded program number has already been used. Change the program number or delete unnecessary programs and execute program registeration again. |
| 074 | ILLEGAL PROGRAM NUMBER | The program number is other than 1 to 9999. Modify the program number. |
| 076 | ADDRESS P NOT DEFINED | Address P (program number) was not commanded in the block which includes an M98, G65, or G66 command. Modify the program. |
| 077 | SUB PROGRAM NESTING ERROR | A level larger than the maximum allowable nesting level of subprogram calls was specified. Modify the program. |
| 078 | NUMBER NOT FOUND | A program number or a sequence number which was specified by ad- dress P in the block which includes an M98, M99, M65 or G66 was not found. The sequence number specified by a GOTO statement was not found. An attempt was made to call a program being edited in the back- ground processing mode. Correct the program. |
| 079 | PROGRAM VERIFY ERROR | In memory or program collation, a program in memory does not agree with that read from an external I/O device. Check both the programs in memory and those from the external device. |
| 080 | G37 ARRIVAL SIGNAL NOT AS- SERTED | In the automatic tool length measurement function (G37), the measurement position reach signal (XAE, YAE, or ZAE) is not turned on within an area specified in parameter (value ε). This is due to a setting or operator error. |
| 081 | OFFSET NUMBER NOT FOUND IN G37 | Tool length automatic measurement (G37) was specified without a H code. (Automatic tool length measurement function) Modify the program. |
| 082 | H–CODE NOT ALLOWED IN G37 | H code and automatic tool compensation (G37) were specified in the same block. (Automatic tool length measurement function) Modify the program. |
| 083 | ILLEGAL AXIS COMMAND IN G37 | In automatic tool length measurement, an invalid axis was specified or the command is incremental. Modify the program. |
| 085 | COMMUNICATION ERROR | When entering data in the memory by using Reader / Puncher interface, an overrun, parity or framing error was generated. The number of bits of input data or setting of baud rate or specification No. of I/O unit is in- correct. |

| Number | Meaning | Contents and remedy |
|--------|--|--|
| 086 | DR SIGNAL OFF | When entering data in the memory by using Reader / Puncher interface, the ready signal (DR) of reader / puncher was turned off. Power supply of I/O unit is off or cable is not connected or a P.C.B. is defective. |
| 087 | BUFFER OVERFLOW | When entering data in the memory by using Reader / Puncher interface, though the read terminate command is specified, input is not interrupted after 10 characters read. I/O unit or P.C.B. is defective. |
| 090 | REFERENCE RETURN INCOM- PLETE | The reference position return cannot be performed normally because the reference position return start point is too close to the reference posi- tion or the speed is too slow. Separate the start point far enough from the reference position, or specify a sufficiently fast speed for reference position return. |
| 091 | MANUAL RETURN IMPOSSIBLE DURING PAUSE | A manual return to the reference position was impossible because of the program being at pause. Press the reset button to cause a manual return. |
| 092 | AXES NOT ON THE REFERENCE POINT | The commanded axis by G28 (automatic reference position return) or G27 (reference position return check) did not return to the reference position. |
| 094 | P TYPE NOT ALLOWED (COORD CHG) | P type cannot be specified when the program is restarted. (After the automatic operation was interrupted, the coordinate system setting operation was performed.) Perform the correct operation according to th operator's manual. |
| 095 | P TYPE NOT ALLOWED (EXT OFS CHG) | P type cannot be specified when the program is restarted. (After the automatic operation was interrupted, the external workpiece offset amount changed.) |
| 096 | P TYPE NOT ALLOWED (WRK OFS CHG) | P type cannot be specified when the program is restarted. (After the automatic operation was interrupted, the workpiece offset amount changed.) |
| 097 | P TYPE NOT ALLOWED (AUTO EXEC) | P type cannot be directed when the program is restarted. (After power ON, after emergency stop or P/S 94 to 97 reset, no automatic operation is performed.) Perform automatic operation. |
| 098 | G28 FOUND IN SEQUENCE RE- TURN | A command of the program restart was specified without the reference position return operation after power ON or emergency stop, and G28 was found during search. |
| 099 | MDI EXEC NOT ALLOWED AFT. SEARCH | After completion of search in program restart, a move command is given with MDI. |
| 100 | PARAMETER WRITE ENABLE | On the PARAMETER(SETTING) screen, PWE (parameter writing enabled) is set to 1. Set it to 0, then reset the system. |
| 101 | PLEASE CLEAR MEMORY | The power turned off while rewriting the memory by program edit opera- tion. When this alarm occurs, set the PWE parameter to 1, then switch on the power while holding down the <delet> key. All programs will be deleted.</delet> |
| 109 | P/S ALARM | A value other than 0 or 1 was specified after P in the G08 code, or no value was specified. |
| 110 | DATA OVERFLOW | The absolute value of fixed decimal point display data exceeds the al- lowable range. Modify the program. |
| 111 | CALCULATED DATA OVERFLOW | The result of calculation turns out to be invalid, an alarm No.111 is issued. -10^{47} to -10^{-29} , 0, 10^{-29} to 10^{47} |
| 112 | DIVIDED BY ZERO | Division by zero was specified. (including tan 90°) |
| 113 | IMPROPER COMMAND | A function which cannot be used in custom macro is commanded. Modify the program. |
| 114 | FORMAT ERROR IN MACRO | Custom macro A specified an undefined H code in a G65 block. There is an error in other formats than <formula>. Modify the program.</formula> |

| Number | Meaning | Contents and remedy |
|--------|--------------------------------------|---|
| 115 | ILLEGAL VARIABLE NUMBER | A value not defined as a variable number is designated in the custom macro or in high-speed cycle machining. The header contents are improper. This alarm is given in the following cases: High speed cycle machining The header corresponding to the specified machining cycle number called is not found. The cycle connection data value is out of the allowable range (0 - 999). The number of data in the header is out of the allowable range (0 - 32767). The start data variable number of executable format data is out of the allowable range (#20000 - #85535). The last storing data variable number of executable format data is out of the allowable range (#85535). The storing start data variable number of executable format data is overlapped with the variable number used in the header. Modify the program. |
| 116 | WRITE PROTECTED VARIABLE | The left side of substitution statement is a variable whose substitution is inhibited. Modify the program. |
| 118 | PARENTHESIS NESTING ERROR | The nesting of bracket exceeds the upper limit (quintuple). Modify the program. |
| 119 | ILLEGAL ARGUMENT | The SQRT argument is negative. Or BCD argument is negative, and other values than 0 to 9 are present on each line of BIN argument. Modify the program. |
| 122 | DUPLICATE MACRO MODAL-CALL | The macro modal call is specified in double. Modify the program. |
| 123 | CAN NOT USE MACRO COMMAND IN DNC | Macro control command is used during DNC operation. Modify the program. |
| 124 | MISSING END STATEMENT | DO – END does not correspond to 1 : 1. Modify the program. |
| 125 | FORMAT ERROR IN MACRO | Custom macro A specified an undefined H code in a G65 block. <formula> format is erroneous. Modify the program.</formula> |
| 126 | ILLEGAL LOOP NUMBER | In DOn, $1 \le n \le 3$ is not established. Modify the program. |
| 127 | NC, MACRO STATEMENT IN SAME BLOCK | NC and custom macro commands coexist. Modify the program. |
| 128 | ILLEGAL MACRO SEQUENCE NUM- BER | The sequence number specified in the branch command was not 0 to 9999. Or, it cannot be searched. Modify the program. |
| 129 | ILLEGAL ARGUMENT ADDRESS | An address which is not allowed in <argument designation=""> is used. Modify the program.</argument> |
| 130 | ILLEGAL AXIS OPERATION | An axis control command was given by PMC to an axis controlled by CNC. Or an axis control command was given by CNC to an axis controlled by PMC. Modify the program. |
| 131 | TOO MANY EXTERNAL ALARM MESSAGES | Five or more alarms have generated in external alarm message. Consult the PMC ladder diagram to find the cause. |
| 132 | ALARM NUMBER NOT FOUND | No alarm No. concerned exists in external alarm message clear. Check the PMC ladder diagram. |
| 133 | ILLEGAL DATA IN EXT. ALARM MSG | Small section data is erroneous in external alarm message or external operator message. Check the PMC ladder diagram. |
| 135 | ILLEGAL ANGLE COMMAND | The index table indexing positioning angle was instructed in other than an integral multiple of the value of the minimum angle. Modify the program. |
| 136 | ILLEGAL AXIS COMMAND | In index table indexing. Another control axis was instructed together with the B axis. Modify the program. |

| Number | Meaning | Contents and remedy |
|--------|--------------------------------------|--|
| 139 | CAN NOT CHANGE PMC CON- TROL AXIS | An axis is selected in commanding by PMC axis control. Modify the program. |
| 141 | CAN NOT COMMAND G51 IN CRC | G51 (Scaling ON) is commanded in the tool offset mode. Modify the program. |
| 142 | ILLEGAL SCALE RATE | Scaling magnification is commanded in other than 1 – 999999. Correct the scaling magnification setting . |
| 143 | SCALED MOTION DATA OVER- FLOW | The scaling results, move distance, coordinate value and circular radius exceed the maximum command value. Correct the program or scaling mangification. |
| 144 | ILLEGAL PLANE SELECTED | The coordinate rotation plane and arc or cutter compensation C plane must be the same. Modify the program. |
| 148 | ILLEGAL SETTING DATA | Automatic corner override deceleration rate is out of the settable range of judgement angle. Modify the parameters (No.1710 to No.1714) |
| 150 | ILLEGAL TOOL GROUP NUMBER | Tool Group No. exceeds the maximum allowable value in the tool life management. Modify the program. |
| 151 | TOOL GROUP NUMBER NOT FOUND | The tool group of the tool life management commanded in the machin- ing program is not set. Modify the value of program or parameter. |
| 152 | NO SPACE FOR TOOL ENTRY | The number of tools within one group in the tool life management exceeds the maximum value registerable. Modify the number of tools. |
| 153 | T-CODE NOT FOUND | In tool life data registration, a T code was not specified where one should be. Correct the program. |
| 154 | NOT USING TOOL IN LIFE GROUP | When the group is not commanded in the tool life management, H99 or D99 was commanded. Correct the program. |
| 155 | ILLEGAL T-CODE IN M06 | In the machining program, M06 and T code in the same block do not cor- respond to the group in use. Correct the program. |
| 156 | P/L COMMAND NOT FOUND | P and L commands are missing at the head of program in which the tool group of the tool life management is set. Correct the program. |
| 157 | TOO MANY TOOL GROUPS | The number of tool groups in the tool life management to be set exceeds the maximum allowable value. Modify the program. |
| 158 | ILLEGAL TOOL LIFE DATA | The tool life to be set is too excessive. Modify the setting value. |
| 159 | TOOL DATA SETTING INCOM- PLETE | During executing a life data setting program, power was turned off. Set again. |
| 175 | ILLEGAL G107 COMMAND | Conditions when performing circular interpolation start or cancel not correct. To change the mode to the cylindrical interpolation mode, specify the command in a format of "G07.1 rotation–axis name radius of cylinder." |
| 176 | IMPROPER G-CODE IN G107 | Any of the following G codes which cannot be specified in the cylindrical interpolation mode was specified. 1) G codes for positioning: G28,, G73, G74, G76, G81 – G89, including the codes specifying the rapid traverse cycle 2) G codes for setting a coordinate system: G52,G92, 3) G code for selecting coordinate system: G53 G54–G59 Modify the program. |
| 177 | CHECK SUM ERROR (G05 MODE) | Check sum error is occurred in the high-speed remote buffer. |
| 178 | G05 COMMANDED IN G41/G42 MODE | G05 was commanded in the G41/G42 mode. Correct the program. |
| 179 | PARAMETER SETTING ERROR | The number of controlled axes set by the parameter 7510 exceeds the maximum number. Modify the parameter setting value. |
| 180 | COMMUNICATION ERROR (REMOTE BUF) | Remote buffer connection alarm has generated. Confirm the number of cables, parameters and I/O device. |

| Number | Meaning | Contents and remedy |
|--------|---|---|
| 181 | FORMAT ERROR IN G81 BLOCK | G81 block format error (hobbing machine) 1) T (number of teeth) has not been instructed. 2) Data outside the command range was instructed by either T, L, Q or P. |
| | (Hobbing machine) | Modify the program. |
| 182 | G81 NOT COMMANDED (Hobbing machine) | G83 (C axis servo lag quantity offset) was instructed though synchro- nization by G81 has not been instructed. Correct the program. (hobbing machine) |
| 183 | DUPLICATE G83 (COMMANDS) (Hobbing machine) | G83 was instructed before canceled by G82 after compensating for the C axis servo lag quantity by G83. (hobbing machine) |
| 184 | ILLEGAL COMMAND IN G81 (Hobbing machine) | A command not to be instructed during synchronization by G81 was instructed. (hobbing machine) 1) A C axis command by G00, G27, G28, G29, G30, etc. was instructed. 2) Inch/Metric switching by G20, G21 was instructed. |
| 185 | | G81 was instructed without performing reference position return after |
| 100 | (Hobbing machine) | power on or emergency stop. (hobbing machine) Perform reference position return. |
| 186 | PARAMETER SETTING ERROR | Parameter error regarding G81 (hobbing machine) |
| | (Hobbing machine) | The C axis has not been set to be a rotary axis. A hob axis and position coder gear ratio setting error |
| 190 | ILLEGAL AXIS SELECT | In the constant surface speed control, the specified axis command (P) contains an illegal value. Correct the program. |
| 194 | SPINDLE COMMAND IN SYN- CHRO–MODE | Cs contour control or rigid tapping was specified during serial spindle synchronous control. Correct the program. |
| 195 | SPINDLE CONTROL MODE SWITCH | The serial spindle control mode was not switched. Check the PMC lad- der program. |
| 197 | C-AXIS COMMANDED IN SPINDLE MODE | A command for Cs–axis movement was issued when the current control mode is not serial spindle Cs contour control. Check the PMC ladder program or machining program. |
| 199 | MACRO WORD UNDEFINED | Undefined macro word was used. Modify the custom macro. |
| 200 | ILLEGAL S CODE COMMAND | In the rigid tap, an S value is out of the range or is not specified. Modify the program. |
| 201 | FEEDRATE NOT FOUND IN RIGID | In the rigid tap, no F value is specified. Correct the program. |
| 202 | POSITION LSI OVERFLOW | In the rigid tap, spindle distribution value is too large. |
| 203 | PROGRAM MISS AT RIGID TAP- PING | In the rigid tap, position for a rigid M code (M29) or an S command is in- correct. Modify the program. |
| 204 | ILLEGAL AXIS OPERATION | In the rigid tap, an axis movement is specified between the rigid M code (M29) block and G84 (G74) block. Modify the program. |
| 205 | RIGID MODE DI SIGNAL OFF | Rigid mode DI signal is not ON when G84 (G74) is executed though the rigid M code (M29) is specified.Consult the PMC ladder diagram to find the reason the DI signal (DGNG061.1) is not turned on. Modify the program. |
| 206 | CAN NOT CHANGE PLANE (RIGID TAP) | Plane changeover was instructed in the rigid mode. Correct the program. |
| 210 | CAN NOT COMAND M198/M199 | M198 and M199 are executed in the schedule operation. M198 is executed in the DNC operation. |
| 211 | CAN NOT COMMAND HIGH– SPEED SKIP | A high–speed skip (G31) was specified during the feed–per–rotation or rigid tapping mode. Correct the program. |
| 212 | ILLEGAL PLANE SELECT | The arbitrary angle chamfering or a corner R is commanded or the plane including an additional axis. Correct the program. |

| Number | Meaning | Contents and remedy |
|--------|---|---|
| 213 | ILLEGAL COMMAND IN SYN- CHRO-MODE | Any of the following alarms occurred in the operation with the simple synchronization control. 1) The program issued the move command to the slave axis. 2) The program issued the manual continuous feed/manual handle feed/incremental feed command to the slave axis. 3) The program issued the automatic reference position return command without executing the manual reference position return after the power was turned on. 4) The difference between the position error amount of the master and slave axes exceeded the value specified in parameter . |
| 214 | ILLEGAL COMMAND IN SYN- CHRO–MODE | Coordinate system is set or tool compensation of the shift type is executed in the synchronous control. Correct the program. |
| 222 | DNC OP. NOT ALLOWED IN BG EDIT | Input and output are executed at a time in the background edition. Execute a correct operation. |
| 224 | RETURN TO REFERENCE POINT | Reference position return has not been performed before the automatic operation starts. Perform reference position return. |
| 230 | R CODE NOT FOUND | The infeed quantity R has not been instructed for the G160 block of th ecanned grinding cycle. Or the R command value is negative. Correct the program. |
| 250 | SIMULTANEOUS M06 AND Z-AXIS MOVEMENT NOT ALLOWED | A tool change (M06) and a Z–axis movement were specified simulta- neously in the DRILL MATE. Correct the program. |

2) Background edit alarm

| Number | Meaning | Contents and remedy |
|--------|------------------------|---|
| ??? | BACKGROUND EDIT ALARM | BP/S alarm occurs in the same number as the P/S alarm that occurs in ordinary program edit. (070, 071, 072, 073, 074 085,086,087 etc.) |
| 140 | SELECTED PROGRAM ALARM | It was attempted to select or delete in the background a program being selected in the foreground. Use background editing correctly. |

| 3) | Absolute | pulse | coder | (APC) | alarm |
|----|----------|-------|-------|-------|-------|
|----|----------|-------|-------|-------|-------|

| Number | Meaning | Contents and remedy |
|--------|--|---|
| 3n0 | nth-axis origin return | Manual reference position return is required for the nth-axis. |
| 3n1 | APC alarm: nth-axis communication | nth–axis APC communication error. Failure in data transmission Possible causes include a faulty APC, cable, or servo interface module. |
| 3n2 | APC alarm: nth-axis over time | nth–axis APC overtime error. Failure in data transmission. Possible causes include a faulty APC, cable, or servo interface module. |
| 3n3 | APC alarm: nth-axis framing | nth–axis APC framing error. Failure in data transmission. Possible causes include a faulty APC, cable, or servo interface module. |
| 3n4 | APC alarm: nth-axis parity | nth–axis APC parity error. Failure in data transmission. Possible causes include a faulty APC, cable, or servo interface module. |
| 3n5 | APC alarm: nth-axis pulse error | nth–axis APC pulse error alarm. APC alarm.APC or cable may be faulty. |
| 3n6 | APC alarm: nth–axis battery voltage 0 | nth–axis APC battery voltage has decreased to a low level so that the data cannot be held. APC alarm. Battery or cable may be faulty. |
| 3n7 | APC alarm: nth-axis battery low 1 | nth–axis axis APC battery voltage reaches a level where the battery must be renewed. APC alarm. Replace the battery. |
| 3n8 | APC alarm: nth-axis battery low 2 | nth–axis APC battery voltage has reached a level where the battery must be renewed (including when power is OFF). APC alarm. |

4) Serial pulse coder (SPC) alarms

When either of the following alarms is issued, a possible cause is a faulty serial pulse coder or cable.

| Number | Meaning | Contents |
|--------|------------------------------------|--|
| 3n9 | SPC ALARM: n AXIS PULSE COD- ER | The n axis (axis 1–8) pulse coder has a fault. |

 The details of serial pulse coder alarm No.3n9 The details of serial pulse coder alarm No. 3n9 are displayed in the diagnosis display (No. 760 to 767, 770 to 777) as shown below.

| | #7 | #6 | #5 | #4 | #3 | #2 | #1 | #0 |
|------------|----|------|------|------|------|------|------|------|
| 760 to 767 | | CSAL | BLAL | PHAL | RCAL | BZAL | CKAL | SPHL |

| CSAL | : | The serial pulse coder is defective. Replace it. |
|------|---|--|
| BLAL | : | The battery voltage is low. Replace the batteries. This alarm |
| | | has nothing to do with alarm (serial pulse coder alarm). |
| PHAL | : | The serial pulse coder or feedback cable is defective. replace |
| | | the serial pulse coder or cable. |
| RCAL | : | The serial pulse coder is defective. Replace it. |
| BZAL | : | The pulse coder was supplied with power for the first time. |
| | | Make sure that the batteries are connected. |
| | | Turn the power off, then turn it on again and perform a |
| | | reference position return. This alarm has nothing to do with |
| | | alarm (serial pulse coder alarm). |
| CKAL | : | The serial pulse coder is defective. Replace it. |
| SPHL | : | The serial pulse coder or feedback cable is defective. |
| | | Replace the serial pulse coder or cable. |

| | #7 | #6 | #5 | #4 | #3 | #2 | #1 | #0 |
|------------|-------|--------|--------|----|----|----|----|----|
| 770 to 777 | DTERR | CRCERR | STBERR | | | | | |

- **DTERR** : The serial pulse coder encountered a communication error. The pulse coder, feedbak cable, or feedback receiver circuit is defective. Replace the pulse coder, feedback cable, or NC-axis board
- **CRCERR**: The serial pulse coder encountered a communication error. The pulse coder, feedback cable, or feedback receiver circuit is defective. Replace the pulse coder, feedback cable, or NC-axis board.
- **STBERR** : the serial pulse coder encountered a communication error. The pulse coder, feedback cable, or feedback receiver circuit is defective.

Replace the pulse coder, feedback cable, or NC-axis board.

5) Servo alarms

| Number | Meaning | Contents and remedy |
|--------|---|---|
| 400 | SERVO ALARM: 1, 2TH AXIS OVERLOAD | 1–axis, 2–axis overload signal is on. Refer to diagnosis display No. 720 or 721 for details. |
| 401 | SERVO ALARM: 1, 2TH AXIS VRDY OFF | 1-axis, 2-axis servo amplifier READY signal (DRDY) went off. |
| 402 | SERVO ALARM: 3, 4TH AXIS OVERLOAD | 3-axis, 4-axis overload signal is on. Refer to diagnosis display No. 722 or 723 for details. |
| 403 | SERVO ALARM: 3, 4TH AXIS VRDY OFF | 3-axis, 4-axis servo amplifier READY signal (DRDY) went off. |
| 404 | SERVO ALARM: n-TH AXIS VRDY ON | Even though the n-th axis (axis 1-8) READY signal (MCON) went off, the servo amplifier READY signal (DRDY) is still on. Or, when the power was turned on, DRDY went on even though MCON was off. Check that the axis card and servo amplifierr are connected. |
| 405 | SERVO ALARM: ZERO POINT RE- TURN FAULT | Position control system fault. Due to an NC or servo system fault in the reference position return, there is the possibility that reference position return could not be executed correctly. Try again from the manual reference position return. |
| 4n0 | SERVO ALARM: n–TH AXIS – EX- CESS ERROR | The position deviation value when the n–th axis stops is larger than the set value. Note) Limit value must be set to parameter for each axis. |
| 4n1 | SERVO ALARM: n–TH AXIS – EX- CESS ERROR | The position deviation value when the n-th axis moves is larger than the set value. Note) Limit value must be set to parameter for each axis. |
| 4n3 | SERVO ALARM: n-th AXIS - LSI OVERFLOW | The contents of the error register for the n–th axis exceeded $\pm 2^{31}$ power. This error usually occurs as the result of an improperly set parameters. |
| 4n4 | SERVO ALARM: n–TH AXIS – DETECTION RELATED ERROR | N-th axis digital servo system fault. Refer to diagnosis display No. 720 and No.727 for details. |
| 4n5 | SERVO ALARM: n–TH AXIS – EX- CESS SHIFT | A speed higher than 4000000 units/s was attempted to be set in the n-th axis. This error occurs as the result of improperly set CMR. |

— 454 —

| Number | Meaning | Contents and remedy |
|--------|---|--|
| 4n6 | SERVO ALARM: n-TH AXIS - DIS- CONNECTION | Position detection system fault in the n-th axis pulse coder (disconnection alarm). |
| 4n7 | SERVO ALARM: n-TH AXIS - PA- RAMETER INCORRECT | This alarm occurs when the n-th axis is in one of the conditions listed below. (Digital servo system alarm) 1) The value set in Parameter No. 8n20 (motor form) is out of the specified limit. 2) A proper value (111 or -111) is not set in parameter No. 8n22 (motor revolution direction). 3) Illegal data (a value below 0, etc.) was set in parameter No. 8n23 (number of speed feedback pulses per motor revolution). 4) Illegal data (a value below 0, etc.) was set in parameter No. 8n24 (number of position feedback pulses per motor revolution). 5) Parameters No. 8n84 and No. 8n85 (flexible field gear rate) have not been set. 6) An axis selection parameter (from No. 269 to 274) is incorrect. 7) An overflow occurred during parameter computation. |

NOTE

If an excessive spindle error alarm occurs during rigid tapping, the relevant alarm number for the tapping feed axis is displayed.

| Details of servo | The detailed descriptions of servo alarm number 414 are displayed with |
|--------------------------------------|--|
| alarm No.4n4 | diagnosis numbers 720 to 727 in the sequence of axis numbers. |

| | #7 | #6 | #5 | #4 | #3 | #2 | #1 | #0 |
|------------|-----|----|-----|------|-----|------|------|------|
| 720 to 727 | OVL | LV | OVC | HCAL | HVA | DCAL | FBAL | OFAL |

| OVL | : | An overload alarm is being generated. |
|------|---|---|
| | | (This bit causes servo alarm No. 400, 402, 406, 490). |
| T X7 | | A low voltage clamp is being generated in service own |

- **LV** : A low voltage alarm is being generated in servo amp. Check LED.
- **OVC** : A overcurrent alarm is being generated inside of digital servo.
- HCAL: An abnormal current alarm is being generated in servo amp. Check LED.
- **HVAL** : An overvoltage alarm is being generated in servo amp. Check LED.
- **DCAL** : A regenerative discharge circuit alarm is being generated in servo amp. Check LED.
- **FBAL** : A disconnection alarm is being generated. (This bit causes servo alarm No.4n6.)
- **OFAL** : An overflow alarm is being generated inside of digital servo.

6) Spindle alarms

| Number | Meaning | Contents and remedy |
|--------|------------------------------------|---|
| 408 | SPINDLE SERIAL LINK START FAULT | This alarm is generated when the spindle control unit is not ready for starting correctly when the power is turned on in the system with the serial spindle. The four reasons can be considered as follows: |
| | | An improperly connected optic cable, or the spindle control unit's power is OFF. When the NC power was turned on under alarm conditions other than SU–01 or AL–24 which are shown on the LED display of the spindle control unit. In this case, turn the spindle amplifier power off once and perform startup again. Other reasons (improper combination of hardware) This alarm does not occur after the system including the spindle control unit is activated. |
| 409 | SPINDLE ALARM DETECTION | A spindle amplifier alarm occurred in a system with a serial spindle. The alarm is indicated as "AL–XX" (where XX is a number) on the display of the spindle amplifier. For details, see 14. Setting bit 7 of parameter No. 0397 causes the spindle amplifier alarm number to appear on the screen. |

7) Over travel alarms

| Number | Meaning | Contents and remedy |
|--------|------------------|---|
| 5n0 | OVER TRAVEL : +n | Exceeded the n-th axis + side stored stroke limit 1, 2. |
| 5n1 | OVER TRAVEL :n | Exceeded the n-th axis - side stored stroke limit 1, 2. |
| 5n4 | OVER TRAVEL : +n | Exceeded the n-th axis + side hardware OT. |
| 5n5 | OVER TRAVEL : -n | Exceeded the n-th axis - side hardware OT. |

8) Macro alarms

| Number | Meaning | Contents and remedy |
|---------------|-------------|--|
| 500 to 599 | MACRO ALARM | This alarm is related to the custom macro, macro executor, or order- made macro (including conversational program inputs). Refer to the relevant manual for details. (The macro alarm number may coincide with an overtravel alarm number. However, they can be distinguished from each other because the overtravel alarm number is accompanied with the description of the alarm. |

9) PMC alarms

| Number | Meaning | Contents and remedy |
|--------|--|--|
| 600 | PMC ALARM : INVALID INSTRUC- TION | An invalid-instruction interrupt occurred in the PMC. |
| 601 | PMC ALARM : RAM PARITY | A PMC RAM parity error occurred. |
| 602 | PMC ALARM : SERIAL TRANSFER | A PMC serial transfer error occurred. |
| 603 | PMC ALARM : WATCHDOG | A PMC watchdog timer alarm occurred. |
| 604 | PMC ALARM : ROM PARITY | A PMC ROM parity error occurred. |
| 605 | PMC ALARM : OVER STEP | The maximum allowable number of PMC ladder program steps was exceeded. |
| 606 | PMC ALARM : I/O MODULE AS- SIGNMENT | The assignment of I/O module signals is incorrect. |
| 607 | PMC ALARM : I/O LINK | An I/O link error occurred. The details are listed below. |

| Number | Details of PMC alarm (No. 607) |
|--------|---|
| 010 | * Communication error (SLC (master) internal register error) |
| 020 | * An SLC RAM bit error occurred (verification error). |
| 030 | * An SLC RAM bit error occurred (verification error). |
| 040 | No I/O unit has been connected. |
| 050 | 32 or more I/O units are connected. |
| 060 | * Data transmission error (no response from the slave) |
| 070 | * Communication error (no response from the slave) |
| 080 | * Communication error (no response from the slave) |
| 090 | An NMI (for other than alarm codes 110 to 160) occurred. |
| 130 | * An SLC (master) RAM parity error occurred (detected by hardware). |
| 140 | * An SLC (slave) RAM parity error occurred (detected by hardware). |
| 160 | * SLC (slave) communication error * AL0 : Watchdog timer DO clear signal received * IR1 : CRC or framing error Watchdog timer alarm Parity error |

Hardware errors are indicated with an asterisk (*).

10) Overheat alarms

| Number | Meaning | Contents and remedy |
|--------|------------------------|--|
| 700 | OVERHEAT: CONTROL UNIT | Control unit overheat Check that the fan motor operates normally, and clean the air filter. |

11) System alarms

(These alarms cannot be reset with reset key.)

| Number | Meaning | Contents and remedy |
|--------|---------------------------------------|--|
| 910 | MAIN RAM PARITY | This RAM parity error is related to low-order bytes. Replace the memory PC board. |
| 911 | MAIN RAM PARITY | This RAM parity error is related to high-order bytes. Replace the memory PC board. |
| 912 | SHARED RAM PARITY | This parity error is related to low–order bytes of RAM shared with the digital servo circuit. Replace the axis control PC board. |
| 913 | SHARED RAM PARITY | This parity error is related to high–order bytes of RAM shared with the digital servo circuit. Replace the axis control PC board. |
| 914 | SERVO RAM PARITY | This is a local RAM parity error in the digital servo circuit. Replace the axis control PC board. |
| 915 | LADDER EDITING CASSETTE RAM PARITY | This RAM parity error is related to low–order bytes of the ladder editing cassette. Replace the ladder editing cassette. |
| 916 | LADDER EDITING CASSETTE RAM PARITY | This RAM parity error is related to high–order bytes of the ladder editing cassette. Replace the ladder editing cassette. |
| 920 | WATCHDOG ALARM | This is a watchdog timer alarm or a servo system alarm for axis 1 to 4. Replace the master or axis control PC board. |
| 921 | SUB CPU WATCHDOG ALARM | This is a watchdog timer alarm related to the sub–CPU board or a servo system alarm for axis 5 or 6. Replace the sub–CPU board or the axis–5/6 control PC board. |
| 922 | 7/8 AXIS SERVO SYSTEM ALARM | This is a servo system alarm related to axis 7 or 8. Replace the axis–7/8 control PC board. |
| 930 | CPU ERROR | This is a CPU error. Replace the master PC board. |
| Number | Meaning | Contents and remedy |
|--------|--|--|
| 940 | PC BOARD INSTALLATION ERROR | PC board installation is incorrect. Check the specification of the PC board. |
| 941 | MEMORY PC BOARD CONNEC- TION ERROR | The memory PC board is not connected securely. Ensure that the PC board is connected securely. |
| 945 | SERIAL SPINDLE COMMUNICA- TION ERROR | The hardware configuration is incorrect for the serial spindle, or a com- munication alarm occurred. Check the hardware configuration of the spindle. Also ensure that the hardware for the serial spindle is con- nected securely. |
| 946 | SECOND SERIAL SPINDLE COM- MUNICATION ERROR | Communication is impossible with the second serial spindle. Ensure that the second serial spindle is connected securely. |
| 950 | FUSE BLOWN ALARM | A fuse has blown. Replace the fuse (+24E; F14). |
| 960 | SUB CPU ERROR | This is a sub–CPU error. Replace the sub–CPU PC board. |
| 998 | ROM PARITY | This is a ROM parity error. Replace the ROM board in which the error occurred. |

12) External alarm

| Number | Meaning | Contents and remedy |
|--------|----------------|---|
| 1000 | EXTERNAL ALARM | This alarm was detected by the PMC ladder program. Refer to the relevant manual from the machine builder for details. |

13) Alarms Displayed on spindle Servo Unit

| Alarm No. | Meaning | Description | Remedy |
|----------------|--|---|--|
| "A" display | Program ROM abnormality (not installed) | Detects that control program is not started (due to program ROM not installed, etc.) | Install normal program ROM |
| AL01 | Motor overheat | Detects motor speed exceeding specified speed excessively. | Check load status. Cool motor then reset alarm. |
| AL02 | Excessive speed deviation | Detects motor speed exceeding specified speed excessively. | Check load status. Reset alarm. |
| AL03 | DC link section fuse blown | Detects that fuse F4 in DC link section is blown (models 30S and 40S). | Check power transistors, and so forth. Replace fuse. |
| AL04 | Input fuse blown. Input power open phase. | Detects blown fuse (F1 to F3), open phase or mo- mentary failure of power (models 30S and 40S). | Replace fuse. Check open phase and power supply refenerative circuit operation. |
| AL05 | Control power supply fuse blown | Detects that control power supply fuse AF2 or AF3 is blown (models 30S and 40S). | Check for control power supply short circuit . Replace fuse. |
| AL07 | Excessive speed | Detects that motor rotation has exceeded 115% of its rated speed. | Reset alarm. |
| AL08 | High input voltage | Detects that switch is flipped to 200 VAC when input voltage is 230 VAC or higher (models 30S and 40S). | Flip switch to 230 VAC. |
| AL09 | Excessive load on main cir- cuit section | Detects abnormal temperature rise of power tran- sistor radiator. | Cool radiator then reset alarm. |
| AL-10 | Low input voltage | Detects drop in input power supply voltage. | Remove cause, then reset alarm. |
| AL-11 | Overvoltage in DC link sec- tion | Detects abnormally high direct current power sup- ply voltage in power circuit section. | Remove cause, then reset alarm. |
| AL-12 | Overcurrent in DC link section | Detects flow of abnormally large current in direct current section of power cirtcuit | Remove cause, then reset alarm. |

| Alarm No. | Meaning | Description | Remedy |
|--------------|--|---|----------------------------------|
| AL-13 | CPU internal data memory abnormality | Detects abnormality in CPU internal data memory. This check is made only when power is turned on. | Remove cause, then reset alarm. |
| AL-15 | Spindle switch/output switch alarm | Detects incorrect switch sequence in spindle switch/output switch operation. | Check sequence. |
| AL-16 | RAM abnormality | Detects abnormality in RAM for external data. This check is made only when power is turned on. | Remove cause, then reset alarm. |
| AL-18 | Program ROM sum check er- ror | Detects program ROM data error. This check is made only when power is turned on. | Remove cause, then reset alarm. |
| AL-19 | Excessive U phase current detection circuit offset | Detects excessive U phase current detection ciu- cuit offset. This check is made only when power is turned on. | Remove cause, then reset alarm. |
| AL-20 | Excessive V phase current detection circuit offset | Detects excessive V phase current detection cir- cuit offset. This check is made only when power is turned on. | Remove cause, then reset alarm. |
| AL-24 | Serial transfer data error | Detects serial transfer data error (such as NC pow- er supply turned off, etc.) | Remove cause, then reset alarm. |
| AL-25 | Serial data transfer stopped | Detects that serial data transfer has stopped. | Remove cause, then reset alarm. |
| AL-26 | Disconnection of speed detection signal for Cs con- touring control | Detects abnormality in position coder signal(such as unconnected cable and parameter setting error). | Remove cause, then reset alarm. |
| AL-27 | Position coder signal discon- nection | Detects abnormality in position coder signal (such as unconnected cable and adjustment error). | Remove cause, then reset alarm. |
| AL-28 | Disconnection of position detection signal for Cs con- touring control | Detects abnormality in position detection signal for Cs contouring control (such as unconnected cable and adjustment error). | Remove cause, then reset alarm. |
| AL–29 | Short-time overload | Detects that overload has been continuously ap- plied for some period of time (such as restraining motor shaft in positioning). | Remove cause, then reset alarm. |
| AL-30 | Input circuit overcurrent | Detects overcurrent flowing in input circuit. | Remove cause, then reset alarm. |
| AL-31 | Speed detection signal dis- connection motor restraint alarm or motor is clamped. | Detects that motor cannot rotate at specified speed or it is detected that the motor is clamped. (but rotates at very slow speed or has stopped). (This includes checking of speed detection signal cable.) | Remove cause, then reset alarm. |
| AL-32 | Abnormality in RAM internal to LSI for serial data transfer. This check is made only when power is turned on. | Detects abnormality in RAM internalto LSI for seri- al data transfer. This check is made only when power is turned on. | Remove cause, then reset alarm. |
| AL-33 | Insufficient DC link section charging | Detects insufficient charging of direct current pow- er supply voltage in power circuit section when magnetic contactor in amplifier is turned on (such as open phase and defectifve charging resistor). | Remove cause, then reset alarm. |
| AL-34 | Parameter data setting be- yond allowable range of val- ues | Detects parameter data set beyond allowable range of values. | Set correct data. |
| AL-35 | Excesive gear ratio data set- ting | Detects gear ratio data set beyond allowable range of values. | Set correct data. |
| AL36 | Error counter over flow | Detects error counter overflow. | Correct cause, then reset alarm. |
| AL-37 | Speed detector parameter setting error | Detects incorrect setting of parameter for number of speed detection pulses. | Set correct data. |

| Alarm No. | Meaning | Description | Remedy | |
|--------------|--|--|--|--|
| AL-39 | Alarm for indicating failure in detecting 1–rotation signal for Cs contouring control | Detects 1–rotaion signal detection failure in Cs contouring contorl. | Make 1–rotaion signal ad- justment. Check cable shield status. | |
| AL-40 | Alarm for indicating 1–rotation signal for Cs contouring con- trol not detected | Detects that 1–rotation signal has not occurred in Cs contouring control. | Make 1–rotaion signal ad- justment. | |
| AL-41 | Alarm for indicating failure in detecting position coder 1–ro- taion signal. | Detects failure in detecting position coder 1–rota- tion signal. | Make signal adjustment for signal conversion cir- cuit. Check cable shield status. | |
| AL-42 | Alarm for indicating position coder 1-rotation signal not detected | Detects that position coder 1–rotation signal has not issued. | Make 1–rotation signal ad- justment for signal conver- sion circuit. | |
| AL-43 | Alarm for indicating discon- nection of position coder sig- nal for differential speed mode | Detects that main spindle position coder signal used for differential speed mode is not connected yet (or is disconnected). | Check that main spindle position coder signal is connected to connector CN12. | |
| AL-46 | Alarm for indicating failure in detecting position coder 1–rotation signal in thread cutting operation. | Detects failure in detecting position coder 1–rota- tion signasl in thread cutting operation. | Make 1–rotation signal ad- justment for signal conver- sion circuit Check cable shield status. | |
| AL-47 | Position coder signal ab- normality | Detects incorrect position coder signal count operation. | Make signal adjustment for signal conversion cir- cuit. Check cable shield status. | |
| AL48 | Position coder 1–rotation sig- nal abnormailty | Detects that occurrence of position coder 1-rota- tion signal has stopped. | Make 1–rotation signal ad- justment for signal conver- sion circuit. | |
| AL-49 | The converted differential speed is too high. | Detects that speed of other spindle converted to speed of local spindle has exceeded allowable limit in differential mode. | Check the position coder state of the other side. | |
| AL-50 | Excessive speed command calculation value in spindle synchronization control | Detects that speed command calculation value exceeded allowable range in spindle synchronization control. | Check parameters such as a position gain. | |
| AL51 | Undervoltage at DC link sec- tion | Detects that DC power supply voltage of power ciucuit has dropped (due to momentary power fail- ure or loose contact of magnetic contactor). | Remove cause, then reset alarm. | |
| AL-52 | ITP signal abnormality I | Detects abnormality in synchronization signal (ITP signal)with CNC (such as loss of ITP signal). | Remove cause, then reset alarm. | |
| AL-53 | ITP signal abnormality II | Detects abnormality in synchronization signal (ITP signal) with CNC (such as loss of ITP signal). | Remove cause, then reset alarm. | |
| AL-54 | Overload current alarm | Detects that excessive current flowed in motor for long time. | Check if overload opera- tion or frequent accelera- tion/deceleration is per- formed. | |
| AL-55 | Power line abnormality in spindle switching/output switching | Detects that switch request signal does not match power line status check signal. | Check operation of mag- netic contractor for power line switching. Check if power line status check signal is processed normally. | |

OPERATION OF PORTABLE TAPE READER

Portable tape reader is the device which inputs the NC program and the data on the paper tape to CNC.

• Names and descriptions of each section



| Table 1 Description of Each | Section |
|-----------------------------|---------|
|-----------------------------|---------|

| No. | Name | Descriptions | | |
|-----|-------------------------------|---|--|--|
| 1 | Light Sources | An LED (Light emitting diode) is mounted for each channel and for the feed hole (9 diodes in total). A built–in Stop Shoe functions to decelerate the tape. The light source is attracted to the optical reader by a magnet so that the tape will be held in the correct position. This unit can be opened upward, by turning the tape reader control switch to the RELEASE position (this turns off the magnet). | | |
| 2 | Optical Reader | Reads data punched on the tape, through a glass window. Dust or scratches on the glass window can result in reading errors. Keep this window clean. | | |
| 3 | Capstan Roller | Controls the feeding of tape as specified by the control unit. | | |
| 4 | Tape Reader Control Switch | A 3-position switch used to control the Tape Reader. RELEASE The tape is allowed to be free, or used to open the light-source. When loading or unloading the tape, select this position. AUTO The tape is set to fixed position by the Stop Shoe. The feed and stop of the tape is controlled by the CNC. To input data from tape, the Light Source must be closed and this position must be selected. MANUAL The tape can be fed in the forward reading direction. if another position is selected, the tape feed is stopped. | | |
| 5 | Tape Box | A Tape Box is located below the Tape Reader. A belt used to draw out a pa- per tape is located inside the box. The paper tape can easily be pulled out using this belt. The tape box accomodates 15 meters of tape. | | |
| 6 | Handle | Used to carry the tape reader. | | |
| 7 | Winder | Used to advance or rewind the tape. | | |
| 8 | Metal A | Fastener (usually kept open) Push Paper tape Vhen removing the rolled tape, reduce the internal diameter by pushing the fastener. | | |
| 9 | Cover lock | Be sure to use the lock for fastening the cover before carrying the tape read- er. | | |

— 462 —

| No. | Name | Descriptions |
|-----|-----------------------------------|---|
| 10 | | When the tape reader is raised, the latch mechanism is activated to fix the tape reader. Thus, the tape reader is not lowered. The latch is locked with the lowering lock lever. The latch is therefore not unlocked even when the tape reader is raised with the handle. |
| | Lowering lock lever | When the latch is locked, the lever is horizontal. To store the tape reader in the box, push the lever to release the lock, then raise the tape reader with the handle to unlock the latch. |
| | | When the latch is unlocked, the tape reader can be stored in the box. |
| | | When storing the tape reader, secure it with the cover lock. |
| 11 | Cable storage | Used to store rolled power and signal cables. The cable length is 1.5 m. |
| 12 | Photoamplifier | For the tape reader |
| 13 | Reader/punch interface adapter | 200 VAC input and 5 VDC output power and reader/punch interface adapter PCB |

Procedure for Operating the Portable Tape Reader

Preparations

Setting the tape

- 1 Unlock the cover locks **9**. Raise the tape reader with the handle **6** until it clicks, then lower the tape reader. The tape reader then appears and is secured. Check that the lowering lock levers **10** are horizontal.
- **2** Take out the signal and power cables from the cable storage **11** and connect the signal cable with the CNC reader/punch interface port and the power cable with the power supply.
- **3** Turn the control switch to the RELEASE position.
- 4 Lift the Light Source Unit, and insert an NC tape between the gap. The tape must be positioned as shown in the figure, when viewed looking downward.



- **5** Pull the tape until the top of the tape goes past the Capstan roller.
- **6** Check that the NC tape is correctly positioned by the Tape Guide.
- 7 Lower the Light Source.
- **8** Turn the switch to the AUTO position.
- **9** Suspend the top and rear–end of the tape in the Tape Box.

| Removing the tape | 10 Turn the switch to the RELEASE position. | | | | |
|-------------------|---|--|--|--|--|
| | 11 Lift the Light Source and remove the tape. | | | | |
| | 12 Lower the Light Source | | | | |
| Storage | 13 Store the cables in the cable storage 11 . | | | | |
| | 14 Push the lowering lock lever 10 at both sides down. | | | | |
| | 15 Raise the tape reader with the handle 6 to unlock the latch, then gently lower it. | | | | |
| | 16 Lock the cover lock 9 and carry the tape reader with the handle 6 . | | | | |
| | CAUTION SETTING OF A TAPE | | | | |
| | When the NC tape is loaded, the Label Skip function activates to read but skip data until first End of Block code (CR in EIA code or LF in ISO code) is read. When loading an NC tape, the location within the tape, from which data reading should be started must properly be selected and the NC tape should be set as shown in the figure below. | | | | |
| | ISO code. | | | | |
| | DISCONNECTION AND CONNECTION OF A PORTABLE TAPE READER CONNECTION CABLE | | | | |
| | Don't disconnect or connect CNC tape reader connection cable (signal cable) without turning off the CNC power supply, otherwise the PCB of the tape reader and master PCB of CNC controller may be broken.Turn off the CN power supply before disconnecting or connecting the connection cable, accordingly. | | | | |

— 464 —

Index

[A]

Absolute and Incremental Programming (G90, G91), 69 Alarm and Self–diagnosis Functions, 324 Alarm Display, 325 Alarm Display (See III–7.1), 271 Alarm List, 445 Altering a Word, 352 Argument specification, 200 Arithmetic and Logic Operation, 224 Automatic Grinding Wheel Diameter Compensation after Dressing, 142 Automatic Insertion of Sequence Numbers, 371 Automatic Operation, 263, 302 Auxiliary Function, 77 Auxiliary Function (M Function), 78

[B]

Background Editing, 366 Boring Cycle (G85), 115 Boring Cycle (G86), 117 Boring Cycle (G88), 121 Boring Cycle (G89), 123 Boring Cycle Back Boring Cycle (G87), 119 Branch and Repetition, 229

[C]

Canned Cycle, 97 Canned Cycle Cancel (G80), 124, 132 Canned Grinding Cycle (for 0–GSD only), 133 Changing Workpiece Coordinate System, 63 Character–to–Codes Correspondence Table, 444 Check by Running the Machine, 264 Checking by Self–diagnostic Screen, 327 Checking the Minimum Grinding Wheel Diameter (for 0–GSD only), 142 Circular Interpolation (G02,G03), 39 Command for Machine Operations – Miscellaneous Function, 23 Compensation Function, 145 Conditional Branch (IF Statement), 229 Continuous–feed Surface Grinding Cycle (G78), 138 Controlled Axes, 29, 30 Coordinate System, 59 Coordinate System on Part Drawing and Coordinate System Specified by CNC - Coordinate System, 16 Coordinate Value and Dimension, 68 Copying an Entire Program, 359 Copying Part of a Program, 360 Creating Programs, 369 Creating Programs Using The Mdi Panel, 370 CRT/MDI Panels and LCD/MDI Panels, 275 Current Block Display Screen, 390 Current Position Display (See III-11.1.1 to 11.1.3), 271 Custom Macro A, 196 Custom Macro B, 214 Custom Macro Command, 197 Cutting Feed, 49 Cutting Feedrate Control, 51 Cutting Speed - Spindle Speed Function, 21

[D]

Data Input/Output, 329 Data Output (See III-8), 273 Decimal Point Programming, 71 Deleting a Block, 354 Deleting a Word, 353 Deleting All Programs, 357 Deleting Blocks, 354 Deleting Multiple Blocks, 355 Deleting One Program, 357 Deleting Programs, 357 Details of Cutter Compensation C, 156 Direct Constant-dimension Plunge Grinding Cycle (G77), 136 Display, 270 Display of Run Time and Parts Count, 386 Display of Software Configuration, 288 Displaying and Entering Setting Data, 409 Displaying and Setting Custom Macro Common Variables, 400 Displaying and Setting Data, 267 Displaying and Setting Parameters, 402 Displaying and Setting Pitch Error Compensation Data, 404 Displaying and Setting Run Time, Parts Count, 411

Displaying and Setting the Software Operator's Panel, 414

- Displaying and Setting the Workpiece Origin Offset Value, 399
- Displaying memory used and a list of programs, 394

Displaying Operator Message, 413

Displaying the Program Number and Sequence Number, 416

Displaying the Program Number, Sequence Number, and Status, and Warning Messages for Data Setting, 416

Displaying the Status and Warning for Data Setting or Input/Output Operation, 417

DNC Operation, 307

Drilling Cycle Counter Boring Cycle (G82), 109

Drilling Cycle, Spot Drilling (G81), 107

Dry Run, 316

Dwell (G04), 54

[E]

Editing a Part Program, 266
Editing Programs, 345
Emergency Stop, 320
Exact Stop (G09, G61) Cutting Mode (G64) Tapping Mode (G63), 52
Extended Part Program Editing Function, 358
External I/O Devices, 282
External Motion Function (G81), 144
External Output Commands, 250

[F]

FANUC FA Card, 285 FANUC Floppy Cassette, 284 FANUC Handy File, 284 FANUC PPR, 285 Feed Functions, 45 Feed– Feed Function, 14 Feedrate Override, 314 File Deletion, 333 File search, 332 Files, 330 Fine Boring Cycle G76, 105 Function Keys, 279 Function Keys and Soft Keys, 278 Functions to Simplify Programming, 96

[G]

G66 (Modal call), 199General Flow of Operation of CNC Machine Tool, 5General Screen Operations, 278

[H]

Heading a Program, 350

High-speed Peck Drilling Cycle (G73), 101

How to Indicate Command Dimensions for Moving the Tool – Absolute, Incremental Commands, 19

How to View the Position Display Change without Running the Machine, 265

[I]

In–feed Grinding Along the Y and Z Axes at The end of Table Swing (for 0–GSD Only), 143

Inch/Metric Conversion(G20,G21), 70

Increment System, 30

Incremental Feed, 294

Input Command from MDI, 189

Inputting a Program, 334

Inputting and Outputting Parameters and Pitch Error Compensation Data, 341

Inputting Custom Macro B Common Variables, 343

Inputting Offset Data, 339

Inputting Parameters, 341

Inputting/Outputt Ing Custom Macro B Common Variables, 343

Inserting ,Altering and Deleting a Word, 346

Inserting a Word, 351

Interference Check, 181

Intermittent-feed Surface Grinding Cycle (G79), 140

Internal Circular Cutting Feedrate Change, 53

Interpolation Functions, 34

[J]

Jog Feed, 292

[K]

Key Input and Input Buffer, 280

Kind of Variables, 203

[L]

Left–handed Rigid Tapping Cycle (G74), 129 Left–handed Tapping Cycle (G74), 103 Limitations, 249 Linear Interpolation (G01), 37 List of Functions and Tape Format, 430 Local Coordinate System, 65

[M]

M98 (Single call), 197 Machine Coordinate System, 60 Machine Lock and Auxiliary Function Lock, 313 Macro Call, 233 Macro Call Using an M Code, 241 Macro Call Using G Code, 240 Macro Instructions (G65), 208 Macro Statements and NC Statements, 228 Manual Absolute On and Off, 297 Manual Handle Feed, 295 Manual Operation, 260, 289 Manual Reference Position Return, 290 Maximum Stroke, 30 MDI Operation, 305 Memory Operation, 303 Merging a Program, 362 Method Of Replacing Battery, 421 Mirror Image, 308 Modal Call (G66), 238 Moving Part of a Program, 361 Multiple M Commands in a Single Block, 79

[N]

Name of Axes, 30 Next Block Display Screen, 391 Nomographs, 437 Normal Direction Control (G150, G151, G152) (for 0–GSD Only), 192 Notes on Reading this Manual, 7

[0]

Offset Data Input and Output, 339 Operating Monitor Display, 387 Operation of Portable Tape Reader, 461 Operation performed when the total depth of cut is reached, 139 Operational Devices, 274 Outputting a Program, 336 Outputting Custom Macro B Common Variable, 344 Outputting Offset Data, 340 Outputting Parameters, 342 Overall Position Display, 384 Overcutting by Cutter Compensation, 186 Overtravel, 321 Overview of Cutter Compensation C (G40 – G42), 150

[P]

Part Drawing and Tool Movement, 15 Parts Count Display, Run Time Display (See III-11.1.5), 272 Peck Drilling Cycle (G83), 111 Peek Rigid Tapping Cycle (G84 or G74), 131 Plane Selection, 67 Plunge Grinding Cycle (G75), 134 Portable Tape Reader, 286 Position Display in the Relative Coordinate System, 382 Position Display in the Work Coordinate System, 381 Positioning (G00), 35 Power Disconnection, 288 Power ON/OFF, 287 Processing Macro Statements, 246 Program Check Screen, 392 Program Components other than Program Sections, 83 Program Configuration, 24, 81 Program Contents Display, 389 Program Display (See III-11.2.1 and III-11.3.1), 270 Program Input/Output, 334 Program Number Search, 356 Program Screen for MDI Operation, 393 Program Section Configuration, 86 Programmable Parameter Entry (G10)(0-GSD), 254

[R]

Radius Direction Error at Circle Cutting, 441

Range of Command Value, 434 Rapid Traverse, 48 Rapid Traverse Override, 315 Reference Position, 55 Reference Position (Machine–Specific Position), 15 Registering Custom Macro Programs, 248 reparatory Function (G Function), 31 Repetition (While Statement), 230 Replacement of Words and Addresses, 364 Replacing Batteries For Absolute Pulse Coder, 423 Replacing CNC Battery For Memory Back–up, 422 Rigid Tapping, 127 Rigid Tapping (G84), 127 Rotary Axis Roll–over, 256

[S]

| Safety Functions, 319 |
|---|
| Sample Program, 244 |
| Screens Displayed by Function Key $\left[\begin{array}{c} DGNOS\\ PARAM \end{array} \right]$, 401 |
| Screens Displayed by Function Key $MENU \\ OFSET$, 397 |
| Screens Displayed by Function Key ALARM, 413 |
| Screens Displayed by Function Key \bigcirc , 381 |
| Screens Displayed by Function Key (In Auto Mode or MDI Mode), 389 |
| Screens Displayed by Function Key |
| (in the Edit Mode), 394 |
| Selecting a Workpiece Coordinate System, 62 |
| Selection of Tool Used for Various Machining – Tool Function, 22 |
| Sequence Number Search, 310 |
| Setting a Workpiece Coordinate System, 61 |
| Setting and Displaying Data, 373 |
| Setting and Displaying the Tool Offset Value, 397 |
| Simple Call (G65), 233 |
| Single Block, 317 |
| Single Direction Positioning (G60), 36 |
| Skip Function (G31) (for 0–GSD only), 43 |
| |

Specifying the Spindle Speed Value Directly (S5–Digit Command), 74 Specifying the Spindle Speed with a Binary Code, 74 Spindle Speed Function (S Function), 73 Status when Turning Power On, when Clear and when Reset, 442 Stroke Check, 322 Subprogram, 92 Subprogram Call Using an M Code, 242 Subprogram call using M code, 197

Subprogram call using T code, 198

Subprogram Calls Using a T Code, 243

Supplementary Explanation for Copying, Moving and Merging, 363

System Variables, 218

[T]

Tape Code List, 427 Tapping Cycle (G84), 113 Test Operation, 312 Testing a Program, 264 The Second Auxiliary Functions (B Codes) (For 0-MD Only), 80 Tool CompensaTion Values, Number of Compensation Values, and Entering Values from the Program (G10), 190 Tool Figure and Tool Motion by Program, 27 Tool Function (T Function), 75 Tool Length Offset (G43,G44,G49), 146 Tool Movement Along Workpiece Parts Figure–Interpolation, 12 Tool Movement by Programing – Automatic Operation, 262 Tool Movement in Offset Mode, 161 Tool Movement in Offset Mode Cancel, 175 Tool Movement in Start-up, 157 Tool Movement Range - Stroke , 28 Tool Path at Corner, 438 Tool Selection Function, 76 Turning on the Power, 287

[U]

Unconditional Branch (GOTO Statement), 229

[V]

Variables, 215

[W]

Warning and Notes on Custom Macro, 213 Word Search, 348

Workpiece Coordinate System, 61

| | | | | Contents |
|--|--|----------------|-----------|----------|
| | | | | Date |
| | | | | Revision |
| | | Total revision | | Contents |
| | | Feb., '97 | Dec., '95 | Date |
| | | 02 | 01 | Revision |

Revision Record FANUC Series 0-MD/0-GSD OPERATOR'S MANUAL (B-62574EN)