FANUC Series 0*i*-MODEL F

DESCRIPTIONS

B-64602EN/01

The products in this manual are controlled based on Japan's "Foreign Exchange and Foreign Trade Law". The export of from Japan subject to an export license by the government of Japan. Other models in this manual may also be subject to export controls.
Further, re-export to another country may be subject to the license of the government of the country from where the product is re-exported. Furthermore, the product may also be controlled by re-export regulations of the United States government.
Should you wish to export or re-export these products, please contact FANUC for advice.

In this manual we have tried as much as possible to describe all the various matters.
However, we cannot describe all the matters which must not be done, or which cannot be done, because there are so many possibilities.
Therefore, matters which are not especially described as possible in this manual should be regarded as "impossible".

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

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 danger 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.

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

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 The Operator’s Manual and Programming 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.

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.

⚠ **CAUTION**

The liquid-crystal display is manufactured with very precise fabrication technology. Some pixels may not be turned on or may remain on. This phenomenon is a common attribute of LCDs and is not a defect.

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.

WARNINGS AND CAUTIONS RELATED TO PROGRAMMING

This section covers the major safety precautions related to programming. Before attempting to perform programming, read the supplied Operator's 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.

⚠ **WARNING**

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.

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.

⚠ **WARNING**

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 Operator's 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.

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.

⚠ WARNING

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 **RESET key**
Pressing the RESET key stops the currently running program. As a result, the servo axes are stopped. However, the RESET key may fail to function for reasons such as an MDI panel problem. So, when the motors must be stopped, use the emergency stop button instead of the RESET key to ensure security.

10 **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.

11 **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.

12 **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.

13 **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.

14 **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.

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 ⚠ 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 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 Section “Method of replacing battery” in the Operator’s Manual (Common to Lathe System/Machining Center System) 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 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 FANUC SERVO MOTOR αi series Maintenance 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 ⚠ and fitted with an insulating cover).

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

⚠ **WARNING**

When using the controller unit, display unit, MDI unit, or machine operator's panel, prevent these units from directly exposing to chips or coolants. Even if direct exposure to coolants is prevented, coolants containing sulfur or chlorine at a high activation level, oil-free synthetic-type coolants, or water-soluble coolants at a high alkali level particularly have large effects on the control unit and peripheral units, possibly causing the following failures.

- Coolants containing sulfur or chlorine at a high activation level
  Some coolants containing sulfur or chlorine are at an extremely high activity level. If such a coolant adheres to the CNC or peripheral units, it reacts chemically with a material, such as resin, of equipment, possibly leading to corrosion or deterioration. If it gets in the CNC or peripheral units, it corrodes metals, such as copper and silver, used as component materials, possibly leading to a defective component.
- Synthetic-type coolants having a high permeability
  Some synthetic-type coolants whose lubricating component is, for example, PAG (polyalkylene glycol) have an extremely high permeability. If such a coolant is used even in equipment having a high closeness, it can readily flow into the CNC or peripheral units through, for example, gaskets. It is likely that, if the coolant gets in the CNC or a peripheral unit, it may deteriorate the insulation and damage the components.
- Water-soluble coolants at a high alkali level
  Some coolants whose pH is increased using alkanolamine are so strong alkali that its standard dilution will lead to pH10 or higher. If such a coolant spatters over the surface of the CNC or peripheral unit, it reacts chemically with a material, such as resin, possibly leading to corrosion or deterioration.

TABLE OF CONTENTS

I. GENERAL

II. NC FUNCTION

3 INTERPOLATION FUNCTION 79

6 GUIDANCE FUNCTION 205

7 AUXILIARY FUNCTION / SPINDLE SPEED FUNCTION 208

APPENDIX

I. GENERAL

1 GENERAL

Applicable models

This manual describes the models indicated in the table below.
In the text, the abbreviations indicated below may be used.

Model name	Abbreviation		
FANUC Series 0*i*-TF	0*i*-TF	Series 0*i*-F	Series 0*i*
FANUC Series 0*i*-MF	0*i*-MF		

NOTE

1 For an explanatory purpose, the following descriptions may be used according to the CNC model :
  - 0i-TF : Lathe system (T series)
  - 0i-MF : Machining center system (M series)

2 Some functions described in this manual may not be applied to some products. For details, refer to the Chapter, "LIST OF SPECIFICATION".

Special symbols

This manual uses the following symbols:

- M

Indicates a description that is valid only for the machine center system.
In a general description of the method of machining, a machining center system operation is identified by a phase such as "for milling machining".

- T

Indicates a description that is valid only for the lathe system.
In a general description of the method of machining, a lathe system operation is identified by a phrase such as "for lathe cutting".

\-

Indicates the end of a description of a system control type.
When a system control type mark mentioned above is not followed by this mark, the description of the system control type is assumed to continue until the next item or paragraph begins. In this case, the next item or paragraph provides a description common to the control types.

- **IP_**

Indicates a combination of axes such as X_ Y_ Z_
In the underlined position following each address, a numeric value such as a coordinate value is placed (used in PROGRAMMING.).

- **;**

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

Related manuals of Series 0*i*- MODEL F

The following table lists the manuals related to Series 0*i*-F. This manual is indicated by an asterisk(*).

Table 1 Related manuals

Manual name	Specification number	
DESCRIPTIONS	B-64602EN	*
CONNECTION MANUAL (HARDWARE)	B-64603EN	
CONNECTION MANUAL (FUNCTION)	B-64603EN-1	
OPERATOR'S MANUAL (Common to Lathe System/Machining Center System)	B-64604EN	
OPERATOR'S MANUAL (For Lathe System)	B-64604EN-1	
OPERATOR'S MANUAL (For Machining Center System)	B-64604EN-2	
MAINTENANCE MANUAL	B-64605EN	
PARAMETER MANUAL	B-64610EN	
Programming		
Macro Executor PROGRAMMING MANUAL	B-63943EN-2	
Macro Compiler PROGRAMMING MANUAL	B-66263EN	
C Language Executor PROGRAMMING MANUAL	B-63943EN-3	
PMC		
PMC PROGRAMMING MANUAL	B-64513EN	
Network		
PROFIBUS-DP Board CONNECTION MANUAL	B-63993EN	
Industrial Ethernet CONNECTION MANUAL	B-64013EN	
Fast Ethernet / Fast Data Server OPERATOR'S MANUAL	B-64014EN	
DeviceNet Board CONNECTION MANUAL	B-64043EN	
FL-net Board CONNECTION MANUAL	B-64163EN	
CC-Link Board CONNECTION MANUAL	B-64463EN	
Operation guidance function		
MANUAL GUIDE *i* (Common to Lathe System/Machining Center System) OPERATOR'S MANUAL	B-63874EN	
MANUAL GUIDE *i* (For Machining Center System) OPERATOR'S MANUAL	B-63874EN-2	
MANUAL GUIDE *i* (Set-up Guidance Functions) OPERATOR'S MANUAL	B-63874EN-1	
MANUAL GUIDE 0*i* OPERATOR'S MANUAL	B-64434EN	
TURN MATE *i* OPERATOR'S MANUAL	B-64254EN	
Dual Check Safety		
Dual Check Safety CONNECTION MANUAL	B-64483EN-2	

Related manuals of SERVO MOTOR α*i*/β*i* series

The following table lists the manuals related to SERVO MOTOR α*i*/β*i* series

Table 2 Related manuals

Manual name	Specification number
FANUC AC SERVO MOTOR α*i* series DESCRIPTIONS	B-65262EN
FANUC AC SPINDLE MOTOR α*i* series DESCRIPTIONS	B-65272EN
FANUC AC SERVO MOTOR β*i* series DESCRIPTIONS	B-65302EN
FANUC AC SPINDLE MOTOR β*i* series DESCRIPTIONS	B-65312EN
FANUC SERVO AMPLIFIER α*i* series DESCRIPTIONS	B-65282EN
FANUC SERVO AMPLIFIER β*i* series DESCRIPTIONS	B-65322EN
FANUC SERVO MOTOR α*i*s series FANUC SERVO MOTOR α*i* series FANUC AC SPINDLE MOTOR α*i* series FANUC SERVO AMPLIFIER α*i* series MAINTENANCE MANUAL	B-65285EN

Manual name	Specification number
FANUC SERVO MOTOR β*i*s series FANUC AC SPINDLE MOTOR β*i* series FANUC SERVO AMPLIFIER β*i* series MAINTENANCE MANUAL	B-65325EN
FANUC AC SERVO MOTOR α*i* series FANUC AC SERVO MOTOR β*i* series FANUC LINEAR MOTOR L*i*S series FANUC SYNCHRONOUS BUILT-IN SERVO MOTOR D*i*S series PARAMETER MANUAL	B-65270EN
FANUC AC SPINDLE MOTOR α*i*/β*i* series, BUILT-IN SPINDLE MOTOR B*i* series PARAMETER MANUAL	B-65280EN

The above servo motors and the corresponding spindles can be connected to the CNC covered in this manual.
This manual mainly assumes that the FANUC SERVO MOTOR α*i* series of servo motor is used. For servo motor and spindle information, refer to the manuals for the servo motor and spindle that are actually connected.

2 LIST OF SPECIFICATION

○ : Standard ● : Standard option
☆ : Option * : Function included in another option
- : Not Available
Note) Some combinations of these options are restricted.
M represents a machining center system (M series).
T represents a lathe system (T series).
For (*1) to (*11) in the table, see Not below the table.

Item		Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
			M	T		
Axis control						
Max. total number of control axes (feed axes + spindle axes) / 2 path system		Total 11 axes / Up to 9 axes each path (*9)	☆	☆		1.1, 1.4
		Total 10 axes / Up to 7 axes each path	○	○		
	Max. feed axes (*8)	Total 9 axes / Up to 7 axes each path	☆	☆	R689	
		Total 8 axes / Up to 5 axes each path	○	○		
	Max. spindle axes	Total 4 axes / Up to 3 axes each path	☆	☆	R604	
		Total 2 axes / Up to 2 axes each path	○	○		
Max. total number of control axes (feed axes + spindle axes) / 1 path system		8 axes	☆	☆		1.1, 1.4
		7 axes	○	-		
		6 axes	-	○		
	Max. feed axes (*8)	7 axes	☆	☆	R689	
		5 axes	○	-		
		4 axes	-	○		
	Max. spindle axes	3 axes	-	☆	R604	
		2 axes	○	○		
Controlled path		2 path	☆	☆	S801	1.3
		1 path	○	○		
Machine groups		Max. 3 groups	☆	☆	S836	1.2
		1 group	○	○		
Max. simultaneously controlled axes (in each path)		Max. 4 axes	○	○	-	1.4
Axis control by PMC		Not available on Cs axes	○	○		1.4, 1.5
Cs contouring control			○	○		1.4, 1.6
Loader control function		Loader 1 path (*10) This cannot be ordered with Peripheral axis control.	☆	☆	R417	
Addition of loader control path		Loader 2 paths (*10) Loader control function is required.	☆	☆	R418	

Item	Specifications	Seriess 0*i*-F		Drawing No.	Section No.
		M	T		
Axis name	Basic three axes are X, Y and Z, additional axes are optional from U, V, W, A, B and C.	○	-	-	1.7
	In case of G code system A, basic 2 axes are X and Z, additional axes are optional from Y, A, B and C.	-	○		
	In case of G code system B/C, basic 2 axes are X and Z, additional axes are optional from Y, U, V, W, A, B and C.	-	○		
Axis name expansion	Max 3 characters	○	○	-	1.7.2
Arbitrary axis name setting	Included in Custom macro function	○	○	-	1.8
Spindle name expansion	Max. 3 characters. Included in Multi-spindle function.	○	○	-	1.9
Peripheral axis control	This cannot be ordered with Loader control function.	☆	☆	R725	1.10
Synchronous/Composite control		☆	☆	S816	1.11
Superimposed control	Changing function of velocity and time constant is not available. (*3)	☆	☆	S818	1.12
Superimposed Control A	Feedrate and acc/dec time of master and slave axis in superimposed control can be set individually. (*3)	☆	☆	R538	-
Synchronous/Composite/Superimposed control by program command	(*3)	☆	☆	S890	5.48
Flexible path axis assignment	(*3)	☆	☆	R607	1.13
Axis synchronous control	Max. 4 pairs	○	○	-	1.14
Angular axis control	Available on arbitrary axes.	☆	☆	J924	1.15
Tandem control		○	○	-	1.16
Tandem disturbance elimination control		☆	☆	S660	1.17
Torque control		○	○		1.18
Pole position detection function		☆	☆	S744	1.19
Control axis detach		○	○		1.20
High precision oscillation function		☆	☆	R662	1.21
Increment system	IS-A, IS-B	○	○	-	1.22
Increment system C	0.0001mm, 0.0001deg, 0.00001inch	○	○	-	1.22
Flexible feed gear	Optional DMR	○	○	-	1.23
Dual position feedback		☆	☆	J704	1.24
HRV2 control		○	○	-	1.25
HRV3 control		○	○	-	1.25
Inch/metric conversion		○	○	-	1.26
Interlock	All axes/each axis/each direction/block start/cutting block start	○	○	-	1.27
Machine lock	All axes/each axis	○	○	-	1.28
Emergency stop		○	○	-	1.29
Overtravel		○	○	-	1.30
Stored stroke check 1		○	○	-	1.31
Stored stroke check 1 area expansion		☆	☆	R552	1.32
Stroke limit external setting		○	○	-	1.33
Stored stroke check 2,3		○	○	-	1.34, 1.35
Stroke limit check before move		○	○	-	1.36

Item	Specifications	Seriess 0*i*-F		Drawing No.	Section No.
		M	T		
Stroke limit area changing function		☆	☆	R585	1.39
Stored stroke limit range switching function by signal		☆	☆	R849	1.40
Chuck and tail stock barrier		-	○	-	1.41
Mirror image	Each axis	○	○	-	1.42
Follow-up		○	○	-	1.43
Servo off/Mechanical handle		○	○	-	1.44
Chamfering on/off		-	○	-	1.45
Interference check for each path	Only for multi path control (*3)	-	☆	J839	1.46
Unexpected disturbance torque detection function		○	○	-	1.47
I/O Link β unexpected disturbance torque detection		☆	☆	S812	-
Position switch		○	○	-	1.48
High speed position switch		☆	☆	J987	1.59
Linear scale I/F with absolute address reference mark		☆	☆	J670	1.50.1
Linear scale I/F expansion with absolute address reference mark		☆	☆	S730	1.50.2
Temporary absolute coordinate setting		☆	☆	J786	1.53
Dual check safety		☆	☆	S661	1.54
Safety spindle speed limit override	Dual check safety is required.	☆	☆	R626	-
Test mode function for Acceptance Test	Dual check safety is required.	☆	☆	R671	-
Axis immediate stop function	AI contour control I or II is required.	☆	☆	R613	1.61

Operation

Automatic operation (memory)		○	○	-	2.1.1
MDI operation		○	○	-	2.1.2
DNC operation		○	○	-	2.1.3
DNC operation with memory card (*2)	CF card and PCMCIA Card Attachment is required.	○	○	-	2.1.4
Schedule function (*2)	CF card and PCMCIA Card Attachment is required (When a memory card is used). RS232C interface is required (when a Floppy Cassette is used).	○	○	-	2.1.5
Program number search		○	○	-	2.2
Sequence number search		○	○	-	2.3
Sequence number comparison and stop		○	○	-	2.4
Program restart		○	○	-	2.5
Quick program restart		☆	☆	R630	2.6
Tool retract and recover		☆	☆	J823	2.7
Manual intervention and return		○	○	-	2.8
Wrong operation prevention		○	○	-	2.10
Retraction for rigid tapping		○	☆	J664	2.11
Retraction for 3-dimensional rigid tapping	Retraction for rigid tapping is required.	☆	☆	R575	2.12
Buffer register		○	○	-	2.13
Dry run		○	○	-	2.14
Single block		○	○	-	2.15
Jog feed		○	○	-	2.16
Manual reference position return		○	○	-	2.17
Manual 2nd/3rd/4th reference position return		☆	☆	R558	2.18

Item	Specifications	Seriess 0*i*-F		Drawing No.	Section No.
		M	T		
Reference position setting without DOG		○	○	-	2.19
Reference position setting with mechanical stopper		○	○	-	2.20
Reference point setting with mechanical stopper by Grid Method		☆	☆	S945	2.21
Reference position return speed set		○	○	-	2.22
Reference position shift		○	○	-	2.23
Manual handle feed	Max. 3 units	○	○	-	2.24.1
Manual handle feed 4/5-units	Max. 5 units	☆	☆	S858	2.24.3
Manual handle feed rate	×1, ×10, ×m , ×n m:0 to 2000, n :0 to 2000	○	○	-	2.24.2
3-dimensional manual feed		☆	-	S679	2.25
Manual handle interruption		○	○	-	2.26
Manual interruption of 3-dimensional coordinate system conversion	3-dimensional coordinate conversion is required	☆	☆	S949	2.26.1
FANUC SERVO MOTOR β series with I/O Link Manual handle interface		☆	☆	S722	2.27
Incremental feed	×1, ×10, ×100, ×1000, ×10000	○	○	-	2.28
Jog and handle simultaneous mode		○	○	-	2.29
Manual numerical command		☆	☆	J667	2.30
Reference position signal output		☆	☆	S629	2.31
Retrace		☆	-	J730	2.32
Manual handle retrace		☆	☆	J998	2.33
Manual handle retrace for multi path	(*3)	☆	☆	R606	2.35
Direction change movement in auxiliary function output block function	Manual handle retrace or manual handle retrace for multi path is required.	☆	☆	S628	2.34
Manual liner/circular interpolation	Only for 1path	☆	☆	J774	2.37
Handle-Synchronous Feed Function	Included in Manual liner/circular interpolation	*	*	-	2.38
Active block cancel		☆	☆	S627	2.39
High speed program check		☆	☆	S880	2.40
Dwell/Auxiliary function time override function		☆	☆	R500	2.41

Interpolation functions

Nano interpolation		○	○	-	3.1
Positioning	G00 (Linear interpolation type positioning is possible)	○	○	-	3.2
Single direction positioning	G60	○	-	-	3.3
Exact stop mode	G61	○	○	-	3.4
Tapping mode	G63	○	○	-	3.5
Cutting mode	G64	○	○	-	3.6
Exact stop	G09	○	○	-	3.7
Linear interpolation		○	○	-	3.9
Circular interpolation		○	○	-	3.10
Dwell	Dwell in seconds and dwell in revolution	○	○	-	3.11
Polar coordinate interpolation		-	○	-	3.12
Cylindrical interpolation		○	○	-	3.13.1
Cylindrical interpolation by plane distance command	Cylindrical interpolation is required.	☆	☆	R578	3.13.2
Helical interpolation	Circular interpolation plus max. 2 axes linear interpolation	○	☆	J819	3.14
Nano smoothing	AI contour control II is required.	☆	-	S687	3.15
Thread cutting, synchronous cutting		○	○	-	3.17

Item	Specifications	Seriess 0*i*-F		Drawing No.	Section No.
		M	T		
Multi threading		-	○	-	3.18
Thread cutting retract		-	○	-	3.19
Continuous threading		-	○	-	3.20
Variable lead thread cutting		-	○	-	3.21
Circular thread cutting		-	☆	J731	3.22
Polygon turning		-	○	-	3.23
Polygon machining with two spindles		-	○	-	3.24
Skip	G31	○	○	-	3.25.1
Multi-step skip		☆	☆	J849	3.25.2
High-speed skip	Input signal is 4 point	○	○	-	3.25.3
Torque limit skip		○	○	-	3.25.5
Reference position return	G28	○	○	-	3.26
Reference position return check	G27	○	○	-	3.26.2
2nd reference position return		○	○	-	3.26.3
3rd/4th reference position return		○	○	-	3.26.3
Normal direction control		○	-	-	3.27
Balanced cutting	Only for multi path control (*3)	-	☆	J834	3.28
Index table indexing		○	-	-	3.29
Continuous high-speed skip		☆	☆	J770	3.25.4
General purpose retract		○	○	-	3.30

Feed function

Rapid traverse rate (Increment system B)	Max. 999.999m/min (1μm)	○	○	-	-
Rapid traverse rate (Increment system C)	Max. 99.9999m/min (0.1μm)	○	○	-	-
Rapid traverse override	F0, 25, 50, 100% or 0-100% (1% Step)	○	○	-	4.2
Feed per minute		○	○	-	4.3
Feed per revolution		○	○	-	4.4
Without position coder feed per revolution		○	○	-	4.5
Without position coder constant surface speed control		○	○	-	4.6
Tangential speed constant control		○	○	-	4.7
Cutting feedrate clamp		○	○	-	4.8
Automatic acceleration/deceleration	Rapid traverse: linear Cutting feed: exponential, linear	○	○	-	4.9
Rapid traverse bell-shaped acceleration / deceleration		○	○	-	4.12
Optimum torque acceleration / deceleration		☆	☆	S675	4.13
Positioning by optimum acceleration		☆	☆	J693	4.14
Linear acceleration/deceleration after cutting feed interpolation		○	○	-	4.15
Bell-type acceleration/ deceleration after cutting feed interpolation		○	○	-	4.16
Linear acceleration/deceleration before cutting feed interpolation	Included in AI preview control, AI contour control I or II	○	*	-	4.17
Feedrate override	0-254%	○	○	-	4.18
2nd feedrate override	0-254%	☆	☆	J810	4.19
One-digit F code feed		○	-	-	4.20
Inverse time feed		○	-	-	4.21
Jog override	0-655.34%	○	○	-	4.22
Override cancel		○	○	-	4.23
Manual per revolution feed		-	○	-	4.24

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
		M	T		
External deceleration		○	○	-	4.25
Automatic corner deceleration	Included in AI preview control, AI contour control I or II	○	*	-	-
Feedrate control with acceleration in circular interpolation	Included in AI contour control I or II on T system	○	*	-	4.26
AI advanced preview control		○	-	-	4.27
AI contour control I		☆	☆	J665	4.27
AI contour control II		☆	☆	S808	4.27
Bell-type acceleration/deceleration before look ahead interpolation	Included in AI preview control, AI contour control I or II	○	*	-	4.28
Jerk control	AI contour control II is required.	☆	-	S678	4.29
Smart Tolerance Control	AI contour control II is required.	☆	-	R696	3.16
Rigid tapping bell-shaped acceleration/deceleration		○	○	-	4.30
Optimum acceleration/deceleration for rigid tapping		☆	☆	R533	4.31
Rapid traverse block overlap		○	○	-	4.10
Programmable rapid traverse overlap		☆	☆	R502	4.11
Error detection		-	○	-	-

Program input

Program code	EIA/ISO	○	○	-	5.1
Label skip		○	○	-	5.2
Parity check	Horizontal and vertical parity	○	○	-	5.3
Control in/out		○	○	-	5.4
Optional block skip	9	○	○	-	5.5,5.6
Max. programmable dimension	±9 digit	○	○	-	5.7
Program file name	32 characters	○	○	-	5.8
Sequence number	N8 digit	○	○	-	5.9
Absolute/incremental programming	Combined use in the same block	○	○	-	5.10
Decimal point programming/ pocket calculator type decimal point programming		○	○	-	5.11
Input unit 10 time multiply		○	○	-	5.12
Diameter/radius programming		○	○	-	5.13
Plane selection	G17, G18, G19	○	○	-	5.14
Rotary axis designation		○	○	-	5.15
Rotary axis roll-over		○	○	-	5.16
Polar coordinate command		○	-	-	5.17
Coordinate system setting		○	○	-	5.18
Automatic coordinate system setting		○	○	-	5.18.2.2
Workpiece coordinate system	G52-G59	○	○	-	5.18.2
Workpiece coordinate system preset		○	○	-	5.19
Addition of workpiece coordinate system	48 pairs	○	-	-	5.21
	300 pairs	☆	-	J919	5.21
Direct input of workpiece origin offset value measured		○	○	-	5.22
Positioning in machine coordinate system with feedrate		☆	☆	R553	-
Manual absolute on and off		○	○	-	5.23
Direct drawing dimension programming		-	○	-	5.24
G code system	A/B/C	-	○	-	5.25
Chamfering/corner R		-	○	-	5.26
Optional chamfering/corner R		○	-	-	5.27

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
		M	T		
Programmable data input	G10	○	○	-	5.28
Programmable parameter input		○	○	-	5.29
Sub program call	10 folds nested	○	○	-	5.30
Custom macro		○	○	-	5.31
Addition of custom macro common variables	#100 - #199, #500 - #999	○	○	-	5.32
Addition of custom macro common variables 1000	#100 - #199, #500 - #999, #98000 - #98499	☆	☆	R687	5.32
Custom macro common variables between each path	Only for multi path control	○	○	-	5.33
Interruption type custom macro	Only for multi path control (*3)	○	○	-	5.34
Canned cycles		-	○	-	5.35
Multiple repetitive cycle		-	○	-	5.36
Multiple repetitive cycle II	Pocket profile	-	○	-	5.36
Canned cycles for drilling		○	○	-	5.39
Circular interpolation by R programming	9 digit	○	○	-	5.40
Mirror image for double turret		-	○	-	5.41
Automatic corner override		○	-	-	5.42
Scaling		○	-	-	5.43
Coordinate system rotation		○	○	-	5.44
3-dimensional coordinate system conversion		☆	☆	J713	5.45
Tilted working plane indexing	Guidance screens is not shown on 8.4"LCD.	☆	-	R522	5.46
Programmable mirror image		○	○		5.47
Figure copying		☆	-	J897	5.49
G code preventing buffering		○	○	-	5.50
Program format for FANUC Series 10/11		○	○	-	5.51
Macro executor		☆	☆	J888	5.52
Macro executor + C language executor		☆	☆	J734	5.53
C language executor additional SRAM 256KB	192KB Non-volatile memory addition.	☆	☆	J736	5.54
C language executor additional SRAM 512KB	448KB Non-volatile memory addition.	☆	☆	S827	5.54
Custom software (Total amount of each path)	512KB	☆	☆	J738#512K	5.5.5
	2MB	☆	☆	J738#2M	
	4MB	☆	☆	J738#4M	
	6MB	☆	☆	J738#6M	
	8MB	☆	☆	J738#8M	
	12MB	☆	☆	J738#12M	
	16MB	☆	☆	J738#16M	
FANUC PICTURE executor	Custom software size 4M bytes or larger is required.	☆	☆	R644	-
FANUC PICTURE function	This function includes custom software size 6M bytes.	☆	☆	S879	
FANUC PICTURE function for non-touch panel display	This function includes custom software size 6M bytes.	☆	☆	S944	-
Coordinate system shift		-	○	-	5.56
Direct input of coordinate system shift		-	○	-	-
Embedded macro		☆	☆	S652#128K	5.57
Small-hole peck drilling cycle		○	-	-	5.58

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
		M	T		
Real time custom macro		☆	☆	S842	5.59
Pattern data input		○	○	-	5.60
M code protect function		☆	☆	R594	5.61
Conversational programming with graphic function		○	○	-	-
Guidance function					
MANUAL GUIDE i		☆	☆	S790	6.1
MANUAL GUIDE i multi-path function		☆	☆	S786	6.2
MANUAL GUIDE i Tilted Working Plane indexing function		☆	-	S788	-
MANUAL GUIDE 0i		☆	☆	S772	6.3
TURN MATE i	(*4)	-	☆	S792 S793	6.4
MDI key operation function		-	☆	S794	-
NC program conversion function		-	☆	S795	-
Expansion of machining cycle		-	☆	S796	-
Auxiliary/Spindle speed function					
Auxiliary function	M8 digit	○	○	-	7.1
2nd auxiliary function	B8 digit	○	○	-	7.2
Auxiliary function lock		○	○	-	7.3
High-speed M/S/T/B interface		○	○	-	7.4
Waiting function	Only for multi path control (*3)	○	○	-	7.5
Waiting M codes of high-speed type	Only for multi path control (*3)	○	○	-	7.6
Multiple command of auxiliary function	5 commands	○	○	-	7.7
Auxiliary function output in moving axis		☆	☆	S889	7.8
Waiting function by specifying start point	Only for multi path control (*3)	☆	☆	S888	7.9
Spindle speed function	S5 digit , binary output	○	○	-	7.10
Spindle serial output	S5 digit , serial output	○	○	-	7.11
Spindle analog output	S5 digit , analog output, up to 1 spindle (*6)	○	○	-	7.12
Constant surface speed control		○	○	-	7.13
Spindle override	0 - 254%	○	○	-	7.14
Actual spindle speed output		-	○	-	7.15
Spindle orientation	All spindles	○	○	-	7.16
Spindle output switching function	All spindles	○	○	-	7.17
Spindle synchronous control	Analog spindle is not available.	○	○	-	7.18
Spindle command synchronous control		☆	-	J748/J858	7.19
		-	☆	J858	
Multi spindle control		☆	○	J859	7.20
Spindle positioning		-	○	-	7.21
Rigid tapping		○	○	-	7.22
FSSB High speed rigid tapping	Analog spindle is not available.	○	○	-	-
Rigid tapping by manual handle		☆	☆	J651	7.23
Arbitrary position reference setting for CS axis		☆	☆	S664	7.24
M code group check		☆	☆	J922	7.25
Spindle speed fluctuation detection		-	○	-	7.26
Spindle control with servo motor		☆	☆	J978	7.28
Servo/spindle synchronous control		☆	☆	J858	7.30
Spindle tandem control	Analog spindle is not available.	☆	☆	J858	-
Arbitrary speed threading	Analog spindle is not available.	☆	☆	R672	7.34

Tool function/Tool compensation

Item	Specifications	Seriess 0*i*-F		Drawing No.	Section No.
		M	T		
Tool function	T7+1 / T6+2 / T5+3 (Tool selection + Tool offset number)	-	○	-	8.1
	T8 digit	○	-	-	
Tool offset pairs (Note) Specify total of tool offset pairs of each path.	128-pairs	-	○	-	8.3
	200-pairs	-	☆	J927	
	400-pairs	○	-	-	
Tool offset memory C	Distinction between geometry and wear, or between cutter and tool length compensation.	○	-	-	8.4
Common offset memory between each path	Only for multi path control (*3)	○	○	-	8.5
Tool length offset		○	-	-	8.6
Tool offset		○	○	-	8.7
Y-axis offset		-	○	-	8.8
4th/5th axis offset		-	☆	R517	-
Tool radius/Tool nose radius compensation		○	○	-	8.9
Cutting point interpolation for cylindrical interpolation		☆	☆	S674	8.10
Tool geometry/wear compensation		-	○	-	8.11
2nd Geometry Tool Offset		-	☆	J980	8.12
Tool management function: 64 pairs	64 tools (*7)	☆	☆	S830	8.13
Tool management function: 240 pairs	240 tools (*7)	☆	☆	S831	8.13
Tool management function: 1000 pairs	1000 tools (*7)	☆	☆	S833	8.13
Tool management function: Customized data expansion (5 to 20)		☆	☆	S834	8.13
Tool management function: Customized data expansion (5 to 40)		☆	☆	S835	8.13
Tool management expansion		☆	☆	S852	8.13.1
Tool management function for oversize tools	Included in Tool management expansion B.	*	*	-	8.13.2
Tool management function for multi-edge tools		☆	☆	R681	-
Tool management tool attachment / detachment function		☆	☆	S997	-
Tool management expansion B		☆	☆	R616	-
Tool offset value counter input		-	○	-	8.14
Tool length measurement		○	-	-	8.15
Automatic tool length measurement		○	-	-	8.16.1
Automatic tool offset		-	○	-	8.16.2
Direct input of tool offset value measured		-	○	-	8.17
Direct input of tool offset value measured B		-	○	-	8.18
Direct input of offset value measured B for 2 spindle lathe		-	☆	J686	-
Tool life management		○	○	-	8.19
Extended tool life management		○	○	-	-
Automatic alteration of tool position compensation		-	☆	J690	-
Tool geometry size data	100-pairs	☆	☆	R589	-
	300-pairs	☆	☆	R590	-

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Sectio n No.
		M	T		
Tool geometry size data - Additional tool type		☆	☆	R685	-

Accuracy compensation function

Backlash compensation		○	○	-	9.1
Backlash compensation for each rapid traverse and cutting feed		○	○	-	9.2
Smooth backlash compensation		○	○	-	9.3
Smart backlash compensation		○	○	-	-
Stored pitch error compensation		☆	☆	J841	9.4
Stored Pitch Error Compensation Total Value Input	Stored pitch error compensation is required	*	*	-	-
Interpolation type pitch error compensation	Stored pitch error compensation is required	☆	☆	S644	9.5
Bi-directional pitch error compensation	Stored pitch error compensation is required	☆	☆	S656	9.6
Extended bi-directional pitch error compensation	Stored pitch error compensation and Bi-directional pitch error compensation are required.	☆	☆	S657	9.7
Inclination compensation	Stored pitch error compensation is required	☆	☆	J981	9.8
Simple straightness compensation	Stored pitch error compensation is required. 1 pair	☆	☆	J799	9.10
Straightness compensation	Stored pitch error compensation is required. 4 pairs	☆	☆	J747	9.9
Interpolation type straightness compensation	128points. Stored pitch error compensation is required.	☆	☆	S639	9.11
Interpolated Straightness Compensation 3072 points	Stored pitch error compensation and Interpolation type straightness compensation are required	☆	☆	R638	-

Electronic gear box

Electronic gear box		☆	-	J779	10.1
Skip function for EGB axis	Electronic gear box is required.	☆	-	J696	10.2
Electronic gear box automatic phase synchronization	Electronic gear box is required.	☆	-	S711	10.3
Flexible synchronization control		☆	☆	S709	10.6.1
Automatic phase synchronization for Flexible synchronization control	Flexible synchronization control is required.	☆	☆	S611	10.6.2
Inter-Path Flexible synchronization control	Flexible synchronization control is required. (*3)	☆	☆	S610	10.6.3
Skip function for Flexible synchronization control	Flexible synchronization control is required.	☆	☆	S612	10.6.4
Hob command by Flexible synchronization control	Flexible synchronization control is required.	☆	☆	R847	10.6.5
U-Axis Control	Included in Electronic gear box.	*	-	-	10.7

Grinding function

Grinding function A	Multi-step skip, Canned cycles for grinding, Continuous dressing, Infeed control	☆	-	S682	3.24.2, 5.38, 5.37
	Multi-step skip, Canned cycles for grinding	-	☆		
Grinding function B	Angular axis control is available in addition to the functions included in Grinding function A.	☆	☆	S683	

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
		M	T		

Editing operation

Part program storage size (*2) (Note) Specify total of part program storage size of each path.	512Kbyte (Only for 1 path control) (*4)	○	○	-	11.1
	1Mbyte (Only for 2 path control) (*3)	○	○	-	
	2Mbyte	☆	☆	J948	
Number of registerable programs	400 (Only for 1 path control)	○	○	-	11.1
	800 (Only for 2 path control) (*3)	○	○	-	
Number of registerable programs expansion 1	Max. 1000 programs	☆	☆	J953	11.1
Part program editing		○	○	-	11.2
Extended part program editing		○	○	-	11.5
Program protect		○	○	-	11.3
Key and program encryption		☆	☆	J778	11.4
Password function		○	○	-	-
Playback		○	○	-	11.6
Machining time stamp		☆	☆	J964	11.7
Background editing		○	○	-	11.8
Multi part program editing	Not available on 8.4" display unit	○	○	-	-
Memory card program edit & operation	Max 63 programs. PC tool for memory card program operation/editing (A08B-9010-J700#ZZ11) is required to convert and store files to memory card.	○	○	-	11.9
Memory card program entry count extension	Max. 1000 programs	☆	☆	S995	11.9
Data server editing/operation	Fast data server is required.	*	*	-	-
Multi-path editing function	(*3)	☆	☆	R615	11.10
High speed program management		○	○	-	-

Setting and display

Status display		○	○	-	12.1
Clock function		○	○	-	12.2
Current position display		○	○	-	12.3
Program comment display	Program name 31 characters	○	○	-	-
Parameter setting and display		○	○	-	12.5
Parameter check sum function		○	○	-	-
Alarm display		○	○	-	12.6
Alarm history display		○	○	-	12.7
Operator message history display		○	○	-	12.8
Operation history display		○	○	-	12.9
Remote diagnostic	Included in Machine remote diagnosis package	*	*	-	-
Run hour and parts count display		○	○	-	12.10
Actual cutting feedrate display		○	○	-	12.11
Display of spindle speed and T code at all screens		○	○	-	12.12
Directory display of floppy cassette		○	○	-	12.13
Optional path name display	Only for multi path control (*3)	○	○	-	12.14
Operating monitor screen		○	○	-	12.15
Servo setting screen		○	○	-	12.16
Spindle setting screen		○	○	-	12.17
Servo waveform display		○	○	-	12.18
Maintenance information screen		○	○	-	12.19
Trouble diagnosis		○	○	-	12.20

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
		M	T		
Machine alarm diagnosis	Guidance table for Machine alarm diagnosis that is included in CNC Application Development Kit (A08B-9010-J555#ZZ12) is necessary for making guidance messages.	☆	☆	S813	12.21
Software operator's panel		○	○	-	12.22
Software operator's panel general purpose switch		○	○	-	12.23
Extended software operator's panel general purpose switch		○	○	-	-
Machine operation menu	Machine operation menu making tool that is included in CNC Application Development Kit (A08B-9010-J555#ZZ12) is necessary for making menu data of machine operation menu.	☆	☆	S844	12.24
FANUC Auto HMI-NC	Either FANUC PICTURE function, FANUC PICTURE function for non-touch panel display or FANUC PICTURE Executor is necessary.	☆	☆	R572	-
FANUC Auto HMI-NC screen enhancement 1	FANUC Auto HMI-NC is necessary.	☆	☆	R653	-
Multi-language display	English	○	○	-	12.25
	Japanese (Chinese character)	○	○	-	
	German	○	○	-	
	French	○	○	-	
	Spanish	○	○	-	
	Italian	○	○	-	
	Chinese (Traditional Chinese)	○	○	-	
	Chinese (Simplified Chinese)	○	○	-	
	Korean	○	○	-	
	Portuguese	○	○	-	
	Dutch	○	○	-	
	Danish	○	○	-	
	Swedish	○	○	-	
	Hungarian	○	○	-	
	Czech	○	○	-	
	Polish	○	○	-	
	Russian	○	○	-	
	Turkish	○	○	-	
	Romanian	○	○	-	
	Bulgarian	○	○	-	
	Slovak	○	○	-	
	Finnish	○	○	-	
	Hindi	○	○	-	
Dynamic display language switching		○	○	-	12.25
Data protection key	4 types	○	○	-	12.26
Protection of data at eight levels		☆	☆	S828	12.27
Warning function against modification of setting		☆	☆	R670	12.28
Erase CRT screen display (*1)	Manual or Automatic	○	○	-	12.29
Parameter setting support screen		○	○	-	12.30
Machining condition selecting function	For T series, AI contour control I or II is required.	☆	☆	S637	12.31

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
		M	T		
Machining quality level adjustment function	AI contour control II, Nano smoothing and Machining condition selecting function are required.	☆	-	R593	12.32
Help function		○	○	-	12.34
Self-diagnosis function		○	○	-	12.35
Periodic maintenance screen		○	○	-	12.36
Display of hardware and software configuration		○	○	-	12.33
Servo information screen		○	○	-	12.37
Spindle information screen		○	○	-	12.37
Graphic display		○	○	-	12.38
Dynamic graphic display		☆	☆	J760	12.39
Touch panel control		☆	☆	J682	12.40
External touch panel interface		☆	☆	J685	12.41
Virtual MDI key		☆	☆	S883	-
CNC screen display	CNC Application Development Kit (A08B-9010-J555 #ZZ12) is necessary.	○	○	-	-
Dual screen of CNC screen display function		☆	☆	S884	-
Basic operation package 2 function	CNC Application Development Kit (A08B-9010-J555#ZZ12) is necessary.	☆	☆	0207-J816	-
Machining status monitor package function	CNC Application Development Kit (A08B-9010-J555#ZZ12) is necessary.	☆	☆	0207-J870	-
CNC screen Web server function		☆	☆	R728	14.1
Power consumption monitoring		○	○	-	12.45
Energy Saving Level Selecting Function		☆	☆	R719	12.46
Machine State Monitoring Function		☆	☆	R717	12.47
Main menu screen		○	○	-	-
Main menu screen customizing function	Main menu screen customization tool that is included in CNC Application Development Kit (A08B-9010-J555#ZZ12) is necessary.	☆	☆	R848	-
Data input/output					
RS232C interface	RS232C (Ch.1) interface	○	○	-	13.1
	RS232C (Ch.2) interface	○	○	-	
Fast data server	DNC operation is available for 1st path control only/Option board is required	☆	☆	S737	13.2
Data server buffer mode	included in Fast data server	*	*	-	13.3
Data server explorer connection	Fast data server is required	☆	☆	R953	13.4
External tool offset		○	○	-	13.5.1
External machine zero point shift		○	○	-	13.5.4
External message		○	○	-	13.5.6
External data input	Including External message, External tool offset, and External machine zero point shift.	○	○	-	13.5
External key input		○	○	-	13.6
External workpiece number search	9999	○	○	-	13.7
External program number search	1 - 9999	○	○	-	13.5.2
Memory card input/output		○	○	-	13.8
USB memory input/output	LCD mounted type Control unit with USB interface is required.	○	○	-	13.9
Screen hard copy		○	○	-	13.10

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
		M	T		
Power Mate CNC manager		○	○	-	13.11
External I/O device control		○	○	-	13.12
One touch macro call		☆	☆	S655	13.13
Automatic data backup		○	○	-	13.14

Interface function

Automatic data backup		○	○	-	14.1
Fast Ethernet	Option board is required.	☆	☆	S707	14.2
PROFIBUS master function	Option board is required.	☆	☆	S731	14.3
PROFIBUS slave function	Option board is required.	☆	☆	S732	14.3
DeviceNet master function	Option board is required.	☆	☆	S723	14.3
DeviceNet slave function	Option board is required.	☆	☆	S724	14.3
FL-net function	Option board is required.	☆	☆	J692	14.4
Safety function by FL-net	FL-net function and Dual check safety are required.	☆	☆	S851	14.4
FL-net/Ethernet coexisting function	Fast Ethernet and FL-net function are required.	*	*	-	14.4
Enhanced Embedded Ethernet function		○	○	-	
CC-Link Remote Device function	Option board is required.	☆	☆	R954	14.3
Robot connection function		☆	☆	R683	12.44
EtherNet/IP Scanner function	Option board is required.	☆	☆	R966	14.4
EtherNet/IP Adapter function	Option board is required.	☆	☆	R967	14.4
EtherNet/IP Adapter Safety function	EtherNet/IP Adapter function and Dual check safety are required.	☆	☆	R976	14.4
Modbus/TCP Server function		☆	☆	R968	14.4
'PROFINET IO Controller function	Option board is required.	☆	☆	R971	14.4
PROFINET IO Device function	Option board is required.	☆	☆	R972	14.4
CNC Status Notification function		☆	☆	R975	14.1

PMC

PMC function	24000 steps	○	○	-	15.1, 15.3
	32000 steps	☆	☆	H990#32K	
	64000 steps	☆	☆	H990#64K	
	100000 steps	☆	☆	H990#100K	
Ladder Dividing Management Function		○	○	-	-
I/O Link i DI/DO points	DI/DO: 2048/2048 points	○	○	-	15.7
1st level execution cycle of ladder	4ms	○	○	-	15.1
Multi-path PMC function	3 paths	☆	☆	R855#3	15.2
PMC symbol, comment and message area expansion	512Kbyte	☆	☆	R856#512K	15.6
	1Mbyte	☆	☆	R856#1M	
PMC multi-language message display function		○	○	-	15.5
Multi-language display of signal comment		○	○	-	-
Step sequence	Only 1st PMC path	☆	☆	S982	15.1
Nonvolatile PMC extra relay function		☆	☆	S984#10K	15.8

Item	Specifications	Seriess 0*i*-F		Drawing No.	Section No.
		M	T		
Nonvolatile PMC data table area expansion (40KB)	40Kbyte	☆	☆	S967# 40K	15.4
Extended PMC ladder instruction function		○	○	-	15.1
PMC Function block function		○	○	-	15.9

Others

Status output signal	NC ready, servo ready, automatic operation, automatic operation start lamp, feed hold, reset, NC alarm, distribution end, rewinding, inch input, cutting, inposition, thread cutting, tapping, etc.		○	○	-	16.1
LCD mounted type Control unit (*1)	8.4" color LCD/MDI Horizontal type	0 slot	●	●	-	-
		2 slots	●	●	-	-
	8.4" color LCD/MDI Horizontal type (with touch panel)	0 slot	●	●	-	-
		2 slots	●	●	-	-
	8.4" color LCD/MDI Vertical type	0 slot	●	●	-	-
		2 slots	●	●	-	-
	8.4" color LCD/MDI Vertical type (with touch panel)	0 slot	●	●	-	-
		2 slots	●	●	-	-
	10.4" color LCD	0 slot	●	●	-	-
		2 slots	●	●	-	-
	10.4" color LCD (with touch panel)	0 slot	●	●	-	-
		2 slots	●	●	-	-
MDI unit	Separate MDI (ONG small , horizontal type for 10.4")		●	●	-	-
	Separate MDI (ONG horizontal type/ONG vertical type for 10.4")		●	●	-	-
	Separate MDI (QWERTY Type A for 10.4") (width 290mm)		●	●	-	-
	Separate MDI (QWERTY Type B for 15") (width 400mm)		●	●	-	-
Machine interface (I/O Link *i*)	I/O unit for power magnetics cabinet DI/DO: 96/64 (with MPG I/F)		●	●	-	-
	I/O module for power magnetics cabinet (without MPG I/F)		●	●	-	-
	Operator's panel I/O module (with MPG I/F)		●	●	-	-
	Operator's panel I/O module (without MPG I/F)		●	●	-	-
	Standard operator's panel		●	●	-	-
	Small operator's panel (Without General DI/DO)		●	●	-	-
	Small operator's panel B (General DI/DO:24/16 points)		●	●	-	-
	Terminal type I/O module		●	●	-	-
	I/O Unit-MODEL A		●	●	-	-
	I/O Unit-MODEL B		●	●	-	-
	Additional peripheral axis (I/O Link β i servo)		●	●	-	-
FANUC I/O Link - AS-i converter			☆	☆	-	-
Manual pulse generator			☆	☆	-	-

Item	Specifications	Seriess 0*i*-F		Draw-ing No.	Section No.
		M	T		
Pendant type manual pulse generator	With axis selection and magnification switches	☆	☆	-	-
Handy machine operator’s panel		☆	☆	-	-
i Pendant		☆	☆	-	-
Connectable servo motor	FANUC AC SERVO MOTOR α*i* series	●	●	-	-
	FANUC AC SERVO MOTOR β*i* series	●	●	-	-
Connectable servo amplifier	FANUC SERVO AMPLIFIER α*i* series	●	●	-	-
	FANUC SERVO AMPLIFIER β*i* series	●	●	-	-
	Analog spindle interface (*5)	●	●	-	-
Separate detector interface unit (for full-closed control) (*8)	Linear / rotary encoder (A/B phase digital interface)	☆	☆	-	-
	Separate Pulsecoder, Linear/rotary encoder (serial interface)	☆	☆	-	-
	Linear/rotary encoder （Analog 1Vp-p interface）	☆	☆	-	-
Connectable spindle motor	FANUC AC SPINDLE MOTOR α*i* series	●	●	-	-
	FANUC AC SPINDLE MOTOR β*i* series	●	●	-	-
Analog servo adapter	For retro fitting only (*5)	●	●	-	-
SERVO GUIDE		☆	☆	-	-
Input power supply	24VDC±10%	○	○	-	-
Ambient temperature of unit	LCD mounted type control unit, display unit for stand-alone type control unit At operating: 0-58°C At nonoperating: -20 - 60°C	●	●	-	-
Ambient relative humidity	Normally: 75%RH or less (No dew, nor frost allowed) Short term (within one month): 95%RH or less(No dew, nor frost allowed)	○	○	-	-
Vibration	IEC 60068-2-6 conforming	○	○	-	-

- **Software of personal computer part in case of the CNC system which is FANUC PANEL *i* or connected with personal computer via HSSB(High Speed Serial Bus)**

Items	Specifications	Remarks
Operating system	(1) Windows® Embedded Standard 2009 (*11) or (2) Windows® Embedded Standard 7	(1) Equivalent to Windows® XP Professional 32-bit Edition (2) Equivalent to Windows® 7 Professional 64-bit Edition (*11)
Extended library	FOCAS2	Option
Software packages	Basic operation package 2	Option
	CNC screen display function	Option
	Ladder editing package	Option
Development tools	Visual Studio® 2008 (*11) Visual Studio® 2010	Microsoft Corp.
	.NET class library for 19" LCD	Option

- **Hardware of HSSB(High Speed Serial Bus) and Required hardware of commercially available personal computer in case of the CNC system which is connected with the personal computer via HSSB(High Speed Serial Bus).**

Items	Specifications	Remarks
CNC side interface	Option board	
Personal computer side interface board	PCI Bus and HSSB for 1 channel PCI Bus and HSSB for 2 channel	For PCI slot in the personal computer Using voltage: +5V only
Connecting cable	Optical fiber cable	Max. length: 100m
Personal computer requirements	CPU: Pentium® or more (*11) PCI slot 1 or more	For environmental requirements of the personal computer, refer to the manual supplied with the machine

- **Software of Simulator for PC**

ITEMS		SPECIFICATIONS	REMARKS
NCGuide	1 user	A08B-9010-J770#ZZ12	Including a DVD and Hardware key (DVD only for "Update") Corresponding OS is as follows. Windows® Vista Business(SP2) (32bit) Windows® 7 Professional (32bit,64bit) Windows® 8, 8.1 Pro (32bit,64bit) (*11)
	10 users	A08B-9010-J771#ZZ12	
	20 users	A08B-9010-J772#ZZ12	
	Site license	A08B-9010-J773#ZZ12	
	Update	A08B-9010-J774#ZZ12	
NCGuide Academic package	For Classroom / 16 users	A08B-9010-J751#ZZ12	
	For Homework / 1 year	A08B-9010-J752#ZZ12	

NOTE

*1 : The control unit is incorporated with display unit.
LCD is manufactured by using high precision technology, however it has points which are always bright or dark.
This phenomenon is caused by LCD's structure, and not defects.

*2 : The part program storage size is a value of "Maximum program size when one program is registered".
The total value of the program size that can be registered decreases when two or more programs are registered.
(The actual registrable value might changes according to the registered number of programs and the program sizes.)

*3 : Only for 2 path control

*4 : Only for 1 path control

*5 : Dual check safety is not available.

*6 : In case of using the serial spindle together, only spindle speed command control and spindle speed command control by PMC can be used, because position coder for analog spindle can not be used. Series 0i Mate-D can not use an analog spindle together with a serial spindle.

*7 : This function includes "Tool management tool attachment/detachment function".

*8 : In case of Servo HRV3 control, the number of controllable position detectors is up to 3, when the number of connected servo motors are up to 5, and the number of controllable detectors is up to 6, when the connected servo motors are from 6 to 10.
In case of Servo HRV2 control, the number of controllable position detectors is up to 8.

*9 : The number of connectable servo motors is up to 9 when Loader control function is not available.

*10 : The number of connectable servo motors is up to 10 in servo HRV3.
The number of connectable servo motors is up to 12 in servo HRV2.

*11 : Intel, Pentium are registered trademarks of Intel Corporation.
MIcrosoft, Windows, and VIsual Studio are registered trademarks of Microsoft Corporation.
Each company's name and product's name is the trademark or registered trademark.

II. NC FUNCTION

1 CONTROLLED AXIS

Chapter 1, "CONTROLLED AXIS", consists of the following sections:

1.1 MAXIMUM TOTAL NUMBER OF CONTROLLED AXES

“Maximum total number of control axes” is sum of “Number of feed axes* and “Number of spindle axes”. The number of maximum controlled axes that can be used, which differs depending on the model and the option configuration, is as given in the table below.

	Series 0*i*-F	
	1-path system	**2-path system**
Maximum total number of control axes	8 axes	11 axes

NOTE

The maximum number of controlled axes that can be used is limited depending on the option configuration.

1.2 NUMBER OF MACHINE GROUPS

If multiple paths are used, several paths can be formed into a group. By doing so, the group can share data, and if an alarm is issued with a path, the other path(s) in the group can be stopped. A group of those paths is referred to as a machine group.
Up to three groups can be used, depending on the type of NC system.
Mainly, the following depend on the machine group:

- Emergency stop signal
- RESET on the MDI
- Operation performed when an alarm is issued

The number of maximum machine groups that can be used, which differs depending on the model and the option configuration, is as given in the table below.

	Series 0*i*-F
Machine groups	3

1.3 NUMBER OF CONTROLLED PATHS

A path represents a group of axes that are controlled by the same NC program.
Up to 10 paths can be used, depending on the type of NC system.
(A path for loader control is included as a path.)
Which machine group the local path must belong to is determined by parameter setting.
The number of maximum controlled paths that can be used, which differs depending on the model and the option configuration, is as given in the table below.

	Series 0*i*-F
Control paths for machining	2
Control paths for loader function	2

1.3.1 Multi-path Control

The multi-path control function, which uses multiple paths, is designed to perform up to ten machining operations independently at the same time. This function is applicable to lathes and automatic lathes which perform cutting simultaneously with multiple tool posts, and machines which require additional control paths such as a loader control path.
Available functions specific to multi-path control include waiting function between each path, interference check for each path, balance cut, synchronous control, composite control, spindle control between each path, and common memory between each path.
A multi-path control system consists of machine groups, controlled paths, and controlled axes. Each component can be set by parameters according to the desired machine configuration.

NOTE
"Multi-path (2-path) control" is an optional function.

1.3.2 Function for Loader Control

The function for loader control is used to control the devices for performing a non-machining operation (peripheral device such as a loader). When this function is valid, the path for performing a loader control is added besides a path for machining. (The added path is called the loader path.)

NOTE
This function is an optional function. The number of the usable loader path is different depending on the option composition.

1.4 NUMBER OF CONTROLLED AXES / NUMBER OF CONTROLLED SPINDLE AXES

The number of controlled axes and controlled spindle axes depends on the model, as shown below.

Item		Series 0*i*-MF		Series 0*i*-TF	
		1-path system	2-path system	1-path system	2-path system
Maximum total number of control axes (total/each path)		8 axes	11 axes/ 9 axes	8 axes	11 axes/ 9 axes
Maximum number of feed axes[*1] (total/each path)	Basic	5 axes	8 axes/5 axes	4 axes	8 axes/5 axes
	Controllable axes expansion	7 axes	9 axes/7 axes	7 axes	9 axes/7 axes
Maximum number of spindle axes (total/each path)	Basic	2 axes	2 axes/2 axes	2 axes	2 axes/2 axes
	Spindle axes expansion	-	4 axes/3 axes	3 axes	4 axes/3 axes
Simultaneously controlled axes(each path)		4 axes in the time	4 axes in the time	4 axes in the time	4 axes in the time
PMC axis control[*2](total)		4 axes int the time	8 axes in the time	4 axes int the time	8 axes in the time
Number of connected servo motors[*3] (HRV3 control/HRV2 control)		10/12	10/12	10/12	10/12

*1: With PMC Axis. Without “Cs Contour Control” and “Spindle Control with Servo Motor”.
*2: Without “Cs Contour Control”

*3: Different according to the main-board.
Total of all path including loader path.
The following are included.
- Servo axis
- Spindle Control with Servo Motor
- EGB dummy axis
- Serial feedback dummy axis

NOTE
1 “Controllable axes expansion” and “Spindle axes expansion” are optional function.
2 The number of connected servo motors is different according to the main-board.

Whether it is counted by feed-axes/spindle-axes is decided depending on the kind of each axis/spindle. How to count axis/spindle is as shown below.

Item		Number of Feed Axes	Number of Spindle Axes	Total Number of Control Axes (Number of Feed Axes + Number of Spindle Axes)	Number of Connected Servo Motors
Servo Axis(including PMC Axis)		1	0	1	1
Analog Spindle[*1]		0	1	1	0
Serial Spindle		0	1	1	0
Cs Contour Control	Cs Axis	0	1	1	0
	Virtual Cs Axis[*2*3]	0	0	0	0
Axis that is controlled by “Spindle Control with Servo Motor”		0	1	1	1
EGB Dummy Axis[*3]		0	0	0	1[*4]
Serial Feedback Dummy Axis[*3]		0	0	0	1[*4]

*1: Up to 1 axis in the total.
*2: Up to 2 axes in the total. Up to 1 axes in each path.
*3: The total of the following numbers of axes is up to 3 axes in the total.
- Virtual Cs axis
- EGB dummy axis
- Serial feedback dummy axis
*4: It is counted by the number of connected servo motors though the motor is not connected.

Loader Path

When the function for loader control is used, the loader path can be used.
In the loader path, the servo axis can be used up to 3 axes. Other axis/spindle cannot be used.
The axis in the loader path is contained in the number of connected servo motors.
The axis in the loader path is not contained in the “total number of control axes” and “number of feed axes”.

NOTE
Function for loader control is optional function. The number of the usable loader path is different depending on the option composition.

1.5 PMC AXIS CONTROL

The PMC can directly control any given axis, independent of the CNC. By specifying an amount of travel, feedrate, and so forth from the PMC, a movement can be made along an axis independently of other axes operated under CNC control. This enables the control of turrets, pallets, index tables and other peripheral devices using any given axes of the CNC.

The following operations can be directly controlled from the PMC:

(1) Rapid traverse with a travel distance specified
(2) Cutting feed with a travel distance specified : Feed per minute
(3) Cutting feed with a travel distance specified : Feed per revolution
(4) Cutting feed with a travel distance specified : Feed per minute
(5) Dwell
(6) Continuous feed
(7) Reference position return
(8) 1st to 4th reference position return
(9) External pulse synchronization - Position coder
(10) External pulse synchronization - 1st to 3rd manual handle
(11) Feedrate control
(12) Torque control command
(13) Auxiliary function, Auxiliary function 2, Auxiliary function 3
(14) Selection of the machine coordinate system

1.6 Cs CONTOURING CONTROL

The Cs contouring control function positions the serial spindle using the spindle motor in conjunction with a dedicated detector mounted on the spindle.
The Cs contouring control function is higher in precision than spindle positioning, and enables positioning with other servo axes. Namely, the Cs contouring control function enables linear interpolation between the spindle and servo axes.

The speed of the serial spindle is controlled by the spindle speed control function, while the spindle positioning is controlled by the Cs contouring control function ("spindle contouring control"). Spindle speed control rotates the spindle using the velocity command, while the spindle contour control rotates the spindle using the move command.
Switching between spindle speed control and Cs contouring control is performed by the signal from the PMC.
In the Cs contouring control mode, the Cs contouring control axis can be operated either manually or automatically, in the same way as normal servo axes.

1.7 NAMES OF AXES

1.7.1 Names of Axes

Axis names can be assigned to axes controlled by the CNC (including PMC controlled axes). An axis name can be freely selected from 'A', 'B', 'C', 'U', 'V', 'W', 'X', 'Y', and 'Z'.

NOTE

1 The same name must not be set for multiple axes. (However, an axis name used in a path may be used in another path.)

NOTE

2 When G code system A is used with a lathe system, U, V, and W must not be used as axis names. Only when G code system B or C is used, U, V, and W can be used as axis names.

1.7.2 Axis Name Expansion

The axis name expansion function enables an axis name to be extended by up to three characters.
In order to extend an axis name:

(1) Enable the parameter for the axis name expansion function.
(2) Set the first character ('A', 'B', 'C', 'U', 'V', 'W', 'X', 'Y', 'Z') in the first axis name parameter.
(3) Set the second character ('0' to '9', 'A' to 'Z') in the second axis name parameter.
(4) Set the third character ('0' to '9', 'A' to 'Z') in the third axis name parameter.

NOTE

1 If the second axis name is not set for an axis, the specification of the third axis name is invalid.
2 If a character from '0' to '9' is set as the second axis name, do not set a character from 'A' to 'Z' as the third axis name.
3 If an axis name ends with a number, '=' is required between the axis name and a command value.
4 In a macro call, no axis name expansion can be used as an argument.
5 If at least one axis in a path uses an extended axis name when the parameter is invalid, subscripts cannot be used for axis names in the path.
6 When G code system A is used for a lathe system, X, Y, Z, or C may be used for the first axis name character of an axis. In this case, when a command containing U, V, W, or H as the first axis name character is specified, it is used as the incremental command for the corresponding axis.
7 In a multi-path system, if an extended axis name is not used on a path or if the parameter is valid and subscripts are not set for axis names, the path name will automatically be the subscript for axis names. To disable the display of axis name subscripts, set a blank (32) of ASCII code in the parameter for specifying an axis name subscript.

The usable names and their allowed combinations are indicated below.

	First axis name character	**Second axis name character**	**Third axis name character**
Setting	A, B, C, U, V, W, X, Y, Z	0 to 9	0 to 9
		A to Z	0 to 9
			A to Z
Correct example <1>	X	1	1
Correct example <2>	X	A	1
Correct example <3>	X	A	B
Incorrect example	X	1	A

1.8 ARBITRARY AXIS NAME SETTING

When the custom macro function is enabled, an indirect command based on an axis number can be specified for an axis address by using AX[(Axis number)], instead of direct axis name specification.
By using AXNUM[(Axis name)], the axis number of an axis name can also be obtained.

1.8.1 Arbitrary Axis Name

By using arbitrary axis name AX[ ], a command for an axis can be specified with an axis number. (AX[ ] must always be followed by '='.)

Format

AX[(Axis number)] = (Numerical value) ;

(Axis number) : 1 to number of controlled axes
(number of controlled axes of each path in the case of a multi-path system)
(Numerical value) : Command value for the axis specified by an axis number

Explanation

If an invalid (Axis number) is specified, an alarm is issued. If a specified axis number has fractional digits, a value rounded off to an integer is used as (Axis number).
As (Axis number), a variable (local variable, common variable, or system variable) can also be specified. When an operation using a variable name as (Axis number) is performed, however, the variable name must be enclosed in brackets ([ ]).
Example)

1. AX[1]=100.0;
   For the first axis, 100.000 is specified.
2. AX[#500]=200.0;
   For the axis with the axis number stored in #500, 200.000 is specified.
3. AX[#500+1]=300.0;
   For the axis with the axis number obtained by adding 1 to the value stored in #500, 300.000 is specified.
4. SETVN 500 [ABC];
   AX[#ABC]=400.0;
   For the axis with the axis number stored in #ABC(#500), 400.000 is specified.
5. SETVN 500 [ABC];
   AX[[#ABC]+1]=500.0;
   For the axis with the axis number obtained by adding 1 to the value stored in #ABC(#500), 500.000 is specified.
6. SETVN 500 [ABC];
   AX[#ABC+1]=500.0;
   An alarm is issued.

1.8.2 AXNUM Function

By using AXNUM[ ], an axis number can be obtained.

Format

AXNUM[(Axis name)];

Explanation

If an invalid (Axis name) is specified, an alarm is issued.

Example)

Suppose that there are three controlled axes and that the first axis name is "X", the second axis name is "Y", and the third axis name is "Z".

1. #500=AXNUM[X];
   In #500, 1 is stored.
2. #501=AXNUM[Y];
   In #501, 2 is stored.
3. #502=AXNUM[Z];
   In #502, 3 is stored.
4. #503=AXNUM[A];
   An alarm occurs.

Example

Sample program where the first axis name is "X", the second axis name is "Y", and the third axis name is "Z1"

```
N10  SETVN 500[AXIS1,AXIS2,AXIS3] ;
N20  [#AXIS1]=AXNUM[X] ;
N30  [#AXIS2]=AXNUM[Y] ;
N40  [#AXIS3]=AXNUM[Z1] ;
N50  G92 AX[#AXIS1]=0 AX[#AXIS2]=0 AX[#AXIS3]=0 ;
N60  G01 F1000.0 ;
N70  AX[#AXIS1]=100.0 AX[#AXIS2]=100.0 AX[#AXIS3]=100.0 ;
N80  G02 AX[#AXIS1]=200. 0 AX[#AXIS1]=200.0 R50.0 ;
N90  M02 ;
```

1.9 SPINDLE NAME EXPANSION

A spindle name can be extended by up to three characters starting with 'S' as the first spindle name. With this function, a command can be specified for each spindle without specifying a P command.
As the second and third spindle names, characters '0' to '9' and 'A' to 'Z' in ASCII code can be freely set. If the second spindle name is not set for a spindle, however, the third spindle name is invalid. If a character from '0' to '9' is set as the second spindle name, do not set a character from 'A' to 'Z' as the third spindle name.
If a spindle name ends with a number, '=' is required between the spindle name and a command value.

The usable names and their allowed combinations are indicated below.

	First spindle name (fixed)	**Second spindle name**	**Third spindle name**
Setting	**S**	**0 to 9**	**0 to 9**
		A to Z	**0 to 9**
			A to Z
Correct example <1>	S	1	1
Correct example <2>	S	A	1
Correct example <3>	S	A	B
Incorrect example	S	1	A

In multi-path control, an extended spindle name is common to all paths. This means that if the first spindle of path 2 is named "SA", and the following is specified for path 1:

SA1000;

1000 is specified for the first spindle of path 2. So, the same expanded spindle name cannot be used with a different path.

1.10 PERIPHERAL AXIS CONTROL

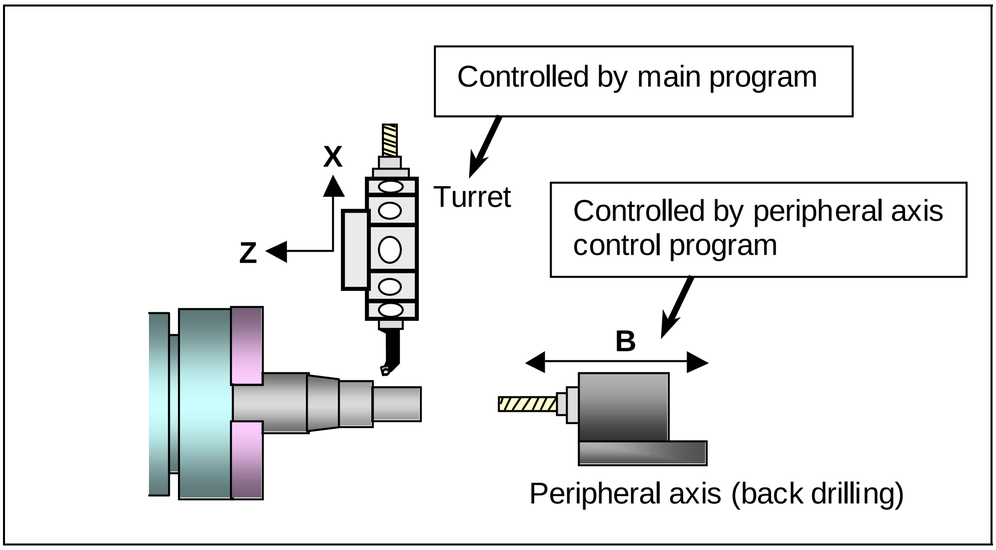
Overview

In addition to a main program, another program can execute during automatic operation. Therefore, peripheral axes such as loader and back drilling can be controlled in parallel with a main program.

Peripheral axis control has three control groups; “peripheral axis control group 1”, “peripheral axis control group 2”, and “peripheral axis control group 3”. Each group can run independently. Therefore, in a multi-path system, peripheral axis control can be applied to each path independently by assigning different control groups for each path.

NOTE

This function is an optional function.



1.11 SYNCHRONOUS / COMPOSITE CONTROL

In multi-path control, movements are usually made on the axes of a path according to a move command for the path (independent control in each path). However, the synchronous/composite control function enables an arbitrary axis of one path to be synchronized with an arbitrary axis of another path (synchronous control).

Moreover, a move command for an arbitrary axis of one path and a move command for an arbitrary axis of another path can be exchanged with each other to make a movement on each axis (composite control).

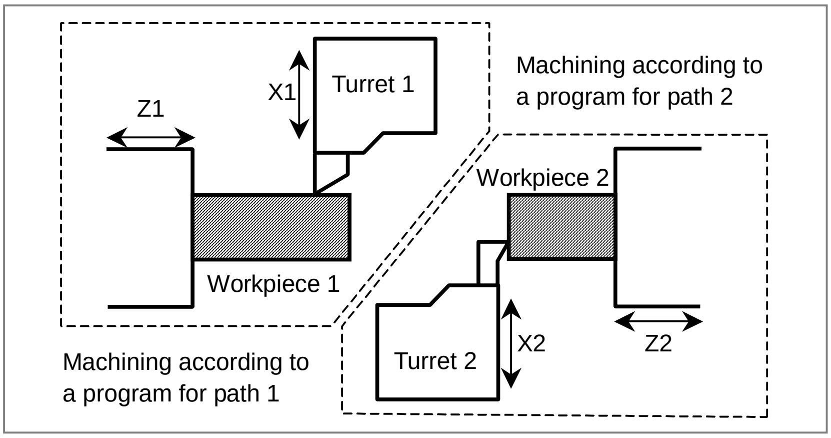
NOTE

This function is an optional function.

Explanation

- Independent control in each path

Movements on the axes (X1, Z1, and so on) of path 1 are made according to a move command for path 1, and movements on the axes (X2, Z2, and so on) of path 2 are made according to a move command for path 2.



- Synchronous control

By applying a move command for an axis (master axis) to a different arbitrary axis (slave axis), the movements on the two axes can be synchronized with each other. Whether to synchronize the movement on a slave axis with the move command for the master axis or make a movement on a slave according to the command for the slave can be chosen using the signal from the PMC.

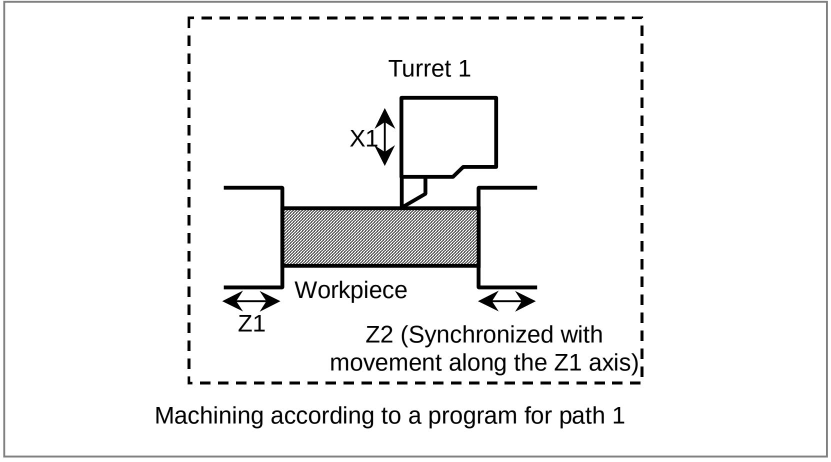
⚠ **CAUTION**

1 Synchronization mentioned above means that a move command for the master axis is also specified for a slave axis at the same time. Synchronization loss compensation, which detects the positional deviation between the master axis and slave axis and compensates for the deviation, is not performed. However, the positional deviation is detected at all times, and if the positional deviation exceeds a certain parameter-set value, the movement on each axis is stopped with an alarm.

2 The master axis and slave axis may belong to the same path, or the master axis may belong to one axis and the slave axis may belong to another. Moreover, multiple slave axes can be specified for one master axis.

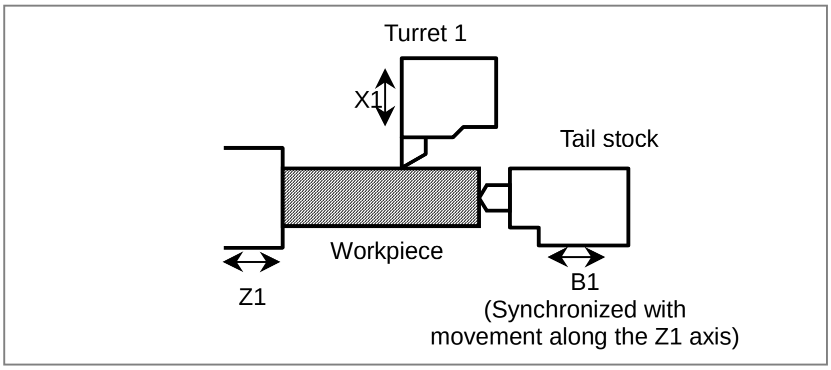
Example 1)

The Z2 axis of path 2 is synchronized with the Z1 axis of path 1.



Example 2)

The B1 axis of path 1 is synchronized with the Z1 axis of path 1.



- Composite control

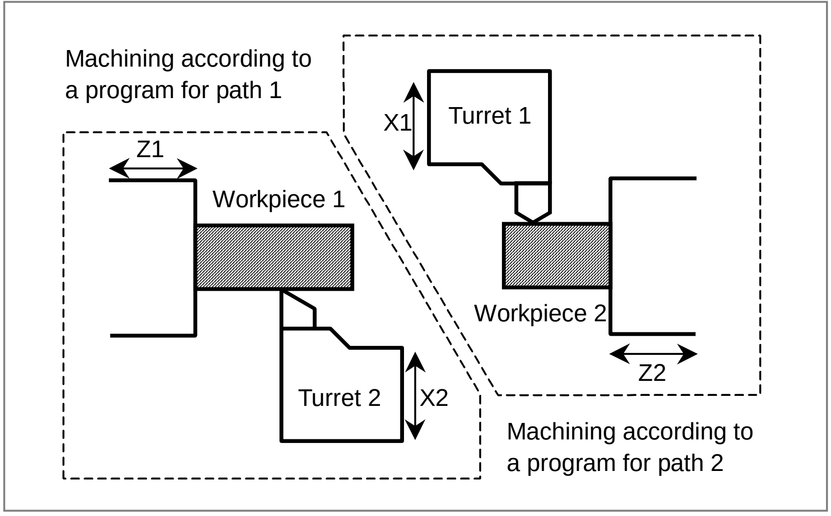
A move command for an arbitrary axis of one path and a move command for an arbitrary axis of another path can be exchanged with each other to make a movement on each axis.

Example)

A move command for the X1 axis of path 1 and a command for the X2 axis of path 2 are exchanged with each other.

The program for path 1 makes movements along the X2 axis and Z1 axis.

The program for path 2 makes movements along the X1 axis and Z2 axis.



1.12 SUPERIMPOSED CONTROL

In multi-path control, usually, movements are made on the axes of path 1 according to a move command for path 1, and movements are made on the axes of path 2 according to a move command for path 2 (independent control in each path). However, the superimposed control function enables the travel distance on an arbitrary axis of one path to be superimposed on the travel distance on an arbitrary axis of another path.

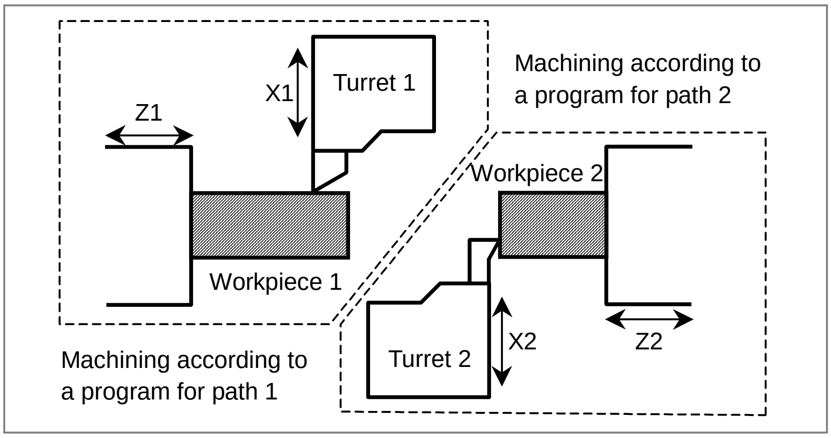
NOTE

This function is an optional function.

Explanation

- Independent control in each path

Movements on the axes (X1, Z1, and so on) of path 1 are made according to a move command for path 1, and movements on the axes (X2, Z2, and so on) of path 2 are made according to a move command for path 2.



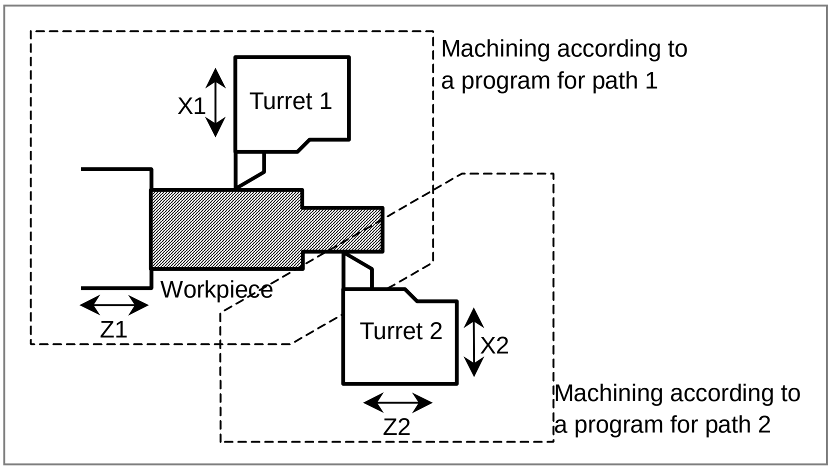
- Superimposed Control

To the travel distance on an axis (slave axis) for which an ordinary move command is executed, the travel distance on the axis (master axis) of another path is added. Superimposed control resembles synchronous control. In superimposed control, however, a movement on the slave axis can be specified with a command for the path to which the slave axis belongs.

The master axis and slave axis may belong to the same path, or the master axis may belong to one axis and the slave axis may belong to another. Moreover, multiple slave axes can be specified for one master axis. By parameter setting, the move directions on the master axis and slave axis can be reversed from each other.

Example)

A move command for the Z1 axis of path 1 is superimposed on the travel distance on the Z2 axis of path 2.



1.13 FLEXIBLE PATH AXIS ASSIGNMENT

This function can remove each controlled axis from the control of each path and assign them as the controlled axis in the other path.

Using this function makes it possible to control one motor in multiple paths. For example, in the machine having the axis configuration shown in Example 1 (X1 and Z in path 1 and X2 in path 2), the Z-axis can be removed from path 1 and assigned to path 2 to form a different axis configuration (X1 in path 1 and X2 and Z in path 2), therefore requiring no dummy axis unlike composite control.

In the rotary index machine shown in Example 2, axes can be switched among paths.

If an assignment command is issued for an axis yet to be removed, the command waits for the axis to be removed. In this case, no waiting M code is needed.

The new axis configuration (after flexible path axis assignment) is preserved even after the CNC power is turned off.

NOTE

This function is an optional function.

Explanation

(Example 1)

In this example, the Z-axis is switched from path 1 to path 2.

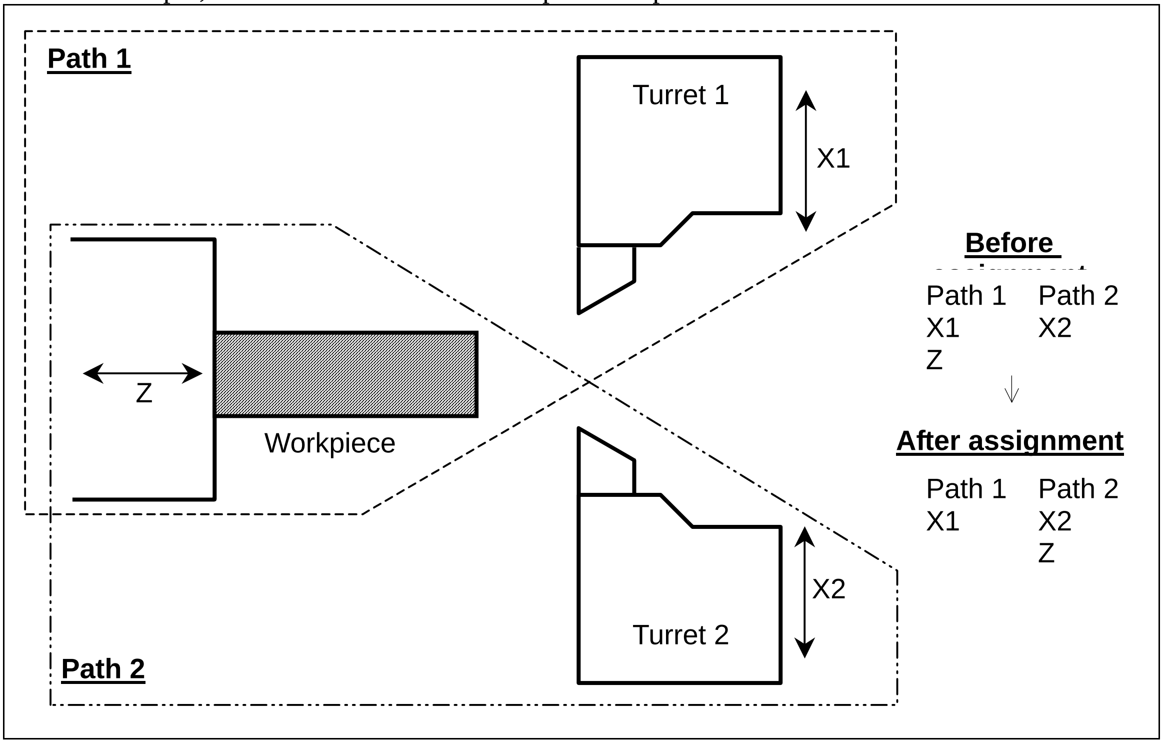
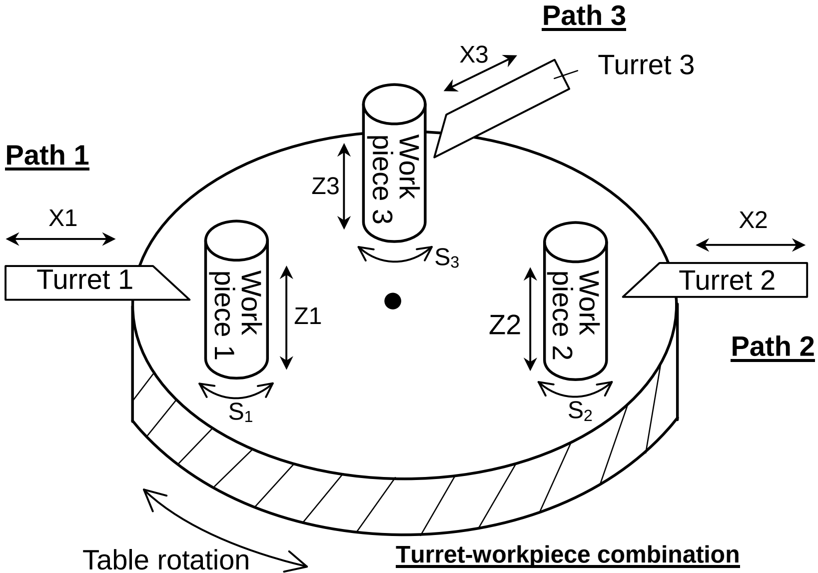


Fig.1.13 (a)

(Example 2)

In this example, the Z1 axis is switched from path 1 to path 2 or 3.
(Rotary index machine)



Turret-workpiece combination

- Table position (1)
  Axis configuration: Path 1(X1－Z1), Path 2(X2－Z2), Path 3(X3－Z3)
  ↓
- Table position (2)
  Z1, Z2, and Z3 are removed.
  Z2, Z3, and Z1 are assigned, respectively, to paths 1, 2, and 3.
  Axis configuration: Path 1(X1－Z2), Path 2(X2－Z3), Path 3(X3－Z1)
  ↓
- Table position (3)
  Z1, Z2, and Z3 are removed.
  Z3, Z1, and Z2 are assigned, respectively, to paths 1, 2, and 3.
  Axis configuration: Path 1(X1－Z3), Path 2(X2－Z1), Path 3(X3－Z2)

Fig.1.13 (b)

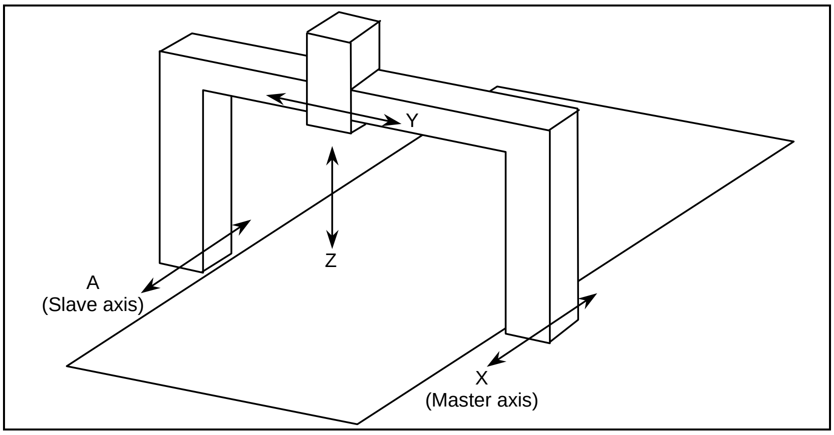
The flexible path axis assignment function provides the following three commands.

1. Controlled-axis removal command
   A specified axis is removed from under control of a specified path.
   No CNC program can direct the removed axis any more.
2. Controlled-axis assignment command
   A specified axis is placed under control of a specified path.
3. Controlled-axis exchange command
   Two specified axes can be exchanged directly.

1.14 AXIS SYNCHRONOUS CONTROL

When a movement is made along one axis by using two servo motors as in the case of a large gantry machine, a command for one axis can drive the two motors by synchronizing one motor with the other. Moreover, by using a feedback signal from each motor, a positional difference (synchronous error) between the two motors is detected to compensate for the synchronous error. When a synchronous error exceeding a set value occurs, a synchronous error check can be made to issue an alarm and stop a movement along the axis.

An axis used as the reference for axis synchronous control is referred to as a master axis (M-axis), and an axis along which a movement is made in synchronism with the master axis is referred to as a slave axis (S-axis).

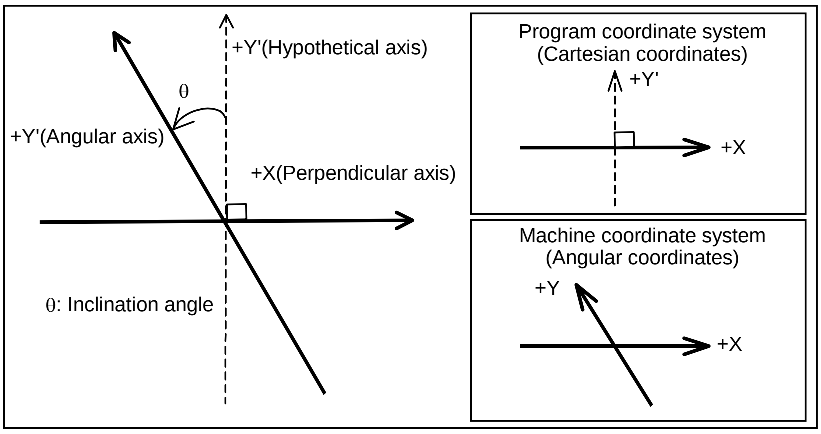


Even when synchronous error compensation is not used, the synchronous establishment function can be used for automatic compensation to eliminate a machine coordinate error in cases such as emergency stop cancellation.
An external signal can be used to turn synchronization on and off. When synchronization is turned on and off using an external signal, synchronous error compensation cannot be used.

1.15 ANGULAR AXIS CONTROL

When the angular axis installed makes an angle other than 90° with the perpendicular axis, the angular axis control function controls the distance traveled along each axis according to the inclination angle as in the case where the angular axis makes 90° with the perpendicular axis.
Arbitrary axes can be specified as a set of an angular axis and perpendicular axis by parameter setting.
The actual distance traveled is controlled according to an inclination angle. However, a program, when created, assumes that the angular axis and perpendicular axis intersect at right angles.

NOTE
This function is an optional function.



1.16 TANDEM CONTROL

If a single motor cannot produce sufficient torque to move a large table, for example, this function allows two motors to be used. By means of this function, two motors can be used to perform movement along a single axis.
Positioning is carried out only for the master axis. The slave axis is used only to produce a torque. By means of this function, double the amount of torque can be obtained.

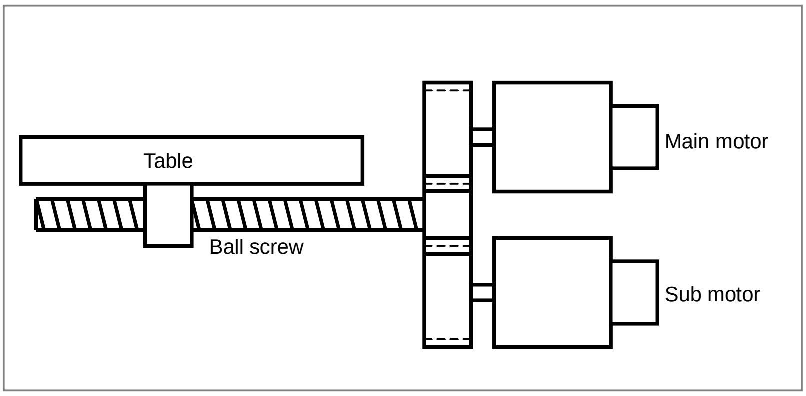


Fig.1.16 (a) Tandem control

The CNC generally processes the two axes of tandem control as a single axis. In the management of servo parameters and the monitoring of servo alarms, however, the two axes are handled individually.

1.17 TANDEM DISTURBANCE ELIMINATION CONTROL

This function suppresses vibration caused by interference between the main axis and sub-axis in position tandem control (feed axis synchronization).

NOTE

This function is an optional function.

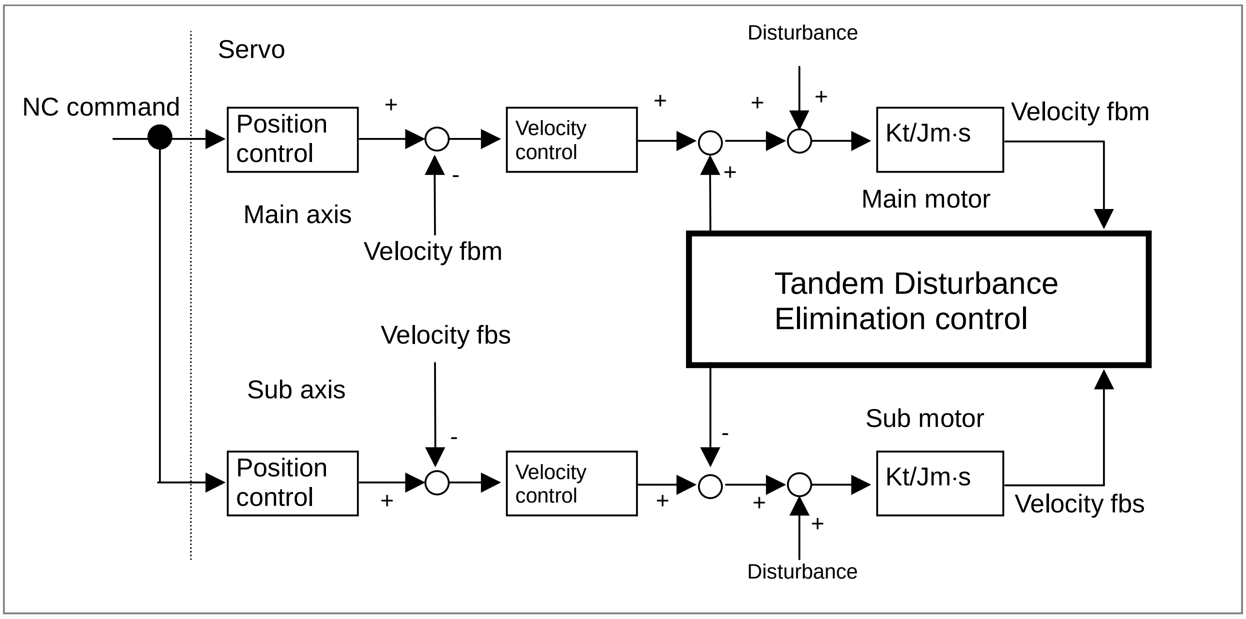


Fig.1.17 (a) Tandem disturbance elimination control

1.18 TORQUE CONTROL

For a PMC controlled axis, continuous feed based on torque control is performed.
Control on a PMC controlled axis can be switched from position control to torque control, so that the servo motor outputs torque as specified by the NC.

1.19 POLE POSITION DETECTION FUNCTION

When a motor manufactured by other than FANUC is driven, the magnetic pole position of the motor is detected.

NOTE

1. This function is an optional function.
2. This function cannot be used with a vertical axis to which force is applied at all times.
3. This function cannot be used with an axis when the axis is completely locked.

1.20 CONTROL AXIS DETACH

These signals release the specified control axes from control by the CNC. When attachments are used (such as a detachable rotary table), these signals are selected according to whether the attachments are mounted. When multiple rotary tables are used in turn, the tables must use motors of the same model.

1.21 HIGH PRECISION OSCILLATION FUNCTION

Overview

In this function, the feedrate of oscillation axis (equivalent to an axis that is moved vertically and repeatedly for grinding) changes along sine curve. This function is effective to improve, the accuracy of movement between upper dead point and lower dead point.
Moreover, look-ahead feed forward function can be applied to oscillation motion, then higher accuracy can be achieved even if oscillation feedrate or upper or lower dead point is changed.

NOTE

This function is an optional function.

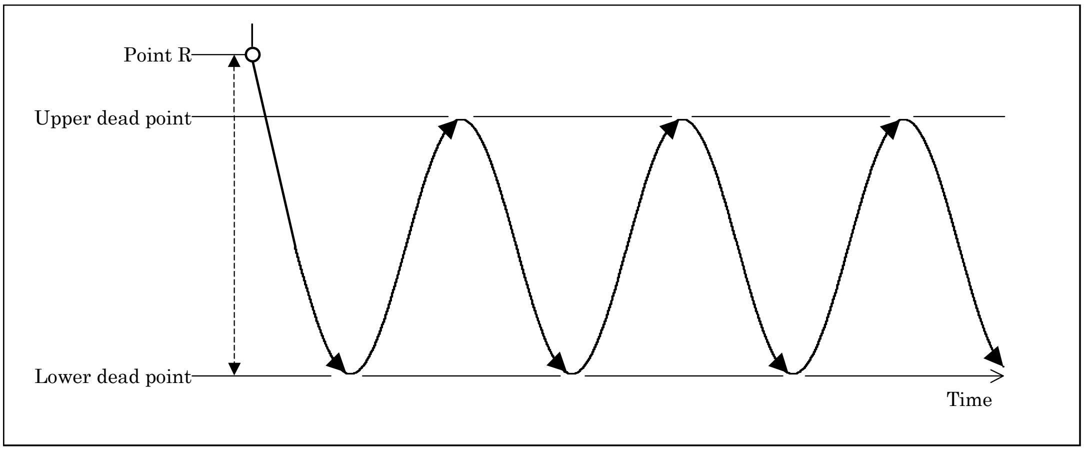


Fig.1.21 (a)

Format

G81.1 Z_ Q_ R_ F_ ;

Z : Upper dead point (In case that an axis is other than the Z-axis, specify the axis address.)

Q : Distance between the upper dead point and lower dead point
(Specify the distance as an incremental value from the upper dead point.)

R : Distance from the upper dead point to point R
(Specify the distance as an incremental value from the upper dead point.)

F : Oscillation base feedrate

G80 ; Cancels oscillation

1.22 INCREMENT SYSTEM

Three types of increment systems are available as indicated in the table below, and can be chosen by parameter setting.

Table 1.22 (a) Increment system

Name of increment system	Least input increment		Least command increment	
IS-A	0.01	mm	0.01	mm
	0.001	inch	0.001	inch
	0.01	deg	0.01	deg
IS-B	0.001	mm	0.001	mm
	0.0001	inch	0.0001	inch
	0.001	deg	0.001	deg
IS-C	0.0001	mm	0.0001	mm
	0.00001	inch	0.00001	inch
	0.0001	deg	0.0001	deg

The least command increment is either metric or inch depending on the machine tool. Set metric or inch to the parameter.

For selection between metric and inch for the least input increment, G code (G20 or G21) or a setting parameter selects it.

By parameter setting, a least input increment 10 times greater than a least command increment can be set as indicated in the table below.

Table 1.22 (b) Increment system

Name of increment system	Least input increment		Least command increment	
IS-B	0.01	mm	0.001	mm
	0.001	inch	0.0001	inch
	0.01	deg	0.001	deg
IS-C	0.001	mm	0.0001	mm
	0.0001	inch	0.00001	inch
	0.001	deg	0.0001	deg

NOTE

When the increment system is IS-A or pocket calculator type decimal point programming is used, a least input increment 10 times greater than a least command increment cannot be set.

1.23 FLEXIBLE FEED GEAR

The detection multiply (DMR) can be extended to set DMR=n/m by using two parameters n and m.

1.24 DUAL POSITION FEEDBACK

In general, a machine with a large backlash may operate stably with a semi-closed loop but may vibrate with a closed loop. This function exercises control so that such a machine can operate stably with a closed loop as in the case of a semi-closed loop.

NOTE
This function is an optional function.

The block diagram of dual position feedback control is shown Fig. 1.24 (a).

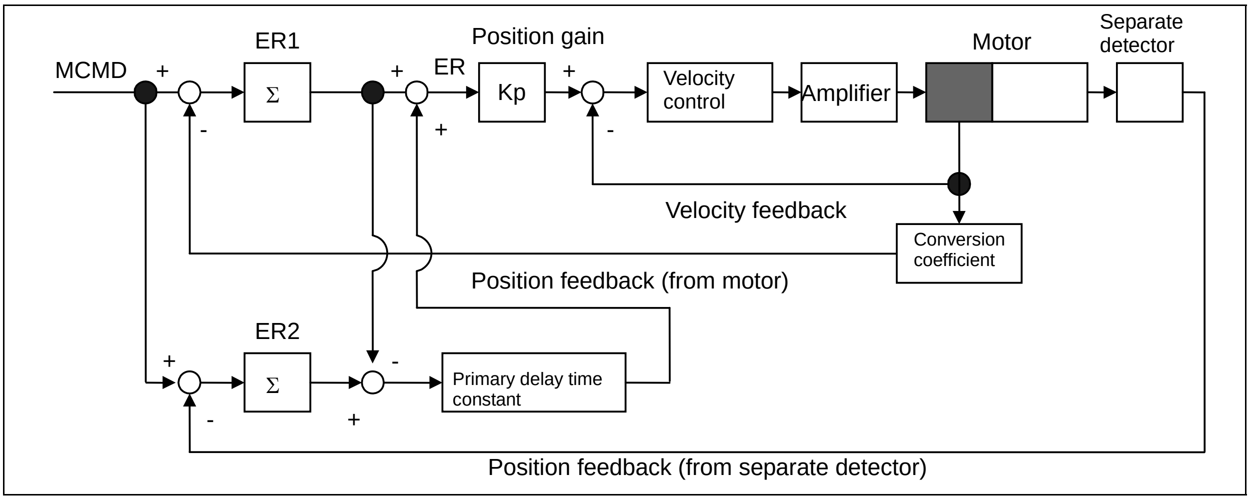


Fig. 1.24 (a) Block diagram of dual position feedback control

1.25 HRV CONTROL

HRV control is a digital servo current control method, and the HRV control system includes servo HRV2, and servo HRV3. By employing these control methods, even higher speed, higher precision, and higher machining speed can be achieved.

- Servo HRV control system

Servo HRV control ─┬ Servo HRV2 control
                   └ Servo HRV3 control

HRV control has three features:

(1) A disturbance elimination filter for eliminating low-frequency vibration from a low-rigidity machine has been developed.
(2) Smoother feed is made possible by a higher-precision servo amplifier and detector.
(3) By employing high-speed DSP, a current control cycle higher than the conventional one is made achievable with the standard servo system.

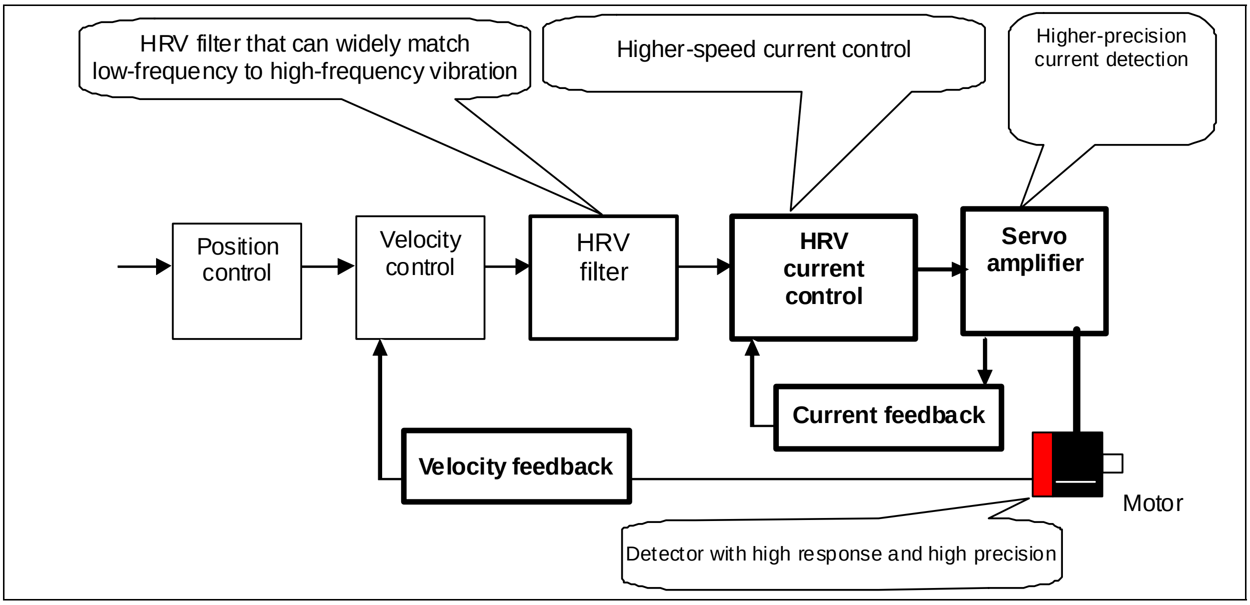


Fig.1.25 (a) HRV control

1.26 INCH/METRIC CONVERSION

Either inch or metric input (least input increment) can be selected by G code.

Format

G20 ;	Inch input
G21 ;	Metric input

Explanation

This G code must be specified in an independent block before setting the coordinate system at the beginning of the program. Do not specify this G code in the middle of a program.
Moreover, inch/metric conversion is possible in setting data setting.
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. The unit of data input for degrees remains unchanged. The unit systems for the following values are changed after inch/metric conversion:

- Feedrate commanded by F code
- Positional command
- Workpiece origin 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.

1.27 INTERLOCK

1.27.1 Start Lock

This function disables movement along axes during automatic operation (memory operation, DNC operation, or MDI operation).

1.27.2 All-axis Interlock

Feed on all axes can be disabled. If all-axis interlock is applied during movement, a gradual stop occurs. When the all-axes interlock signal is canceled, movement restarts.

1.27.3 Each-axis Interlock

Feed on a specified axis can be disabled, independent of other axes. If each-axis interlock is applied to an axis during cutting feed, a gradual stop occurs on all axes of the movable machine section.
When the interlock signal is canceled, movement restarts.

1.27.4 Each-axis Direction Interlock

For each axis, axial movement can be disabled in a specified axis direction only. If each-axis interlock is applied to an axis during cutting feed, a gradual stop occurs on all axes of the movable machine section. When the interlock signal is canceled, movement restarts.

1.27.5 Block Start Interlock

During automatic operation, the start of the next block can be disabled. A block whose execution has already been started continues to be executed up to the end of the block.
When block start interlock is canceled, the execution of the next block starts.

1.27.6 Cutting Block Start Interlock

During automatic operation, the start of a block including a move command other than a command for positioning can be disabled.
When cutting block start interlock is canceled, the execution of the next block starts.
If spindle rotation is specified or the spindle speed is changed, the next cutting block can be executed at the desired spindle speed by applying cutting block start interlock until the spindle reaches the desired speed.

1.28 MACHINE LOCK

1.28.1 All-axis Machine Lock

The change of the position display can be monitored without moving the machine.
When all-axis machine lock signal is set to “1”, output pulses (move commands) to the servo motors are stopped in manual or automatic operation. The commands are distributed, however, updating the absolute and relative coordinates. The operator can therefore check if the commands are correct by monitoring the position display.
Machine lock during operation can be enabled even in the middle of block execution.

1.28.2 Each-axis Machine Lock

With the each-axis machine lock signal, machine lock can be applied to each axis.

1.29 EMERGENCY STOP

An emergency stop stops all commands and instantly stops the machine. Connect the emergency stop signal to both of the control unit side and servo unit side.

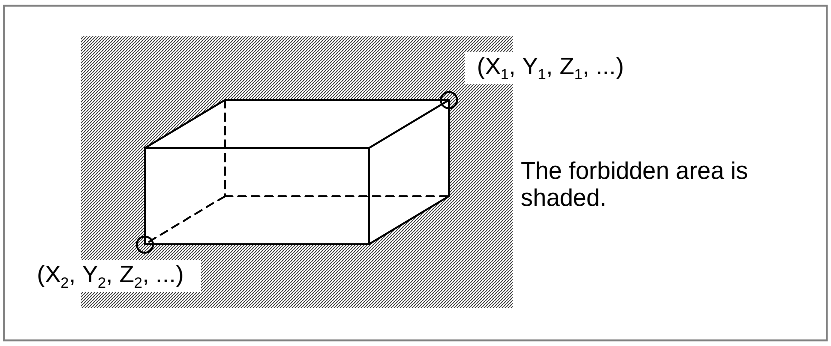
When an emergency stop is applied, servo system activation is canceled, and the servo ready signal is turned off. However, the travel distance of the machine during that time is reflected in the current position, so that the position data is not lost (follow-up). If the position detection system is normal, operation can be restarted after emergency stop cancellation without performing a reference position return operation again.

1.30 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 alarm is displayed.
An overtravel signal is provided for each direction on each axis.

1.31 STORED STROKE CHECK 1

A machine movable range is set with coordinates in the machine coordinate system in parameters. If the machine attempts to move beyond the range, it is decelerated and stopped and an alarm is displayed. This function is enabled after manual reference position return is performed after power-on.
It can be used instead of an overtravel limit switch (hardware component).
When both functions are used, both are valid.
Unlike an overtravel limit switch, this function checks whether the position at which the machine is stopped after decelerated from the current position is beyond the range.
The stroke check 1 release signal common to all axes can be set to 1 so that the control unit does not make stroke check 1.



1.32 STORED STROKE CHECK 1 AREA EXPANSION

In stored stroke check 1, up to eight different forbidden areas can be defined and selected.
Since the number of selectable forbidden areas increases, different forbidden areas can be used for different machine specifications.

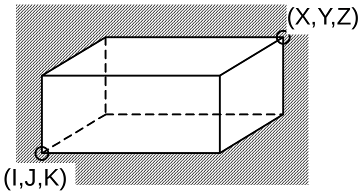
NOTE
This function is an optional function.

1.33 STROKE LIMIT EXTERNAL SETTING

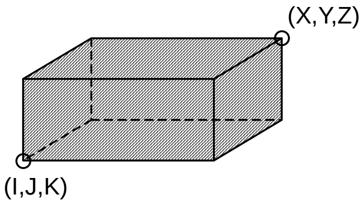
When a tool is changed, the tool tip is aligned with the end of the limit area and signals are input. This operation sets the machine position (machine coordinates) at that time as the limit position in stored stroke check parameters. A setting signal is provided for each direction of each axis.

1.34 STORED STROKE CHECK 2

For stored stroke check 2, the outside or inside of the area specified by parameters or a program is defined as the forbidden area. As a limit position, specify a distance from the origin of the machine coordinate system. This function is enabled after manual reference position return is performed at power-on. When the limits are specified in a program, they can be set for the X-, Y-, and Z-axes. For this reason, the forbidden area can be changed according to the workpiece. Whether to define the inside or outside of the specified area as the forbidden area is determined by setting the corresponding parameter.



When the outside of the specified area is defined as the forbidden area.



When the inside of the specified area is defined as the forbidden area.

Format

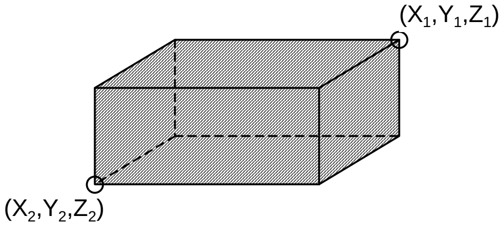
G22 X_ Y_ Z_ I_ J_ K_ ; Stored stroke check 2 on

X, Y, Z : Coordinates in the + direction of stored stroke check 2
I, J, K : Coordinates in the - direction of stored stroke check 2
The address is X, Y, Z, I, J, or K. X and I, Y and J, and Z and K specify a forbidden area for the X-axis, Y-axis, and Z-axis of the basic three axes, respectively. If an address is omitted, a stroke check is make according to the parameter settings.

G23 ; Stored stroke check 2 off

1.35 STORED STROKE CHECK 3

The inside of a parameter-set area is a forbidden area.



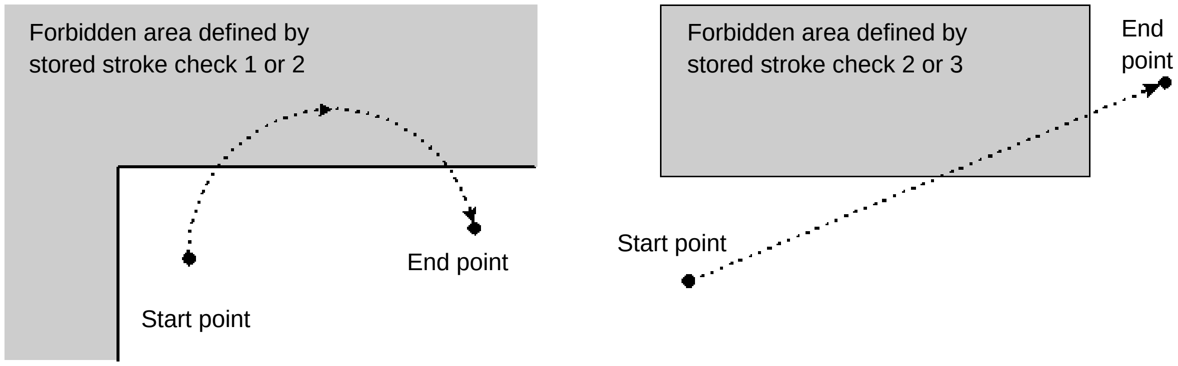
When the inside of the specified area is defined as the forbidden area.

1.36 STROKE LIMIT CHECK BEFORE MOVE

During automatic operation, before the movement specified by a given block is started, whether the tool enters the inhibited area defined by stored stroke check 1, 2, or 3 is checked by determining the coordinate of the end point from the current position of the machine and a specified amount of travel. If the tool is found to enter the inhibited area defined by a stored stroke limit, the tool is stopped immediately upon the start of movement for that block, and an alarm is displayed.

1.37 CHECK OF THE TOOL PATH BETWEEN BLOCKS BY STROKE LIMIT CHECK BEFORE MOVE

In Stroke limit check before move, whether the tool enters the forbidden area defined by Stored stroke limit 1, 2, or 3 is checked on the tool path of movement command in addition to checking the end point.



When forbidden area is outside. **When forbidden area is inside.**

Immediately upon movement commencing from the start point, the tool is stopped to enable a stroke limit check before moving to be performed before movement.

▨ : Forbidden area

1.38 CHECKING THE STORED STROKE DURING THE TIME FROM POWER–ON TO THE REFERENCE POSITION ESTABLISHMENT

This function stores the machine coordinates present immediately before the power is turned off. Therefore, immediately after the power is turned on again, this function can restore the approximate machine coordinates and enables the function for checking the stored stroke during the time from power–on to the reference position establishment.
Even before the reference position is established by manual reference position return, the stored stroke check can be performed using approximate machine coordinates.

1.39 STROKE LIMIT AREA CHANGING FUNCTION

This function can be used to rewrite the parameters that set the + side coordinate value and the - side coordinate values of the stroke limits even when the axis is traveling. The parameters can be rewritten by the PMC window function (WINDW: SUB52), FOCAS2, and a C Language Executor. The new stroke limit range is enabled immediately after the parameters are rewritten by any of these functions.

The machining cycle time can be reduced because this function can rewrite parameters even if some axes are moving.

NOTE

This function is an optional function.

1.40 STORED STROKE LIMIT RANGE SWITCHING FUNCTION BY SIGNAL

The range stored stroke limit can be switched by input signal of PMC. Therefore, the range stored stroke limit can easily be set again.

NOTE

1 This function is an optional function.
2 Stroke limit area changing function is a function to rewrite the parameter (No.1320-No.1327). This function is a function that switches the range of stroke limit to the value that is set to data table (D) of PMC without rewriting the parameter (No.1320-No.1327). This function can switch the range of stored stroke limit more high speed than stroke limit area changing function.

1.41 CHUCK AND TAIL STOCK BARRIER (T SERIES)

T

The chuck and tail stock barrier function prevents damage to the machine by checking whether the tool tip interferes with either the chuck or tail stock.

Specify an area into which the tool may not enter (entry-prohibition area). This is done using the special setting screen, according to the shapes of the chuck and tail stock. If the tool tip should enter the set area during a machining operation, this function stops the tool and outputs an alarm message.

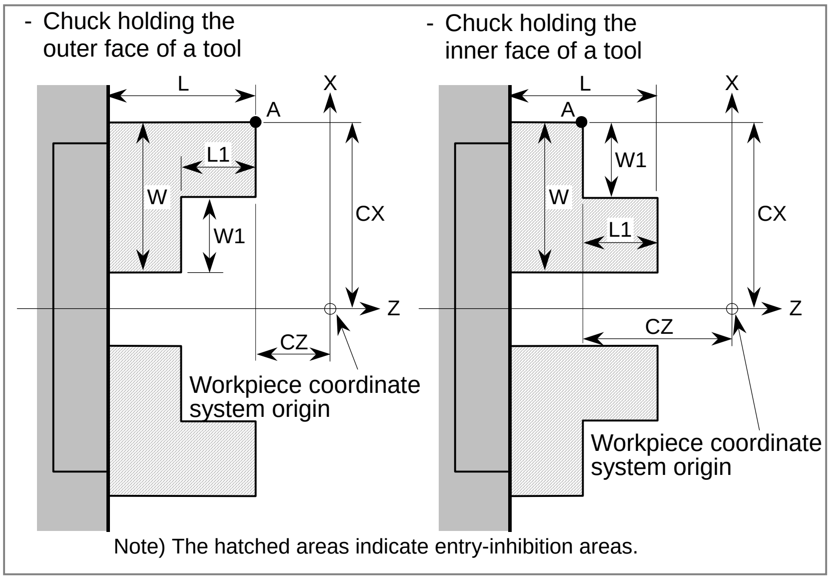
The tool can be removed from the prohibited area only by retracting it in the direction from which the tool entered the area.

This function can be enabled or disabled by G22 (stored stroke check 2 on), G23 (stored stroke check 2 off), and a machine-side signal.

G code	Tail stock barrier signal	Tail stock barrier	Chuck barrier
G22	0	Valid	Valid
	1	Invalid	Valid
G23	0	Invalid	Invalid
	1	Invalid	Invalid

Explanation

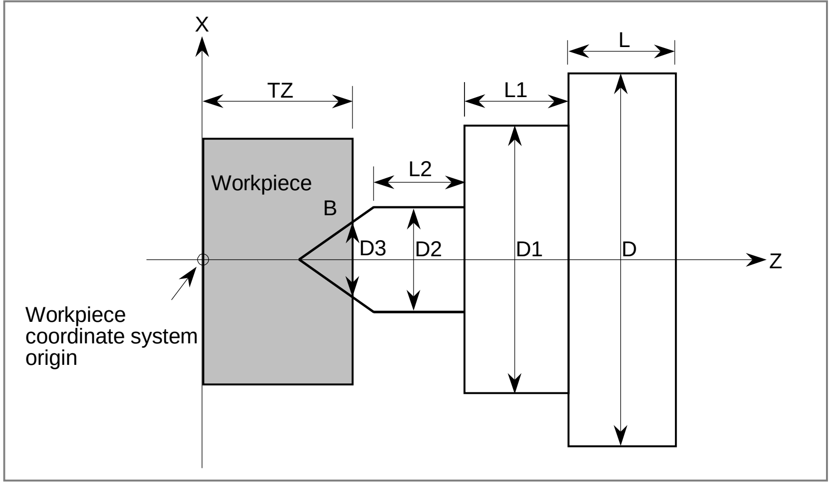
- Chuck figure definition



Note) The hatched areas indicate entry-inhibition areas.

Symbol	Description
TY	Chuck-shape selection (0: Holding the inner face of a tool, 1: Holding the outer face of a tool)
CX	Chuck position (along X-axis)
CZ	Chuck position (along Z-axis)
L	Length of chuck jaws
W	Depth of chuck jaws (radius)
L1	Holding length of chuck jaws
W1	Holding depth of chuck jaws (radius)

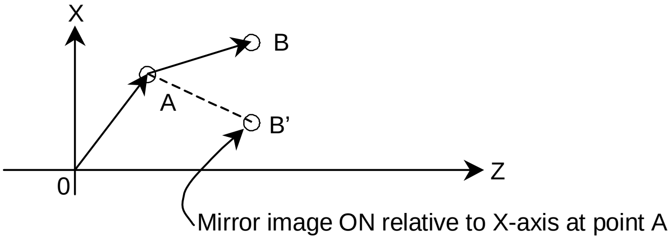
- Tail stock figure definition



Symbol	Description
TZ	Tail stock position (along the Z-axis)
L	Tail stock length
D	Tail stock diameter
L1	Tail stock length (1)
D1	Tail stock diameter (1)
L2	Tail stock length (2)
D2	Tail stock diameter (2)
D3	Tail stock hole diameter (3)

1.42 MIRROR IMAGE

Mirror image can be applied to each axis, either by signals or by parameters (setting input is acceptable). All movement directions are reversed during automatic operation along axes to which a mirror image is applied.



Mirror image (Example for lathe system)

However, the following directions are not reversed:

- Direction of manual operation and direction of movement, from the intermediate position to the reference position during automatic reference position return (for the machining center system and lathe system)
- Shift direction for boring cycles (G76 and G87) (for machining center system only)

Signals indicate whether mirror image is applied to each axis. System variable contains the same information.

1.43 FOLLOW-UP

If the machine moves in the state in which position control on controlled axes is disabled (during servo-off, emergency stop, or servo alarm), feedback pulses are accumulated in the error counter. The CNC reflects the machine movement corresponding to the error count in the current position managed by the CNC. This operation is referred to as follow-up. When follow-up is performed, the current position managed by the CNC does not shift from the actual machine position.

So, operation can be restarted after emergency stop cancellation or servo alarm cancellation without performing a reference position return operation again.

You can select whether to perform follow-up for axes when the servo is turned off.

Follow-up is always performed during emergency stop or a servo alarm.

1.44 SERVO OFF / MECHANICAL HANDLE FEED

Place the controlled axes in the servo off state, stop the current to the servo motor, which disables position control. However, the position detection feature functions continuously, so the current position is not lost.

These signals are used to prevent the servo motors from overloading when the tools on the axes are mechanically clamped under certain machining conditions on the machine, or to move the machine by driving the motors by mechanical handles.

1.45 CHAMFERING ON/OFF (T SERIES)

T

In the threading cycle (G76), which is a multiple repetitive cycle for turning, and in the threading cycle (G92), which is a canned cycle, threading can be selected with the chamfering signal.

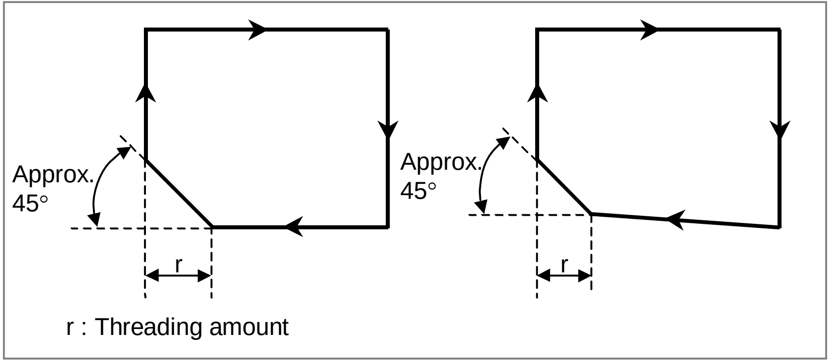


Fig. 1.45 (a) Straight threading and taper threading

1.46 INTERFERENCE CHECK FOR EACH PATH (T SERIES)

T

When tool posts on individual paths machine the same workpiece simultaneously, the tool posts can approach each other very closely. If the tool posts interfere with each other due to a program error or any other setting error, a serious damage such as a tool or machine destruction can occur.

If such a command that causes tool posts of paths to interfere with each other is specified, this function gradually stops the tool posts before the tool posts actually interfere with each other.

The contours and shapes of the tool posts on individual paths are checked to determine whether or not an interference occurs.

This function enables an interference check between two paths or interference check among multiple paths. Which check to make can be determined by parameter setting.

NOTE
This function is an optional function.

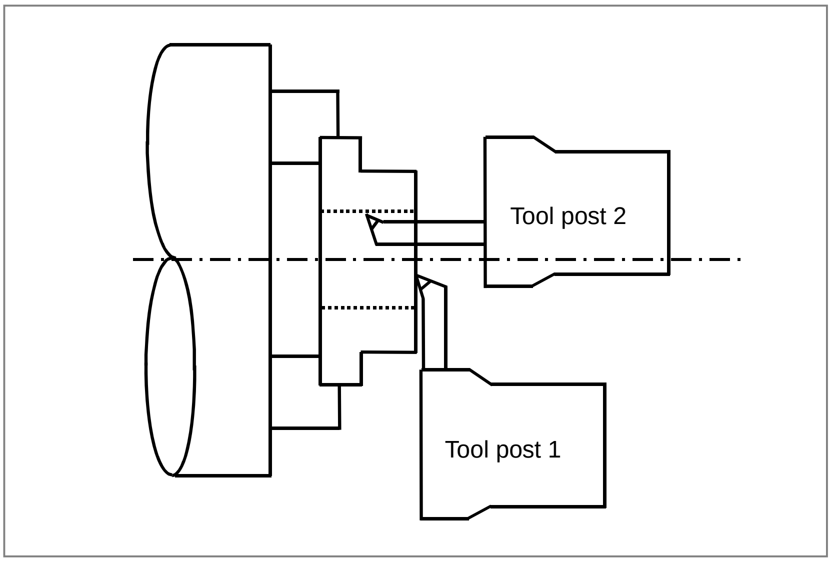


Fig.1.46 (a)

To make a path interference check, data including the relationships between the tool posts on individual paths and interference forbidden areas (that is, tool shapes) needs to be set.
Based on the interference forbidden areas of the tool currently selected on the tool post of each path and tool posts, an inter-path interference check determines whether the tools and tool posts interfere with each other, by checking whether those forbidden areas overlap each other as a result of movement of each tool post. If an interference occurs, the interfering tool posts gradually stop with an alarm.

1.47 UNEXPECTED DISTURBANCE TORQUE DETECTION FUNCTION

Machine collision, defective, and damaged cutters cause a large load torque on the servo and spindle motors, compared with normal rapid traverse or cutting feed. This function detects the disturbance torque on the motors and sends this value as an estimated load torque to the PMC. If the detected disturbance torque value is abnormally great compared with the value specified in the parameter, the function stops the servo motor as early as possible or reverses the motor by an appropriate value specified in a parameter, in order to minimize possible damage to the machine.
The unexpected disturbance torque detection function is further divided as follows:

(1) Estimated disturbance torque output function
 The CNC is always calculating the estimated disturbance torque for the motor (excluding acceleration/deceleration torque). The estimated disturbance torque output function enables the PMC to read the calculated torque using the window function.

(2) Unexpected disturbance torque detection alarm function
 This function stops motors or reverses them by an amount specified in a parameter, causing the CNC to output an alarm, whenever the disturbance torque is greater than the value specified in a parameter. (The function to reverse motors is effective only for servo motors.)
 This function can be used for the trigger condition of recording CNC information by Machine state monitoring function.

Unexpected disturbance torque detection can also be disabled only for specific axes by using parameter for the unexpected disturbance torque detection function and unexpected disturbance torque detection ignore signals. (This function is effective only for servo motors.)

1.48 POSITION SWITCH

Position switch signals can be output to the PMC while the machine coordinates along a controlled axes are within a parameter-specified ranges.
Using parameters, specify arbitrary controlled axes and machine coordinate operating ranges for which position switch signals are output.
Up to 10 position switch signals can be output.
Parameter can be set to use up to 16 position switch signals.

1.49 HIGH-SPEED POSITION SWITCH

The high-speed position switch function monitors the current position at shorter intervals than the normal position switch function to output a high-speed precise position switch signal.
In the same way as for the normal position switch function, using parameters, specify arbitrary controlled axes and machine coordinate operating ranges for which position switch signals are output.
Up to 6 high-speed position signals can be output. Parameter can be set to use up to 16 high-speed position switch signals.

NOTE
This function is an optional function.

1.50 LINEAR SCALE WITH ABSOLUTE ADDRESS REFERENCE MARK

1.50.1 Linear Scale Interface with Absolute Address Reference Mark

With this function, an absolute position can be identified if the interval of reference marks is known, because the intervals of two reference marks (one-rotation signals) differ from each other by a certain distance. This CNC measures one-rotation signal intervals by making a slight movement on an axis to calculate an absolute position. So, a reference position can be established without making a movement to the reference position on the axis.

NOTE
This function is an optional function.

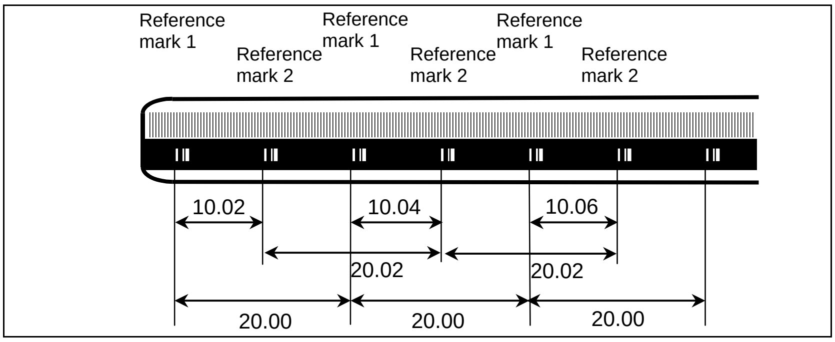


Fig.1.50.1 (a)

1.50.2 Linear Scale with Absolute Address Reference Mark Expansion

When a G00 command is specified or a move command based on jog feed is specified, this function enables a reference mark interval measurement to be made automatically in order to establish a reference position.

NOTE
This function is an optional function.

1.51 LINEAR SCALE WITH DISTANCE-CODED REFERENCE MARKS (SERIAL)

By using High-resolution serial output circuit for the linear scale with distance-coded reference marks (serial), the CNC measures the interval of referenced mark by axis moving of short distance and determines the absolute position.
This function enables high-speed high-precision detection by using High-resolution serial output circuit.
It is available that using maximum stroke 30 meters length.

1.52 ABSOLUTE POSITION DETECTION

An absolute position detector (absolute pulse coder) is an incremental pulse coder with an absolute counter. It detects the absolute position based on the value of the absolute counter. For an axis on which an absolute position detector is mounted, no reference position return is required at power-on because the machine position is always stored with batteries if the power to the CNC is turned off.

When the machine position has been brought into correspondence with the absolute position detector, the current position is read from the absolute counter at CNC power on and the machine and workpiece coordinate systems are automatically set using the value. In this case, you can immediately start automatic operation.

Restrictions described in the OPERATOR’S MANUAL and others that include those listed below are removed:

- "Reference position return is required after power-on."
- "The CNC can be used after reference position return is performed after power-on."

1.53 TEMPORARY ABSOLUTE COORDINATE SETTING

In the full closed system with an inner absolute position pulse coder (serial pulse coder) and an incremental scale, the position is set by using absolute position data from the inner absolute position pulse coder at the power on sequence. After that, the position is controlled with incremental data from the incremental scale. The position just after power on sequence is rough, and the manual reference position return is required to get the accurate position.

NOTE
This function is an optional function.

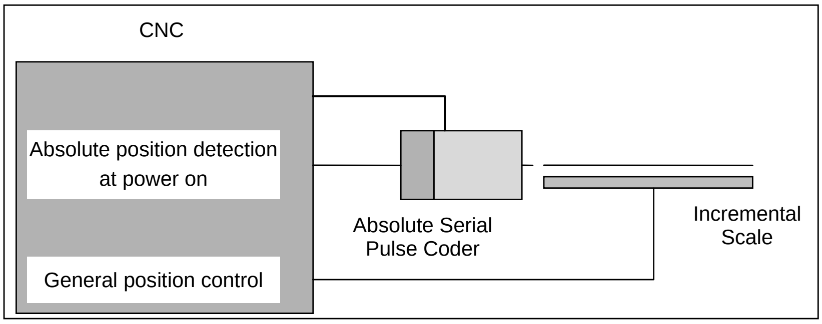


Fig.1.53 (a) The system with the Temporary Absolute Coordinate Setting

With this function, the position at the power on is rough, but the following functions are available before the reference position return.

- Stroke limit check
- Position switch

1.54 DUAL CHECK SAFETY

Setup for machining, which includes attaching and detaching a workpiece to be machined, and moving it to the machining start point while viewing it, is performed with the protection door opened. The dual check safety function provides a means for ensuring a high level of safety with the protection door opened.

NOTE
This function is an optional function.

The simplest method of ensuring safety when the protection door is open is to shut off power to the motor drive circuit by configuring a safety circuit with a safety relay module. In this case, however, no movements can be made on a move axis (rotation axis). Moreover, since the power is shut off, some time is required before machining can be restarted. This drawback can be corrected by adding a motor speed detector to ensure safety. However, the addition of an external detector may pose a response problem, and the use of many safety relay modules results in a large and complicated power magnetics cabinet circuit.

With the dual check safety function, two independent CPUs built into the CNC monitor the speed and position of motors in dual mode. An error in speed and position is detected at high speed, and power to the motor is shut off via two independent paths. Processing and data related to safety is cross-checked by two CPUs. To prevent failures from being built up, a safety-related hardware and software test must be conducted at certain intervals time.

The dual check safety system need not have an external detector added. Instead, only a detector built into a servo motor or spindle motor is used. This configuration can be implemented only when those motors, detectors built into motors, and amplifiers that are specified by FANUC are used. When an abnormality related to safety occurs, the dual check safety function stops operation safely.

The dual check safety function ensures safety with the power turned on, so that an operator can open the protection door to work without turning off the power. A major feature of the dual check safety function is that the required time is very short from the detection of an abnormality until the power is shut off. A cost advantage of the dual check safety function is that external detectors and safety relays can be eliminated or simplified.

If a position or speed mismatch is detected by a cross-check using two CPUs, the safety function of the Dual Check Safety works the power to be shut off (MCC off) to the motor drive circuit.

1.55 FUNCTION OF DECELERATION STOP IN CASE OF POWER FAILURE

If a power failure occurs during an axial movement, this function stops the movement by decreasing the speed on each axis at a rate specified in parameter. This function prevents the machine from being damaged by an overrun.

1.56 ABSOLUTE POSITION DETECTOR OF ROTARY SCALE WITHOUT ROTARY DATA

This manual describes how to deal with an absolute position detector (absolute pulse coder) or a rotary scale with distance-coded reference marks (serial), when the rotary scale without rotary data is used, such as Heidenhain rotary scale RCN223, RCN723, RCN220, or Futaba rotary scale FRR902L3DB.

1.57 AXIS IMMEDIATE STOP FUNCTION

When the movement long an axis must be immediately stopped, the axis immediate stop function stops the movement using the axis immediate stop start signal and outputs an alarm. In the AI advanced preview control (M Series) / AI contour control mode, this function changes the acceleration rate in acceleration/deceleration before interpolation and stops the movement immediately.

NOTE

This function is an optional function.

2 OPERATION

Chapter 2, "OPERATION", consists of the following sections:

2.1 OPERATION MODE

2.1.1 Automatic Operation (Memory Operation)

Program registered in the memory can be executed.

2.1.2 MDI Operation

Multiple blocks can be input and executed on the MDI unit.

2.1.3 DNC Operation

A program can be executed while being read from the input device connected with the RS232C interface.

2.1.4 DNC Operation with Memory Card

A program can be executed while being read from the memory card.

2.1.5 Schedule Operation

Programs can be executed by specifying their program file numbers on the memory card in the sequence in which they are to be executed and the number of times that they are to be executed.

2.2 PROGRAM SEARCH

By operating the MDI panel, a program to be executed can be selected from the programs stored in the program memory.

2.3 SEQUENCE NUMBER SEARCH

By operating the MDI panel, a block can be selected according to a sequence number in the currently selected program in the program memory.
When a program is to be executed starting with a block in the middle of the program, the sequence number of the block is to be specified to search for the sequence number.

2.4 SEQUENCE NUMBER COMPARISON AND STOP

If a block containing a specified sequence number appears in the program being executed, operation enters single block mode after the block is executed. By setting operation, the operator can set a sequence number through the MDI panel. This function is useful for checking a program, because the program can be stopped at a desired position without modifying the program.

2.5 PROGRAM RESTART

When the tool is broken during automatic operation, or when a machining operation interrupted by a holiday is to be restarted after the holiday, you can restart machining from a desired block by specifying the sequence number of the block or the number of blocks from the beginning of the program to the block where machining is to restart.
This function can also be used as a high-speed program check function.

P type:	Restart after a tool is broken down
Q type:	Restart after the power is turned off (after a holiday, etc.) or after emergency stop is canceled

2.6 QUICK PROGRAM RESTART

This function allows program restart with simpler operations.

- Enables the block at which operation was interrupted to be checked easily on the program restart setting screen, which is provided specifically for this function.
- Automatically extracts blocks (such as positioning and auxiliary function blocks) from which to easily restart machining with automatic operation and displays them on the program restart setting screen. Allows the operator to specify a block from which to restart machining just by selecting a displayed block.
- Also allows the operator to restart machining from a block which is not displayed on the program restart setting screen.
- Keeps storing automatically extracted data after power-off.
- The following two types of restart methods are available:
  - Search method :
    - Simulates a program from the beginning to the block from which to restart machining while restoring modal and other information.
  - Direct jump:
    - Available only for restarting machining from an automatically extracted block. Jumps to the block from which to restart machining at a high speed. It is necessary to restore the status by MDI or manual operation because modal and other information is not restored in this mode.
- The auxiliary function output in program restart function is also available (only for the search method).

NOTE
This function is an optional function.

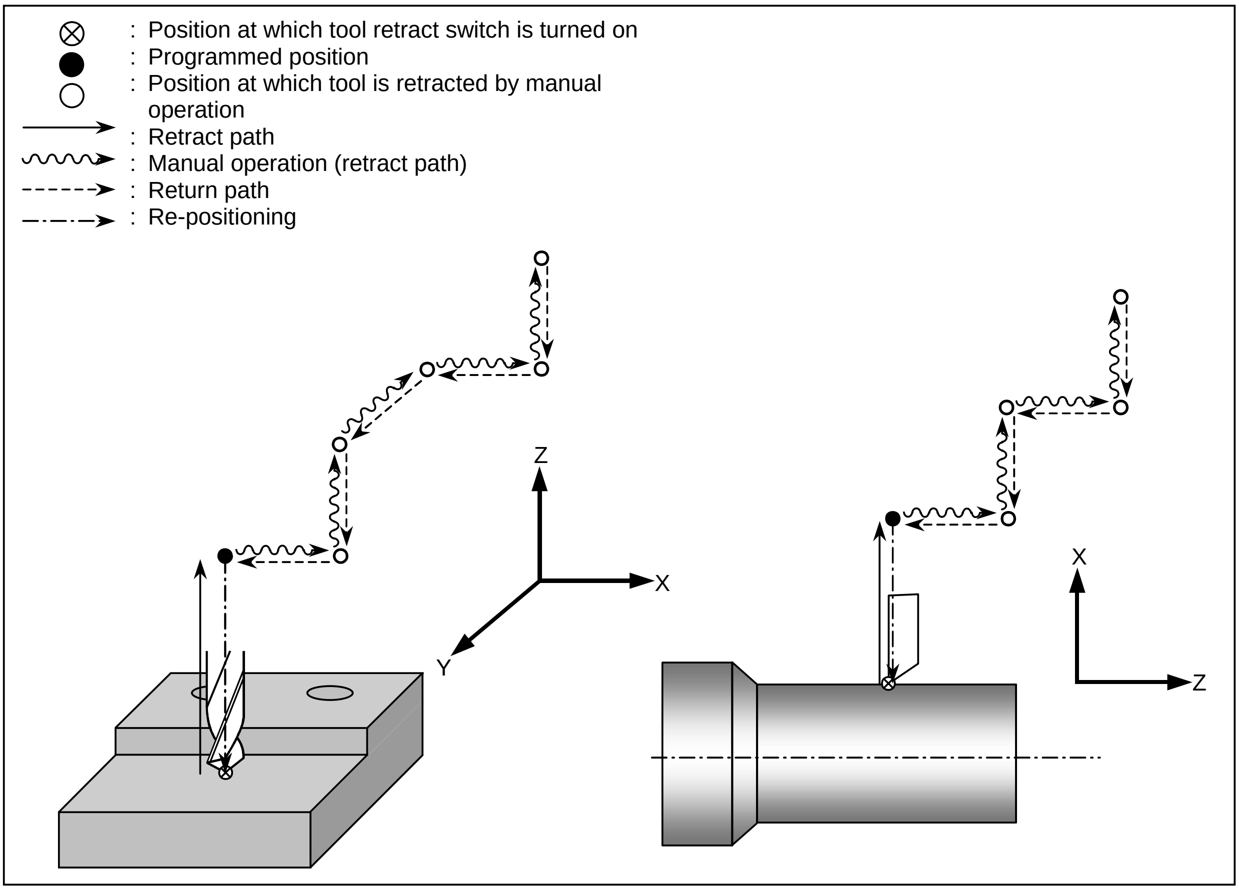
2.7 TOOL RETRACT AND RECOVER

2.7.1 Tool Retract and Recover

You can efficiently perform tool retraction for changing a damaged tool or checking the machining status, as well as tool recovery for restarting machining.

If you set a retraction (position) with a program in advance, you can perform retraction using a tool retraction signal, which you can use for retraction when you detect tool damage, for example.

1. When you input a tool retraction signal during the execution of automatic operation, retraction is performed up to the retraction position specified in the program.
2. By inputting a tool retraction signal, the system enters tool retraction mode.
3. Then, if you switch to manual mode and move the tool with manual operations (jog feed, incremental feed, handle feed, and manual numeric command), up to 20 points on the tool path are recorded automatically.
4. When you input a tool recovery signal, the tool automatically returns to the retraction position, moving backward along the paths along which it has moved with manual operations. (recovery)
5. When you perform a cycle start, recovery (repositioning) is performed up to the position at which the tool retraction signal is input.



NOTE

This function is an optional function.

Format

Specify a retraction axis and distance in the following format:

Specify the amount of retraction, using G10.6.

G10.6 IP_ ;

IP:	In incremental mode, retraction distance from the position where the retract signal is turned on In the absolute mode, retraction distance to an absolute position The specified amount of retraction is effective until G10.6 is next executed. To cancel the retraction, specify the following:

G10.6 ; (as a single block containing no other commands)

G10.6 is a one-shot G code of group 00.

2.7.2 Improvement of Tool compensation for Tool Retract and Recover

In this function, when the recovery operation or re-positioning operation is started, the updated compensation value is used. Therefore the restart operation is performed with the updated compensation value.

If this function is effective, when the recovery operation or re-positioning operation is started, the updated compensation value is used and the restart operation is performed with the updated compensation value.

Thus, if the compensation value is updated after exchanging the tool, the tip point of tool becomes the same position as it of before exchanging tool in the restart operation thereafter.

2.8 MANUAL INTERVENTION AND RETURN

If you use feed hold to stop the tool from moving an axis during automatic operation and restarts the tool after manual intervention, for example, for checking a cutting surface, the tool can resume automatic operation after automatically returning to the pre-intervention position.

2.9 MALFUNCTION PREVENT FUNCTIONS

These functions monitor the CNC internal status and check that related data is within the allowable range. If an invalid state due to a deteriorated hardware component or noise is detected, these functions stop the machine with an alarm to prevent any malfunction.

The following malfunction prevention functions are available:
- Checking the maximum speed of the servo motor
- Checking the maximum acceleration of the servo motor
- Checking the maximum speed of the spindle motor
- Checking the stored stroke limit at the end point
- Monitoring execution of NC command analysis
- Monitoring execution of acceleration/deceleration after interpolation

2.10 WRONG OPERATION PREVENTION FUNCTION

An improper tool offset setting or an improper operation of the machine can result in the workpiece being cut inadequately or the tool being damaged. Also, if data is lost due to an operation mistake, it takes extra time to recover from the mistake.
The wrong operation prevention functions described below are meant to prevent the operator from performing any unintended operation.

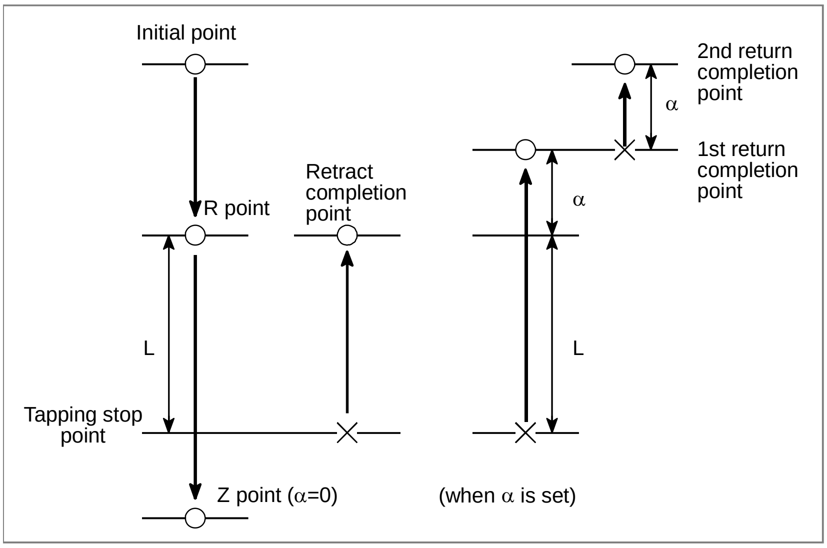
1. Functions that are used when data is set
    - Data check to verify that the offset data is within the valid setting range
    - Incremental input operation confirmation
    - Prohibition of the absolute input by the soft key to prevent any improper absolute or incremental input operation
    - Confirmation of any operation of deleting the program or all data
    - Confirmation of a data update during the data setting process
2. Functions that are used when the program is executed
    - Highlighting of updated modal information
    - Display of the executed block status prior to the program execution
    - Display of the axis status, such as the mirror image function enabled or the interlock function enabled
    - Check for starting from the middle of the program
    - Data check to verify that the offset data is within the effective setting range
    - Maximum incremental value check

2.11 RETRACTION FOR RIGID TAPPING

When rigid tapping is stopped, either as a result of an emergency stop or a reset, the tap may cut into the workpiece. The tap can subsequently be drawn out by using a PMC signal. Also, the tap can be drawn out with a G30 program command, by using an appropriate parameter setting.

NOTE
For T Series, this function is an optional function.

This function automatically stores information relating to the tapping executed most recently. When a tap retraction signal is input, the tap is removed from the hole, based on the stored information. The tap is pulled toward the R point. When a retract value α is set in parameter, the retraction distance can be increased by α.



2.11.1 Retraction for Rigid Tapping by Using the G30 Command

Format

G30 P99 M29 S_ ;

M29_ : M code for specifying rigid tapping that is set in a parameter
S_ : Specify S specified for rigid tapping. (Can be omitted.)

NOTE
1. When retraction for rigid tapping by using G30 is selected, retraction for rigid tapping cannot be performed by using a signal.
2. The command for retraction for rigid tapping is a one shot command.

2.12 RETRACTION FOR 3-DIMENSIONAL RIGID TAPPING

When 3-dimensional rigid tapping or rigid tapping in tilted working plane indexing command mode is stopped by a result of a power shutdown, emergency stop, or reset, the tap may cut into the workpiece. The tap can subsequently be drawn out by using a PMC signal or a program command.
This function automatically stores information on tapping executed most recently. When a tap retraction signal is input or G30 is specified, only retraction in the rigid tapping cycle is executed, based on the stored information, and the tap is removed and pulled toward the R point.

NOTE
1. This function is an optional function.
2. For T series, to use this function, the both options for "Retraction for rigid tapping" and this function are required.

NOTE

This function is an optional function.

2.13 BUFFER REGISTER

The CNC contains a buffer register so that a pause in CNC command operation caused by program reading and preprocessing time can be suppressed.

2.14 DRY RUN

The tool is moved at a constant feedrate regardless of the feedrate specified in the program. This function is used, for example, to check the movement of the tool without a workpiece.
Dry run is valid only for automatic operation.

2.15 SINGLE BLOCK

When the single block signal is set to "1" during automatic operation, the CNC enters the automatic operation stop state after executing the current block. In subsequent automatic operation, the CNC enters the automatic operation stop state after executing each block in the program. When the single block signal is set to "0", normal automatic operation is restored.
Single block operation is valid only for automatic operation.

2.16 JOG FEED

- Jog feed

Each axis can be moved in the + or - direction for the time the button is pressed. Feedrate is the parameter set speed with override of: 0 to 655.34%, 0.01% step.
The parameter set feedrate can be set to each axis.

- Manual rapid traverse

Each axis can be fed in a rapid feed to the + or - direction for the time the button is pressed.
Rapid traverse override is also possible.

2.17 MANUAL REFERENCE POSITION RETURN

Positioning to the reference position can be done by manual operation.
With jog feed mode, manual reference position return signals, and signal for selecting manual reference position return axis on, the tool the machine is turned on, it decelerates, and when it is turned off again, it stops at the first grid point.
Upon completion of manual reference position return operation, the reference position return completion signal is sent.
By performing manual reference position return, the machine coordinate system and the workpiece coordinate system is established.
A grid method of manual reference position return is available.
The grid method can shift the reference position with the grid shift function.

2.18 MANUAL 2ND/3RD/4TH REFERENCE POSITION RETURN FUNCTION

Overview

This function enables positioning to the 2nd/3rd/4th reference position by JOG feed operation in manual reference position return mode.

Use the manual 2nd/3rd/4th reference position return select 1 signal and the manual 2nd/3rd/4th reference position return select 2 signal to select the reference position to which to return, then set the feed axis direction selection signal to “1” in manual reference position return mode to enable 2nd/3rd/4th reference position return.
The feedrate is set by rapid traverse rate parameter for each axis. (Rapid traverse override is effective.) Parameter can be used to select rapid traverse rate parameter for each axis or manual rapid traverse rate parameter .
When the tool returns to the reference position of a controlled axis, the following reference position return end signal is set to “1”.
Although one axis can be moved at the same time, using parameter allows up to three axes to move at the same time.
The 2nd, 3rd, and 4th reference positions must be set in advance in parameters with coordinates in the machine coordinate system. 2nd/3rd/4th reference position return can be used only after the relevant reference position has been established.

NOTE

This function is an optional function.

Caution

⚠ CAUTION

1. When high-speed manual reference position return is used, selecting a feed axis direction selection signal in manual reference position return mode may position the tool at the reference position irrespective of the direction specified by the feed axis direction selection signal, depending on the current position.
2. For manual 1st reference position return, be sure to set the manual 2nd/3rd/4th reference position return select 1/2 signals to “0”.

Note

NOTE

1. Once the reference position return end signal has been set to “1”, the machine cannot run again in JOG mode until it is released from the reference position return mode.
2. The reference position return end signal will be set to “0” in the following cases:
   - The tool moved from the reference position.
   - Emergency stop is applied.
   - A servo alarm is raised.
   - The servo is turned off.
3. Do not change the manual 2nd/3rd/4th reference position return select 1/2 signal during moving for reference position return. Even if the signal is changed, the tool returns to the reference position selected at the start.
4. When 1st reference position return has never be performed (the 1st reference position has not been established), returning to the 2nd/3rd/4th reference position is not executed.

NOTE

5 This function cannot be used in composite control or superimposed control mode.

6 This function cannot be used for the Cs contour control axis.

2.19 REFERENCE POSITION SETTING WITHOUT DOG

This function moves the machine to around the reference position set for each axis in the jog feed mode. Then it sets the reference position for the machine in the manual reference position return mode without the deceleration signal for reference position return. With this function, the machine reference position can be set at a given position without the deceleration signal for reference position return.
If the absolute-position detector is provided, the set reference position is retained after the power is turned off. In this case, when the power is turned on again, there is no need for setting the reference position again.

- Procedure for setting the reference position

(1) Feed the tool, along the axis for which the reference position is to be set, by jog feed in the reference position return direction. Stop the tool near the reference position, but do not exceed the reference position.

(2) Enter manual reference position return mode, then set "1" for the feed axis direction selection signal (for the positive or negative direction) for the axis.

(3) The CNC positions the tool to the nearest grid line (based on one-rotation signals from the position detector) in the reference position return direction specified with parameter. The point at which the tool is positioned becomes the reference position.

(4) The CNC checks that the tool is positioned to within the in-position area, then sets the completion signal for reference position return and the reference position establishment signal to "1".

- Manual reference position return

When the feed axis and direction selection signal is set to "1" in manual reference position return mode after the reference position has been established, the tool is positioned to the reference position regardless of the direction specified by the feed axis and direction selection signal. The completion signal for reference position return is then set to "1".

2.20 REFERENCE POINT SETTING WITH MECHANICAL STOPPER

This function automates the procedure of butting the tool against a mechanical stopper on an axis to set a reference position. The purpose of this function is to eliminate the variations in reference position setting that arise depending on the operator, and to minimize work required to make fine adjustments after reference position setting.
Select the axis for which the reference position is to be set, then perform cycle start. Then, the following operations are performed automatically:

1. The torque (force) of the selected axis is reduced to make the butting feedrate constant, and the tool is butted against the mechanical stopper. Then, the tool is withdrawn a parameter-set distance from the mechanical stopper.
2. Again, the torque (force) of the selected axis is reduced, and the tool is butted against the mechanical stopper. Then, the tool is withdrawn a parameter-set distance from the mechanical stopper.
3. The withdrawal point on the axis is set as the reference position.

2.21 REFERENCE POINT SETTING WITH MECHANICAL STOPPER BY GRID METHOD

A reference position return for an axis can be made by pushing the axis against the mechanical stopper without using a limit switch or deceleration dog. As this reference position setting uses a grid method, its precision is on the same level as for a manual reference position return. Unlike the deceleration dog, however, the mechanical stopper cannot be shifted for reference position setting. It requires using also the "reference position shift function" or "grid shift function."

NOTE

This function is an optional function.

2.22 REFERENCE POSITION RETURN FEEDRATE SETTING

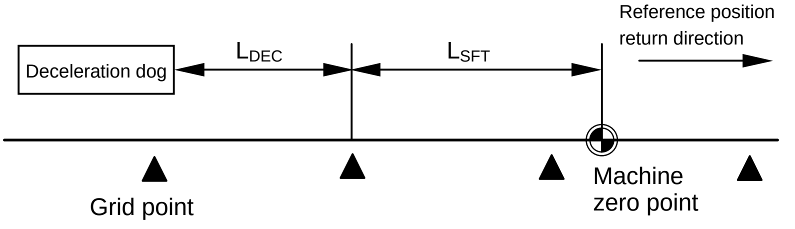
If a value is set with the parameter for reference position return feedrate setting, the manual and automatic reference position return feedrates and automatic rapid traverse rate before coordinate system establishment in the first reference position return operation after power-on follow the parameter settings for each axis.

After a reference position return operation is completed and a reference position is established, the manual reference position return feedrate set in the parameter for each axis is used.

2.23 REFERENCE POSITION SHIFT

In reference position return operation based on the grid method, the reference position can be shifted without moving the deceleration dogs, by setting a reference position shift amount in a parameter.

This function eliminates the need for adjusting the deceleration dogs at the time of reference position adjustment, so that the time required for reference position adjustment can be reduced remarkably.



L_{DEC} : Distance from the turning-off of the deceleration dog to the first grid point (grid point when the reference position shift amount is 0)

L_{SFT} : Reference position shift amount

2.24 MANUAL HANDLE FEED

2.24.1 Manual Handle Feed (1 Unit)

By rotating the manual pulse generator on the machine operator's panel in handle mode, the axis feed corresponding to the amount of rotation can be performed. A desired axis can be selected using the handle axis selection switch. Additionally, a 2nd, as well as 3rd manual pulse generator can be rotated to move the axis for the equivalent distance. Manual handle feed of 3 axes can be done at a time.

NOTE

The manual handle feed 1/2/3-units is basic function.
The manual handle feed 4/5-units is an optional function.

2.24.2 Manual Handle Feed Magnification

The manual pulse generator generates 100 pulses per rotation. Move amount per pulse can be specified from the following magnifications:
×1, ×10, ×m, ×n.
m and n are parameters set values of 1 to 2000.
Move distance is :
(Least command increment) × (magnification)
Moreover, the magnifications can be increased by a factor of 10 by parameter setting.

Table 2.24.2 (a) Travel increment in manual handle feed

Increment system	Travel increment	
	Metric input (mm)	Inch input (inch)
IS-A	0.01, 0.1, m/100, n/100	0.001, 0.01, m/1000, n/1000
IS-B	0.001, 0.01, m/1000, n/1000	0.0001, 0.001, m/10000, n/10000
IS-C	0.0001, 0.001, m/10000, n/10000	0.00001, 0.0001, m/100000, n/100000

2.25 3-DIMENSIONAL MANUAL FEED (M SERIES)

M

This function enables the use of the following functions.

- Tool axis direction handle feed/tool axis direction jog feed/tool axis direction incremental feed
- Tool axis right-angle direction handle feed/tool axis right-angle direction jog feed/tool axis right-angle direction incremental feed
- Tool tip center rotation handle feed/tool tip center rotation jog feed/tool tip center rotation incremental feed
- Table vertical direction handle feed/table vertical direction jog feed/table vertical direction incremental feed
- Table horizontal direction handle feed/table horizontal direction jog feed/table horizontal direction incremental feed

A handle interrupt can be generated for each handle feed. Handle interrupts work according to the corresponding handle feed specifications described hereinafter unless otherwise noted.

NOTE

1. This function is an optional function.
2. 3-dimensional manual feed (handle interrupt) must not be generated when a rotation axis command is being executed during automatic operation.
3. 3-dimensional manual feed is disabled when the manual reference position return mode is selected.
4. If per-axis interlock is enabled to at least one of 3-dimensional manual feed axes, movement with manual feed is not performed.

2.25.1 Tool Axis Direction Handle Feed / Tool Axis Direction Jog Feed / Tool Axis Direction Incremental Feed

This function moves the tool or table in the tool axis direction of the tool inclined by the rotation of the rotation axis, by the specified amount of travel by handle feed/jog feed/incremental feed.
As the rotation axes for controlling the tool rotate, the tool axis direction changes according to the rotation axis angle.
Through appropriate parameter settings, you can move the tool or table in the Z direction in the feature coordinate system during a tilted working plane indexing command.

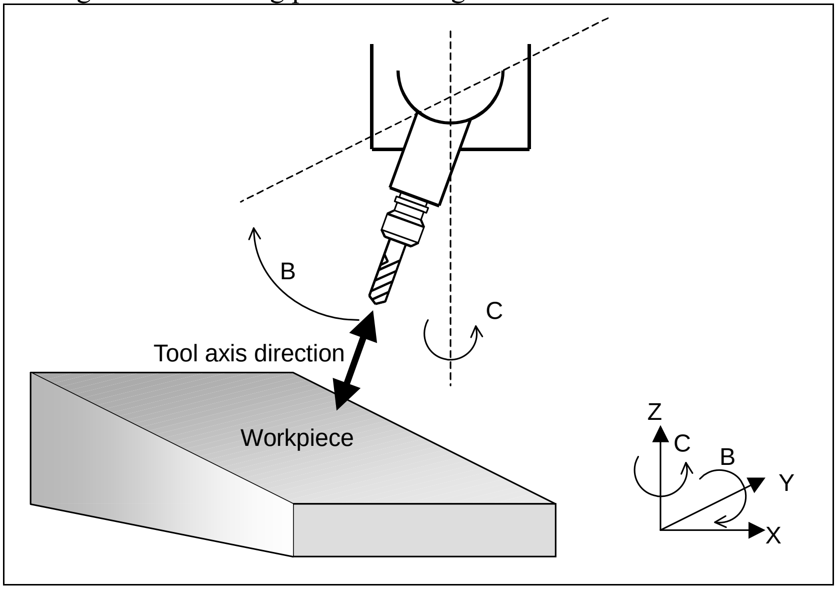


Fig. 2.25.1 (a)

2.25.2 Tool Axis Right-Angle Direction Handle Feed / Tool Axis Right-Angle Direction Jog Feed / Tool Axis Right-Angle Direction Incremental Feed

This function moves the tool or table in a specified direction perpendicular to the tool axis of the tool inclined by the rotation of the rotation axis, by the specified amount of travel by handle feed/jog feed/incremental feed.
A tool axis right-angle direction is perpendicular to the tool axis direction, and there are two tool axis right-angle directions for selection by parameter setting.
As the rotation axes for controlling the tool rotate, the tool axis right-angle direction changes according to the rotation axis angle.
Through appropriate parameter settings, you can move the tool or table in the latitudinal or longitudinal direction determined with a tool direction vector.
Also, through appropriate parameter settings, you can move the tool or table in the X or Y direction in the feature coordinate system during a tilted working plane indexing command.

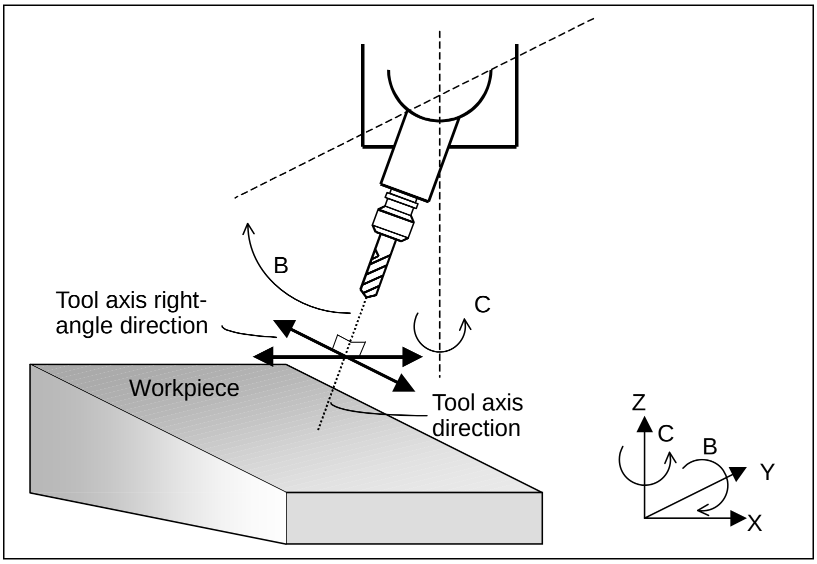


Fig. 2.25.2 (a)

2.25.3 Tool Tip Center Rotation Handle Feed / Tool Tip Center Rotation Jog Feed / Tool Tip Center Rotation Incremental Feed

This function makes a movement along the linear axes (X-axis, Y-axis, Z-axis) by the specified amount of travel by handle feed/jog feed/incremental feed so that when a rotation axis is rotated, its rotation does not change the relative relationship between the tool tip position and workpiece (table).

- The Fig. 2.25.3 (a) shows an example where the tool is rotated on the rotation axis. In this case, the linear axes are moved so that the position of the tool tip is not moved with respect to the workpiece.

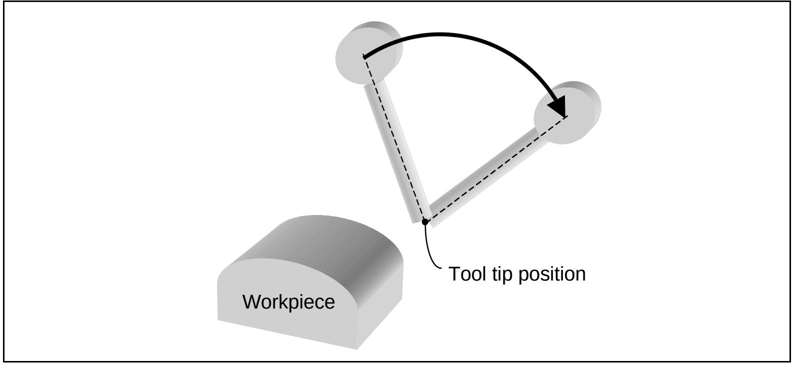


Fig. 2.25.3 (a)

- The Fig. 2.25.3 (b) shows an example where the table is rotated on the rotation axis. As in the previous case, the linear axes are moved so that the position of the tool tip is not moved with respect to the workpiece (table).

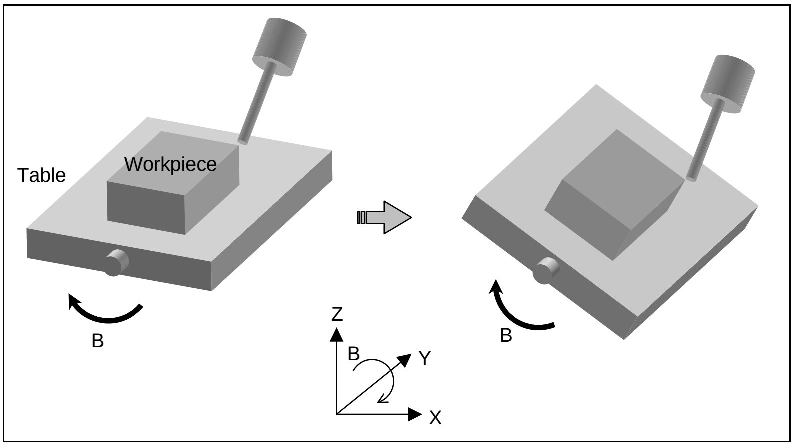


Fig. 2.25.3 (b)

2.25.4 Table Vertical Direction Handle Feed / Table Vertical Direction Jog Feed / Table Vertical Direction Incremental Feed

This function moves the tool in the table vertical direction by the specified amount of travel by handle feed/jog feed/incremental feed.
The table vertical direction is a direction vertical to the table. When the rotation axes for controlling the table rotate, the table vertical direction changes according to the rotation axis angle.
Through appropriate parameter settings, you can move the tool or table in the Z direction in the feature coordinate system during a tilted working plane indexing command.

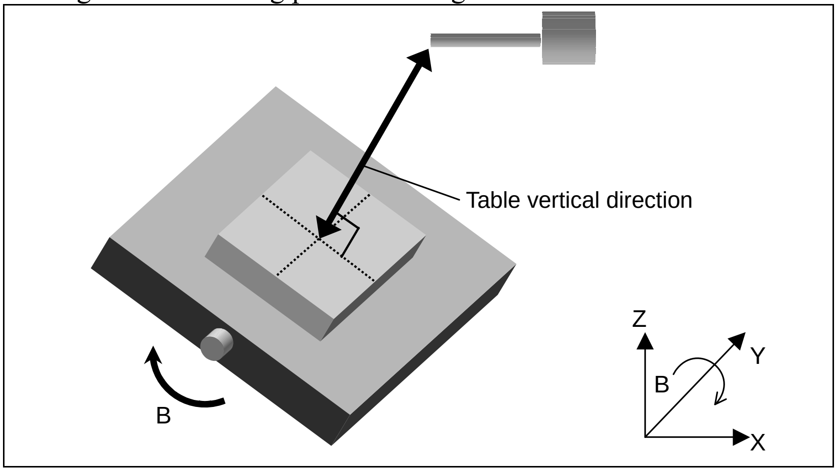


Fig. 2.25.4 (a)

2.25.5 Table Horizontal Direction Handle Feed / Table Horizontal Direction Jog Feed / Table Horizontal Direction Incremental Feed

This function moves the tool in a table horizontal direction by the specified amount of travel by handle feed/jog feed/incremental feed.
A table horizontal direction is perpendicular to the table vertical direction (mentioned in previous section), and there are two table horizontal directions for selection by parameter setting.
As the rotation axes for controlling the table rotate, the table horizontal direction changes according to the rotation axis angle.
Through appropriate parameter settings, you can move the tool or table in the latitudinal or longitudinal direction determined with a tool direction vector.
Also, through appropriate parameter settings, you can move the tool or table in the X or Y direction in the feature coordinate system during a tilted working plane indexing command.

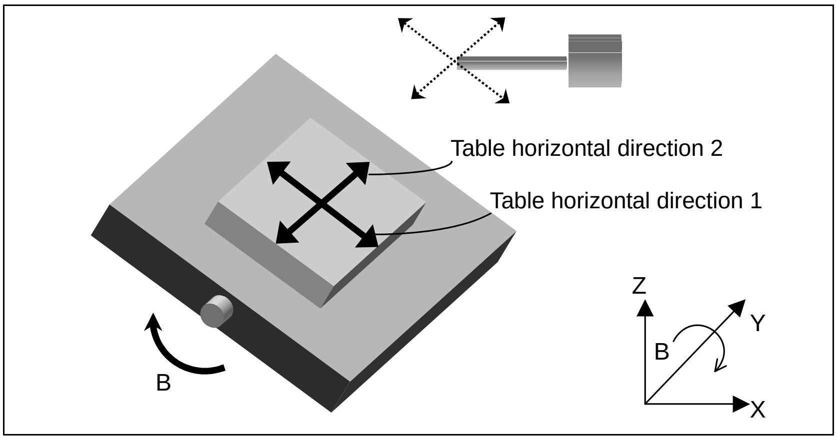


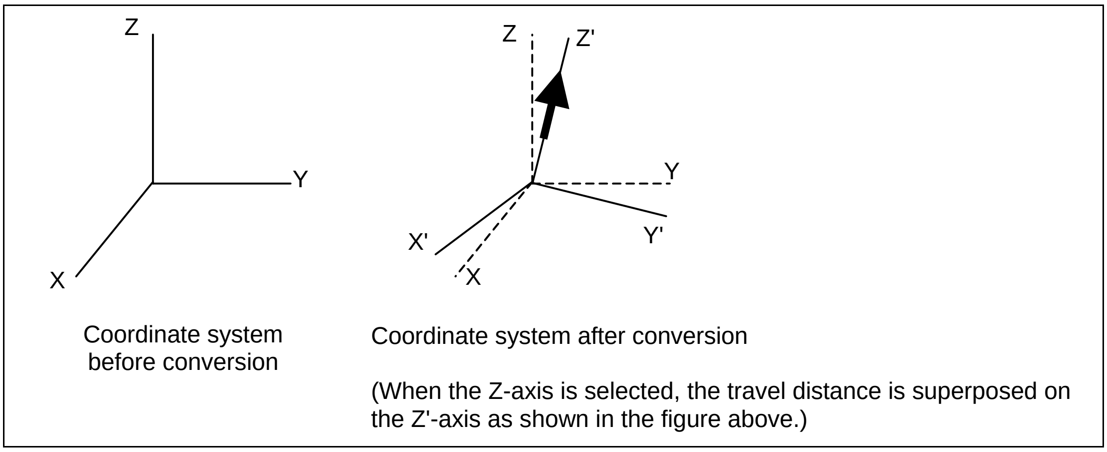
Fig. 2.25.5 (a)

2.26 MANUAL HANDLE INTERRUPTION

By rotating the manual pulse generator in the automatic operation mode (manual data input, DNC operation, or memory operation) or in the memory editing mode, handle feed can be superimposed on movement by automatic operation. A handle interruption axis is selected using the manual handle interruption axis selection signal.
The minimum unit of travel distance per scale division is the least input increment.

2.26.1 Manual Interruption of 3-dimensional Coordinate System Conversion

When the manual pulse generator is rotated in the 3-dimensional coordinate conversion mode, the travel distance specified by the manual pulse generator is superposed on the travel distance by automatic operation in the direction of the selected handle feed axis on the coordinate (program coordinate) system after 3-dimensional coordinate conversion.



(When the Z-axis is selected, the travel distance is superposed on the Z'-axis as shown in the figure above.)

Fig.2.26.1 (a)

NOTE

To use this function, the option "3-dimensional coordinate system conversion" is required.

2.27 FANUC SERVO MOTOR β Series (I/O OPTION) MANUAL HANDLE INTERFACE

This function controls manual handle feed for β servo unit by using a manual pulse generator on the NC control side. Pulses from manual pulse generator are transferred from the NC control side to β servo unit through I/O Link. Still further, this function can control the magnification of pulses by changing the parameter. This function is available on the peripheral control interface.

NOTE
This function is an optional function.

2.28 INCREMENTAL FEED

Pressing a certain button on the machine operator's panel changes a specified travel distance in the + direction or - direction. The minimum unit of travel distance is the least input increment. It can be multiplied by a magnification of 10, 100, or 1000. By parameter setting, it can be additionally multiplied by 10. The table below lists travel increments.

Table 2.28 (a) Travel increments in incremental feed

Increment system	Travel increment	
	Metric input (mm)	Inch input (inch)
IS-A	0.01, 0.1, 1.0, 10.0	0.001, 0.01, 0.1, 1.0
IS-B	0.001, 0.01, 0.1, 1.0	0.0001, 0.001, 0.01, 0.1
IS-C	0.0001, 0.001, 0.01, 0.1	0.00001, 0.0001, 0.001, 0.01

⚠ **CAUTION**
For an axis with its diameter specified in incremental feed, the travel distance is based on the diameter value.

2.29 JOG AND HANDLE SIMULTANEOUS MODE

Usually, manual handle feed is enabled only when the operation mode is set to the manual handle feed mode. By parameter setting, however, manual handle feed can be performed even in the jog feed mode.

2.30 MANUAL NUMERICAL COMMAND

The manual numerical command function allows data programmed through the MDI to be executed in jog mode. Whenever the system is ready for jog feed, a manual numerical command can be executed. The following eight functions are supported:
Movements on the axes and commands based on the M, S, T, and B functions can be disabled by parameter setting.

(1) Positioning (G00)
(2) Linear interpolation (G01)
(3) Automatic reference position return (G28)
(4) 2nd/3rd/4th reference position return (G30)
(5) M codes (auxiliary functions)
(6) S codes (spindle speed functions)
(7) T codes (tool functions)
(8) B codes (second auxiliary functions)

NOTE
This function is an optional function.

2.31 REFERENCE POSITION SIGNAL OUTPUT FUNCTION

If, after the establishment of the reference position on each axis, the coordinates in the machine coordinate system match the reference position, this function outputs a signal as the DO signal of the PMC.
Also, if the coordinates in the machine coordinate system matches the second reference position, the function outputs a signal as the DO signal of the PMC.

NOTE
This function is an optional function.

2.32 RETRACE (M SERIES)

M

The tool can retrace the path along which the tool has moved so far (reverse execution). Furthermore, the tool can move along the retraced path in the forward direction (forward reexecution). After forward reexecution is performed until the tool reaches the position at which reverse execution started, machining is continued as programmed.

NOTE
This function is an optional function.

2.33 MANUAL HANDLE RETRACE

In this function, the program can be executed both forward and backward with a manual handle (manual pulse generator) under automatic operation.
Therefore, errors of a program, interference, and so on can be checked easily by working a machine actually.

NOTE
This function is an optional function.

2.34 AUXILIARY FUNCTION OUTPUT BLOCK REVERSE MOVEMENT FOR MANUAL HANDLE RETRACE

This function enables reverse movement during manual handle retrace even if a move command and an auxiliary function (M/S/T/B code) are specified in the same block.

NOTE
This function is an optional function.

2.35 MANUAL HANDLE RETRACE FUNCTION FOR MULTI-PATH

In the manual handle retrace function for multi-path systems, when the operation of re-forward movement is performed, the movement timing of all-path movement can be made identical to that of forward movement.

NOTE

This function is an optional function.

2.36 EXTENSION OF THE MANUAL HANDLE RETRACE FUNCTION

In manual handle retrace function, the following operations that were impossible so far become available.

(1) Forward movement of rigid tapping
(2) Forward movement of threading
(3) Forward movement of PMC axis control
(4) Backward movement of the orientation by the G00 command on the Cs contour control axis
(5) Backward movement of polygon turning with two spindles
(6) Backward movement of balance cutting

NOTE

This function is included in the option "Multi-system for manual handle retrace function".

2.37 MANUAL LINEAR/CIRCULAR INTERPOLATION

In manual handle feed or jog feed, the following types of feed operations are possible along with the conventional feed operation with simultaneous single-axis control (for X, Y, Z, or other axis).

- Feed along a tilted straight line in the XY, YZ, or ZX plane based on simultaneous 2-axis control (linear feed)
- Feed along a circle in the XY, YZ, or ZX plane based on simultaneous 2-axis control (circular feed)

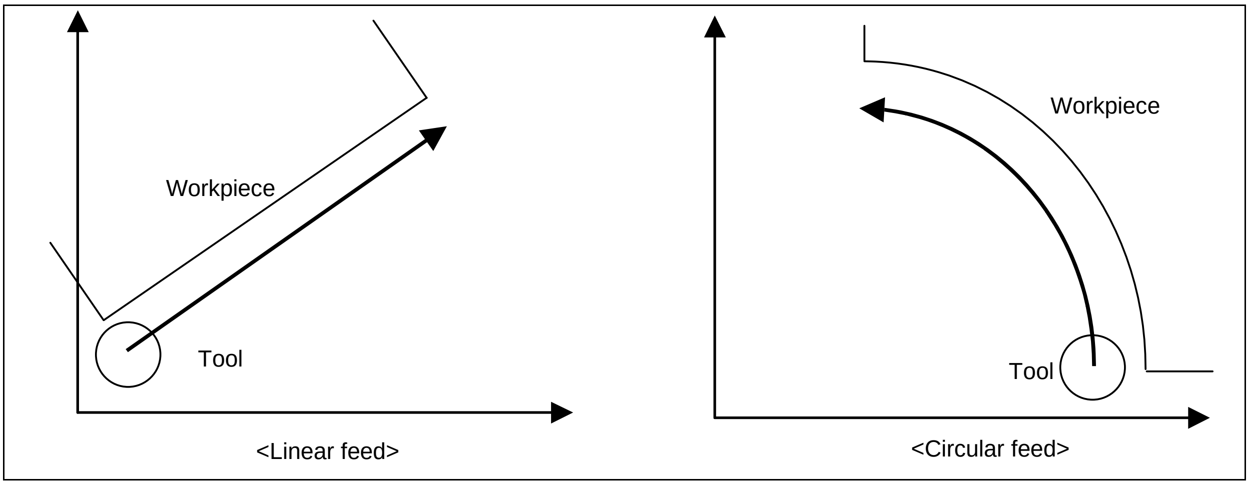


Fig.2.37 (a)

NOTE

1. This function is an optional function.
   To use the 4th or 5th manual pulse generator, the option "Manual handle feed 4/5-unit" is required.
2. Two control axes should be included in the three standard axes.

2.38 HANDLE-SYNCHRONOUS FEED

Overview

The tool is fed at a program-specified feedrate or at a feedrate that matches a dry run feedrate in cutting feed blocks (such as linear interpolation (G01) and circular interpolation (G02 and G03)) during automatic operation. This function enables the tool to be fed in synchronization with the rotation of a manual handle (manual pulse generator).

NOTE

This function is included in the option "Manual linear/circular interpolation".
To use the 4th or 5th manual pulse generator, the option "Manual handle feed 4/5-unit" is required.

The manual linear/circular interpolation signals and the usage selection of manual linear/circular interpolation signals select the manual handle with which the tool is to be synchronized.

The feedrate for handle-synchronous feed is controlled in such a way that the tangential feedrate is commensurate with the rotation speed of the manual handle.

The travel distance of the tool per manual handle pulse (travel distance amount of tangential movement) is determined by the manual handle feed travel distance selection signals.

The combination of signals determines which feedrate (program-specified feedrate (F command), a dry run feedrate, or a feedrate synchronized with the rotation of the manual handle) is to be used in a cutting feed block, as shown in the table below. These signals can be switched even in the middle of a block.

Table2.38 (a)

Dry run signal DRN	Handle-synchronous feed signal HREV	Cutting feedrate
0	0	Program-specified feedrate
0	1	Feedrate synchronized with the rotation of the manual handle
1	0	Dry run feedrate
1	1	Dry run feedrate

It is possible to make effective only one direction of rotation of a manual handle setting the parameter. When parameter is set to 1, the effective direction of the manual pulse generator is only CW (clockwise) if the handle rotation direction selection signal is set to "0". Also, the effective direction of the manual pulse generator is only CCW (counterclockwise) if the handle rotation direction selection signal is set to "1".

When parameter is set to 0, both directions of the manual pulse generator are effective as usual.

Limitations

- Unit system of the control axis

The unit system of the control axis is decided by the setting for the reference axis selected by parameter No. 1031.

- Manual mode

This function cannot be used in manual mode.

- Different magnification in each axis

Manual handle feed axis is not applied when this function is executing, it is not possible to apply a different magnification to the manual handle pulse on a per-axis basis.

Note

NOTE

The direction of manual handle rotation does not influence the direction of tool movement. That is to say, rotating the manual handle backward does not cause the tool to reverse.

Handle-synchronous feed disregards the sign of the pulse from the manual pulse generator. (The absolute values of pulses are used.) Therefore, the tool moves along a programmed path through a distance that matches the rotation number of the manual handle, regardless of the direction of rotation.

2.39 ACTIVE BLOCK CANCEL FUNCTION

This function cancels or stops a block whose execution has been started with a machining program, using a DI signal from the PMC. If the program is restarted from a stopped state, machining is resumed with the next block because the remaining amount of movement for the block of interest has been cleared. The function is intended to reduce the time required in machining program checks.

NOTE

This function is an optional function.

2.40 HIGH SPEED PROGRAM CHECK FUNCTION

The program format check and the stroke limit check are available without axes movements.

The program check is executed with the maximum feedrate of CNC system and without the acceleration/deceleration regardless of the specified data.

NOTE

This function is an optional function.

2.41 DWELL/AUXILIARY FUNCTION TIME OVERRIDE FUNCTION

This function applies override to the dwell and auxiliary(M/S/T/B) function in the range of 0% to 100% in steps of 1%.

If the override is less than 100% and applied to auxiliary(M/S/T/B) function, the next block is executed after a shortage to the actual processing time, which is considered to be 100%.

NOTE

This function is an optional function.

3 INTERPOLATION FUNCTION

Chapter 3, "INTERPOLATION FUNCTION", consists of the following sections:

3.1 NANO INTERPOLATION

As the unit of output to the servo system, the detection unit is usually used. However, this function enables output to the servo system to be performed using the detection unit multiplied by 1/1000 for improved precision in machining. This function remarkably improves surface roughness in particular.
However, positioning precision depends on machine conditions such as the resolution of the detector.

NOTE

This function is most effective if the resolution of a detector is finer than the detection unit. Even if the resolution of a detector is the same as the detection unit (as in the case of a closed loop), this function is effective when the feed-forward function is used.

3.2 POSITIONING

This command moves a tool to the position in the workpiece coordinate 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.

Either of the following tool paths can be selected according to parameter.

- Nonlinear interpolation type positioning
  The tool is positioned with the rapid traverse rate for each axis separately. The tool path is normally straight.
- Linear interpolation type positioning.
  The tool is positioned within the shortest possible time at a speed that is not more than the rapid traverse rate for each axis.

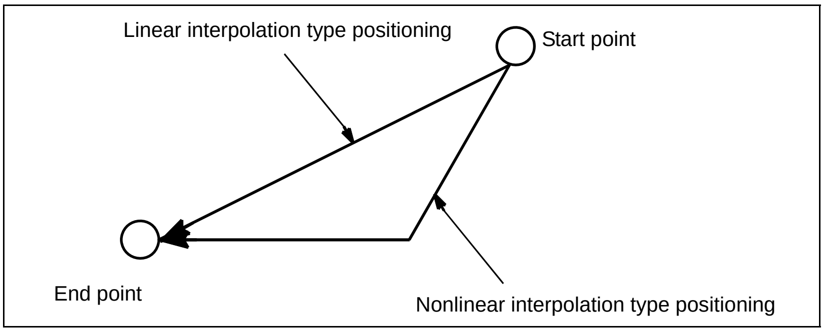


Fig.3.2 (a)

In the positioning mode based on G00, the machine is accelerated up to a specified feedrate at the start of a block. At the end of the block, the machine is decelerated to a stop to make an in-position check (to see if the machine has arrived at a specified position).
By parameter setting, an in-position check for each block can be disabled.

Format

G00 IP_ ;

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

3.3 SINGLE DIRECTION POSITIONING (M SERIES)

M

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

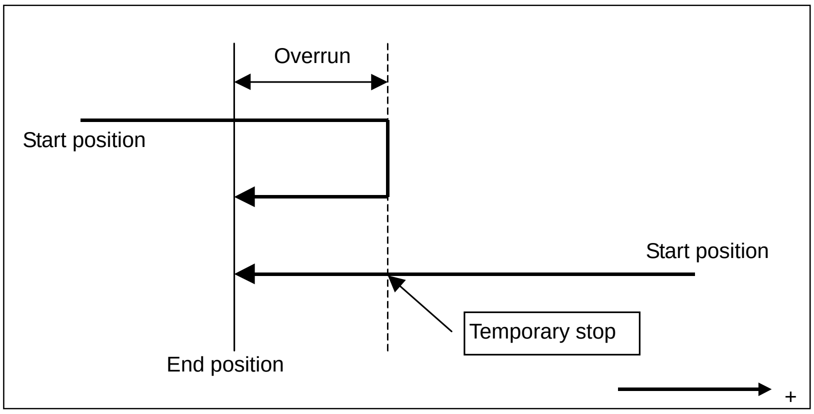


Fig.3.3 (a) Direction positioning process

An overrun and a positioning direction are set by the parameter.
Even when a commanded positioning direction coincides with that set by the parameter, the tool stops once before the end point.
G60, which is an one-shot G-code, can be used as a modal G-code in group 01 by setting the parameter.
This setting can eliminate specifying a G60 command for every block. Other specifications are the same as those for an one-shot G60 command.
When an one-shot G code is specified in the single direction positioning mode, the one-shot G command is effective like G codes in group 01.

Format

G60 IP_ ;

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

3.4 EXACT STOP MODE

When G61 is commanded, deceleration of cutting feed command at the end point and in-position check is performed per block thereafter. This G61 is valid until G62 (automatic corner override), G63 (tapping mode), or G64 (cutting mode), is commanded.

3.5 TAPPING MODE

When G63 is commanded, feedrate override is ignored (always regarded as 100%), and feed hold also becomes invalid. Cutting feed does not decelerate at the end of block to transfer to the next block. This G63 is valid until G61 (exact stop mode), G62 (automatic corner override), or G64 (cutting mode) is commanded.

3.6 CUTTING MODE

When G64 is commanded, deceleration at the end point of each block thereafter is not performed and cutting goes on to the next block. This command is valid until G61 (exact stop mode), G62 (automatic corner override), or G63 (tapping mode) is commanded. However, in G64 mode, feedrate is decelerated to zero and in-position check is performed in the following case;

- Positioning mode (G00, G60)
- Block with exact stop check (G09)
- Next block is a block without movement command

3.7 EXACT STOP

The feedrate of a block specifying continuous cutting feed can be controlled using NC commands as described below.

Move command in blocks commanded with G09 decelerates at the end point, and in–position check is performed. When the feed motor enters the effective area, the movement of the next block starts. This function is used when sharp edges are required for workpiece corners in cutting feed.

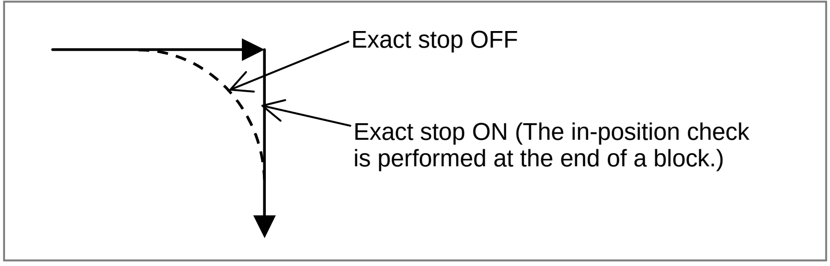


Fig.3.7 (a)

3.8 IN-POSITION CHECK SIGNAL

On general CNCs, the feedrate during cutting feed never becomes 0 between two successive blocks.
So, a corner may be rounded.

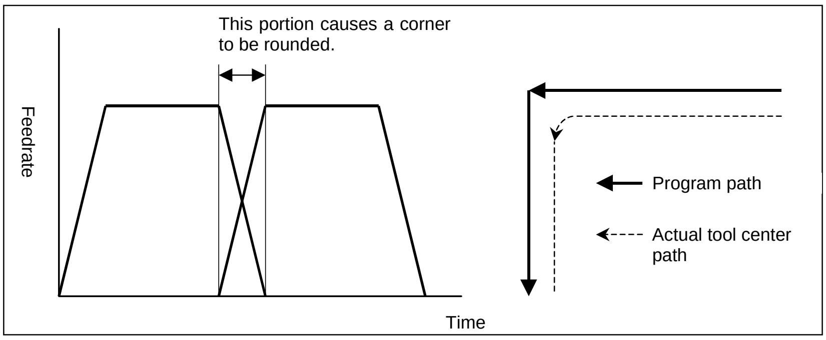


Fig.3.8 (a)

By using the in-position check signal, control can be exercised so that machining can proceed to the next block after checking that the acceleration/deceleration processing of the previous block is completed and the effective area has been entered.

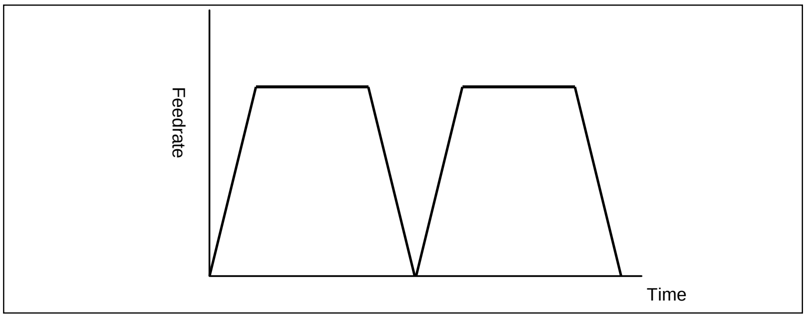


Fig.3.8 (b)

3.9 LINEAR INTERPOLATION

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.

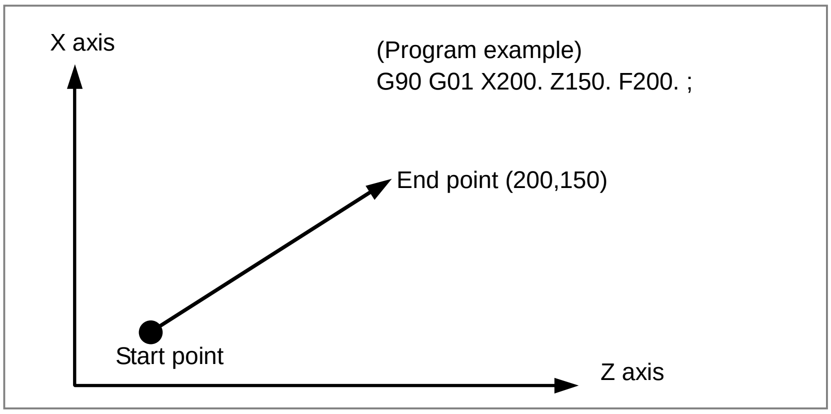


Fig.3.9 (a)

Format

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)

3.10 CIRCULAR INTERPOLATION

Circular interpolation of optional angle from 0° to 360° can be specified.
G02: Clockwise (CW) circular interpolation
G03: Counterclockwise (CCW) circular interpolation

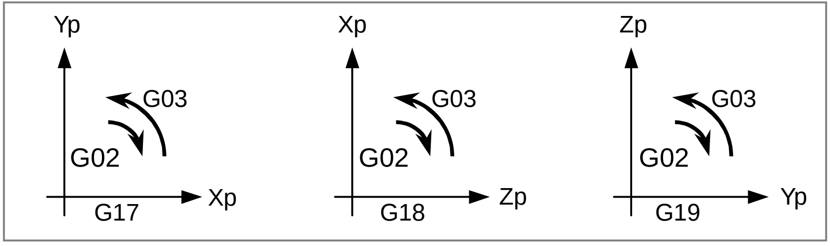


Fig.3.10 (a)

Feedrate of the tangential direction takes the speed specified by the F code. Planes to perform circular interpolation is specified by G17, G18, G19. Circular interpolation can be performed not only on the X, Y, and Z axis but also on the parallel axes of the X, Y, and Z axes.

G17: Xp-Yp plane where Xp: X axis or its parallel axis
G18: Zp-Xp plane Yp: Y axis or its parallel axis
G19: Yp-Zp plane Zp: Z axis or its parallel axis

Parameter is set to decide which parallel axis of the X, Y, Z axes to be the additional axis.

Format

Arc in the XpYp plane

$$\mathbf{G17}\left\{\begin{matrix}\mathbf{G02}\\\mathbf{G03}\end{matrix}\right\}\ \mathbf{Xp\_\ Yp\_}\left\{\begin{matrix}\mathbf{I\_\ J\_}\\\mathbf{R\_}\end{matrix}\right\}\ \mathbf{F\_\ ;}$$

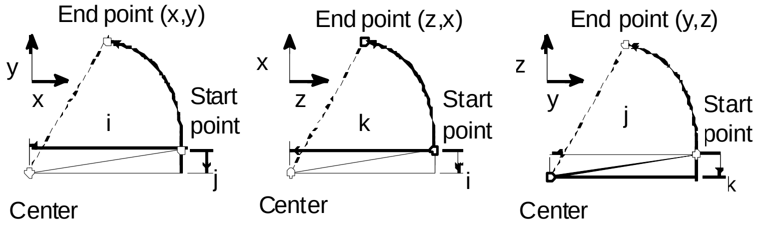
Arc in the ZpXp plane

$$\mathbf{G18}\left\{\begin{matrix}\mathbf{G02}\\\mathbf{G03}\end{matrix}\right\}\ \mathbf{Zp\_\ Xp\_}\left\{\begin{matrix}\mathbf{K\_\ I\_}\\\mathbf{R\_}\end{matrix}\right\}\ \mathbf{F\_\ ;}$$

Arc in the YpZp plane

$$\mathbf{G19}\left\{\begin{matrix}\mathbf{G02}\\\mathbf{G03}\end{matrix}\right\}\ \mathbf{Yp\_\ Zp\_}\left\{\begin{matrix}\mathbf{J\_\ K\_}\\\mathbf{R\_}\end{matrix}\right\}\ \mathbf{F\_\ ;}$$

I_, J_, K_ : Distance of the X, Y, Z axes from the start point to the center of the circle
R_ : Arc radius (For an arc having a central angle of 180° or greater, specify an R value with a minus sign. A complete circumference cannot be specified.)



3.11 DWELL

By specifying a dwell, the execution of the next block is delayed by the specified time. (Dwell per second)
By setting parameter in the feed per revolution mode, the execution of the next block is delayed until the rotation count of the spindle reaches the specified number. (Dwell per revolution)

Format

M

G04 X_; or **G04 P_ ;**

X_ : Specify a time or spindle speed (decimal point permitted)
P_ : Specify a time or spindle speed (decimal point not permitted)

T

G04 X_ ; or **G04 U_ ;** or **G04 P_ ;**

X_ : Specify a time or spindle speed (decimal point permitted)
U_ : Specify a time or spindle speed (decimal point permitted)
P_ : Specify a time or spindle speed (decimal point not permitted)

3.12 POLAR COORDINATE INTERPOLATION (T SERIES)

T

Polar coordinate interpolation is a function that exercises contour control in converting a command programmed in a Cartesian coordinate system to the movement of a linear axis (movement of a tool) and the movement of a rotary axis (rotation of a workpiece). This function is useful for grinding a cam shaft.

Format

G12.1;	Starts polar coordinate interpolation mode (enables polar coordinate interpolation)
⋮	Specify linear or circular interpolation using coordinates in a Cartesian coordinate system consisting of a linear axis and rotary axis (hypothetical axis).
G13.1 ;	Polar coordinate interpolation mode is cancelled (for not performing polar coordinate interpolation)

Specify G12.1 and G13.1 in Separate Blocks.

Explanation

- Polar coordinate interpolation mode (G12.1)

The axes of polar coordinate interpolation (linear axis and rotary axis) should be specified in advance, with corresponding parameters. Specifying G12.1 places the system in the polar coordinate interpolation mode, and selects a plane (called the polar coordinate interpolation plane) formed by one linear axis and a hypothetical axis intersecting the linear axis at right angles. The linear axis is called the first axis of the plane, and the hypothetical axis is called the second axis of the plane. Polar coordinate interpolation is performed in this plane.

In the polar coordinate interpolation mode, both linear interpolation and circular interpolation can be specified by absolute or incremental programming.

Cutter compensation can also be performed. The polar coordinate interpolation is performed for a path obtained after cutter compensation.

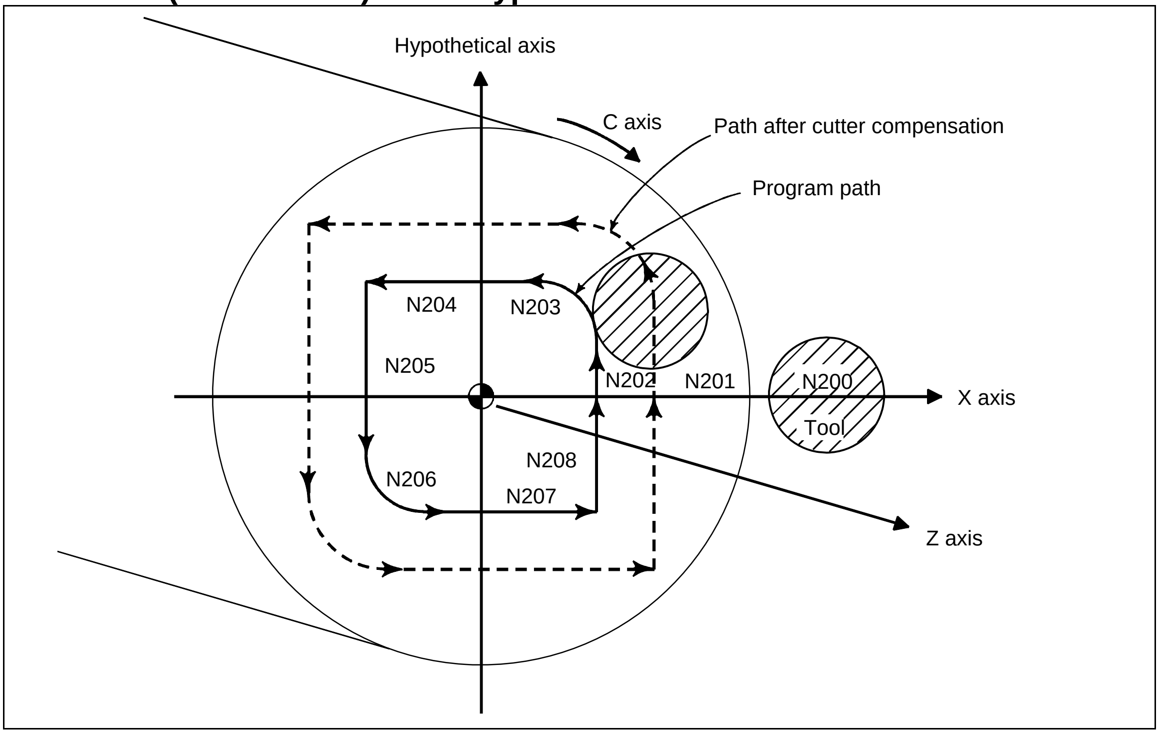
The tangential velocity in the polar coordinate interpolation plane (Cartesian coordinate system) is specified as the feedrate, using F.

- Polar coordinate interpolation cancel mode (G13.1)

Specifying G13.1 cancels the polar coordinate interpolation mode.

Example

- Polar coordinate interpolation in a Cartesian coordinate system consisting of the X axis (a linear axis) and a hypothetical axis



```
O001 ;
   .
N0010 T0101
   .
N0100 G90 G00 X60.0 C0 Z_ ;            Positioning to start point
N0200 G12.1 ;                          Start of polar coordinate interpolation
N0201 G42 G01 X20.0F_ ;              ┐
N0202 C10.0 ;                        │
N0203 G03 X10.0 C20.0 R10.0 ;        │
N0204 G01 X-20.0 ;                   │  Geometry program
N0205 C-10.0 ;                       │
N0206 G03 X-10.0 C-20.0 I10.0 J0 ;   ├─ (program based on cartesian coordinates on
N0207 G01 X20.0 ;                    │  X axis - hypothetical axis plane)
N0208 C0 ;                           │
N0209 G40 X60.0 ;                    │
N0210 G13.1 ;                        ┘  Cancellation of polar coordinate interpolation
N0300 Z_ ;
N0400 X_ C_ ;
   .
N0900M30 ;
```

3.13 CYLINDRICAL INTERPOLATION

3.13.1 Cylindrical Interpolation

In cylindrical interpolation, the amount of movement of a rotary axis specified by angle is converted to the amount of movement on the circumference to allow linear interpolation and circular interpolation with another axis. Since programming is enabled with the cylinder side face expanded, programs such as a program for grooving cylindrical cams can be created very easily.

Format

G07.1 IP r ; Starts the cylindrical interpolation mode (enables cylindrical interpolation).

:

G07.1 IP 0 ; The cylindrical interpolation mode is cancelled.

IP : An address for the rotary axis

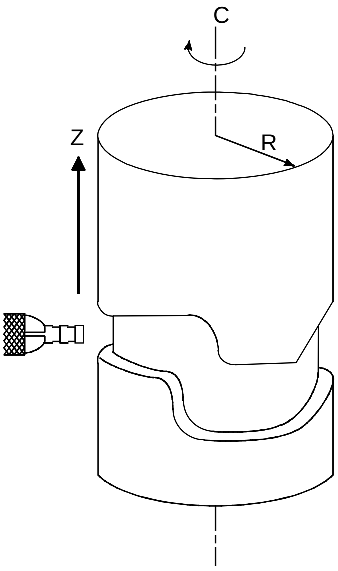
r : The radius of the workpiece

Specify G07.1 IPr ; and G07.1 IP0; in separate blocks.

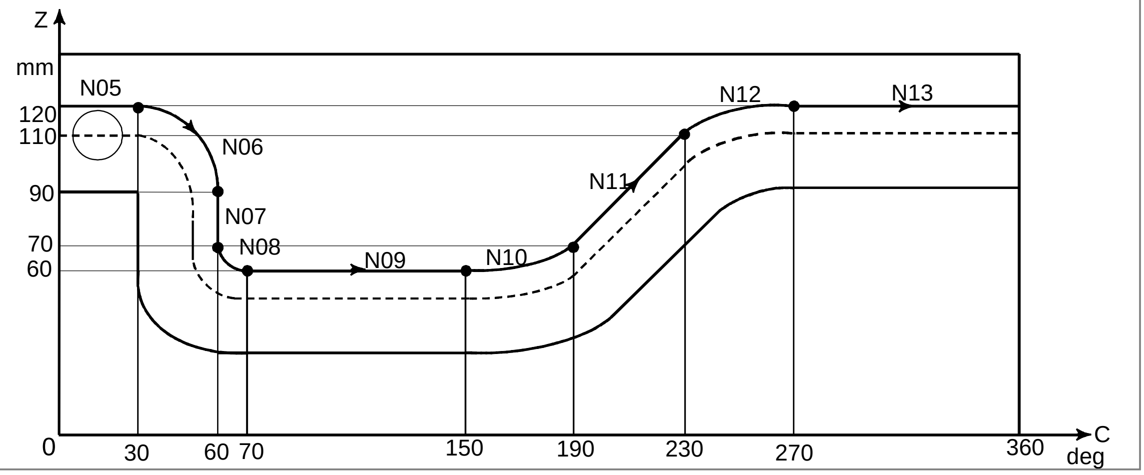
Example

Example of a cylindrical interpolation programs

```
O0001 (CYLINDRICAL INTERPOLATION);
N01 G00 G90 Z100.0 C0 ;
N02 G01 G91 G18 Z0 C0 ;
N03 G07.1 C57299 ;
N04 G90 G01 G42 Z120.0 D01 F250 ;
N05 C30.0 ;
N06 G03 Z90.0 C60.0 R30.0 ;
N07 G01 Z70.0 ;
N08 G02 Z60.0 C70.0 R10.0 ;
N09 G01 C150.0 ;
N10 G02 Z70.0 C190.0 R75.0 ;
N11 G01 Z110.0 C230.0 ;
N12 G03 Z120.0 C270.0 R75.0 ;
N13 G01 C360.0 ;
N14 G40 Z100.0 ;
N15 G07.1 C0 ;
N16 M30 ;
```



Note) Sample program where the C axis is parallel to the X-axis.



3.13.2 Cylindrical Interpolation by Plane Distance Command

Overview

In the conventional rotary axis command in cylindrical interpolation, the angle of the rotary axis is specified.

This function enables the rotary axis command in cylindrical interpolation to be specified by distance on the developed plane by setting parameters.

> **NOTE**
> This function is an optional function.

Format

G07.1 IP r ; Starts the cylindrical interpolation mode (enables cylindrical interpolation).

:

G07.1 IP 0 ; The cylindrical interpolation mode is cancelled.

IP : An address for the rotary axis

r : The radius of the workpiece

Specify G07.1 IPr ; and G07.1 IP0; in separate blocks.

Explanation

By using bit 2 (DTO) of parameter No. 3454, it is possible to switch the rotation axis command during cylindrical interpolation between the angle of the rotation axis and the distance on the developed plane.

- In the case of the angle of the rotation axis
  The rotation axis command in cylindrical interpolation mode is executed with the angle of the rotation axis. From the program, specify the angle of the rotation axis that corresponds to the specified point on the developed plane.
  The rotation axis command uses the angle of the rotation axis [deg].

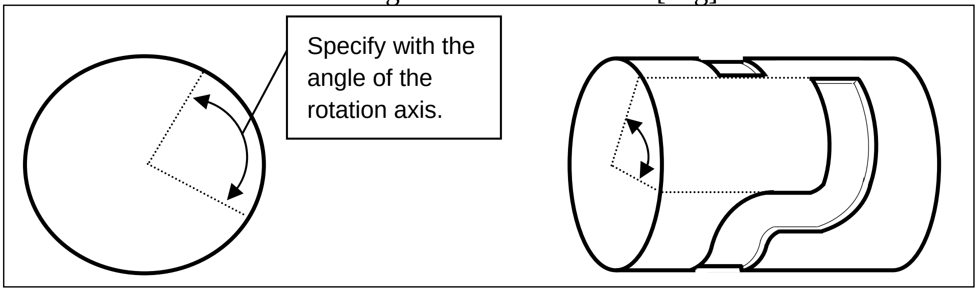


Fig.3.13 (a)

- In the case of the distance on the developed plane
  The rotation axis command in cylindrical interpolation is executed with the distance on the developed plane. The rotation axis command uses the distance on the developed plane and, therefore, the command unit varies depending on which of inch or metric input to use.

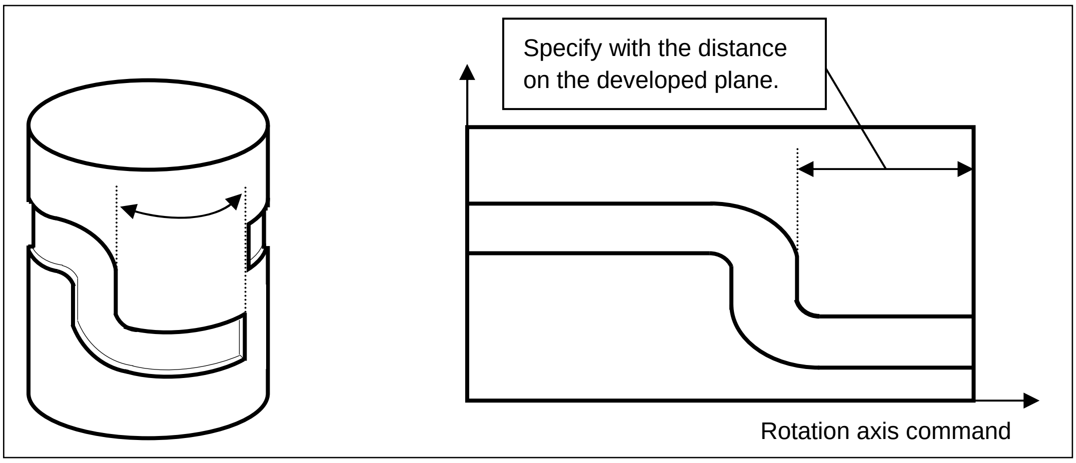


Fig.3.13 (b)

3.14 HELICAL INTERPOLATION

Helical interpolation which moved helically is enabled by specifying up to two other axes which move synchronously with the circular interpolation by circular commands.
A tangential velocity of an arc in a specified plane or a tangential velocity about the linear axis can be specified as the feedrate, depending on the setting of parameter.

NOTE

The specification of this function depends on the model.
- M series : basic function
- T series : optional function.

- When a feedrate along an arc is specified

Therefore, the feedrate of the linear axis is as follows:

$$F \times \frac{\text{Length of linear axis}}{\text{Length of circular arc}}$$

Determine the feedrate so the linear axis feedrate does not exceed any of the various limit values.

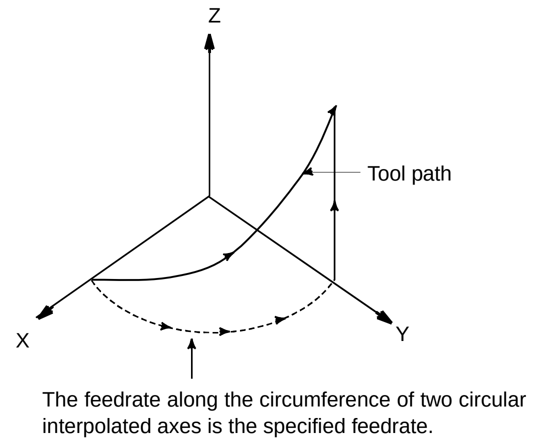


Fig.3.14 (a) The feedrate along an arc

- When a feedrate along the tool path including a linear axis is specified

Therefore, the tangential velocity of the arc is expressed as follows:

$$F \times \frac{\text{Length of arc}}{\sqrt{(\text{Length of arc})^2 + (\text{Length of linear axis})^2}}$$

The velocity along the linear axis is expressed as follows:

$$F \times \frac{\text{Length of linear axis}}{\sqrt{(\text{Length of arc})^2 + (\text{Length of linear axis})^2}}$$

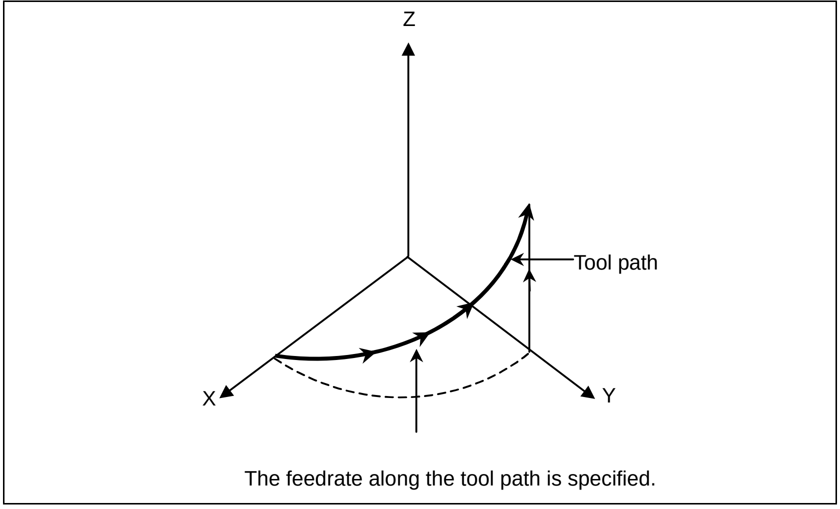


Fig.3.14 (b) The feedrate along the tool path

Format

Arc of XpYp plane

$$\mathbf{G17}\left\{\begin{matrix}\mathbf{G02}\\ \mathbf{G03}\end{matrix}\right\}\ \mathbf{Xp\_\ Yp\_}\left\{\begin{matrix}\mathbf{I\_\ J\_}\\ \mathbf{R\_}\end{matrix}\right\}\ \alpha\_\ (\beta\_)\ \mathbf{F\_}\ ;$$

Arc of ZpXp plane

$$\mathbf{G18}\left\{\begin{matrix}\mathbf{G02}\\ \mathbf{G03}\end{matrix}\right\}\ \mathbf{Zp\_\ Xp\_}\left\{\begin{matrix}\mathbf{K\_\ I\_}\\ \mathbf{R\_}\end{matrix}\right\}\ \alpha\_\ (\beta\_)\ \mathbf{F\_}\ ;$$

Arc of YpZp plane

$$\mathbf{G19}\left\{\begin{matrix}\mathbf{G02}\\ \mathbf{G03}\end{matrix}\right\}\ \mathbf{Yp\_\ Zp\_}\left\{\begin{matrix}\mathbf{J\_\ K\_}\\ \mathbf{R\_}\end{matrix}\right\}\ \alpha\_\ (\beta\_)\ \mathbf{F\_}\ ;$$

α, β : Any one axis where circular interpolation is not applied. Up to two other axes can be specified.

3.15 NANO SMOOTHING (M SERIES)

M

When a desired sculptured surface is approximated by minute segments, the nano smoothing function generates a smooth curve inferred from the programmed segments and performs necessary interpolation.
The nano smoothing function infers a curve from a programmed figure approximated with segments within tolerance. If the spacing between adjacent inflection points or programmed points is not constant, this function can generate a smoother curve.
The interpolation of the curve reduces the segment approximation error, and the nano interpolation makes the cutting surface smoother.

NOTE
This function is an optional function.

Format

G5.1 Q3 Xp0 Yp0 Zp0 [α0] [β0] ; : Nano smoothing mode on
G5.1 Q0 ; : Nano smoothing mode off

Xp : X-axis or an axis parallel to the X-axis
Yp : Y-axis or an axis parallel to the Y-axis
Zp : Z-axis or an axis parallel to the Z-axis
α, β : Rotary axis

NOTE
1 Specify G5.1 alone in a block. (Avoid specifying any other G code in the same block.)
2 Specify position 0 for the axis programmed in the nano smoothing mode on block. The specified axis is subjected to nano smoothing, but no movement is made even in the absolute programming mode.

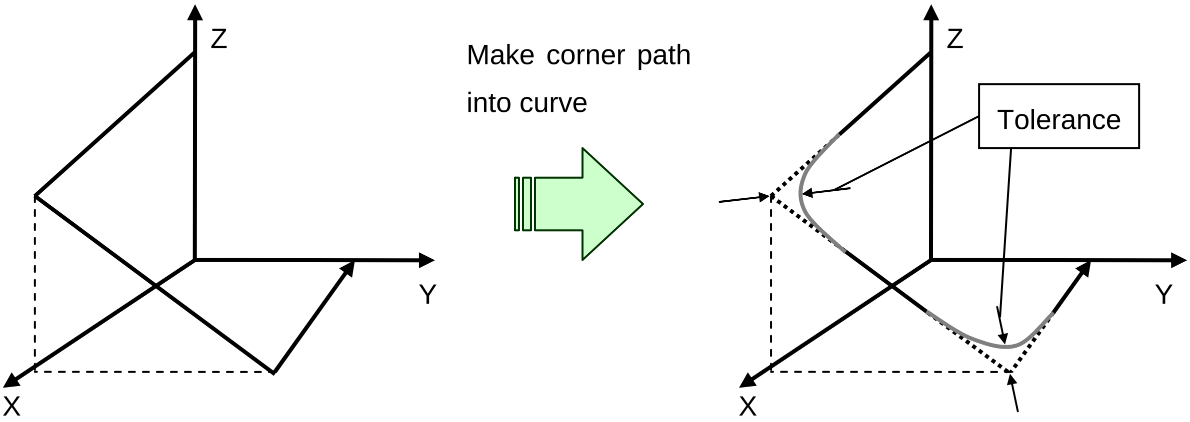
3.16 SMART TOLERANCE CONTROL (M SERIES)

M

Two functions that generate smooth machining path within specified tolerance, and realize high-speed and high-precision machining are introduced as follows.

Making corner path into curve

In the conventional AI contour control, direction and curvature of specified path are discontinuous at each joint of linear interpolation blocks and circular interpolation blocks. This function makes direction and curvature of corner paths continuous, and corner paths are made into curves so that the precision at each joint of linear interpolation blocks and circular interpolation blocks is within the tolerance specified parameter No. 19596 or G code.
Therefore, setting the machining precision gets easier and high-precision machining is possible regardless of feedrate.



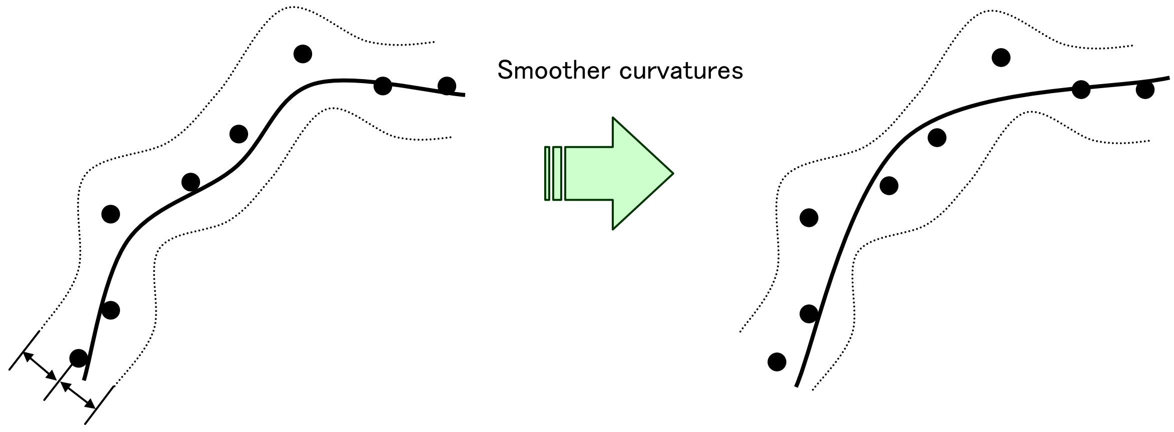
Specified path

Path when tolerance control is effective

Fig.3.16 (a) Making corner path into curve

Smoothing small line segments

This function makes a path defined by small line segments into a curvature, furthermore it makes smooth the connection of curvatures.



Nano smoothing

Tolerance control

Fig.3.16 (b) Smoothing small line segments

Format

G05.1 Q3 Xp0 Yp0 Zp0; **Smart tolerance control mode on**

G10.8 L4 I_ Q_; **Specify tolerance**

:

G05.1 Q0; **Smart tolerance control mode off**

Xp: X-axis or an axis parallel to the X-axis

Yp: Y-axis or an axis parallel to the Y-axis

Zp: Z-axis or an axis parallel to the Z-axis

I: Tolerance for linear axis at corners

Q: Tolerance for linear axis on curves

NOTE

1. Specify G05.1 alone in a block.
   (Avoid specifying any other G code in the same block)
2. Bit 0 (ATC) of parameter No. 19594 must be set to 1 when using this function.
3. Unit of "I" in G10.8 command depends on the increment system of the basic axis.
4. The value of parameter No. 19596 and No. 19597 are effective as tolerance between G05.1 Q3 command and G10.8 L4 command.
5. Smart tolerance control is disabled when the tolerance for linear axis is set to 0.
6. Smart tolerance control mode is activated at the start of automatic operation by setting bit 0 (CAT) of parameter No. 11785.

3.17 THREAD CUTTING, SYNCHRONOUS CUTTING

By feeding the tool synchronizing with the spindle rotation, threading of the specified lead is performed. In addition to straight threads, taper threads and scroll threads can be cut with equal leads.

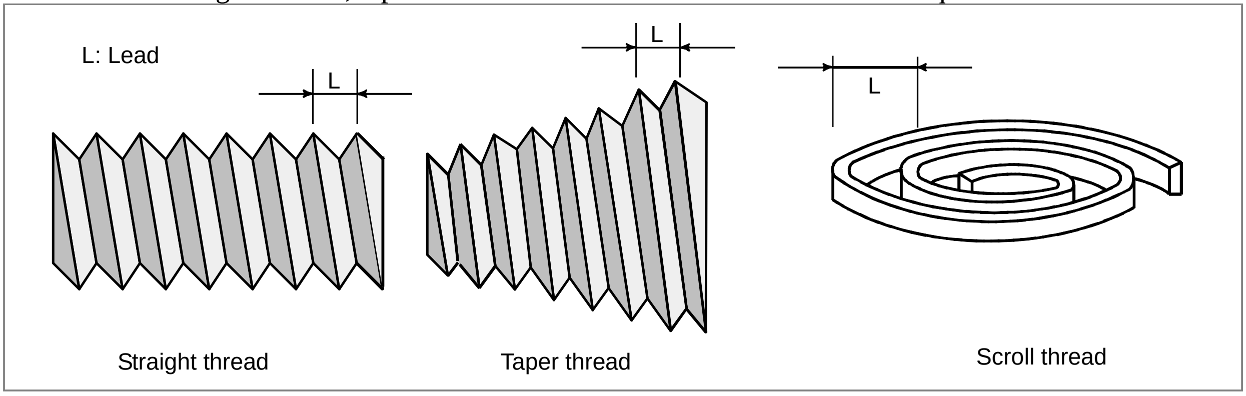


Fig.3.17 (a)

Format

G33 IP_ F_ ;

F_ : Lead along the long axis
(G32 when G code system A with lathe system)

Explanation

To form a single thread, threading is generally performed several times from rough machining to finish machining along the same path.

Threading starts in synchronism with a one-rotation signal from the position coder attached to the spindle. So, the tool path and cutting start point on the periphery of a workpiece remain unchanged, regardless of the number of threading operations performed. In this case, however, the shaft must rotate at a constant speed during operations from rough machining to finish machining. If the spindle speed changes, an accurate thread may not be produced.

The following shows the specifiable lead range:

Table3.17 (a)

	Least command increment		Specifiable lead range	
Metric input	0.01	mm	0.001 to 5000.0000	mm/rev
	0.001	mm	0.00001 to 500.00000	mm/rev
	0.0001	mm	0.000001 to 50.000000	mm/rev
Inch input	0.001	inch	0.00001 to 500.00000	inch/rev
	0.0001	inch	0.000001 to 50.000000	inch/rev
	0.00001	inch	0.0000001 to 5.0000000	inch/rev

NOTE

Leads exceeding the maximum cutting feed speed when converted to per minute feed speed cannot be specified.

3.18 MULTIPLE THREADING (T SERIES)

T

Using the Q address to specify an angle between the one-spindle-rotation signal and the start of threading shifts the threading start angle, making it possible to produce multiple-thread screws with ease.

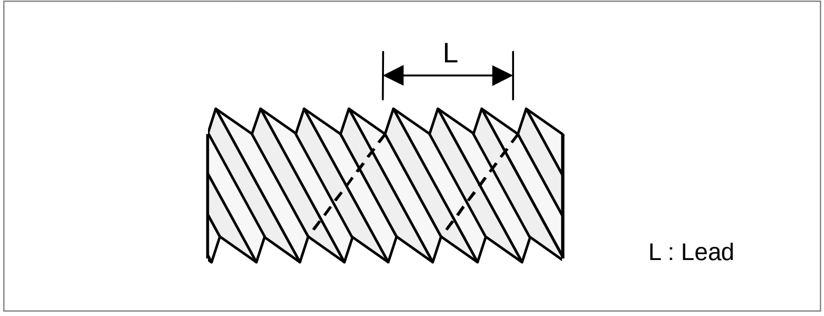


Fig.3.18 (a)

Format

(Constant lead threading)

G32 IP _ F_ Q_ ;

IP : End point

F_ : Lead in longitudinal direction

G32 IP _ Q_ ;

Q_ : Threading start angle

Explanation

- Available threading commands

G32 : Constant lead threading
G34 : Variable lead threading
G76 : Combined threading cycle
G92 : Threading cycle

3.19 THREADING RETRACT (T SERIES)

3.19.1 Threading Retract (Canned Cycle) (T Series)

T

Feed hold may be applied during threading. In this case, the tool immediately retracts with chamfering and returns to the start point on the second axis (X-axis), then the first axis (Z-axis) on the plane.

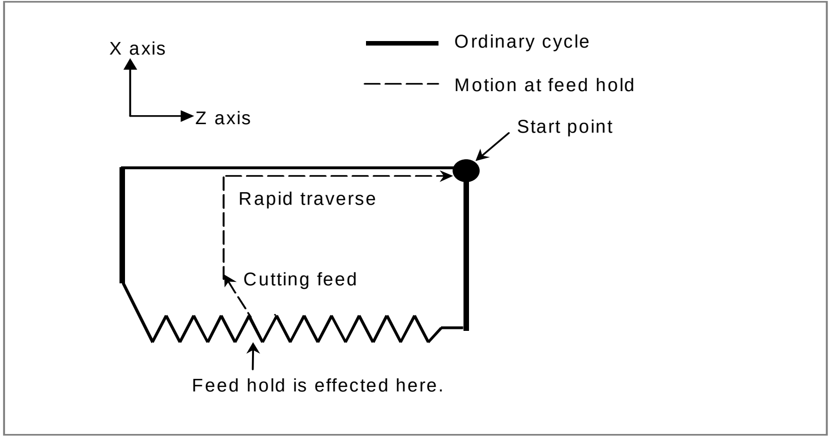


Fig.3.19.1 (a)

The amount of chamfering during retraction is the same as that of chamfering at the end point.

3.19.2 Threading Retract (Multiple Repetitive Cycle) (T Series)

T

If feed hold is applied during threading in the multiple repetitive threading cycle (G76), chamfering for threading is performed and the tool returns to the threading cycle start point and stops.
If a cycle start operation is performed here, machining restarts with the threading cycle to which feed hold was applied.

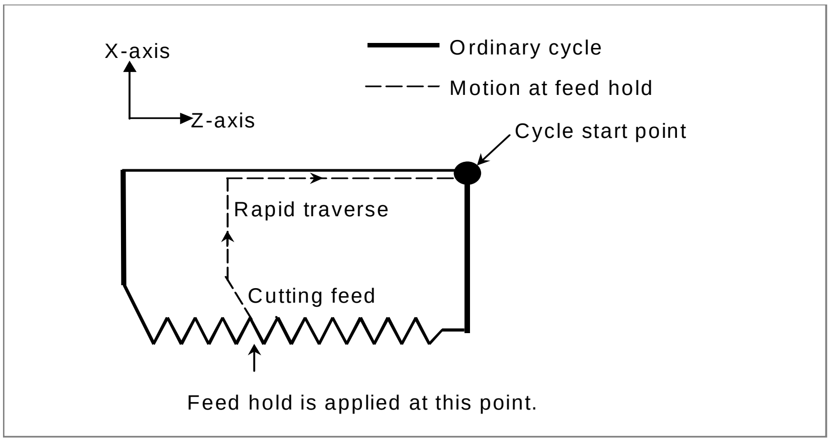


Fig.3.19.2 (a)

The amount of chamfering during retraction is the same as that of chamfering at the end point.

3.20 CONTINUOUS THREADING (T SERIES)

T

Threading blocks can be programmed successively to eliminate a discontinuity due to a discontinuous movement in machining by adjacent blocks.

Fig. 3.20 (a) Image of continuous threading

3.21 VARIABLE LEAD THREADING (T SERIES)

T

Specifying an increment or a decrement value for a lead per screw revolution enables variable lead threading to be performed.

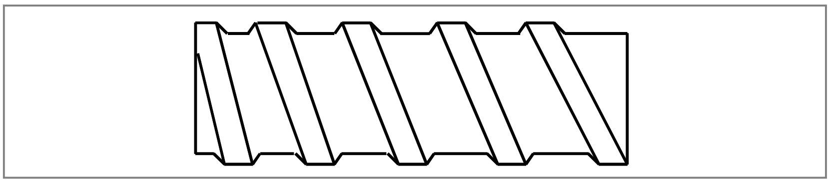


Fig. 3.21 (a) Variable lead screw

Format

G34 IP_ F_ K_ Q_ ;

IP_ : End point
F_ : Lead in longitudinal axis direction at the start point
K_ : Increment and decrement of lead per spindle revolution
Q_ : Shift amount of starting angle of thread cutting

Explanation

Address other than K are the same as in straight/taper threading with G32.

The K value depends on the increment system of the reference axis, as indicated in following table.

Table3.21 (a)

increment system of the reference axis	Metric input (mm/rev)	Inch input(inch/rev)
IS-A	±0.001 to ±500.000	±0.00001 to ±50.00000
IS-B	±0.0001 to ±500.0000	±0.000001 to ±50.000000
IS-C	±0.00001 to ±50.00000	±0.0000001 to ±5.0000000

3.22 CIRCULAR THREAD CUTTING (T SERIES)

T

Using the G35 and G36 commands, a circular thread, having the specified lead in the direction of the major axis, can be machined.

NOTE
This function is an optional function.

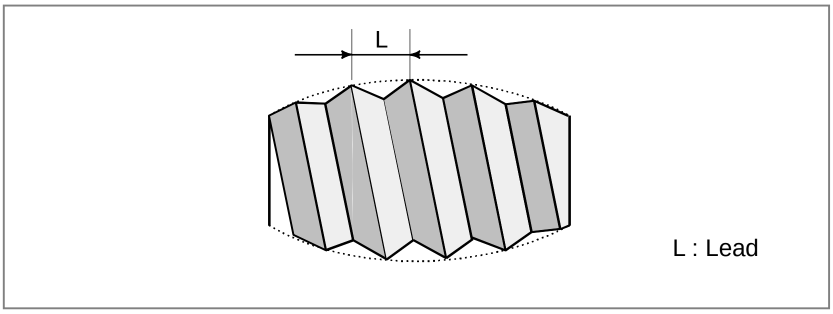


Fig. 3.22 (a) Example of circular thread cutting

Format

A sample format for the G18 plane (Z-X plane) is indicated below. When using the format for the G17 plane (X-Y plane), change the addresses Z, X, K, and I to X, Y, I, and J respectively. When using the format for the G19 plane (Y-Z plane), change the addresses Z, X, K, and I to Y, Z, J, and K respectively.

$$\left\{\begin{matrix}\mathbf{G35}\\ \mathbf{G36}\end{matrix}\right\}\ \mathbf{X(U)\_\ Z(W)\ \_}\ \left\{\begin{matrix}\mathbf{I\_\ K\_}\\ \mathbf{R\_}\end{matrix}\right\}\ \mathbf{F\_\ Q\_\ ;}$$

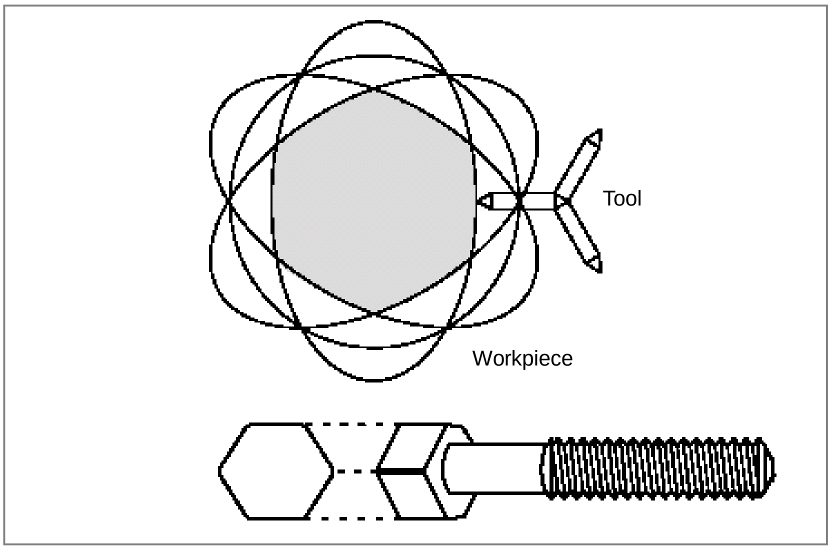
G35 : Clockwise circular thread cutting command
G36 : Counterclockwise circular thread cutting command

X(U), Z(W) : Specify the arc end point (in the same way as for G02, G03).
I,K : Specify the arc center relative to the start point, using relative coordinates (in the same way as for G02, G03).
R : Specify the arc radius.
F : Specify the lead in the direction of the major axis.
Q : Specify the shift of the thread cutting start angle (0° to 360°, with least input increment of 0.001) (The value cannot be programmed with a decimal point.)

3.23 POLYGON TURNING (T SERIES)

T

Polygon turning means machining a workpiece to a polygonal figure by rotating the workpiece and tool at a certain ratio.



By changing conditions which are rotation ratio of workpiece and tool and number of cutters, the workpiece can be machined to a square or hexagon. The machining time can be reduced as compared with polygonal figure machining using the polar coordinate interpolation. The machined figure, however, is not exactly polygonal. Generally, polygon turning is used for the heads of square and/or hexagon bolts or hexagon nuts.

Format

G51.2 P_ Q_ ;

P,Q : Rotation ratio of spindle and Y-axis
Specify range:
P : Integer from 1 to 999
Q : Integer from -999 to -1 or from 1 to 999
When Q is a positive value, Y-axis makes positive rotation.
When Q is a negative value, Y-axis makes negative rotation.

G50.2 ; Polygon turning cancel

NOTE

Specify G50.2 and G51.2 in a single block.

Explanation

A CNC controlled axis (servo axis) is assigned to the tool rotation axis.

This rotary axis of tool is called Y-axis in the following description. The Y-axis is controlled by the G51.2 command, so that the ratio of the rotation speeds of the spindle (previously specified by S-command) and the tool becomes the specified ratio.

When simultaneous start is specified by G51.2, the one-rotation signal sent from the position codes set on the spindle is detected. After one-rotation signal detection, the Y-axis is controlled using the rotation ratio of the spindle and Y-axis specified by P and Q. So, a position coder needs to be attached to the spindle. This control will be maintained until the polygon turning cancel command is executed (G50.2).

3.24 POLYGON TURNING WITH TWO SPINDLES (T SERIES)

T

When two or more serial spindles are used, the workpiece rotation axis (master axis) and tool rotation axis (polygon synchronization axis) are synchronized at a certain speed ratio.

With this function, it is also possible to specify the phase difference between the master and polygon synchronization axes.

The polygonal turning with two spindles can use different spindle speeds for the same workpiece, because it performs automatic phase compensation when a polygon synchronization mode command is issued or the S command is changed during polygon synchronization mode.

With a multi-path system, polygonal turning is possible on each path.

By default, the first and second spindles in each system are selected as the master axis and polygon synchronization axis for each system. By setting parameters, however, any spindles belonging to the same system or different systems can be selected as the master axis and polygon synchronization axis.

Format

This is the same as the program command format for polygon turning except for the following points.

1) The command position (R) can be used.
2) Repeated specification in polygon synchronization mode is allowed.

G51.2 Pp Qq Rr ; Polygon synchronization mode start

The command above starts the polygon synchronization mode or modifies specified values (P, Q, R) in the polygon synchronization mode.

P : Master axis rotation ratio
Specifiable range: Integer from 1 to 999
(The rotation direction of the master axis is determined by an ordinary command [such as M03/M04] used for spindle control.)

Q : Polygon synchronization axis rotation ratio
Specifiable range: Integer from 1 to 999 or from -1 to -999
(The rotation direction of a polygon synchronization axis is determined by the sign of a Q command value. Depending on the parameter, however, the rotation direction of the first axis is followed. In this case, no negative value may be specified for Q.)

R : Specification of phase between the master axis and polygon synchronization axis (Specify a relative value.)
Valid specifiable range: 0 ≤ phase angle < 360 deg
The increment system is the same as for the reference axis.
However, the least command increment is 360/4096 (deg), and a specified value smaller than the least command increment is rounded for use.
(R is omissible. If R is not specified even once after the start of the polygon synchronization mode, phase 0 (deg) is used.
If phase control is disabled by parameter setting, the specification of R is ignored. No particular alarm is issued.)

G50.2 ; Polygon turning cancel

3.25 SKIP FUNCTION

3.25.1 Skip Function

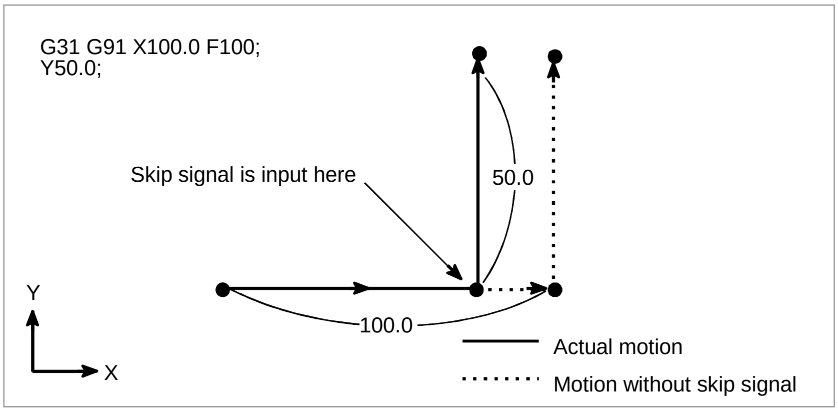
Linear interpolation can be commanded by specifying axial move following the G31 command, like G01. If an external skip signal is input during the execution of this command, execution of the command is interrupted and the next block is executed.

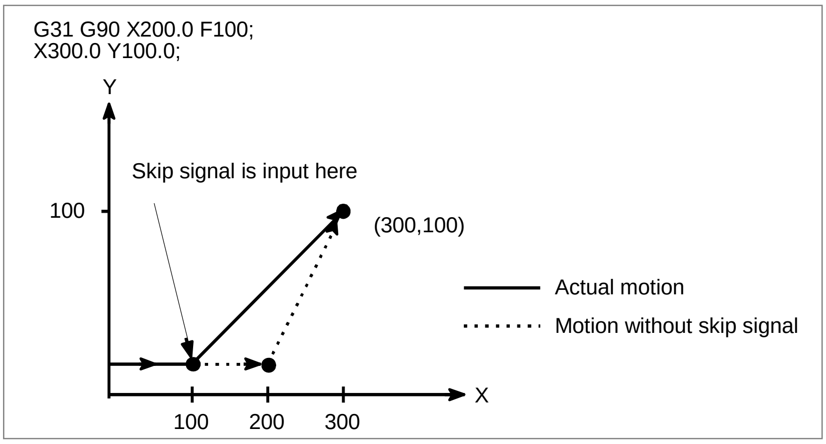
The skip function is used when the end of machining is not programmed but specified with a signal from the machine, for example, in grinding. It is used also for measuring the dimensions of a workpiece.

Format

G31 IP_ ;

G31 : One-shot G code (If is effective only in the block in which it is specified)





3.25.2 Multi-step Skip

In a block specifying P1 to P4 after G31, the multi-step skip function stores coordinates in a custom macro variable when a skip signal (4-point or 8-point ; 4-point when a high-speed skip signal is used) is turned on. In the block where Q1 to Q4 are specified after G04, dwell can be skipped when skip signals (4-point or 8-point ; 4-point when a high-speed skip signal is used) are input.

A skip signal from equipment such as a fixed-dimension size measuring instrument can be used to skip programs being executed.

In plunge grinding, for example, a series of operations from rough machining to spark-out can be performed automatically by applying a skip signal each time rough machining, semi-fine machining, fine-machining, or spark-out operation is completed.

NOTE
This function is an optional function.

Format

Move command

G31 P_ IP_ F_ ;

P_ : P1 to P4
IP_ : End point
F_ : Feedrate

Dwell

G04X(U,P)_(Q_) ;

X(U,P)_ : Dwell time
Q_ : Q1 to Q4

Explanation

Parameters can be used to specify whether the 4-point or 8-point skip signal is used (when a high-speed skip signal is used). Specification is not limited to one-to-one correspondence. It is possible to specify that one skip signal correspond to two or more G codes. The multi-skip function can also be set to skip dwell.

3.25.3 High-speed Skip

The skip function operates based on a high-speed skip signal (connected directly to the CNC; not via the PMC) instead of an ordinary skip signal. In this case, up to eight signals can be input.
Delay and error of skip signal input is 0 to 2 msec at the NC side (not considering those at the PMC side). This high-speed skip signal input function keeps this value to 0.1 msec or less, thus allowing high precision measurement.

Format

G31 IP ;

G31; One-shot G code (If is effective only in the block in which it is specified)

3.25.4 Continuous High-speed Skip Function

Overview

The continuous high-speed skip function is used to read absolute coordinates using high-speed skip signals. Inputting a high-speed skip signal in a G31P90 block causes absolute coordinates to be stored in custom macro variables #5061 to #5080. For a system with more than 20 axes, they are stored in variables #100151 to #100182. An axis movement does not stop even if a signal is input, so that the coordinates of multiple points can be read.
Using parameter, it is possible to enable both the rising and falling edges of a high-speed skip signal.

NOTE
This function is an optional function.

Format

G31 P90 IP ;

G31; One-shot G code (If is effective only in the block in which it is specified)

3.25.5 Torque Limit Skip

When the movement command following G31 P99 (or G31 P98) is executed with the servo motor torque limit overridden, cutting feed similar to linear interpolation (G01) can be performed. When the servo motor torque reaches the torque limit (overridden servo motor torque limit) by pushing or the skip signal (including the high-speed skip signal) is input during the movement, the remaining movement commands are canceled and then the next block is executed. (The operation that executes the next block by canceling the remaining movement command is called skip operation later.)

The servo motor torque limit can be overridden by:

(1) Executing the torque limit override command for the PMC window.

(2) Executing the address Q command in the same block in which G31 P99 (or G31 P98) exists.

Format

G31 P98 Q_ α_ F_ ;

G31 P99 Q_ α_ F_ ;

G31 : Skip command (one-shot G code)
P98 : Performs a skip operation if the torque of the servo motor reaches the limit value.
P99 : Performs a skip operation if the torque of the servo motor reaches the limit value or if a skip signal is input.
Q : Override value for the torque limit
Range of valid settings: 1 to 254 (%)
0 to 255 correspond to 0% to 100%.
The Q command is optional. If omitting it, specify the torque limit command in the PMC window beforehand. If it is omitted, and no torque limit override is set beforehand, alarm PS0035 is issued.
If an attempt is made to specify a value out of range, alarm PS0366 is issued.
The specified override value is effective only to the block in which it is specified.
At the end of the skip operation, the override value returns to the one assumed immediately before the G31 command.
α : Axis address on any one axis
F : Feedrate

3.26 REFERENCE POSITION RETURN

3.26.1 Automatic Reference Position Return

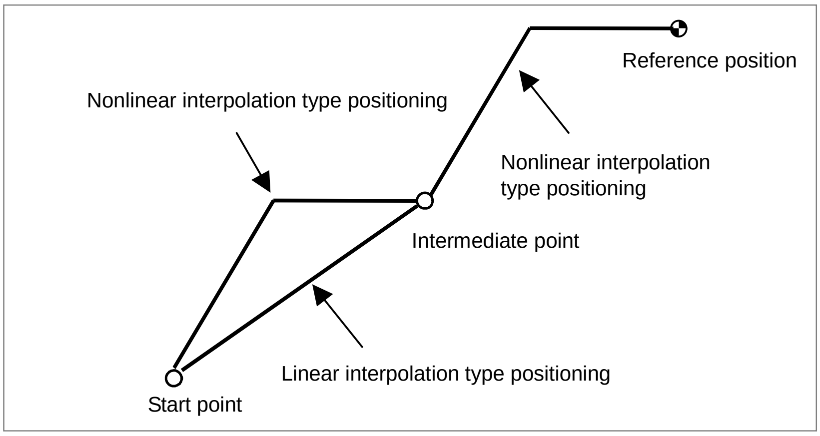
- Return to reference position (G28)

With the G28 command, the commanded axis is positioned to the reference position via the commanded intermediate point. After positioning, the reference position return end lamp lights.

The reference position must be set in parameter (with the coordinates specified in the machine coordinate system,) before issuing the G28 command.

The tool moves to the intermediate point or reference position at the rapid traverse rate.

For the tool path up to the intermediate point, a selection can be made between nonlinear interpolation type positioning or linear interpolation type positioning.



By parameter setting, linear interpolation positioning can be specified also for the tool path from the intermediate point to the reference position.

Format

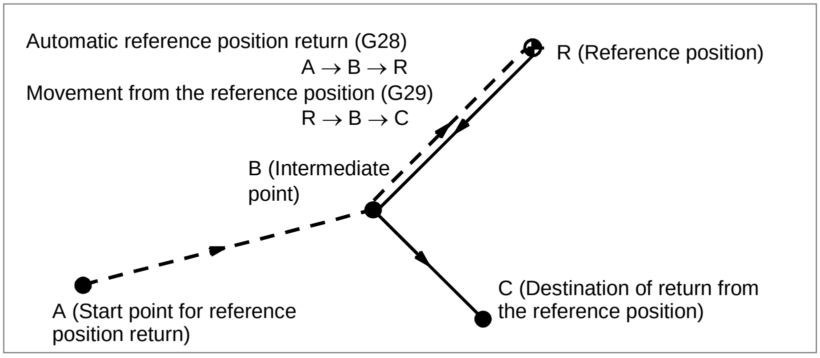
G28 IP_ ;

IP_ : Specify the intermediate position in the absolute coordinate system. (absolute/incremental programming)

There is no need to calculate an actual travel distance between the intermediate position and the reference position.

- Movement from reference position (G29)

Based on the G29 command, the tool is positioned along the specified axis at the point specified by G29 through an intermediate point specified by G28.



Format

G29 IP_ ;

IP : Specify the destination of return from the reference position in the absolute coordinate system. (absolute/incremental programming)

The intermediate point is determined by G28 or G30 specified immediately before this command.

3.26.2 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 position along a specified axis, the lamp for the axis for indicating the completion of reference position return goes on.

If the tool has not reached the reference position, an alarm is issued.

When no movement was made along the axis, whether the current position is the reference position is checked.

Format

G27 IP_ ;

IP : Specify positioning to the reference position in the absolute coordinate system so as to return to the reference position. (absolute/incremental programming)

3.26.3 Second, Third, and Fourth Reference Position Return

The G30 command positions the tool to the 2nd, 3rd, or 4th reference position, via the specified intermediate point. Upon completion of positioning, the 2nd, 3rd, or 4th reference position return completion lamp is turned on.

Before issuing the G30 command, The 2nd, 3rd, or 4th reference position must be set in parameters with coordinates in the machine coordinate system.

For the tool path to the intermediate point and the tool path from the intermediate point to the reference position, a selection can be made between nonlinear interpolation position gain and linear interpolation positioning as in the case of G28.

Return to the 2nd, 3rd, or 4th reference position can be performed only after the reference position has been established.

Format

G30 P2 IP_ ; 2nd reference position return (P2 can be omitted.)

G30 P3 IP_ ; 3rd reference position return

G30 P4 IP_ ; 4th reference position return

IP_ : Specify the intermediate point in the absolute coordinate system. (absolute/incremental programming)

There is no need to calculate an actual travel distance between the intermediate point and the reference position.

3.26.4 In-position Check Disable Reference Position Return

You can disable in-position check at a middle point and reference position by specifying G28.2 or G30.2 as a reference point return command.

Format

G28.2 IP_; Reference position return

G30.2 P2 IP_; 2nd reference position return (P2 can be omitted.)

G30.2 P3 IP_; 3rd reference position return

G30.2 P4 IP_; 4th reference position return

IP : Specify the intermediate position in the absolute coordinate system. (absolute/incremental programming)

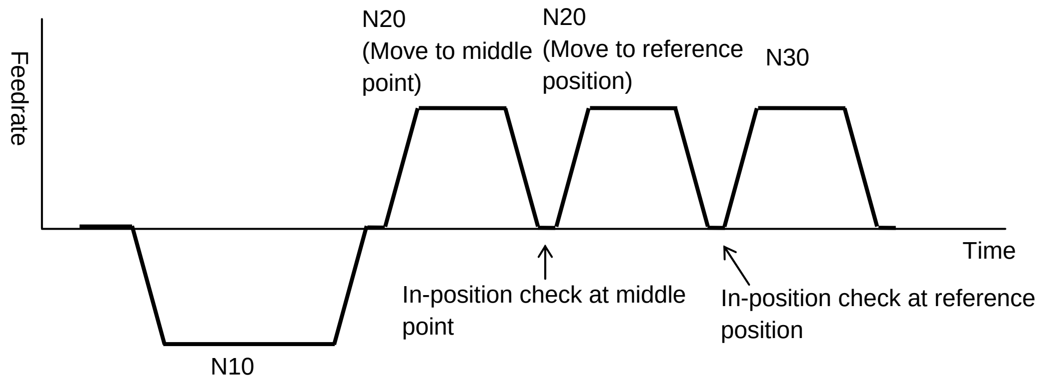
Note When 3rd/4th reference position is used, 3rd/4th reference position option is required.

Example

(1) Reference position return by G28

```
O0002;
N10 G00 X-100.0;
N20 G28 X-50.0;
N30 G00 X50.0;
N40 M30;
```

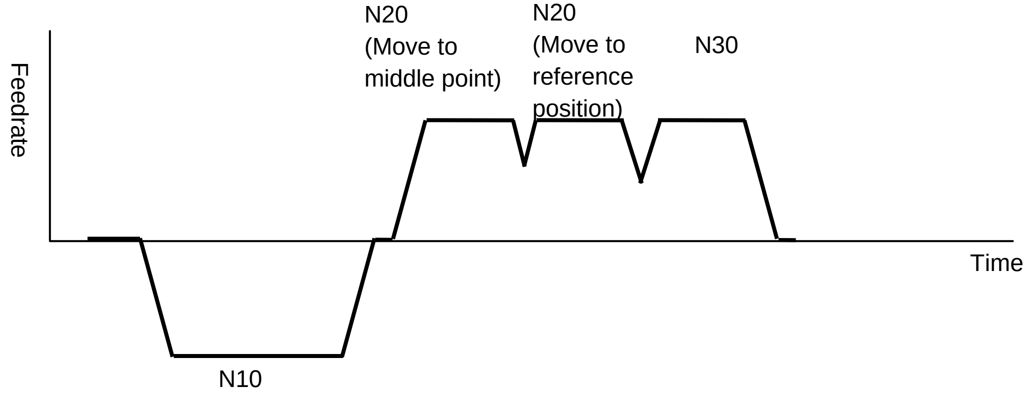
NOTE)- The program is started at coordinate X = 0.0.
- The X-coordinate of the first reference position is 0.0.



(2) Reference position return by G28.2

```
O0003;
N10 G00 X-100.0;
N20 G28.2 X-50.0;
N30 G00 X50.0;
N40 M30;
```

NOTE)- The program is started at coordinate X = 0.0.
- The X-coordinate of the first reference position is 0.0.



3.26.5 Floating Reference Position Return

Tools can be returned to the floating reference position. A floating reference position is a position on a machine tool, and serves as a standard position for machine tool operation. A floating reference position need not always be fixed, but can be moved as required.

Format

G30.1 IP_ ;

IP_ : Specify the intermediate point to the floating reference position in the absolute coordinate system. (absolute/incremental programming)

Explanation

It is possible to return the tool to the floating reference position by commanding the G30.1.

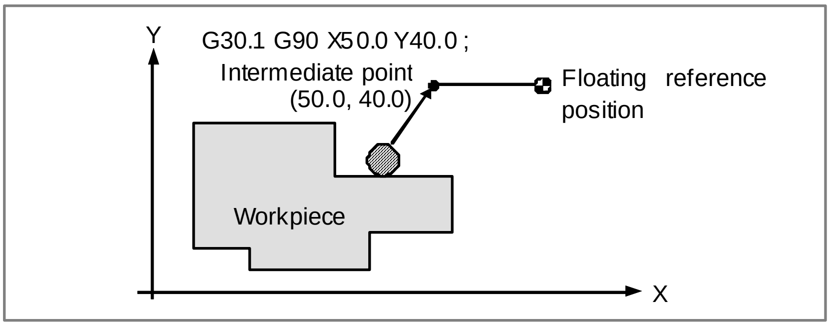
The floating reference position is located on the machine and can be a standard position for some sort of machine operation. It is not always a fixed position and may vary in some cases. The floating reference position can be set using the soft keys of MDI and can be memorized even after power is turned off.

Generally, the position where the tools can be changed on machining center or milling machine is at a set position, the tools cannot be replaced at any just position. Normally the tool change position can be at any of the No. 1 to No. 4 reference positions. The tool can be moved to these positions easily by G30 command. However, depending on the machine, the tools can be replaced at any position as long as it does not contact the workpiece.

For the machinery such as these, in order to reduce the cycle time, it is advantageous to replace tools at a position as close as possible to the workpiece. For this purpose, the tool change position should be changed for each workpiece and this function can easily perform it. The tool change position which is suitable for workpieces can be memorized as the floating reference position and it is possible to return to the tool change position by commanding the G30.1.

Floating reference position return can be used after the reference position has been established.

Example



3.27 NORMAL DIRECTION CONTROL (M SERIES)

M

The rotary axis (C axis) can be controlled by commanding the G41.1 or G42.1 so that the tool constantly faces the direction perpendicular to the advancing direction during cutting.

Format

G41.1 ; Normal direction control left side on

G42.1 ; Normal direction control right side on

G40.1 ; Normal direction control cancellation mode

When the workpiece is on the right-hand side of the tool as viewed in the tool advancing direction, normal direction control left side (G41.1) is specified.

Once G41.1 or G42.1 is specified, normal direction control is enabled (the normal direction control mode is set).

When G40.1 is specified, the normal direction control mode is canceled.

Explanation

In the normal direction control, control is made so that the tool may be perpendicular to the advancing direction on the X-Y plane.

On the angle of C axis, the +X direction is defined to be 0 degrees viewed from the rotation center of C axis. Then, the +Y direction, –X direction, and –Y direction are defined to be 90, 180, and 270 degrees, respectively.

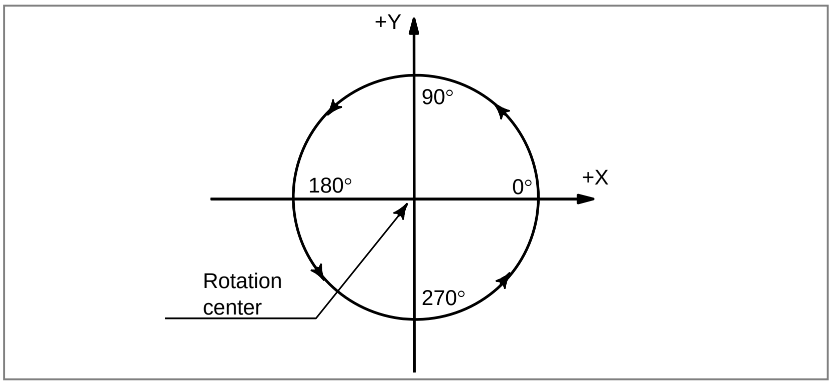


Fig. 3.27 (a) Angle of the C axis

When shifting to the normal direction control mode from the cancellation mode, the C axis becomes perpendicular to the advancing direction where the G41.1 or G42.1 is at the starting point of commanded block.

Between blocks, the traveling of C axis is automatically inserted so that the C axis faces the normal direction at the starting point of each block according to the change of traveling direction.

In the cutter compensation mode, the C axis is controlled to face the normal direction relative to the tool path direction after compensation.

The feedrate of rotation of C axis inserted at the starting point of each block becomes the federate set by parameters. However, when dry run is valid, the feedrate is set to the dry run rate. Also, in the case of rapid traverse (G00), it becomes the rapid traverse rate. In the case of circular interpolation, the C axis is allowed to be rotated first so that the C axis faces perpendicular to the circular starting point. At this time, the C axis is controlled so that it constantly faces the normal direction along with the move of circular interpolation.

NOTE

The rotation of C axis during normal direction control is controlled at short distance so that 180 degrees or less may result.

3.28 BALANCE CUTTING (T SERIES)

T

When a thin workpiece is to be machined as shown below, a precision machining can be achieved by machining each side of the workpiece with a tool simultaneously; this function can prevent the workpiece from warpage that can result when only one side is machined at a time (see the figure below). When both sides are machined at the same time, the movement of one tool must be in phase with that of the other tool. Otherwise, the workpiece can vibrate, resulting in poor machining. With this function, the movement of one tool post can be easily synchronized with that of the other tool post.

NOTE

1. This function is an optional function.
2. When the "mirror image for double turrets" function is selected, the balance cutting function cannot be used.

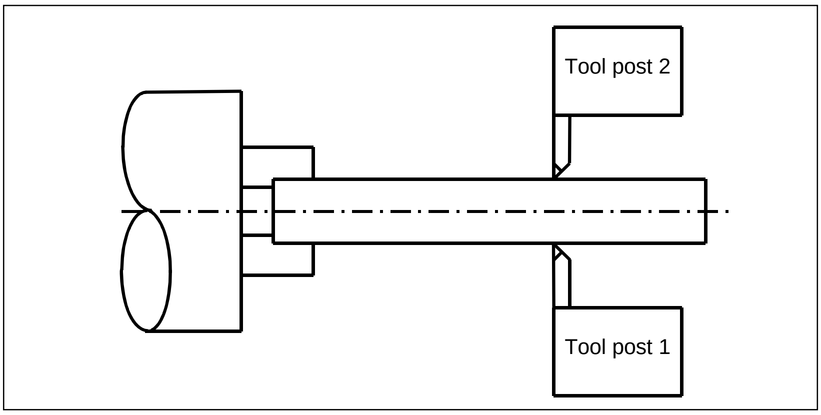


Fig. 3.28 (a) Example of balance cutting

Format

G68 (P_) ; Balance cutting mode on

P_ : Number which specifies a balance cutting combination
(1) In the binary value specification mode, specify the sum of the binary values corresponding to the numbers of paths which require balance cutting.
(2) In the path number specification mode, specify the numbers of all paths that require balance cutting in combination.
(3) When address P is not specified, balance cutting is performed for paths 1 and 2.

G69 ; Balance cutting mode cancel

Specifying G68, which turns balance cutting mode on, causes balance cutting to be performed with the tool post of path 1 and that of path 2. Alternatively, specifying address P in the same block as that containing the G68 command causes balance cutting to be performed between any tool posts.

One of two types of values, binary value or path number, can be set with address P to be specified in a block including G68 for balance cutting mode on. A selection can be made by parameter setting.

3.29 INDEX TABLE INDEXING (M SERISE)

M

By specifying indexing positions (angles) for the indexing axis (one rotary axis, A, B, or C), the index table of the machining center can be indexed.

Before and after indexing, the index table is automatically unclamped or clamped .

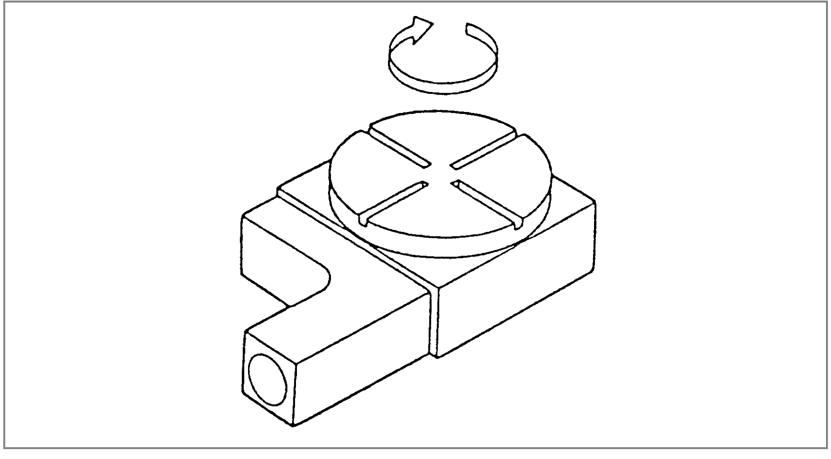


Fig. 3.29 (a) Example of index table

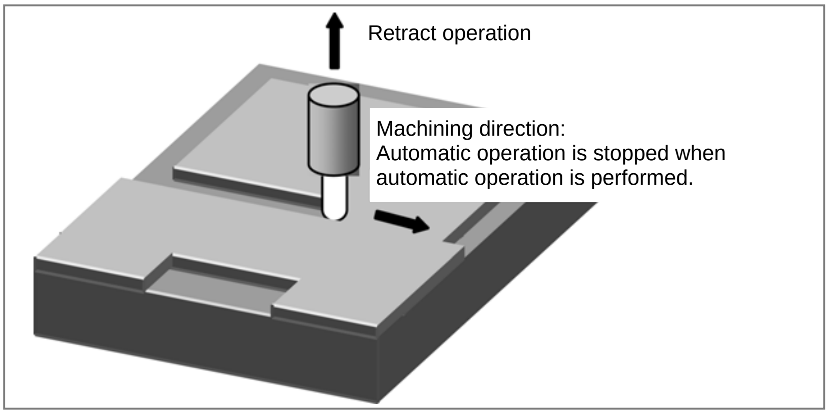
NOTE

This function cannot be used on an axis on which the pole position detection function is used.

3.30 GENERAL PURPOSE RETRACT

In automatic operation mode or in manual operation mode, setting the retract signal to 1 causes this function to capture the rise of this signal, causing the tool to move (retract) along the axis for which a retract amount is specified for parameter. After the end of retraction, the retract completion signal is output. This function is intended to retract the tool from the workpiece immediately when a tool breakage is detected.

- The feedrate assumed during retract is the same as that set in parameter. A feedrate override is invalid.
- A feed hold is invalid to movement during retraction.
- If the retract signal is set to 1 during automatic operation, a retract operation is performed and automatic operation is stopped.
- The retract completion signal becomes 0 when the tool has moved along one of the retract axes.



4 FEED FUNCTION

Chapter 4, "FEED FUNCTION", consists of the following sections:

4.1 RAPID TRAVERSE

The positioning command (G00) positions the tool by rapid traverse.

Format

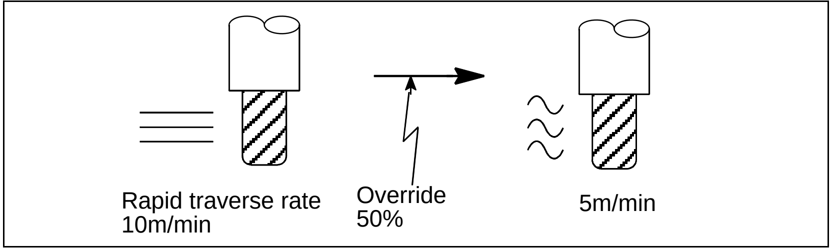
G00 IP_ ;

G00 : G code (group 01) for positioning (rapid traverse)
IP_ : Dimension word for the end point

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 parameter (in-position check).
A rapid traverse rate is set for each axis by parameter, so no rapid traverse feedrate need to be programmed.
An override can be applied to the rapid traverse rate.

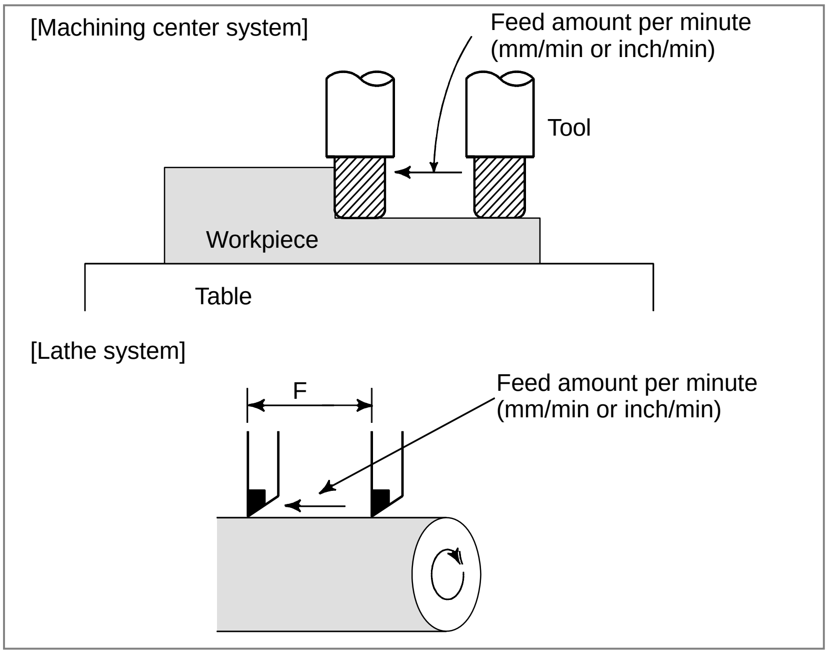
4.2 RAPID TRAVERSE OVERRIDE

An override of four steps (F0, 25%, 50%, and 100%) can be applied to the rapid traverse rate. F0 is set by a parameter .
Also, 1% rapid traverse override select signal or 0.1% rapid traverse override select signal allows rapid traverse override every 1% or 0.1% in the range of 0 to 100%.



4.3 FEED PER MINUTE

After specifying G94 (G98 for lathe system) (in the feed per minute mode), the amount of feed of the tool per minute is specified by setting a number after F. G94 (G98 for lathe system) is a modal code. Once a G94 (G98 for lathe system) is specified, it is valid until G95 (G99 for lathe system) (feed per revolution) is specified. At power-on, the feed per minute mode (feed per revolution mode for lathe system) is set.
An override from 0% to 254% (in 1% steps) can be applied to feed per minute, using the feedrate override signal.



Format

M

G94 ; G code for feed per minute (Group 05)
F_ ; Feedrate (mm/min or inch/min)

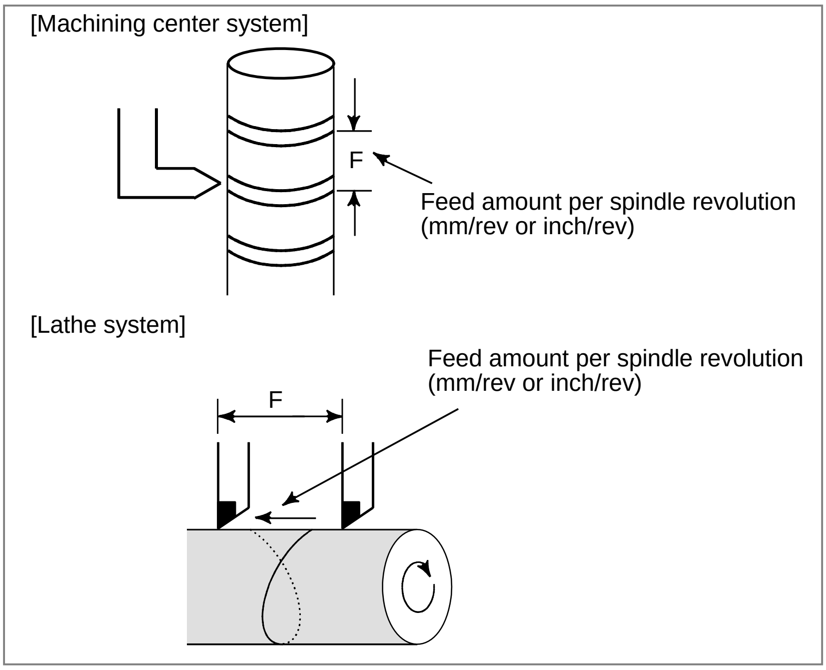
T

G98 ; G code for feed per minute (Group 05)
F_ ; Feedrate (mm/min or inch/min)

4.4 FEED PER REVOLUTION

After specifying G95 (G99 for lathe system) (in the feed per revolution mode), the amount of feed of the tool per spindle revolution is to be directly specified by setting a number after F. G95 (G99 for lathe system) is a modal code. Once a G95 is specified, it is valid until G94 (G98 for lathe system) (feed per minute) is specified.

An override to 0% from 254% (in steps of 1%) can be applied to feed per rotation, using the feedrate override signals.



Format

M

G95 ; G code for feed per revolution (Group 05)
F_ ; Feedrate (mm/rev or inch/rev)

T

G99 ; G code for feed per revolution (Group 05)
F_ ; Feedrate (mm/rev or inch/rev)

4.5 FEED PER REVOLUTION WITHOUT POSITION CODER

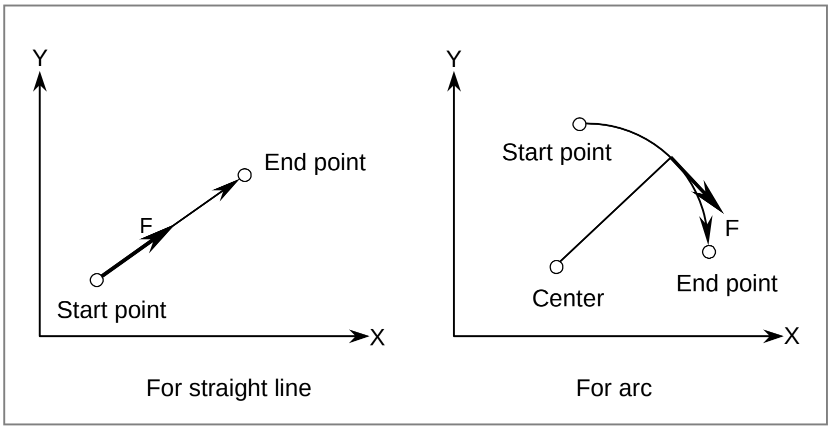
Even when no position coder is attached, a command for feed per revolution can be enabled by parameter setting. (The CNC converts a command for feed per revolution to a command for feed per minute.)

4.6 CONSTANT SURFACE SPEED CONTROL WITHOUT POSITION CODER

In general, feed per revolution cannot be performed on a machine with no position coder attached or used. By parameter setting, feed per revolution can be enabled by spindle commands including a command for constant surface speed control.

4.7 TANGENTIAL SPEED CONSTANT CONTROL

In cutting feed, it is controlled so that speed of the tangential direction is always the same commanded speed.



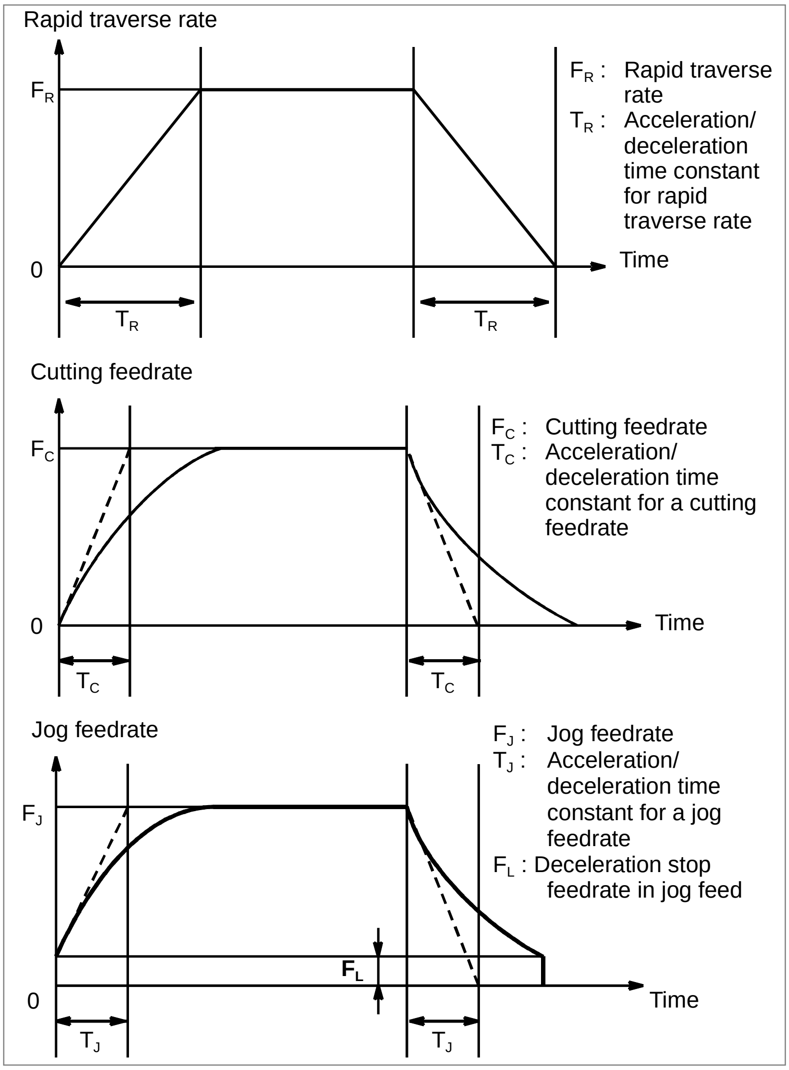
4.8 CUTTING FEEDRATE CLAMP

A maximum allowable cutting feedrate can be set on an axis-by-axis basis with a parameter. The cutting feedrate is clamped to such a maximum feedrate that the result of interpolation does not exceed the maximum allowable feedrate on a specified axis.

4.9 AUTOMATIC ACCELERATION/DECELERATION

To prevent a shock from occurring on the mechanical system, acceleration/deceleration is automatically applied to the tool to enable smooth starting and stopping when the tool starts and ends moving. In addition, acceleration/deceleration is applied automatically when the feedrate changes, so that the feedrate can be changed smoothly.
This means that no programming consideration is required for acceleration/deceleration.

Rapid traverse	:	Linear acceleration/deceleration (time constant per axis is set by parameter)
Cutting feed	:	Exponential acceleration/deceleration (time constant per axis is set by parameter)
Jog feed	:	Exponential acceleration/deceleration (time constant per axis is set by parameter)



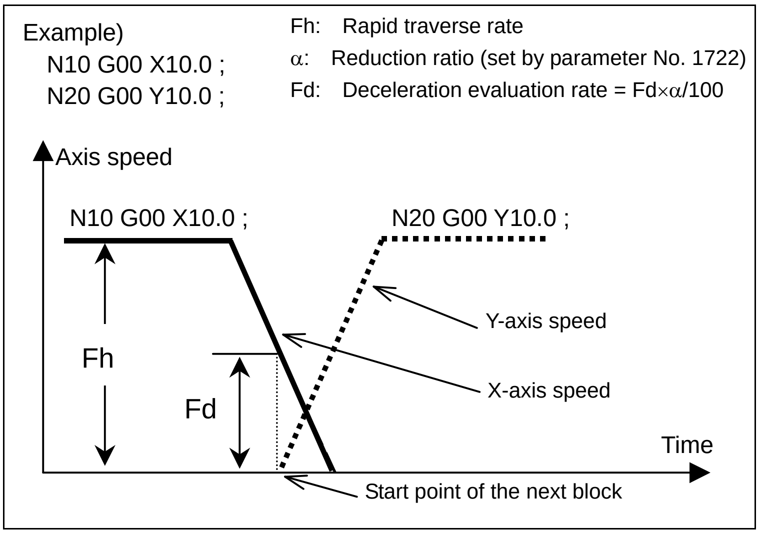
F_R : Rapid traverse rate
T_R : Acceleration/deceleration time constant for rapid traverse rate

F_C : Cutting feedrate
T_C : Acceleration/deceleration time constant for a cutting feedrate

F_J : Jog feedrate
T_J : Acceleration/deceleration time constant for a jog feedrate
F_L : Deceleration stop feedrate in jog feed

4.10 RAPID TRAVERSE BLOCK OVERLAP

If rapid traverse blocks continue or the block next to a rapid traverse block does not move, the next block can be executed when the feedrate of each axis of a rapid traverse block is decreased to the reduction ratio set by the parameter.


Example)
N10 G00 X10.0 ;
N20 G00 Y10.0 ;

Fh: Rapid traverse rate
α: Reduction ratio (set by parameter No. 1722)
Fd: Deceleration evaluation rate = Fd×α/100

4.11 PROGRAMMABLE RAPID TRAVERSE OVERLAP

The programmable rapid traverse overlap supports:

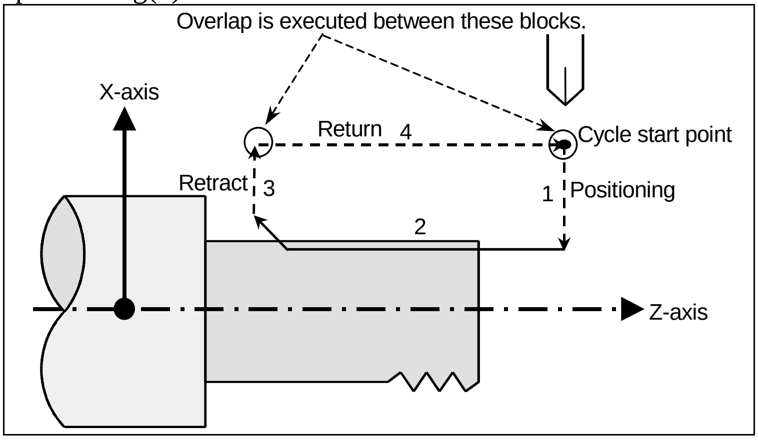
- Changing the feedrate reduction ratio for rapid traverse overlap from the macro program becomes possible.
- Shortening the cycle time becomes possible by doing rapid traverse overlap between threading cycle blocks.

- Programmable rapid traverse overlap

The feedrate reduction ratio for rapid traverse overlap is specified by the parameter. The feedrate reduction ratio can be specified by the system variables.

- Rapid traverse overlap between threading cycle blocks

Rapid traverse overlap is executed between "retract(3)" and "return(4)" and between "return(4)" and "next-threading cycle positioning(1)".

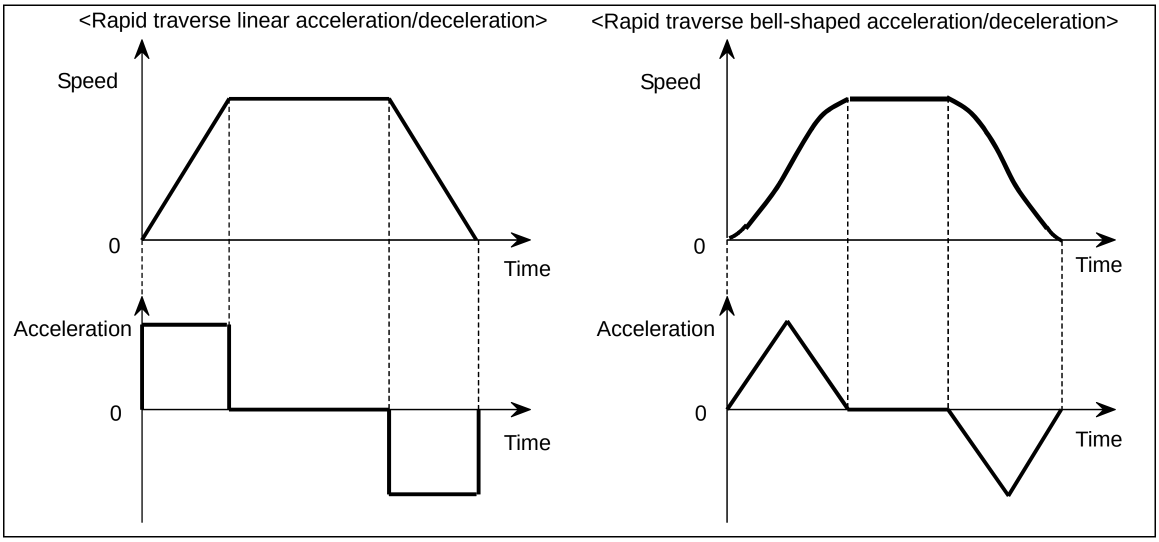


NOTE

This function is an optional function.

4.12 RAPID TRAVERSE BELL-SHAPED ACCELERATION/DECELERATION

Rapid traverse bell-shaped acceleration/deceleration smoothly increases or decreases the rapid traverse rate, reducing the stress and strain imposed on the machine due to the variation in the acceleration with changes in the feedrate. As the time constant for bell-shaped acceleration/deceleration can be smaller than that for linear acceleration/deceleration, the time needed for acceleration/ deceleration can be reduced.



4.13 OPTIMUM TORQUE ACCELERATION/DECELERATION

This function enables acceleration/deceleration in accordance with the torque characteristics of the motor and the characteristics of the machines due to its friction and gravity.

NOTE
This function is an optional function.

Usually, because of the friction of the machine, gravity, the torque characteristics of the motor, and other factors, the acceleration/ deceleration performance (torque for acceleration/deceleration) is different with direction of movement, acceleration or deceleration. In this function, acceleration pattern of rapid traverse for the following situations, plus movement and acceleration, plus movement and deceleration, minus movement and acceleration, minus movement and deceleration can be set into parameters according to the torque for acceleration/deceleration of each situation.
Acceleration/deceleration can be performed according to these parameter setting, so that the most of the capability of the motor can be used and positioning time can be reduced.

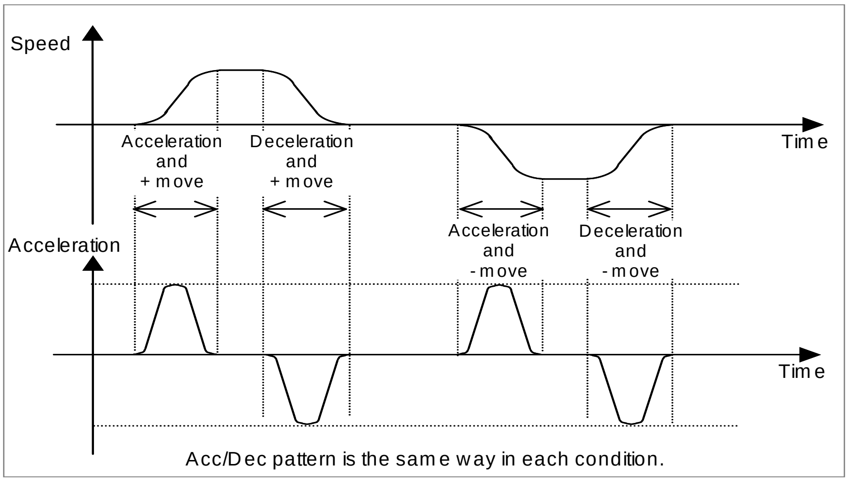


Fig. 4.13 (a) Conventional acceleration/declaration

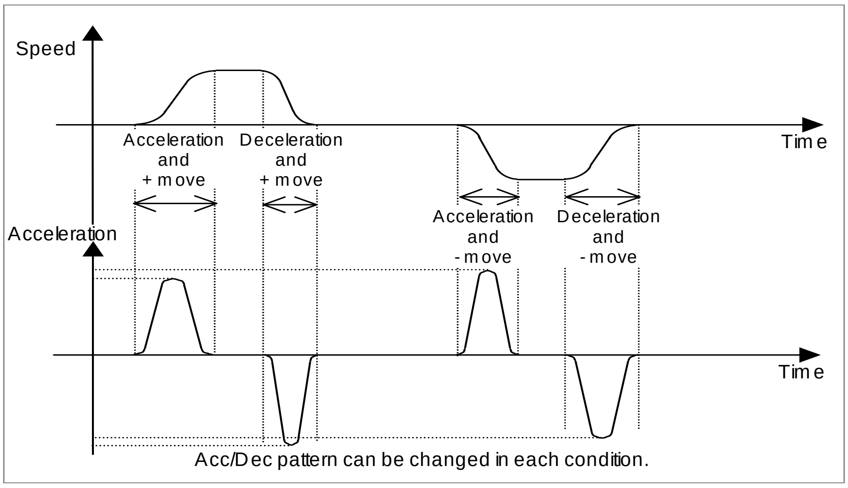


Fig. 4.13 (b) Acceleration/deceleration with this function

4.14 POSITIONING BY OPTIMUM ACCELERATION

When rapid traverse is specified during automatic operation, the rapid traverse rate, time constant, and loop gain can be switched according to the travel distance of the block by the function for positioning by optimum accelerations. So, the time required for positioning and in-position check operations can be reduced, resulting in cycle time reduction.

When rapid traverse (G00) is specified during automatic operation, the rapid traverse rate, time constant, and loop gain are switched to one of seven levels according to the travel distance of the block. Parameters are used to specify the rapid traverse rates, time constants, and loop gains corresponding to travel distances.
This function is disabled for linear interpolation type rapid traverse and cutting feed.

NOTE

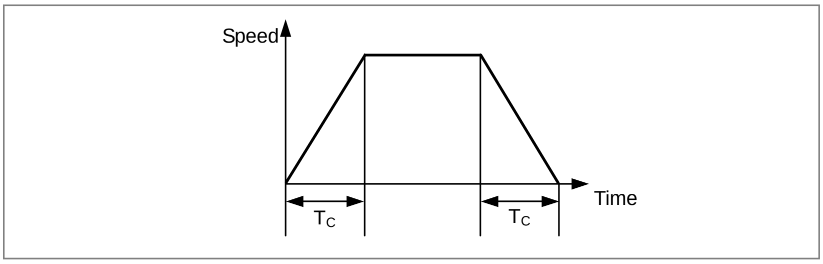
This function is an optional function.

4.15 LINEAR ACCELERATION/DECELERATION AFTER CUTTING FEED INTERPOLATION

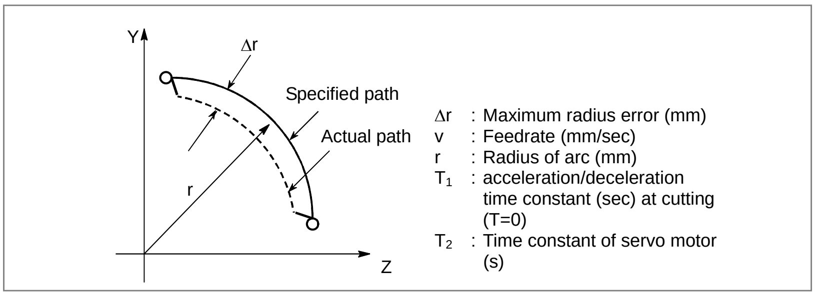
If linear acceleration/deceleration after interpolation for cutting feed is enabled, acceleration/ deceleration is performed as follows:

Cutting feed	:	Linear acceleration/deceleration (constant acceleration time) Specify the acceleration/deceleration time constant for each axis in parameter.
Jog feed	:	Exponential or linear acceleration/deceleration (constant acceleration time) Specify the acceleration/deceleration time constant for each axis in parameter

If an identical time constant is specified, linear acceleration/ deceleration can halve the delay relative to the programmed time, in comparison with exponential acceleration/deceleration, thus reducing the time needed for acceleration and deceleration.



If circular interpolation is performed, especially when high-speed cutting is being performed, the actual tool path created after acceleration/deceleration will deviate from the programmed arc in the radial direction. This deviation can also be reduced, in comparison with exponential acceleration/deceleration, by applying linear acceleration/deceleration.



The maximum value of an error in the radius direction can be approximately obtained by the following expressions:
For exponential acceleration/deceleration:

$$\Delta \mathrm{r} = \left(\frac{1}{2}T_1{}^2 + \frac{1}{2}T_2{}^2\right)\frac{\mathrm{v}^2}{\mathrm{r}}$$

For linear acceleration/deceleration after interpolation:

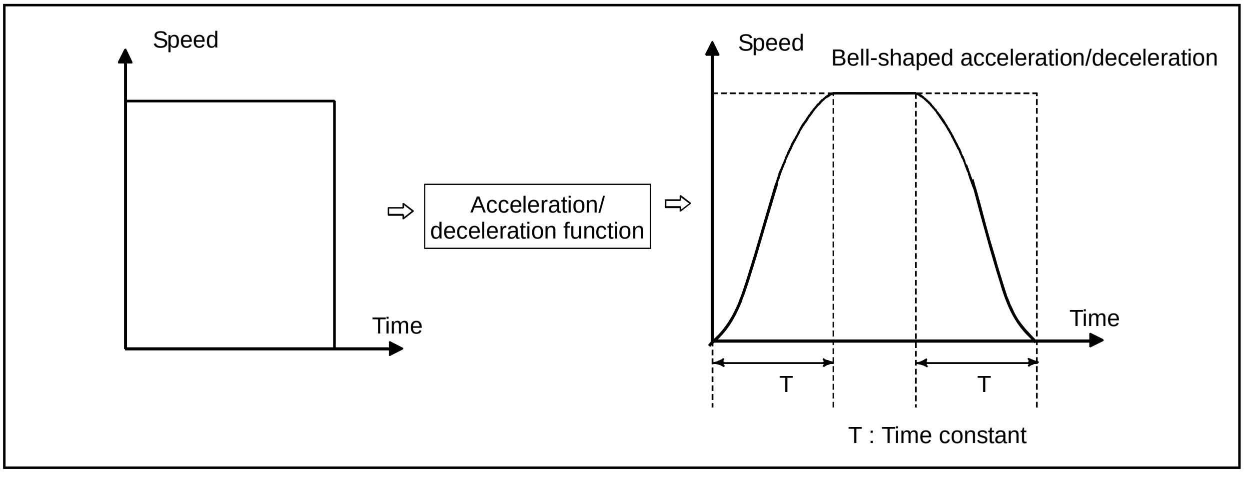
$$\Delta \mathrm{r} = \left(\frac{1}{24}T_1{}^2 + \frac{1}{2}T_2{}^2\right)\frac{\mathrm{v}^2}{\mathrm{r}}$$

Accordingly, if the same time constant for acceleration/deceleration is used, the value of an error that occurs with linear acceleration/deceleration is 1/12 of the value of an error that occurs with exponential acceleration/deceleration when an error due to the time constant of the servo motor is not considered.

4.16 BELL-SHAPED ACCELERATION/DECELERATION AFTER CUTTING FEED INTERPOLATION

The bell-shaped acceleration/deceleration after cutting feed interpolation provides smooth acceleration and deceleration to reduce stress and strain on the machine.

Cutting feed:
  Bell-shaped acceleration/deceleration (constant acceleration time)
  Specify the acceleration/deceleration time constant for each axis in parameter.
Jog feed:
  Exponential or bell-shaped acceleration/deceleration (constant acceleration time)
  Specify the acceleration/deceleration time constant for each axis in parameter.



The time constants for cutting feed and for jog feed on each axis are specified in parameters respectively, in the same way as exponential acceleration/deceleration. The values specified for the FL feedrate for cutting feed and the FL feedrate for jog feed are ignored (always assumed to be 0).

4.17 LINEAR ACCELERATION/DECELERATION BEFORE CUTTING FEED INTERPOLATION

Linear acceleration/deceleration can be applied to the tangential feedrate of a specified feedrate. Thus, unlike acceleration/ deceleration after interpolation applied to each axis, this function can eliminate machining profile errors caused by delay in acceleration/ deceleration.

NOTE

1 For M series, this function is a basic function.
2 For T series, this function is included in the following option.
  - AI contour control I
  - AI contour control II

4.18 FEEDRATE OVERRIDE

The cutting feedrate (per minute feed (G94) and per rotation feed (G95)) can be overrided by :
0 to 254% (in steps of 1%).

M

In inverse time (G93), feedrate converted to feed per minute is overridden. Feedrate override cannot be performed to One-digit F code feed.

Feedrate also cannot be performed to functions as threading and tapping in which override is inhibited.

4.19 SECOND FEEDRATE OVERRIDE

Cutting feedrate can be overrided by:
0 to 254% (in steps of 1%)
A second override can be performed on feedrates once overrided.
No override can be performed on functions as threading and tapping in which override is inhibited.
This function is used for controlling feedrate in adaptive control, etc.

NOTE

This function is an optional function.

4.20 ONE-DIGIT F CODE FEED (M SERIES)

M

If a one-digit number from 1 to 9 is specified after F, the parameter-set feedrate corresponding to the specified number is set.
When F0 is commanded, rapid traverse rate is set.
If the manual pulse generator is rotated with the signal set to 1, the feedrate corresponding to the currently selected number is increased or decreased.
A set or modified feedrate is preserved while the power is turned off.
The current feedrate is displayed on the screen.

4.21 INVERSE TIME FEED (M SERIES)

M

Feedrate of the tool can be specified by the move distance of the block and inverse time (FRN). When F0 is specified, an alarm is issued.

- Linear interpolation (G01)
  FRN=1/Time (min) = Speed/Distance

Speed	:	mm/ min	(metric input)
		inch/ min	(inch input)
Distance	:	mm	(metric input)
		inch	(inch input)

- Circular interpolation (G02, G03)
  FRN=1/Time (min) = Speed/Circle radius

Speed	:	mm/ min	(metric input)
		inch/ min	(inch input)
Circle radius	:	mm	(metric input)
		inch	(inch input)

4.22 JOG OVERRIDE

The jog feedrate and incremental feedrate can be overridden by:

0 to 655.34% (in steps of 0.01%)

4.23 OVERRIDE CANCEL

Feedrate override and the second feedrate override can be clamped to 100% by a signal from the machine side.

4.24 DWELL/AUXILIARY FUNCTION TIME OVERRIDE FUNCTION

This function applies override to the dwell and auxiliary(M/S/T/B) function in the range of 0% to 100% in steps of 1%.

If the override is less than 100% and applied to auxiliary(M/S/T/B) function, the next block is executed after a shortage to the actual processing time, which is considered to be 100%.

4.25 MANUAL PER REVOLUTION FEED (T SERIES)

T

Jog feedrate can be specified by feed per revolution.

4.26 EXTERNAL DECELERATION

The control axis is externally decelerated. The feedrate is decelerated by the signals from the machine. The deceleration rate is set by the parameters.

The signal are provided for each axis and direction.

As five types of deceleration condition settings can be dynamically selected by the signals.

External deceleration can be applied to rapid traverse rate, cutting feedrate, and manual handle feedrate.

- Rapid traverse and cutting feed

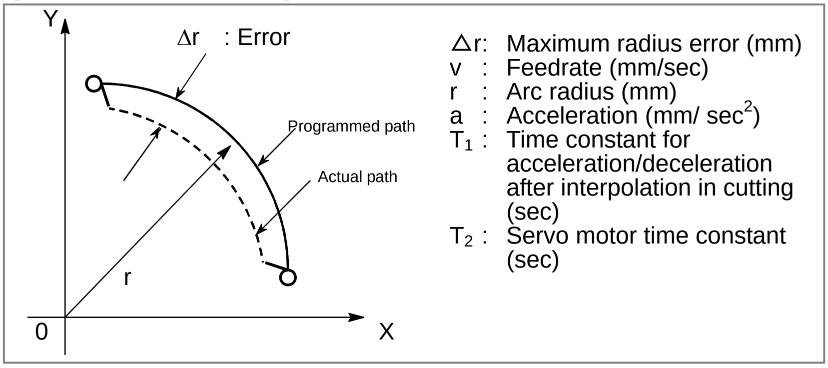
Five types of deceleration conditions can be set by the parameters.
When deceleration conditions specified by multiple signals input during machining are conflicted, the condition with the lowest external deceleration rate is applied.
External deceleration settings 2, 3, 4 and 5 can be switched between enabled and disabled by the parameters.

- Manual handle feed

Five types of deceleration conditions can be set by the parameters. For handle feed, the maximum feedrate is switched when the signal in the positive or negative direction for the handle axis is set to 0. When multiple conditions are conflicted, the condition with the lowest maximum feedrate is applied.

4.27 SPEED CONTROL WITH ACCELERATION IN CIRCULAR INTERPOLATION

When cutting is performed at high speed based on circular interpolation, or helical interpolation, the actual tool path has an error relative to the programmed path. This error can be expressed approximately by the following equation in circular interpolation:



$$\Delta r = \frac{1}{2}\left(T_1{}^2 + T_2{}^2\right)\frac{v^2}{r} = \frac{1}{2}\left(T_1{}^2 + T_2{}^2\right)\cdot a \cdots\cdots\cdots\cdots\cdots\cdots \text{(Equation 1)}$$

In actual machining, the allowable error Δr is given, so that the maximum allowable acceleration rate a (mm/sec^2) is determined by Equation 1.
In acceleration-based speed control in circular interpolation, when a feedrate that causes a radius error to exceed the allowable error along a programmed arc with an arbitrary radius is specified, the feedrate for circular cutting is automatically clamped according to the parameter setting.
Let A be an allowable acceleration rate calculated from an allowable acceleration rate set for each axis. Then, the maximum allowable speed v for the programmed radius r is:

$$\mathrm{v} = \sqrt{\mathrm{A} \cdot \mathrm{r}} \quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \text{(Equation 2)}$$

If a specified feedrate exceeds the speed v found from Equation 2, the feedrate is automatically clamped to v.
Parameter is used to specify an allowable acceleration rate. If allowable acceleration rates for two circular interpolation axes differ from each other, the smaller allowable acceleration rate is selected.
If the radius of an arc is small, the calculated reduced speed v may become very low. To prevent the feedrate from becoming too low in such a case, a minimum allowable feedrate can be set in parameter.

Notes

M

This function can be used in the normal mode.

This function can be used in AI advanced preview control (M series)/AI contour control mode.

4.28 AI ADVANCED PREVIEW CONTROL (M SERIES) / AI CONTOUR CONTROL I / AI CONTOUR CONTROL II

These functions are provided for high-speed, high-precision machining. This function enables suppression of acceleration/deceleration delays and servo delays that become larger with increases in the feedrate and reduction of machining profile errors.
In the descriptions below that are common to AI advanced preview control (M Series), AI contour control I and AI contour control II, the term "AI contour control" is used.

NOTE

The AI contour control I and AI contour control II are an optional function.

Format

G05.1 Q_ ;

Q1 : AI contour control mode on
Q0 : AI contour control mode off

NOTE

1. Always specify G05.1 in an independent block.
2. The AI contour control mode is also canceled by a reset.
3. The AI contour control mode can be turned on at the start of automatic operation by setting parameter.

The AI contour control mode can be controlled also with the formats that have been used for the conventional advanced preview control.

G08 P_ ;

P1 : AI contour control mode on
P0 : AI contour control mode off

NOTE

1. Always specify G08 in an independent block.
2. The AI contour control mode is also canceled by a reset.
3. Valid functions are limited depending on the command format. For details, see the description of "Valid functions".

- Valid functions

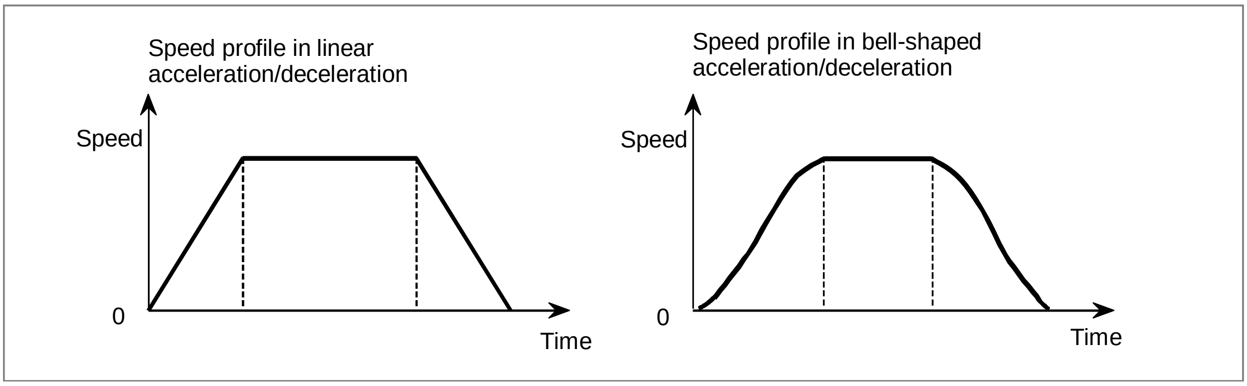
The functions listed below are valid in the AI contour control mode.

Table 4.28 (a) Functions Effective under AI Contour Control

	AI advanced preview control (M series)	**AI contour control I**	**AI contour control II**
Model	0*i*-MF	0*i*-TF 0*i*-MF	0*i*-TF 0*i*-MF
Basic/option	Basic	Option	Option
Number of blocks read ahead	20 (When G8 is specified: 1)	40 (When G8 is specified: 1)	200 (When G8 is specified: 1)
Look-ahead acceleration/deceleration before interpolation	Enabled	Enabled	Enabled
Look-ahead bell-shaped acceleration/ Deceleration before interpolation	Enabled	Enabled	Enabled
Function for changing time constant of bell-shaped acceleration/deceleration	Enabled	Enabled	Enabled
Advanced feed forward	Enabled	Enabled	Enabled
Acceleration setting for each axis	Enabled	Enabled	Enabled
Speed control based on the feedrate difference on each axis	Enabled	Enabled	Enabled
Speed control with Acceleration in circular interpolation	Enabled	Enabled	Enabled
Speed control with the acceleration on each axis	Enabled (When G8 is specified: Not enabled)	Enabled (When G8 is specified: Not enabled)	Enabled (When G8 is specified: Not enabled)
Smooth speed control	Not enabled	Not enabled	Enabled (When G8 is specified: Not enabled)
Speed control with cutting load	Not enabled	Not enabled	Enabled
Disregard of feedrate command	Not enabled	Not enabled	Enabled
Jerk control (M Type) **- Speed control with change of acceleration on each Axis** **- Look-ahead smooth bell-shaped acceleration/deceleration before Interpolation**	Not enabled	Not enabled	Optional function (When G8 is specified: Not enabled)
Nano smoothing (M Type)	Not enabled	Not enabled	Optional function

4.29 BELL-SHAPED ACCELERATION/DECELERATION BEFORE LOOK-AHEAD INTERPOLATION

By producing a bell-shaped feedrate profile for acceleration/ deceleration before interpolation in AI advanced preview control (M Series), AI contour control I or AI contour control II, machining profile errors caused by delay in acceleration/ deceleration can be eliminated, and a shock to the machine can be reduced for smoother acceleration/deceleration.



NOTE

This function is included in "AI advanced preview control (M series)", "AI contour control I " and "AI contour control II ".

4.30 JERK CONTROL (M SERIES)

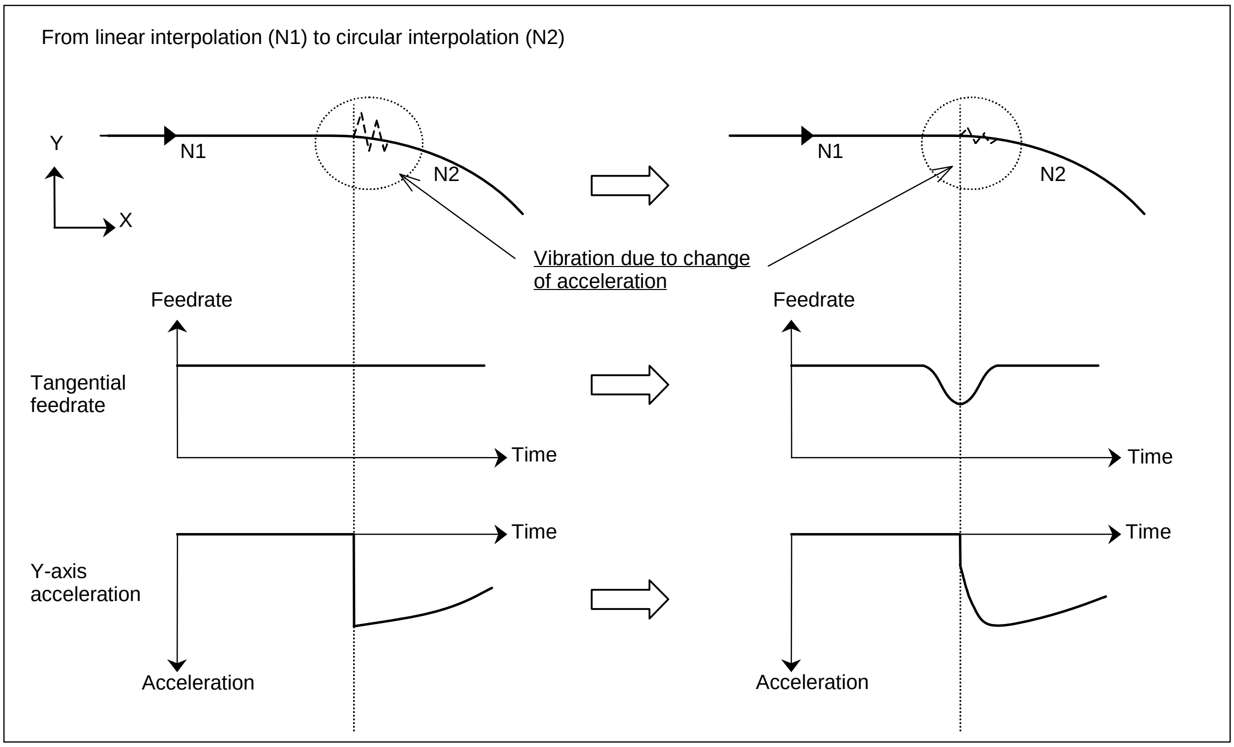
M

In portions in which acceleration changes largely, such as a portion where a programmed figure changes from a straight line to curve, vibration or shock on the machine may occur. Speed control with change of acceleration on each axis is a function to suppress machining errors due to vibration and machine shock generated by change of acceleration. This function obtains a feedrate so that change of acceleration is within the parameter-set permissible acceleration change amount for each axis, and performs deceleration by using acceleration/deceleration before interpolation.

NOTE

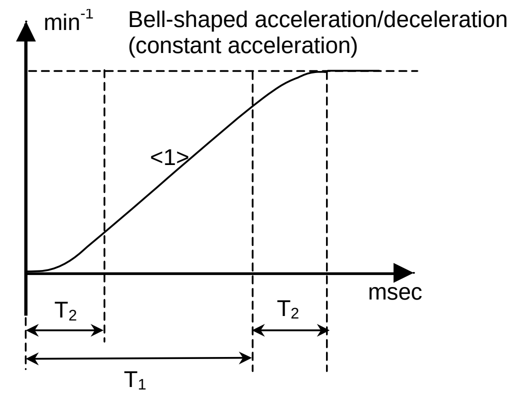
This function is an optional function.
To use this function, the both options for "AI contour control II " and this function are required.

In the following example, the Y-axis acceleration changes largely at the contact point between a linear interpolation and circular interpolation, so deceleration is performed.



4.31 RIGID TAPPING BELL-SHAPED ACCELERATION/DECELERATION

In rigid tapping, bell-shaped acceleration/deceleration of constant acceleration type can be applied by parameter setting.
In parameters, set a time required until a maximum spindle speed is reached (linear portion) and a time required for the curved portion.
The actual time constant for the linear portion is the proportion of the maximum spindle speed to a specified S value.
The time constant for the curved portion is determined not by an S command but by a set value.
Accordingly, the actual time constant is [proportionally calculated value for linear portion + value for curved portion].



Set time constants T_1 and T_2 in parameters.
(From the maximum speed and setting of T_1, the acceleration of linear portion <1> is constant.)

NOTE
In 3-dimensional rigid tapping, bell-shaped acceleration/deceleration is disabled, and linear acceleration/deceleration is applied.

4.32 OPTIMUM ACCELERATION/DECELERATION FOR RIGID TAPPING

This function can be used to flexibly set the acceleration/deceleration during cutting in rigid tapping according to the torque characteristics of a spindle motor and the mechanical characteristics such as machine friction. Depending on the torque characteristics of a spindle motor and the mechanical characteristics, the acceleration/deceleration performance (referred to below as the maximum acceleration curve) that can be output is not symmetrical in the low-speed and high-speed parts.
The conventional acceleration/deceleration (linear-shaped/bell-shaped) is symmetrical, so it was impossible to make the most of the motor performance.
This function can be used to perform acceleration/deceleration so that the actual speed curve can follow the maximum acceleration curve as close as possible. This can make the most of the motor performance and reduce the cutting time. When rigid tapping is used in an area where the acceleration of the spindle motor is constant, however, reduction in the cutting time cannot be expected.
If the acceleration pattern is set for the parameter for each gear, rigid tapping by acceleration following the maximum acceleration curve.
In acceleration/deceleration during withdrawal, the acceleration/deceleration pattern during cutting is used.

NOTE
This function is an optional function.

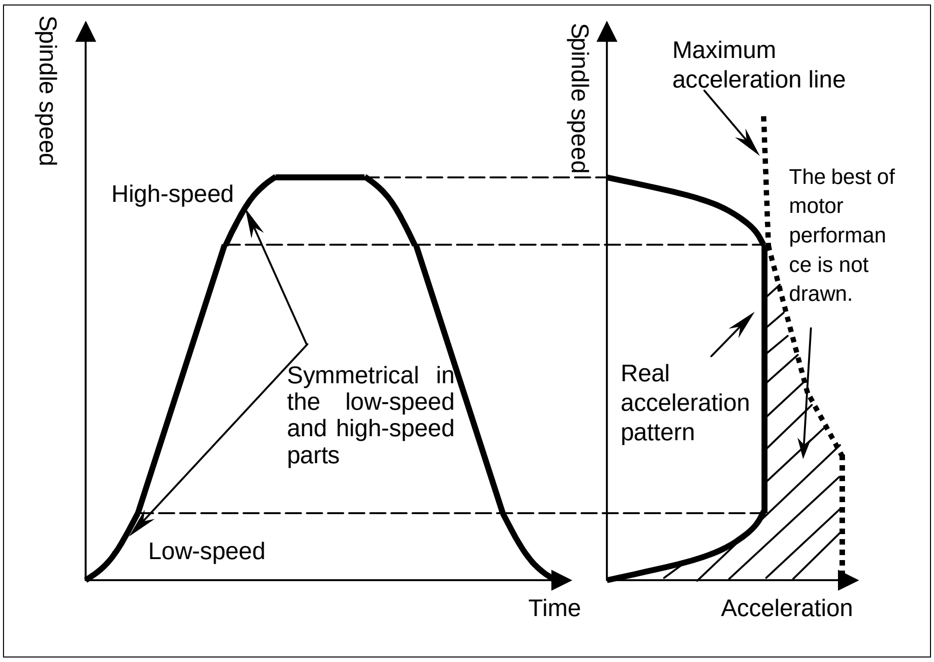


Fig. 4.32 (a) Conventional acceleration/deceleration (bell-shaped)

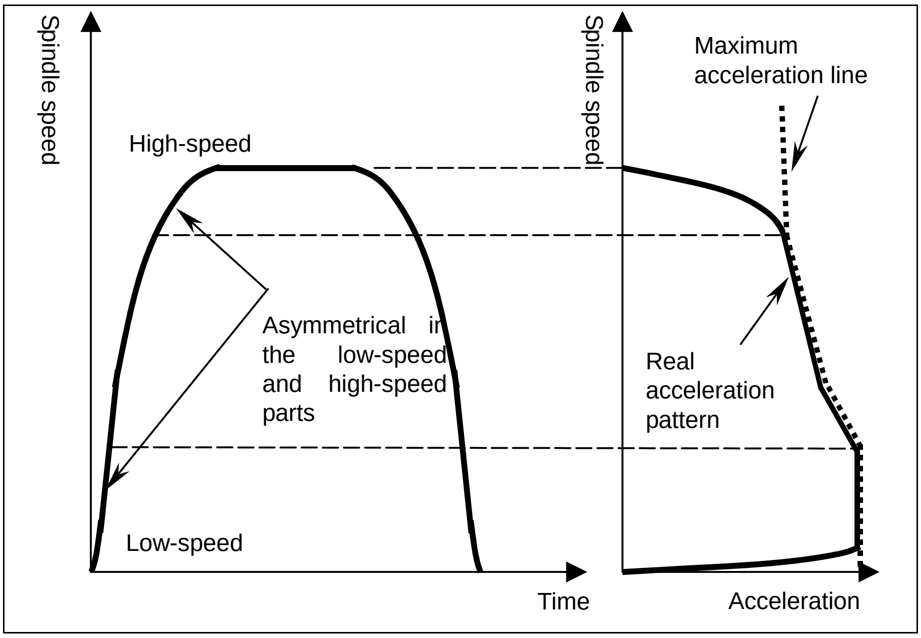


Fig. 4.32 (b) Acceleration/deceleration in which the maximum acceleration curve is followed by the actual acceleration curve by this function

NOTE

This function is an optional function.

4.33 SPEED COMMAND EXTENSION IN LEAST INPUT INCREMENTS C, D, AND E

Overview

When the least input increment C (IS-C) is selected, the limitations indicated in Table 4.33 (a) and Table 4.33 (b) have conventionally been applied to the speed and acceleration parameters. For example, when the unit of data is mm, and the increment system is IS-C, the maximum rapid traverse rate and maximum cutting speed have been unable to exceed 100000 mm/min due to the limitations on the valid data range.

- Conventional speed and angular velocity parameters

Table 4.33 (a)

Unit of data	Increment system	Minimum data unit	Valid data range
mm/min degree/min	IS-C	0.0001	0.0000 to +99999.9999
inch/min	IS-C	0.00001	0.00000 to +4000.00000

- Conventional acceleration and angular acceleration parameters

Table 4.33 (b)

Unit of data	Increment system	Minimum data unit	Valid data range
mm/sec^2 degree/sec^2	IS-C	0.0001	0.0000 to +99999.9999
inch/sec^2	IS-C	0.00001	0.00000 to +9999.99999

With an axis for which parameter is set to 1 and IS-C is specified as the increment system, this function extends the parameter input limitations to those indicated in Table 4.33 (c) and Table 4.33 (d).

With this function, values greater than the conventionally allowed values can be set in the speed and acceleration parameters.

For example, the upper limit of the maximum rapid traverse rate and maximum cutting speed after this function is applied is:

- About 1000000 mm/min when the unit of data is mm, and IS-C is used

- Speed and angular velocity parameters when this function is used

Table 4.33 (c)

Unit of data	Increment system	Minimum data unit	Valid data range
mm/min degree/min	IS-C	0.001	0.000 to +999000.000
inch/min	IS-C	0.0001	0.0000 to +9600.0000

- Acceleration and angular acceleration parameters when this function is used

Table 4.33 (d)

Unit of data	Increment system	Minimum data unit	Valid data range
mm/sec^2 degree/sec^2	IS-C	0.001	0.000 to +999999.999
inch/sec^2	IS-C	0.0001	0.0000 to +99999.9999

NOTE

Note that the least input increment and valid data range of set data vary from the conventional ones.

5 PROGRAM INPUT

Chapter 5, "PROGRAM INPUT", consists of the following sections:

5.1 PROGRAM CODE

Either EIA code or ISO code may be used as program code. Which program code is used for an input program is automatically identified according to the first end-of-block code (EIA: CR or ISO: LF).
For the usable program codes, see the list of program codes in Appendix.

5.2 LABEL SKIP

In one of the following cases, the label skip function is enabled, and "LSK" is indicated on the screen:

- When the power is turned on
- When the CNC is reset

When the label skip function is enabled, all codes are ignored until the first end-of-block (EOB) code is read. A portion ignored is referred to as a "leader portion", and the data after the first end-of-block (EOB) code is regarded as "significant information".

5.3 PARITY CHECK

- Program horizontal direction parity check (TH check)
  Each character of an input program is checked for parity. If a parity error is detected, an alarm is issued.
- Program vertical direction parity check (TV check)
  Each block of an input program is checked for parity. If a block (from the code immediately after an EOB to the next EOB) includes an odd number of characters, an alarm is issued.

The TH check function and TV check function do not operate on a portion skipped by the label skip function. In a comment portion, no TH check is made. However, whether to include the number of characters of a comment portion for a TV check can be chosen by parameter setting. The TV check function can be enabled or disabled by setting through the MDI unit.

5.4 CONTROL-IN / CONTROL-OUT

Any information enclosed by the control-out and control-in codes is regarded as a comment.
A portion ignored is referred to as a "comment portion".
In a comment portion, the reset code (ISO code: % or EIA code: ER) cannot be used.

Table 5.4 (a)

Name	ISO code	EIA code
Control-out	(	2-4-5 channel on
Control-in	)	2-4-7 channel on

5.5 OPTIONAL BLOCK SKIP

If a slash followed by a number (/n) is coded at the start of a block, and optional block skip switch n on the machine operator's panel is turned on, the information of the block where number /n corresponding to switch number n is specified is ignored.
If optional block skip switch n is turned off, the information of the block where /n is specified is not ignored. This means that the block where /n is specified can be skipped according to the choice by the operator.
As n, 1 can be used. The number 1 of /1 can be omitted.
Example)

```
/1 N12345 G00 X100. Z200. ;
```

5.6 OPTIONAL BLOCK SKIP EXTENSION

As n of /n for optional block skip, a number from 2 to 9 can be used.

5.7 MAXIMUM COMMAND VALUES

The basic addresses and specifiable value ranges are indicated below (Table 5.7 (a)). Note, however, that the information below represents restrictions imposed by the CNC, and is totally irrelevant to the restrictions imposed by each machine.

Table 5.7 (a) Major addresses and ranges of command values

Function		Address	Input in mm	Input in inch
Program number		O (*1)	1 to 99999999	1 to 99999999
Sequence number		N	1 to 99999999	1 to 99999999
Preparatory function		G	0 to 99.9	0 to 99.9
Dimension word	Increment system IS-A	X,Y,Z,U,V,W,A,B,C,I, J,K,R (*2)	±999999.99 mm ±999999.99 deg.	±99999.999 inch (*3) ±999999.99 deg.
	Increment system IS-B		±999999.999 mm ±999999.999 deg.	±99999.9999 inch (*3) ±999999.999 deg.
	Increment system IS-C		±99999.9999 mm ±99999.9999 deg.	±9999.99999 inch (*3) ±99999.9999 deg.
Feed per minute	Increment system IS-A	F	0.01 to 999000.00mm/min	0.001 to 96000.000inch/min
	Increment system IS-B		0.001 to 999000.000mm/min	0.0001 to 9600.0000inch/min
	Increment system IS-C		0.0001 to 99999.9999mm/min	0.00001 to 4000.00000inch/min
Feed per revolution		F	0.001 to 50000mm/rev	0.0001 to 50.0000inch/rev
Spindle speed function		S (*4)	0 to 99999999	0 to 99999999
Tool function		T (*4)	0 to 99999999	0 to 99999999
Auxiliary function		M (*4)	0 to 99999999	0 to 99999999
		B (*4)	0 to 99999999	0 to 99999999
Offset number (M series only)		H, D	0 to 999	0 to 999
Dwell	Increment system IS-A	X, U (T series only)	0 to 999999.99 sec	0 to 999999.99 sec
	Increment system IS-B		0 to 99999.999 sec	0 to 99999.999 sec
	Increment system IS-C		0 to 9999.9999 sec	0 to 9999.9999 sec

Function	Address	Input in mm	Input in inch
Dwell	P	1 to 99999999	1 to 99999999
Designation of a program number	P	1 to 99999999	1 to 99999999
Number of sub program repetitions	L	1 to 99999999	1 to 99999999
	P	0 to 9999	0 to 9999

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

*2 When address I, J, K, or R is used to specify the radius for circular interpolation, the specifiable range is as follows (Table 5.7 (b)):

Table 5.7 (b) Specifiable range when address I, J, K, or R is used to specify the radius for circular interpolation

Increment system	Input in mm	Input in inch
IS-A	±999999999.99 mm	±99999999.999 inch
IS-B	±999999999.999 mm	±99999999.9999 inch
IS-C	±99999999.9999 mm	±9999999.99999 inch

*3 For inch input/millimeter machines, the maximum specifiable range of dimension words is as follows (Table 5.7 (c)):

Table 5.7 (c) Maximum specifiable range of dimension words

Increment system	Maximum specifiable range
IS-A	±39370.078 inch
IS-B	±39370.0787 inch
IS-C	±3937.00787 inch

*4 A maximum specifiable value for addresses M, S, T, and B is 99999999. Note that, however, values longer than the permissible number of digits set in parameters cannot be specified. The values and uses for some codes are limited by parameter setting. (For example, some M codes are not buffered.) For details, refer to the parameter manual.

5.8 PROGRAM NAME

Desired program names can be given to part programs in program memory.
Program names can be set as follows:

- Program names are up to 32 characters long.
- The following characters can be used in program names:
  Alphabetical characters (uppercase and lowercase letters), numeric characters, and the symbols below:
  – + _ .
  Because "." and ".." are reserved program names, they cannot be used.

Similarly, an arbitrary folder name can also be assigned to a folder (other than the initial folders) for storing programs.

Program names are associated with program numbers as explained below.
When the program name of a program consists of "O" plus a numeric with the following limitation, the program can be handled also by program number.

- The numeric must be a leading zero suppressed value from 1 to 9999.

When the program name of a program does not have the above format, the program cannot be handled by program number.
When a program name consisting of "O" plus a numeric does not satisfy the above limitation, the program cannot be created.

Example)

Program names that can be treated as program numbers

O123	Program number 123
O1	Program number 1
O3000	Program number 3000
O9999	Program number 9999

Program names that cannot be treated as program numbers

ABC
o123
O123.4

NOTE

1. Program names must each be unique in the same folder.
2. When the program name of a program is not treated as a program number, the program is restricted as follows:
   - The program cannot be specified by program number.
   - Information output by program number is impossible.

5.9 SEQUENCE NUMBER

At the start of a block in a program, an 8-digit sequence number can be specified after address N. The sequence numbers of the program being executed are displayed on the screen at all times. The sequence number search function can be used to search for a sequence number in a program.

5.10 ABSOLUTE PROGRAMMING / INCREMENTAL PROGRAMMING

There are two ways to programming travels of the each axis; the absolute programming, and the incremental programming. In the absolute programming, coordinate value of the end position is programmed. The incremental programming is used to program the amount of each axis movement.
G90 and G91 are used to command absolute or incremental programming, respectively.

Format

G90 IP_ ; Absolute programming
G91 IP_ ; Incremental programming

T

When G code system A is used on the lathe system, an absolute/incremental programming is not identified by a G90/G91 command but by an address word.

Table 5.10 (a)

Move command axis	Absolute programming	Incremental programming
X axis	X	U
Z axis	Z	W
Y axis	Y	V
C axis	C	H
A axis	A	None
B axis	B	None

5.11 DECIMAL POINT PROGRAMMING / POCKET CALCULATOR TYPE DECIMAL POINT PROGRAMMING

Numerical values can be entered with a decimal point. The decimal point can basically be used for a command value with a unit such as for distance, angle, time, or speed, and the position of the decimal point represents mm, inch, deg, or sec.
There are two types of decimal point programming: decimal point programming and pocket calculator type decimal point programming.
When pocket calculator type decimal point programming is used, a value without decimal point is considered to be specified in mm, inch, deg, or sec. When standard type decimal point programming is used, such a value is considered to be specified in least input increments. Select either pocket calculator type decimal point programming or standard type decimal point programming by using the parameter. Values can be specified both with and without decimal point in a single program.

Table 5.11 (a)

Program command	Pocket calculator type decimal point programming	Standard type decimal point programming
X1000 Command value without decimal point	1000mm Unit :mm	1mm Unit : Least input increment (0.001mm)
X1000.0 Command value with decimal point	1000mm Unit :mm	1000mm Unit :mm

5.12 INPUT UNIT 10 TIME MULTIPLY

By parameter setting, a least input increment 10 times greater than a least command increment can be set as indicated in the table below.

Table 5.12 (a) Least input increments 10 times greater than least command increments

Increment system	Least input increment		Least command increment	
IS-B	0.01	mm	0.001	mm
	0.001	inch	0.0001	inch
	0.01	deg	0.001	deg
IS-C	0.001	mm	0.0001	mm
	0.0001	inch	0.00001	inch
	0.001	deg	0.0001	deg

NOTE

When the increment system is IS-A or pocket calculator type decimal point input is used, a least input increment 10 times greater than a least command increment cannot be set.

5.13 DIAMETER PROGRAMMING / RADIUS PROGRAMMING

When turning is performed, the cross section of a workpiece is usually a circle. The size of a circle may be specified by its diameter or radius.
When the diameter is specified, it is called diameter programming and when the radius is specified, it is called radius programming.
Whether to use radius programming or diameter programming can be chosen for each axis by parameter setting.

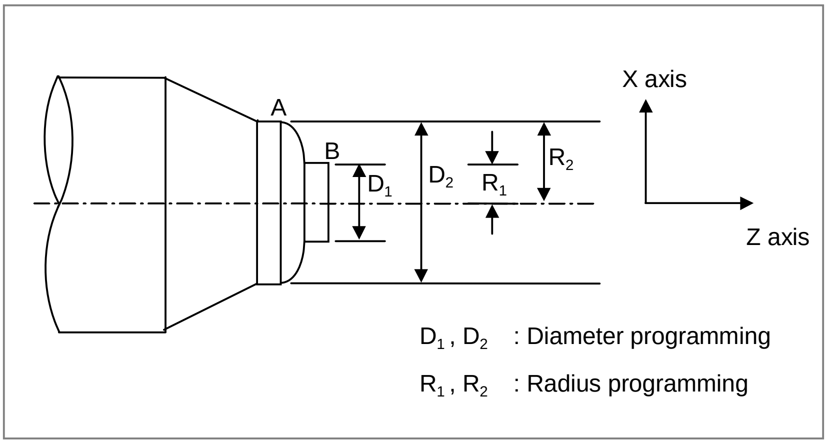


Fig. 5.13 (a)

5.14 PLANE SELECTION

A plane to be used for circular interpolation, plane to be used for cutter compensation, plane to be used for coordinate system rotation, and plane perpendicular to hole machining can be selected using G codes.

G code	Selected plane
G17	Xp-Yp plane
G18	Zp-Xp plane
G19	Yp-Zp plane

Xp : X axis or an axis parallel to it
Yp : Y axis or an axis parallel to it
Zp : Z-axis or an axis parallel to it

Xp, Yp, Zp are determined by the axis address appeared in the block in which G17, G18 or G19 is commanded.
Parameter 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 movement instruction is irrelevant to the plane selection.

Example

- Example 1

When X and U are axes parallel to each other, Y and V are axes parallel to each other, and Z and W are axes parallel to each other

G17 X_ Y_	XY plane
G17 U_ Y_	UY plane
G18 X_ W_	WX plane
G18 U_ W_	WU plane

- Example 2

In a block where none of G17, G18, and G19 is specified, the plane remains unchanged.

G18 X_ Z_	ZX plane
X_ Y_	The plane remains unchanged (ZX plane).

- Example 3

If a block specifies G17, G18, or G19 but has no axis address specified, the omission of the addresses of three basic axes is assumed.

G17	XY plane
G17 X_	XY plane
G17 U_	UY plane

5.15 ROTARY AXIS SPECIFICATION

By parameter setting, a controlled axis can be set as a rotary axis that operates according to a command based on an angular displacement.
Two types of rotary axes are available for selection by parameter setting as indicated below.

Rotary axis (A type)	Rotary axis (B type)
(1) Inch/metric conversion is not done. (2) Machine coordinate values are rounded in 0 to 360°. Absolute coordinate values and relative coordinate values are rounded or not rounded by parameters. (3) Stored pitch error compensation is the rotary axis type. (4) Automatic reference position return (G28, G30) is done in the reference position return direction and the move amount does not exceed one rotation.	(1) Inch/metric conversion is not done. (2) Machine coordinate values, absolute coordinate values, and relative coordinate values are linear axis type. (Is not rounded in 0 to 360°). (3) Stored pitch error compensation is linear axis type (4) Cannot be used with the rotary axis roll-over function and the index table indexing function (M series).

5.16 ROTARY AXIS ROLL-OVER

The absolute coordinate values and relative coordinate values on a rotary axis are rounded to coordinates within one rotation at all times. This function can prevent a coordinate overflow from occurring.
For an incremental command, a command value itself serves as a travel distance.
For an absolute command, a command value is rounded to within one rotation, and a rounded coordinate represents an end point. In this case, whether the direction of movement follows the sign of a command value or the shortcut direction with a shorter travel distance is used can be selected by parameter setting.

5.17 POLAR COORDINATE COMMAND (M SERIES)

M

The end point coordinate value can be input in polar coordinates (radius and angle). Use G15, G16 for polar coordinates command.

Format

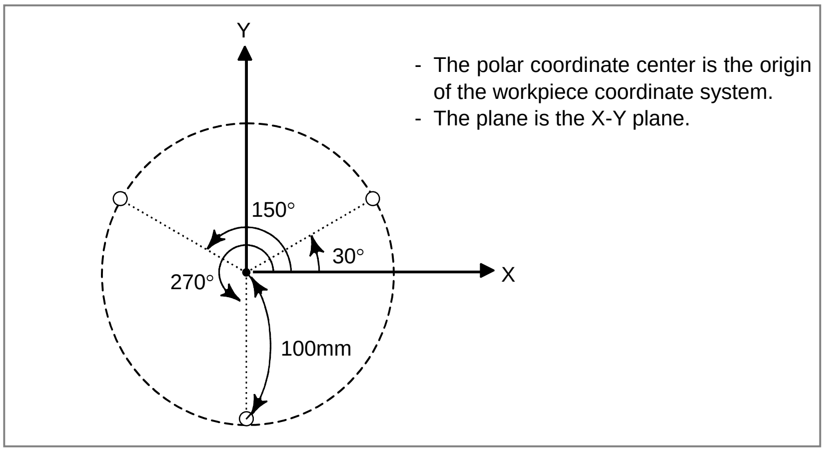
G16; Polar coordinate system command starts
G15; Polar coordinate system command cancel

Explanation

Plane selection of the polar coordinates is done same as plane selection in circular interpolation, using G17, G18, G19.
Command radius in the first axis of the selected plane, and angle in the second axis. For example, when the X-Y plane is selected, command radius with address X, and angle with address Y. The plus direction of the angle is counter clockwise direction of the selected plane first axis + direction, and the minus direction the clockwise direction.
Both radius and angle can be commanded in either absolute or incremental programming (G90, G91).
If a radius value is specified in the absolute mode, the polar coordinate center is the origin of the workpiece coordinate system (or the origin of a local coordinate system if set). If a radius value is specified in the incremental mode, the polar coordinate center is the current position.

Example

Bolt hole circle



- The polar coordinate center is the origin of the workpiece coordinate system.
- The plane is the X-Y plane.

Fig. 5.17 (a)

N1 G17 G90 G16 ;	Polar coordinate command, X-Y plane
N2 G81 X100.0 Y30.0 Z-20.0 R-5.0 F200.0 ;	100mm radius, 30° angle
N3 Y150.0 ;	100mm radius, 150° angle
N4 Y270.0 ;	100mm radius, 270° angle
N5 G15 G80 ;	Polar coordinate command cancel

5.18 COORDINATE SYSTEM SETTING

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 in one of following three coordinate systems:

- Machine coordinate system
- Workpiece coordinate system
- Local coordinate system

As required, specify the position that the tool must reach, by using coordinates in a coordinate system.

5.18.1 Machine Coordinate System

The machine coordinate system is a coordinate system whose origin is a machine-specific point used as a machine reference. A coordinate system in which the reference position becomes the parameter-preset coordinate value when manual reference position return is performed, is set. With G53 command, the machine coordinate system is selected and the axis is able to be moved in rapid traverse to the position expressed by the machine coordinates.

Specifying P1 in a G53 block enables the high-speed G53 function.

In this case, this function enables the inter-rapid traverse block overlap function between machine coordinate selection command (G53) and positioning (rapid traverse) command (G00) blocks, thus making it possible to execute the next rapid traverse command (G00) without decelerating to a stop at the end of the machine coordinate selection command (G53). Therefore, high-speed positioning is available even when the machine coordinate selection command (G53) is used.

Positioning on a machine coordinate system with feedrate can be specified with G53.2. When G53.2 is used, G53.2 option is required

Format

G53 IP_ P1;

IP_ : Absolute command dimension word
P1 : Enables the high-speed G53 function.

G53.2 G01 IP__F__;

IP_ : Absolute command dimension word
F_ : Feedrate

5.18.2 Workpiece Coordinate System

A coordinate system in which the zero point is set to a fixed point on the workpiece, to make programming simple.

A workpiece coordinate system may be set by using the following three methods:

- Using G92 (G50 when G code system A is used on the lathe system)
  A workpiece coordinate system is established using the numeric value programmed after G92.
- Automatic setting
  A workpiece coordinate system is automatically established upon a manual reference position return, as specified in a parameter.
- Using G54 to G59
  Six workpiece coordinate systems must be set from the MDI unit in advance. The workpiece coordinate system to be used is selected by specifying a code selected from G54 to G59.

5.18.2.1 Setting a workpiece coordinate system

By using the following program command, a workpiece coordinate system can be set so that the current tool position is at a specified position.

Format

G92 IP_ ;
(G50 when G code system A is used on the lathe system)

Example

- **Example 1**

As shown below, place the tool tip at the start point of the program, and specify G92 at the start of the program.

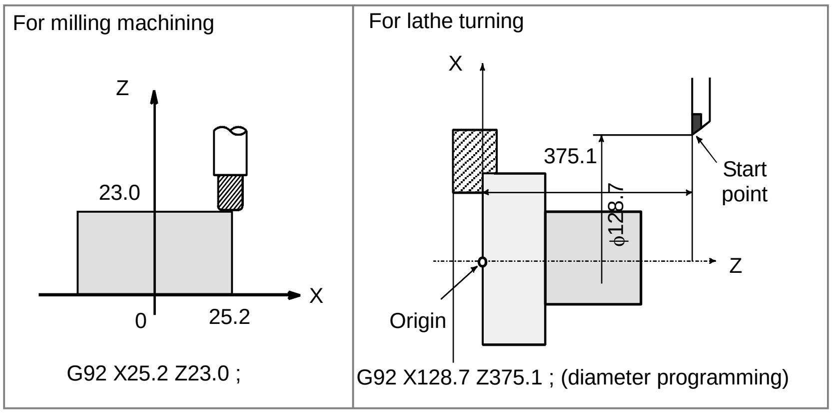
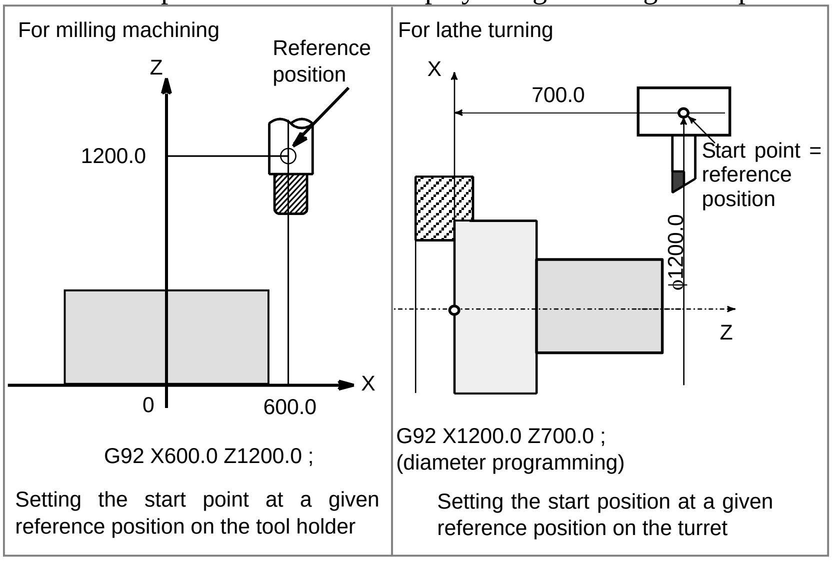


Fig. 5.18.2.1 (a)

- Example 2

Set the reference position on the tool holder or turret as shown in the figure below, then specify G92 at the beginning of the program. By specifying an absolute programming in this condition, the reference position is moved to a specified position. To move the tool tip to a specified position, compensate the distance between the reference position and the tool tip by using tool length compensation.



G92 X600.0 Z1200.0 ;

Setting the start point at a given reference position on the tool holder

G92 X1200.0 Z700.0 ;
(diameter programming)

Setting the start position at a given reference position on the turret

Fig. 5.18.2.1 (b)

When a new workpiece coordinate system is created by specifying G92, it is determined so that a given point on the tool has a given coordinate value. So, there is no need to be concerned with old workpiece coordinate systems. Particularly when the start point for machining is determined based on the workpiece, the G92 command is useful. In this case, a desired coordinate system can newly be created even if an old workpiece coordinate system is invalid.

5.18.2.2 Automatic coordinate system setting

When manual reference position return is performed, a workpiece coordinate system can be set automatically so that the current tool position at the reference position becomes a desired position which is set using a parameter in advance. This functions as if G92IP__; were specified at the reference position. This function can be used when the workpiece coordinate system function is not provided.

5.18.2.3 Setting a workpiece coordinate system

Six workpiece coordinate systems can be set. Set the distance between the machine zero point and the origin of each of the six workpiece coordinate systems (workpiece origin offset value) in advance. There are two setting methods.

- Using the MDI
- Using a program
  Setting based on programmable data input (G10L2Pp)
  Setting based on custom macros

With a command from G54 to G59, one of six coordinate systems can be selected.
Workpiece coordinate systems 1 to 6 are established correctly when a reference position return operation has been performed after power-up. G54 is selected after the power is turned on.

All of the six workpiece coordinate systems can be shifted by a specified amount (external workpiece origin offset value). An external workpiece origin offset value can be set through the MDI unit.

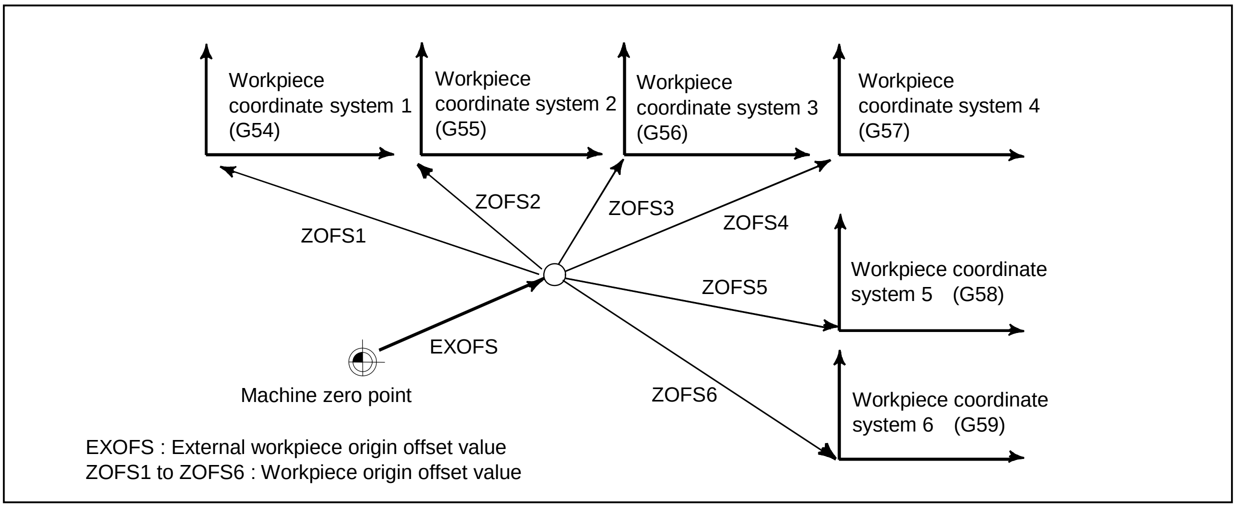


Fig. 5.18.2.3 (a) Workpiece coordinate system

Format

G54		G54	Workpiece coordinate system 1
G55		G55	Workpiece coordinate system 2
G56	IP_ ;	G56	Workpiece coordinate system 3
G57		G57	Workpiece coordinate system 4
G58		G58	Workpiece coordinate system 5
G59		G59	Workpiece coordinate system 6

5.18.3 Local Coordinate System

With G52 commanded, the local coordinate system with the commanded position as zero point can be set. Once the local coordinate system is set, values specified in subsequent move commands are regarded as coordinate values on that coordinate system. Coordinates once set is valid till a new G52 is commanded. This is used when, for example, programming of a part of the workpiece becomes easier if there is a zero point besides the workpiece coordinates zero point.

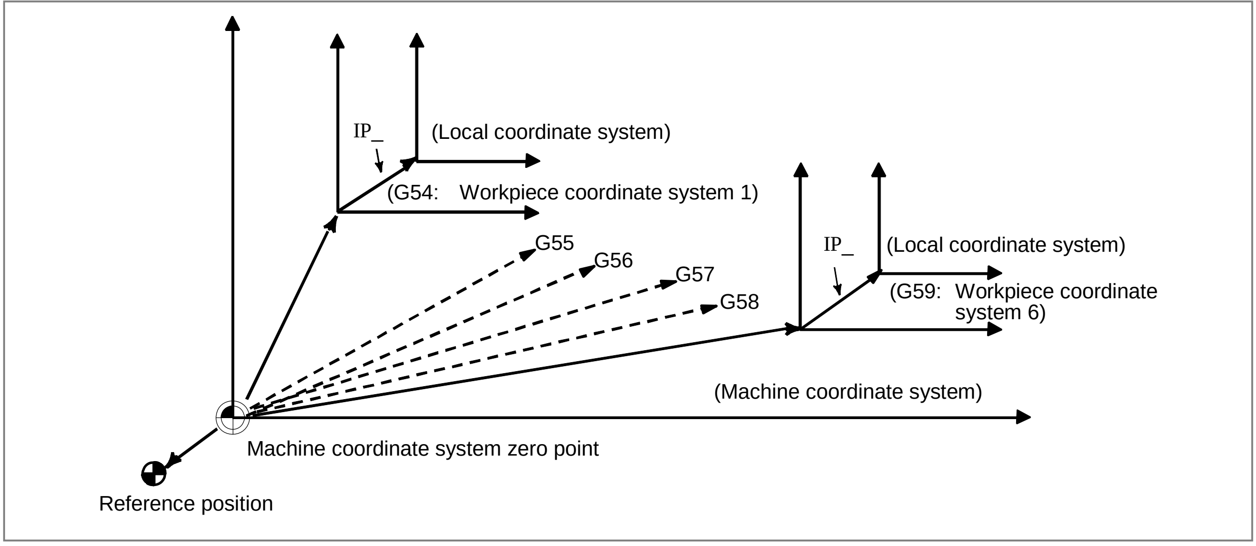


Fig. 5.18.3 (a) Local coordinate system

Format

G52 IP_ ;	Setting the local coordinate system
G52 IP 0 ;	Canceling of the local coordinate system

Explanation

When local coordinate system is set, local coordinate systems 1 to 6, corresponding to workpiece coordinate systems 1 to 6 is set. Distance between zero points are all the same preset value.

5.19 WORKPIECE COORDINATE SYSTEM PRESET

The workpiece coordinate system with its zero point away by the workpiece origin offset value from the machine coordinate system zero point is set by returning the tool to the reference position by a manual operation. Also, when the absolute position detector is provided, the workpiece coordinate system is automatically set by reading the machine coordinate value from the detector when power on without performing manual reference position return operation. The set workpiece coordinate system may shift by any of the following commands or operation:

- Manual intervention performed when the manual absolute signal is off
- Move command executed in the machine lock state
- Movement by handle interruption
- Operation using the mirror image function
- Shifting the workpiece coordinate system by setting the local coordinate system or workpiece coordinate system

By G code specification or MDI operation, a workpiece coordinate system shifted by an operation above can be preset to a workpiece coordinate system offset from the machine zero point by a workpiece origin offset as in the case of manual reference position return.

Format

G92.1 IP 0;

IP 0 : Specifies axis addresses subject to the workpiece coordinate system preset operation. Axes that are not specified are not subject to the preset operation.

5.20 EACH AXIS WORKPIECE COORDINATE SYSTEM PRESET SIGNALS

The each axis workpiece coordinate system preset signals are functions for presetting a workpiece coordinate system shifted due to manual intervention, a machine lock, etc. to a workpiece coordinate system offset from the pre-shift machine zero point by a workpiece origin offset value, using an input signal.

5.21 ADDITION OF WORKPIECE COORDINATE SYSTEM PAIR (M SERIES)

M

Besides the six workpiece coordinate systems based on G54 to G59 (standard workpiece coordinate systems), up to 48 additional workpiece coordinate systems can be used basically, or up to 300 additional workpiece coordinate systems can be used optionally.

As with the workpiece coordinate system based on G54 to G59, the following workpiece origin offset setting and modification methods are available:

- MDI-based method
- Program-based method
  Setting based on programmable data input (G10L2Pp)
  Setting based on custom macros

Format

G54.1 Pn ;

n = 1 to 48 or 1 to 300 (number of the additional workpiece coordinate system)

5.22 DIRECT INPUT OF WORKPIECE ORIGIN OFFSET VALUE MEASURED

By directly entering the measured deviation of the actual coordinate system from a programmed workpiece coordinate system, the workpiece origin offset at the cursor is automatically set so that a command value matches the actual measurement.

5.23 MANUAL ABSOLUTE ON AND OFF

Whether the travel distance of the tool manually moved is to be added to the absolute coordinates in the workpiece coordinate system can be selected using an input signal from the PMC.
If the manual absolute is turned on, the travel distance of the tool manually moved is added to the absolute coordinates.
If the manual absolute is turned off, the travel distance of the tool manually moved is ignored and is not added to the absolute coordinates. So, by the travel distance of the tool manually moved, the workpiece coordinate system can be shifted.

5.24 DIRECT DRAWING DIMENSION PROGRAMMING (T SERIES)

T

Angles of straight lines, chamfering values, corner R values, and other dimensional values on machining drawings can be programmed by directly inputting these values. In addition, the chamfering and corner R can be inserted between straight lines having an arbitrary angle.

Format

The straight line angle, chamfering value, or corner R must be specified with a comma as follows:

,A_ : Angle
,C_ : Chamfering
,R_ : Corner R

NOTE

When A or C is not used as an axis name, the line angle, chamfering value, or corner R can be specified in the parameter without comma as follows:

A_
C_
R_

5.25 G CODE SYSTEM

5.25.1 G Code for Lathe System

T

With the CNC for the lathe system, the G codes listed below are available. Three types of G code systems are usable: A, B, and C. A G code system can be selected by parameter setting.
In this manual, G code system B is used for the descriptions of the preparatory functions below.

Table 5.25.1 (a) G code list

G code system			Group	Function
A	B	C		
◤ G00	◤ G00	◤ G00	01	Positioning (Rapid traverse)
G01	G01	G01		Linear interpolation (Cutting feed)
G02	G02	G02		Circular interpolation CW or helical interpolation CW
G03	G03	G03		Circular interpolation CCW or helical interpolation CCW
G04	G04	G04	00	Dwell
G05.1	G05.1	G05.1		AI contour control
G05.4	G05.4	G05.4		HRV3 on/off
G07.1 (G107)	G07.1 (G107)	G07.1 (G107)		Cylindrical interpolation
G08	G08	G08		AI contour control (Advanced preview control compatible command)
G09	G09	G09		Exact stop
G10	G10	G10		Programmable data input
G10.6	G10.6	G10.6		Tool retract and recover
G11	G11	G11		Programmable data input mode cancel
G12.1 (G112)	G12.1 (G112)	G12.1 (G112)	21	Polar coordinate interpolation mode
◤ G13.1 (G113)	◤ G13.1 (G113)	◤ G13.1 (G113)		Polar coordinate interpolation cancel mode
G15	G15	G15	24	Polar coordinates command cancel
G16	G16	G16		Polar coordinates command
G17	G17	G17	16	XpYp plane selection
◤ G18	◤ G18	◤ G18		ZpXp plane selection
G19	G19	G19		YpZp plane selection
G20	G20	G70	06	Input in inch
G21	G21	G71		Input in mm
◤ G22	◤ G22	◤ G22	09	Stored stroke check function on
G23	G23	G23		Stored stroke check function off
◤ G25	◤ G25	◤ G25	08	Spindle speed fluctuation detection off
G26	G26	G26		Spindle speed fluctuation detection on
G27	G27	G27	00	Reference position return check
G28	G28	G28		Return to reference position
G28.2	G28.2	G28.2		In-position check disable reference position return
G29	G29	G29		Movement from reference position
G30	G30	G30		2nd, 3rd and 4th reference position return
G30.2	G30.2	G30.2		In-position check disable 2nd, 3rd, or 4th reference position return
G31	G31	G31		Skip function
G32	G33	G33	01	Threading
G34	G34	G34		Variable lead threading
G35	G35	G35		Circular threading CW

G code system			**Group**	**Function**
A	**B**	**C**		
G36	G36	G36	01	Circular threading CCW or Automatic tool offset (X axis)
G37	G37	G37		Automatic tool offset (Z axis)
G37.1	G37.1	G37.1		Automatic tool offset (X axis)
G37.2	G37.2	G37.2		Automatic tool offset (Z axis)
G38	G38	G38		Tool radius/tool nose radius compensation: with vector held
G39	G39	G39		Tool radius/tool nose radius compensation: corner rounding interpolation
◤ G40	◤ G40	◤ G40	07	Tool radius/tool nose radius compensation : cancel
G41	G41	G41		Tool radius/tool nose radius compensation : left
G42	G42	G42		Tool radius/tool nose radius compensation : right
G40.1	G40.1	G40.1	19	Normal direction control cancel mode
G41.1	G41.1	G41.1		Normal direction control left on
G42 .1	G42 .1	G42 .1		Normal direction control right on
G43.7 (G44.7)	G43.7 (G44.7)	G43.7 (G44.7)	23	Tool offset (ATC type with lathe system)
G50	G92	G92	00	Coordinate system setting or max spindle speed clamp
G50.3	G92.1	G92.1		Workpiece coordinate system preset
G50.1	G50.1	G50.1	22	Programmable mirror image cancel
G51.1	G51.1	G51.1		Programmable mirror image
◤ G50.2 (G250)	◤ G50.2 (G250)	◤ G50.2 (G250)	20	Polygon turning cancel
G51.2 (G251)	G51.2 (G251)	G51.2 (G251)		Polygon turning
G50.4	G50.4	G50.4	00	Cancel synchronous control
◤ G50.5	◤ G50.5	◤ G50.5		Cancel composite control
G50.6	G50.6	G50.6		Cancel superimposed control
G50.9	G50.9	G50.9		Auxiliary function output in moving axis
G51.4	G51.4	G51.4		Start synchronous control
G51.5	G51.5	G51.5		Start composite control
G51.6	G51.6	G51.6		Start superimposed control
G52	G52	G52		Local coordinate system setting
G53	G53	◤ G53		Machine coordinate system setting
G53.1	G53.1	G53.1		Tool axis direction control
G53.6	G53.6	G53.6		Tool center point retention type tool axis direction control
◤ G54 (G54.1)	◤ G54 (G54.1)	G54 (G54.1)	14	Workpiece coordinate system 1 selection
G55	G55	G55		Workpiece coordinate system 2 selection
G56	G56	G56		Workpiece coordinate system 3 selection
G57	G57	G57		Workpiece coordinate system 4 selection
G58	G58	G58		Workpiece coordinate system 5 selection
G59	G59	G59		Workpiece coordinate system 6 selection
G61	G61	G61	15	Exact stop mode
G63	G63	G63		Tapping mode
G64	G64	G64		Cutting mode
G65	G65	G65	00	Macro call
G66	G66	G66	12	Macro modal call A
G66.1	G66.1	G66.1		Macro modal call B
◤ G67	◤ G67	◤ G67		Macro modal call A/B cancel
G68	G68	G68	04	Mirror image on for double turret or balance cutting mode
G68.1	G68.1	G68.1	17	Coordinate system rotation start or 3-dimensional coordinate system conversion mode on
◤ G69	◤ G69	◤ G69	04	Mirror image off for double turret or balance cutting mode cancel

G code system			Group	Function
A	B	C		
G69.1	G69.1	G69.1	17	Coordinate system rotation cancel or 3-dimensional coordinate system conversion mode off
G70	G70	G72	00	Finishing cycle
G71	G71	G73		Stock removal in turning
G72	G72	G74		Stock removal in facing
G73	G73	G75		Pattern repeating cycle
G74	G74	G76		End face peck drilling cycle
G75	G75	G77		Outer diameter/internal diameter drilling cycle
G76	G76	G78	00	Multiple-thread cutting cycle
G71	G71	G72	01	Traverse grinding cycle
G72	G72	G73		Traverse direct sizing/grinding cycle
G73	G73	G74		Oscillation grinding cycle
G74	G74	G75		Oscillation direct sizing/grinding cycle
◤ G80	◤ G80	◤ G80	10	Canned cycle cancel for drilling
G81.1	G81.1	G81.1	00	High precision oscillation function
G81	G81	G81	10	Spot drilling (FS10/11-T format)
G82	G82	G82		Counter boring (FS10/11-T format)
G83	G83	G83		Cycle for face drilling
G83.1	G83.1	G83.1		High-speed peck drilling cycle (FS10/11-T format)
G83.5	G83.5	G83.5		High-speed peck drilling cycle
G83.6	G83.6	G83.6		Peck drilling cycle
G84	G84	G84		Cycle for face tapping
G84.2	G84.2	G84.2		Rigid tapping cycle (FS10/11-T format)
G85	G85	G85		Cycle for face boring
G87	G87	G87		Cycle for side drilling
G87.5	G87.5	G87.5		High-speed peck drilling cycle
G87.6	G87.6	G87.6		Peck drilling cycle
G88	G88	G88		Cycle for side tapping
G89	G89	G89		Cycle for side boring
G90	G77	G20	01	Outer diameter/internal diameter cutting cycle
G92	G78	G21		Threading cycle
G94	G79	G24		End face turning cycle
G91.1	G91.1	G91.1	00	Maximum specified incremental amount check
G96	G96	G96	02	Constant surface speed control
◤ G97	◤ G97	◤ G97		Constant surface speed control cancel
G96.1	G96.1	G96.1	00	Spindle indexing execution (waiting for completion)
G96.2	G96.2	G96.2		Spindle indexing execution (not waiting for completion)
G96.3	G96.3	G96.3		Spindle indexing completion check
G96.4	G96.4	G96.4		SV speed control mode ON
G98	G94	G94	05	Feed per minute
◤ G99	◤ G95	◤ G95		Feed per revolution
-	◤ G90	◤ G90	03	Absolute programming
-	G91	G91		Incremental programming
-	G98	G98	11	Canned cycle : return to initial level
-	G99	G99		Canned cycle : return to R point level

When the clear state is set at power-up or reset, the G codes are placed in the states marked with ◤ as indicated in Table.

5.25.2 G Code System for Machining Center

M

With the CNC for the machining center system, the G codes listed below are available.

Table 5.25.2 (a) G code list

G code	Group	Function	
◤ G00	01	Positioning (rapid traverse)	
◤ G01		Linear interpolation (cutting feed)	
G02		Circular interpolation CW or helical interpolation CW	
G03		Circular interpolation CCW or helical interpolation CCW	
G04	00	Dwell	
G05.1		AI advanced preview control / AI contour control / Nano smoothing	
G05.4		HRV3, 4 on/off	
G07.1 (G107)		Cylindrical interpolation	
G08		AI advanced preview control / AI contour control (advanced preview control compatible command)	
G09		Exact stop	
G10		Programmable data input	
G10.6		Tool retract and recover	
G10.8		Programmable internal data change	
G11		Programmable data input mode cancel	
G12.1	21	Polar coordinate interpolation mode	
◤ G13.1		Polar coordinate interpolation cancel mode	
◤ G15	17	Polar coordinates command cancel	
G16		Polar coordinates command	
◤ G17	02	XpYp plane selection	Xp: X axis or its parallel axis
◤ G18		ZpXp plane selection	Yp: Y axis or its parallel axis
◤ G19		YpZp plane selection	Zp: Z axis or its parallel axis
G20 (G70)	06	Input in inch	
G21 (G71)		Input in mm	
◤ G22	04	Stored stroke check function on	
G23		Stored stroke check function off	
◤ G25	19	Spindle speed fluctuation detection off	
G26		Spindle speed fluctuation detection on	
G27	00	Reference position return check	
G28		Automatic return to reference position	
G28.2		In-position check disable reference position return	
G29		Movement from reference position	
G30		2nd, 3rd and 4th reference position return	
G30.2		In-position check disable 2nd, 3rd, or 4th reference position return	
G31		Skip function	
G31.8		EGB-axis skip	
G33	01	Threading	
G34		Variable lead threading	
G35		Circular threading CW	
G36		Circular threading CCW	
G37	00	Automatic tool length measurement	
G38		Tool radius/tool nose radius compensation : preserve vector	
G39		Tool radius/tool nose radius compensation : corner circular interpolation	
◤ G40	07	Tool radius/tool nose radius compensation : cancel	
G41		Tool radius/tool nose radius compensation : left	
G42		Tool radius/tool nose radius compensation : right	

G code	Group	Function
G40.1	18	Normal direction control cancel mode
G41.1		Normal direction control on : left
G42.1		Normal direction control on : right
G43	08	Tool length compensation +
G44		Tool length compensation -
G43.7		Tool offset
G45	00	Tool offset : increase
G46		Tool offset : decrease
G47		Tool offset : double increase
G48		Tool offset : double decrease
◤ G49 (G49.1)	08	Tool length compensation cancel
◤ G50	11	Scaling cancel
G51		Scaling
◤ G50.1	22	Programmable mirror image cancel
G51.1		Programmable mirror image
G50.2	31	Polygon turning cancel
G51.2		Polygon turning
G50.4	00	Cancel synchronous control
G50.5		Cancel composite control
G50.6		Cancel superimposed control
G50.9		Auxiliary function output in moving axis
G51.4		Start synchronous control
G51.5	00	Start composite control
G51.6		Start superimposed control
G52		Local coordinate system setting
G53		Machine coordinate system setting
G53.1		Tool axis direction control
G53.6		Tool center point retention type tool axis direction control
◤ G54 (G54.1)	14	Workpiece coordinate system 1 selection
G55		Workpiece coordinate system 2 selection
G56		Workpiece coordinate system 3 selection
G57		Workpiece coordinate system 4 selection
G58		Workpiece coordinate system 5 selection
G59		Workpiece coordinate system 6 selection
G60	00	Single direction positioning
G61	15	Exact stop mode
G62		Automatic corner override
G63		Tapping mode
◤ G64		Cutting mode
G65	00	Macro call
G66	12	Macro modal call A
G66.1		Macro modal call B
◤ G67		Macro modal call A/B cancel
G68	16	Coordinate system rotation start or 3-dimensional coordinate conversion mode on
◤ G69		Coordinate system rotation cancel or 3-dimensional coordinate conversion mode off
G68.2		Tilted working plane indexing
G68.3		Tilted working plane indexing by tool axis direction
G68.4		Tilted working plane indexing (incremental multi-command)
G70.7	00	Finishing cycle
G71.7		Outer surface rough machining cycle
G72.7		End rough machining cycle
G73.7		Closed loop cutting cycle
G74.7		End cutting off cycle
G75.7		Outer or inner cutting off cycle

G code	Group	Function
G76.7	00	Multiple threading cycle
G72.1		Figure copying (rotary copy)
G72.2		Figure copying (linear copy)
G73	09	Peck drilling cycle
G74		Left-handed tapping cycle
G75	01	Plunge grinding cycle
G76	09	Fine boring cycle
G77	01	Plunge direct sizing/grinding cycle
G78		Continuous-feed surface grinding cycle
G79		Intermittent-feed surface grinding cycle
◤ G80	09	Canned cycle cancel Electronic gear box : synchronization cancellation
◤ G80.4	34	Electronic gear box: synchronization cancellation
G81.4		Electronic gear box: synchronization start
G81	09	Drilling cycle or spot boring cycle Electronic gear box : synchronization start
G81.1	00	High precision oscillation
G82	09	Drilling cycle or counter boring cycle
G83		Peck drilling cycle
G84		Tapping cycle
G84.2		Rigid tapping cycle (FS10/11 format)
G84.3		Left-handed rigid tapping cycle (FS10/11 format)
G85		Boring cycle
G86		Boring cycle
G87		Back boring cycle
G88		Boring cycle
G89		Boring cycle
◤ G90	03	Absolute programming
◤ G91		Incremental programming
G91.1	00	Checking the maximum incremental amount specified
G92		Setting for workpiece coordinate system or clamp at maximum spindle speed
G92.1		Workpiece coordinate system preset
G93	05	Inverse time feed
◤ G94		Feed per minute
G95		Feed per revolution
G96	13	Constant surface speed control
◤ G97		Constant surface speed control cancel
G96.1	00	Spindle indexing execution (waiting for completion)
G96.2		Spindle indexing execution (not waiting for completion)
G96.3		Spindle indexing completion check
G96.4		SV speed control mode ON
◤ G98	10	Canned cycle : return to initial level
G99		Canned cycle : return to R point level
G107	00	Cylindrical interpolation
G112	21	Polar coordinate interpolation mode
G113		Polar coordinate interpolation mode cancel
◤ G160	20	In-feed control cancel
G161		In-feed control

When the clear state is set at power-on or reset, the G codes are placed in the states marked with ◤ as indicated in table.

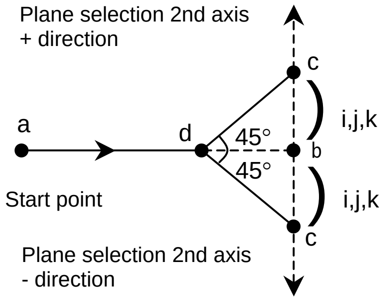
5.26 CHAMFERING AND CORNER R (T SERIES)

T

A chamfer or corner R can be inserted between two blocks which intersect at a right angle as follows.

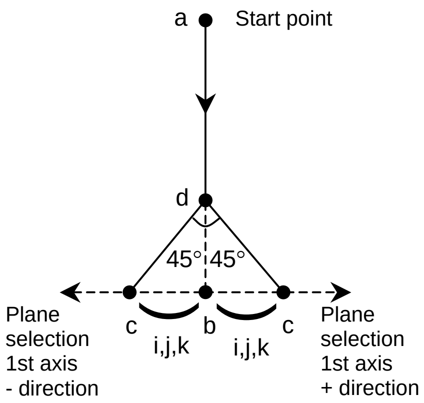
Format

- Chamfering
Plane selection 1st axis → Plane selection 2nd axis
(G17 plane : $X_P \rightarrow Y_P$, G18 plane : $Z_P \rightarrow X_P$, G19 plane : $Y_P \rightarrow Z_P$)

<table>
<tr><th colspan="3">Format</th></tr>
<tr><td colspan="3">G17 plane : G01 X_P(U)_ J(C)±j ;
G18 plane : G01 Z_P(W)_ I(C)±i ;
G19 plane : G01 Y_P(V)_ K(C)±k ;</td></tr>
<tr><th colspan="2">Description of symbols</th><th>Tool movement</th></tr>
<tr><td>X_P(U)_
Y_P(V)_
Z_P(W)_</td><td>Specifies movement from point a to point b with an absolute or incremental programming in the figure on the right.
X_P is the X-axis (basic axis) or its parallel axis, Y_P is the Y-axis (basic axis) or its parallel axis, and Z_P is the Z-axis (basic axis) or its parallel axis.</td><td rowspan="2">

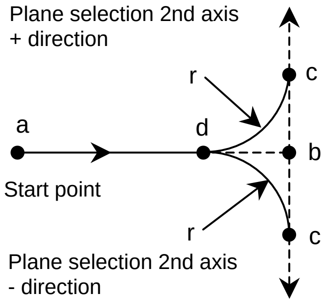
Movements are made from a to d to c.
(In the plane selection 2nd axis + direction if the I, J, K, or C command specifies +, or in the plane selection 2nd axis - direction if I, J, K, or C command specifies -)</td></tr>
<tr><td>I(C)±i
J(C)±j
K(C)±k</td><td>Specify the distance between points b and c in the right-hand figure with address I, J, K, or C followed by a sign. (Use I, J, or K when bit 4 (CCR) of parameter No. 3405 is set to 0 or C when the bit is set to 1.)</td></tr>
</table>

- Chamfering Plane selection 2nd axis → Plane selection 1st axis (G17 plane : $Y_P \rightarrow X_P$, G18 plane : $X_P \rightarrow Z_P$, G19 plane : $Z_P \rightarrow Y_P$)

<table>
<tr><th colspan="3">Format</th></tr>
<tr><td colspan="3">G17 plane : G01 Y_P(V)_ I(C)±i ;
G18 plane : G01 X_P(U)_ K(C)±k ;
G19 plane : G01 Z_P(W)_ J(C)±j ;</td></tr>
<tr><th colspan="2">Description of symbols</th><th>Tool movement</th></tr>
<tr><td>X_P(U)_
Y_P(V)_
Z_P(W)_</td><td>Specifies movement from point a to point b with an absolute or incremental programming in the figure on the right.
X_P is the X-axis (basic axis) or its parallel axis, Y_P is the Y-axis (basic axis) or its parallel axis, and Z_P is the Z-axis (basic axis) or its parallel axis.</td><td rowspan="2">Movements are made from a to d to c.
(In the plane selection 1st axis + direction if the I, J, K, or C command specifies +, or in the plane selection 1st axis - direction if I, J, K, or C command specifies -)

 </td></tr>
<tr><td>I(C)±i
J(C)±j
K(C)±k</td><td>Specify the distance between points b and c in the right-hand figure with address I, J, K, or C followed by a sign. (Use I, J, or K when bit 4 (CCR) of parameter No. 3405 is set to 0 or C when the bit is set to 1.)</td></tr>
</table>

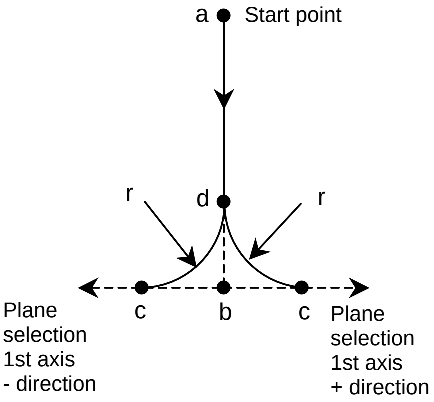
- Corner R
Plane selection 1st axis → Plane selection 2nd axis
(G17 plane : $X_P \rightarrow Y_P$, G18 plane : $Z_P \rightarrow X_P$, G19 plane : $Y_P \rightarrow Z_P$)

Format
G17 plane : G01 X_P(U)_ R<u>±r</u> ; **G18 plane : G01 Z_P(W)_ R<u>±r</u> ;** **G19 plane : G01 Y_P(V)_ R<u>±r</u> ;**

Description of symbols		**Tool movement**
X_P(U)_ **Y_P(V)_** **Z_P(W)_**	Specifies movement from point a to point b with an absolute or incremental programming in the figure on the right. X_P is the X-axis (basic axis) or its parallel axis, Y_P is the Y-axis (basic axis) or its parallel axis, and Z_P is the Z-axis (basic axis) or its parallel axis.	 Movements are made from a to d to c. (In the plane selection 2nd axis + direction if the R command specifies +r, or in the plane selection 2nd axis - direction if the R command specifies -r)
R<u>±r</u>	Specify the radius of an arc connecting points d and c in the right-hand figure with address R followed by a sign.	

- Corner R
Plane selection 2nd axis → Plane selection 1st axis
(G17 plane : Y_P → X_P, G18 plane : X_P → Z_P, G19 plane : Z_P → Y_P)

Format
G17 plane : G01 Y_P(V)_ R±r ; **G18 plane : G01 X_P(U)_ R±r ;** **G19 plane : G01 Z_P(W)_ R±r ;**

Description of symbols		Tool movement
X_P(U)_ **Y_P(V)_** **Z_P(W)_**	Specifies movement from point a to point b with an absolute or incremental programming in the figure on the right. X_P is the X-axis (basic axis) or its parallel axis, Y_P is the Y-axis (basic axis) or its parallel axis, and Z_P is the Z-axis (basic axis) or its parallel axis.	Movements are made from a to d to c. (In the plane selection 1st axis + direction if the R command specifies +r, or in the plane selection 1st axis - direction if the R command specifies -r) 
R±r	Specify the radius of an arc connecting points d and c in the right-hand figure with address R followed by a sign.	

5.27 OPTIONAL CHAMFERING AND CORNER R (M SERIES)

M

Chamfering and corner R blocks can be inserted automatically between the following:

- Between linear interpolation and linear interpolation blocks
- Between linear interpolation and circular interpolation blocks
- Between circular interpolation and linear interpolation blocks
- Between circular interpolation and circular interpolation blocks

- Chamfering

When the above specification is added to the end of a block that specifies linear interpolation (G01) or circular interpolation (G02 or G03), a chamfering block is inserted.

,C_

After C, specify the distance from the hypothetical corner intersection to the start and end points. The hypothetical corner point is the corner point that would exist if chamfering were not performed.

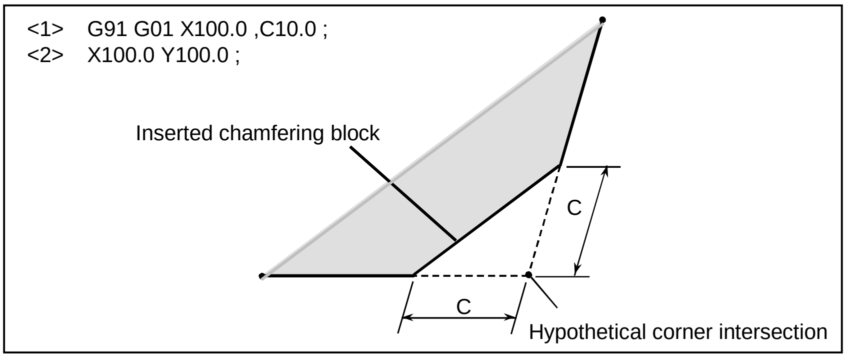


Fig. 5.27 (a)

- Corner R

When the above specification is added to the end of a block that specifies linear interpolation (G01) or circular interpolation (G02 or G03), a corner R block is inserted.

,R_

After R, specify the radius for corner R.

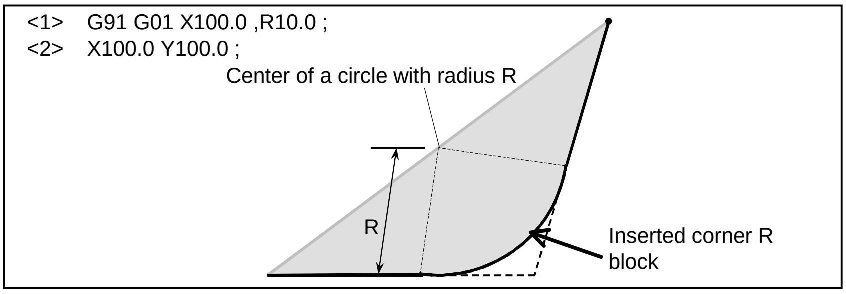


Fig. 5.27 (b)

5.28 PROGRAMMABLE DATA INPUT

By executing programs specified in the formats after a G10 command, various types of data as indicated in the table below can be set.

Data	Format
Pitch error compensation	G10 L50
Workpiece origin offset value	G10 L2
Tool compensation value (M series)	G10 L10/L11/L12/L13
Tool compensation value (T series)	G10 L10/L11
Tool management data	G10 L75
Cartridge management table	G10 L76
Customize data	G10 L77 P1
Name of tool life status	G10 L77 P2
Data input mode cancel	G11

5.28.1 Setting the Pitch Error Compensation Data

By using a programmed command, pitch error compensation data can be set.
This function can be used, for example, to modify pitch error compensation data at the time of attachment replacement.
There are two types of the input format of pitch error compensation data (Incremental value type and total value type). The input format is selected by the parameter.

Format

G10L50 ; Pitch error compensation data entry mode setting
N_(L1)R_ ; Pitch error compensation data entry
:
G11 ; Pitch error compensation data entry mode cancel

N_ : Compensation position number for pitch errors compensation +10,000
L1 : Input format of pitch error compensation data (Total value type)
R_ : Pitch error compensation data

5.28.2 Setting the Workpiece Origin Offset Value

By specifying a G10 command, a workpiece origin offset modification can be made. When G10 is specified in the absolute mode (G90), a specified workpiece origin offset becomes a new workpiece origin offset. When G10 is specified in the incremental mode (G91), a specified workpiece origin offset added to the currently set workpiece origin offset becomes a new workpiece origin offset.

Format

G10 L2 Pp IP_ ;

p=0 : External workpiece origin offset value
p=1 to 6 : Workpiece origin offset value correspond to workpiece coordinate system 1 to 6
IP_ : For an absolute command, workpiece origin offset for each axis.
For an incremental command, value to be added to the set workpiece origin offset for each axis
(the result of addition becomes the new workpiece origin offset).

5.28.3 Setting the Tool Compensation Offset Value

M

By specifying a G10 command, a tool compensation value setting/modification can be made.
When G10 is specified in the absolute mode (G90), a specified value becomes a new tool compensation value.
When G10 is specified in the incremental mode (G91), a specified value added to the currently set tool compensation value becomes a new tool compensation value.

Format

- **For tool compensation memory A**

G10 L11 P_ R_ Q_ ;

P_ : Tool compensation number
R_ : Tool compensation value
Q_ : Imaginary tool nose number

- For tool compensation memory C

G10 L_ P_ R_ Q_ ;

L_ : Type of compensation memory
  L10 : Geometry compensation value corresponding to an H code
  L11 : Wear compensation value corresponding to an H code
  L12 : Geometry compensation value corresponding to a D code
  L13 : Wear compensation corresponding to a D code
P_ : Tool compensation number
R_ : Tool compensation value
Q_ : Imaginary tool nose number

NOTE

1 Address R follows the increment system for tool offset values.
2 If L is omitted for compatibility with the conventional CNC format, or L1 is specified, the same operation as when L11 is specified is performed.

T

By specifying a G10 command, a tool compensation value setting/modification can be made.
If G code system B or C is used, a specified value becomes a new tool compensation value when G10 is specified in the absolute mode (G90). When G10 is specified in the incremental mode (G91), a specified value added to the currently set tool compensation value becomes a new tool compensation value. However, address R is handled as an absolute command, and address C is handled as an incremental command, regardless of the setting of G90/G91.

Format

- For G code system A

G10 P_ X_ Z_ R_ Q_ Y_ ;

or

G10 P_ U_ W_ C_ Q_ V_ ;

P_ : Tool compensation number
  0 : Workpiece coordinate system shift command
  Offset number : Tool compensation value or tool wear compensation value command
  10000 + offset number : Tool geometry compensation value command
  20000 + offset number : Second geometry tool compensation value command
X_ : X axis compensation value (absolute value)
U_ : X axis compensation value (incremental value)
Z_ : Z axis compensation value (absolute value)
W_ : Z axis compensation value (incremental value)
R_ : Tool nose radius compensation value (absolute value)
C_ : Tool nose radius compensation value (incremental value)
Y_ : Y axis compensation value (absolute value)
V_ : Y axis compensation value (incremental value)
Q_ : Imaginary tool nose number

- For G code system B/C

G10 P_ X_ Z_ R_ Q_ Y_ ;

X_ : X axis compensation value
Z_ : Z axis compensation value
R_ : Tool nose radius compensation value (absolute value)
C_ : Tool nose radius compensation value (incremental value)
Y_ : Y axis compensation value
Q_ : Imaginary tool nose number

When the FS10/11 program format is valid, the following format can be used in addition to the formats above:

G10 L_ P_ X_ Z_ R_ Q_ Y_ ; (G code system A/B/C)

or

G10 L_ P_ U_ W_ C_ Q_ V_ ; (G code system A)

L_ : Type of tool compensation memory
  L10 : Geometry compensation value
  L11 : Wear compensation value
P_ : Tool compensation number

NOTE

1. Addresses X, Z, R, Y, U, W, C, and V follow the increment system of tool offset values.
   When a second tool geometry compensation value is specified, addresses R, C, and Q cannot be specified.
2. Address C used for a tool nose radius compensation value is valid only when an axis named C is used.
3. Set a tool nose radius compensation value or imaginary tool nose number when the cutter compensation function or the tool nose radius compensation function is used.
4. Set a Y axis compensation value when the Y-axis offset function is used.
5. Set a tool geometry compensation value when the tool geometry/wear compensation function is used.
6. Set a second geometry tool compensation value when the second geometry tool compensation function is used.
7. For compatibility with the conventional models, addresses U, W, and V are handled as incremental commands for X, Z, and Y axis compensation values, respectively, when G code system B or C is used.

5.28.4 Setting the Tool Management Data

5.28.4.1 Registering new tool management data

Tool management data can be registered. When data is output to an external device from the tool management data screen, this format is used.
The specification of those items that are not registered may be omitted.

Format

G10 L75 P1;
N_ ; Tool management data number specification
T_ C_ L_ I_ B_ Q_ H_ D_ S_ F_ J_ K_ ;
P0 R_ ; Customization data 0
P1 R_ ; Customization data 1
P2 R_ ; Customization data 2
P3 R_ ; Customization data 3
P4 R_ ; Customization data 4
N_ ; Tool management data number
:
G11;

N_	: Tool management data No. 1 to 64	(1 to 240, 1 to 1000)
T_	: Tool type No. (T)	0 to 99,999,999
C_	: Tool life counter	0 to 99,999,999
L_	: Maximum tool life	0 to 99,999,999
I_	: Noticed life	0 to 99,999,999
B_	: Tool life state	0 to 4
Q_	: Tool information	Bit format (8 bits)
H_	: Tool length compensation	No. (H)0 to 999 (M series)
D_	: Cutter compensation No. (D)	0 to 999 (M series)
S_	: Spindle speed (S)	0 to 99,999
F_	: Cutting feedrate (F)	0 to 99,999,999
J_	: Tool geometry compensation No. (G)	0 to 999 (T series)
K_	: Tool geometry compensation No. (W)	0 to 999 (T series)
P_	: Customization data No.	0 to 4 (0 to 20, 0 to 40)
R_	: Customization data value	-99,999,999 to 99,999,999

Specify customization data in the following format:

P (customization-number) R (value)

Use the bit format only when specifying the customization data 0 (P0). Specify other data in the binary format. The specification of customization data that need not be set may be omitted.

5.28.4.2 Modifying tool management data

Tool management data can be modified.
The specification of those items that are not modified may be omitted.

Format

```
G10 L75 P2 ;
N_ ;
T_ C_ L_ I_ B_ Q_ H_ D_ S_ F_ J_ K_ ;
P_ R_ ;
N_ ;
  :
G11 ;
```

5.28.4.3 Deleting tool management data

The data of a specified data number can be deleted from tool management data.
The cartridge management table data corresponding to a deleted tool management data number is also deleted. (The tool management data number in the cartridge management table is cleared to 0.)

Format

```
G10 L75 P3 ;
N_ ;
N_ ;
  :
N_ ;
G11 ;
```

5.28.4.4 Registering new cartridge management table data

A tool management data number can be registered with a free pot in the cartridge management table.

Format

```
G10 L76 P1 ;
N cartridge-number P pot-number R tool-management-data-number ;
N cartridge-number P pot-number R tool-management-data-number ;
N cartridge-number P pot-number R tool-management-data-number ;
N cartridge-number P pot-number R tool-management-data-number ;
G11 ;
```

For a spindle position table and standby position table, only cartridge number data is specified.

5.28.4.5 Modifying the cartridge management table

Tool management data numbers in the cartridge management table can be modified.

Format

```
G10 L76 P2 ;
N cartridge-number P pot-number R tool-management-data-number ;
N cartridge-number P pot-number R tool-management-data-number ;
N cartridge-number P pot-number R tool-management-data-number ;
N cartridge-number P pot-number R tool-management-data-number ;
G11 ;
```

For a spindle position table and standby position table, only cartridge number data is specified.

5.28.4.6 Deleting cartridge management table data

Tool management data numbers can be deleted from the cartridge management table.

Format

```
G10 L76 P3 ;
N cartridge-number P pot-number ;
N cartridge-number P pot-number ;
N cartridge-number P pot-number ;
N cartridge-number P pot-number ;
G11 ;
```

For a spindle position table and standby position table, only cartridge number data is specified.

5.28.4.7 Naming customization data

The display name of customization data (0 to 40) can be set.

Format

```
G10 L77 P1 ;
N_ ;
P_ R_ ;
P_ R_ ;
;
N_ ;
P_ R_ ;
P_ R_ ;
G11 ;
```

N_ : Customization data No. (0 to 40)
P_ : Character No. (1 to 16)
R_ : Character code (ANK or shifted JIS)
- When a shifted JIS code is used, an area for two characters is used.
- Specify 0 to clear data.

- A set display name can be checked on the tool management data screen only.
- When no name is registered, a name such as the default "Customize 3" is displayed.
- A name consisting of up to 16 characters is displayed. For a name shorter than 16 characters, register 0 in the empty area. Those characters that are registered immediately before 0 are displayed.
- To clear data, set 0 as a character code.
- No character code data check is made.

5.28.4.8 Naming tool life states

The display name of a tool life state (0 to 4) can be set.

Format

G10 L77 P2 ;
N_ ;
P_ R_ ;
P_ R_ ;
N_ ;
P_ R_ ;
P_ R_ ;
G11 ;

N_ : Tool life state (0 to 4)
P_ : Character No. (1 to 12)
R_ : Character code (ANK or shifted JIS)
- When a shifted JIS code is used, an area for two characters is used.
- Specify 0 to clear data.

- A set display name can be checked on the tool management data screen only.
- If no name is registered, a default such as "Remaining" and "Unremaining" is displayed.
- A name consisting of up to 12 characters is displayed. For a name shorter than 12 characters, register 0 in the empty area. Those characters that are registered immediately before 0 are displayed.
- To clear data, set 0 as a character code.
- No character code data check is made.
- The defaults are as follows:
  0: Invalid
  1: Remaining
  2: Remaining
  3: Unremaining
  4: Broken

5.29 PROGRAMMABLE PARAMETER INPUT

By programming, the values of parameters can be set.
This function can be used for applications such as modification to a maximum cutting feedrate or cutting time constant due to a machining condition change.
This function is used for the maximum cutting feedrate or cutting time constants are changed to meet changing machining conditions.

Format

G10L52 ; Parameter entry mode setting
N_R_ ; For parameters other than the axis type
N_P_R_ ; For axis type parameters
:
G11 ; Parameter entry mode cancel

N_ : Parameter number
R_ : Parameter setting value (Leading zeros can be omitted.)
P_ : Axis number 1 to maximum controlled axis number (to be specified when an axis type parameter or spindle type parameter is specified)

NOTE
Some parameters cannot be set, when this command is used.

- Parameter for another path

By specifying a path number, it is possible to write to a parameter for that path. There are two ways to specify a path number.

- Adding a path number to a parameter number
- Specifying a path number using a system variable

5.30 SUB PROGRAM CALL

If a program contains a fixed sequence or frequently repeated pattern, such a sequence or pattern can be stored as a sub program in memory to simplify the program. M98 is used to call a sub program, and M99 is used to return from a sub program. Up to ten levels of sub programs can be nested.

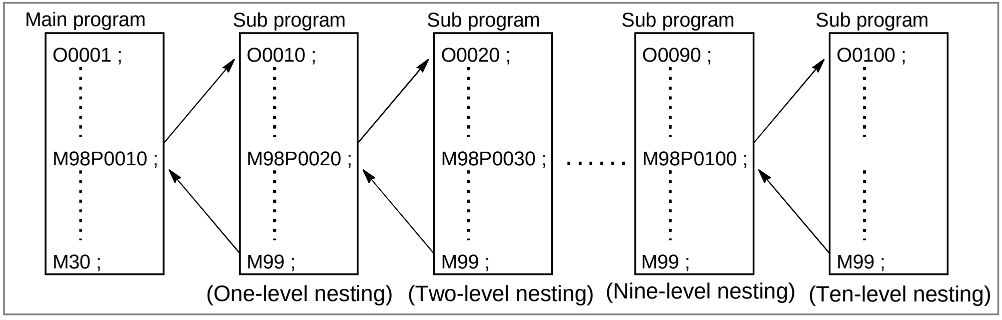


Fig. 5.30 (a)

Format

- Sub program configuration

One sub program

```
Oxxxx ;
   :
   :
M99;
```

Sub program number or sub program file name
(or the colon (:) optionally in the case of ISO)
Program end

M99 need not constitute a separate block as indicated below.

Example) X100.0Y100.0M99;

- Sub program call

- When a sub program with a 4-digit or shorter program number is called

M98 Pxxxx xxxx ;

Sub program number

Number of times the sub program is called repeatedly

- When a sub program with a 5-digit or longer program number is called

M98 Pxxxxxxxx Lxxxxxxxx ;

Sub program number

Number of times the sub program is called repeatedly

- When a sub program is called by file name

M98 <xxxx> Lxxxxxxxx ;

Sub program file name

Number of times the sub program is called repeatedly

NOTE

1 When a sub program with a sub program number shorter than 4 digits is called, the length of the sub program number must be adjusted to 4 digits by adding 0(s) to the beginning of the program number.
Example)
P100100: Call sub program No. 100 ten times.
P50001: Call sub program No. 1 five times.

2 If the number of times the sub program is called repeatedly is omitted when a sub program with a 4-digit or shorter program number is called, the sub program is called just once.
In this case, it is not necessary to adjust the sub program number length to 4 digits as described in Item 1 above.

3 When calling a sub program with a 4-digit or shorter program number, do not specify address L in the same block.

4 When calling a sub program with a 5-digit or longer program number, do not omit the specification of the number of repeats.

5 When calling a sub program by file name, be sure to specify the file name immediately after M98.

- Return from a sub program

M99 ;

M99 need not be specified in a block that specifies no other commands.

5.31 CUSTOM MACRO

A function covering a group of instructions is stored in the memory like the sub program. The stored function is represented by one instruction and is executed by simply writing the represented instruction. The group of instructions registered is called the custom macro body, and the representative instruction, the custom macro instruction.

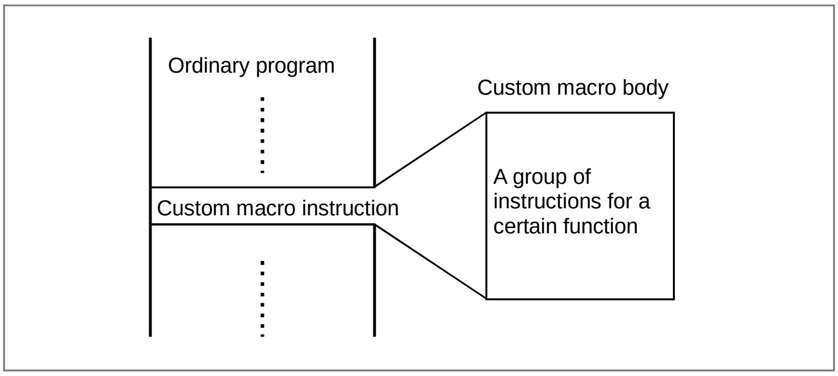


Fig. 5.31 (a)

The programmer need not remember all the instructions in the custom
macro body. He needs only to remember the representative, custom macro instruction.
The greatest feature in custom macro is that variables can be used in the custom macro body. Operation between the variables can be done, and actual values can be set in the variables by custom macro instructions.

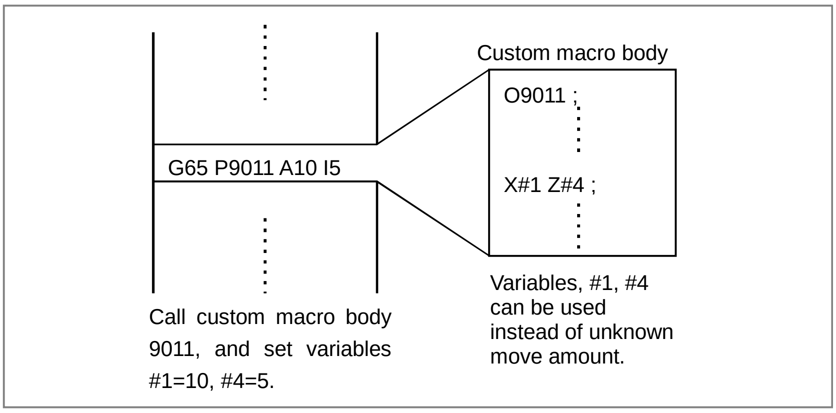


Fig. 5.31 (b)

Example

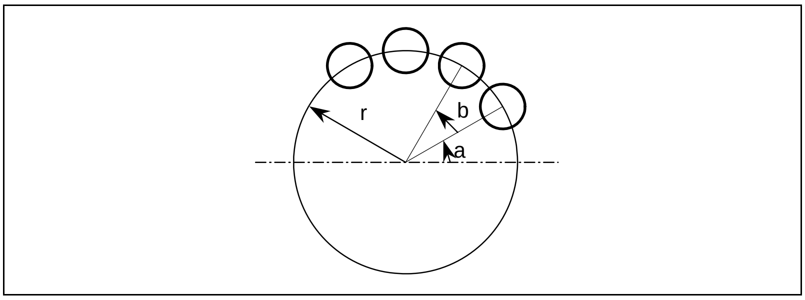


Fig. 5.31 (c)

Bolt hole circle as shown above (Fig. 5.31 (c)) can be programmed easily. Program a custom macro body of a bolt hole circle; once the custom macro body is stored, operation can be performed as if the CNC itself has a bolt hole circle function. The programmer need only to remember the following command, and the bolt hole circle can be called any time.

- Format

G65 Pp Rr Aa Bb Kk ;

p : Macro number of the bolt hole circle
r : Radius
a : Initial angle
b : Angle between holes
k : Number of holes

With this function, the CNC can be graded up by the user himself. Custom macro bodies may be offered to the users by the machine tool builder, but the users still can make custom macro himself.
The following functions can be used for programming the custom macro body.

Explanation

- Use of Variable

Variables : #i (i=1, 2, 3, . . .)
Quotation of variables : F#33 (#33: speed expressed by variables)

Various operation can be done between variables and constants.
The following operands, and functions can be used:
+ (sum), - (difference), * (product), / (quotient), OR (logical sum), XOR (exclusive logical sum), AND (logical product), SIN (sine), ASIN (arc sine), COS (cosine), ACOS (arc cosine), TAN (tangent), ATAN (arc tangent), SQRT (square roots), ABS (absolute value), BIN (conversion from BCD to binary), BCD (conversion from binary to BCD), FIX (truncation below decimal point), FUP (raise fractions below decimal point), ROUND (round), LN (natural logarithm), EXP (exponential function), POW (power), ADP (decimal point addition)
Example) #5=SIN [[#2+#4]*3.14+#4]*ABS[#10]

- Control command

Program flow in the custom macro body is controlled by the following command.

- **If [<conditional expression>] GOTO n (n = sequence number)**

  When <conditional expression> is satisfied, the next execution is done from block with sequence number n.
  When <conditional expression> is not satisfied, the next block is executed.
  When the [<IF conditional expression>] is committed, it executes from block with n unconditionally.
  The following <conditional expressions> are available:

#j EQ #k	whether #j = #k
#j NE #k	whether #j ≠ #k
#j GT #k	whether #j > #k
#j LT #k	whether #j < #k
#j GE #k	whether #j ≥ #k
#j LE #k	whether #j ≤ #k

- **IF [<conditional expression>] THEN macro statement**

  If the specified conditional expression is satisfied, a predetermined macro statement is executed. Only a single macro statement is executed.

- **WHILE (<conditional expression>) DO m (m = 1, 2, 3)**
  :
  END m

  While <conditional expression> is satisfied, blocks from DO m to END m is repeated.
  When <conditional expression> is no more satisfied, it is executed from the block next to END m block.

```
Example)

    #120=1 ;
        WHILE [#120 LE 10] DO 1 ;
        ----------------------------
                                     }
                                     }  Repeated 10 times.
        #120=#120+1 ;                }
        ----------------------------
        END1
```

- Format of custom macro body

The format is the same as the sub program.

```
O Macro number ;

Custom macro body

M99
```

- Custom macro instruction

- Simple call

G65 P (macro number) L (times to repeat) <argument assignment> ;

A value is set to a variable by <argument assignment>.

Write the actual value after the address.

Example)

A5.0 E3.2 M13.4

There is a regulation on which address (A - Z) corresponds to which variable number.

- Modal call (calling a move command)

G66 P (macro number) L (times to repeat) <argument assignment> ;

Each time a move command is executed, the specified custom macro body is called. This can be canceled by G67.

This function is useful when drilling cycles are programmed as custom macro bodies.

- Modal call (calling each block)

G66.1 P (macro number) L (times to repeat) <argument assignment> ;

This type of call unconditionally calls a custom macro body specified in each command block of an NC command. This state is canceled by G67.

- Macro call using G code

The macro can also be called by the parameter-set G codes. Instead of commanding:

N_ G65 Pxxxx <argument assignment> ;

macro can be called just by commanding:

N_ Gyy <argument assignment> ;

G code (Gyy) for calling the macro, and macro program number (Pxxxx) to be called, are coupled together and set as parameter.

- Macro call using M code

Custom macros can be called by pre-determined M codes which are set by parameters.

The following command

N_ G65 Pxxxx <Argument assignment> ;

is equivalent to the following command:

N_ Myy <Argument assignment> ;

The correspondence between M codes (Myy) and program number (Pxxxx) of a macro shall be set by a parameter.
M codes are not sent out the same as the sub program call using M code.

- Sub program call using M code

An M code can be set by parameter to call a sub program. Instead of commanding:
N_ G_ X_ Y_ . . . M98 Pxxxx ; ,
the same operation can be performed simply by commanding:
N_ G_ X_ Y_ . . . Myy ;.
As for M98, M codes are not transmitted.
The M code (Myy) for calling the sub program and the sub program number (Pxxxx) to be called are coupled together and set by parameter.

- Sub program call using T code

By setting parameter, sub program can be called by T codes. When commanded:
N_ G_ X_ Y_ . . . Tt ;
the same operation is done as when commanded:
#149=t ;
N_ G_ X_ Y_ . . . M98 P9000 ;
The T code t is stored as arguments of common variable #149.

- Sub program call using S code

By setting parameter, sub program can be called by S codes. When commanded:
N_ G_ X_ Y_ . . . Ss ;
the same operation is done as when commanded:
#147=s ;
N_ G_ X_ Y_ . . . M98 P9029 ;
The S code s is stored as arguments of common variable #147.

- Sub program call using second auxiliary function code

By setting parameter, sub program can be called by second auxiliary function codes (B). When commanded:
N_ G_ X_ Y_ . . . Bb ;
the same operation is done as when commanded:
#146=b ;
N_ G_ X_ Y_ . . . M98 P9028 ;
The B code b is stored as arguments of common variable #146.

- Sub program call using specific address

By setting parameter, sub program can be called by specific addresses (A). When commanded:
N_ G_ X_ Y_ . . . Ab ;
the same operation is done as when commanded:
#146=b ;
N_ G_ X_ Y_ . . . M98 P9004 ;
The A address b is stored as arguments of common variable #146.
However, a restriction is imposed on the codes that can be set in specific addresses.

- Types of variables

Variables are divided into local variables, common variables, and system variables, according to their variable numbers. Each type has different use and nature.

- Local variables #1 to #33

Local variables are variables used locally in the macro. Accordingly, in case of multiples calls (calling macro B from macro A), the local variable used in macro A is never destroyed by being used in macro B.

- Common variables #100 - #149, #500 - #549

Compared with local variables used locally in a macro, common variables are common throughout the main program, each sub program called from the main program, and each macro. The common variable #i used in a certain macro is the same as the common variable #i used in other macros. Therefore, a common variable #i calculated in a macro can be used in any other macros.

Common variables basically have the read/write attribute. By parameter setting, however, the common variable with a specified variable number can be protected (to make the common variable read-only). The system does not define common variable applications. Instead, the user can freely use common variables.

Common variables #100 to #149 are cleared when the power is turned off. The values of common variables #500 to #549 are preserved even when the power is turned off.

NOTE

With the custom macro common variable addition, the common variables can be extended to #100 to #199 and #500 to #999.

In this case, common variables #100 to #199 are cleared when the power is turned off. The values of common variables #500 to #999 are preserved even when the power is turned off.

- System variables

A variable with a certain variable number has a certain value. If the variable number is changed, the certain value is also changed. The certain value are the following:

- 128 DI points (for read only)
- 128 DO points (for output only)
- Tool compensation value, workpiece origin offset value, and workpiece coordinate system value
- Position information (current position, skip position, block end position, etc.)
- Modal information (F code, G code for each group, etc.)
- Alarm message (Set alarm number and alarm message, and the CNC is set in an alarm status. The alarm number and message is displayed.)
- A date (year, month, day) and time (hour, minute, second) are indicated.
- Clock (Time can be known. A time can also be preset.)
- Single block stop, auxiliary function end wait hold
- Feed hold, Feedrate override, Exact stop inhibition
- The number of machining parts is indicated. It can be preset.

By adding a path number to the high-order 8th and 9th digits of a common variable or a system variable, it is possible to read and write a variable for another path.

- External output commands

Value of variables or characters can be output to external devices with custom macro command. Results in measurement is output using custom macro.

- Parameter reading

By using the operation command PRM, it is possible to read a NC parameter.

By specifying a path number, it is possible to read a parameter for that path. There are two ways to specify a path number.

- Adding a path number to a parameter number
- Specifying a path number using a system variable

- Limitation

- Usable variables

See "Types of variables".

- **Usable variable values**
  Negative value : $-1.8*10^{308}$ to $-2.2*10^{-308}$
  Positive value : $2.2*10^{-308}$ to $1.8*10^{308}$
- **Constants usable in <expression>**
  Maximum : ±999999999999
  Minimum : ±0.00000000001
  Decimal point allowed
- **Arithmetic precision**
  15-digit decimal number
- **Macro call nesting**
  Maximum 5 folds.
- **Repeated ID numbers**
  1 to 3
- **[ ] nesting**
  Maximum 5 folds.
- **Sub program call nesting**
  15 folds (including macro call nesting)

5.32 ADDITION OF CUSTOM MACRO COMMON VARIABLES

600 common variables consisting of #100 to #199 and #500 to #999 can be used. Common variables #100 to #199 are cleared when the power is turned off. The values of common variables #500 to #999 are preserved even when the power is turned off.

5.33 CUSTOM MACRO COMMON VARIABLES BETWEEN EACH PATH

In a multi-path system, some or all custom macro common variables (#100 to #149 (#199) and #500 to #599 (#999)) can be made common to all paths by parameter setting.
Also, whether or not to make the variables common can be selected on a path-by-path basis by other parameter setting.

5.34 INTERRUPTION TYPE CUSTOM MACRO

When a program is being executed, another program can be called by inputting an interrupt signal (UINT) from the machine.
This function is referred to as an interruption type custom macro function.

Format

M96 Pxxxxxxxx ;	Enables custom macro interrupt
M97 ;	Disables custom macro interrupt

Explanation

Use of the interruption type custom macro function allows the user to call a program during execution of an arbitrary block of another program. This allows programs to be operated to match situations which vary from time to time.

(1) When a tool abnormality is detected, processing to handle the abnormality is started by an external signal.
(2) A sequence of machining operations is interrupted by another machining operation without the cancellation of the current operation.
(3) At regular intervals, information on current machining is read.

:

Listed above are examples like adaptive control applications of the interruption type custom macro function.

Example)

When M96 Pxxxx is specified in a program, subsequent program operation can be interrupted by an interrupt signal (UINT) input to execute the program specified by Pxxxx in following figure. When the interrupt signal (UINT, marked with an asterisk (*) in figure) is input during execution of the interrupt program or after M97, it is ignored.

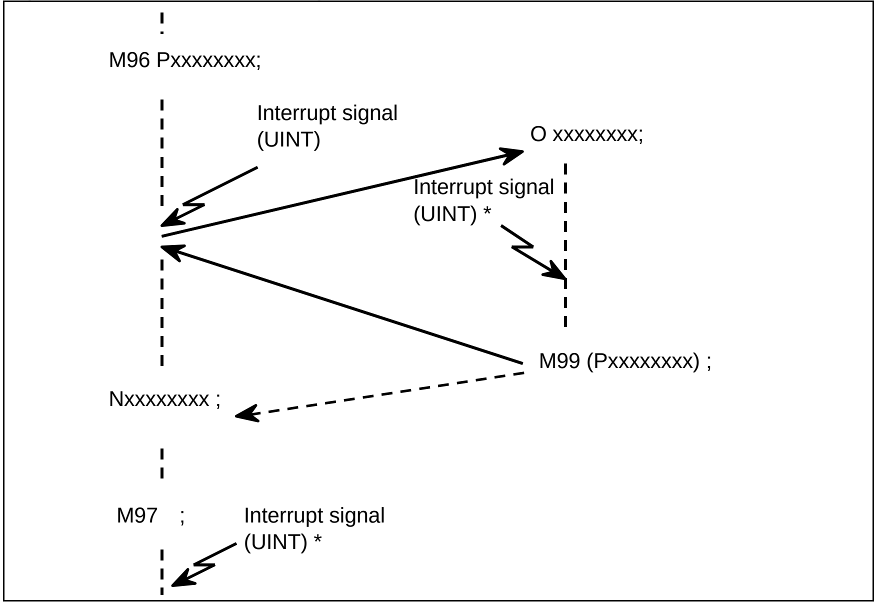


Fig. 5.34 (a)

5.35 CANNED CYCLE (T SERIES)

T

There are three canned cycles : the outer diameter/internal diameter cutting canned cycle (G77), the threading canned cycle (G78), and the end face turning canned cycle (G79).

5.35.1 Outer Diameter/Internal Diameter Cutting Cycle

Format

- **Straight cutting cycle**

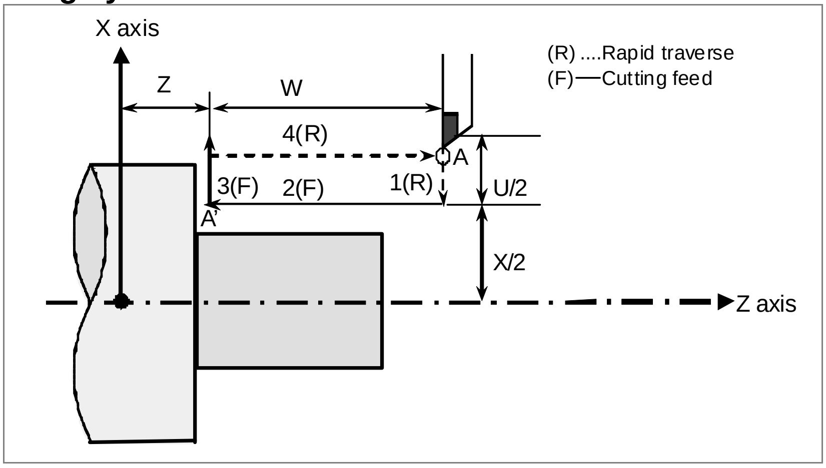


Fig. 5.35.1 (a)

G77 X(U)_ Z(W)_ F_ ; (G90 with G code system A)

X_,Z_ : Coordinates of the cutting end point (point A' in the Fig. 5.35.1 (a)) in the direction of the length

U_,W_ : Travel distance to the cutting end point (point A' in the Fig. 5.35.1 (a)) in the direction of the length

F_ : Cutting feedrate

- **Taper cutting cycle**

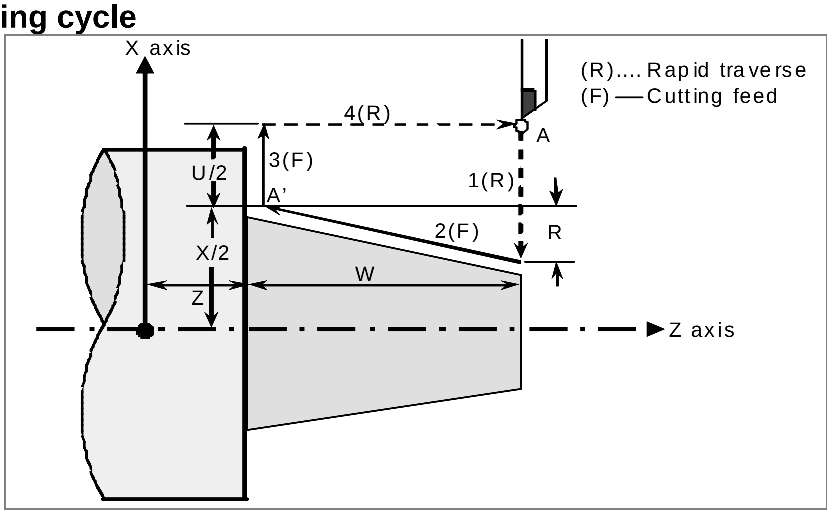


Fig. 5.35.1 (b)

G77 X(U)_ Z(W)_ R_ F_ ; (G90 with G code system A)

X_,Z_ : Coordinates of the cutting end point (point A' in the Fig. 5.35.1 (b)) in the direction of the length
U_,W_ : Travel distance to the cutting end point (point A' in the Fig. 5.35.1 (b)) in the direction of the length
R_ : Taper amount (R in the above figure)
F_ : Cutting feedrate

5.35.2 Threading Cycle

Format

- **Straight threading cycle**

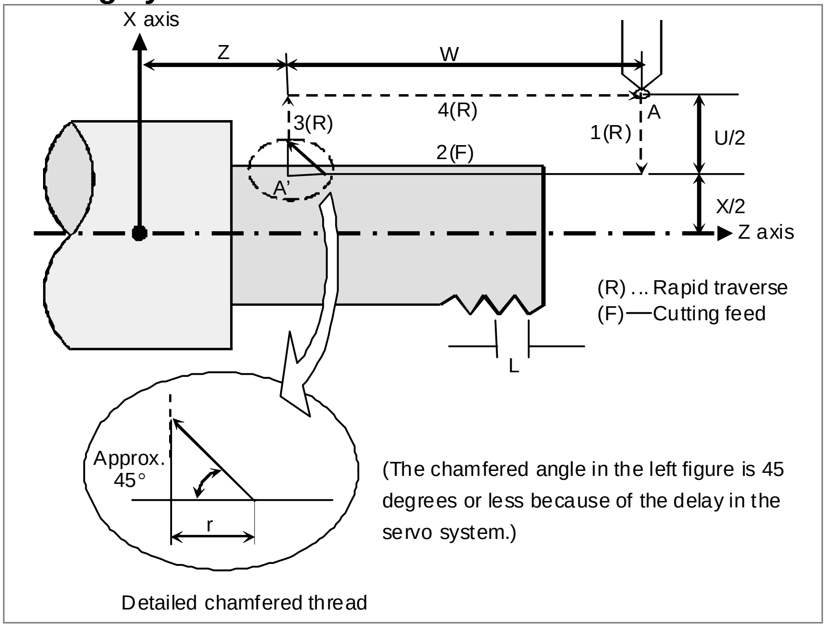


Fig. 5.35.2 (a)

G78 X(U)_ Z(W)_ F_ Q_ ; (G92 with G code system A)

X_,Z_ : Coordinates of the cutting end point (point A' in the Fig. 5.35.2 (a)) in the direction of the length
U_,W_ : Travel distance to the cutting end point (point A' in the Fig. 5.35.2 (a)) in the direction of the length
Q_ : Angle for shifting the threading start angle
(Increment: 0.001 degrees, Valid setting range: 0 to 360 degrees)
F_ : Thread lead (L in the above figure)

- Taper threading cycle

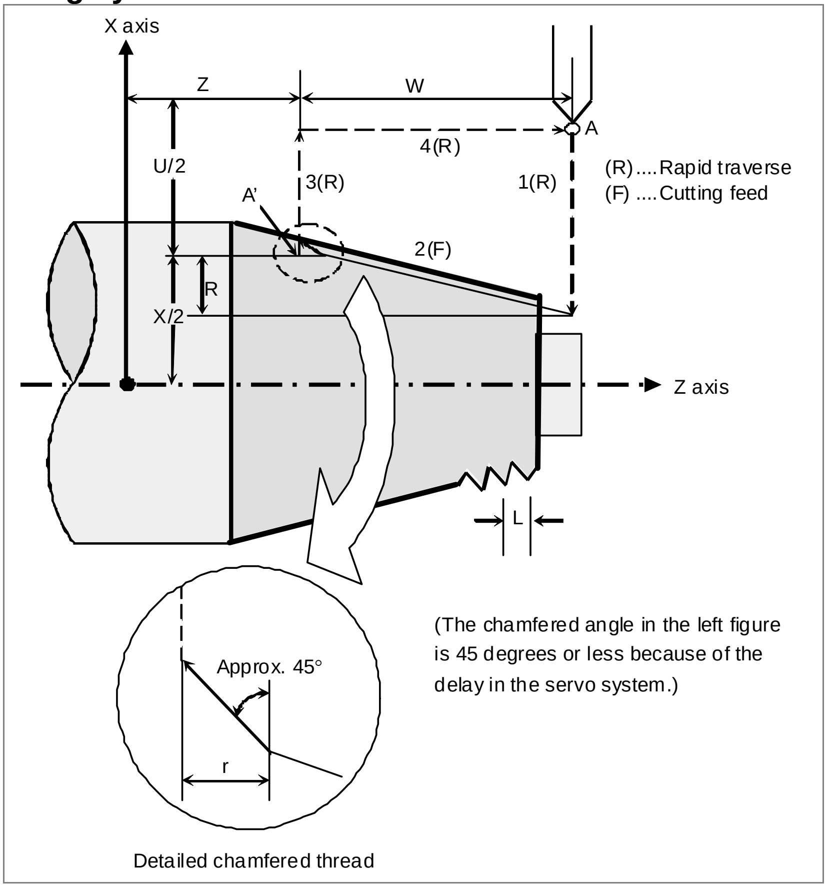


Fig. 5.35.2 (b)

G78 X(U)_ Z(W)_ R_ F_ Q_ ; (G92 with G code system A)

X_,Z_ : Coordinates of the cutting end point (point A' in the Fig. 5.35.2 (b)) in the direction of the length
U_,W_ : Travel distance to the cutting end point (point A' in the Fig. 5.35.2 (b)) in the direction of the length
Q_ : Angle for shifting the threading start angle (Increment: 0.001 degrees, Valid setting range: 0 to 360 degrees)
R_ : Taper amount (R in the above figure)
F_ : Thread lead (L in the above figure)

5.35.3 End Face Turning Cycle

Format

- **Face cutting cycle**

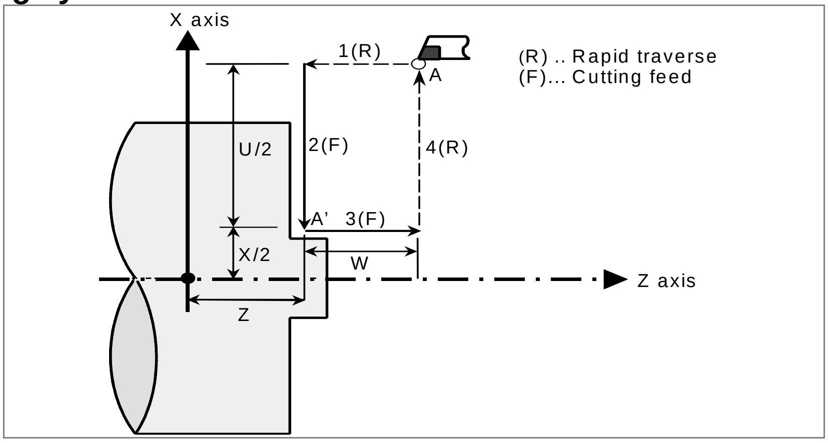


Fig. 5.35.3 (a)

G79 X(U)_ Z(W)_ F_ ; (G94 with G code system A)

X_,Z_ : Coordinates of the cutting end point (point A' in the Fig. 5.35.3 (a)) in the direction of the end face
U_,W_ : Travel distance to the cutting end point (point A' in the Fig. 5.35.3 (a)) in the direction of the end face
F_ : Cutting feedrate

- **Taper cutting cycle**

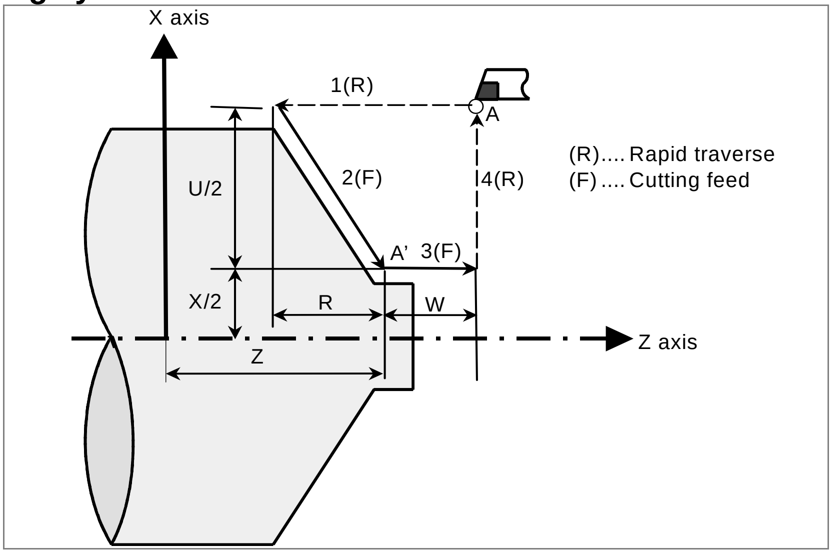


Fig. 5.35.3 (b)

G79 X(U)_ Z(W)_ R_ F_ ; (G94 with G code system A)

X_,Z_ : Coordinates of the cutting end point (point A' in the Fig. 5.35.3 (b)) in the direction of the end face
U_,W_ : Travel distance to the cutting end point (point A' in the Fig. 5.35.3 (b)) in the direction of the end face
R_ : Taper amount (R in the above figure)
F_ : Cutting feedrate

5.36 MULTIPLE REPETITIVE CYCLE (T SERIES)

T

This function is canned cycles to make CNC programming easy. For instance, the data of the target figure (finishing shape) describes the tool path for rough machining. And also, a canned cycles for the threading is available.

5.36.1 Stock Removal in Turning

There are two types of stock removals in turning : Type I and II.
To specify the stock removal in turning, command G71 in T series (G-code system A) and G71.7 in M series.

Format

ZpXp plane

G71 U(Δd) R(e) ;

G71 P(ns) Q(nf) U(Δu) W(Δw) F(f) S(s) T(t) ;

N (ns) ;

...

N (nf) ;

The move command between A and B is specified in the blocks from sequence number ns to nf.

Δd : Depth of cut
The cutting direction depends on the direction AA'.
e : Escaping amount
ns : Sequence number of the first block for the program of target figure (finishing shape)
nf : Sequence number of the last block for the program of target figure (finishing shape)
Δu : Distance of the finishing allowance in the direction of the second axis on the plane (X-axis for the ZX plane)
Δw : Distance of the finishing allowance in the direction of the first axis on the plane (Z-axis for the ZX plane)
f,s,t : Any F , S, or T function contained in blocks ns to nf in the cycle is ignored, and the F, S, or T function in this G71 block is effective.

Explanation

- Type I

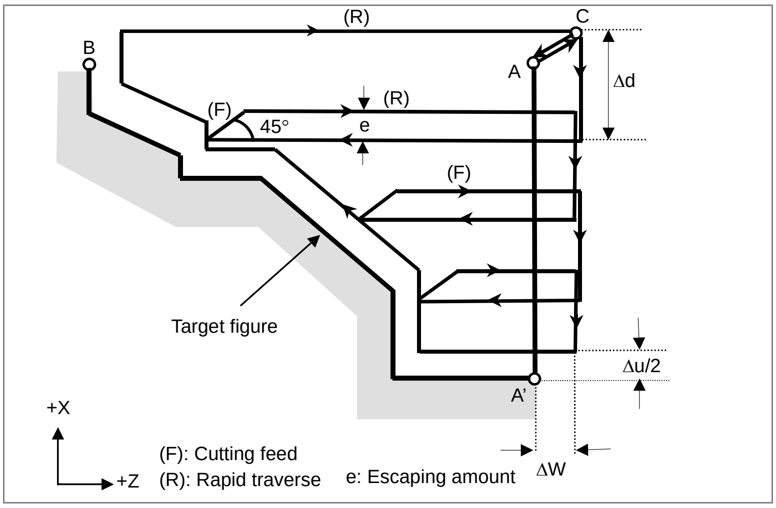


Fig. 5.36.1 (a)

- Type II

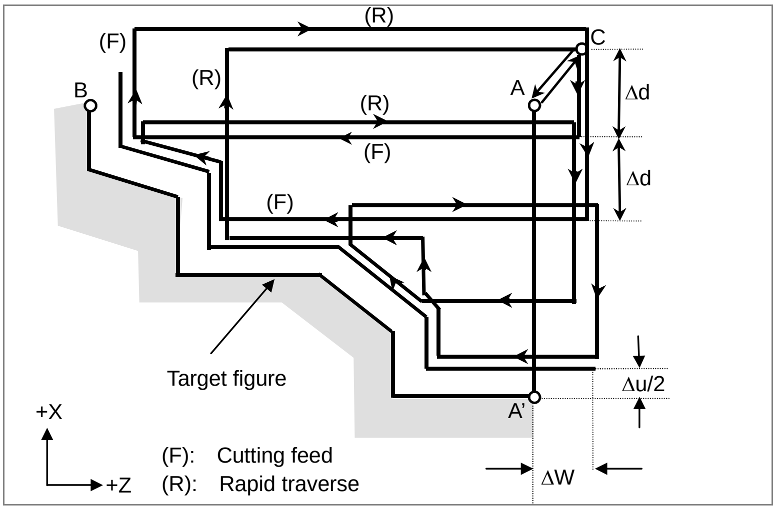


Fig. 5.36.1 (b)

When the target figure of A → A' → B is programmed, cutting is performed with a depth of cut of Δd per cut, leaving the finishing allowances Δu/2 and Δw. After the last cut is performed in the plane second axis direction (X-axis direction in the case of the ZX plane), rough finishing is performed along the target figure. Upon completion of rough finishing, the block that follows the sequence block specified by Q is executed.

- Selection of type I or II

When the target figure has pockets, be sure to use type II.

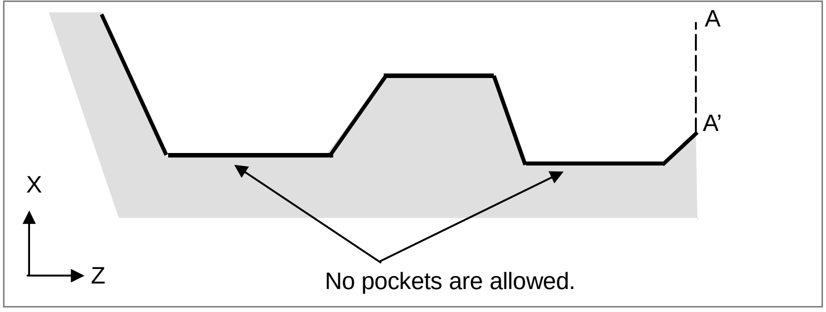


Fig. 5.36.1 (c)

Escaping operation after rough cutting in the direction of the first axis on the plane (Z-axis for the ZX plane) differs between types I and II. With type I, the tool escapes to the direction of 45 degrees. With type II, the tool cuts the workpiece along the target figure. When the target figure has no pockets, determine the desired escaping operation and select type I or II.

- Selecting type I or II

In the start block for the target figure (sequence number ns), select type I or II.

(1) When type I is selected
Specify the second axis on the plane (X-axis for the ZX plane). Do not specify the first axis on the plane (Z-axis for the ZX plane).

(2) When type II is selected
Specify the second axis on the plane (X-axis for the ZX plane) and first axis on the plane (Z-axis for the ZX plane).
When you want to use type II without moving the tool along the first axis on the plane (Z-axis for the ZX plane), specify the incremental programming with travel distance 0 (W0 for the ZX plane).

Example)

Type I

```
G71 10.0 R5.0 ;
G71 P100 Q200....;
N100 X(U)_ ;
    :
    :
N200.............;
```

Type II

```
G71 10.0 R5.0 ;
G71 P100 Q200....;
N100 X(U)_ Z(W)_;
    :
    :
N200.............;
```

5.36.2 Stock Removal in Facing

This cycle is the same as G71 except that cutting is performed by an operation parallel to the second axis on the plane (X-axis for the ZX plane).
To specify the stock removal in facing, command G72 in T series (G-code system A) and G72.7 in M series.

Format

ZpXp plane

G72 W(Δd) R(e) ;

G72 P(ns) Q(nf) U(Δu) W(Δw) F(f) S(s) T(t) ;

N (ns) ;
...
N (nf) ;

The move command between A and B is specified in the blocks from sequence number ns to nf.

Δd : Depth of cut
The cutting direction depends on the direction AA'.
e : Escaping amount
ns : Sequence number of the first block for the program of target figure (finishing shape)
nf : Sequence number of the last block for the program of target figure (finishing shape)
Δu : Distance of the finishing allowance in the direction of the second axis on the plane (X-axis for the ZX plane)
Δw : Distance of the finishing allowance in the direction of the first axis on the plane (Z-axis for the ZX plane)
f,s,t : Any F , S, or T function contained in blocks ns to nf in the cycle is ignored, and the F, S, or T function in this G72 block is effective.

Explanation

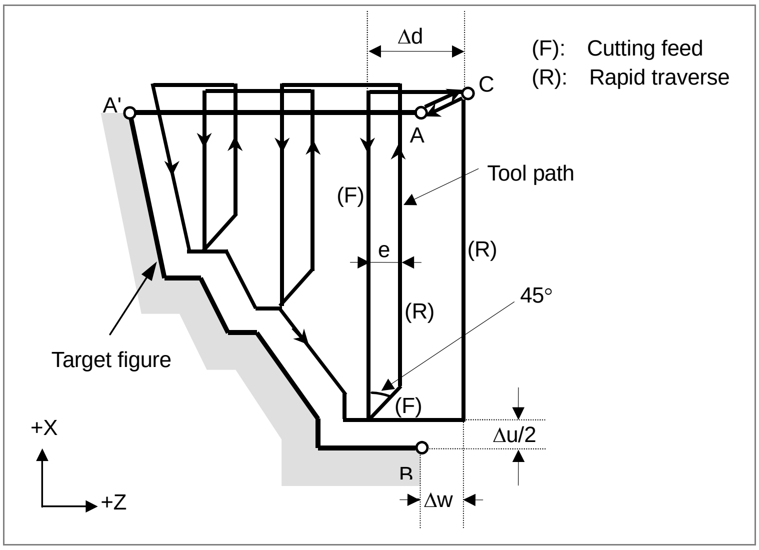


Fig. 5.36.2 (a)

When a target figure passing through A, A', and B in this order is given by a program, the specified area is removed by Δd (depth of cut), with the finishing allowance specified by Δu/2 and Δw left.

- Selection of type I or II

For G72, there are types I and II.

When the target figure has pockets, be sure to use type II.

Escaping operation after rough cutting in the direction of the second axis on the plane (X-axis for the ZX plane) differs between types I and II. With type I, the tool escapes to the direction of 45 degrees. With type II, the tool cuts the workpiece along the target figure. When the target figure has no pockets, determine the desired escaping operation and select type I or II.

- Selecting type I or II

In the start block for the target figure (sequence number ns), select type I or II.

(1) When type I is selected
Specify the first axis on the plane (Z-axis for the ZX plane). Do not specify the second axis on the plane (X-axis for the ZX plane).

(2) When type II is selected
Specify the second axis on the plane (X-axis for the ZX plane) and first axis on the plane (Z-axis for the ZX plane).
When you want to use type II without moving the tool along the second axis on the plane (X-axis for the ZX plane), specify the incremental programming with travel distance 0 (U0 for the ZX plane).
Example)

```
Type I    Type II
G72 10.0 R5.0 ;              G72 10.0 R5.0 ;
G72 P100 Q200....;           G72 P100 Q200....;
N100 Z(W)_ ;                 N100 X(U)_ Z(W)_;
   :
   :
N200..............;          N200..............;
```

- Differences from G71 (Type I)

G72 differs from G71 in the following points:

(1) G72 cuts the workpiece with moving the tool in parallel with the second axis on the plane (X-axis on the ZX plane).

(2) In the start block in the program for a target figure (block with sequence number ns), only the first axis on the plane (Z-axis (W-axis) for the ZX plane) must be specified.

- Differences from G71 (Type II)

G72 differs from G71 in the following points:

(1) G72 cuts the workpiece with moving the tool in parallel with the second axis on the plane (X-axis on the ZX plane).

(2) The figure need not show monotone increase or decrease in the direction of the first axis on the plane (Z-axis for the ZX plane) and it may have concaves (pockets). The figure must show monotone change in the direction of the second axis on the plane (X-axis for the ZX plane), however.

(3) When a position parallel to the second axis on the plane (X-axis for the ZX plane) is specified in a block in the program for the target figure, it is assumed to be at the bottom of a pocket.

(4) After all rough cutting terminates along the second axis on the plane (X-axis for the ZX plane), the tool temporarily returns to the start point. Then, rough cutting as finishing is performed.

5.36.3 Pattern Repeating

This function permits cutting a fixed pattern repeatedly, with a pattern being displaced bit by bit. By this cutting cycle, it is possible to efficiently cut working whose rough shape has already been made by a rough machining, forging or casting method, etc.

To specify the pattern repeating, command G73 in T series (G-code system A) and G73.7 in M series.

Format

ZpXp plane

G73 W(Δk) U(Δi) R(d) ;

G73 P(ns) Q(nf) U(Δu) W(Δw) F(f) S(s) T(t) ;

N (ns) ;

...

N (nf) ;

The move command between A and B is specified in the blocks from sequence number ns to nf.

Δi : Distance of escape in the direction of the second axis on the plane (X-axis for the ZX plane)

Δk : Distance of escape in the direction of the first axis on the plane (Z-axis for the ZX plane)

d : The number of division (This value is the same as the repetitive count for rough cutting.)

ns : Sequence number of the first block for the program of target figure (finishing shape)

nf : Sequence number of the last block for the program of target figure (finishing shape)

Δu : Distance of the finishing allowance in the direction of the second axis on the plane (X-axis for the ZX plane)

Δw : Distance of the finishing allowance in the direction of the first axis on the plane (Z-axis for the ZX plane)

f, s, t : Any F, S, and T function contained in the blocks between sequence number "ns" and "nf" are ignored, and the F, S, and T functions in this G73 block are effective.

Explanation

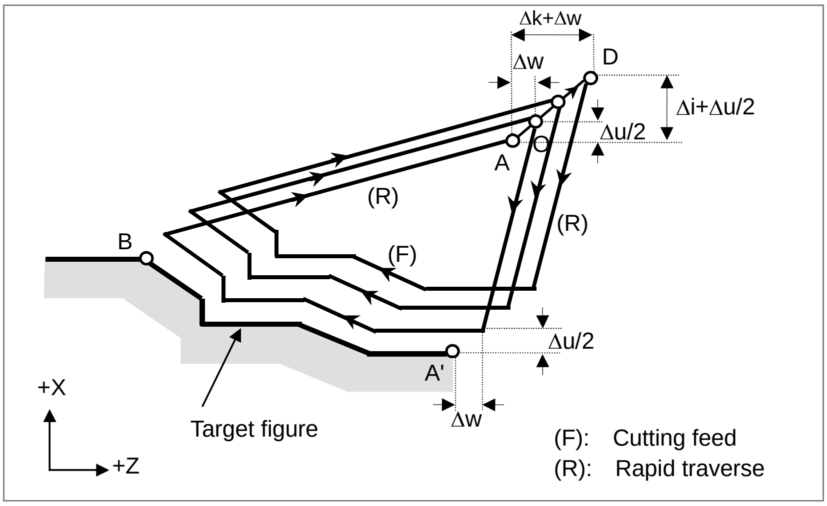


Fig. 5.36.3 (a)

When a target figure passing through A, A', and B in this order is given by a program, rough cutting is performed the specified number of times, with the finishing allowance specified by Δu/2 and Δw left.

5.36.4 Finishing Cycle

After rough cutting by G71, G72 or G73, the following command permits finishing.
To specify the finishing cycle, command G70 in T series (G-code system A) and G70.7 in M series.

Format

G70 P(ns) Q(nf) ;

ns : Sequence number of the first block for the program of target figure (finishing shape)
nf : Sequence number of the last block for the program of target figure (finishing shape)

Explanation

The blocks with sequence numbers ns to nf in the program for a target figure are executed for finishing. The F, S, T, M, and second auxiliary functions specified in the G71, G72, or G73 block are ignored and the F, S, T, M, and second auxiliary functions specified in the blocks with sequence numbers ns to nf are effective.
When cycle operation terminates, the tool is returned to the start point in rapid traverse and the next G70 cycle block is read.

5.36.5 End Face Peck Drilling Cycle

The operation shown in the Fig. 5.36.5 (a) is performed according to the command described below. This function enables chip breaking in outer diameter cutting. If the second axis on the plane (X-axis (U-axis) for the ZX plane) and address P are omitted, operation is performed only along the first axis on the plane (Z-axis for the ZX plane), that is, a peck drilling cycle is performed.
To specify the end face peak drilling cycle, command G74 in T series (G-code system A) and G74.7 in M series.

Format

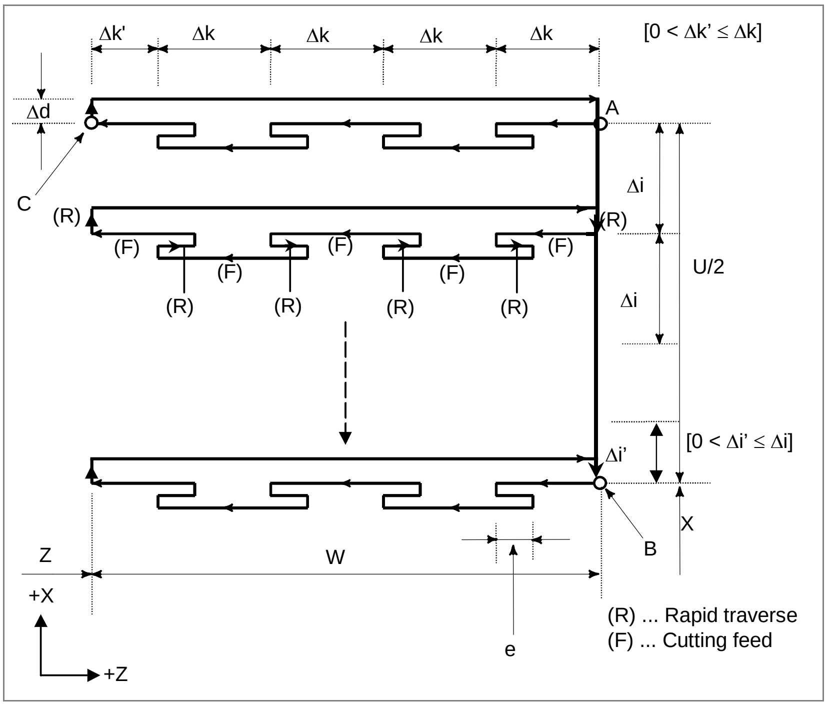


Fig. 5.36.5 (a)

G74R (e) ;

G74X(U)_ Z(W)_ P(Δi) Q(Δk) R(Δd) F (f) ;

e	: Return amount
X_,Z_	: Coordinate of the second axis on the plane (X-axis for the ZX plane) at point B and coordinate of the first axis on the plane (Z-axis for the ZX plane) at point C
U_,W_	: Travel distance along the second axis on the plane (U for the ZX plane) from point A to B and travel distance along the first axis on the plane (W for the ZX plane) from point A to C
Δi	: Travel distance in the direction of the second axis on the plane (X-axis for the ZX plane)
Δk	: Depth of cut in the direction of the first axis on the plane (Z-axis for the ZX plane)
Δd	: Escaping amount of the tool at the cutting bottom
f	: Feedrate

5.36.6 Outer Diameter / Internal Diameter Drilling Cycle

The operation shown in the Fig. 5.36.6 (a) is performed according to the command described below. This cycle is equivalent to G74 except that the second axis on the plane (X-axis for the ZX plane) changes places with the first axis on the plane (Z-axis for the ZX plane). This cycle enables chip breaking in end facing. It also enables grooving during outer diameter cutting and cutting off (when the Z-axis (W-axis) and Q are omitted for the first axis on the plane).

To specify the outer diameter / internal diameter drilling cycle, command G75 in T series (G-code system A) and G75.7 in M series.

Format

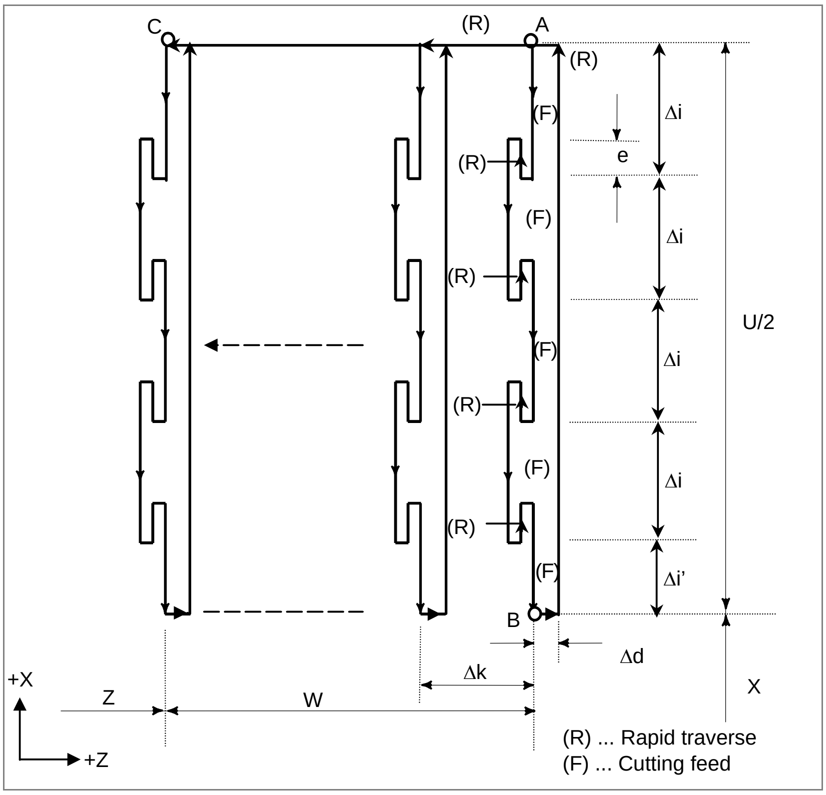


Fig. 5.36.6 (a)

G75R (e) ;

G75X(U)_ Z(W)_ P(Δi) Q(Δk) R(Δd) F (f) ;

e : Return amount

X_,Y_,Z_ : Coordinate of the second axis on the plane (X-axis for the ZX plane) at point B and coordinate of the first axis on the plane (Z-axis for the ZX plane) at point C

(U_,V_,W_) : Travel distance along the second axis on the plane (U for the ZX plane) from point A to B and travel distance along the first axis on the plane (W for the ZX plane) from point A to C

ΔI : Depth of cut in the direction of the second axis on the plane (X-axis for the ZX plane)

Δk : Travel distance in the direction of the first axis on the plane (Z-axis for the ZX plane)

Δd : Escaping amount of the tool at the cutting bottom

f : Feedrate

5.36.7 Multiple Threading Cycle

The threading cycle with a constant depth of cut using a one-side cutter as shown in the figure below is executed according the command described below (Fig. 5.36.7 (a)).
To specify the multiple threading cycle, command G76 in T series (G-code system A) and G76.7 in M series.

Format

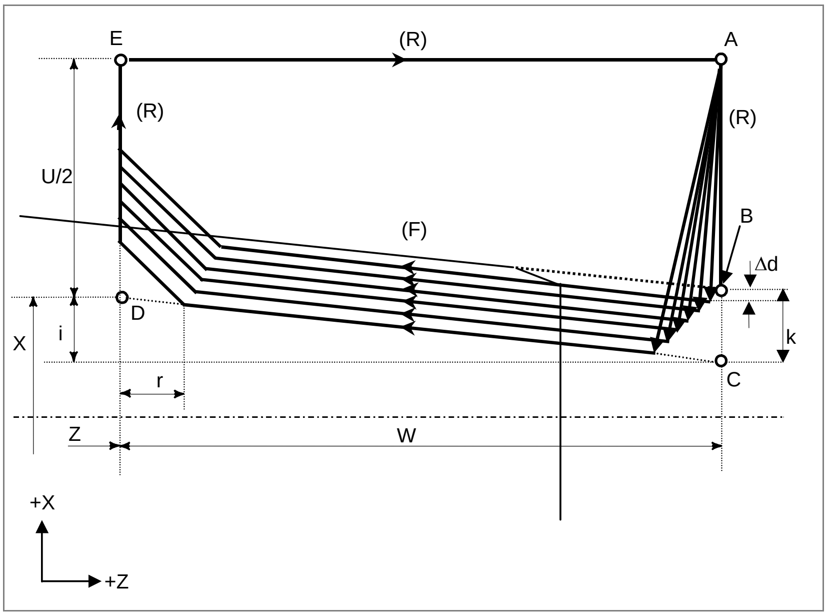


Fig. 5.36.7 (a)

G76 P(m) (r) (a) Q(Δdmin) R(d) ;

G76 X(U)_ Z(W)_ R(i) P(k) Q(Δd) F (L) ;

m : Repetitive count in finishing (1 to 99)
r : Chamfering amount (0 to 99)
a : Angle of tool nose
One of six kinds of angle, 80°, 60°, 55°, 30°, 29°, and 0°, can be selected, and specified by 2-digit number.
m, r, and a are specified by address P at the same time.
(Example) When m=2, r=1.2L, a=60°, specify as shown below (L is lead of thread).

```
P   02   12   60
    |    |    └── a
    |    └── r
    └── m
```

Δdmin : Minimum cutting depth
d : Finishing allowance
X_, Z_ : Coordinates of the cutting end point (point D in the Fig. 5.36.7 (a)) in the direction of the length
U_, W_: Travel distance to the cutting end point (point D in the Fig. 5.36.7 (a)) in the direction of the length
i : Taper amount (If i = 0, ordinary straight threading can be made.)
k : Height of thread
Δd : Depth of cut in 1st cut
L : Lead of thread

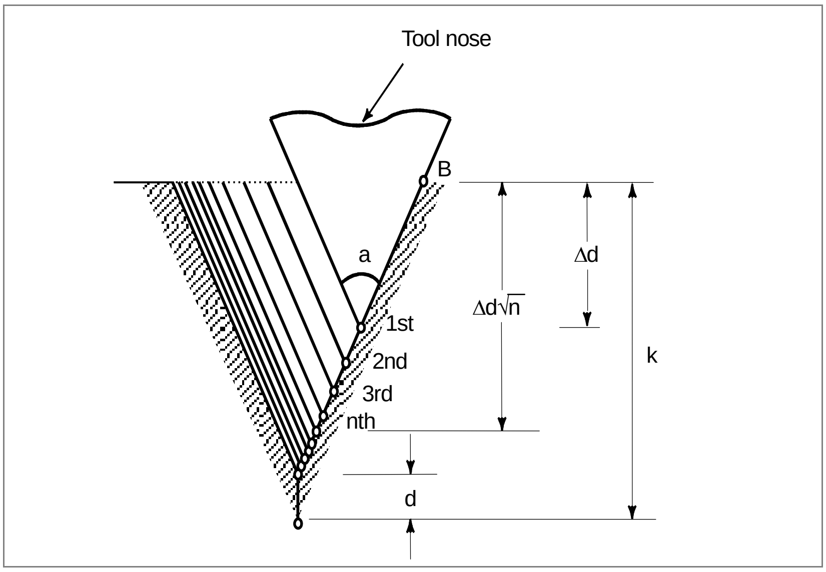


Fig. 5.36.7 (b) Detail of cutting

5.37 IN-FEED CONTROL (FOR GRINDING MACHINE) (M SERIES)

M

Overview

Each time an external signal is input at the table swing end point, a workpiece is cut by a constant depth of cut along a programmed figure on the specified Y-Z plane. This makes it possible to perform grinding and cutting in a timely manner and facilitating the grinding of a workpiece with a profile.

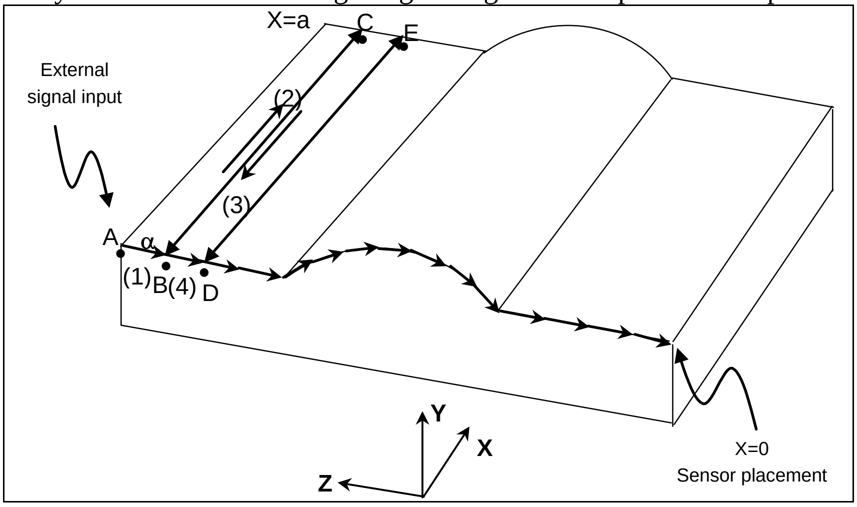


Fig. 5.37 (a)

For example, it is possible to machine a workpiece with a profile programmed with linear interpolation, circular interpolation, and linear interpolation on the YZ plane, such as that shown in the Fig. 5.37 (a).

A sensor is placed at a X = 0 position so that the external signal is input when the sensor detects the grinding wheel. When the program is started at point A, the machine is first placed in the state in which it waits for the input of the external signal. Then, when the sensor detects the grinding wheel, the external signal is input, and the machine makes a cut by the constant amount α along the programmed profile on the specified YZ plane and moves to point B (operation (1)). The machine is then placed in the state in which it waits for the input of the external signal again, and performs a grinding operation along the X-axis. It grinds from point B to point C (operation (2)) and grinds back from point C to point B (operation (3)). When the machine returns to point B, the sensor detects the grinding wheel again, and the external signal is input, so that the machine makes a cut by the amount of α and moves to point D (operation (4)). At point D, the machine performs a grinding operation along the X-axis.

Afterwards, each time the external signal is input, the machine makes a cut by the amount of α along the profile program, so that the workpiece is machined to a profile such as that shown in the Fig. 5.37 (a)..

NOTE

This function is included in the option "Grinding function A".

5.38 CANNED GRINDING CYCLE (FOR GRINDING MACHINE)

Overview

With the canned grinding cycle, repetitive machining operations that are specific to grinding and are usually specified using several blocks can be specified using one block including a G function. So, a program can be created simply. At the same time, the size of a program can be reduced, and the memory can be used more efficiently. Four types of canned grinding cycles are available:

T

- Traverse grinding cycle
- Traverse direct constant-size grinding cycle
- Oscillation grinding cycle
- Oscillation direct constant-size grinding cycle

M

- Plunge grinding cycle
- Direct constant-dimension plunge grinding cycle
- Continuous-feed surface grinding cycle
- Intermittent-feed surface grinding cycle

NOTE

This function is included in the option "Grinding function A" and "Grinding function B".

5.39 CANNED CYCLE FOR DRILLING

M

The canned cycles for drilling are functions for performing machining operations such as boring, drilling, and tapping with a more simplified command.

The table below indicates the relationships between positioning planes and drilling axes.

Table 5.39 (a)

G code	Positioning plane	Drilling axis
G17	Xp-Yp plane	Zp
G18	Zp-Xp plane	Yp
G19	Yp-Zp plane	Xp

Xp: X axis or an axis parallel to the X axis
Yp: Y axis or an axis parallel to the Y axis
Zp: Z axis or an axis parallel to the Z axis

The following canned cycles for drilling are available:

Table 5.39 (b)

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

NOTE

The 3 basic axes (X, Y, and Z) are needed.

T

When the canned cycles for drilling are used, a machining operation specified using several blocks can be specified using a single block including a G function, making programming much simpler.

The Table 5.39 (c) indicates the relationships between positioning axes and drilling axes.

Table 5.39 (c)

G code	Positioning axis	Drilling axis
G83, G84, G85	X axis, C axis	Z axis
G87, G88, G89	Z axis, C axis	X axis

The canned cycles for drilling in the Table 5.39 (d) are available :

Table 5.39 (d)

G code	Drilling axis	Drilling	Operation at the bottom of a hole	Retraction	Application
G80					Cancel
G83	Z axis	Intermittent feed Feed	Dwell	Rapid traverse	Front drilling cycle
G84	Z axis	Feed	Dwell → Spindle CCW	Feed	Front tapping cycle
G85	Z axis	Feed	Dwell	Feed	Front boring cycle
G87	X axis	Intermittent feed Feed	Dwell	Rapid traverse	Side drilling cycle
G88	X axis	Feed	Dwell → Spindle CCW	Feed	Side tapping cycle
G89	X axis	Feed	Dwell	Feed	Side boring cycle

NOTE

X,Z in 3 basic axes are needed.

5.40 CIRCULAR INTERPOLATION BY R PROGRAMMING

In the case of circular interpolation command, 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.)

Example)

For arc <1> (less than 180°)
G91 G02 X60.0 Y55.0 R50.0 F300.0 ;

For arc <2> (greater than 180°)
G91 G02 X60.0 Y55.0 R-50.0 F300.0 ;

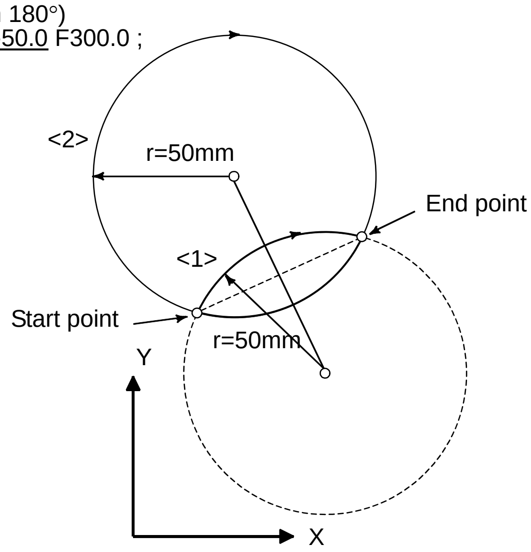


Fig. 5.40 (a)

5.41 MIRROR IMAGE FOR DOUBLE TURRET (T SERIES)

T

By applying mirror image to the X-axis with a G code, a machining program for the opposite tool post can be created for symmetric cutting as if the program were created in the coordinate system on the same side.

Format

G68 : Double turret mirror image on
G69 : Mirror image cancel

Explanation

When G68 is designated, the coordinate system is shifted to the double turret side, and the X-axis sign is reversed from the programmed command to perform symmetrical cutting.
To use this function, set the distance between the two tool posts to a parameter.

Example

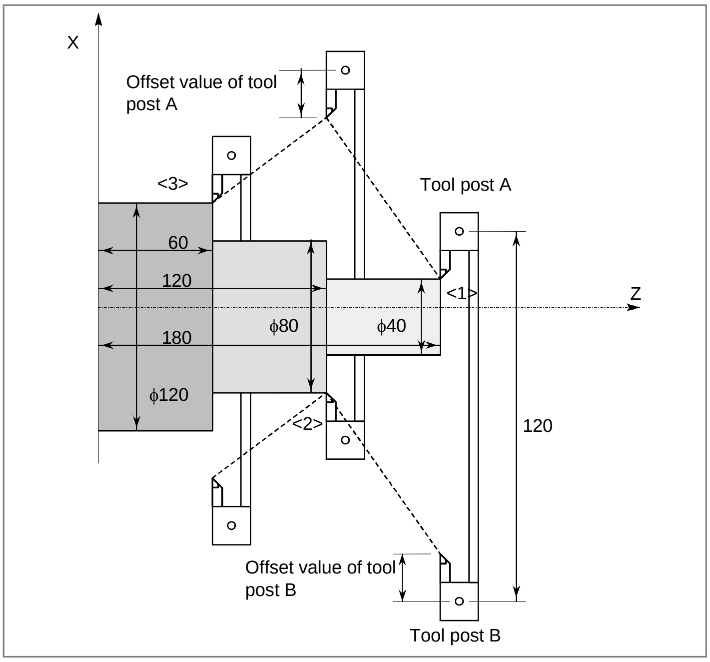


Fig. 5.41 (a)

X40.0 Z180.0 T0101 ;	Position tool post A at <1>
G68 ;	Shift the coordinate system by the distance A to B (120mm), and turn mirror image on.
X80.0 Z120.0 T0202 ;	Position tool post B at <2>
G69 ;	Shift the coordinate system by the distance B to A, and cancel mirror image.
X120.0 Z60.0 T0101 ;	Position tool post A at <3>

(*) In this example, a diameter value is specified for the X-axis.

5.42 AUTOMATIC CORNER OVERRIDE (M SERIES)

M

When G62 is commanded during cutter compensation, cutting feed rate is automatically overridden at corner. The cutting quantity per unit time of the corner is thus controlled not to increase. This G62 is valid till G61 (exact stop mode), G64 (cutting mode), or G63 (tapping mode) is commanded.

5.43 SCALING (M SERIES)

M

A programmed figure can be magnified or reduced (scaling).
The magnification rate can be specified in the program.
Unless specified in the program, the magnification rate specified in the parameter is applied.
Two types of scaling are available, one in which the same magnification rate is applied to each axis and the other in which different magnification rates are applied to different axes. Which type to use is determined by parameter setting.

Format

- Scaling up or down along all axes at the same rate of magnification

G51 IP_ P_ ; Scaling start
: Scaling is effective.
: (Scaling mode)
G50 ; Scaling cancel

IP_ : Absolute programming for center coordinate value of scaling
P : Scaling magnification

- Scaling of each axis

G51 IP_ I_ J_ K_ ; Scaling start
: Scaling is effective.
: (Scaling mode)
G50 ; Scaling cancel

IP_ : Absolute programming for center coordinate value of scaling
I_J_K_ : Scaling magnification for basic 3 axes (X, Y, and Z axes) respectively

⚠ **CAUTION**

1 Specify G51 in a separate block.
2 After the figure is enlarged or reduced, specify G50 to cancel the scaling mode.

Explanation

- Scaling along each axis at the same rate of magnification

A move command in the scaling mode is scaled by the magnification specified by P_, with the point specified by X_Y_Z_ used as the center.

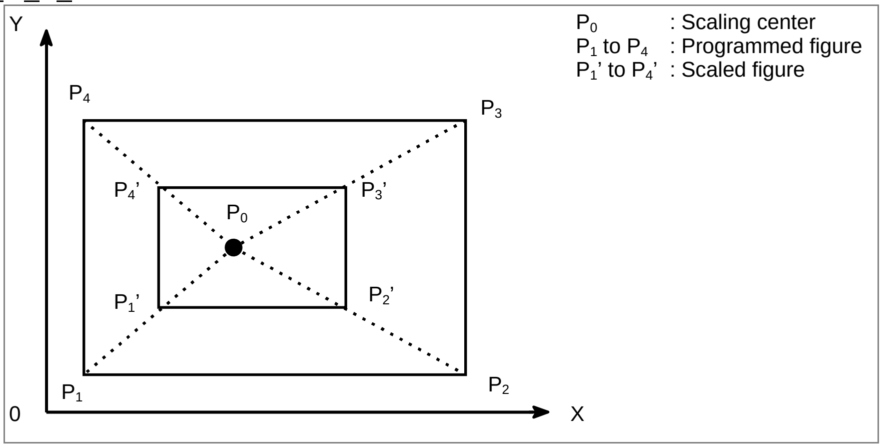


Fig. 5.43 (a)

- Scaling of each axis

A move command in the scaling mode is scaled by the magnification for each axis specified by I_J_K_, with the point specified by X_Y_Z_ used as the center. If I, J, and K are not specified, scaling is performed by a parameter-set magnification.

Also when a negative magnification is specified, a mirror image is applied. The axis subject to the mirror image is the one that contains the scaling center.

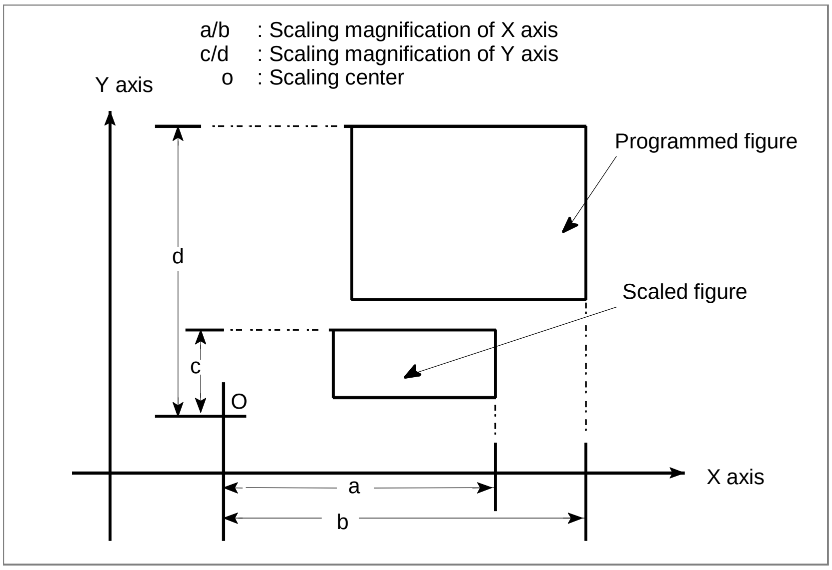


Fig. 5.43 (b)

- Scaling center

Even in incremental command (G91) mode, the scaling center coordinates IP_ specified in the G51 block are assumed those of an absolute position.
If the scaling center coordinates are omitted, the position assumed when G51 is specified is assumed the scaling center.

5.44 COORDINATE SYSTEM ROTATION

A programmed shape can be rotated. By using this function it becomes possible, for example, to modify a program using a rotation command when a workpiece has been placed with some angle rotated from the programmed position on the machine. Further, when there is a pattern comprising some identical shapes in the positions rotated from a shape, the time required for programming and the length of the program can be reduced by preparing a sub program of the shape and calling it after rotation.

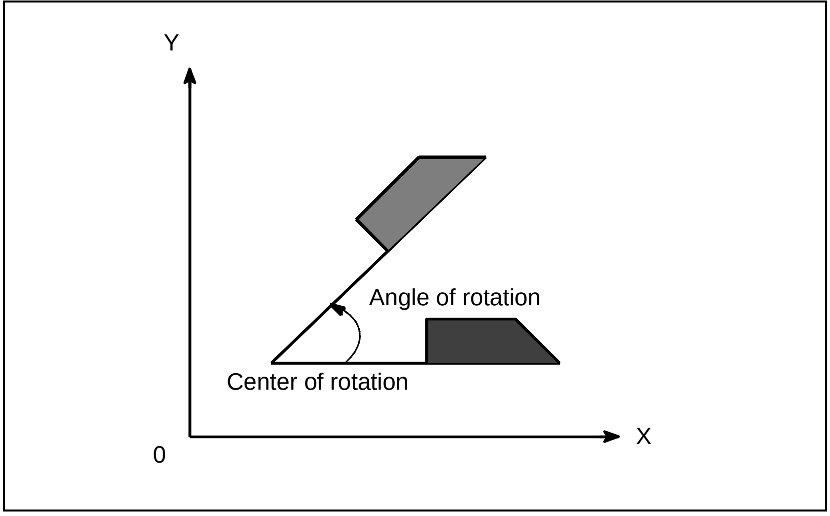


Fig. 5.44 (a)

Format

For M series

$\left\{\begin{array}{l}\textbf{G17}\\ \textbf{G18}\\ \textbf{G19}\end{array}\right\}$ **G68 α_ β_ R_ ;**	Start rotation of a coordinate system.
:	Coordinate system rotation mode
G69 ;	Coordinate system rotation cancel command

For T series

$\left\{\begin{array}{l}\textbf{G17}\\ \textbf{G18}\\ \textbf{G19}\end{array}\right\}$ **G68.1 α_ β_ R_ ;**	Start rotation of a coordinate system.
:	Coordinate system rotation mode
G69.1 ;	Coordinate system rotation cancel command

G17 (G18 or G19) :Select the plane in which contains the figure to be rotated.

α, β : Specify two coordinates (from among X, Y, and Z) of the rotation center that match G17, G18, and G19. The values specified as the coordinates of the rotation center must always be absolute values.

R : Specifies the rotation angle as an absolute value. Counterclockwise rotation is assumed to be positive. However, setting parameter enables the use of an incremental value.
Unit : 0.001 deg
Valid data range : -360,000 to +360,000

Explanation

Once this function is specified, a subsequent command is rotated about the center specified by α_ β_ by the angle specified by R_.

The rotation plane (G17, G18, G19) selected when G68 (G68.1) was specified is used. G17, G18, or G19 may not be specified in the block including G68 (G68.1), but must not be specified in the coordinate system rotation mode.

If α_ β_ is omitted, the position where G68 (G68.1) was specified becomes the center of rotation.

- Relationship with 3-dimensional coordinate conversion

Both coordinate system rotation and 3-dimensional coordinate conversion use the same G codes. The G code with I, J, and K is processed as a command for 3-dimensional coordinate conversion. The G code without I, J, and K is processed as a command for 2-dimensional coordinate system rotation.

5.45 3-DIMENSIONAL COORDINATE CONVERSION

Coordinate conversion about an axis can be carried out if the center of rotation, direction of the axis of rotation, and angular displacement are specified. This function is very useful in 3-dimensional machining by a die-sinking machine or similar machine. For example, by executing a program created on the XY plane after applying 3-dimensional coordinate conversion, machining can be performed on an arbitrary plane in a 3-dimensional space.

By specifying rigid tapping in the 3-dimensional coordinate conversion mode, tapping operation can be performed in the angle direction specified by the 3-dimensional coordinate conversion command (3-dimensional rigid tapping).

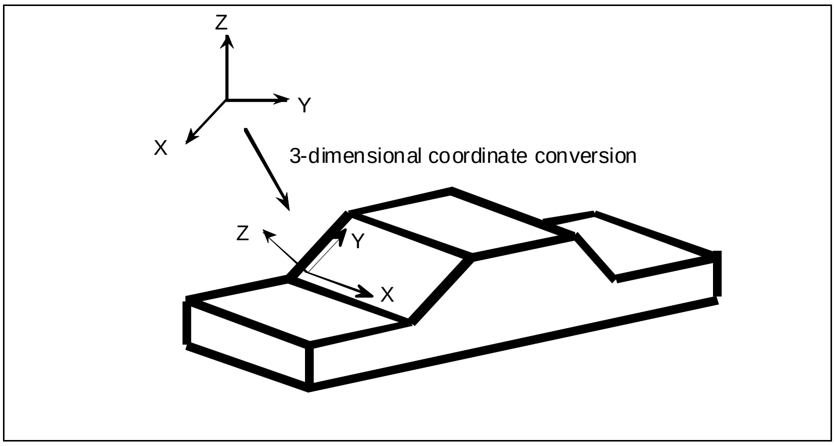


Fig. 5.45 (a) Image of 3-dimensional coordinate conversion

NOTE
This function is an optional function.

Format

For M series

G68 Xp$_{\underline{X1}}$ **Yp**$_{\underline{y1}}$ **Zp**$_{\underline{z1}}$ **I**$_{\underline{i1}}$ **J**$_{\underline{j1}}$ **K**$_{\underline{k1}}$ **R**$\underline{\alpha}$ **;**	Starting 3-dimensional coordinate conversion
: **:**	3-dimensional coordinate conversion mode
G69 ;	Canceling 3-dimensional coordinate conversion

For T series

G68.1 Xp$_{\underline{X1}}$ **Yp**$_{\underline{y1}}$ **Zp**$_{\underline{z1}}$ **I**$_{\underline{i1}}$ **J**$_{\underline{j1}}$ **K**$_{\underline{k1}}$ **R**$\underline{\alpha}$ **;**	Starting 3-dimensional coordinate conversion
: **:**	3-dimensional coordinate conversion mode
G69.1 ;	Canceling 3-dimensional coordinate conversion

Xp, Yp, Zp : Center of rotation (absolute coordinates) on the X, Y, and Z axis or parallel axes
I, J, K : Direction of the axis of rotation
R : Angular displacement

Example)

3-dimensional coordinate conversion can be executed twice.

N1 G68 Xp $_{\underline{x1}}$ Yp $_{\underline{y1}}$ Zp $_{\underline{z1}}$ I $_{\underline{i1}}$ J $_{\underline{j1}}$ K $_{\underline{k1}}$ R $\underline{\alpha}$;
N2 G68 Xp $_{\underline{x2}}$ Yp $_{\underline{y2}}$ Zp $_{\underline{z2}}$ I $_{\underline{i2}}$ J $_{\underline{j2}}$ K $_{\underline{k2}}$ R $\underline{\beta}$;
:
Nn G69 ;

5.46 TILTED WORKING PLANE INDEXING (M SERIES)

5.46.1 Tilted Working Plane Indexing (M Series)

M

When a figure such as a hole or pocket is cut on a plane tilted relative to the reference surface of the workpiece, this function defines a coordinate system (referred to a "feature coordinate system") tied to the plane. By specifying a position in such a coordinate system, a program can be created more easily.

A feature coordinate system is defined on a workpiece coordinate system.

See the Fig. 5.46 (a) for the relationship between a feature coordinate system and workpiece coordinate system.

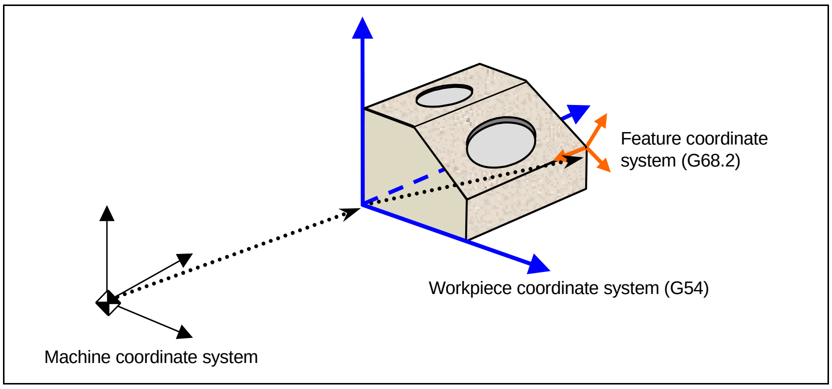


Fig. 5.46 (a)

This function is applicable to the following machine configurations. (See Fig. 5.46 (a).)

(1) Tool rotation type machine controlled with two tool rotation axes
(2) Table rotation type machine controlled with two table rotation axes
(3) Mixed-type machine controlled with one tool rotation axis and one rotary axis

The function can also be used for a machine configuration in which the rotary axis for controlling the tool does not intersect the rotary axis for controlling the table.

NOTE
This function is an optional function.

Format

- Tilted working plane indexing command (Eulerian angle)

G68.2 $X_{\underline{x0}}$ $Y_{\underline{y0}}$ $Z_{\underline{z0}}$ $I\underline{\alpha}$ $J\underline{\beta}$ $K\underline{\gamma}$; Tilted working plane indexing command

G69 ; Cancels the tilted working plane indexing command.

X, Y, Z : Feature coordinate system origin
I, J, K : Euler's angle for determining the orientation of the feature coordinate system

- Tool axis direction control

G53.1 ; Controls the tool axis direction.

The G68.2 command causes the programming coordinate system to switch to the feature coordinate system. The commands in all subsequent blocks are assumed to be specified in the feature coordinate system until G69 is commanded.

If G68.2 specifies the relationship between the feature coordinate system and the workpiece coordinate system, tool axis direction control command G53.1 automatically specifies the +Z direction of the feature coordinate system as the tool axis direction even if no angle is specified for the rotary axis.

With the tilted working plane indexing, a tilted working plane can be specified based on the following methods:

- Tilted working plane indexing based on Eulerian angle
- Tilted working plane indexing based on roll-pitch-yaw
- Tilted working plane indexing based on three points
- Tilted working plane indexing based on two vectors
- Tilted working plane indexing based on projection angles

The user can select most suitable commands for various types of machining.

5.46.2 Tilted Working Plane Indexing by Tool Axis Direction (M Series)

M

By specifying G68.3, a coordinate system (feature coordinate system) where the tool axis direction is the +Z-axis direction can be automatically specified. When a feature coordinate system is used, a program for cutting a hole or pocket in a plane tilted relative to the workpiece coordinate system can be made simpler. This function can automatically generate a feature coordinate system that is normal to the tool direction.

When G68.3 is specified in a block, the coordinate system for programming is changed to a feature coordinate system. All commands after the block are regarded as commands in the feature coordinate system until G69 is commanded.

Format

G68.3 X $\underline{x_0}$ Y $\underline{y_0}$ Z $\underline{z_0}$ Rα ; Tilted working plane indexing command

G69 ; Cancel tilted working plane indexing command (M series).

G69.1; Cancel tilted working plane indexing command (T series).

X,Y,Z : Origin of a feature coordinate system (absolute)
By default, the current position is used as the origin of the feature coordinate system.
R : Angular displacement about the Z-axis in the feature coordinate system.
The default is 0°.

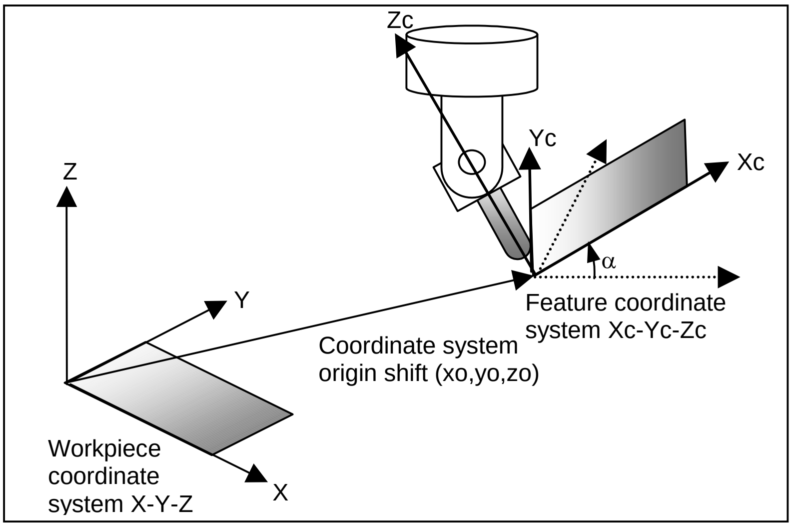


Fig. 5.46 (a)

5.47 PROGRAMMABLE MIRROR IMAGE

Mirror image can be commanded on each axis by programming. Ordinary mirror image (commanded by remote switch or setting) comes after the programmable mirror image is applied.

Format

- Setting of programmable mirror image

Following format is commanded and mirror image is commanded to each axis (as if mirror was set on the axis).

```
G51.1 IP_;
```

- Programmable mirror image cancel

Following format is commanded and the programmable mirror image is canceled.

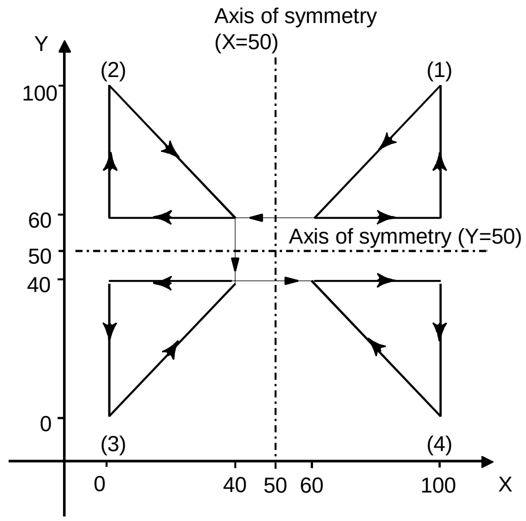
```
G50.1 IP_;
```

⚠ **CAUTION**

If mirror image is specified only for one axis on the specified plane, the operation of the commands is as follows:

- Arc command: The rotation direction is reversed.
- Cutter compensation: The offset direction is reversed.
- Coordinate system rotation: The rotation angle is reversed.

When shape of the workpiece is symmetric to an axis, a program for machining the whole part can be prepared by programming a part of the workpiece using programmable mirror image and sub program.



Main program

```
N10 G00 G90 ;
N20 M98 P9000 ;
N30 G51.1 X50.0 ;
N40 M98 P9000 ;
N50 G51.1 Y50.0 ;
N60 M98 P9000 ;
N70 G50.1 X0 ;
(Cancel only X axis)
N80 M98 P9000 ;
N90 G50.1 Y0 ;
```

Sub program

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

Fig. 5.47 (a)

5.48 SYNCHRONOUS, COMPOSITE, AND SUPERIMPOSED CONTROL BY PROGRAM COMMAND

Overview

Synchronous control, composite control, and superimposed control can be started or canceled using a program command instead of a DI signal.
Synchronous control, composite control, and superimposed control based on a DI signal is also possible.
For the basic operations of synchronous control, composite control, and superimposed control, see Sections, "SYNCHRONOUS CONTROL AND COMPOSITE CONTROL" and "SUPERIMPOSED CONTROL".

Format

G51.4 P_ Q_ (L_) ;	**Synchronous control start (L is omissible.)**
G50.4 Q_ ;	**Synchronous control cancel**

P: Synchronous master axis ID number
Q: Synchronous slave axis ID number
L: Parking command
  1: Master parking (slave parking cancel)
  2: Slave parking (master parking cancel)
  0: No parking (parking cancel)
  (When L is omitted, the specification of L0 is assumed.)

G51.5 P_ Q_ ;	**Composite control start**
G50.5 P_ Q_ ;	**Composite control cancel**

P: Composite axis 1 ID number
Q: Composite axis 2 ID number

G51.6 P_ Q_ ;	**Superimposed control start**
G50.6 Q_ ;	**Superimposed control cancel**

P: Superimposed master axis ID number
Q: Superimposed slave axis ID number

As an ID number, set a unique value for identifying each axis in parameter No. 12600 for both of P and Q.

G51.4/G50.4, G51.5/G50.5, and G51.6/G50.6 are one-shot G codes of group 00.

NOTE

This function is an optional function.
To use this function, in addition to the this option, the option for "Synchronous and Composite Control" or "Superimposed Control" or "Superimposed Control A" is required.

5.49 FIGURE COPY (M SERIES)

M

Machining can be repeated after moving or rotating the figure using a sub program.

NOTE
This function is an optional function.

Format

- Rotational copy

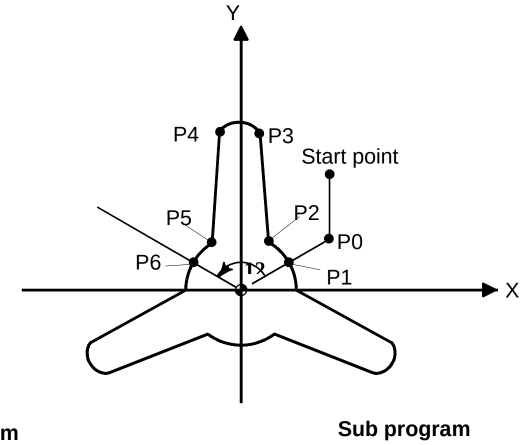
Xp-Yp plane (specified by G17) : **G72.1 P_ L_ Xp_ Yp_ R_ ;**

Zp-Xp plane (specified by G18) : **G72.1 P_ L_ Zp_ Xp_ R_ ;**

Yp-Zp plane (specified by G19) : **G72.1 P_ L_ Yp_ Zp_ R _;**

P : Sub program number
L : Number of times the operation is repeated
Xp : Center of rotation on the Xp axis
(Xp : X-axis or an axis parallel to the X-axis)
Yp : Center of rotation on the Yp axis
(Yp: Y-axis or an axis parallel to the Y-axis)
Zp : Center of rotation on the Zp axis
(Zp: Z-axis or an axis parallel to the Z-axis)
R : Angular displacement
(A positive value indicates a counterclockwise angular displacement. Specify an incremental value.)

Specify a plane selection command (G17, G18, or G19) to select the plane on which the rotational copy is made.



Main program

```
O1000 ;
N10 G92 X40.0 Y50.0 ;
N20 G00 G90 X_ Y_ ;                  (P0)
N30 G01 G17 G41 X_ Y_ D01 F10 ;      (P1)
N40 G72.1 P2000 L3 X0 Y0 R120.0 ;
N50 G40 G01 X_ Y_ I_ J_ ;
N60 G00 X40.0 Y50.0 ;
N70 M30 ;
```

Sub program

```
O2000 G03 X_ Y_ R30.0 ;    (P2)
N100 G01 X_ Y_ ;           (P3)
N200 G03 X_ Y_ R10.0 ;     (P4)
N300 G01 X_ Y_ ;           (P5)
N400 G03 X_ Y_ R30.0 ;     (P6)
N500 M99;
```

Fig. 5.49 (a) Example of rotation copy

- Linear copy

Xp-Yp plane (specified by G17) :

G72.2 P_ L_ I_ J_ ;

Zp-Xp plane (specified by G18) :

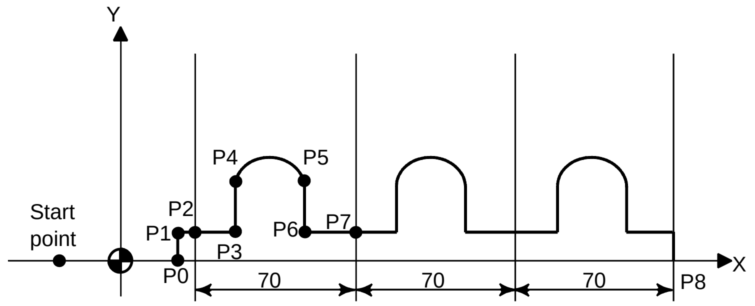
G72.2 P_ L_ K_ I_ ;

Yp-Zp plane (specified by G19) :

G72.2 P_ L_ J_ K_;

P : Sub program number
L : Number of times the operation is repeated
I : Shift along the Xp axis
J : Shift along the Yp axis
K : Shift along the Zp axis

Specify a plane selection command (G17, G18, or G19) to select the plane on which the linear copy is made.



Main program

```
O1000 ;
N10 G92 X-20.0 Y0 ;
N20 G00 G90 X0 Y0 ;
N30 G01 G17 G41 X20.0 Y0.0 D01 F10 ; (P0)
N40 Y20.0 ;                          (P1)
N50 X30.0 ;                          (P2)
N60 G72.2 P2000 L3 I70.0 J0 ;
N70 X_ Y_ ;                          (P8)
N80 X0 ;
N90 G00 G40 X-20.0 Y0 ;
N100 M30 ;
```

Sub program

```
O2000 G90 G01 X_ ;   (P3)
N100 Y_ ;            (P4)
N200 G02 X_ I_ ;     (P5)
N300 G01 Y_ ;        (P6)
N400 X_ ;            (P7)
N500 M99 ;
```

Fig. 5.49 (b) Example of linear copy

NOTE

In a sub program for rotational copy, rotational copy cannot be specified again. Similarly, in a sub program for linear copy, linear copy cannot be specified again. However, in a sub program for rotational copy, linear copy can be specified. Similarly, in a sub program for linear copy, rotational copy can be specified.

5.50 G CODE PREVENTING BUFFERING

By specifying G04.1, buffering of the following block from the block of G04.1 is prevented until finishing the block of G04.1. (In this chapter, preventing buffering by G04.1 is called non-buffering command by G code. On the other hand, preventing buffering by M codes, command of only G31, command of only G53 and etc. are called non-buffering commands of normal.)

The processing time of non-buffering command by G code reduce compared with non-buffering commands of normal. Therefore, the cycle time of automatic operation can be reduced by using non-buffering command by G code instead of non-buffering commands of normal.

5.51 FANUC Series 10/11 PROGRAM FORMAT

By setting a setting parameter, the following functions programmed in the Series 10/11 program format can be executed by memory operation:

M

- Sub program call (M98)
- Canned cycle for drilling (G73, G74, G76, G80 to G89)

T

- Sub program call (M98)
- Canned cycle (G77, G78, G79) (G90, G92, G94 with G code system A)
- Multiple repetitive canned cycle (G71 to G76)
- Canned cycle for drilling (G80 to G85)

NOTE

Address and specifiable value range
The ordinary program format restriction is imposed on the specifiable value range for basic addresses used. If a value beyond the allowable range is specified, an alarm is issued. Moreover, the usable addresses may be limited.

5.52 MACRO EXECUTOR

Some NC programs such as programs created using custom macros need not be modified once created. Others such as machining programs differ depending on the machining target. This function can convert a custom macro program created by the machine tool builder to an executable macro program, load the executable macro program (P-CODE macro) into F-ROM (Flash ROM module), and execute it.

The function which converts a custom macro program to an executable macro program is called the macro compiler. The function which reads and executes a P-CODE macro is called the macro executor.

Features

- The execution speed is high because a custom macro program is loaded after converted to an executable so that the machining time can be reduced and the machining precision can be improved.
- Any custom macro is not destroyed because it is loaded into F-ROM so that reliability is improved.
- A program converted to execution format is not displayed on the program screen, so the know-how possessed by each machine tool builder can be protected.
- An execution format macro program is registered in the F-ROM, so the program storage space can be efficiently used.
- The user can call the execution format macro program with an easy call procedure without being conscious of the registered program.
- An original screen can be created by using the graphic display or selecting screens by the soft key. The machine tool builder can extend the control function by using such functions as machining program creation and edit control, RS232C interface control, and PMC data input/output functions.

NOTE

This function is an optional function.

5.53 C LANGUAGE EXECUTOR

The C Language Executor function is used to customize screen display and implement a mechanism for user-specific operation as with the macro executor function. Instead of macro statements, application programs for display and operation can be created using the general C language.

Features

- Low-cost customization

No special additional hardware is needed to run the C Language Executor and application programs. All available display units are supported. User applications can be included in the current CNC system.

NOTE

To execute the C Language Executor and application programs, the size of the flash ROM/DRAM may need to be increased.

- Application development on a personal computer

Application programs can be developed using an ordinary personal computer. Program development, from program creation and editing to compilation/linkage, can also be performed on a personal computer. And, to a certain extent, debugging is also possible on the personal computer.

- High compatibility with C Language Executor application programs for personal computers

The function library provided by the C Language Executor has excellent compatibility with the ANSI standards and MS-C. Therefore, application programs for ordinary personal computers can be transported to the CNC, except when they are dependent on particular hardware.

- Integration of CNC software and applications

An application program created by the machine tool builder is executed as one task of the CNC software. The application program can display its own screens in place of existing CNC screens. In addition, the application program can read and write CNC system data via libraries provided by the C Language Executor. This enables operation of the application program to be integrated with CNC software.

- Using the C Language Executor with the macro executor

The C Language Executor can be used with the macro executor. The screen display portion of a macro program already created by the machine tool builder can be replaced with a program coded in C. This can prevent existing software resources from becoming useless.

NOTE

This function is an optional function.

5.54 ADDITION OF C LANGUAGE EXECUTOR SRAM

If the SRAM area required for applications created with the C Language Executor exceeds the standard size, which is 63K bytes, the SRAM area can optionally be expanded to up to 512K bytes.

NOTE

This function is an optional function.

5.55 CUSTOM SOFTWARE SIZE

The required size of custom software to be used with the macro executor and C Language Executor can be selected from the following:
512K / 2M / 4M / 6M / 8M / 12M / 16M (Unit : byte)

When a multi-path system is used, the total size required for all paths needs to be selected.
When using the C Language Executor, select 2M byte or more.

NOTE
This function is an optional function.

5.56 WORKPIECE COORDINATE SYSTEM SHIFT (T SERIES)

T

When the coordinate system actually set by the G50 command or the automatic system setting deviates from the programmed workpiece system, the set coordinate system can be shifted.
Set the desired shift amount in the workpiece coordinate system shift memory.

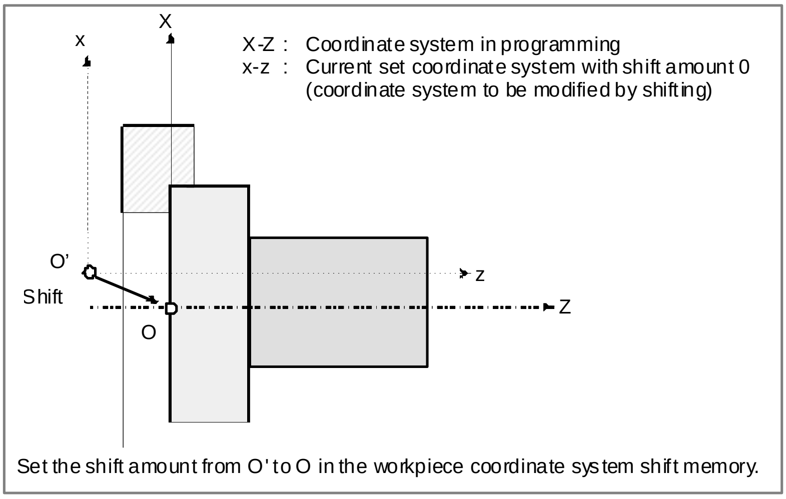


Fig. 5.56 (a)

5.57 EMBEDDED MACRO

This function protects a program created by a machine tool builder, by storing the program in an exclusive folder to embedded macros (hereinafter referred to as an MTB1 folder) and assigning an attribute to the folder.
At the same time, the following functions are available:

(1) An exclusive program memory capacity of 100K bytes (corresponding to about 260 m) is provided in addition the ordinary program memory capacity.
The number of registerable programs remains unchanged from the ordinary number of registerable programs.
(2) To the MTB1 folder, an attribute such as edit prohibition and edit/display prohibition can be assigned. Moreover, by using an exclusive password and keyword, a set value can be locked.
(3) A program stored in the MTB1 folder can be used by calling based on a code such as M/T codes, macro calling based on a G code, macro calling based on G65/G66, and subprogram calling based on an M code such as M98.
(4) Custom macro common variables (#200 to #499) are added.

NOTE

1. This function is an optional function.
2. This function differs from the embedded macro function of the FS16*i*/18*i*/21*i*-B in the following:
   - Program storage method
   - Program creation procedure
   - Program protection mechanism, etc.

5.58 SMALL-HOLE PECK DRILLING CYCLE (M SERIES)

M

An arbor with the overload torque detection function is used to retract the tool when the overload torque detection signal (skip signal) is detected during drilling. Drilling is resumed after the spindle speed and cutting feedrate are changed. These steps are repeated in this peck drilling cycle.

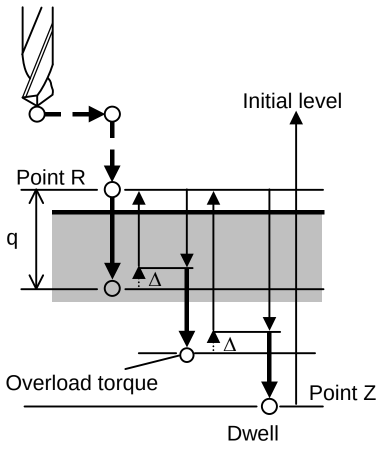
The mode for the small–hole peck drilling cycle is selected when the M code in parameter is specified. The cycle can be started by specifying G83 in this mode. This mode is canceled when G80 is specified or when a reset occurs.

Format

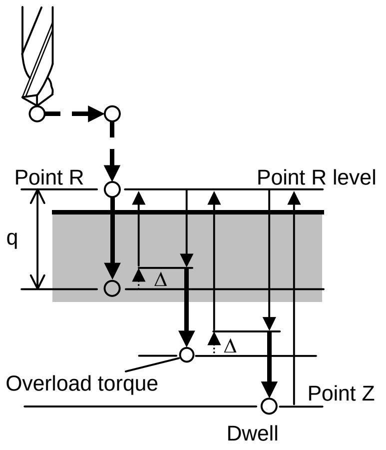
G83 X_ Y_ Z_ R_ Q_ F_I_ K_P ;

X_ Y_ : Hole position data
Z_ : Distance from point R to the bottom of the hole
R_ : Distance from the initial level to point R
Q_ : Depth of each cut
F_ : Cutting feedrate
I_ : Forward or backward traveling speed (same format as F above)
K_ : Number of times the operation is repeated (if required)
P_ : Dwell time at the bottom of the hole
(If this is omitted, P0 is assumed as the default.)

G83 (G98)



G83 (G99)



Δ: Initial clearance when the tool is retracted to point R and the clearance from the bottom of the hole in the second or subsequent drilling (parameter 5174)

q: Depth of each cut

- - → Path along which the tool travels at the rapid traverse rate

➡ Path along which the tool travels at the programmed cutting feedrate

→ (┈┈▶) Path along which the tool travels at the forward or backward rate during the cycle specified with parameters

5.59 REAL TIME CUSTOM MACRO

Used with an NC program, the real time custom macro function controls peripheral axes and signals.
If a macro statement is used together with an NC statement, a program using the conventional custom macro function executes the macro statement immediately when the macro statement is read. So, the macro statement cannot be executed independently of the NC statement.
On the other hand, the real time custom macro function enables the following control operations when a real time macro command (RTM command) is coded in an NC program.

- A real time macro command starts operation in synchronism with the NC statement and is executed independently. During NC program execution, a real time macro command can be executed at the same time.
- Some PMC interface signals can be read and written. In an NC program, a motion using a signal as a trigger can be coded.
- Variables dedicated to a real time macro command can be read and written.
- A real time macro command can exercise axis control. (If PMC axis control is enabled, this axis control can be used.)
- Multiple real time macro commands can be executed at the same time. Multiple real time custom macro statements can be coded in an NC program and can be controlled independently of each other.

NOTE
This function is an optional function.

- Real time macro command format

The RTM command is a command with two slashes (//) prefixed at the start of a block.

//n <real-time-macro-statement>
or
//n ZDO ;
<real-time-macro-statement>
:
ZEND ;

n : Modal ID (1 to 10) (Omissible)
When a proper number is coded in n, a modal real time macro command is specified.
When n is omitted, a one-shot real time macro command is specified.

- Variables Dedicated To Real Time Custom Macros

These variables are dedicated to real time custom macros. The variables are classified as system variables and RTM variables.

System variables

System variables dedicated to real time custom macros

Format

#IOp [m, n] Bit-by-bit read/write
#IOpB [m] Byte-by-byte read/write
#IOpW [m] Word-by-word read/write
#IOpD [m] Double word-by-double word read/write

p : Signal type (X, G, F, Y, D, R)
m : Signal byte address
n : (Used for bit-by-bit read/write only) Signal address bit number (0 to 7)

Real time macro variables (RTM variables)

The real time macro variables (RTM variables) are variables dedicated to real time custom macros.
The RTM variables are classified as volatile real time macro variables (volatile RTM variables) and nonvolatile real time macro variables (nonvolatile RTM variables).

The data of a nonvolatile RTM variable is preserved even when the power is turned off.
The data of a volatile RTM variable is cleared to 0 when the power is turned off.

Format

#RV [m] Volatile RTM variable
m : Volatile RTM variable number (0 to 99)
#RVS [n] Nonvolatile RTM variable
n : Nonvolatile RTM variable number (0 to 31)

5.60 PATTERN DATA INPUT

In the program of the fixed form processing with the custom macro, the operator select the processing pattern on the menu screen and specified the size, number and so on to the variable on the custom macro screen. As above mentioned, this function enables users to perform programming simply without programming using an existing NC language. With the aid of this function, a machine tool builder can prepare the program of a hole machining cycle (such as a boring cycle or tapping cycle) using the custom macro function, and can store it into the program memory. This cycle is assigned pattern names, such as BOR1, TAP3, and DRL2. An operator can select a pattern from the menu of pattern names displayed on the screen. Data (pattern data) which is to be specified by the operator should be created in advance with variables in a drilling cycle. The operator can identify these variables using names such as DEPTH, RETURN RELIEF, FEED, MATERIAL or other pattern data names. The operator assigns values (pattern data) to these names. The operator selects the pattern on the menu screen, and the selected pattern number is assigned to the system variable. The custom macro of the selected pattern can be started by starting a program then referring to the system variable in the program.

5.61 M-CODE PROTECT FUNCTION

This function permits the execution of specific M-codes (miscellaneous functions) only if they are specified from within a program called with a macro call.
This makes it possible to protect specific M-codes for use in macro programs for machine control from illegal use in user machining programs.

NOTE
This function is an optional function.

6 GUIDANCE FUNCTION

Chapter 6, "GUIDANCE FUNCTION", consists of the following sections:

6.1 MANUAL GUIDE *i*

MANUAL GUIDE *i* is an integrated operation and programming guidance function for assisting machine operators in all routine operations from machining program creation to actual machining.

> **NOTE**
> This function is an optional function.

6.1.1 Basic Function

MANUAL GUIDE *i* features a single screen, called an integrated operation screen, on which all operations, from machining program creation to actual machining, are collected. On this screen, detail data can be displayed by opening windows as required.
The integrated operation screen has advanced ISO code program editing functions, such as copy & paste, word search, redo/undo, fixed-phrase insertion, M code listing, and guidance message display.

6.1.2 Milling Cycle

Advanced milling cycle is enabled.
Hole machining, facing, contouring, pocketing, grooving, and emboss machining can be performed.
(The above machining operations except facing and emboss machining support polar coordinates and cylindrical coordinates.)
With this function, complicated machining can be performed by creating a simple program.

6.1.3 Turning Cycle

Advanced turning cycle is enabled.
Hole machining, turning, residual machining by turning, threading, and grooving can be performed.
With this function, complicated machining can be performed by creating a simple program.

6.1.4 Animation

Background machining simulation (a machine can be used to check machining programs while it is running on another machining program) is available.
This function makes it easy to check the milling, turning, or tilted-surface machining is being carried out, using tool path drawing or animation.
Foreground tool path drawing (the path of a tool can be drawn while the tool is being used in machining) is also available.

6.1.5 Set-up Guidance Function

This function supports set-up operations before machining and measurements after machining.
Tool offset measurement, workpiece centering, and measurement of machined workpieces inside the machine can be performed.
In addition, manual measurement and automatic measurement are enabled.
By using this function, set-up operations required for machining ranging from tool offset measurement to measurement of machined workpieces within the machine, and precision checking after machining can be performed easily.

6.2 MANUAL GUIDE *i* MULTI-PATH LATHE FUNCTION

T

This function assists the user in programming operations and actual machining on multi-path laths.
It is possible to use process editing based on a process directory, which enables the user to recognize at a glance which path (tool post) is to be used for a specific spindle by dividing an ISO code program into units of processes.
Using animation to check machining programs makes it possible to simulate simultaneous machining of a single work on multiple paths (tool posts).

NOTE
This function is an optional function.
To use this function, the both options for "MANUAL GUIDE *i* " and this function are required.

6.3 MANUAL GUIDE 0*i*

MANUAL GUIDE 0*i* is an easy-to-operate programming guidance function tailored only to programming.

NOTE
This function is an optional function.

6.3.1 Basic Functions

MANUAL GUIDE 0*i* provides the menu screens for operations required to create a machining program. On these menu screens, it is possible to select the operation screen for specifying a tool or spindle, guiding G codes or M codes, creating a machining cycle, or contour programming.

6.3.2 Milling Cycle (M series)

M

The fixed form machining cycles (drilling, facing, pocketing, and grooving) frequently used by a machining center or milling machine are provided.
This function can be used to create a machining cycle program with a simple operation.

6.3.3 Turning Cycle (T Series)

T

Machining cycles required for lathes are provided.
Turning drilling, turning, threading, and grooving can be performed.
This function can be used to create a machining cycle program with a simple operation.

6.3.4 Contour Programming Function

MANUAL GUIDE 0*i* can easily input a contour profile consisting of straight lines and arcs and convert it into NC command blocks (G01/G02/G03). In addition, advanced profile calculation functions including seven auxiliary calculation functions can be used.

6.4 TURN MATE *i* (T SERIES)

T

TURN MATE *i* is an integrated operation guidance function that achieves machining by general lathes without using an NC machining program.
If data is input according to guidance displayed on the screen, turning can be performed easily without using an NC program.

NOTE
This function is an optional function.

6.4.1 Basic Functions

TURN MATE *i* has all information required for operation on one operation screen. Only if information on the touch panel of TURN MATE *i* is pressed, the corresponding operation screen can be displayed.
On the operation screens, the workpiece coordinate system, spindle speed, feedrate, and other items can be set.

6.4.2 Turning Cycle

All cycle machining required for general lathes is covered. The machining profiles frequently used for a drawing are collected as fixed format patterns to save the effort of inputting profiles.
Turning, grooving, threading, thread repair, and turning drilling can be performed.
This function can be used to perform complicated machining with a simple input operation.
In addition, machining cycles can be converted into NC statements. Up to 20 machining cycles can be executed continuously.

7 AUXILIARY FUNCTION / SPINDLE SPEED FUNCTION

Chapter 7, "AUXILIARY FUNCTION / SPINDLE SPEED FUNCTION", consists of the following sections:

7.1 AUXILIARY FUNCTION

When a 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. However, up to 5 M codes can be specified.
The maximum number of digits can be specified by parameter setting.

The following M codes have special meanings.

- M02, M30 : End of program
- M00 : Program stop

- M01 : Optional stop

M98, M99, M198, the M code for calling a sub program (set by a parameter), the M code for calling a custom macro (set by a parameter), and the M codes for turning an interrupt macro on and off (M96 and M97, or parameter-set M codes) are processed internally by the CNC, so no signal is output for these M codes.

7.2 SECOND AUXILIARY FUNCTION

If a value with a maximum of eight digits is specified after address B, the code signal and strobe signal are transferred for calculation of the rotation axis. The code signal is retained until the next B code is specified.

Only one B code can be specified for each block. The maximum number of digits and decimal point programming can be specified by parameter.

In addition, the address used for specifying the second auxiliary function can be changed to an address other than address B (address A, C, U, V, or W) by setting parameter.

NOTE

In lathe systems, addresses U, V, and W can be used only with G code system B or C.

7.3 AUXILIARY FUNCTION LOCK

The auxiliary function lock signal disables execution of a specified M, S, T, or B function. This means that the code signal and strobe signal are not output.

This function is used together with the machine lock function at the time of program checking.

Even in the auxiliary function lock state, M00, M01, M02, M30, M98, M99, and M198 (sub program call function) are executed. The M code for calling a sub program (set by a parameter) and the M code for calling a custom macro (set by a parameter) are also executed.

7.4 HIGH-SPEED M/S/T/B INTERFACE

The communication of execution command signal (strobe signal) and completion signal in the M/S/T/B function were simplified to realize a high-speed execution of M/S/T/B function.

The time required for cutting can be minimized by speeding up the execution time of M/S/T/B function.

The description below uses the auxiliary functions (M code commands) as an example. The same description applies to the spindle speed function (S code), tool function (T code) and 2nd auxiliary function (B code).

(1) Assume that the following program is given:
  Mxx;
  Myy;

(2) In response to an M command, the CNC system sends out the code signals.
  The CNC system inverts the logical level of the strobe signal, that is, from "0"to "1", or from "1" to "0".

(3) The CNC system inverts the strobe signal, then when the logical level of the auxiliary function completion signal becomes the same as the strobe signal, the CNC assumes the completion of PMC sequence.
  With the usual method, the operation is assumed to be completed when a falling edge ("1" to "0") of the M/S/T/B completion signal is received after a rising edge ("0" to "1") of the signal is detected. This new system, on the other hand, assumes the operation has been completed upon detection of only one transition of the completion signal.

In addition , the conventional system uses only one completion signal common to the M/S/T/B functions. This new system uses a different completion signal for each of the M, S, T, and B functions.
The Fig. 7.4 (a) below shows the timing chart of these signals with the new system. For comparison, Fig. 7.4 (b) shows the timing chart of the conventional system.

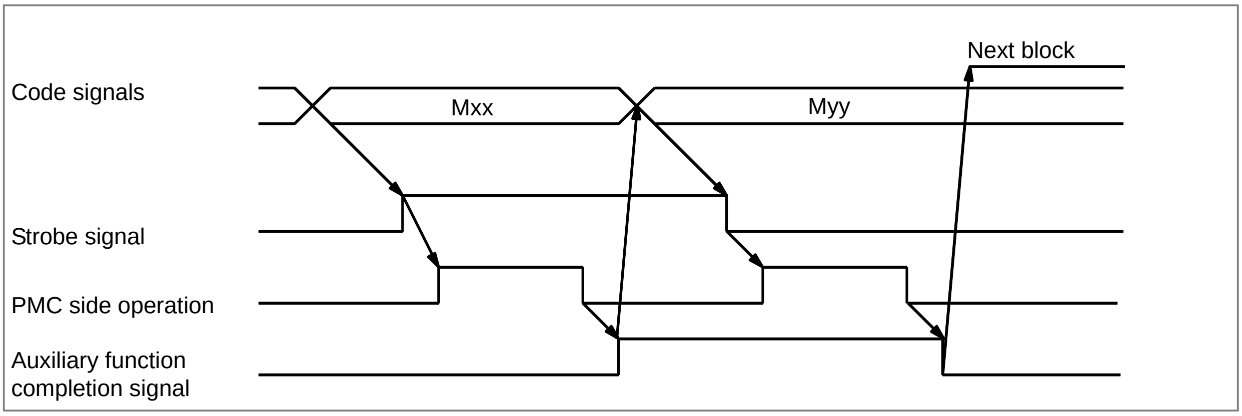


Fig. 7.4 (a) Timing chart of the high-speed system

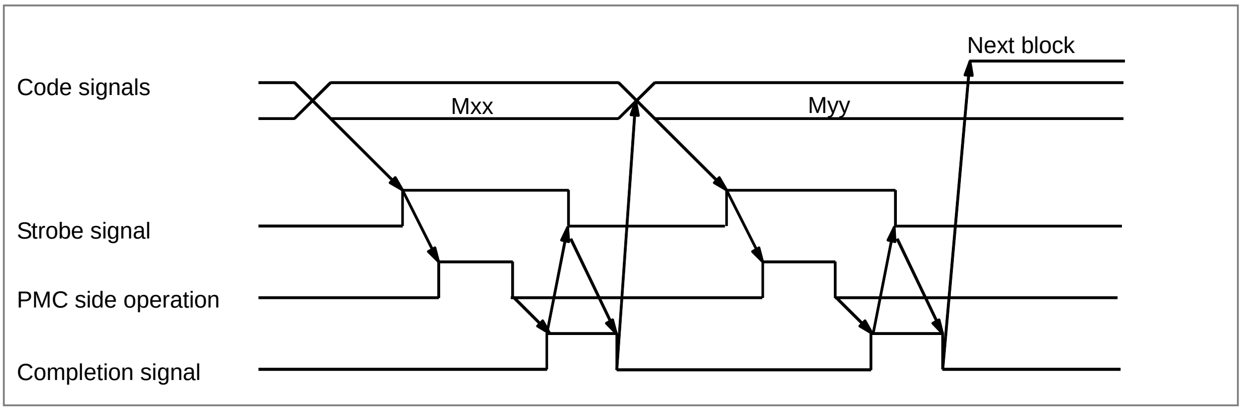


Fig. 7.4 (b) Timing chart of the conventional system

NOTE

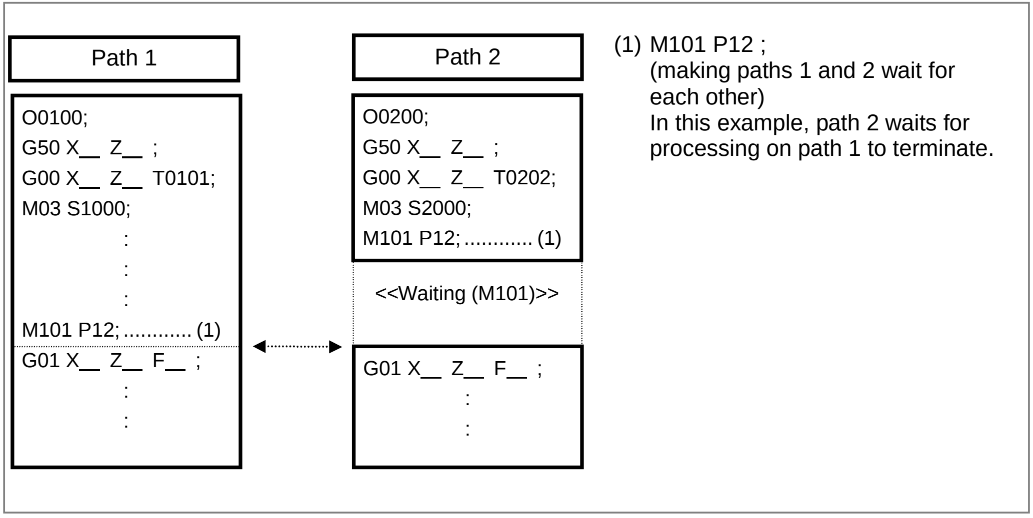
1 Which system, the conventional system or high-speed system, is to be used for transferring the strobe signal and completion signal can be specified by parameter setting.
2 The conventional system used just one completion signal, which is common to the M, S, T, and B functions. In the high-speed system, a separate completion signal is provided for each of the M, S, T, and B functions.

7.5 WAITING FUNCTION

Control based on M codes is used to cause one path to wait for the other during machining. When an M code for waiting is specified in a block for one path during automatic operation, the other path waits for the same M code to be specified before staring the execution of the next block.
A range of M codes used as M codes for waiting is to be set in the parameters beforehand. Waiting can be ignored using a signal.

Example)

When M codes for waiting are set to M101.



Path 1:

```
O0100;
G50 X__ Z__ ;
G00 X__ Z__ T0101;
M03 S1000;
        :
        :
        :
M101 P12; ............ (1)
G01 X__ Z__ F__ ;
        :
        :
```

Path 2:

```
O0200;
G50 X__ Z__ ;
G00 X__ Z__ T0202;
M03 S2000;
M101 P12; ............ (1)
G01 X__ Z__ F__ ;
        :
        :
```

(1) M101 P12 ;
(making paths 1 and 2 wait for each other)
In this example, path 2 waits for processing on path 1 to terminate.

⚠ **CAUTION**

1 Be sure to specify a waiting M code in an independent block.
2 Unlike other M codes, waiting M codes are not output to PMCs.
3 If you want to independently operate one path only, you need not delete that waiting M code. Using the waiting invalidation signal, you can invalidate the waiting M code specified in the machining program. For details, refer to the manual issued by the machine tool builder.

7.6 WAITING M CODES OF HIGH-SPEED TYPE

Waiting M codes of high-speed type are useful function to reduce the cycle time of a program operation. The usage of waiting M codes of high-speed type is similar to normal waiting M codes. For instance, the ways to set a path by address P and to use a no-wait signal or so on are the same.

7.7 MULTIPLE COMMAND OF AUXILIARY FUNCTION

Usually, one block has been able to contain only one M code. Parameter setting makes it possible to specify up to 5 M codes in one block at the same time.

Up to 5 M codes specified in a block are simultaneously output to the machine. This means that compared with the conventional method of a single M command in a single block, a shorter cycle time can be realized in machining.

(Example)

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

7.8 AUXILIARY FUNCTION OUTPUT IN MOVING AXIS

By specifying absolute coordinate values and auxiliary functions (M, B) at G50.9 block, the auxiliary functions are output to PMC when absolute coordinate enters the specified area in the next movement block. G50.9 can be specified up to 2 blocks consecutively. In other words, auxiliary function output points in a movement block can be specified up to two.

Code signals and strobe signals are output to the same signal address as a usual auxiliary function.

NOTE

This function is an optional function.

7.9 WAITING FUNCTION BY SPECIFYING START POINT

Control based on M codes is used to make one path to wait for the other during machining. When an M code for waiting is specified in a block of a path during automatic operation, the other path waits for the same M code to be specified before staring the execution of the next block.

In this function, by specifying a start point with a waiting M code, absolute coordinate value of that path or the other path can be used as the condition for waiting.

NOTE

This function is an optional function.

7.10 SPINDLE SPEED FUNCTION (S CODE OUTPUT)

A numeric value following address S specifies the spindle speed.

The specified value is output to the PMC in the form of 32-bit binary code. The code is maintained until another S is specified. In addition, the maximum number of input digits can be specified with a parameter.

7.11 SPINDLE SERIAL OUTPUT

The speed of the serial interface spindle is controlled.

The spindle speed is specified by up to five digits numeric value following address S. A speed command is output to the spindle motor according to the specified spindle speed. During constant surface speed control, a speed command is output according to the spindle speed reached after constant surface speed control.

7.12 SPINDLE ANALOG OUTPUT

An analog spindle is subjected to spindle rotation control.

A 5-digit (maximum) numeric value following address S specifies the rotation speed of the spindle. A speed command voltage is output to the spindle motor according to the specified spindle speed. During constant surface speed control, a speed command is output according to the spindle speed that will be reached after constant surface speed control.

7.13 CONSTANT SURFACE SPEED CONTROL

When a surface speed (in m/min or feet/min) is specified with an S code (a numeric value following S), the spindle speed is controlled so that the surface speed can be kept constant with respect to the change in tool position.

With a machine having (or using) no position coder, feed per revolution is usually impossible. However, parameter setting makes it possible to specify feed per revolution by using spindle commands including a constant surface speed control command. (Without position coder constant surface speed control)

Format

G96 S_ : Constant surface speed control mode

S_ : Surface speed (m/min or feet/min)

G97 S_ : Constant surface speed control cancel mode

S_ : Spindle speed (min^{-1})

Constant surface speed control command G96 is a modal G code.
Once G96 is specified, the constant surface speed control mode (G96 mode is active) and assumes the specified S value is assumed to be the surface speed.

The axis on which the calculation for constant surface speed control is based can be set with a parameter or by using the following command:

G96 P_ ;

P0 : Axis set in the parameter
P1 : 1st axis to Pn : n-th axis (n is the axis number in the path.)

The maximum spindle speed can also be set by using the following command:

G92 S_ ;

S_ : Maximum spindle speed (min^{-1})

7.14 SPINDLE OVERRIDE

To the spindle speed specified by S code, an override from 0 to 254% can be applied (in steps of 1%).

7.15 ACTUAL SPINDLE SPEED OUTPUT (T SERIES)

T

Actual spindle speed calculated by the return signal of the position coder on the spindle is output in 16-bit binary code.

7.16 SPINDLE ORIENTATION

You can perform spindle orientation simply by mounting a position coder on the spindle. Stoppers or pins for physically stopping the spindle at a specified position are not necessary. A spindle can be instantly oriented, even when rotating at high-speed, thereby greatly reducing the orientation time.

When a spindle other than the first spindle in each path is oriented, spindle orientation expansion is used. Spindle orientation can be performed for a spindle with a spindle number up to the maximum number of controlled spindles.

7.17 SPINDLE OUTPUT SWITCHING FUNCTION

Spindle output switching switches between the two windings, one for low speed and the other for high speed, incorporated into the special spindle motors. This ensures that the spindle motor demonstrates stable output characteristics over a wide range.

When spindle output switching is performed for a spindle other than the first spindle in each path, spindle output switching expansion is used.
Spindle output switching can be performed for a spindle with a spindle number up to the maximum number of controlled spindles.

7.18 SPINDLE SYNCHRONOUS CONTROL

In a machine having two or more spindles (such as a multi-path lathe), this function controls more than one spindle synchronously.
When a workpiece is switched between two spindles during spindle rotation, or when the spindle speeds of two spindles are accelerated or decelerated while the spindles are holding a workpiece, the spindles can be rotated at the same speed. (Spindle speed synchronization)
When a non-standard workpiece is switched, the rotations of the two spindles can be made in phase with each other (matching of angular displacement). (Phase synchronization)

7.19 SIMPLE SPINDLE SYNCHRONOUS CONTROL

Spindles can be synchronized with each other for control.
Two spindles can be combined, one as a master spindle and the other as a slave spindle, so that Cs contouring control, rigid tapping, spindle positioning, and normal spindle rotation control can also be used with the slave spindle under control by the master spindle.
A combination of master and slave spindles subjected to simple spindle synchronous control can be selected freely from spindles on the same path.
Multiple combinations of a master spindle and slave spindle can be placed under simple spindle synchronous control.
This function can be used with serial spindles.

Unlike spindle synchronous control, simple spindle synchronous control does not guarantee spindle operation synchronization between a master spindle and slave spindle. Moreover, applicable spindle functions differ.

The major differences from spindle synchronous control are indicated below (Table 7.19 (a)).

Table 7.19 (a)

Function		Simple spindle synchronous control	Spindle synchronous control
Combination with other spindle functions	Spindle rotation control	∆(Same speed command is used.)	O(Synchronization / phase matching is possible.)
	Cs contour control	✓	
	Rigid tapping	✓	
	Spindle positioning	✓	
Synchronization of multiple slave spindles		✓	✓
Parking function		✓	

NOTE
This function is an optional function.

7.20 MULTI SPINDLE CONTROL

In addition to the first spindle, the second to fourth spindles can be controlled using an S command from the CNC.
Spindle commands are specified using a single S command as conventionally done. A spindle is selected depending on the signal from the PMC or the address P command.
Gear change between two stages can be made for the second, third, and fourth spindles in the same manner as for the first spindle.
Parameter setting makes it possible to set a maximum spindle speed for each spindle and clamp the speed of each spindle at the corresponding maximum spindle speed.
The position coder interfaces for the second to fourth spindles can be selected and added.
The first to fourth position coders are selected by signals from the PMC or the address P command.

NOTE

The specification of this function depends on the model.
- M series : optional function
- T series : basic function

There are three types of multi spindle control methods: a method that can use the SIND (spindle motor speed control by the PMC) function only for the first spindle (TYPE-A), a method that can use the SIND function for the four spindles independently (TYPE-B), and a method that is equivalent to TYPE-B and can use address P to select the spindle to be controlled by the S command.
These methods can be selected by parameter setting.

An S command is sent as a speed command to each spindle selected, using a spindle selection signals. Each spindle rotates at the specified speed. If a spindle is not sent a spindle selection signal, it continues to rotate at its previous speed. This allows the spindles to rotate at different speeds at the same time.
Signals are provided to stop the corresponding spindles, so unnecessary spindles can be kept stopped.

- When multi spindle control is disabled

Only the first spindle can be controlled using the S command even if there are more than one spindle.

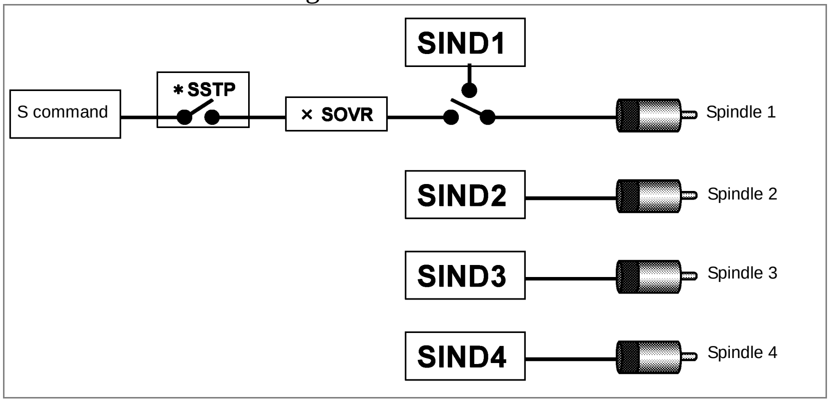


Fig. 7.20 (a) When multi spindle control is disabled

- Multi-spindle control (TYPE-A)

When the first spindle is selected with the SWS1 signal, the SIND signal is used to determine whether the spindle analog voltage is controlled by the PMC or CNC; then signals R01I to R12I are used to set that spindle’s analog voltage. These signals do not affect the second to fourth spindles.
The PMC-based polarity (rotation direction) control signals will function for any spindle selected by SWS1 to SWS4.

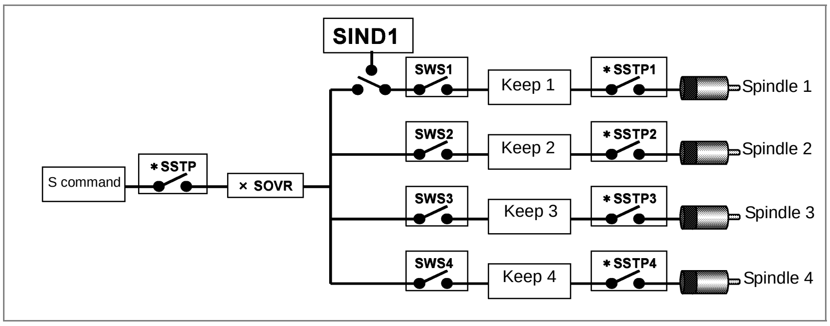


Fig. 7.20 (b) Multi-spindle control (TYPE-A)

- Multi-spindle control (TYPE-B)

Each spindle has its own SIND, SSIN and SGN signals.

Each of these signals functions regardless of selection state of the spindle selection signals.

When either the spindle selection signals or the SIND signal for the first, second, third, or fourth spindle is set to "1", the polarity control signals will function.

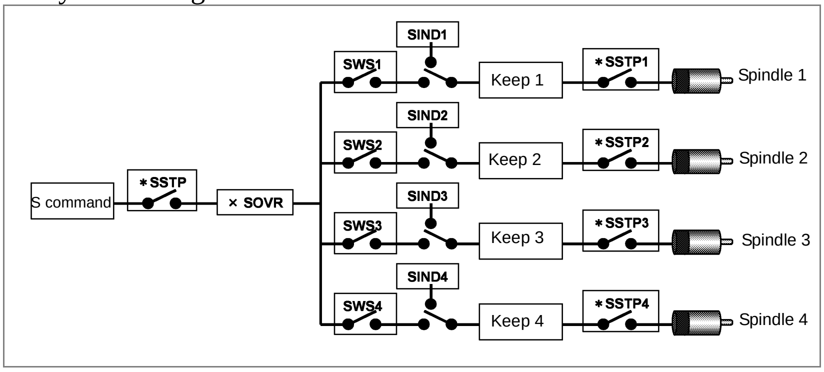


Fig. 7.20 (c) Multi-spindle control (TYPE-B)

- Multi spindle control: when a spindle is selected by address P

This control method is basically the same as TYPE-B. The first, second, third, and fourth spindles each have their own SIND, SSIN, and SGN signals. A spindle is selected by the P command instead of the spindle selection signals. The relationship between the P code and the selected spindle is set in parameter.

Polarity (rotation direction) control signals for each spindle are valid only for the spindle selected by the P command or for the spindle of which SIND signal is "1".

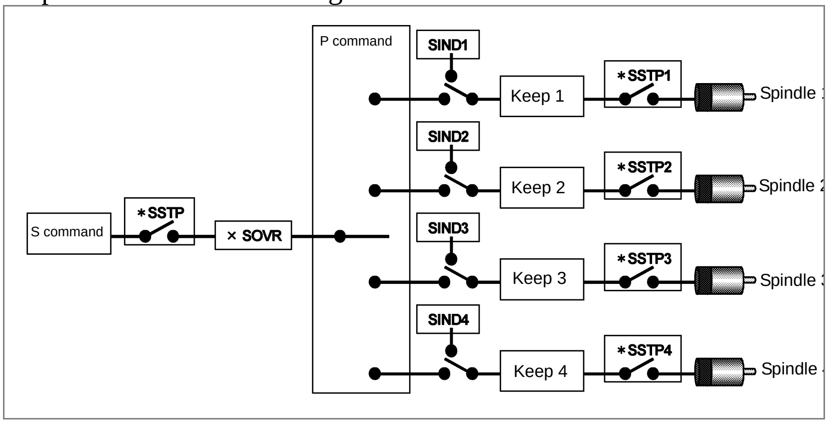


Fig. 7.20 (d) When a spindle is selected by address P

In case of multi-path control, the P code becomes valid in all paths. For example, suppose that the P code to select the first spindle of path 2 is set to 21. When the following is specified in path 1:

S1000 P21;

Then, 1000 is specified for the first spindle in path 2. Therefore, the same P code cannot be used even in different paths.

When the spindle is selected by address P, position coder feedback is automatically selected by address P for feed per revolution, thread cutting, and so on.

- Extended spindle name

Usually, S is used as commands for spindles. If the parameter settings and the conditions listed below are all satisfied, extended spindle names can be used. Extended spindle names can consist of up to three characters beginning with S as the first spindle name character, and use of an extended spindle name allows a command to be executed for a specific spindle.

- The serial spindle function is enabled.
- The multi spindle function is enabled.

7.21 SPINDLE POSITIONING (T SERIES)

T

The workpiece mounted on the spindle can be positioned at a certain angle by moving the spindle attached to the spindle motor by the certain angle. This function is called the spindle positioning function. During turning, use of this function allows machining such as drilling at any position on the circumference of the workpiece.
The spindle position is detected by the position coder attached to the spindle.
Whether to use the spindle for spindle positioning (spindle positioning mode) or to use the spindle for spindle rotation (spindle rotation mode) is command by special M code (set by parameters).

- Movement command

G00 C_ ;(C_ is an arbitrary axis address.)
The spindle is positioned to the commanded position by rapid traverse. Absolute (G90) and incremental (G91) commands are available. Decimal point input is also available.

- Increment system

Least input increment : 0.001deg

Detection unit : $\frac{360 \times N}{4096}$ deg

N represents the gear ratio of the spindle to position coder. (1, 2, 4, 8, . . .)

7.22 RIGID TAPPING

In tapping, the feed amount of drilling axis for one rotation of spindle should be equal to the pitch of screw of tapper. Namely, the following conditions must be satisfied in the best tapping:

P=F/S
where P : Pitch of screw of tapper (mm, inch)
F : Feed rate of drilling axis (mm/min, inch/min)
S : Spindle speed (min^{-1})

The rotation of spindle and feed of Z axis are independently controlled in the tapping cycle G74/G84 (machining center system), G84/G88 (lathe system). Therefore, the above conditions may not always be satisfied. Especially at the hole bottom, both the rotation of spindle and feed of drilling axis reduce the speed and stop. After that, they move in the inverse direction while increasing the speed. However, the above conditions may not be satisfied in general since each acceleration/deceleration is performed independently. Therefore, in general, the feed is compensated by mounting a spring to the inside of holder of tapper to improve the accuracy of tap cutting.
The rotation of spindle and feed of drilling axis are controlled so that they are always synchronous each other in the rigid tapping cycle. Namely, in other than rigid tapping, control for speed only is performed.

In the rigid tapping however, position control is also performed during the rotation of spindle, that is, the rotation of spindle and feed of drilling axis are controlled as linear interpolation of two axes.
This allows the following condition to be satisfied also during acceleration/deceleration at the hole bottom and a tapping of improved accuracy to be made.
P = F/S

Rigid tapping can be performed by executing any of the following commands:

- M29 Sxxxxx before tapping command G74/G84 or G84/G88
- M29 Sxxxxx in the same block as tapping command G74/G84 or G84/G88
- G74/G84 or G84/G88 as rigid tapping G code (Whether G74/G84 is used as rigid tapping G code or ordinary tapping G code can be selected with a parameter.)

7.23 RIGID TAPPING BY MANUAL HANDLE

After a programmed command for rigid tapping is executed in MDI mode to form the rigid tapping mode, then switching to the handle mode is made, rigid tapping can be performed by moving the tapping axis with the manual handle.

In the programmed command to form the rigid tapping mode, the tapping axis must be specified. In this case, a value for preventing the tapping axis from operating must be specified.
Example)

```
M29 S100 ;
G91 G84 Z0 F1000 ;
```

The rotation direction of the spindle is determined by the specified tapping cycle G code and parameter.

NOTE
This function is an optional function.

7.24 ARBITRARY POSITION REFERENCE SETTING FOR Cs AXIS FUNCTION

When a reference position return command (G28 or a manual reference position return) is executed for the first time since a serial spindle is placed in the Cs contour control mode, an arbitrary position can be set as the reference position by parameter setting.
In this case, the system performs reference position return, assuming that the current position is the reference position, so the system does not move the spindle to position it to the reference position.

Because the reference position return operation does not involve positioning to the spindle position specific to the spindle, this function can reduce the cycle time.

NOTE
This function is an optional function.

7.25 M CODE GROUP CHECK FUNCTION

When multiple M commands in a single block (enabled when the parameter is set) are used, you can check the following items. You can also select whether to check the items by the parameter.

(1) M code to be specified in a single block containing no other M codes
If an M code which must be specified in a single block containing no other M codes is specified together with another M code, an alarm is issued.

(2) M codes in the same group
If multiple M codes in the same group are specified together, an alarm is issued.

NOTE
This function is an optional function.

7.26 SPINDLE SPEED FLUCTUATION DETECTION (T SERIES)

T

With this function, an overheat alarm is raised and the spindle speed fluctuation detection alarm signal SPAL is issued when the spindle speed deviates from the specified speed due to machine conditions.
This function is useful, for example, for preventing the seizure of the guide bushing.

7.27 Cs CONTOUR CONTROL AXIS COORDINATE ESTABLISHMENT

Shifting a serial spindle from spindle rotation control to Cs contour control results in its current position being lost.
This function is intended to establish the current position without making a reference position return. This is done by setting the Cs axis coordinate establishment request signal to "1".

NOTE
Once the power has been turned on, this function remains enabled until the power is turned off after the return of the Cs contour axis to the reference position.

7.28 SPINDLE CONTROL WITH SERVO MOTOR

Servo motors can be controlled by spindle functions like spindle rotation commands or rigid tapping.

(1) Spindle control with servo motor
Rotation commands (S command) can be used to control the speed of spindles by regarding servo motors as spindles (live tools). No reference position return is necessary to switch between rotation and positioning commands.

(2) Spindle indexing
There are two types of spindle indexing. The first type allows the next-block command to be executed before spindle indexing is finished. The second type allows the next block to be executed only after spindle indexing is completed.
With the first type, it is possible to issue commands to axes other than the axis for which a spindle indexing command has been issued, before the next command is issued to the axis. The next command can be issued only after it is confirmed that spindle indexing is completed. Using this function can reduce the wait time. In addition, an axis can be stopped at a specified point by issuing a spindle indexing command to the axis when the spindle is rotating.

(3) Rigid tapping with servo motor
Rigid tapping can be carried out by regarding a servo motor as a rotation axis.

(4) Threading, feed per revolution, and constant surface speed control
Threading, feed per revolution, and constant surface speed control can be carried out by regarding a servo motor as a spindle.

(5) Spindle output control with PMC
Spindle output control with PMC can be carried out by regarding a servo motor as a spindle controlled axis.

NOTE
This function is an optional function.

Notes

(1) Using this function requires the multi-spindle control option (M series).
If this function is used under the conditions below, the multi-spindle control option is not required (M series).
  (a) Two or more spindles are not used on any path.
  (b) The spindle gear type is T.
  (c) No G code (G96.1, G96.2, G96.3, or G96.4) commands are used.
  (d) Spindle indexing is not used.

(2) This function handles a servo motor used as a spindle controlled axis as one of controlled spindles.

Table 7.28 (a) Maximum number of controllable spindles and spindle controlled axes per system

Maximum number of spindles in an entire system	Spindles + spindle controlled axes with servo motor	Spindle only
1-path system / M series : 2 axes	2 axes	2 axes
1-path system / T series : 3 axes	3 axes	3 axes
2-path system / M series : 4 axes 2-path system / T series : 4 axes	4 axes	4 axes

Spindle motors and supported functions

Table 7.28 (b)

Spindle Function	Conventional spindle control	Spindle control with servo motor
Threading/feed per revolution	✓	✓
Polygon machining	✓	(*1)
Spindle speed fluctuation detection	✓	
Spindle synchronous control	✓	
Simple spindle synchronous control	✓	
Polygon machining with two spindles	✓	
Spindle orientation Multi-point orientation Spindle output switching	✓	
Inter-path spindle control	✓	✓
Constant surface speed control	✓	✓
Multi-spindle control	✓	✓
Rigid tapping	✓	✓
Spindle output control with PMC	✓	✓
Actual spindle speed output	✓	✓
Spindle indexing		✓

NOTE
Servo motor can be used as tool rotation axis. (*1)

7.29 SPINDLE REVOLUTION NUMBER HISTORY FUNCTION

This function counts the spindle revolution number and displays the total revolution number as diagnosis data.
Total revolution number data as diagnosis data can be read with the PMC window function, and can be used for spindle unit life management on the PMC ladder.

7.30 SERVO/SPINDLE SYNCHRONOUS CONTROL

This function provides the following functions to use a servo motor as a spindle:

(1) Servo motor spindle control
Can rotate the servo motor at the rotation speed specified with an input signal.

(2) Servo motor spindle synchronization
Can rotate the servo motor in synchronization with the feedback pulses from the position coder of the spindle.

(3) Differential speed synchronization
  (a) Can superimpose a command from the CNC on the servo motor in servo motor spindle synchronization.
  (b) Can superimpose the rotation speed specified with an input signal on the servo motor in servo motor spindle synchronization.

NOTE
This function is an optional function. The servo/spindle synchronous control option is required.

7.31 HIGH-PRECISION SPINDLE SPEED CONTROL

Overview

A large-scale machine cuts a large-diameter workpiece by turning it.
Conventional spindle commands (S code commands) cannot specify a surface speed (cutting speed) for large-diameter workpieces in detail because they specify the spindle speed using an integer value.
This function uses numbers having decimal points, thus enabling the cutting speed to be specified in detail.

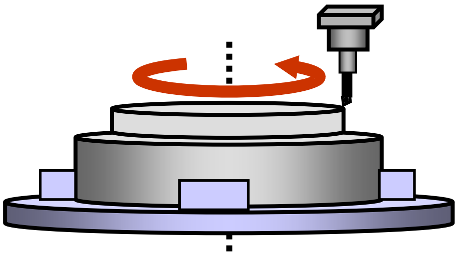


Fig. 7.31 (a) Cutting a large-diameter workpiece by turning it

7.32 SIMPLE SPINDLE ELECTRONIC GEAR BOX

This function executes spindle synchronous control between two serial spindles based on input signals so that one spindle (slave spindle) follows the other spindle (master spindle). It uses a method of referencing directly feedback pulses to enable the slave spindle to follow fluctuations in the master spindle speed with a small error, thereby achieving high-precision spindle synchronous control.
An example of using the function might be rotary guide bush control where two spindles are used.
An electronic gear box (hereafter called EGB) for the spindle is used to make the two spindles synchronize with each other.
Using the function requires the spindle serial output and the Cs contour control.

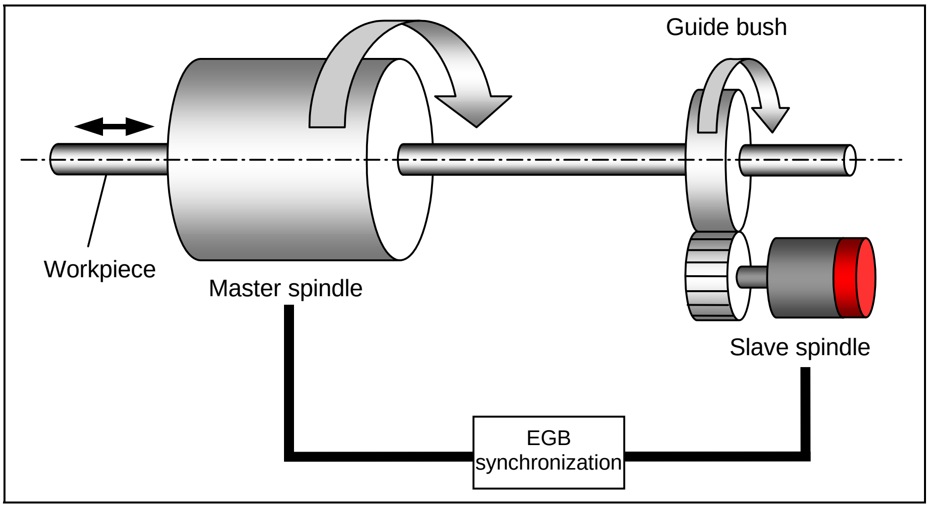


Fig. 7.32 (a)

7.33 SPINDLE SPEED COMMAND CLAMP

By using this function, spindle speed can be clamped to the maximum speed by setting on internal relay (R signal) data. Spindle speed is clamped to maximum speed immediately if the spindle speed exceeds the maximum speed. For example, this function can be used to change maximum speed depending on selected tool.

7.34 ARBITRARY SPEED THREADING

7.34.1 Arbitrary Speed Threading

Overview

In conventional threading, spindle speed could not be changed because tool movement could not synchronize with spindle rotation. This function makes it possible to change spindle speed during threading. This function can be used with constant lead threading, threading cycle (T series), and multiple threading cycle (T series). Changing spindle speed can prevent the vibration that occurs during threading at the specific spindle speed for a large size machine. Besides, for repetitive machining, the same thread shape can be machined even if spindle speed is changed between rough machining and finishing machining. Cs contour control is required for this function. For M series, thread cutting, synchronous cutting option is required.

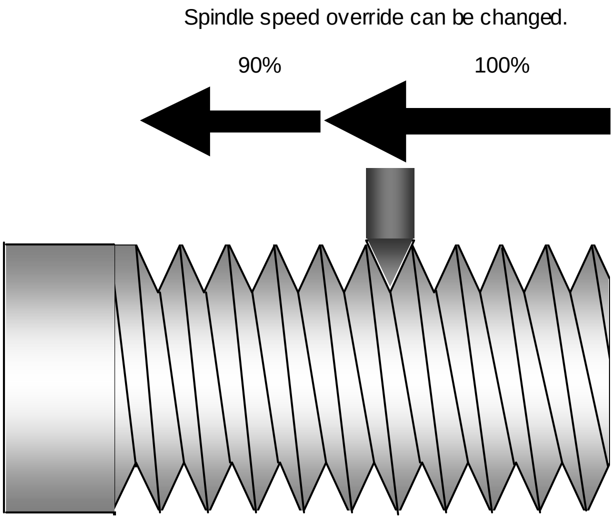


Fig.7.34.1 (a) Arbitrary speed threading

Furthermore, even if the workpiece which is not finished the threading is removed from the chuck, re-machining of this workpiece can be achieved. Refer to the section “Re-machining thread” for details.

NOTE
This function is an optional function.

Format

Mxx ; **Start of arbitrary speed threading**
G32IP_F_; **Threading command**
Myy ; **Cancel of arbitrary speed threading**
xx : M code set in parameter
yy : M code set in parameter

7.34.2 Re-machining Thread

Overview

This function makes it possible to re-machine the same thread even if the workpiece which is not finished threading is removed from a chuck once. In order to re-machine the thread, groove of thread is measured after mounting the workpiece on a chuck. Then, the thread is re-machined by the same machining program.

NOTE
This function is included in the option "Arbitrary Speed Threading".

8 TOOL FUNCTION / TOOL COMPENSATION FUNCTION

Chapter 8, "TOOL FUNCTION / TOOL COMPENSATION FUNCTION", consists of the following sections:

8.1 TOOL FUNCTION

A tool can be selected by specifying a tool number of up to eight digits numeric value following address T.
When a T code is specified, the code signal corresponding to the tool number and the strobe signal are sent to the machine to select a tool. This code signal is maintained until another T code is specified.
Just one T code can be specified per block. The maximum number of digits can be specified by parameter setting.

T

In lathe systems, part of the numeric following the T code is used also as a tool offset number specifying a value such as a tool offset value.
An offset number may be specified by the lowest one digit, lowest two digits, or lowest three digits of the numeric following the T code as shown below.
The number of the digits of the offset number can be selected by parameter setting.

- **Specifying an offset number with the lowest one digit**

T○○○○○○○ ☆
↑ ↑
Tool number Offset number

- **Specifying an offset number with the lowest two digits**

  T○○○○○○ ☆☆
  ↑ ↑
  Tool number Offset number

- **Specifying an offset number with the lowest three digits**

  T○○○○○ ☆☆☆
  ↑ ↑
  Tool number Offset number

Specifying an offset number means selecting the corresponding offset value and also means starting tool offset.
When 0 is specified as the offset number, tool offset is canceled.

8.2 EXTENDED TOOL SELECTION FUNCTION

T

In lathe system machines, tools are changed mainly with the following two methods:
(1) With a turret holding multiple tools, tools are changed by turning the turret (T code).
(2) With an automatic tool changer (ATC), tools are changed by using both cartridge indexing (T code) and tool change (such as M06) commands.

To support the tool change method explained in (2) above, the following tool selection specifications apply to this function:
<1> Tool compensation by a T code is disabled. This means that the T code performs auxiliary functions only.
<2> Tool compensation is enabled by using a G code instead of the T code. In this case, the following types of tool compensation are enabled:
  - Tool length compensation (G43)
  - Tool offset (G43.7) (compensation equivalent to that of a T code)
  - Compensation cancel (G49)

Format

M06 T_ ; Change tool
:
Gxx D_ ; Tool compensation start
:
: Machining program
G49 ; Tool compensation cancel

Gxx : Type of tool compensation
  G43/G44 : Tool length compensation
  G43.7 : Tool offset
D_ : Tool compensation number

8.3 TOOL OFFSET PAIRS

M

The number of tool offset pairs used in the entire system is 400.
In a multi-path system, the number of tool offset pairs used in each path can be set by a parameter. The number of tool offset pairs used in the entire system must not exceed 400.

T

The number of tool offset pairs used in the entire system is 128.
In a multi-path system, the number of tool offset pairs used in each path can be set by a parameter. The number of tool offset pairs used in the entire system must not exceed 128.

NOTE

The number of tool offset pairs in the entire system can be expanded to 200 with option.

8.4 TOOL COMPENSATION MEMORY

M

One of the tool compensation memory A/C can be selected according to the configuration of offset amount.

Explanation

- Tool compensation memory A

There is no difference between geometry compensation memory and wear compensation memory in tool compensation memory A. Therefore, amount of geometry offset and wear offset together is set as the offset memory. There is also no differences between cutter compensation (D code) and tool length compensation (H code).

Fig. 8.4 (a) Setting example of tool compensation memory A

Offset number	Compensation value (geometry + wear)	Common to D and H codes
001	10.000	For D code
002	20.000	For D code
003	100.000	For H code
:	:	:

- Tool compensation memory C

Memory for geometry compensation and wear compensation is separate in tool compensation memory C. Geometry compensation and wear compensation can thus be set separately. Separate memories are prepared for cutter compensation (for D code) and for tool length compensation (for H code).

Fig. 8.4 (b) Setting example of tool compensation memory C

Offset number	D code		H code	
	For geometry compensation	For wear compensation	For geometry compensation	For wear compensation
001	10.000	0.100	100.000	0.100
002	20.000	0.200	200.000	0.300
:	:	:	:	:

T

- Tool offset

When the tool offset function is used, data that can be set in tool compensation memory is the compensation values for the X- and Z-axes.

- Tool nose radius compensation

When the cutter or tool nose radius compensation function is provided, compensation value R and imaginary tool nose direction T are added to the tool compensation memory setting items.

- Y-axis offset

When Y-axis offset is provided, the Y-axis offset setting screen is added. Data that can be set in Y-axis offset memory is the compensation value for the Y-axis.

- Second geometry tool offset

For compensation for the difference in tool mounting position or in selected position, second geometry tool offset can be provided in addition to tool offset. Data that can be set for second geometry tool offset is the compensation values for the X-, Z-, and Y-axes.

- Interference check for each path

A contact inhibition area for the interference check for each path can be set for each tool number. The data to be set for the contact inhibition area is X, Z, I, and K coordinates. These data items are not tool compensation values, but the number of sets varies depending on the number of tool compensation pairs.

- Tool geometry / wear compensation

The tool compensation memory configuration differs depending on whether the tool geometry and wear compensation functions are provided or not. The following data items in tool compensation memory are affected:

1. X- and Z-axis compensation values in tool offset
2. Compensation value R when tool nose radius compensation is provided
3. Y-axis compensation value when Y-axis offset is provided

When the tool geometry and wear compensation functions are not provided, there is no distinction between geometry compensation memory and wear compensation memory. Therefore, a sum of the geometry compensation value and wear compensation value is set in compensation memory.

When the tool geometry and wear compensation functions are provided, geometry compensation memory and wear compensation memory are prepared separately. Therefore, geometry compensation values and wear compensation values can be set separately.

8.5 COMMON COMPENSATION MEMORY BETWEEN EACH PATH

In multi-path systems, part or all of tool compensation memory provided for each path can be shared among multiple paths by parameter setting.

Example)

Sharing tool compensation numbers 1 to 20 between path 1 and path 2

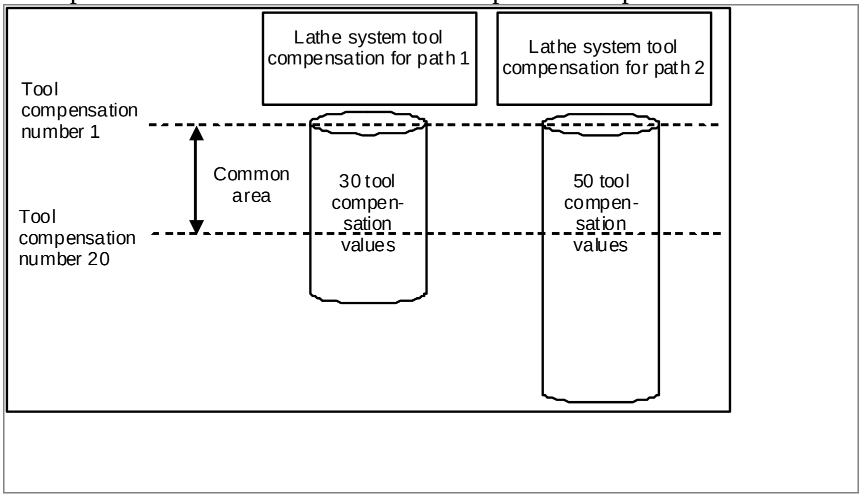


Fig. 8.5 (a) Sharing tool compensation numbers 1 to 20 between path 1 and path 2

8.6 TOOL LENGTH COMPENSATION (M SERIES)

M

When the difference between the tool length assumed at the time of programming and the tool length of the tool actually used for machining is set in offset memory, the difference in tool length can be corrected without modifying the program.

G43 and G44 specify the offset direction, and a number following the tool length compensation specification address (H code) specifies the tool length compensation amount set in the offset memory.

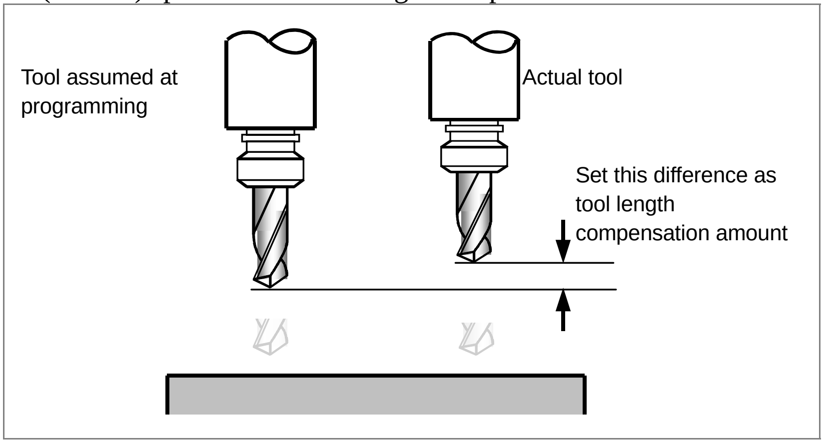


Fig. 8.6 (a) Tool length compensation

Format

Type	Format	Description
Tool length compensation A	**G43 Z_ H_ ;** **G44 Z_ H_ ;**	G43 : Positive offset G44 : Negative offset G17 : XY plane selection G18 : ZX plane selection G19 : YZ plane selection α : Address of a specified axis H : Address for specifying the tool length compensation value X, Y, Z : Offset move command
Tool length compensation B	**G17 G43 Z_ H_ ;** **G17 G44 Z_ H_ ;** **G18 G43 Y_ H_ ;** **G18 G44 Y_ H_ ;** **G19 G43 X_ H_ ;** **G19 G44 X_ H_ ;**	
Tool length compensation C	**G43 α_H_ ;** **G44 α_H_ ;**	
Tool length compensation cancel	**G49 ; or H0 ;**	

Explanation

One of the following three methods is available, depending on the type of axis that can be subject to tool length compensation:

- **Tool length compensation A**
  Compensates for the difference in tool length along the basic Z-axis.
- **Tool length compensation B**
  Compensates for the difference in tool length in the direction normal to a selected plane.
- **Tool length compensation C**
  Compensates for the difference in tool length along a specified axis.

One of the above methods can be selected by parameter setting.

- Direction of the offset

When G43 is specified, the tool length compensation 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.

When the specification of an axis is omitted, a movement is made by the tool length compensation value.

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 compensation value

The tool length compensation value assigned to the number (offset number) specified in the H code is selected from offset memory and added to or subtracted from the moving command in the program.

A tool length compensation value is to be set in the offset memory corresponding to an offset number.

- Specification of tool length compensation along two or more axes

Tool length compensation B can be executed along two or more axes when the axes are specified in two or more blocks.

By setting the parameter, cutter compensation C can also can be executed along two or more axes when the axes are specified in two or more blocks.

- Tool length compensation cancel

To cancel tool length compensation, specify G49 or H0. After G49 or H0 is specified, the system immediately cancels the offset mode.

8.7 TOOL OFFSET

M

The programmed travel distance of the tool can be increased or decreased by a specified tool offset value or by twice the offset value.

The tool offset function can also be applied to an additional axis.

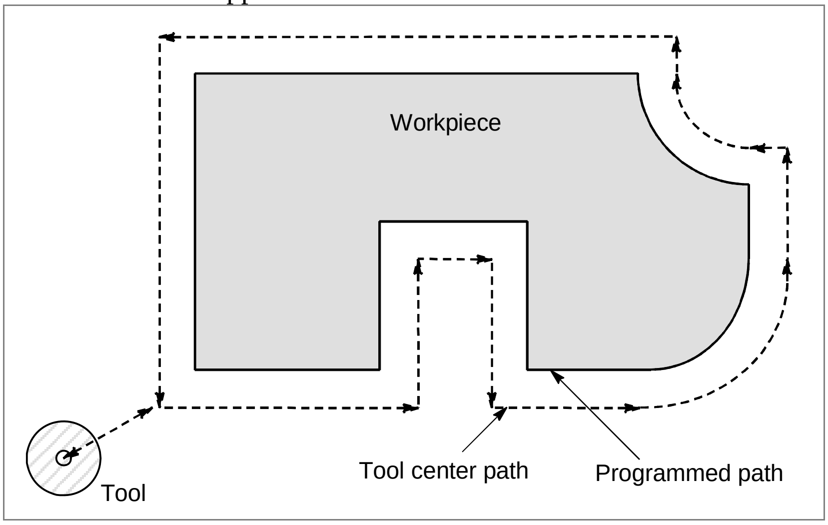


Fig. 8.7 (a) Tool offset

Format

G45 IP_ D_ ; Increase the travel distance by the tool offset value

G46 IP_ D_ ; Decrease the travel distance by the tool offset value

G47 IP_ D_ ; Increase the travel distance by twice the tool offset value

G48 IP_ D_ ; Decrease the travel distance by twice the tool offset value

G45 to 48 : One-shot G code for increasing or decreasing the travel distance
IP_ : Command for moving the tool
D Code for specifying the tool offset value

T

Tool offset is used to compensate for the difference when the tool actually used differs from the imagined tool used in programming (usually, standard tool).

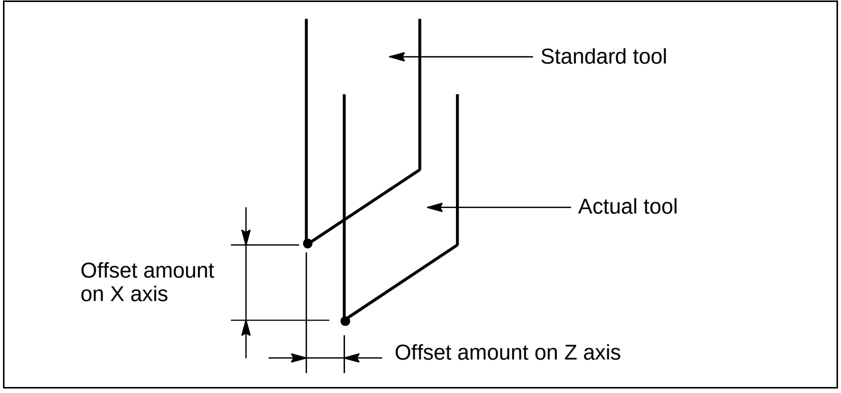


Fig. 8.7 (b) Tool offset

8.8 Y-AXIS OFFSET (T SERIES)

T

When the Y axis, one of the basic three axes, is used with a lathe system, this function performs Y axis offset.

NOTE

1 When Y-axis offset is used, the Y-axis must always be a linear axis.

2 Y-axis offset does not allow direct input of tool offset values.

8.9 CUTTER OR TOOL NOSE RADIUS COMPENSATION

- Cutter compensation

Use of this function can offset a programmed tool path by the tool radius set in the CNC when machining is performed.

When the radius of the tool to be used for machining is measured and set as the offset value in the CNC, the tool moves along the offset path to cut a programmed profile. Therefore, even when the tool diameter changes, you must only change the offset value and need not modify the program.

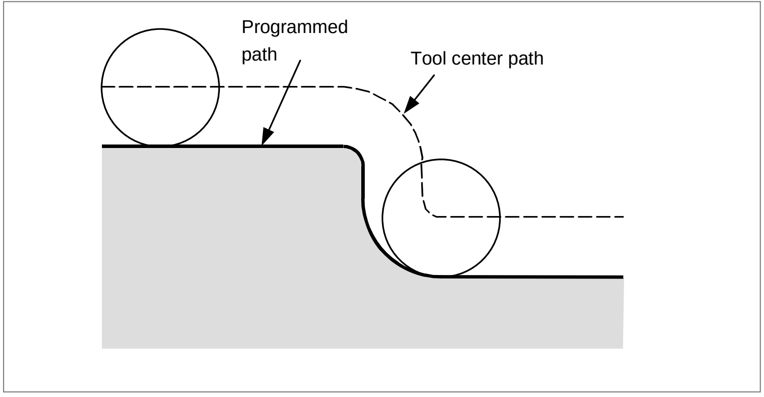


Fig. 8.9 (a) Cutter compensation

- Tool nose radius compensation

Use of this function can offset a programmed tool path by the tool nose radius set in the CNC when machining is performed. When a machining profile is programmed using this function, and the radius of the tool nose to be used for actual machining is measured and set as the offset value in the CNC, the tool moves along the offset path to cut the programmed profile. Therefore, even when the tool nose radius changes, you must only change the offset value and need not modify the program.

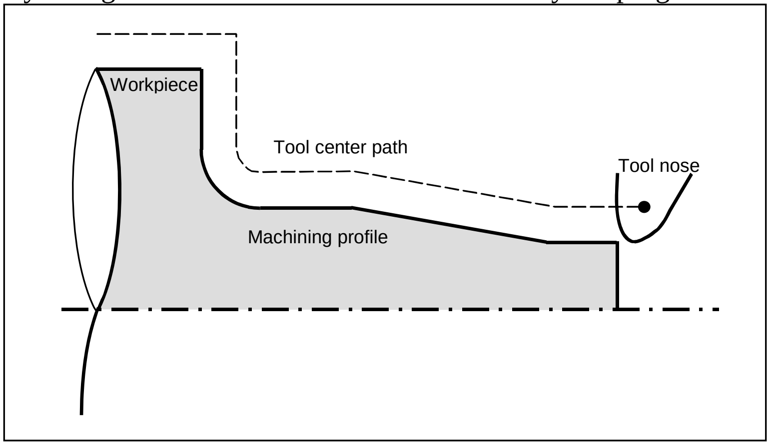


Fig. 8.9 (b) Tool nose radius compensation

To obtain the actual offset tool path, the CNC internally calculates intersections of a straight line and a straight line, an arc and an arc, and a straight line and an arc automatically. The programmer only has to program a machining profile, therefore the programming can be done very easily.

Format

G40 : Cutter or tool nose radius compensation cancel
G41 : Cutter or tool nose radius compensation : left
G42 : Cutter or tool nose radius compensation : right

Explanation

- Cutter or tool nose radius compensation cancel

G41 and G42 are commands to place the system in cutter or tool nose radius compensation mode. G41 causes offset on the left side of the tool advance direction, and G42 causes offset on the right side of the tool advance direction. G40 cancels cutter or tool nose radius compensation.

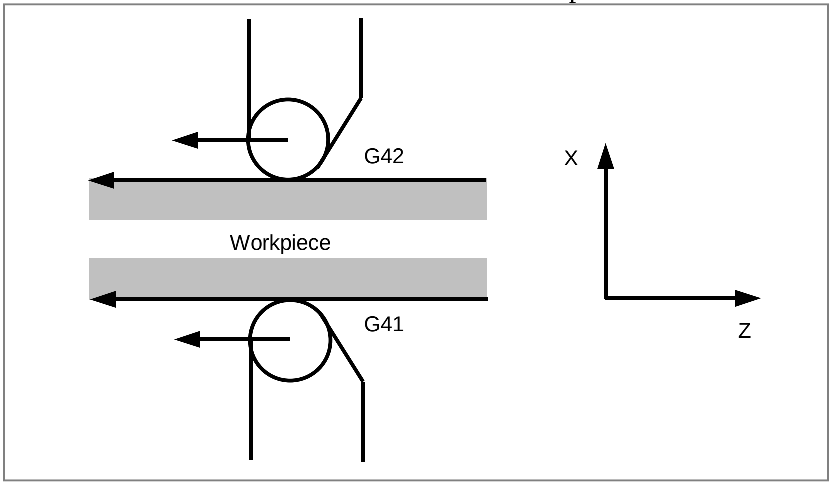


Fig. 8.9 (c) Cutter or tool nose radius compensation cancel

- Imaginary tool nose (tool nose radius compensation)

The tool nose at position A in Fig. 8.9 (d) does not actually exist.

The imaginary tool nose is required because it is usually more difficult to set the actual tool nose radius center to the start point than the imaginary tool nose.

Also when imaginary tool nose is used, the tool nose radius need not be considered in programming. Programming can be performed in the same way as when a tool having no tool nose radius is used.

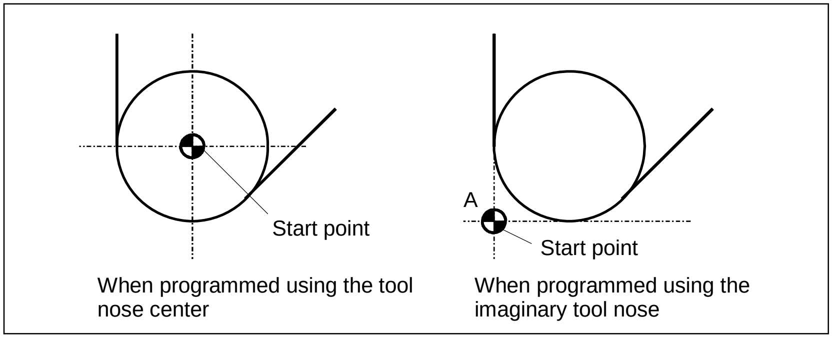


Fig. 8.9 (d) Imaginary tool nose (tool nose radius compensation)

The position relationship when the tool is set to the start point is shown in Fig. 8.9 (e).

The start point or reference position in the tool nose can be set in offset memory in the same way as the tool nose radius compensation value.

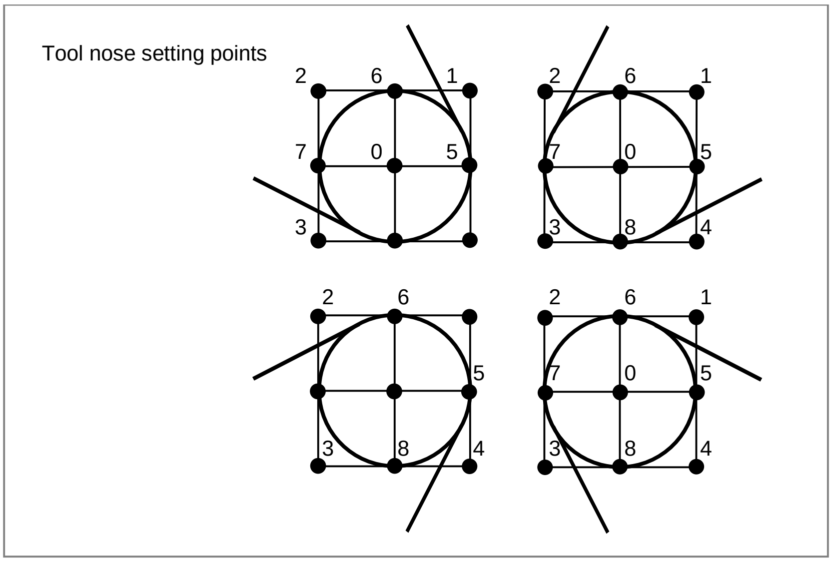


Fig. 8.9 (e) Tool nose setting points

- Specifying a compensation value and imaginary tool nose position (T code, T series)

A compensation value and imaginary tool nose position can be set in tool compensation memory.

By specifying an offset number in part of a T code, the cutter or tool nose radius compensation values and imaginary tool nose position in the tool compensation memory area corresponding to the offset number are selected.

- Specifying a compensation value and imaginary tool nose position (D code, M series)

A compensation value and imaginary tool nose position can be set in tool compensation memory.

By specifying an offset number in a D code, the cutter or tool nose radius compensation values and imaginary tool nose position in the tool compensation memory area corresponding to the offset number are selected.

An offset number can be specified in a H code by parameter setting.

- Plane selection (G17, G18, G19)

Cutter or tool nose radius compensation can be applied not only to the XY, ZX, and YZ planes but also axes parallel to X, Y, and Z.

The target plane for cutter or tool nose radius compensation is selected by specifying G17, G18, or G19.

G17 : Xp-Yp plane (Xp: X axis or its parallel axis)
G18 : Zp-Xp plane (Yp: Y axis or its parallel axis)
G19 : Yp-Zp plane (Zp: Z axis or its parallel axis)

An additional axis can be set as an axis parallel to the X-, Y-, or Z-axis by parameter setting.

- Vector retention (G38)

In cutter or tool nose radius compensation, specifying G38 in offset mode can maintain the compensation vector at the end point of the previous block without calculation of the intersection.

- Corner circular interpolation (G39)

By specifying G39 in offset mode during cutter or tool nose radius compensation, corner circular interpolation can be performed. The radius of the corner circular interpolation equals the compensation value.

- Interference check

A state in which the tool cuts a workpiece excessively is called "interference". A check is made to see whether interference occurs as a result of cutter or tool nose radius compensation.

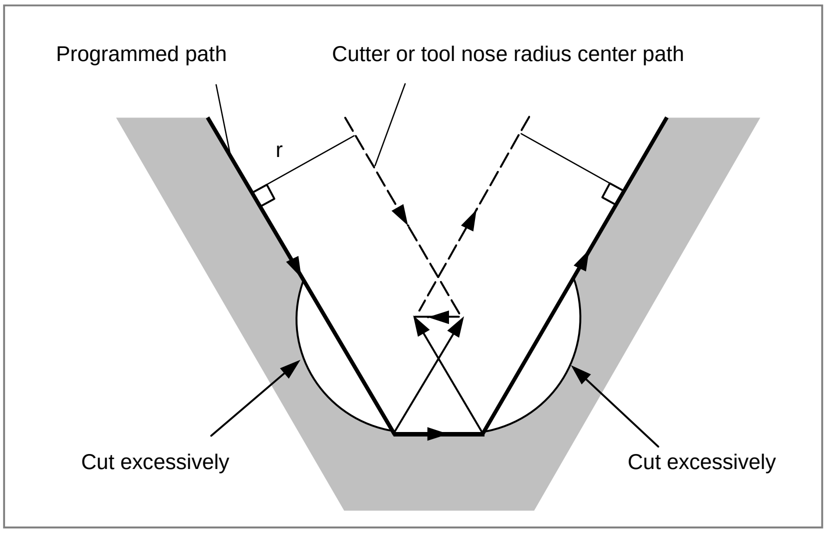


Fig. 8.9 (f) Interference check

8.10 CUTTING POINT INTERPOLATION FOR CYLINDRICAL INTERPOLATION

The conventional cylindrical interpolation function controls the tool center so that the tool axis can always move along a specified path on the cylindrical surface, towards the rotary axis (cylindrical axis) of the workpiece. The cutting point interpolation for cylindrical interpolation function controls the tool so that the tangent of the tool and the cutting surface of a contour can always pass the rotation center of the workpiece. This means that the cutting surface of the contour is always perpendicular to the cylinder. With this function, the figure on the cutting surface can always be kept constant regardless of the cutter compensation value of the tool used.

NOTE
This function is an optional function.

Format

This command is specified in the same way as for the conventional cylindrical interpolation function.

G07.1 IPr ; Circular interpolation mode on start (enabling cylindrical interpolation)
:
G07.1 IP0 ; Circular interpolation mode cancel

IP : One rotary axis address
r : Cylinder radius of rotary axis
Specify each of G07.1 IPr; and G07.1 IP0; singly in a block.
G107 can also be specified.

Explanation

- Comparison with conventional circular interpolation

As shown in Fig. 8.10 (a), the tool is controlled in the offset axis (Y-axis) direction that is perpendicular to the tool center and workpiece rotation center.

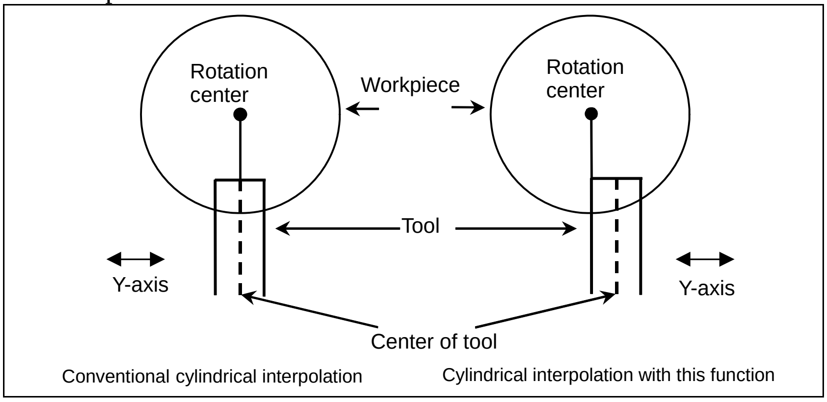
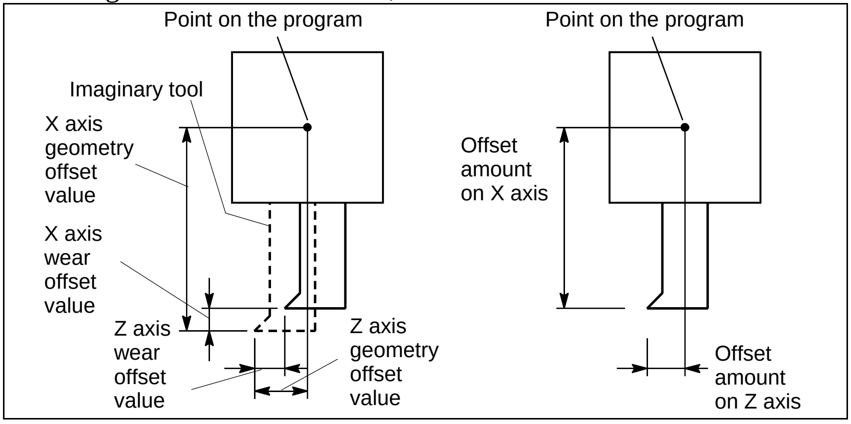


Fig. 8.10 (a) Comparison with conventional circular interpolation

8.11 TOOL GEOMETRY OFFSET AND TOOL WEAR OFFSET (T SERIES)

T

Tool geometry offset and tool wear offset are possible to divide the tool offset to the tool geometry offset for compensating the tool shape or tool mounting position and the tool wear offset for compensating the tool nose wear. The tool geometry offset value and tool wear offset value can be set individually. When these values are not distinguished from each other, the total of the values is set as the tool offset value.



If tool geometry offset and tool wear offset are distinguished from each other

If tool geometry offset and tool wear offset are not distinguished from each other

Fig. 8.11 (a) Tool geometry offset and tool wear offset

8.12 SECOND GEOMETRY TOOL OFFSET (T SERIES)

T

This function allows use of second geometry tool offset in addition to tool offset in order to compensate for the difference in tool mounting position or in selected position. Data that can be set for second geometry tool offset is the X-, Z-, and Y-axis compensation values.
In contrast to this offset, the ordinary tool geometry offset is called the first geometry tool offset.
It is possible to apply a tool offset value (tool wear offset + tool geometry offset) in the reverse direction, using an appropriate signal.
This function may be used if the offset value differs even with a single tool for a mechanical reason depending on the mounting position (inside/outside) or the selected position (right/left).

NOTE
This function is an optional function.

Format

- When assigning the same number as the tool selection number to the geometry offset number

M○○ (M code that enables second geometry tool offset) ;

T○○○○ ○○ ;

Tool wear offset number
First geometry tool offset number or first + second geometry tool offset numbers

- When assigning the same number as the wear offset number to the geometry offset number

M○○ (M code that enables second geometry tool offset) ;

T○○○○ ○○ ;

Tool wear offset number + first geometry tool offset number or wear + first + second geometry tool offset numbers

8.13 TOOL MANAGEMENT FUNCTION

The tool management function totally manages tool information including information about tool offset and tool life.

A tool type number is specified with a T code. The tool type number is any number the user can define freely. With tool type numbers, tools can be grouped by various conditions such as life, compensation value, and cutting conditions. When each type is assumed to have a single tool, tool type numbers are equivalent to unique tool numbers.

For each tool, an information storage area is prepared in the CNC. This area contains information such as a tool type number, tool life, tool status (such as a breakage condition), tool compensation number, spindle speed, cutting feedrate, and freely-definable customize data. Such data is called tool management data.

A cartridge management table that links cartridge information and tool management data is provided so that the CNC can manage the cartridges of the machine and tool change operations. In addition, areas for managing the tools in the spindle and tool standby positions are prepared.

When a tool type number is specified using a T code command, a tool that has the tool type number and the shortest life is searched for, and the cartridge number and pot number of the location where the tool is stored are output to the PMC. Then, a tool change operation using the cartridge number and pot number is enabled by the PMC ladder sequence.

Tool information in the CNC is managed by the tool management data and cartridge management table (including the spindle management table and standby position table).

NOTE

1. There are three options for this function: options for up to 64 pairs, for 240 pairs, and for 1000 pairs. Select one of the options according to the number of tool management data pairs used.
2. Although customize data 0 to 4 are normally used, expanded customize data 5 to 20 or 5 to 40 are made available by using an option.

8.13.1 Tool Management Extension Function

- Customization of tool management data display

With the tool management data screen display customization function, the display positions of screen elements (type number, tool information, life counter, and so forth) on the tool management screen can be changed and whether to display or hide such screen elements can be chosen using the G10 format. This function enables a customized tool management screen to be configured.

- Setting of spindle position/standby position display

In MG on the tool management data screen, a spindle position or standby position is displayed as a number such as 11, 12, and 13. With the spindle position/standby position display setting function, three arbitrary characters can be displayed using the G10 format.

- Input of customize data with the decimal point

With the function for input of customize data with the decimal point, the number of decimal places can be set using the G10 format for each customize data item (customize data 1, ..., 40) to enable data input with the decimal point.

- Protection of various tool information items with the KEY signal

When tool management data is in the edit state, various information items can be modified. By setting the parameter, tool management data can be protected with the KEY signal so that various information items are not registered, modified, and deleted.

- Selection of a tool life count period

A tool life count period can be chosen between 1 sec and 8 msec on a tool-by-tool basis.

- Individual data screen

All data for a specified tool can be extracted and displayed.

- Total life time display for tools of the same type

The remaining lives of tools with the same type numbers are totaled, and totals are displayed in order by tool type number or by remaining life. Also, tools with the same tool type numbers are displayed in a list.

NOTE

This function is an optional function.
To use this function, the both options for "Tool management function" and this function are required.

8.13.2 Tool Management Function Oversize Tools Support

Tool management function oversize tools support is added to the tool management function.
The figure of an oversize tool can be defined freely, and the figure of each oversize tool is registered. When an oversize tool is stored in a cartridge, interference with tools stored in other pots is considered. This function is usable with cartridges of chain type and matrix type.

NOTE

This function is included in the option "Tool management extension B function".
To use this function, the both options for "Tool management function" and "Tool management extension B function" are required.

8.14 TOOL OFFSET VALUE COUNTER INPUT (T SERIES)

T

A value displayed as a relative coordinate position can be set as an offset value in tool compensation memory by using a soft key.
First, display the offset value screen. This screen also displays the relative coordinate position. Next, select a reference tool, and manually set it at the fixed point on the machine. Reset the relative coordinates displayed on the screen to zero. Then, manually set the tool to be measured at the same fixed point on the machine. At this time, the relative coordinate position shows the difference between the reference tool and the tool to be measured, and this value can be set as the offset value.

8.15 TOOL LENGTH MEASUREMENT (M SERIES)

M

The value displayed as a relative position can be set in the offset memory as an offset value by a soft key.
Switch to the offset value display screen on the screen. Relative positions are also displayed on this screen. Then select the reference tool and set it at the fixed point on the machine by manual operation. Reset the displayed relative position to zero. Set the tool for measurement at the same fixed point on the machine by manual operation. The relative position display at this point shows difference between the reference tool and the tool measured and the relative position display value is then set as offset amounts.

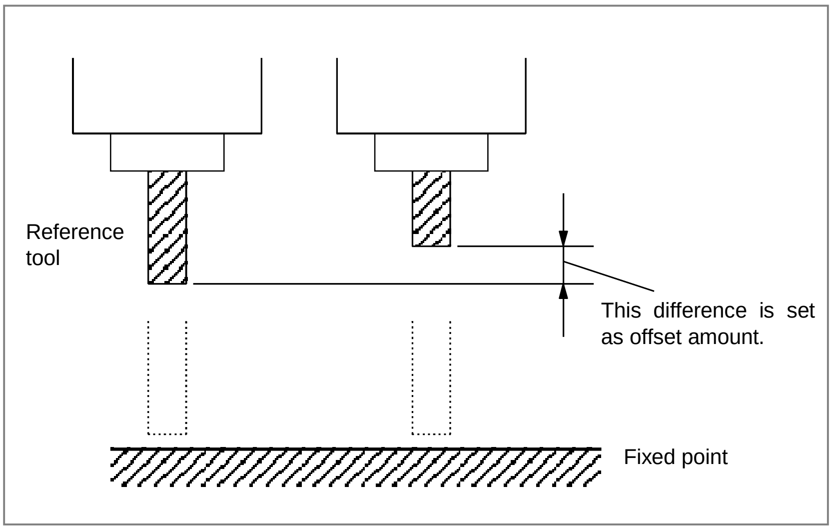


Fig. 8.15 (a) Tool length measurement

8.16 AUTOMATIC TOOL LENGTH MEASUREMENT / AUTOMATIC TOOL OFFSET

8.16.1 Automatic Tool Length Measurement (M SERIES)

M

Difference between the coordinate value of tool when tool end has reached the measuring position and coordinate value of the measuring position is automatically measured, calculated, and added to the currently set tool offset amount by CNC system. The machine must be equipped with measuring devices, for example touch sensor, so that a signal is sent when the tool end has reached the measuring position.

Format

G92 IP_ ;	Sets the workpiece coordinate system. (It can be set with G54 to G59.)
Hxx ;	Specifies an offset number for tool length compensation.
G90 G37 IP_ ;	Absolute programming

G37 is valid only in the block in which it is specified.
IP_ indicates the X-, Y-, Z-, or fourth axis.

Explanation

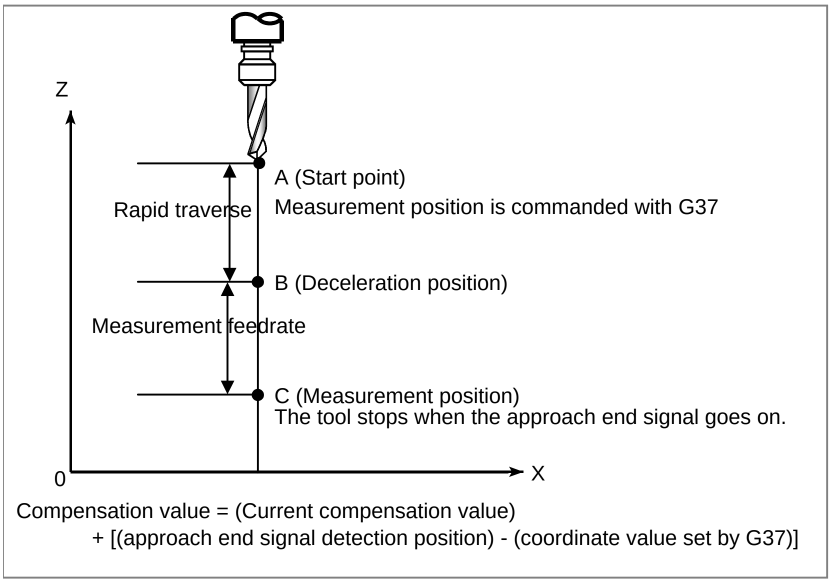


Fig. 8.16 (a) Automatic tool length measurement

When G37 is specified, the tool moves from the start point to deceleration position by rapid traverse, then from the deceleration position the tool decelerates to a parameter-set measurement federate and moves until the approach end signal is issued from the measuring device. That is, the tool movement stops when the tool tip has reached the measurement position.

The difference between the coordinate value of the tool when it reaches the measurement position and the coordinate value specified in G37 is added to the tool length compensation value currently used.

8.16.2 Automatic Tool Offset (T SERIES)

T

Difference between the coordinate value of tool when tool end has reached the measuring position and coordinate value of the measuring position is automatically measured, calculated, and added to the currently set tool offset amount by CNC system. The machine must be equipped with measuring devices, for example touch sensor, so that a signal is sent when the tool end has reached the measuring position.

Format

G36 X_ ; or G37 Z_ ;

G36 and G37 are effective only in the block in which these commands are specified.
X_, Z_ : Absolute programming
(G37.1X_; or G37.2Z_, which depends on the parameter setting.)

Explanation

Execution of above command moves the tool at the rapid traverse rate toward the measurement position, lowers the feedrate halfway, then continues to move it until the approach end signal from the measuring instrument is issued.

When the tool tip reaches the measurement position, the measuring instrument outputs the measurement position reach signal to the CNC which stops the tool.

The difference between the coordinate value of the tool when the measurement position is reached and the coordinate value specified by G36 or G37 is added to the currently used tool offset value.

8.17 DIRECT INPUT OF TOOL OFFSET VALUE MEASURED / DIRECT INPUT OF COORDINATE SYSTEM SHIFT (T SERIES)

T

This is a function of setting an offset value by key-inputting a workpiece diameter manually cut and measured from the MDI keyboard.

First the workpiece is cut in the longitudinal or in the cross direction manually. When a button on the machine operator's panel is pressed upon completion of the cutting, the workpiece coordinate value at that time is recorded. Then, withdraw the tool, stop the spindle, and measure the diameter if the cutting was on the longitudinal direction or distance from the standard face if it was on the facing. (The standard face is made as Z = 0.) When the measured value is entered into the offset number desired plus 100, NC inputs the difference between the input measured value and the coordinate value recorded in NC, as the offset value of the offset number.

The workpiece coordinate system can be shifted using the technique of directly inputting the measured value for offset. This technique is used when the coordinate system planned in the program does not match with the coordinate system set by the G92 command or by the automatic coordinate system setting.

The procedures are the same as those for direct input for offset, except a difference of using the standard tool.

When offset and shift values are set, a movement can be made along the axis for which the values are set to retract the tool by using the position record signal.

8.18 DIRECT INPUT OF TOOL OFFSET VALUE MEASURED B (T SERIES)

T

By installing the touch sensor and by manually making the tool contact the touch sensor, it is possible to set the offset amount of that tool automatically in the tool offset amount memory.

It is also possible to set the workpiece coordinate system shift amount automatically.

The tool setter function for single-turret two-spindle lathes is also available, which allows use of direct input of offset value measured B for 2 spindle lathe.

8.19 TOOL LIFE MANAGEMENT

Tools are classified into several groups, and a tool life (use count or use duration) is specified for each group in advance. Each time a tool is used, its life is counted, and when the tool life expires, a new tool that is sequenced next within the same group is selected automatically. With this function, the tool life can be managed while machining is being performed continuously. Data for tool life management consists of tool group numbers, tool life values, tool numbers, and codes for specifying a tool offset value. These data items are registered in the CNC.

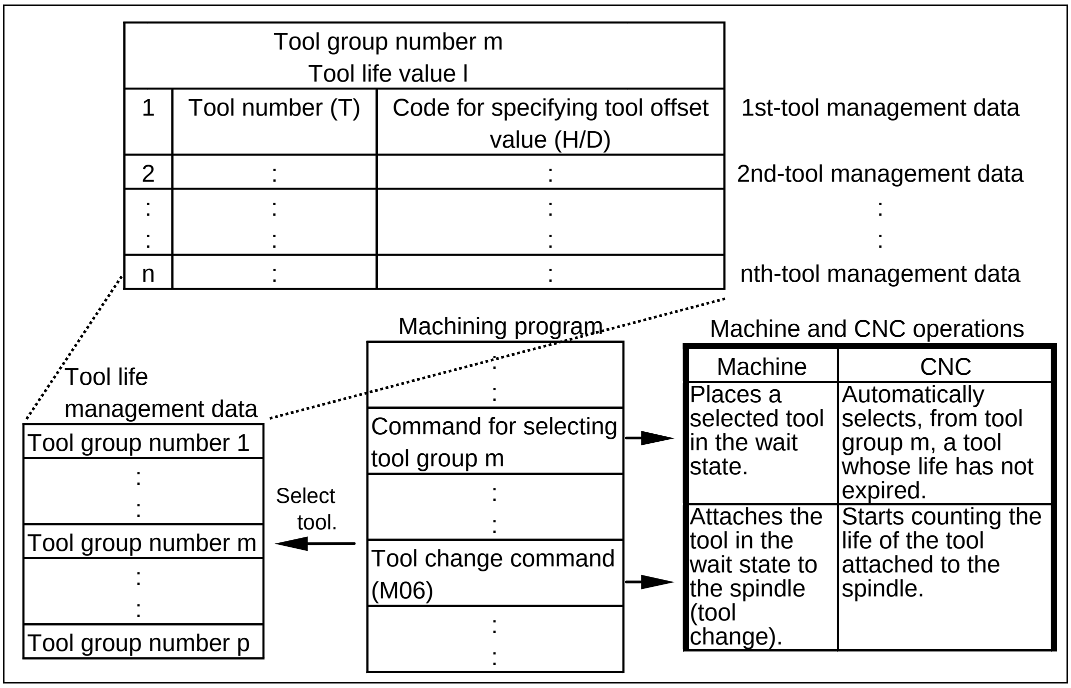


Fig. 8.19 (a) Tool selection from machining program

M

A group is selected by a T code, and tool life counting is started by the M06 command.

T

The T series has two tool change types (the turret type and the ATC type). The turret type uses only a T code to select a group, specify a tool offset value, and start tool life counting, whereas the ATC type, like the M series, uses a T code to select a group and the M06 command to start tool life counting. With the ATC type, only a D code is used for specifying a tool offset value. The tool change type is selected by bit 3 (TCT) of parameter No. 5040.

⚠ **CAUTION**

This function cannot be used if bit 1 (LGN) of parameter No. 5002 is set to 1 to use the same number as the tool selection number to specify a geometric offset number.

- Life management B function

If the tool life management B function is enabled, the maximum tool life value can be extended, and the tool life expiration prior notice signal can be output to post tool life expiration in advance when the remaining life (the life value minus the life counter value) has reached the remaining life setting. The remaining life setting is registered as tool life management data in the CNC in advance.

The tool life management B function is enabled by setting bit 4 (LFB) of parameter No. 6805 to 1.

M

If the tool life management B function is enabled, the function for selecting a tool group by an arbitrary group number can be used.

T

The tool life management B function can be used. However, the function for selecting a tool group by an arbitrary group number can be used only if the tool change type is the ATC type (bit 3 (TCT) of parameter No. 5040 = 1).

- Maximum number of tool life management groups and multipath system

Up to 256 tool life management groups can be used in the entire CNC system. For each path, set a maximum number of groups to be used in parameter No. 6813.

The maximum number of groups must be a multiple of the minimum number of groups (eight groups). If the maximum number of groups is 0, the tool life management function is disabled.

Multipath system (when tool life management is performed for three paths)
Suppose that, of the total groups in the system (256 groups), 128 groups are allocated to path 1, 64 groups to path 2, and 64 groups to path 3. Then, set parameter No. 6813 as follows:

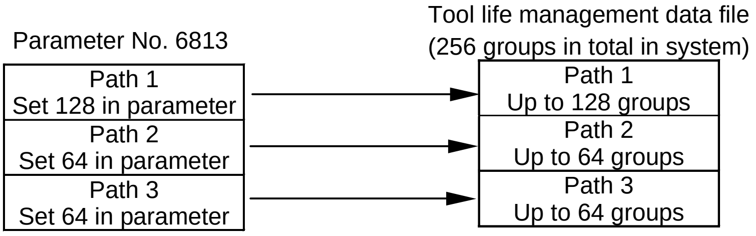


Fig. 8.19 (b) Group allocation in entire system

9 ACCURACY COMPENSASION FUNCTION

Chapter 9, "ACCURACY COMPENSASION FUNCTION", consists of the following sections:

9.1 BACKLASH COMPENSATION

Function for compensating for lost motion on the machine. Set a compensation value in parameter, in detection units from 0 to ±9999 pulses for each axis.

9.2 BACKLASH COMPENSATION FOR EACH RAPID TRAVERSE AND CUTTING FEED

Since different backlash compensation values can be used for cutting feed and rapid traverse, the machining precision is improved.
Let the measured backlash amount at the time of cutting feed be A and the measured backlash amount at rapid traverse be B. Then, the output backlash compensation value varies according to the change in feed (cutting feed and rapid traverse) and the change in movement direction as listed in the Table 9.2 (a):

Table 9.2 (a) Backlash compensation value output

Change in feed **Change in movement direction**	**Cutting feed** ↓ **Cutting feed**	**Rapid traverse** ↓ **Rapid traverse**	**Rapid traverse** ↓ **Cutting feed**	**Cutting feed** ↓ **Rapid traverse**
Movement in same direction	0	0	±α	±(-α)
Movement in opposite directions	±A	±B	±(B+α)	±(B+α)

- α= (A-B) / 2
- Signs (+ and –) indicate directions.

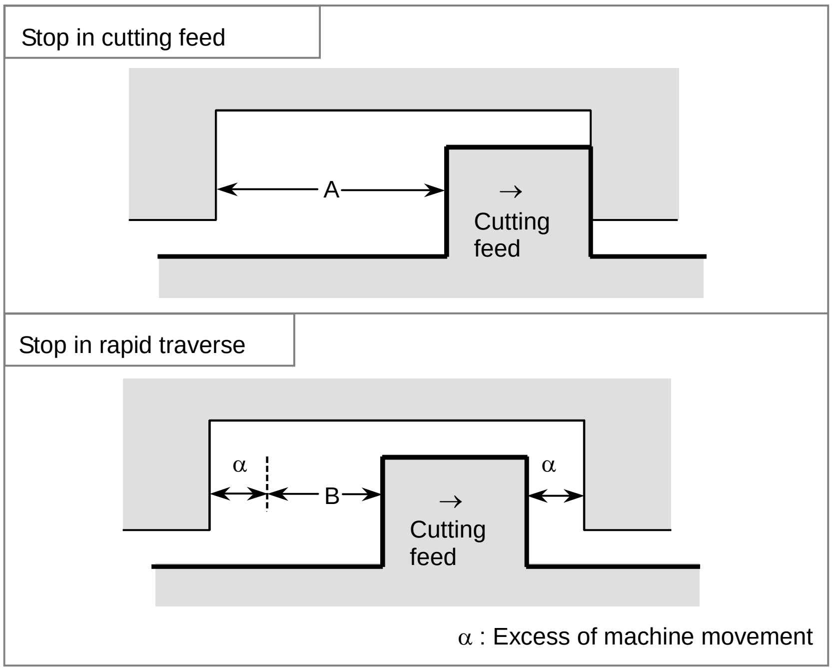


Fig. 9.2 (a) Backlash compensation

9.3 SMOOTH BACKLASH COMPENSATION

In ordinary backlash compensation, backlash compensation pulses are all output at a position where the axis movement direction is inverted.

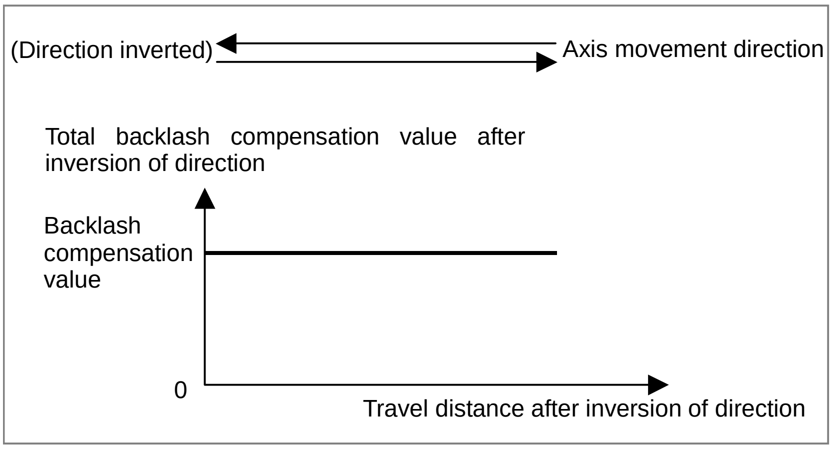
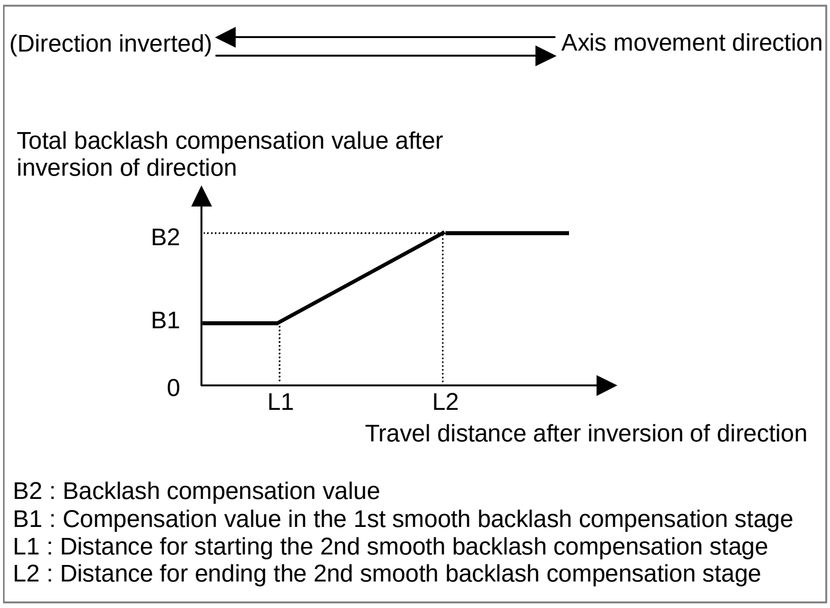


Fig. 9.3 (a) Ordinary backlash compensation

In smooth backlash compensation, backlash compensation pulses are output according to the distance from the position where the axis movement direction is inverted, so fine backlash compensation can be performed according to the machine characteristics.



B2 : Backlash compensation value
B1 : Compensation value in the 1st smooth backlash compensation stage
L1 : Distance for starting the 2nd smooth backlash compensation stage
L2 : Distance for ending the 2nd smooth backlash compensation stage

Fig. 9.3 (b) Smooth backlash compensation

9.4 STORED PITCH ERROR COMPENSATION

The errors caused by machine position, as pitch error of the feed screw, can be compensated. This function is for better machining precision.
The input type of pitch error compensation data can be selected from the incremental value type and the total value type by the parameter.
As the offset data are stored in the memory as parameters, compensations of dogs and settings can be omitted.
Offset intervals are set constant by parameters (per axis).

The total number of compensation points is as follows:

Total compensation points = 1536 points

Distribution to each axis can be determined arbitrarily (by parameter setting). The number of compensation pulses at each position is as follows:

Compensation pulses = (-128 to +127) × (magnification)

The compensation pulse unit is the same as the detection unit.
The magnification is as follows:

Magnification: 0 to 100, parameter setting (for each axis)

NOTE
This function is an optional function.

9.5 INTERPOLATION TYPE PITCH ERROR COMPENSATION

In stored pitch error compensation, the pitch error compensation pulse at each pitch error compensation point is output in the interval between that point and the next compensation point, as shown in the Fig. 9.5 (a).

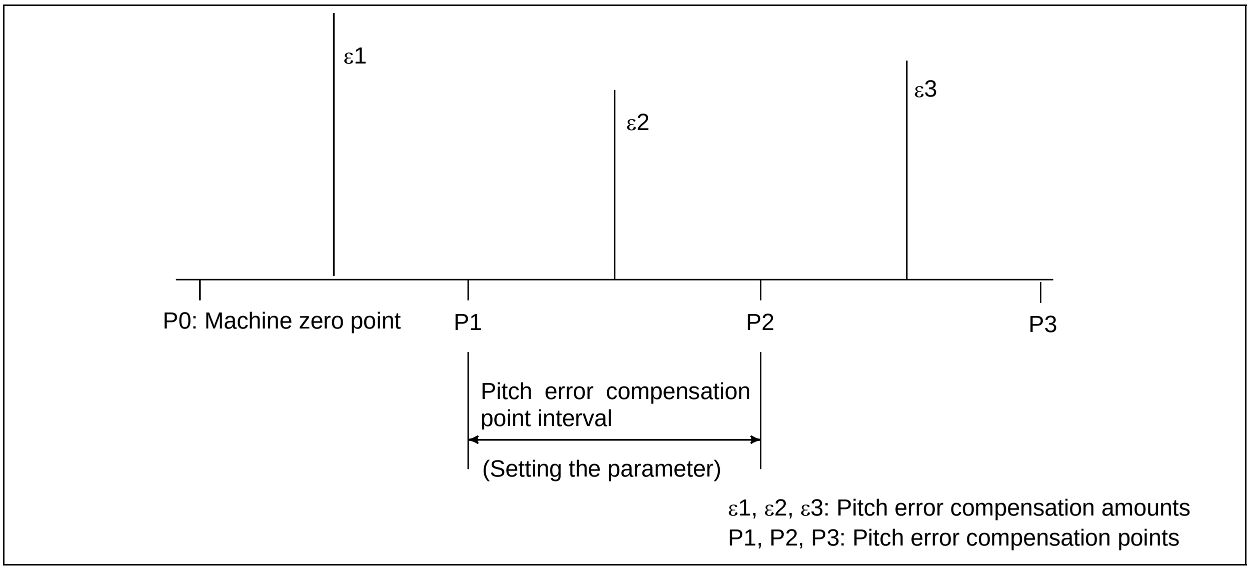


Fig. 9.5 (a) Stored pitch error compensation

In interpolation type pitch error compensation, the compensation amount at each error compensation point is divided into pulses in the interval between that point and the next point on the travel axis and output, as shown in the Fig. 9.5 (b).

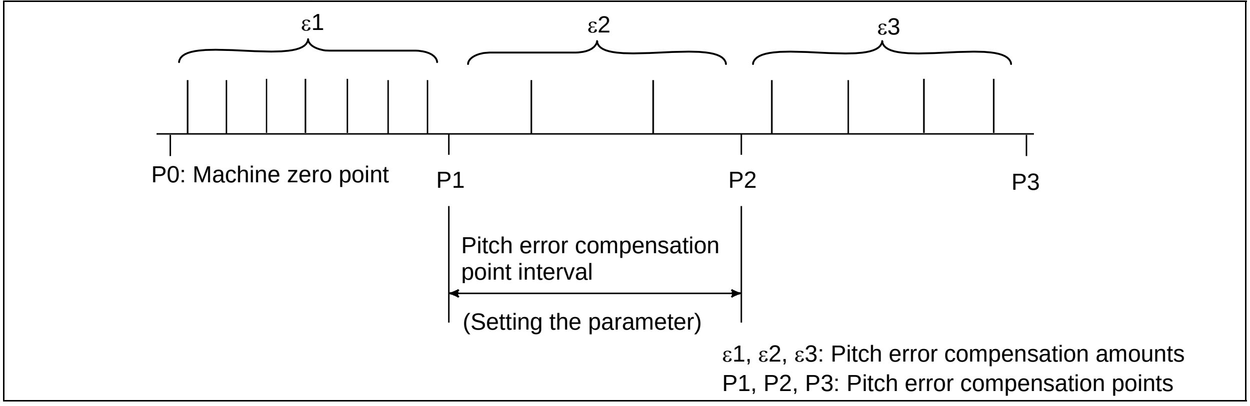


Fig. 9.5 (b) Interpolation type pitch error compensation

NOTE

This function is an optional function.
To use this function, the both options for "Stored pitch error compensation" and this function are required.

9.6 BI-DIRECTIONAL PITCH ERROR COMPENSATION

In bi-directional pitch error compensation, different pitch error compensation amounts can be set for travel in the positive direction and that in the negative direction, so that pitch error compensation can be performed differently in the two directions, in contrast to stored pitch error compensation, which does not distinguish between the directions of travel. In addition, when the direction of travel is reversed, the compensation amount is automatically calculated from the pitch error compensation data to perform compensation in the same way as in backlash compensation. This reduces the difference between the paths in the positive and negative directions.

NOTE

This function is an optional function.
To use this function, the both options for "Stored pitch error compensation" and this function are required.

9.7 EXTENDED BI-DIRECTIONAL PITCH ERROR COMPENSATION

In bi-directional pitch error compensation, it is possible to use 0 to 1535, 3000 to 4535 points as the compensation points. By using this function, the compensation points are extended and it is possible to use 0 to 2559, 3000 to 5559 points as the compensation points.

NOTE

This function is an optional function.
To use this function, the options for "Stored pitch error compensation", "Bi-directional pitch error compensation" and this function are required.

9.8 INCLINATION COMPENSATION

By compensating for those errors in tools such as feed screws that depend on the position of the machine system in detection units, machining precision can be improved and mechanical life can be prolonged. Compensation is performed along an approximate straight line formed with a parameter-specified compensation point and a compensation amount related to it.

Three approximate straight lines are formed with four parameter-specified compensation points and compensation amounts related to the respective compensation points. Inclination compensation is carried out along these approximate straight lines at pitch error compensation intervals. The inclination compensation amount is added to the pitch error compensation amount.

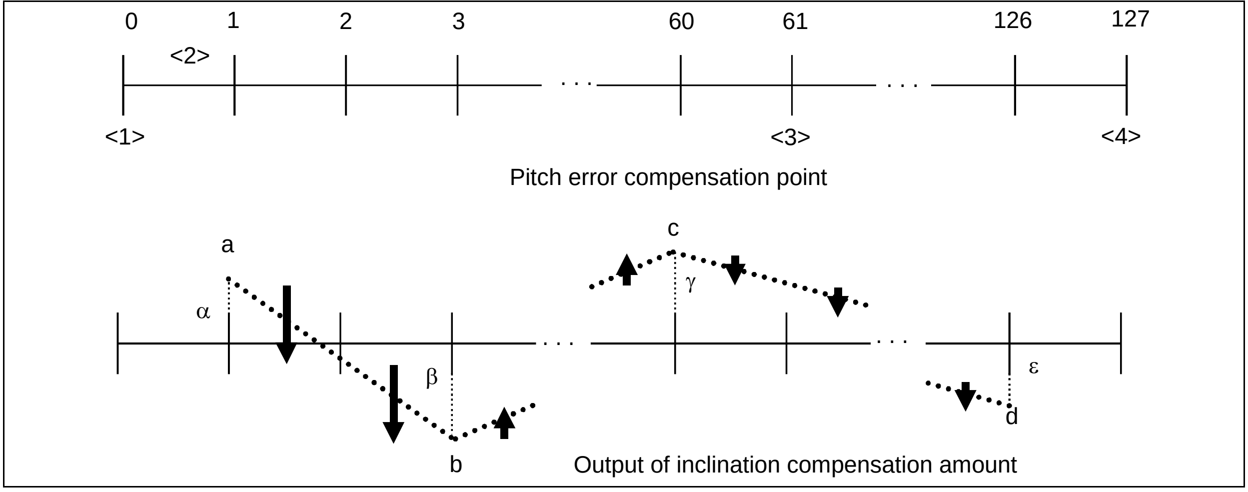


Fig. 9.8 (a)

To perform inclination compensation, stored pitch error compensation must be set for the axis subject to compensation.

(1) Number of the most distant pitch error compensation point on the - side
(2) Pitch error compensation point interval
(3) Number of the pitch error compensation point of the reference position
(4) Number of the most distant pitch error compensation point on the + side

Inclination compensation parameters must be set.

a,b,c,d : Compensation point numbers.

α,β,γ,ε : Compensation amounts at compensation points a, b, c, and d

In Fig. 9.8 (a), a, b, c, and d are 1, 3, 60, and 126, respectively.

Unlike stored pitch error compensation, whose amount is set up for an individual compensation point, an amount of inclination compensation is calculated for individual compensation points by setting up four typical points and compensation amounts for them.

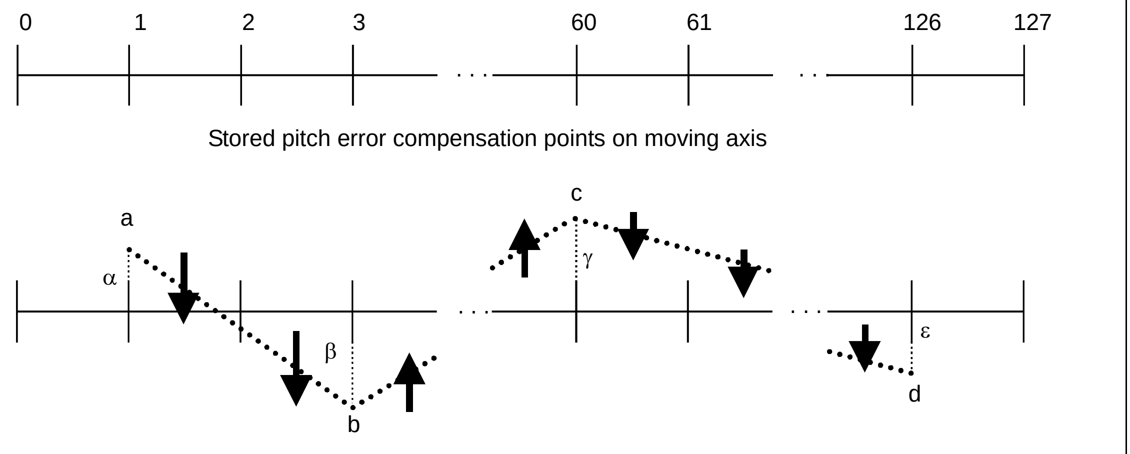
Example)

In Fig. 9.8 (a), the compensation amounts at the individual compensation points located between points a and b are (β-α)/(b-a).

9.9 STRAIGHTNESS COMPENSATION

For a machine tool with a long stroke, deviations in straightness between axes may affect the machining accuracy. For this reason, when an axis moves, other axes are compensated in detection units to improve straightness. This improvement results in better machining accuracy. As a moving axis (set in a parameter) moves, compensation is applied to the compensation axis (set in a parameter) during pitch error compensation for the moving axis.

- Pitch error compensation points on moving axis and compensation value



Output timing of stored pitch error compensation of moving axis and output compensation pulses

a , b , c , d:	Compensation position numbers of the moving axis (This number is originally a pitch error compensation position number.)
α, β, γ, ε :	Compensation amount for the compensation position number (Amount of compensation for compensation axis)

Fig. 9.9 (a) Pitch error compensation points on moving axis and compensation value

From the current position of the moving axis, the compensation amount for the corresponding compensation axis is calculated. Compensation for the compensation axis is performed at the timing of the output of pitch error compensation of the moving axis.

In the Fig. 9.9 (a), while the moving axis moves from compensation point a to compensation point b, the compensation amount (β-α)/(b-a) is output for the compensation axis at the output timing of pitch error compensation of the moving axis.

NOTE

1. This function is an optional function.
   To use this function, the both options for "Stored pitch error compensation" and this function are required.
2. The straightness compensation function can be used after a moving axis and its compensation axis have returned to the reference position.

NOTE

3 Straightness compensation data is superposed on stored pitch error compensation data and output. Straightness compensation is performed at pitch error compensation intervals.

4 Straightness compensation does not allow the moving axis to be used as a compensation axis. To implement such compensation, use inclination compensation.

9.10 SIMPLE STRAIGHTNESS COMPENSATION

Straightness compensation can specify 6 combinations of the moving axis and compensation axis.
But, simple straightness compensation can specify 1 combinations of the moving axis and compensation axis. Other specifications are the same as Straightness compensation.
Please refer to subsection “Straightness compensation”.

NOTE

1 This function is an optional function.
To use this function, the both options for "Stored pitch error compensation" and this function are required.

2 Simple straightness compensation function can be used after a moving axis and its compensation axis have returned to the reference position.

3 Simple straightness compensation data is superposed on stored pitch error compensation data and output. Simple straightness compensation is performed at pitch error compensation intervals.

4 Simple straightness compensation does not allow the moving axis to be used as a compensation axis. To implement such compensation, use inclination compensation.

9.11 INTERPOLATION TYPE STRAIGHTNESS COMPENSATION

In conventional straightness compensation, for each interval between pitch error compensation points set by parameters, the amount of all straightness compensation at the compensation point is output and compensation is performed.
This function equally divides the amount of compensation for each interval between pitch error compensation points for compensation data set using 128 straightness compensation points and outputs it as a compensation pulse.

NOTE

This function is an optional function.
To use this function, the both options for "Stored pitch error compensation" and this function are required.

10 ELECTRONIC GEAR BOX

Chapter 10, "ELECTRONIC GEAR BOX", consists of the following sections:

10.1 ELECTRONIC GEAR BOX (M SERIES)

M

This function enables fabrication of high-precision gears, screws, and other components by rotating the workpiece in synchronization with a rotating tool or by moving the tool in synchronization with a rotating workpiece. The rate of synchronization can be specified with a program. The synchronization of tool and workpiece axes with this function adopts a system in which the synchronization is directly controlled by digital servo, so that the workpiece axis can follow up the speed fluctuations on the tool axis with no error, thereby allowing fabrication of high-precision cogwheels. In the subsequent explanation, the Electronic Gear Box is called the EGB.

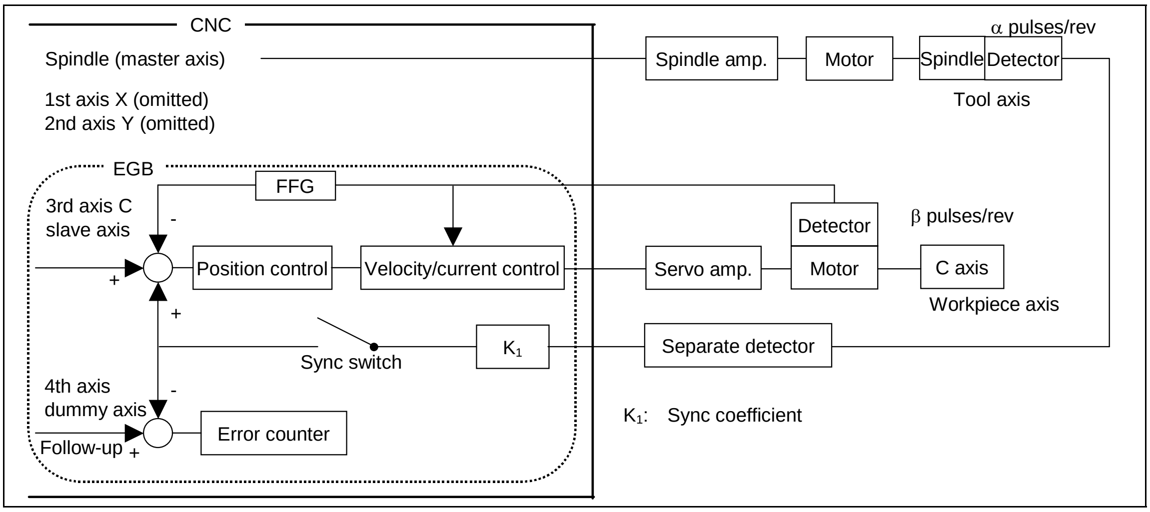


Fig. 10.1 (a)

Format

G81 T_ L_ (Q_ P_) ; Starts synchronization.

T : Number of teeth
L : Number of hob threads
Q : Module or diametral pitch
Specify a module in the case of metric input.
Specify a diametral pitch in the case of inch input.
P : Gear helix angle

G80 ; Ends synchronization.

- Synchronization coefficient

$$\text{Synchronization coefficient} = K_1 = \frac{L}{T} \times \frac{\beta}{\alpha}$$

L : Number of hob threads
T : Number of teeth
α : Number of pulses of the position detector per rotation about the EGB master axis (tool axis)
β : Number of pulses of the position detector per rotation about the EGB slave axis (workpiece axis)

10.2 SKIP FUNCTION FOR EGB AXIS (M SERIES)

M

This function enables the skip or high-speed skip signal (these signals are collectively called skip signals in the remainder of this manual) for the EBG slave axis in synchronization mode with the EGB (electronic gear box).
This function has features such as the following:

1. If a skip signal is input while an EGB axis skip command block is being executed, this block does not terminate until the specified number of skip signals have been input.
2. If a skip signal is input while an EGB axis skip command block is being executed, the tool remains in synchronous mode and moves, not stopping on the EGB slave axis.
3. The machine coordinates assumed when skip signals are input and the number of input skip signals are stored in specified custom macro variables.

> **NOTE**
> This function is an optional function.
> To use this function, the both options for "Electronic gear box" and this function are required.

Format

G31.8 G91 α0 P_ Q_ (R_) ; EGB skip command

α : EGB slave axis.
P : Number of the first one of the custom macro variables used to store the machine coordinates assumed when skip signals are input.
Q : Number of skip signals that can be input during the execution of G31.8
R : Number of the custom macro variable used to store the number of input skip signals.

10.3 ELECTRONIC GEAR BOX AUTOMATIC PHASE SYNCHRONIZATION (M SERIES)

M

In the electronic gear box (EGB), when the start or cancellation of synchronization is specified, the synchronizing state is changed gradually by applying acceleration/deceleration. This is because if synchronization is started or canceled immediately, a shock applies to the machine. Therefore, synchronization can be started or canceled while the spindle is rotating. Also, synchronization ratio can be changed while the spindle is rotating.

In addition, automatic phase synchronization is performed in such a way that, at the start of synchronization, the position of the machine coordinate origin for the workpiece axis matches the spindle position determined by the one-rotation signal. With this synchronization, the same operation is performed as synchronization start caused by a one-rotation signal in hobbing synchronization when using the functions of a hobbing machine.
The spindle corresponds to the EGB master axis and the workpiece axis corresponds to an EGB slave axis (servo axis or Cs contouring axis).

NOTE
This function is an optional function.
To use this function, the both options for "Electronic gear box" and this function are required.

Format

- Acceleration/deceleration type

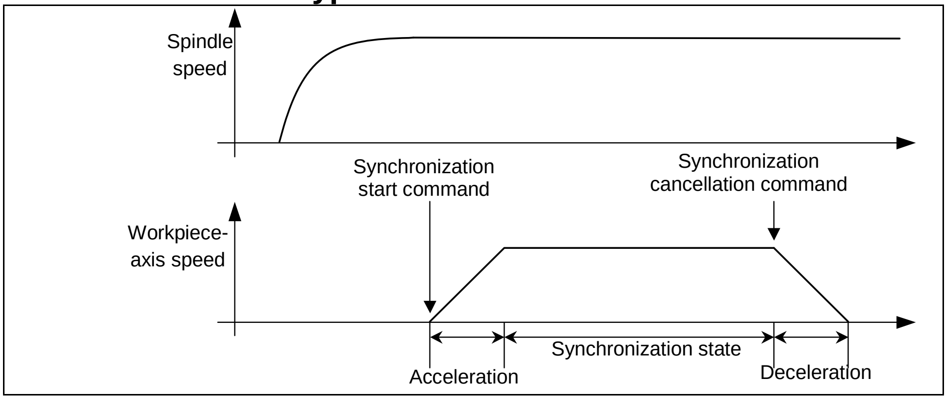


Fig. 10.3 (a)

G81 T _ L _ R1 ; Synchronization start
T : Number of teeth
L : Number of hob threads
G80 R1 ; Synchronization cancellation

- Acceleration/deceleration plus automatic phase synchronization type

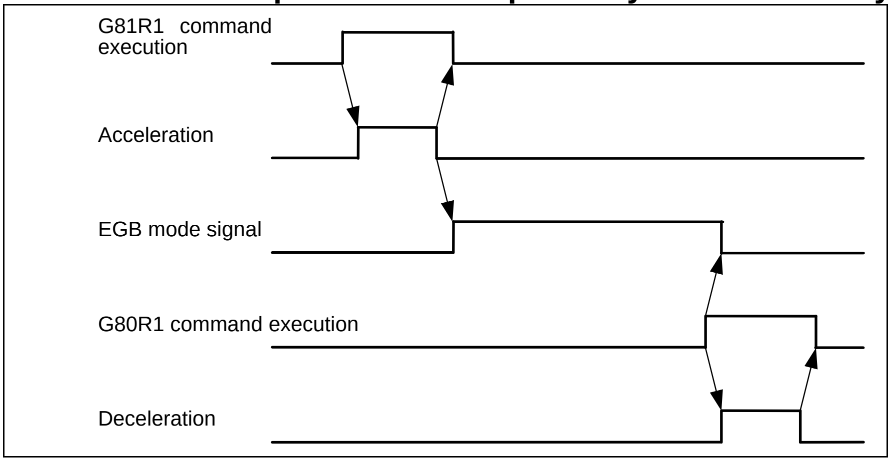


Fig. 10.3 (b)

G81 T _ L _ R2 ; Synchronization start
T : Number of teeth
L : Number of hob threads
G80 R1 ; Synchronization cancellation

10.4 SIGNAL-BASED SERVO EGB SYNCHRONOUS CONTROL

This function can use input signals to make the spindle (master axis) synchronize with the servo motor (slave axis). It is possible to make the servo motor synchronize with the spindle without using programmed commands. An example of using this function might be rotary guide bushing control between the servo motor and spindle.

Because the electronic gear box (EGB), which uses digital servo for direct control, is used as a method for synchronization between the master and slave axes, the slave axis can follow the speed change of the master axis, thus realizing high-precision machining.
Using bit 0 (SVE) of parameter No. 7786 can select whether to enable/disable this function.
The PMC input signal is used to turn on/off the EGB synchronous mode.

NOTE
Using this function does not requires the electronic gear box option. When the option of the electronic gear box is effective, this function cannot be used.

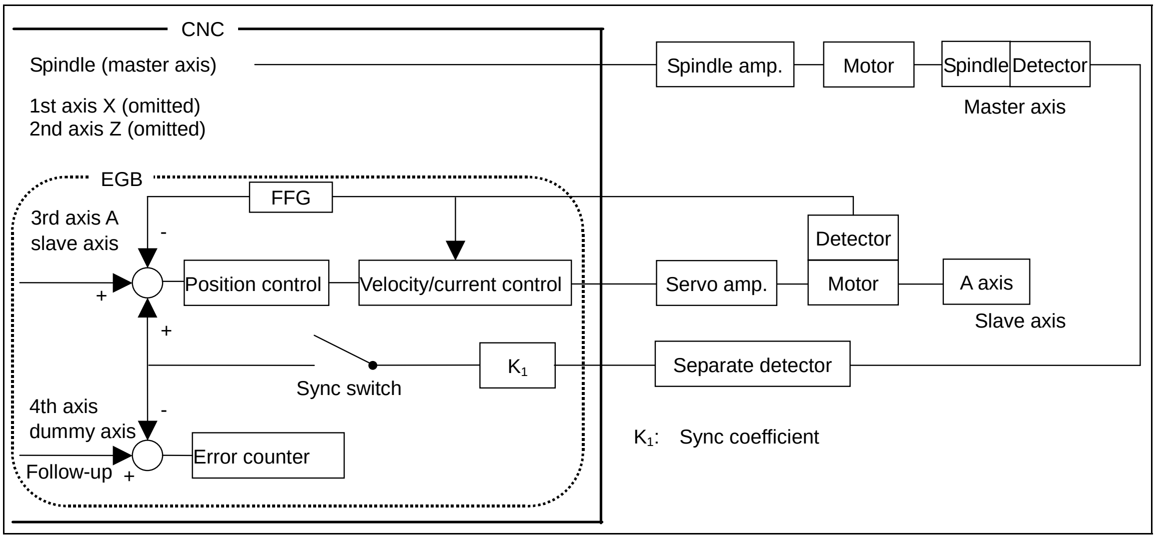


Fig. 10.4 (a)

10.5 ELECTRONIC GEAR BOX (FSSB TYPE) (M SERIES)

M

By the FSSB communication between a CNC and a spindle amplifier, rotational position information of a spindle can be transmitted from a spindle amplifier to a servo control on CNC. Electronic Gear Box (FSSB type) is a new function that a servo axis follows a spindle axis by utilizing this FSSB communication (the electronic gear box is hereinafter called the EGB). And it is a feature of this function that the separate detector interface unit becomes unnecessary compared with the ordinary function.

10.6 FLEXIBLE SYNCHRONIZATION CONTROL

10.6.1 Flexible Synchronization Control

This function is provided for those machines like hobbing machines that require the synchronization of various multiple gear ratios.

Synchronization with this function enables up to four pairs to be operated independently and simultaneously. This achieves special functions for hobbing machines such as the synchronization of the hobbing axis and a single workpiece axis, Z - C synchronization in helical gear cutting, and Y- C synchronization in a hobbing axis shift.

Specifications for flexible synchronization control are as follows:

1) A master axis number, a slave axis number, and a gear ratio are set in parameters.
2) There can be up to four groups to these parameters. Synchronization of the four groups can be executed at the same time.
3) A single slave axis can be specified for multiple master axes.
4) Synchronization is started and canceled with DI signals from the PMC.
   If DI signals are to be switched during automatic operation, this needs to be performed with an M code set in a parameter.
5) Two Cs axes can be used as master and slave axes.

- Block diagram

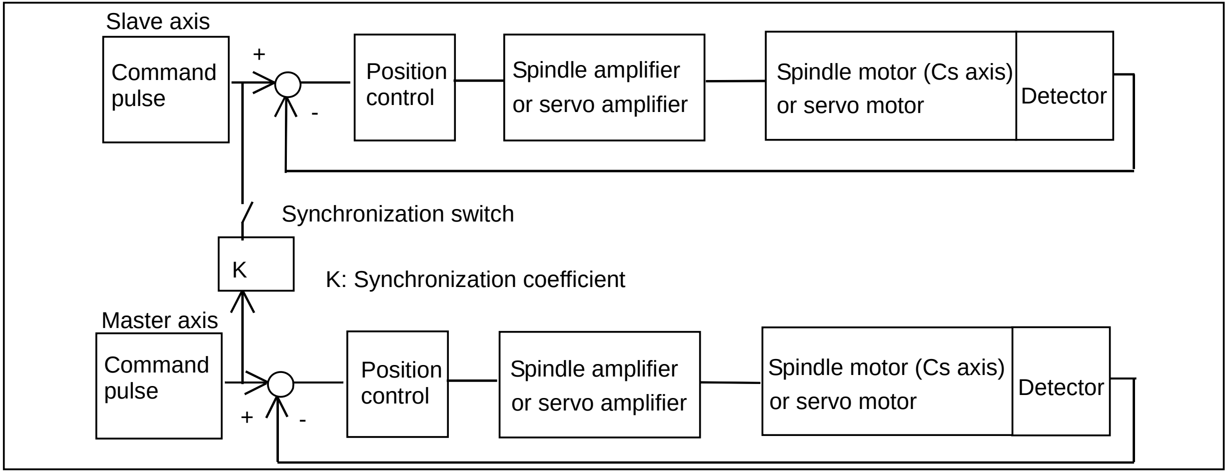


Fig.10.6.1 (a)

NOTE

This function is an optional function.

10.6.2 Automatic Phase Synchronization for Flexible Synchronization Control

Overview

This function applies acceleration/deceleration when the start or cancellation of synchronization is specified in flexible synchronization control.

This acceleration/deceleration allows synchronization to be started or canceled while the tool is moving along the master axis.

This function can also execute automatic phase synchronization so that the slave axis machine coordinate position at the start of synchronization matches the machine coordinate system zero point of the master axis (the machine coordinate is 0).

Notes

NOTE

1 This function is an optional function.
To use this function, the both options for "Flexible synchronization control" and this function are required.

2 The next block is not executed until acceleration/deceleration at the start or cancellation of synchronization is completed during automatic operation.

3 Due to an error produced when the output pulses for the slave axis are calculated, the phase of the slave axis may not be matched by least input increment. This error is not accumulated.

4 This function is disabled in the following functions:
  - AI advanced preview control (M Series)
  - AI contour control I
  - AI contour control II

10.6.3 Inter-path Flexible Synchronization Control

Overview

Inter-path flexible synchronization control enables flexible synchronization control between axes in different paths in a multi-path system.

Up to four slave axes can be specified in one path.

An axis in another path can be specified as the master axis of each slave axis.

Synchronization for all synchronization pairs in all paths can be executed simultaneously.

Example)

In a multi-path system with the following axis configuration, not only synchronization between the C1 axis in path 1 (master axis) and the A1 axis in path 1 (slave axis), but also synchronization between the C1 axis in path 1 (master axis) and the A2 axis in path 2 (slave axis) can be performed.

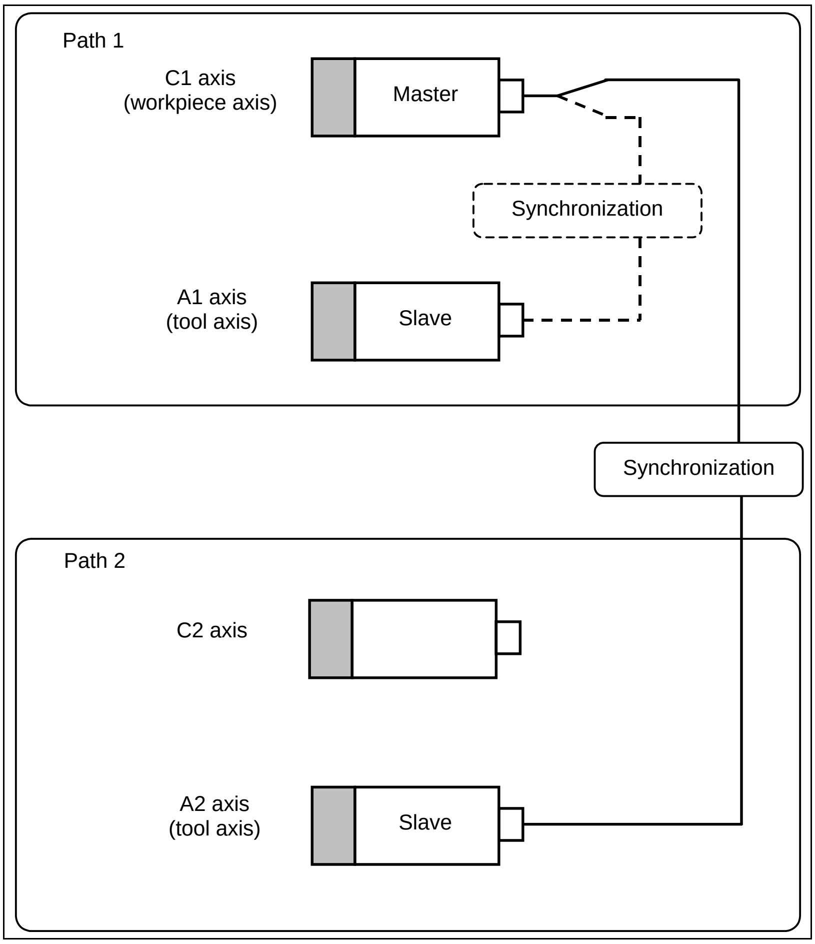


Fig.10.6.3 (a)

Restrictions

The following functions cannot be specified in the inter-path flexible synchronization mode.
If any of these functions is specified in the inter-path flexible synchronization mode, alarm is issued.

- Reference return in Cs contouring control (G00, G28)
- Skip function (G31)
- Automatic tool length measurement/Automatic tool offset function
- Automatic reference return operation of low-speed type (G28)
- High-speed program check function

These functions can be specified when flexible synchronization control and the inter-path flexible synchronization mode are turned off.

NOTE

This function is an optional function.
To use this function, the both options for "Flexible synchronization control" and this function are required.

10.6.4 Skip Function for Flexible Synchronization Control

This function enables the skip or high-speed skip signal (in the following explanation, these signals are collectively called skip signal) for the slave axis that is moved by command of the master axis in the flexible synchronization control mode.
This function has features such as the following:

- If a skip signal is input while a skip command for flexible synchronization control block is being executed, this block does not terminate until the specified number of skip signals have been input.
- The machine coordinates assumed when skip signals are input and the number of input skip signals are stored in specified custom macro variables.
- The total number of the skip signal inputs is stored in another specified custom macro variable.

NOTE

This function is an optional function.
To use this function, the both options for "Flexible synchronization control" and this function are required.

10.6.5 Hob Command by Flexible Synchronization Control

Overview

A command compatible with that for a hobbing machine can be used as a synchronization command of flexible synchronization control.

NOTE

This function is an optional function.
To use this function, the both options for "Flexible synchronization control" and this function are required.

Format

G81 T_ L_ (I_) (Q_ P_) ; **Starts synchronization.**

T : Number of teeth
L : Number of hob threads
Q : Module or diametral pitch
Specify a module in the case of metric input.
Specify a diametral pitch in the case of inch input.
P : Gear helix angle

G80 ; **Ends synchronization.**

- Synchronization coefficient

$$\text{Synchronization coefficient} = K_1 = \frac{L}{T} \times \frac{\beta}{\alpha}$$

L : Number of hob threads
T : Number of teeth
α : Number of pulses of the position detector per rotation about a master axis (Parameter setting)
β : Number of pulses of the position detector per rotation about a slave axis (Parameter setting)

10.7 U-AXIS CONTROL (M SERIES)

M

Conventionally, the control of an axis on a spindle, such as the U-axis of a vertical lathe, from a motor mounted in a location other than the spindle has required a mechanism, consisting of a planetary gear box and differential gears, to prevent the axis from moving as the spindle is rotating.

The U-axis control function enables the U-axis to remain in a fixed position or to move at a programmed speed without using a mechanism such as a planetary gear box. This is done by causing the U-axis motor to rotate in such a way that U-axis movement, which would be caused by the rotation of the spindle, is canceled out.

An electronic gear box (EGB) is used to cause the U-axis motor to rotate in sync with the spindle. (EGB option is necessary.)
With the EGB, the servo CPU processes signals, received from the position coder mounted on the spindle, at high speed to control U-axis movement. It is capable of high-precision synchronous control. The EGB requires axis control circuits for two axes (the U-axis and U'-axis). It acquires the pulses necessary for synchronization from the separate detector feedback connector on the U'-axis side. The spindle, which is a synchronous standard, is called master axis. The U-axis, which moves synchronized with the master axis, is called slave axis. U'-axis, which acquires the pulses necessary for synchronization, is called dummy axis.

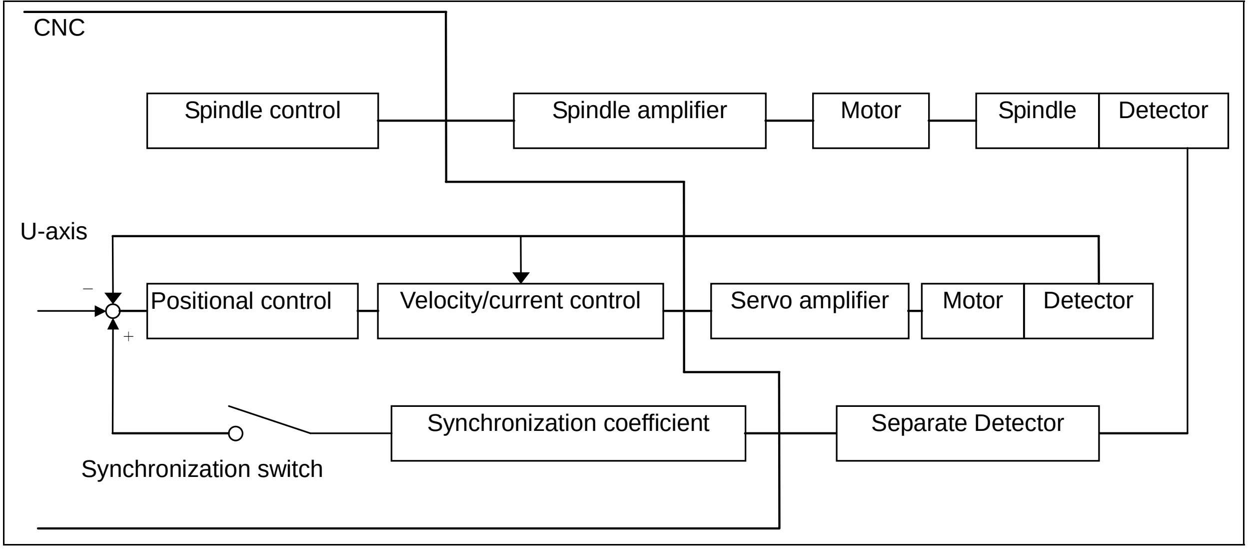


Fig. 10.7 (a) Block diagram of U-axis control

NOTE
This function is included in the option "Electronic gear box".

11 EDITING OPERATION

Chapter 11, "EDITING OPERATION", consists of the following sections:

11.1 PART PROGRAM STORAGE SIZE / NUMBER OF REGISTERABLE PROGRAMS

Table 11.1 (a) lists the combinations of the part program storage sizes and the number of registrable programs.

Table 11.1 (a) Combinations of the part program storage sizes and the number of registrable programs

Part program storage size	**Number of registerable programs**			
	Number of Registered programs expansion 1 : Non		Number of Registered programs expansion 1 : Yes	
	1-path system	2-path system	1-path system	2-path system
512Kbyte	400	-	1000	-
1Mbyte	-	800	-	1000
2Mbyte	400	800	1000	1000

NOTE

Part program storage size 2Mbyte and Number of Registered programs expansion 1 are optional.

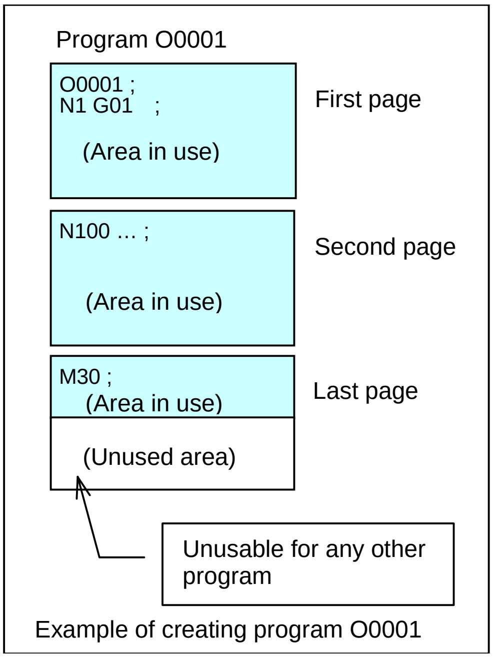


Fig. 11.1 (a) Example when the program O0001 is created

NOTE
1. Creating one folder results in the number of programs yet to be registerable decreasing one.
2. The program storage size means the maximum size of a program if the program is the one and only program registered.
3. If more than one program is registered, the total size of registerable programs reduces for the following reason.
   FS0*i*-F manage programs in page units. The unit of program storage is managed also in page units. When a program is created, as many pages as necessary to store the program are secured, and the program is stored on these pages. Generally, the last program storage page has an unused area (Fig. 11.1 (a)). This unused area cannot be used to store any other program. For the sake of program management, it is regarded as an area in use.
   FS0*i*-C uses a similar way of management, but the unit of pages in it differs from that in FS0*i*-F. So, if more than one program is registered in FS0*i*-F, the total program size of registerable programs in FS0*i*-F differs from that in FS0*i*-C.

11.2 PROGRAM EDITING

Following program editing operations are possible.

1. Creating new programs
2. Deleting programs
   - Deleting one program
   - Deleting specified programs at a time
   - Deleting all programs in a folder at a time
3. Copying and moving programs
   Copying and moving programs from a folder to folder
4. Editing a word
   - Inserting a word or block
   - Overwriting a word
   - Replacing a word or address

- Deleting a word or block
- Copying, deleting, and pasting a specified range

5 Editing characters
- Inserting characters
- Overwriting characters
- Replacing a character string
- Deleting characters
- Copying, deleting, and pasting a specified range

6 Search
- Searching for a program
- Searching for a word or address
- Repeating search

7 Rearranging program memory contents
An NC program on program memory is first stored in a continuous storage area. Editing the NC program divides the NC program storage area into several separate blocks, and as editing is repeated, these blocks are further divided into smaller blocks. When the NC program is divided into small blocks, the memory operation processing speed may decrease, or the program memory space used may increase excessively. So, program memory must be rearranged to store the program in a continuous area again.

11.3 PROGRAM PROTECT

The following functions are provided to protect programs:

1 Protection by edit/display disable attribute
This function sets an edit or display disable attribute for programs and folders to disable editing or display of the contents of the programs and folders. When display of a program is disabled, external output of the program is also disabled.

2 Protection by parameter
By using a password parameter and a keyword parameter, this function provides a lock for a parameter that protects programs with program Nos. O9000 to O9999, and programs and folders with the edit or display disable attribute.

When locked (the password ≠ the keyword), the parameter for protection cannot be changed. In this case, program protection for programs with program Nos. O9000 to O9999, and programs and folders with the edit or display disable attribute cannot be released unless the keyword is set correctly.
The values set in both the password and keyword parameters are not displayed. The lock is released by setting the keyword parameter to the same value as the value set in advance in the password parameter. When 0 is displayed in the password parameter, the password parameter is not yet set.

3 Key and program encryption
See "KEY AND PROGRAM ENCRYPTION" in the next section.

11.4 KEY AND PROGRAM ENCRYPTION

Program contents can be protected by setting parameters for encryption and for the program security range.

1 When the password and security range parameters are specified, the display, editing, and input/output operations are disabled for the programs within the security range.

2 The programs in the security range can be encrypted before being output. Once encrypted, the programs cannot be decrypted. In addition, encrypted programs can be input directly.

NOTE
This function is an optional function.

11.5 EXTENDED PART PROGRAM EDITING

The following operations, which were enabled by the extended part program editing function before, can be performed without the optional function now:

- Replacement of words and addresses
- Cut and paste of words and characters
- Copy and move of programs

11.6 PLAYBACK

In the TEACH IN JOG or TEACH IN HANDLE mode, you can create a program while inserting the coordinate of the current position along each axis in the absolute coordinate system when the tool is moved by manual operation into the program.
You can input the words other than axis names in the same way as in the EDIT mode.

11.7 MACHINING TIME STAMP

The execution times of the most recently executed ten programs can be displayed in hours, minutes, and seconds.
The calculated machining time can be inserted as a comment of the program to check the machining time on the program directory screen.

NOTE
This function is an optional function.

11.8 BACKGROUND EDITING

While a program is being executed, another program can be edited. Such an edit operation is called background editing (BG editing). Background editing can be performed in exactly the same way as ordinary editing (foreground editing).

11.9 MEMORY CARD PROGRAM OPERATION/EDITING

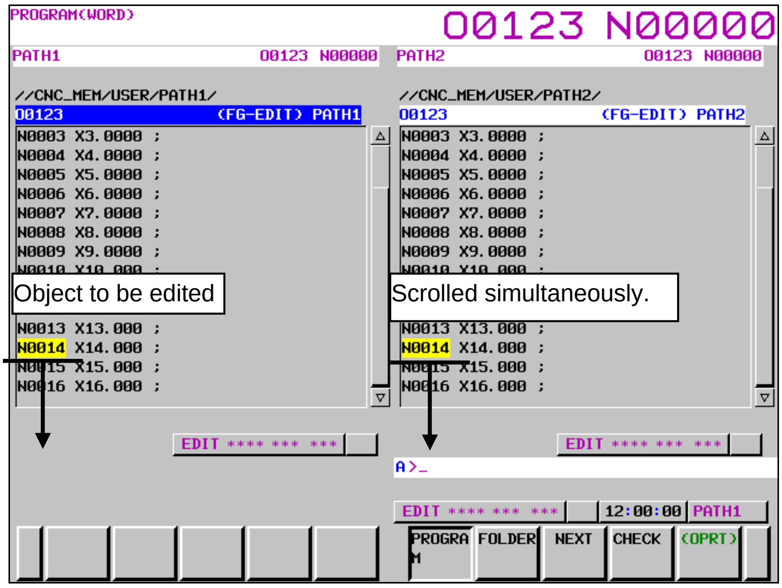
A program in a program storage file (named "FANUCPRG.BIN") on the memory card can be selected as a main program to perform memory operation.
Moreover, the contents of a program storage file can be referenced using the program directory screen, and a program in a program storage file can be edited using the program editing screen.

A program storage file can be created using a memory card program tool on a personal computer. To use a created program storage file, the file is written to a memory card prepared in the FAT format.

The number of program storage file entries (total number of program files and folders) on the memory card can be selected from the following:
- 63, 500, 1000
To use, a program storage file that allows more than 63 entries to be created, the option for "memory card program entry count extension" is required.

11.10 MULTI-PATH EDITING FUNCTION

When the program of the path to be edited is scrolled in the simultaneous editing of multi-path programs screen, other path programs that are displayed on the same screen can be scrolled simultaneously.



```
N0003 X3.0000 ;
N0004 X4.0000 ;
N0005 X5.0000 ;
N0006 X6.0000 ;
N0007 X7.0000 ;
N0008 X8.0000 ;
N0009 X9.0000 ;
N0010 X10.000 ;
N0013 X13.000 ;
N0014 X14.000 ;
N0015 X15.000 ;
N0016 X16.000 ;
```

Object to be edited

Scrolled simultaneously.

Fig. 11.10 (a) Synchronous scroll function

This function provides the simultaneous scroll mode in which all programs being edited simultaneously are scrolled and the single scroll mode in which only the program to be edited is scrolled. It is possible to switch between these modes easily through soft key operation.

In the simultaneous scroll mode, when a waiting M-code comes due to a scroll, waiting is performed by stopping the scroll of the path until the arrival of the same waiting M-code of other paths. It is, therefore, possible to edit the program while confirming the waiting of each path.
Moreover, the waiting search function enables the cursor positions of all paths being edited simultaneously to be moved to specified waiting M-code simultaneously.

NOTE
This function is an optional function.

12 SETTING AND DISPLAY

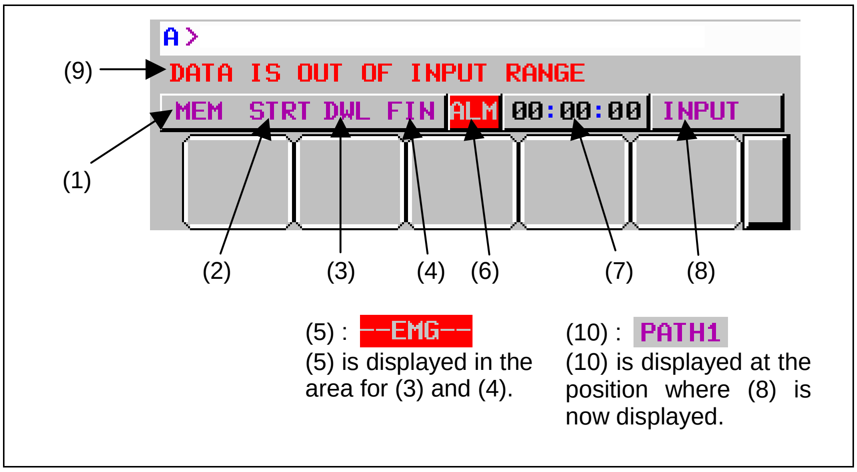
In the subsequent explanation in this chapter, the screen descriptions are provided based on a 10.4" LCD of the 12-soft-key type.

Chapter 12, "SETTING AND DISPLAY", consists of the following sections:

12.1 STATUS DISPLAY

The current mode, automatic operation state, alarm state, and program editing state are displayed on the bottom right on the screen allowing the operator to readily understand the operation condition of the system.
If data setting or the input/output operation is incorrect, the CNC does not accept the operation and a warning message is displayed. This prevents invalid data setting and input/output errors.

Explanation



```
A>
DATA IS OUT OF INPUT RANGE
MEM STRT DWL FIN ALM 00:00:00 INPUT
```

(5) is displayed in the area for (3) and (4).

(10) is displayed at the position where (8) is now displayed.

Fig. 12.1 (a) Positions of status display items

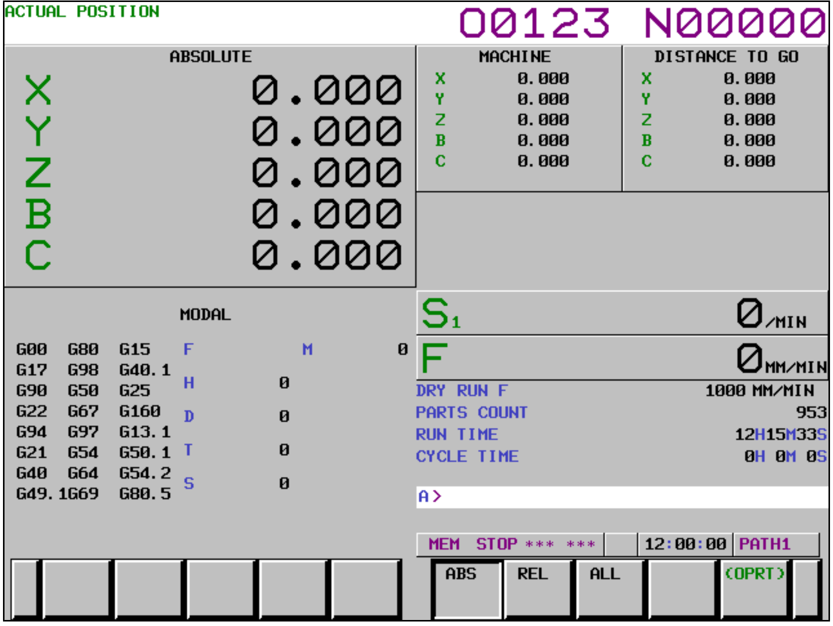
(1) Current mode
(2) Automatic operation status
(3) Axis moving status/dwell status
(4) State in which an auxiliary function is being executed
(5) Emergency stop or reset status
(6) Alarm status
(7) Current time
(8) Program editing status
(9) Warning for data setting or input/output operation
(10) Path name

12.2 CLOCK FUNCTION

Time is displayed in the hour/minute/second format on each display screen. Some screens allow display of the year, month, and day.
The custom macro system variable can be used to read the time. On PMC, a window command can be used to know the current time.

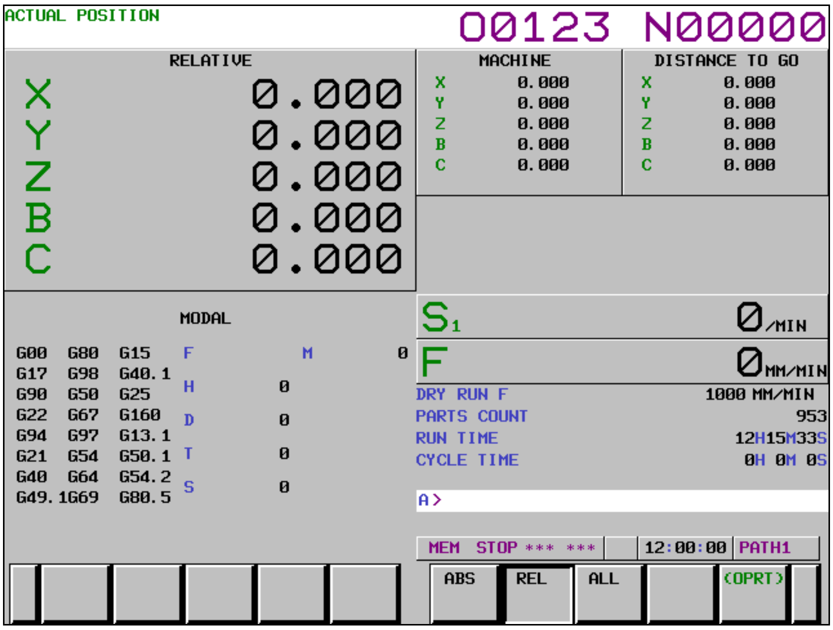
12.3 CURRENT POSITION DISPLAY

The current position in the relative, workpiece, and machine coordinate systems and the remaining distance are displayed.



```
ACTUAL POSITION                              O0123 N00000
            ABSOLUTE              MACHINE         DISTANCE TO GO
X           0.000            X    0.000       X    0.000
Y           0.000            Y    0.000       Y    0.000
Z           0.000            Z    0.000       Z    0.000
B           0.000            B    0.000       B    0.000
C           0.000            C    0.000       C    0.000

          MODAL                S1                        0/MIN
G00  G80  G15   F       M    0 F                         0MM/MIN
G17  G98  G40.1 H       0      DRY RUN F          1000 MM/MIN
G90  G50  G25   D       0      PARTS COUNT                953
G22  G67  G160  T       0      RUN TIME              12H15M33S
G94  G97  G13.1 S       0      CYCLE TIME               0H 0M 0S
G21  G54  G50.1                A>
G40  G64  G54.2
G49.1G69  G80.5                MEM  STOP *** ***  12:00:00  PATH1
```

Fig. 12.3 (a) Current position (absolute coordinate) screen

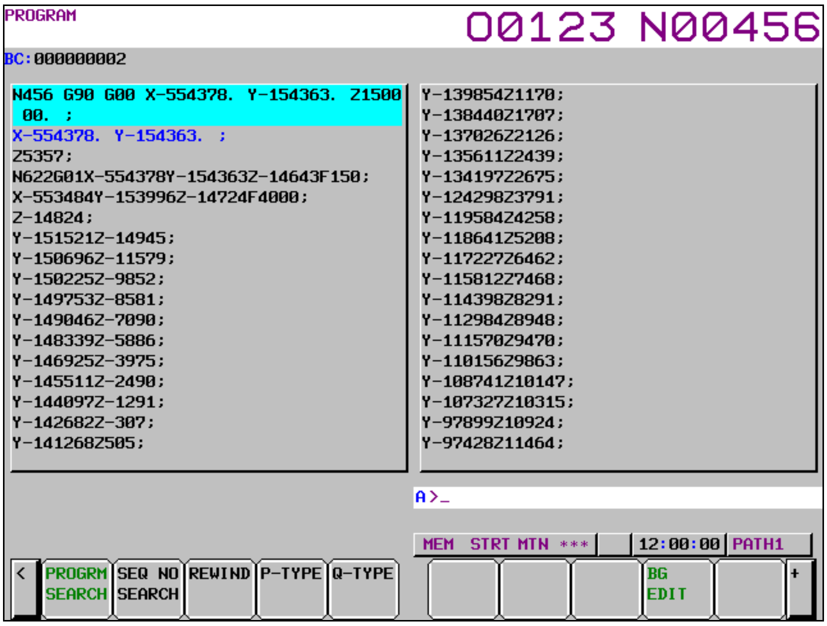


```
ACTUAL POSITION                              O0123 N00000
            RELATIVE              MACHINE         DISTANCE TO GO
X           0.000            X    0.000       X    0.000
Y           0.000            Y    0.000       Y    0.000
Z           0.000            Z    0.000       Z    0.000
B           0.000            B    0.000       B    0.000
C           0.000            C    0.000       C    0.000

          MODAL                S1                        0/MIN
G00  G80  G15   F       M    0 F                         0MM/MIN
G17  G98  G40.1 H       0      DRY RUN F          1000 MM/MIN
G90  G50  G25   D       0      PARTS COUNT                953
G22  G67  G160  T       0      RUN TIME              12H15M33S
G94  G97  G13.1 S       0      CYCLE TIME               0H 0M 0S
G21  G54  G50.1                A>
G40  G64  G54.2
G49.1G69  G80.5                MEM  STOP *** ***  12:00:00  PATH1
```

Fig. 12.3 (b) Current position (relative coordinate) screen

12.4 PROGRAM DISPLAY

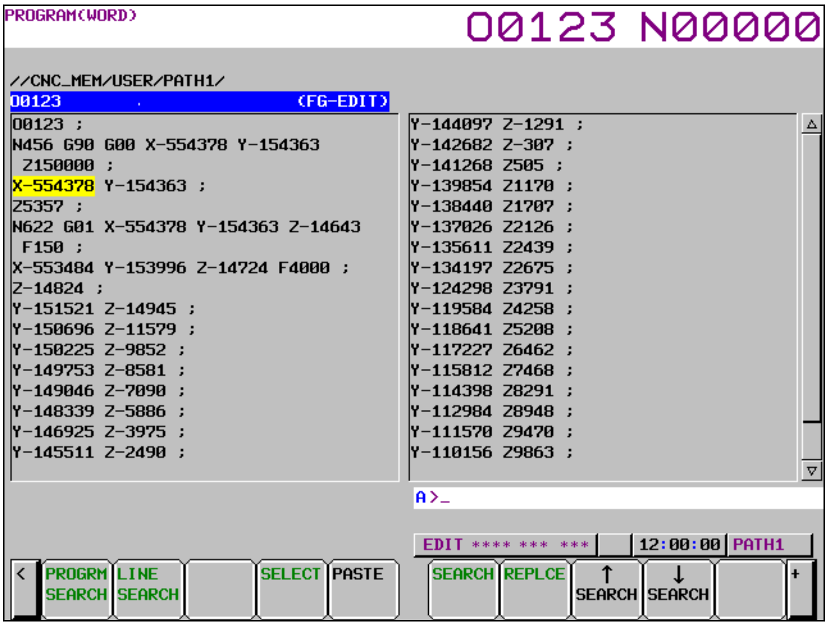
The contents of a program being edited or executed are displayed.
In addition, a list of the file names of programs registered in program memory is displayed. The list screen displays the used memory size, the remaining memory size, and the size and date of update for each program are indicated.



```
N456 G90 G00 X-554378. Y-154363. Z1500
 00. ;
X-554378. Y-154363. ;
Z5357;
N622G01X-554378Y-154363Z-14643F150;
X-553484Y-153996Z-14724F4000;
Z-14824;
Y-151521Z-14945;
Y-150696Z-11579;
Y-150225Z-9852;
Y-149753Z-8581;
Y-149046Z-7090;
Y-148339Z-5886;
Y-146925Z-3975;
Y-145511Z-2490;
Y-144097Z-1291;
Y-142682Z-307;
Y-141268Z505;
```

```
Y-139854Z1170;
Y-138440Z1707;
Y-137026Z2126;
Y-135611Z2439;
Y-134197Z2675;
Y-124298Z3791;
Y-119584Z4258;
Y-118641Z5208;
Y-117227Z6462;
Y-115812Z7468;
Y-114398Z8291;
Y-112984Z8948;
Y-111570Z9470;
Y-110156Z9863;
Y-108741Z10147;
Y-107327Z10315;
Y-97899Z10924;
Y-97428Z11464;
```

Fig. 12.4 (a) Screen displaying a program being executed

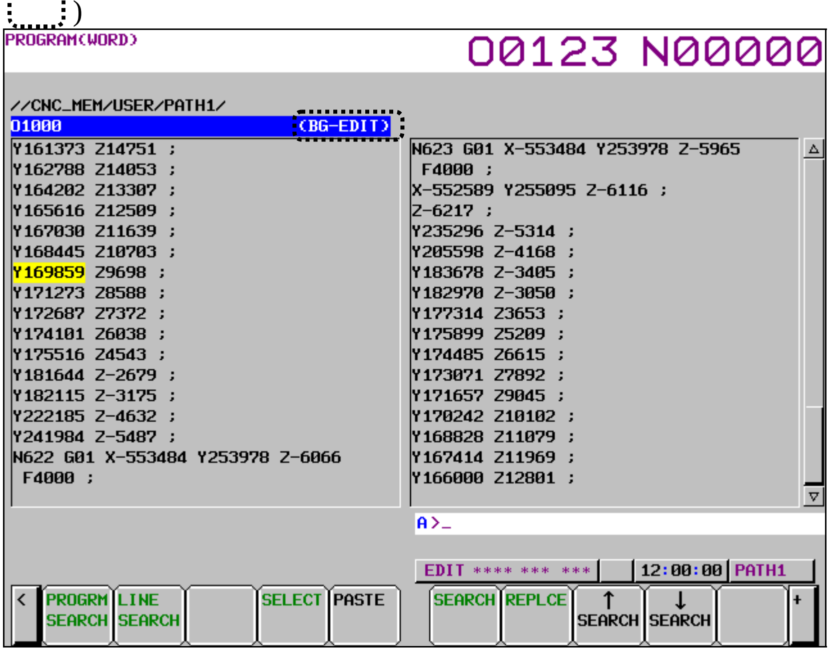


```
O0123 ;
N456 G90 G00 X-554378 Y-154363
 Z150000 ;
X-554378 Y-154363 ;
Z5357 ;
N622 G01 X-554378 Y-154363 Z-14643
 F150 ;
X-553484 Y-153996 Z-14724 F4000 ;
Z-14824 ;
Y-151521 Z-14945 ;
Y-150696 Z-11579 ;
Y-150225 Z-9852 ;
Y-149753 Z-8581 ;
Y-149046 Z-7090 ;
Y-148339 Z-5886 ;
Y-146925 Z-3975 ;
Y-145511 Z-2490 ;
```

```
Y-144097 Z-1291 ;
Y-142682 Z-307 ;
Y-141268 Z505 ;
Y-139854 Z1170 ;
Y-138440 Z1707 ;
Y-137026 Z2126 ;
Y-135611 Z2439 ;
Y-134197 Z2675 ;
Y-124298 Z3791 ;
Y-119584 Z4258 ;
Y-118641 Z5208 ;
Y-117227 Z6462 ;
Y-115812 Z7468 ;
Y-114398 Z8291 ;
Y-112984 Z8948 ;
Y-111570 Z9470 ;
Y-110156 Z9863 ;
```

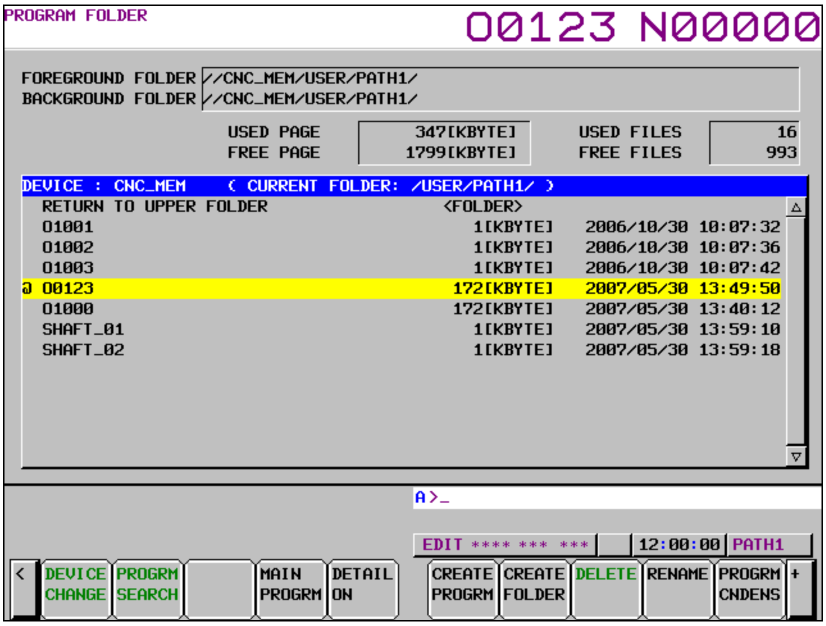
Fig. 12.4 (b) Screen displaying a program being edited

During background editing, "(BG-EDIT)" is displayed at the right of the program name. (Part enclosed by a dotted rectangle ⬚)



```
PROGRAM(WORD)                                    O0123 N00000

//CNC_MEM/USER/PATH1/
O1000                        (BG-EDIT)
Y161373 Z14751 ;                        N623 G01 X-553484 Y253978 Z-5965
Y162788 Z14053 ;                         F4000 ;
Y164202 Z13307 ;                        X-552589 Y255095 Z-6116 ;
Y165616 Z12509 ;                        Z-6217 ;
Y167030 Z11639 ;                        Y235296 Z-5314 ;
Y168445 Z10703 ;                        Y205598 Z-4168 ;
Y169859 Z9698 ;                         Y183678 Z-3405 ;
Y171273 Z8588 ;                         Y182970 Z-3050 ;
Y172687 Z7372 ;                         Y177314 Z3653 ;
Y174101 Z6038 ;                         Y175899 Z5209 ;
Y175516 Z4543 ;                         Y174485 Z6615 ;
Y181644 Z-2679 ;                        Y173071 Z7892 ;
Y182115 Z-3175 ;                        Y171657 Z9045 ;
Y222185 Z-4632 ;                        Y170242 Z10102 ;
Y241984 Z-5487 ;                        Y168828 Z11079 ;
N622 G01 X-553484 Y253978 Z-6066        Y167414 Z11969 ;
 F4000 ;                                Y166000 Z12801 ;

                                        A>_

                                        EDIT **** *** ***    12:00:00 PATH1
```

Fig. 12.4 (c) Program display screen displayed during background editing

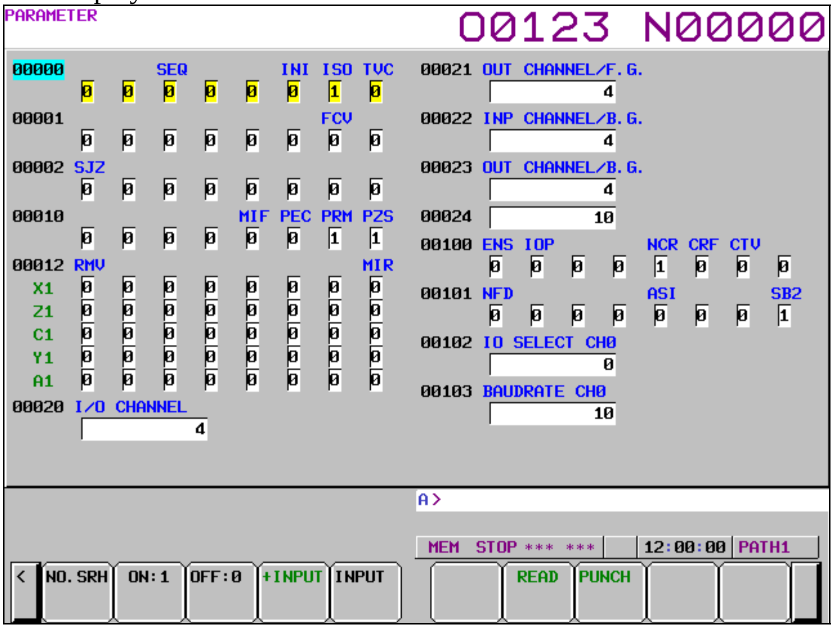


```
PROGRAM FOLDER                                   O0123 N00000

 FOREGROUND FOLDER //CNC_MEM/USER/PATH1/
 BACKGROUND FOLDER //CNC_MEM/USER/PATH1/

                USED PAGE     347[KBYTE]     USED FILES      16
                FREE PAGE    1799[KBYTE]     FREE FILES     993

DEVICE : CNC_MEM    ( CURRENT FOLDER: /USER/PATH1/ )
  RETURN TO UPPER FOLDER                <FOLDER>
  O1001                                    1[KBYTE]   2006/10/30 10:07:32
  O1002                                    1[KBYTE]   2006/10/30 10:07:36
  O1003                                    1[KBYTE]   2006/10/30 10:07:42
@ O0123                                  172[KBYTE]   2007/05/30 13:49:50
  O1000                                  172[KBYTE]   2007/05/30 13:40:12
  SHAFT_01                                 1[KBYTE]   2007/05/30 13:59:10
  SHAFT_02                                 1[KBYTE]   2007/05/30 13:59:18

                                        A>_

                                        EDIT **** *** ***    12:00:00 PATH1
```

Fig. 12.4 (d) Program folder screen

12.5 PARAMETER SETTING AND DISPLAY

Parameter settings are displayed.

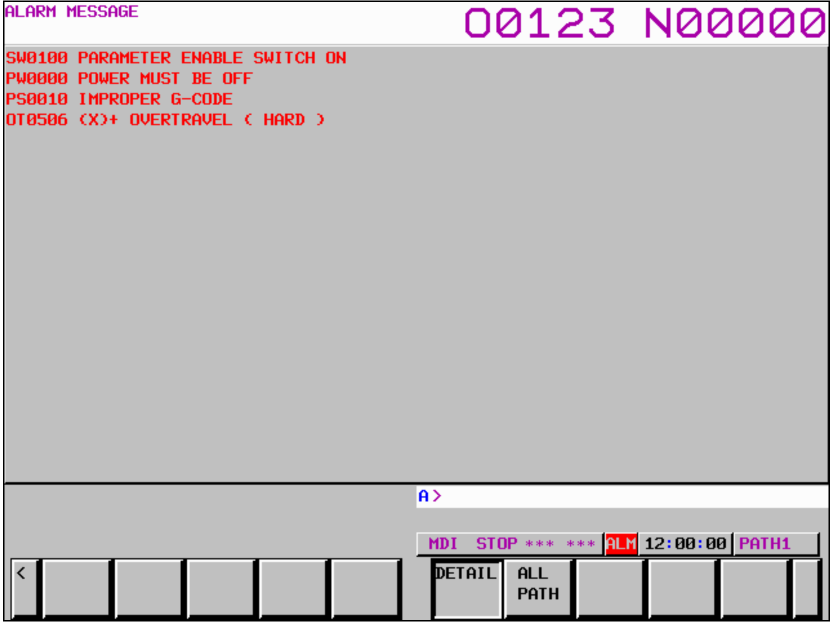


```
PARAMETER                                   O0123 N00000

00000           SEQ           INI ISO TVC    00021 OUT CHANNEL/F.G.
        0   0   0   0   0   0   1   0                          4
00001                             FCV        00022 INP CHANNEL/B.G.
        0   0   0   0   0   0   0   0                          4
00002 SJZ                                    00023 OUT CHANNEL/B.G.
        0   0   0   0   0   0   0   0                          4
00010                     MIF PEC PRM PZS    00024            10
        0   0   0   0   0   0   1   1        00100 ENS IOP        NCR CRF CTV
00012 RMV                             MIR          0   0   0   0   1   0   0   0
   X1   0   0   0   0   0   0   0   0        00101 NFD            ASI        SB2
   Z1   0   0   0   0   0   0   0   0              0   0   0   0   0   0   0   1
   C1   0   0   0   0   0   0   0   0        00102 IO SELECT CH0
   Y1   0   0   0   0   0   0   0   0                              0
   A1   0   0   0   0   0   0   0   0        00103 BAUDRATE CH0
00020 I/O CHANNEL                                                 10
                  4

                                             A>
                                             MEM STOP *** ***  12:00:00 PATH1
```

Fig. 12.5 (a) Parameter screen

12.6 ALARM DISPLAY

The currently issued alarm number and a brief description of the alarm are displayed.



```
ALARM MESSAGE                               O0123 N00000
SW0100 PARAMETER ENABLE SWITCH ON
PW0000 POWER MUST BE OFF
PS0010 IMPROPER G-CODE
OT0506 (X)+ OVERTRAVEL ( HARD )

                                             A>
                                             MDI STOP *** *** ALM 12:00:00 PATH1
```

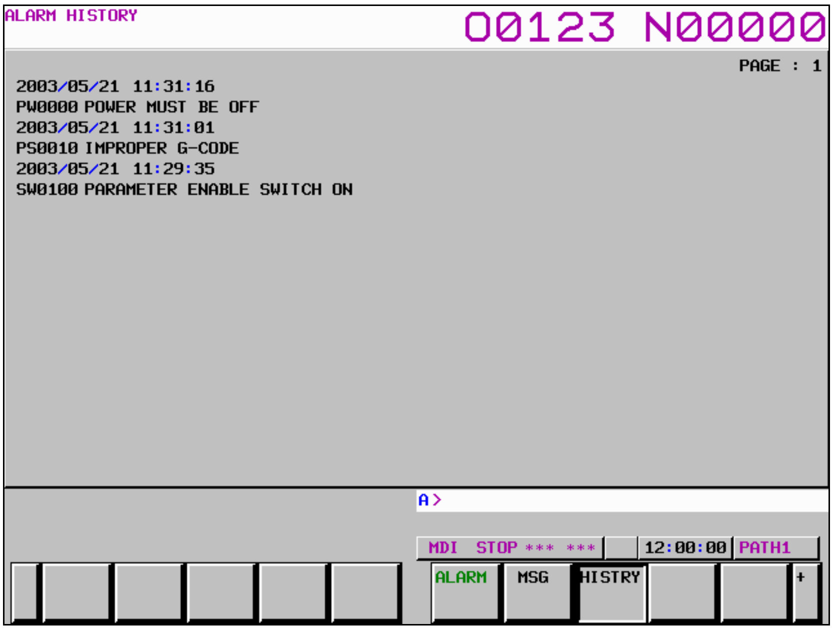
Fig. 12.6 (a) Alarm screen

12.7 ALARM HISTORY DISPLAY

A history of alarms that have been issued in the CNC can be stored. The stored history can be referenced on the alarm history screen.

The following information is displayed for each alarm:

(1) Date and time of alarm issuance
(2) Alarm type
(3) Alarm number
(4) Alarm message (This item may not appear for some alarms.)



```
2003/05/21  11:31:16
PW0000 POWER MUST BE OFF
2003/05/21  11:31:01
PS0010 IMPROPER G-CODE
2003/05/21  11:29:35
SW0100 PARAMETER ENABLE SWITCH ON
```

Fig. 12.7 (a) Alarm history screen

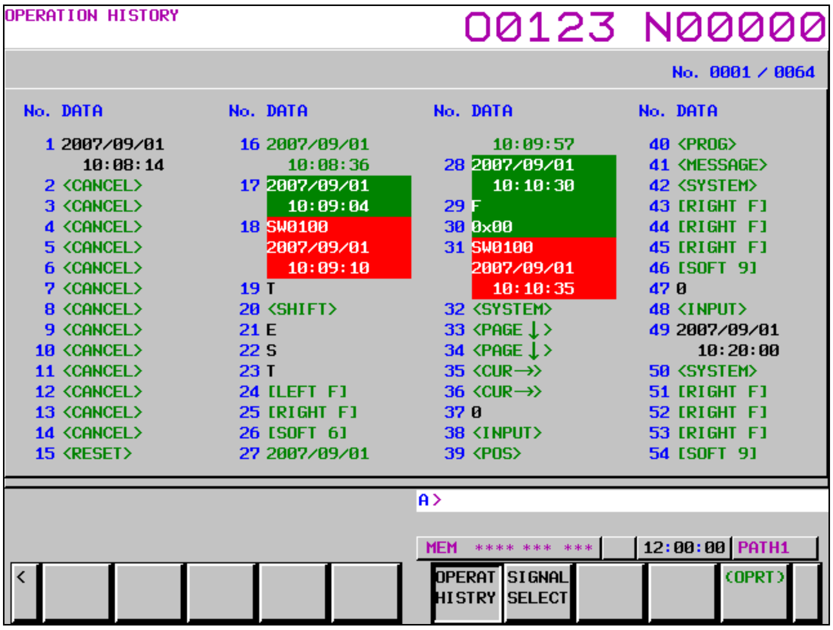
12.8 OPERATOR MESSAGE HISTORY DISPLAY

A history of operator messages can be stored. The stored history can be referenced on the operator message history screen.

12.9 OPERATION HISTORY DISPLAY

A history of events such as operations performed by the operator, alarms issued, and external operator messages can be recorded and referenced, and history data can be output.
The following data is recorded as history data:
(1) MDI key operations performed by the operator
(2) On/off status transition of I/O signals (X, Y, G, and F)
(3) Alarms issued
(4) Time stamp (date and time)
(5) External operator message

Recorded data can be referenced on the operation history screen.
In addition to display of history data, search, erasure, and output to external I/O devices can be performed for history data. While this screen is being displayed, no history data is recorded.

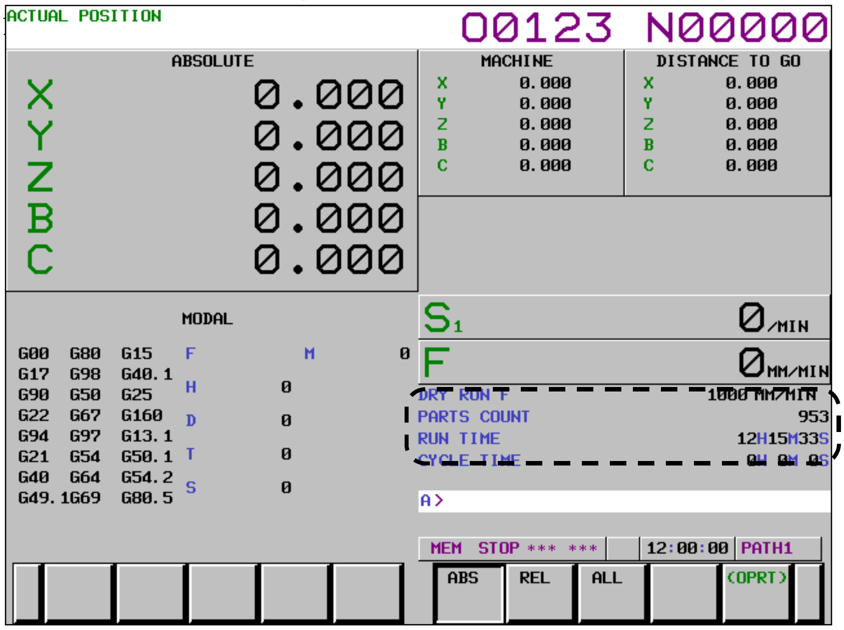


```
OPERATION HISTORY                                   O0123 N00000
                                                      No. 0001 / 0064
No. DATA              No. DATA              No. DATA              No. DATA
  1 2007/09/01         16 2007/09/01              10:09:57         40 <PROG>
       10:08:14             10:08:36        28 2007/09/01          41 <MESSAGE>
  2 <CANCEL>           17 2007/09/01              10:10:30         42 <SYSTEM>
  3 <CANCEL>                10:09:04        29 F                   43 [RIGHT F]
  4 <CANCEL>           18 SW0100            30 0x00                44 [RIGHT F]
  5 <CANCEL>              2007/09/01        31 SW0100              45 [RIGHT F]
  6 <CANCEL>                10:09:10           2007/09/01          46 [SOFT 9]
  7 <CANCEL>           19 T                       10:10:35         47 0
  8 <CANCEL>           20 <SHIFT>           32 <SYSTEM>            48 <INPUT>
  9 <CANCEL>           21 E                 33 <PAGE ↓>            49 2007/09/01
 10 <CANCEL>           22 S                 34 <PAGE ↓>                 10:20:00
 11 <CANCEL>           23 T                 35 <CUR→>              50 <SYSTEM>
 12 <CANCEL>           24 [LEFT F]          36 <CUR→>              51 [RIGHT F]
 13 <CANCEL>           25 [RIGHT F]         37 0                   52 [RIGHT F]
 14 <CANCEL>           26 [SOFT 6]          38 <INPUT>             53 [RIGHT F]
 15 <RESET>            27 2007/09/01        39 <POS>               54 [SOFT 9]

                                    A>
                                    MEM **** *** ***     12:00:00  PATH1
```

Fig. 12.9 (a) Operation history screen

12.10 RUN HOUR AND PARTS COUNT DISPLAY

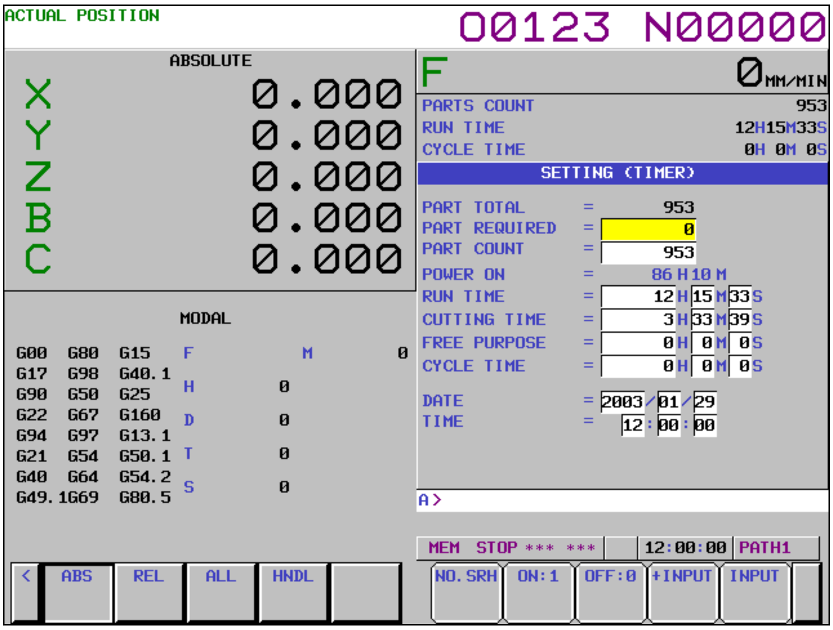
On the current position screen, operation times (run time and cycle time) and the number of machined parts are displayed. (Area indicated by ⬚)



```
ACTUAL POSITION                                     O0123 N00000
           ABSOLUTE                 MACHINE          DISTANCE TO GO
X              0.000            X     0.000          X     0.000
Y              0.000            Y     0.000          Y     0.000
Z              0.000            Z     0.000          Z     0.000
B              0.000            B     0.000          B     0.000
C              0.000            C     0.000          C     0.000

            MODAL                S1                          0/MIN
G00  G80  G15   F          M   0 F                           0MM/MIN
G17  G98  G40.1                  DRY RUN F              1000MM/MIN
G90  G50  G25   H       0        PARTS COUNT                   953
G22  G67  G160  D       0        RUN TIME                12H15M33S
G94  G97  G13.1                  CYCLE TIME                0H 0M 0S
G21  G54  G50.1 T       0
G40  G64  G54.2                  A>
G49.1G69  G80.5 S       0
                                 MEM  STOP *** ***    12:00:00  PATH1
```

Fig. 12.10 (a) Current position screen

The cumulative value of automatic operation time, the cumulative value of cutting time, and timer values that can be set freely can be modified and preset by MDI unit.



ACTUAL POSITION O0123 N00000

ABSOLUTE

X 0.000
Y 0.000
Z 0.000
B 0.000
C 0.000

F 0 MM/MIN
PARTS COUNT 953
RUN TIME 12H15M33S
CYCLE TIME 0H 0M 0S

SETTING (TIMER)

PART TOTAL = 953
PART REQUIRED = 0
PART COUNT = 953
POWER ON = 86H10M
RUN TIME = 12H15M33S
CUTTING TIME = 3H33M39S
FREE PURPOSE = 0H 0M 0S
CYCLE TIME = 0H 0M 0S
DATE = 2003/01/29
TIME = 12:00:00

MODAL

```
G00   G80   G15    F        M      0
G17   G98   G40.1
G90   G50   G25    H        0
G22   G67   G160   D        0
G94   G97   G13.1
G21   G54   G50.1  T        0
G40   G64   G54.2  S        0
G49.1G69    G80.5
```

A>

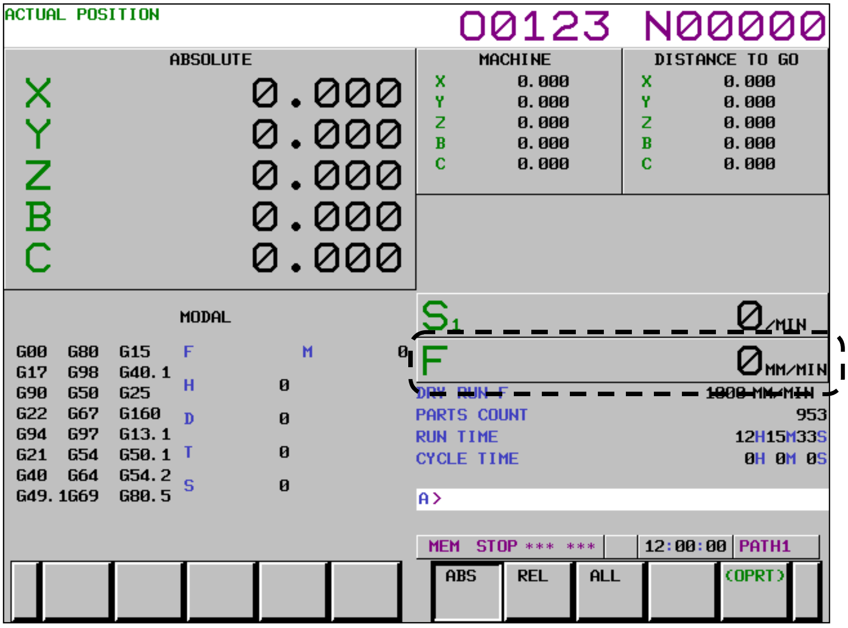
MEM STOP *** *** 12:00:00 PATH1

Fig. 12.10 (b) Setting (timer) screen

12.11 ACTUAL CUTTING FEEDRATE DISPLAY

The actual machine federate per minute can be indicated on the current position screen and program check screen by parameter setting.

(Area indicated by ⬚)



ACTUAL POSITION O0123 N00000

ABSOLUTE

X 0.000
Y 0.000
Z 0.000
B 0.000
C 0.000

MACHINE

X 0.000
Y 0.000
Z 0.000
B 0.000
C 0.000

DISTANCE TO GO

X 0.000
Y 0.000
Z 0.000
B 0.000
C 0.000

S_1 0/MIN
F 0MM/MIN
DRY RUN F 1000MM/MIN
PARTS COUNT 953
RUN TIME 12H15M33S
CYCLE TIME 0H 0M 0S

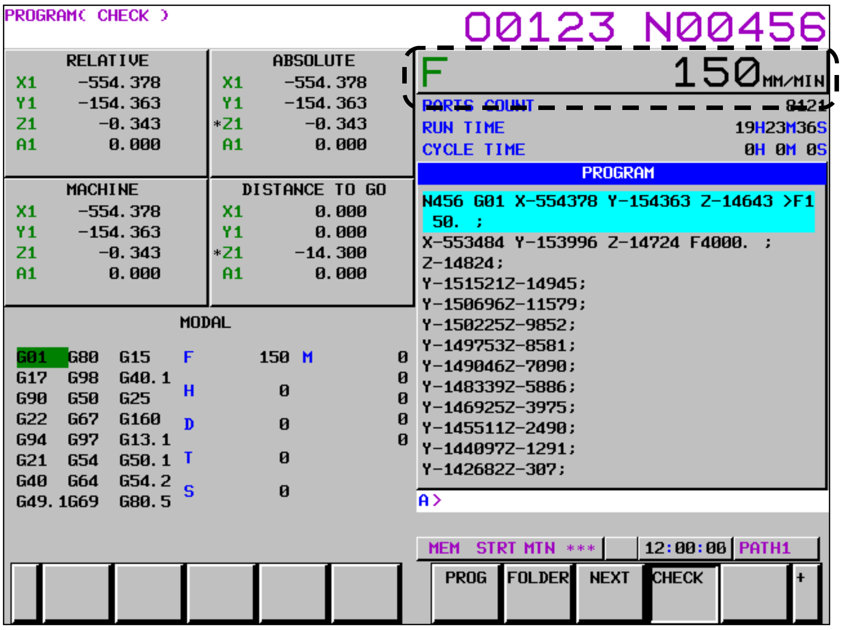
MODAL

```
G00   G80   G15    F        M      0
G17   G98   G40.1
G90   G50   G25    H        0
G22   G67   G160   D        0
G94   G97   G13.1
G21   G54   G50.1  T        0
G40   G64   G54.2  S        0
G49.1G69    G80.5
```

A>

MEM STOP *** *** 12:00:00 PATH1

Fig. 12.11 (a) Current position screen



```
PROGRAM( CHECK )                                  O0123 N00456
        RELATIVE                  ABSOLUTE        F                   150MM/MIN
 X1     -554.378         X1     -554.378          PARTS COUNT              8121
 Y1     -154.363         Y1     -154.363          RUN TIME            19H23M36S
 Z1       -0.343        *Z1       -0.343          CYCLE TIME            0H 0M 0S
 A1        0.000         A1        0.000                   PROGRAM
        MACHINE                DISTANCE TO GO     N456 G01 X-554378 Y-154363 Z-14643 >F1
 X1     -554.378         X1        0.000           50. ;
 Y1     -154.363         Y1        0.000          X-553484 Y-153996 Z-14724 F4000. ;
 Z1       -0.343        *Z1      -14.300          Z-14824;
 A1        0.000         A1        0.000          Y-151521Z-14945;
                  MODAL                           Y-150696Z-11579;
                                                  Y-150225Z-9852;
 G01   G80   G15    F        150  M          0    Y-149753Z-8581;
 G17   G98   G40.1                           0    Y-149046Z-7090;
 G90   G50   G25    H          0             0    Y-148339Z-5886;
 G22   G67   G160   D          0             0    Y-146925Z-3975;
 G94   G97   G13.1                           0    Y-145511Z-2490;
 G21   G54   G50.1  T          0                  Y-144097Z-1291;
 G40   G64   G54.2                                Y-142682Z-307;
 G49.1G69    G80.5  S          0                  A>

                                                  MEM  STRT MTN ***     12:00:00 PATH1
```

Fig. 12.11 (b) Program check screen

12.12 DISPLAY OF SPINDLE SPEED AND T CODE AT ALL SCREENS

The spindle speed calculated from feedback pulses from the position coder mounted on the spindle and a T code specified by program execution can always displayed by parameter setting.

12.13 DIRECTORY DISPLAY OF FLOPPY CASSETTE

A list of file names in an external I/O device can be displayed on the screen (directory display).
File names that can be displayed in the directory are up to 17 characters long.

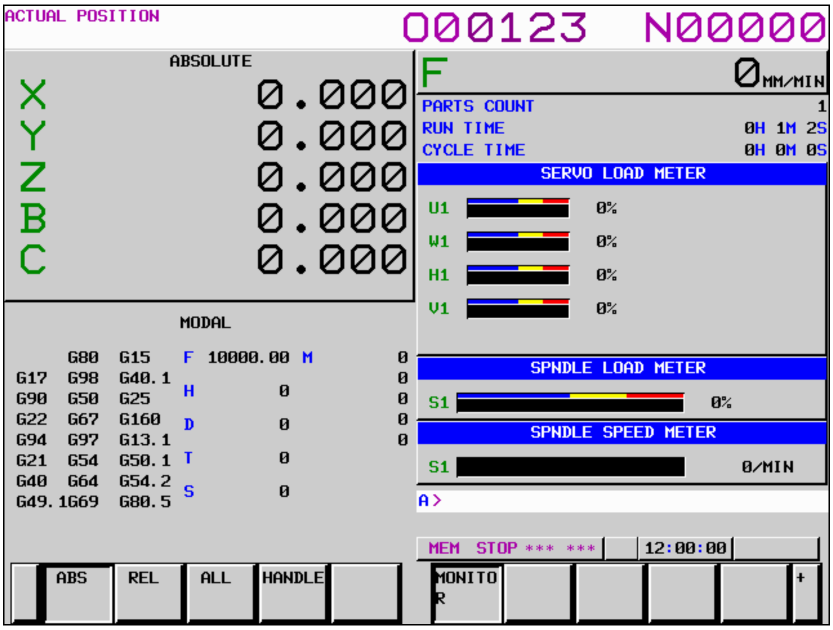
When a part program stored in program memory is written to an external I/O device, a program number can be assigned as the file name. When data other than part programs is written, a name specific to the data can be assigned.

12.14 OPTIONAL PATH NAME DISPLAY

An arbitrary character string can be displayed as a path name indicated in the status display field by parameter setting. Up to seven characters including numerical, alphabetical, and katakana characters and symbols can be displayed.

12.15 OPERATING MONITOR SCREEN

Load values (torque values) of spindle and servo motors are displayed in the bar graph form on the screen. In the bar graph display, the latest sampling values are indicated. By setting the rated load value of the motor corresponding to each load meter in a parameter in advance, the load meter shows 100% when the load value equals the rated load value.



```
ACTUAL POSITION                              O00123   N00000
                  ABSOLUTE          F                     0MM/MIN
X                    0.000          PARTS COUNT                 1
Y                    0.000          RUN TIME              0H 1M 2S
Z                    0.000          CYCLE TIME            0H 0M 0S
B                    0.000                SERVO LOAD METER
C                    0.000          U1                 0%
                                    W1                 0%
                                    H1                 0%
                                    V1                 0%
                    MODAL
     G80  G15   F 10000.00 M      0
G17  G98  G40.1                   0       SPNDLE LOAD METER
G90  G50  G25   H       0         0
G22  G67  G160                    0  S1                     0%
G94  G97  G13.1 D       0         0       SPNDLE SPEED METER
G21  G54  G50.1 T       0
G40  G64  G54.2                      S1                   0/MIN
G49.1G69  G80.5 S       0           A>

                                    MEM  STOP *** ***   12:00:00
```

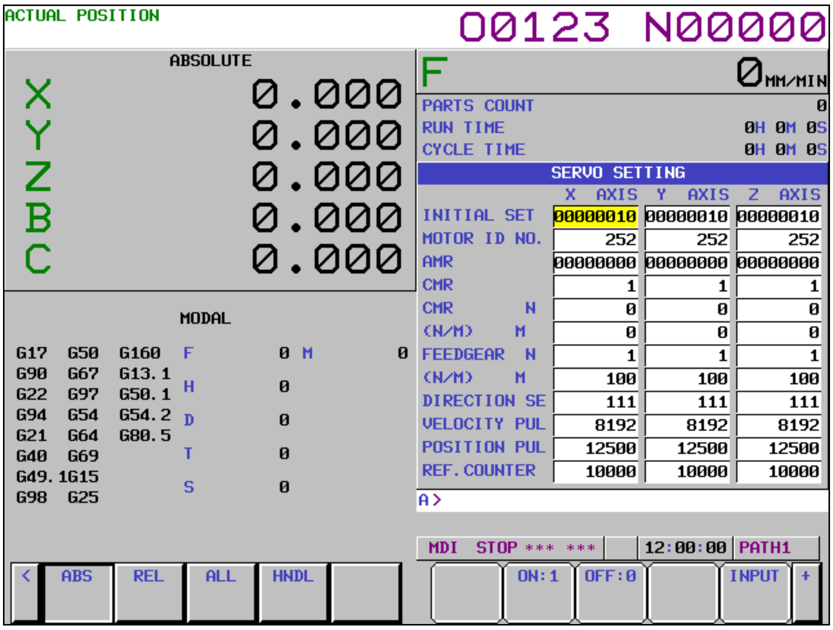
Fig. 12.15 (a) Operating monitor screen

12.16 SERVO SETTING SCREEN

Necessary parameters for basic servo motor tuning and the status monitor are displayed for each axis.

12.16.1 Servo Setting Screen

This screen summarizes necessary parameters for initial standard setting of servo motors. Parameter setting is also possible on this screen.



```
ACTUAL POSITION                          O0123 N00000
            ABSOLUTE                     F                 0MM/MIN
X                  0.000                 PARTS COUNT             0
Y                  0.000                 RUN TIME         0H 0M 0S
Z                  0.000                 CYCLE TIME       0H 0M 0S
B                  0.000
C                  0.000

              MODAL
G17  G50  G160   F        0  M         0
G90  G67  G13.1  H        0
G22  G97  G50.1  D        0
G94  G54  G54.2  T        0
G21  G64  G80.5  S        0
G40  G69
G49.1G15
G98  G25
```

SERVO SETTING

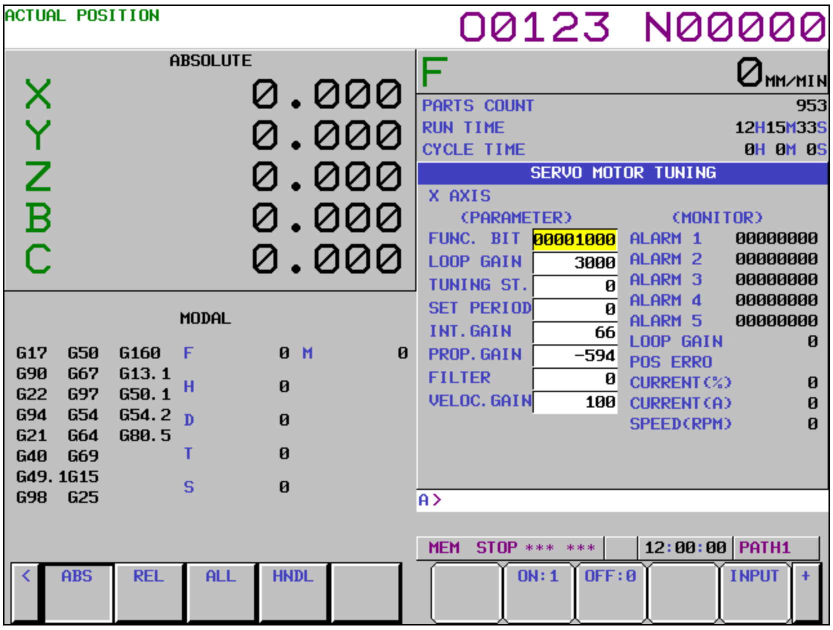
	X AXIS	Y AXIS	Z AXIS
INITIAL SET	00000010	00000010	00000010
MOTOR ID NO.	252	252	252
AMR	00000000	00000000	00000000
CMR	1	1	1
CMR N	0	0	0
(N/M) M	0	0	0
FEEDGEAR N	1	1	1
(N/M) M	100	100	100
DIRECTION SE	111	111	111
VELOCITY PUL	8192	8192	8192
POSITION PUL	12500	12500	12500
REF.COUNTER	10000	10000	10000

```
A>
MDI  STOP *** ***      12:00:00 PATH1
```

Fig. 12.16.1 (a) Servo parameter setting screen

12.16.2 Servo Motor Tuning Screen

This screen summarizes necessary parameters for basic servo motor tuning and the status monitor for each axis.



```
ACTUAL POSITION                          O0123 N00000
            ABSOLUTE                     F                 0MM/MIN
X                  0.000                 PARTS COUNT           953
Y                  0.000                 RUN TIME        12H15M33S
Z                  0.000                 CYCLE TIME       0H 0M 0S
B                  0.000
C                  0.000

              MODAL
G17  G50  G160   F        0  M         0
G90  G67  G13.1  H        0
G22  G97  G50.1  D        0
G94  G54  G54.2  T        0
G21  G64  G80.5  S        0
G40  G69
G49.1G15
G98  G25
```

SERVO MOTOR TUNING

X AXIS

(PARAMETER)		(MONITOR)	
FUNC. BIT	00001000	ALARM 1	00000000
LOOP GAIN	3000	ALARM 2	00000000
TUNING ST.	0	ALARM 3	00000000
SET PERIOD	0	ALARM 4	00000000
INT.GAIN	66	ALARM 5	00000000
PROP.GAIN	-594	LOOP GAIN	0
FILTER	0	POS ERRO	
VELOC.GAIN	100	CURRENT(%)	0
		CURRENT(A)	0
		SPEED(RPM)	0

```
A>
MEM  STOP *** ***      12:00:00 PATH1
```

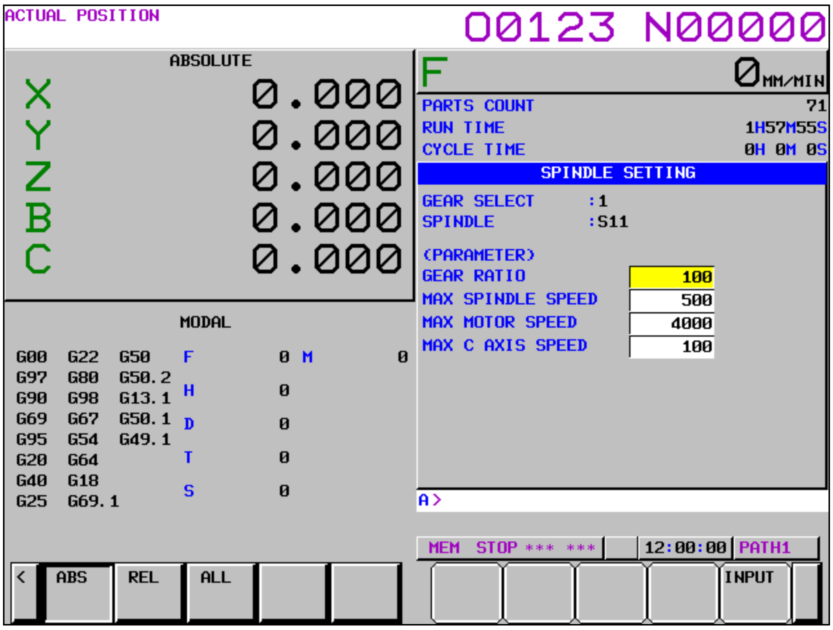
Fig. 12.16.2 (a) Servo motor tuning screen

12.17 SPINDLE SETTING SCREEN

Spindle-related parameters are set and displayed. Data other than parameters can also be displayed. Spindle setting, spindle tuning, and spindle monitor screens are provided.

12.17.1 Spindle Setting Screen

This screen summarizes necessary parameters for initial spindle setting.
Parameters can also be set.

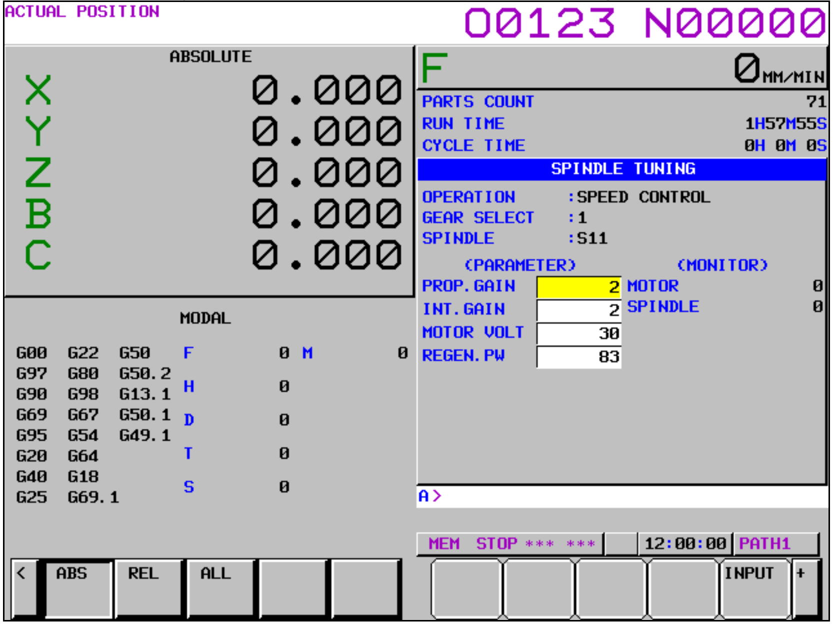


```
ACTUAL POSITION                        O0123 N00000
            ABSOLUTE            F              0MM/MIN
X                 0.000         PARTS COUNT          71
Y                 0.000         RUN TIME       1H57M55S
Z                 0.000         CYCLE TIME     0H 0M 0S
B                 0.000              SPINDLE SETTING
C                 0.000         GEAR SELECT      :1
                                SPINDLE          :S11
                                <PARAMETER>
                                GEAR RATIO               100
                                MAX SPINDLE SPEED        500
             MODAL              MAX MOTOR SPEED         4000
G00  G22  G50    F      0 M   0 MAX C AXIS SPEED         100
G97  G80  G50.2  H      0
G90  G98  G13.1  D      0
G69  G67  G50.1  T      0
G95  G54  G49.1  S      0
G20  G64
G40  G18
G25  G69.1                      A>
                                MEM STOP *** ***   12:00:00 PATH1
```

Fig. 12.17.1 (a) Spindle setting screen

12.17.2 Spindle Tuning Screen

This screen summarizes necessary parameters for basic spindle tuning and the status monitor.

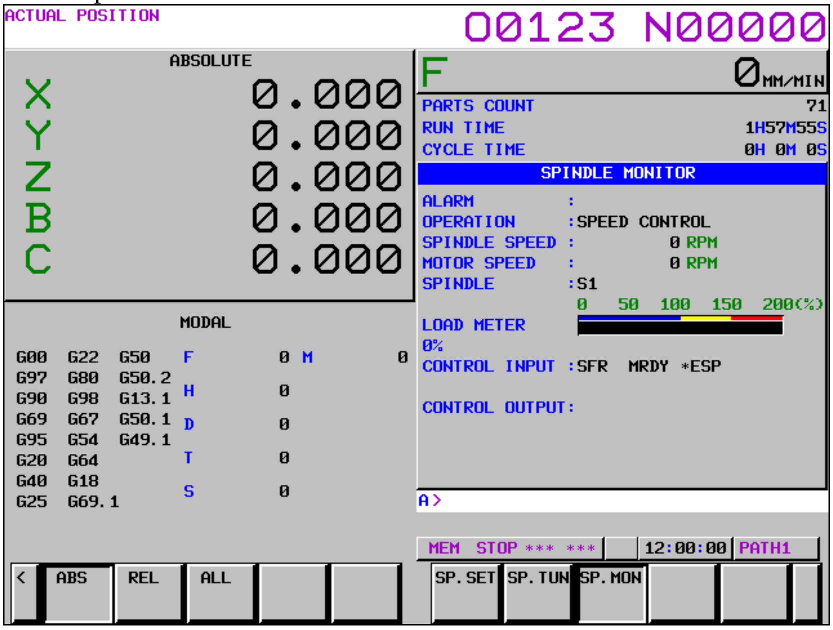


```
ACTUAL POSITION                        O0123 N00000
            ABSOLUTE            F              0MM/MIN
X                 0.000         PARTS COUNT          71
Y                 0.000         RUN TIME       1H57M55S
Z                 0.000         CYCLE TIME     0H 0M 0S
B                 0.000               SPINDLE TUNING
C                 0.000         OPERATION     :SPEED CONTROL
                                GEAR SELECT   :1
                                SPINDLE       :S11
                                    <PARAMETER>      <MONITOR>
                                PROP.GAIN       2 MOTOR      0
             MODAL              INT.GAIN        2 SPINDLE    0
G00  G22  G50    F      0 M   0 MOTOR VOLT     30
G97  G80  G50.2  H      0       REGEN.PW       83
G90  G98  G13.1  D      0
G69  G67  G50.1  T      0
G95  G54  G49.1  S      0
G20  G64
G40  G18
G25  G69.1                      A>
                                MEM STOP *** ***   12:00:00 PATH1
```

Fig. 12.17.2 (a) Spindle tuning screen

12.17.3 Spindle Monitor Screen

This screen summarizes spindle information.


ACTUAL POSITION

O0123 N00000

ABSOLUTE

X 0.000

Y 0.000

Z 0.000

B 0.000

C 0.000

F 0MM/MIN

PARTS COUNT 71

RUN TIME 1H57M55S

CYCLE TIME 0H 0M 0S

SPINDLE MONITOR

ALARM :

OPERATION :SPEED CONTROL

SPINDLE SPEED : 0 RPM

MOTOR SPEED : 0 RPM

SPINDLE :S1

LOAD METER 0%

CONTROL INPUT :SFR MRDY *ESP

CONTROL OUTPUT:

MODAL

G00 G22 G50

G97 G80 G50.2

G90 G98 G13.1

G69 G67 G50.1

G95 G54 G49.1

G20 G64

G40 G18

G25 G69.1

F 0 M 0

H 0

D 0

T 0

S 0

A>

MEM STOP *** *** 12:00:00 PATH1

Fig. 12.17.3 (a) Spindle monitor screen

12.18 SERVO WAVEFORM DISPLAY

Servo data (including the error amount, torque amount, and the number of distributed pulses) and the waveforms of signals transferred between the CNC and PMC can be displayed.

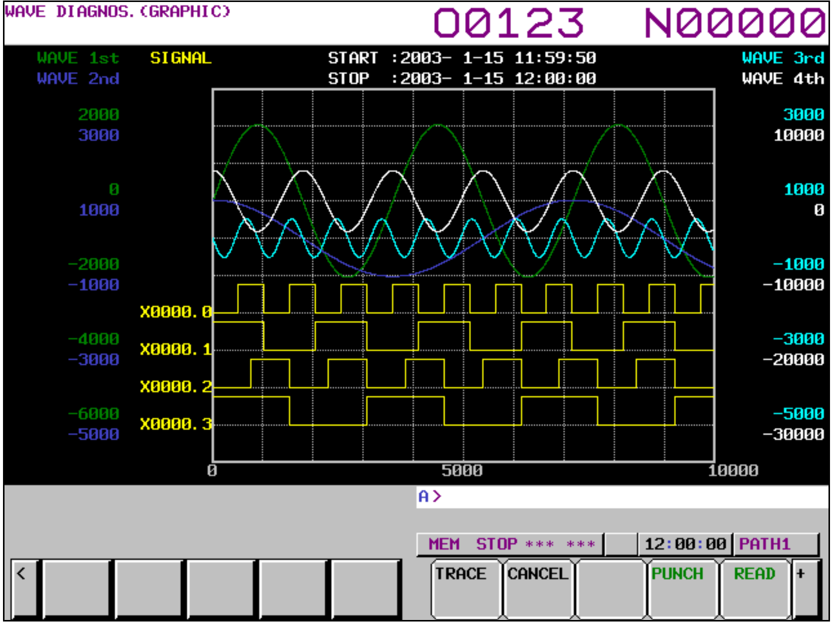


Fig. 12.18 (a) Waveform diagnosis screen

12.19 MAINTENANCE INFORMATION SCREEN

On the maintenance information screen, a history of maintenance operations can be recorded.
The following operations can be performed:

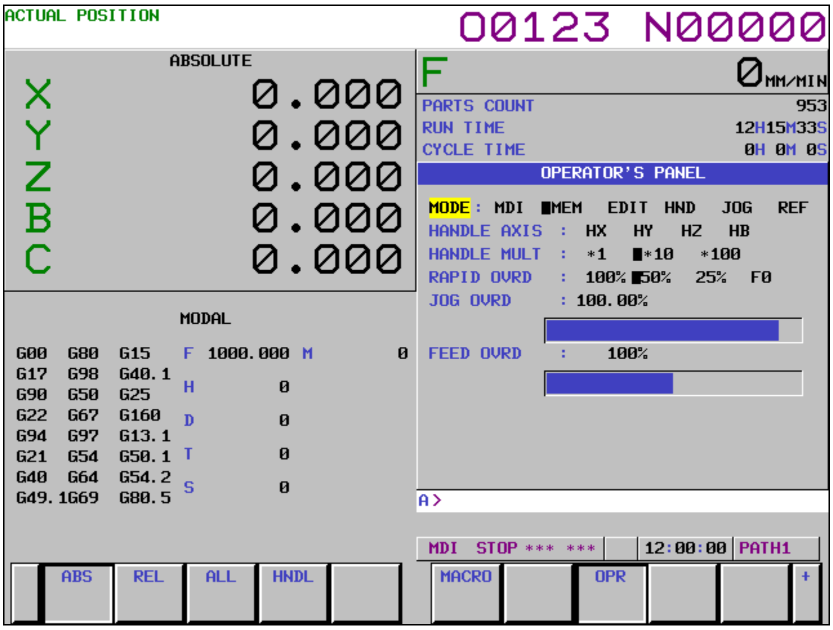
- Input of alphabetical characters by MDI (En-size kana characters can be input only in Japanese display mode.)
- Reference of the record screen by line-by-line scroll
- Input/output of maintenance information after editing
- Write and save in FLASH-ROM
- Display of Double-byte code (Shift-JIS) (Input is enabled by file only.)

12.20 SOFTWARE OPERATOR'S PANEL

The functions of switches on the machine operator's panel are made available by using the MDI unit. This means that functions such as a mode selection and selection of jog feedrate override can be made by operating the MDI unit, so corresponding switches on the machine operator's panel need not be used.
This function is enabled only when the software operator's panel screen is displayed. The user can make selection operations by moving the cursor by using the cursor keys while checking the screen. The following functions can be performed:

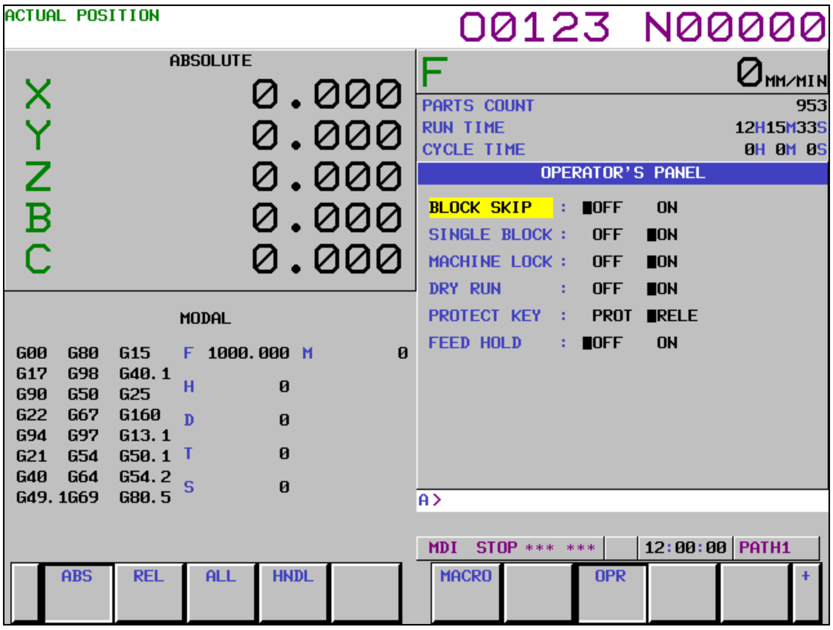
A : Mode selection
B : Selection of jog feed axis, manual rapid traverse
C : Selection of manual pulse generator feed axis, selection of manual pulse magnification
D : Jog federate, federate override, rapid traverse override
E : Optional block skip, single block, machine lock, dry run
F : Protect key
G : Feed hold
H : General purpose switch

A parameter that enables and disables MDI operations is provided for each of operation groups A to G indicated above.



```
ACTUAL POSITION                              O0123 N00000
               ABSOLUTE                F                  0MM/MIN
 X                0.000                PARTS COUNT             953
 Y                0.000                RUN TIME          12H15M33S
 Z                0.000                CYCLE TIME           0H 0M 0S
 B                0.000                       OPERATOR'S PANEL
 C                0.000                 MODE : MDI ■MEM EDIT HND JOG REF
                                        HANDLE AXIS : HX  HY  HZ  HB
                                        HANDLE MULT : *1 ■*10 *100
                                        RAPID OVRD  : 100% ■50% 25% F0
                                        JOG OVRD    : 100.00%
                 MODAL
 G00  G80  G15    F 1000.000 M       0  FEED OVRD   :   100%
 G17  G98  G40.1
 G90  G50  G25    H        0
 G22  G67  G160   D        0
 G94  G97  G13.1
 G21  G54  G50.1  T        0
 G40  G64  G54.2  S        0
 G49.1G69  G80.5
                                       A>
                                        MDI  STOP *** ***  12:00:00 PATH1
```

Fig. 12.20 (a) Software operator's panel screen (1/2)



```
ACTUAL POSITION                              O0123 N00000
               ABSOLUTE                F                  0MM/MIN
 X                0.000                PARTS COUNT             953
 Y                0.000                RUN TIME          12H15M33S
 Z                0.000                CYCLE TIME           0H 0M 0S
 B                0.000                       OPERATOR'S PANEL
 C                0.000                 BLOCK SKIP    : ■OFF   ON
                                        SINGLE BLOCK  :  OFF  ■ON
                                        MACHINE LOCK  :  OFF  ■ON
                                        DRY RUN       :  OFF  ■ON
                 MODAL                  PROTECT KEY   :  PROT ■RELE
                                        FEED HOLD     : ■OFF   ON
 G00  G80  G15    F 1000.000 M       0
 G17  G98  G40.1
 G90  G50  G25    H        0
 G22  G67  G160   D        0
 G94  G97  G13.1
 G21  G54  G50.1  T        0
 G40  G64  G54.2  S        0
 G49.1G69  G80.5
                                       A>
                                        MDI  STOP *** ***  12:00:00 PATH1
```

Fig. 12.20 (b) Software operator's panel screen (2/2)

12.21 SOFTWARE OPERATOR'S PANEL GENERAL PURPOSE SWITCH

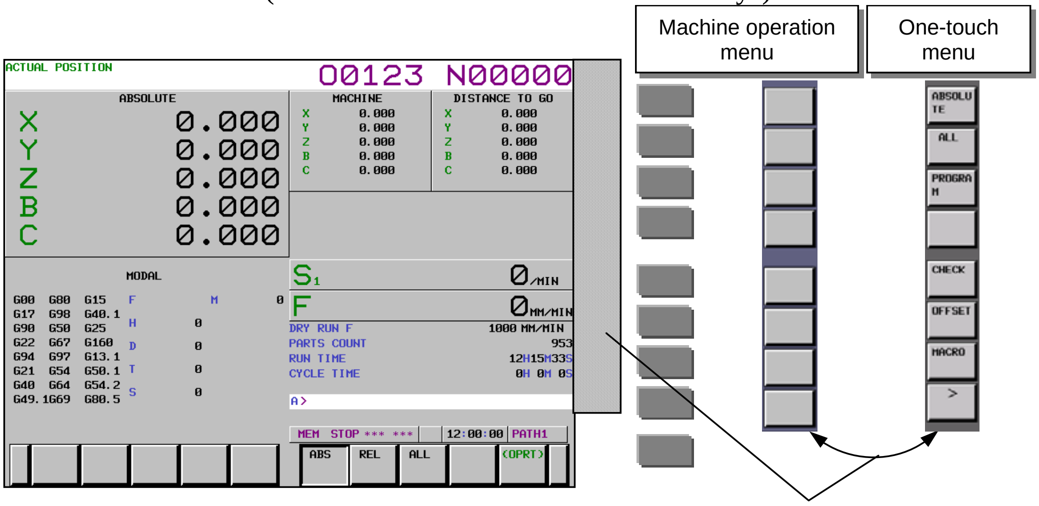
This function allows the user to assign arbitrary signals to general purpose switches 1 to 16 on the software operator's panel.

12.22 MACHINE OPERATION MENU

Soft keys displayed on the CNC standard screen can be used as machine operation menu keys. Machine tool builders can easily customize the soft keys to organize menus in a hierarchical form and specify the labels of the soft keys.

Data on what the machine tool builders customized is created as a machine operation menu definition file and registered with the CNC of interest.

The customization function can be used with 8.4-inch LCD unit (only those with horizontal soft keys), 10.4/15/19-inch LCD units (those with vertical or horizontal soft keys).

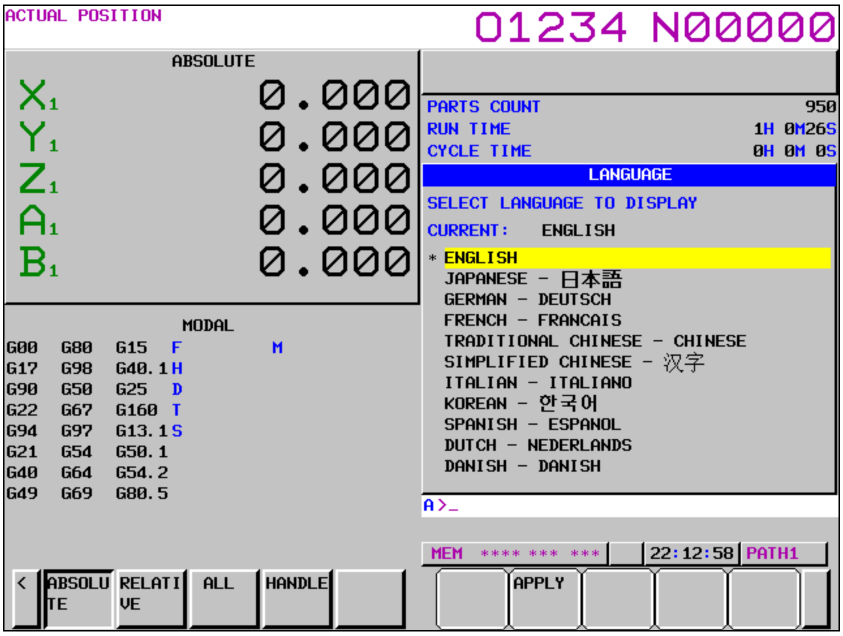


The ninth vertical soft key can be used to specify whether to show or hide menus and which menu to select.

Fig. 12.22 (a) 10.4-inch LCD unit with vertical soft keys

12.23 MULTI-LANGUAGE DISPLAY

The following 23 languages are prepared as the display languages.
English, Japanese, German, French, Spanish, Italian, Chinese (traditional), Chinese (simplified), Korean, Portuguese, Dutch, Danish, Swedish, Hungarian, Czech, Polish, Russian, Turkish, Bulgarian, Rumanian, Slovak, Finnish and Hindi.
When a desired language is selected and set on the language screen, the display language can be changed immediately.



```
ACTUAL POSITION                                O1234 N00000
            ABSOLUTE
X1          0.000
Y1          0.000          PARTS COUNT                  950
Z1          0.000          RUN TIME                  1H 0M26S
A1          0.000          CYCLE TIME                0H 0M 0S
B1          0.000                    LANGUAGE
                           SELECT LANGUAGE TO DISPLAY
              MODAL        CURRENT:   ENGLISH
G00  G80  G15   F     M    * ENGLISH
G17  G98  G40.1 H            JAPANESE - 日本語
G90  G50  G25   D            GERMAN - DEUTSCH
G22  G67  G160  T            FRENCH - FRANCAIS
G94  G97  G13.1 S            TRADITIONAL CHINESE - CHINESE
G21  G54  G50.1              SIMPLIFIED CHINESE - 汉字
G40  G64  G54.2              ITALIAN - ITALIANO
G49  G69  G80.5              KOREAN - 한국어
                             SPANISH - ESPANOL
                             DUTCH - NEDERLANDS
                             DANISH - DANISH
                           A>_

                           MEM **** *** ***   22:12:58 PATH1
```

Fig. 12.23 (a) Language screen

> **NOTE**
> When PMC signals are used to change the display language, the display language cannot be changed on the language screen.

12.23.1 Changing the Display Language by PMC Signals

PMC signals can be used to change the display language of the CNC screen. In this function, a dial or switch on the machine operator's panel can also be used to change the language of the CNC screen.

12.24 DATA PROTECTION KEY

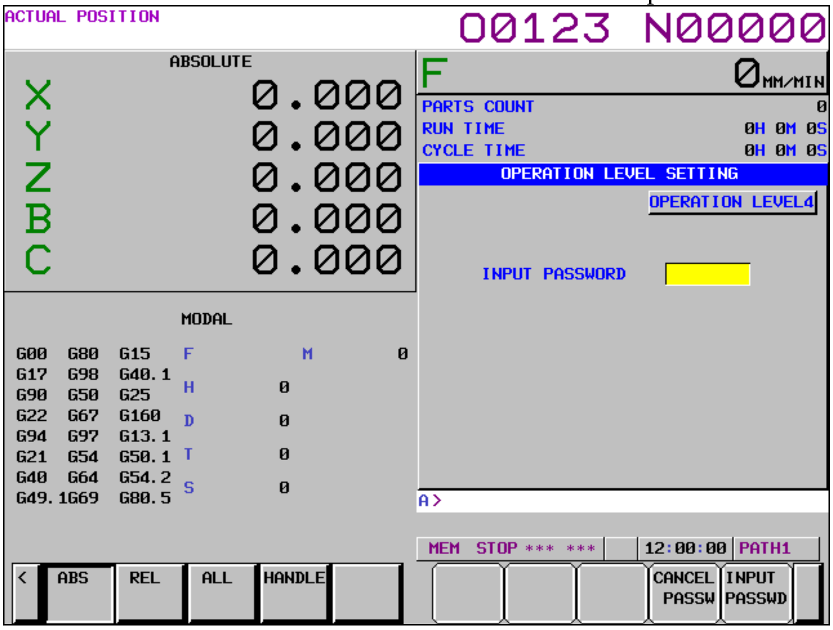
To protect various types of data in the CNC, data protection keys can be provided for the machine. Depending on the type of the data to be protected, the following four input signals are provided:

- KEY1
  Enables input of tool offset values and workpiece origin offset values.
- KEY2
  Enables input of settings.
- KEY3
  Enables program registration and editing.
- KEY4
  Enables input of PMC data.

12.25 PROTECTION OF DATA AT EIGHT LEVELS

Eight operation levels can be set for CNC and PMC operations, and eight protect levels can be set for various types of CNC and PMC data.

When CNC and PMC data is modified or output externally, the operation level and protect level are compared to determine whether to allow the modification or external output.

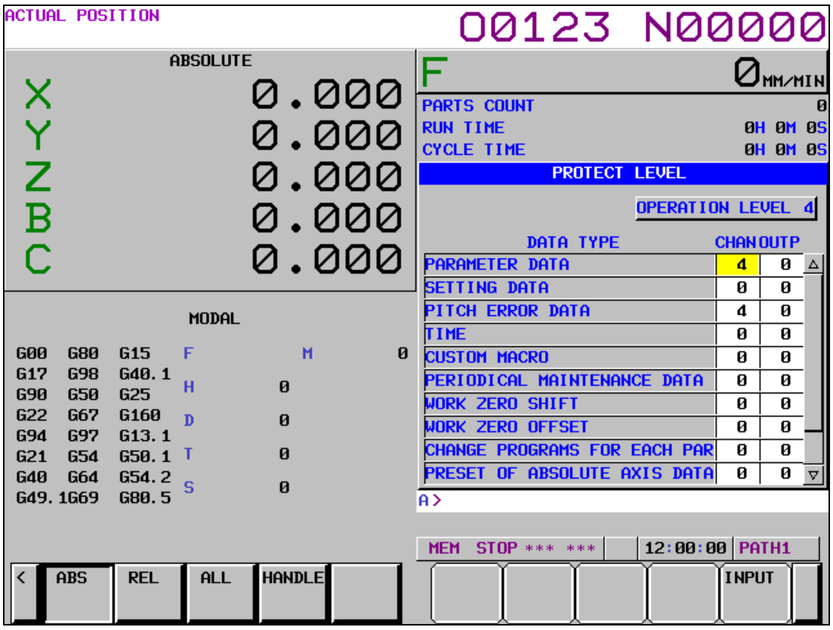


OPERATION LEVEL SETTING

OPERATION LEVEL4

INPUT PASSWORD

Fig. 12.25 (a) Operation level setting screen



PROTECT LEVEL

OPERATION LEVEL 4

DATA TYPE	CHAN	OUTP
PARAMETER DATA	4	0
SETTING DATA	0	0
PITCH ERROR DATA	4	0
TIME	0	0
CUSTOM MACRO	0	0
PERIODICAL MAINTENANCE DATA	0	0
WORK ZERO SHIFT	0	0
WORK ZERO OFFSET	0	0
CHANGE PROGRAMS FOR EACH PAR	0	0
PRESET OF ABSOLUTE AXIS DATA	0	0

Fig. 12.25 (b) Protect level setting screen

NOTE

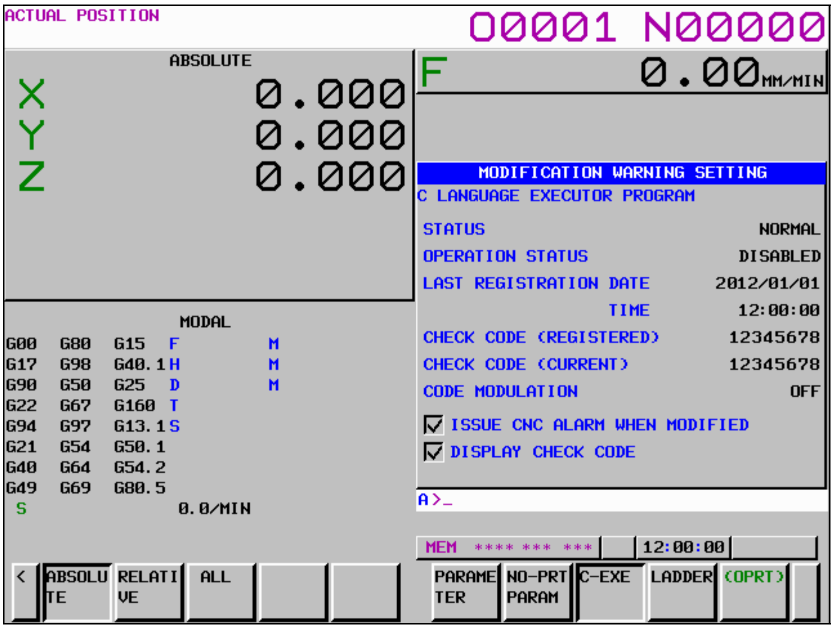
This function is an optional function.

12.26 WARNING FUNCTION AGAINST MODIFICATION OF SETTING

This function is to protect important setting of parameters, C Language Executor programs, or ladder programs on CNC and to detect unauthorized modifications on them. The following parameters and programs can be protected.

- CNC Parameters (which are selected to be protected)
- C Language Executor program
- Ladder program (PMC ladder program of each path)
- Dual Check Safety PMC Ladder program

After the registration of above parameters, C Language Executor programs, or ladder programs to this function, those are verified by CNC, when the power of CNC is turned on. If any modification is applied to registered parameters, C Language Executor programs, or ladder programs the signal that means some modification is applied is output. And in this case, it is also possible to generate the alarm.



MODIFICATION WARNING SETTING

C LANGUAGE EXECUTOR PROGRAM

STATUS NORMAL
OPERATION STATUS DISABLED
LAST REGISTRATION DATE 2012/01/01
TIME 12:00:00
CHECK CODE (REGISTERED) 12345678
CHECK CODE (CURRENT) 12345678
CODE MODULATION OFF
☑ ISSUE CNC ALARM WHEN MODIFIED
☑ DISPLAY CHECK CODE

Fig 12.26 (a) MODIFICATION WARNING SETTING screen

NOTE

This function is an optional function.

12.27 ERASE CRT SCREEN DISPLAY

If the same characters are left displayed in the same position of the screen for an extended time, the service life of the display will be shortened. To prevent this, the CNC screen display can be erased.

The screen display can be erased by pressing keys (pressing CAN key and the function key at the same time) (screen erasure). The screen display can also be erased automatically when no key operation has been performed for a parameter-set time (automatic screen erasure).

12.28 PARAMETER SET SUPPORTING SCREEN

The parameter set supporting screen allows parameter setting and tuning for the following purposes:

(1) Displaying a list of the least required parameters for machine startup so that the machine can be started easily
(2) Displaying a servo tuning screen, spindle tuning screen, and machining parameter tuning screen to allow smooth tuning

On this screen, the following items can be set and tuned:

Support of startup
- System setting
- Axis setting
- FSSB (amplifier)
- FSSB (axis)
- Servo setting
- Spindle setting
- Miscellany

Support of tuning
- Servo tuning
- Spindle tuning
- AICC tuning

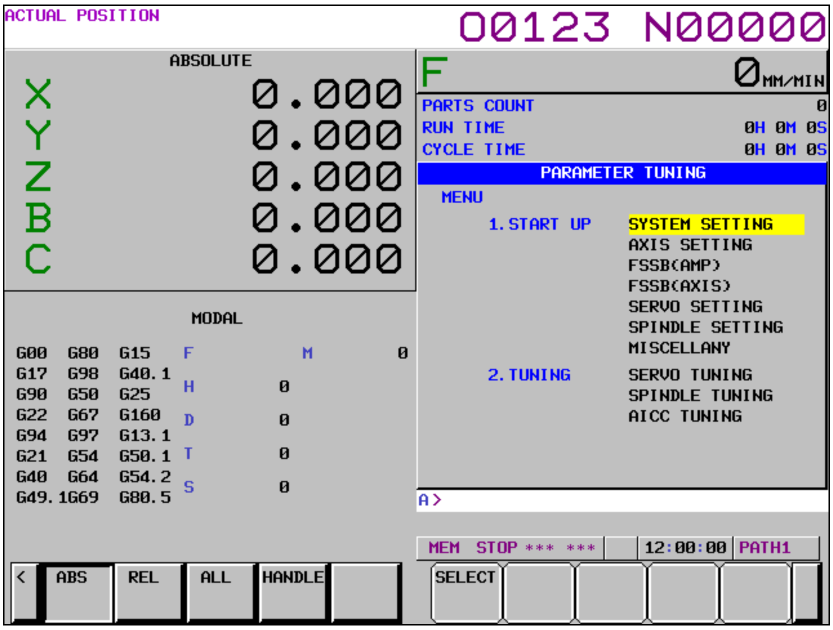
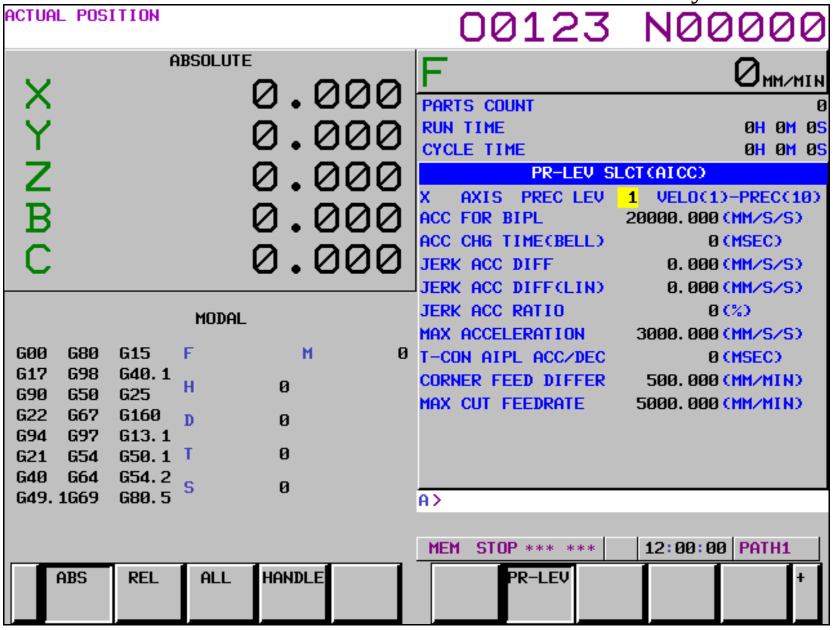


Fig. 12.28 (a) Menu screen for parameter tuning

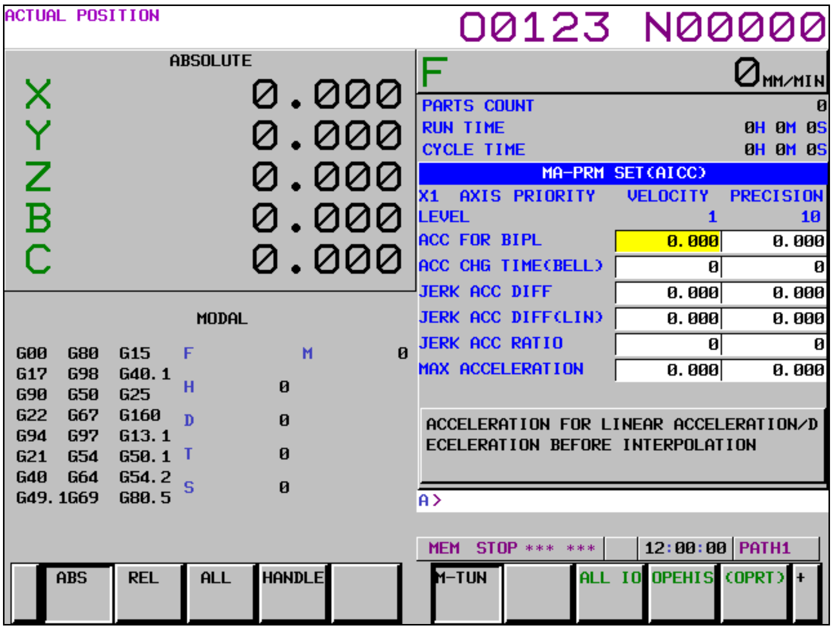
12.29 MACHINING CONDITION SELECTING FUNCTION

In AI advanced preview control (M Series) / AI contour control, when speed-oriented and precision-oriented parameter sets are set, and a precision level is set according to the machining condition such as rough machining or finish machining on the precision level selection screen or with a programmed command, parameters that meet the condition can be calculated automatically for machining.



```
ACTUAL POSITION                              O0123 N00000
          ABSOLUTE                  F                  0MM/MIN
X                  0.000            PARTS COUNT                0
Y                  0.000            RUN TIME             0H 0M 0S
Z                  0.000            CYCLE TIME           0H 0M 0S
B                  0.000                    PR-LEV SLCT(AICC)
C                  0.000            X   AXIS  PREC LEV  1  VELO(1)-PREC(10)
                                    ACC FOR BIPL       20000.000(MM/S/S)
            MODAL                   ACC CHG TIME(BELL)         0(MSEC)
G00  G80  G15   F         M     0   JERK ACC DIFF          0.000(MM/S/S)
G17  G98  G40.1                     JERK ACC DIFF(LIN)     0.000(MM/S/S)
G90  G50  G25   H       0           JERK ACC RATIO             0(%)
G22  G67  G160  D       0           MAX ACCELERATION    3000.000(MM/S/S)
G94  G97  G13.1                     T-CON AIPL ACC/DEC         0(MSEC)
G21  G54  G50.1 T       0           CORNER FEED DIFFER   500.000(MM/MIN)
G40  G64  G54.2                     MAX CUT FEEDRATE    5000.000(MM/MIN)
G49.1G69  G80.5 S       0
                                    A>
                                    MEM  STOP *** ***    12:00:00 PATH1
```

Fig. 12.29 (a) Precision level selection screen



```
ACTUAL POSITION                              O0123 N00000
          ABSOLUTE                  F                  0MM/MIN
X                  0.000            PARTS COUNT                0
Y                  0.000            RUN TIME             0H 0M 0S
Z                  0.000            CYCLE TIME           0H 0M 0S
B                  0.000                    MA-PRM SET(AICC)
C                  0.000            X1  AXIS PRIORITY  VELOCITY  PRECISION
                                    LEVEL                     1         10
            MODAL                   ACC FOR BIPL          0.000      0.000
G00  G80  G15   F         M     0   ACC CHG TIME(BELL)        0          0
G17  G98  G40.1                     JERK ACC DIFF         0.000      0.000
G90  G50  G25   H       0           JERK ACC DIFF(LIN)    0.000      0.000
G22  G67  G160  D       0           JERK ACC RATIO            0          0
G94  G97  G13.1                     MAX ACCELERATION      0.000      0.000
G21  G54  G50.1 T       0
G40  G64  G54.2                     ACCELERATION FOR LINEAR ACCELERATION/D
G49.1G69  G80.5 S       0           ECELERATION BEFORE INTERPOLATION
                                    A>
                                    MEM  STOP *** ***    12:00:00 PATH1
```

Fig. 12.29 (b) Machining parameter setting screen

NOTE

This function is an optional function.

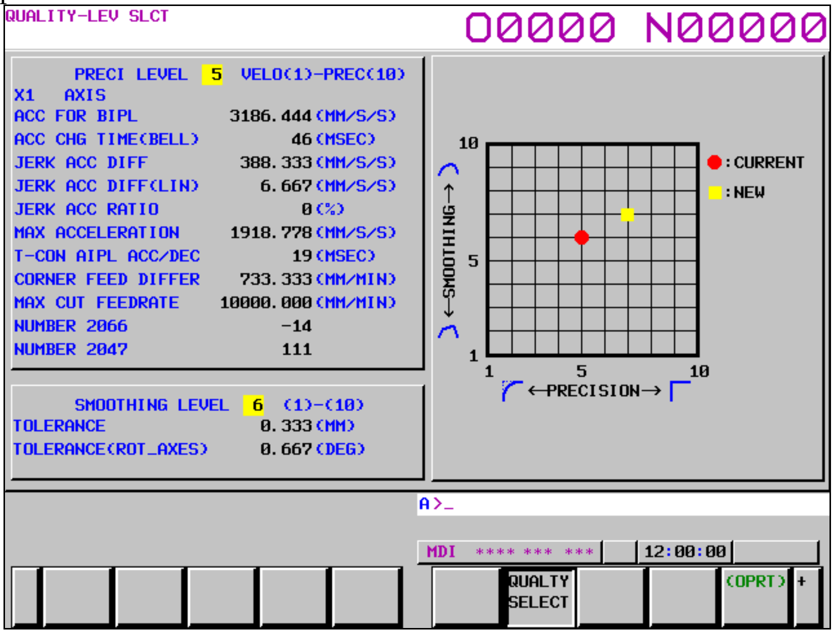
12.30 MACHINING QUALITY LEVEL ADJUSTMENT (M SERIES)

M

In nano smoothing, if the “level 1” and “level 10” parameters of a precision level and smoothing level are set in order to specify a precision level and smoothing level according to the machining condition during machining, the parameter values corresponding to the condition can be automatically calculated for machining.

On the machining quality level adjustment screen, the machining quality/precision/speed level in nano smoothing can easily be adjusted.

This function is optional.



```
QUALITY-LEV SLCT                                  O0000 N00000

     PRECI LEVEL   5  VELO(1)-PREC(10)
X1    AXIS
ACC FOR BIPL           3186.444(MM/S/S)
ACC CHG TIME(BELL)           46(MSEC)
JERK ACC DIFF           388.333(MM/S/S)
JERK ACC DIFF(LIN)        6.667(MM/S/S)
JERK ACC RATIO                0(%)
MAX ACCELERATION       1918.778(MM/S/S)
T-CON AIPL ACC/DEC           19(MSEC)
CORNER FEED DIFFER      733.333(MM/MIN)
MAX CUT FEEDRATE      10000.000(MM/MIN)
NUMBER 2066                 -14
NUMBER 2047                 111

     SMOOTHING LEVEL  6  (1)-(10)
TOLERANCE                 0.333(MM)
TOLERANCE(ROT_AXES)       0.667(DEG)

A>_
MDI  **** *** ***      12:00:00
```

Fig. 12.30 (a) Machining quality level selection screen

NOTE

This function is an optional function.

To use this function, the options for "AI contour control II", "Nano smoothing", "Machining condition selecting function" and this function are required.

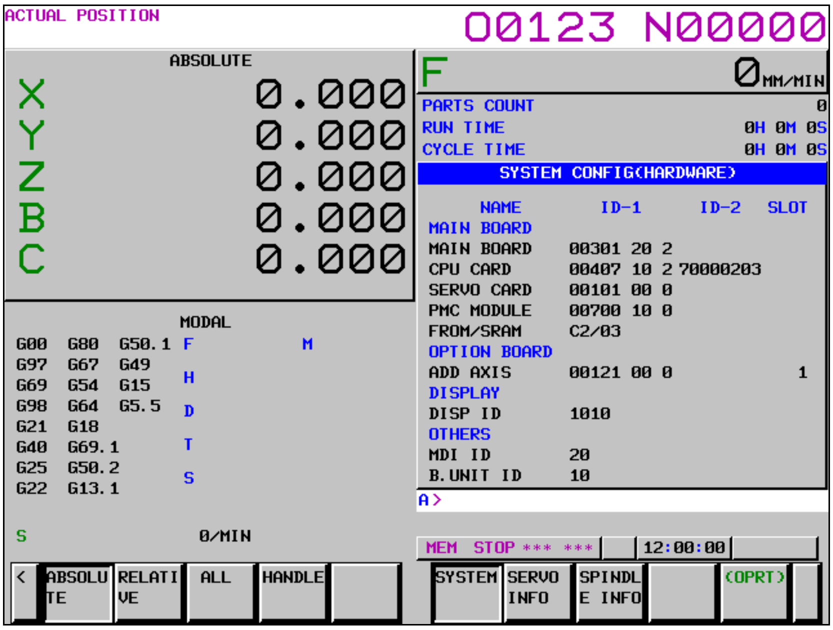
12.31 SYSTEM CONFIGURATION SCREEN

This screen displays the necessary hardware and software configuration for CNC maintenance.

The system configuration screen shows the following:

- Hardware Configuration Screen
- Software Configuration Screen

12.31.1 Hardware Configuration Screen



```
ACTUAL POSITION                          O0123 N00000
            ABSOLUTE             F                 0MM/MIN
X                 0.000          PARTS COUNT             0
Y                 0.000          RUN TIME         0H 0M 0S
Z                 0.000          CYCLE TIME       0H 0M 0S
B                 0.000             SYSTEM CONFIG(HARDWARE)
C                 0.000               NAME      ID-1      ID-2   SLOT
                                  MAIN BOARD
             MODAL                MAIN BOARD   00301 20 2
G00  G80  G50.1 F         M       CPU CARD     00407 10 2 70000203
G97  G67  G49                     SERVO CARD   00101 00 0
G69  G54  G15   H                 PMC MODULE   00700 10 0
G98  G64  G5.5                    FROM/SRAM    C2/03
G21  G18        D                 OPTION BOARD
G40  G69.1                        ADD AXIS     00121 00 0              1
G25  G50.2      T                 DISPLAY
G22  G13.1                        DISP ID      1010
                S                 OTHERS
                                  MDI ID       20
                                  B.UNIT ID    10
                                 A>
S               0/MIN
                                 MEM  STOP *** ***      12:00:00
```

Fig. 12.31.1 (a) Hardware configuration screen

(1) NAME

MAIN BOARD

- Displays information on the main board, and cards and modules on the main board.

OPTION BOARD

- Displays information on the board installed in the option slot.

DISPLAY

- Displays information on the display unit.

OTHERS

- Displays information on other components (such as an MDI and a basic unit).

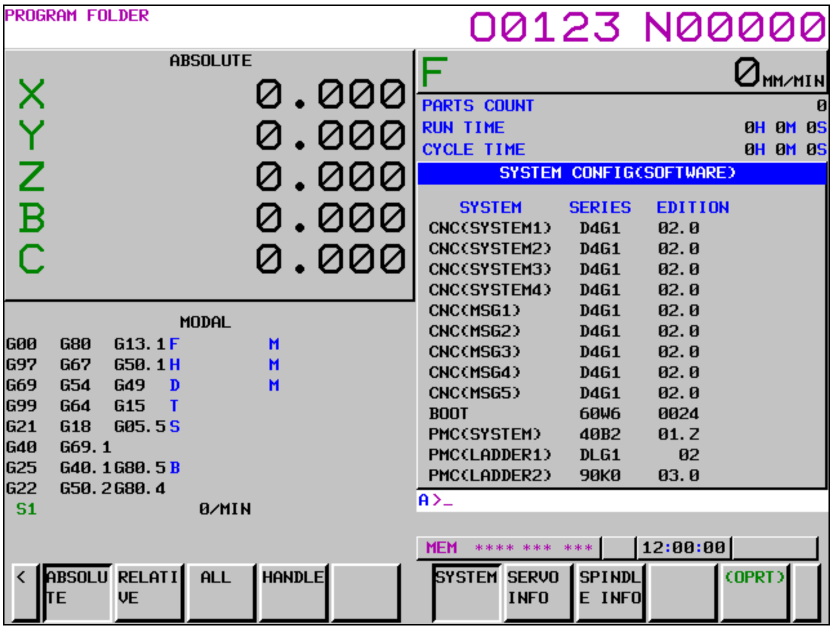
(2) ID-1 / ID-2

- Displays ID information.

(3) SLOT

- Displays the number of the slot in which the option board is inserted.

12.31.2 Software Configuration Screen



```
PROGRAM FOLDER                                   O0123 N00000
              ABSOLUTE               F                    0MM/MIN
X                 0.000              PARTS COUNT                0
Y                 0.000              RUN TIME            0H 0M 0S
Z                 0.000              CYCLE TIME          0H 0M 0S
B                 0.000                SYSTEM CONFIG(SOFTWARE)
C                 0.000                  SYSTEM      SERIES   EDITION
                                       CNC(SYSTEM1)  D4G1     02.0
                  MODAL                CNC(SYSTEM2)  D4G1     02.0
G00  G80  G13.1F        M              CNC(SYSTEM3)  D4G1     02.0
G97  G67  G50.1H        M              CNC(SYSTEM4)  D4G1     02.0
G69  G54  G49  D        M              CNC(MSG1)     D4G1     02.0
G99  G64  G15  T                       CNC(MSG2)     D4G1     02.0
G21  G18  G05.5S                       CNC(MSG3)     D4G1     02.0
G40  G69.1                             CNC(MSG4)     D4G1     02.0
G25  G40.1G80.5B                       CNC(MSG5)     D4G1     02.0
G22  G50.2G80.4                        BOOT          60W6     0024
 S1              0/MIN                 PMC(SYSTEM)   40B2     01.Z
                                       PMC(LADDER1)  DLG1       02
                                       PMC(LADDER2)  90K0     03.0
                                      A>_

                                      MEM **** *** ***   12:00:00
```

Fig. 12.31.2 (a) Software configuration screen

(1) SYSTEM
Software type

(2) SERIES
Software series

(3) EDITION
Software edition

12.32 HELP SCREEN

Detailed information about alarms issued in the CNC and how to operate the CNC is displayed on the screen. The initial menu screen, alarm detail screen, operation method screen, and parameter table screen are provided.

12.32.1 Initial Menu Screen

This screen shows a list of help functions. From the initial menu screen, you can display the alarm detail screen, operation method screen, and parameter table screen.

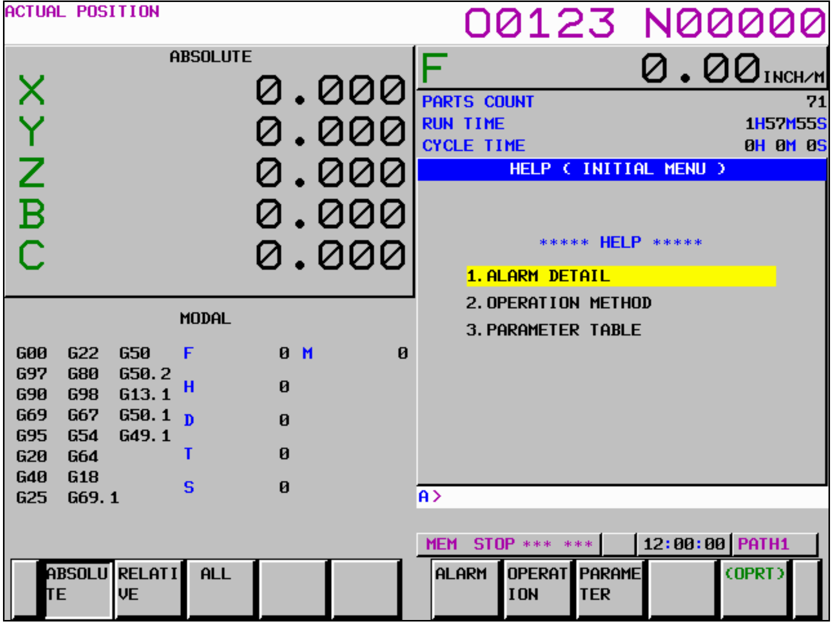
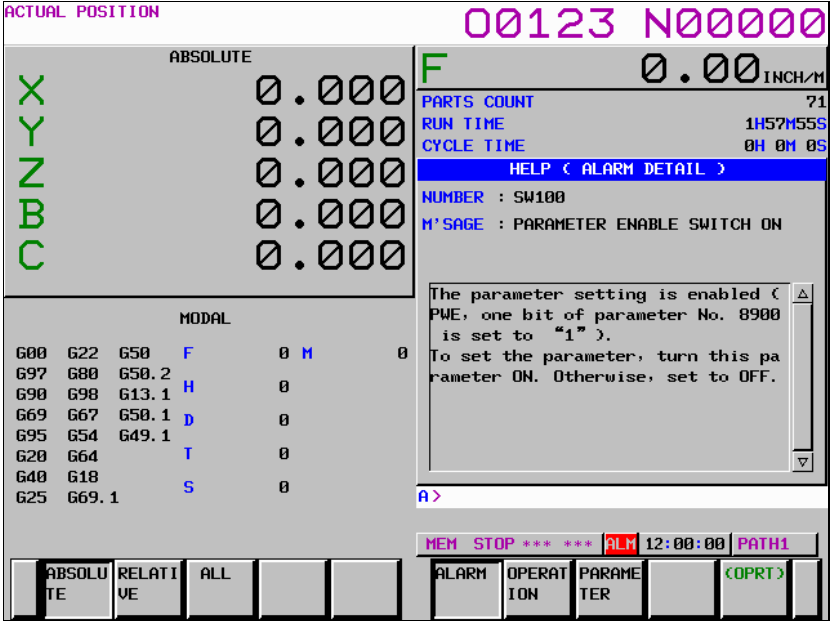


Fig. 12.32.1 (a) Initial Menu Screen

12.32.2 Alarm Detail Screen

This screen provides detailed information about alarms as reference information. The causes of alarms and the methods of releasing them are displayed.



NUMBER : SW100

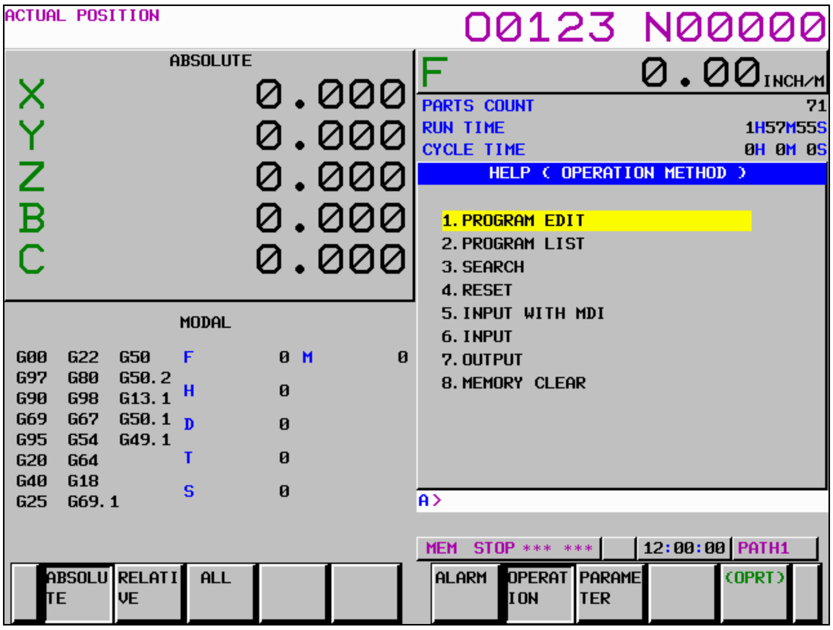
M'SAGE : PARAMETER ENABLE SWITCH ON

The parameter setting is enabled (PWE, one bit of parameter No. 8900 is set to "1").
To set the parameter, turn this parameter ON. Otherwise, set to OFF.

Fig. 12.32.2 (a) Alarm detail screen

12.32.3 Operation Method Screen

When you are uncertain about how to operate the CNC, see the operation method screen on which a list of operations is indicated.

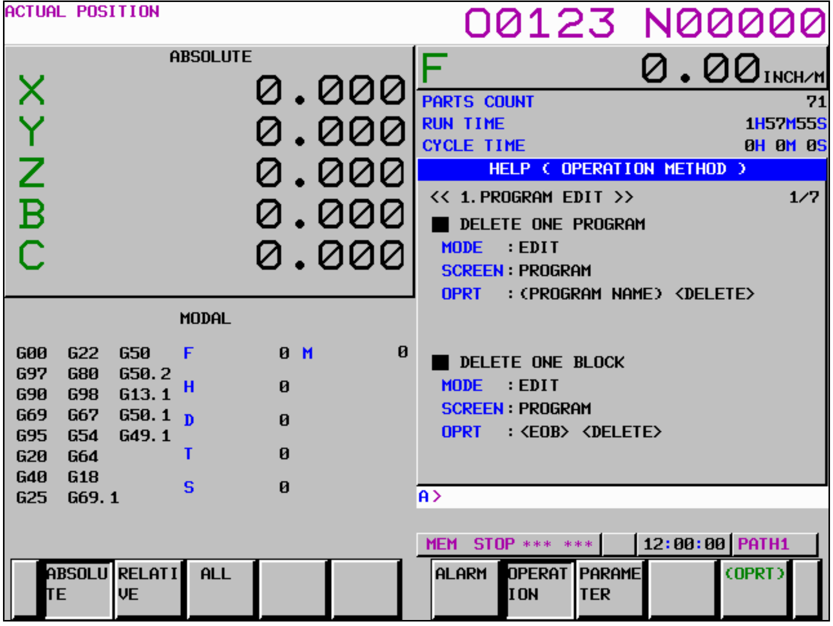


HELP (OPERATION METHOD)

1. PROGRAM EDIT
2. PROGRAM LIST
3. SEARCH
4. RESET
5. INPUT WITH MDI
6. INPUT
7. OUTPUT
8. MEMORY CLEAR

Fig. 12.32.3 (a) Operation method screen

The operation method for an item selected on the operation method screen is displayed.



HELP (OPERATION METHOD)

<< 1. PROGRAM EDIT >> 1/7

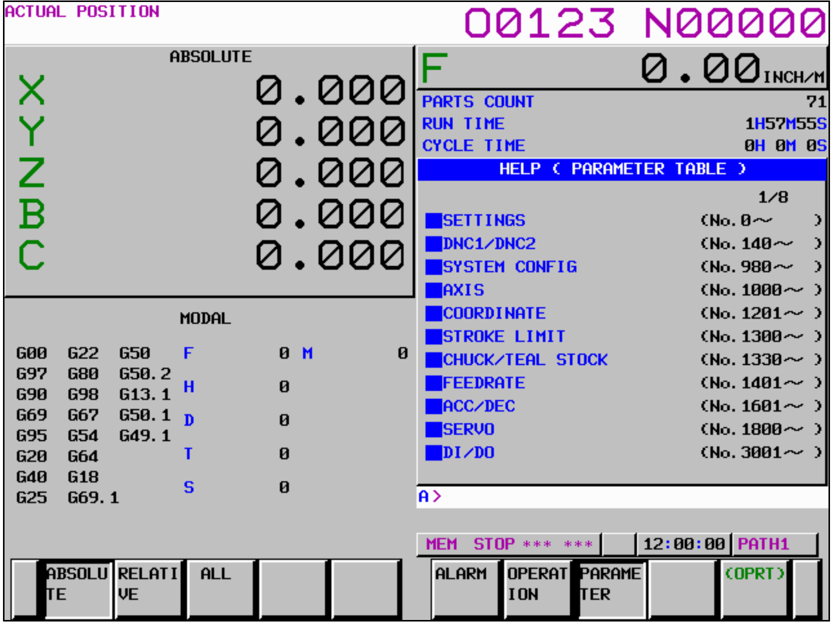
■ DELETE ONE PROGRAM
MODE : EDIT
SCREEN : PROGRAM
OPRT : (PROGRAM NAME) <DELETE>

■ DELETE ONE BLOCK
MODE : EDIT
SCREEN : PROGRAM
OPRT : <EOB> <DELETE>

Fig. 12.32.3 (b) Operation information screen

12.32.4 Parameter Table Screen

When you are uncertain about the number of the system parameter you want to set or reference, see the parameter table screen on which a list of parameter numbers is indicated for each function.

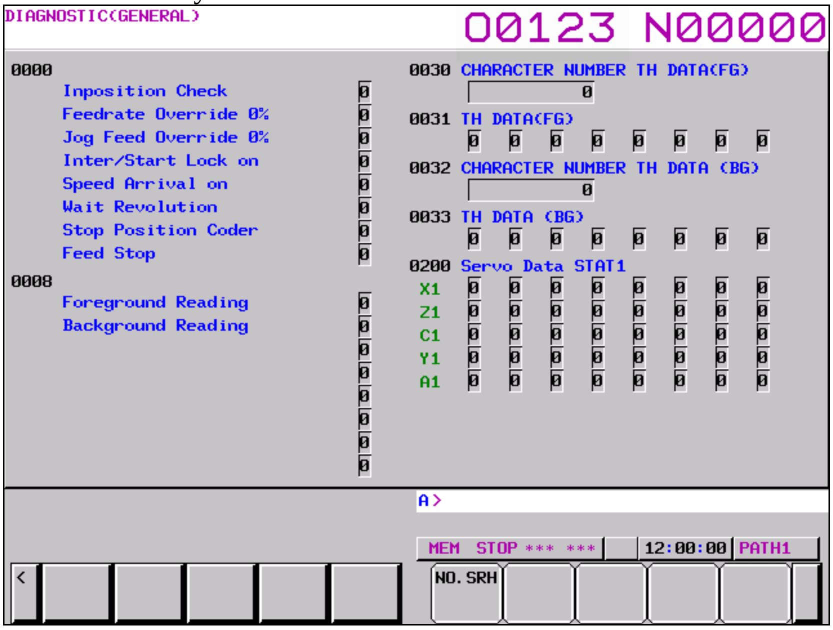


```
ACTUAL POSITION                                   O0123 N00000
              ABSOLUTE                  F                0.00INCH/M
X                          0.000        PARTS COUNT                    71
Y                          0.000        RUN TIME                 1H57M55S
Z                          0.000        CYCLE TIME                0H 0M 0S
B                          0.000              HELP ( PARAMETER TABLE )
C                          0.000                                     1/8
                                        SETTINGS            (No.0~    )
                                        DNC1/DNC2           (No.140~  )
                                        SYSTEM CONFIG       (No.980~  )
                                        AXIS                (No.1000~ )
                MODAL                   COORDINATE          (No.1201~ )
                                        STROKE LIMIT        (No.1300~ )
G00  G22  G50    F          0 M      0  CHUCK/TEAL STOCK    (No.1330~ )
G97  G80  G50.2                         FEEDRATE            (No.1401~ )
G90  G98  G13.1  H          0           ACC/DEC             (No.1601~ )
G69  G67  G50.1  D          0           SERVO               (No.1800~ )
G95  G54  G49.1                         DI/DO               (No.3001~ )
G20  G64         T          0
G40  G18                                A>
G25  G69.1       S          0
                                        MEM STOP *** ***   12:00:00 PATH1
```

Fig. 12.32.4 (a) Parameter table screen

12.33 SELF-DIAGNOSIS SCREEN

The current internal status of the system is indicated.

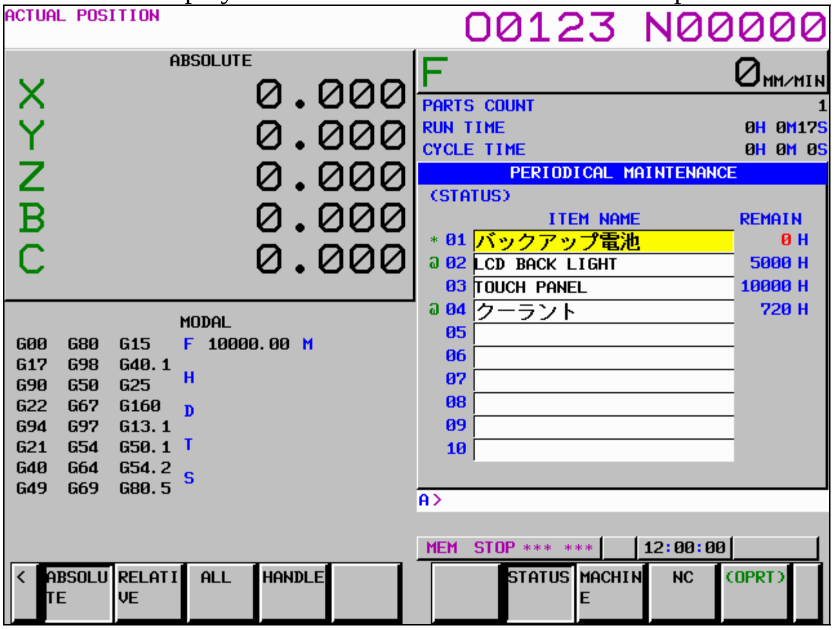


```
DIAGNOSTIC(GENERAL)                               O0123 N00000
0000                                    0030 CHARACTER NUMBER TH DATA(FG)
     Inposition Check            0                          0
     Feedrate Override 0%        0      0031 TH DATA(FG)
     Jog Feed Override 0%        0           0  0  0  0  0  0  0  0
     Inter/Start Lock on         0      0032 CHARACTER NUMBER TH DATA (BG)
     Speed Arrival on            0                          0
     Wait Revolution             0      0033 TH DATA (BG)
     Stop Position Coder         0           0  0  0  0  0  0  0  0
     Feed Stop                   0      0200 Servo Data STAT1
0008                                     X1  0  0  0  0  0  0  0  0
     Foreground Reading          0       Z1  0  0  0  0  0  0  0  0
     Background Reading          0       C1  0  0  0  0  0  0  0  0
                                 0       Y1  0  0  0  0  0  0  0  0
                                 0       A1  0  0  0  0  0  0  0  0
                                 0
                                 0
                                 0
                                 0
                                        A>
                                        MEM STOP *** ***   12:00:00 PATH1
```

Fig. 12.33 (a) Diagnosis screen

12.34 PERIODIC MAINTENANCE SCREEN

On the periodic maintenance screen, you can check whether the service lives of consumables that require periodic replacement (for example, the backup battery, the back light and touch panel of the LCD unit, etc.) have expired or not from displayed information such as the machine operation time.



X 0.000
Y 0.000
Z 0.000
B 0.000
C 0.000

G00 G80 G15 F 10000.00 M
G17 G98 G40.1 H
G90 G50 G25 D
G22 G67 G160 T
G94 G97 G13.1 S
G21 G54 G50.1
G40 G64 G54.2
G49 G69 G80.5

F 0MM/MIN
PARTS COUNT 1
RUN TIME 0H 0M17S
CYCLE TIME 0H 0M 0S

(STATUS)

	ITEM NAME	REMAIN
* 01	バックアップ電池	0 H
@ 02	LCD BACK LIGHT	5000 H
03	TOUCH PANEL	10000 H
@ 04	クーラント	720 H
05		
06		
07		
08		
09		
10		

Fig. 12.34 (a) periodic maintenance screen

12.35 SERVO AND SPINDLE INFORMATION SCREENS

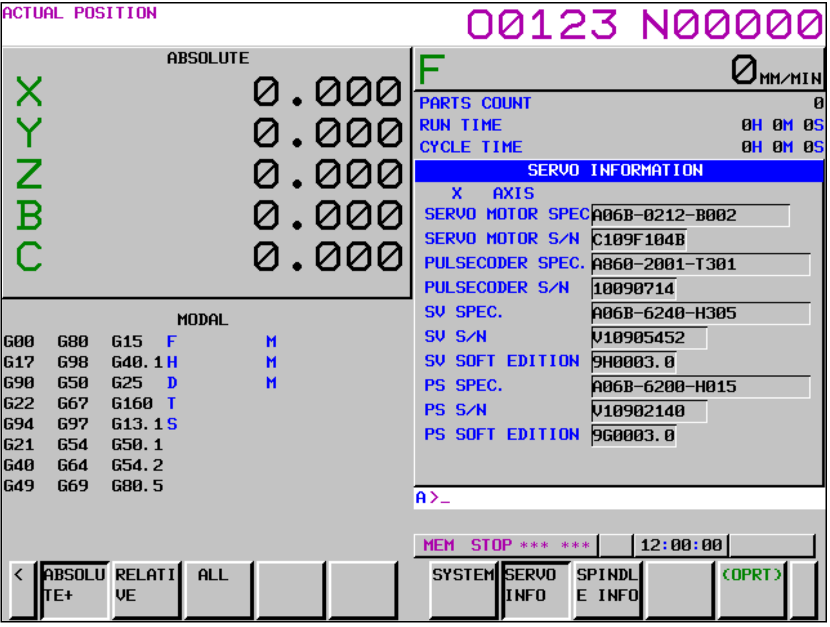
In the αi servo system and αi spindle system , ID information output from each of the connected units is obtained and output to the CNC screen.

ID information is automatically read from each of the connected units during first startup of the CNC and then recorded. During the second or later startup, the ID information recorded during first startup can be compared with the ID information read this time on the screen to check whether the configuration of the connected units is changed. (If there is a difference between them, the alarm mark (*) appears.)

The recorded ID information can be edited. Therefore, the ID information of an unit that does not have ID information can be displayed. (However, the alarm mark (*) indicating a difference between these IDs appears.)

12.35.1 Servo Information Screen

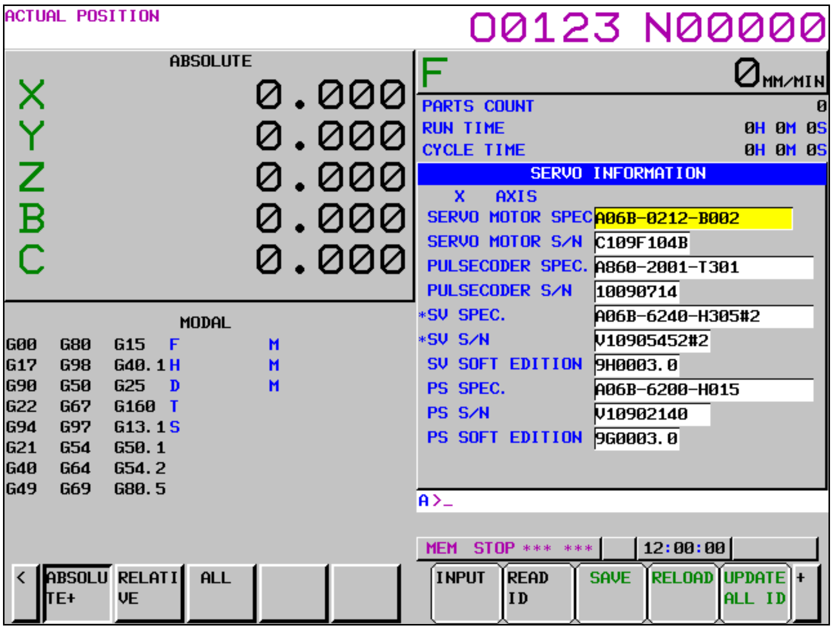
- Displaying the servo information screen



```
ACTUAL POSITION                     O0123 N00000
        ABSOLUTE                    F          0 MM/MIN
X            0.000                  PARTS COUNT            0
Y            0.000                  RUN TIME        0H 0M 0S
Z            0.000                  CYCLE TIME      0H 0M 0S
B            0.000                      SERVO INFORMATION
C            0.000                    X  AXIS
                                    SERVO MOTOR SPEC  A06B-0212-B002
          MODAL                     SERVO MOTOR S/N   C109F104B
G00  G80  G15   F        M          PULSECODER SPEC.  A860-2001-T301
G17  G98  G40.1 H        M          PULSECODER S/N    10090714
G90  G50  G25   D        M          SV SPEC.          A06B-6240-H305
G22  G67  G160  T                   SV S/N            V10905452
G94  G97  G13.1 S                   SV SOFT EDITION   9H0003.0
G21  G54  G50.1                     PS SPEC.          A06B-6200-H015
G40  G64  G54.2                     PS S/N            V10902140
G49  G69  G80.5                     PS SOFT EDITION   9G0003.0
                                    A>_
                                    MEM  STOP *** ***   12:00:00
```

Fig. 12.35.1 (a) Servo information screen

- Editing the servo information screen

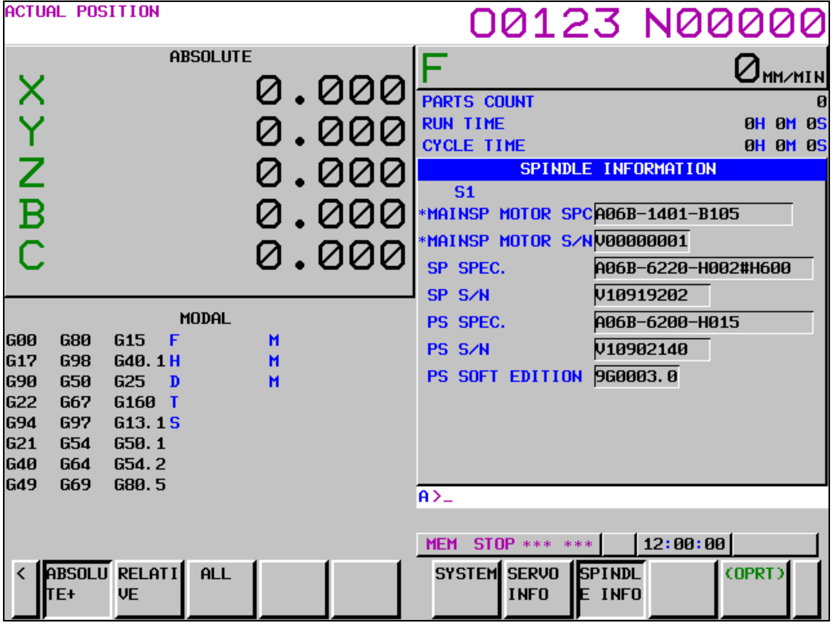


```
ACTUAL POSITION                     O0123 N00000
        ABSOLUTE                    F          0 MM/MIN
X            0.000                  PARTS COUNT            0
Y            0.000                  RUN TIME        0H 0M 0S
Z            0.000                  CYCLE TIME      0H 0M 0S
B            0.000                      SERVO INFORMATION
C            0.000                    X  AXIS
                                    SERVO MOTOR SPEC  A06B-0212-B002
          MODAL                     SERVO MOTOR S/N   C109F104B
G00  G80  G15   F        M          PULSECODER SPEC.  A860-2001-T301
G17  G98  G40.1 H        M          PULSECODER S/N    10090714
G90  G50  G25   D        M         *SV SPEC.          A06B-6240-H305#2
G22  G67  G160  T                  *SV S/N            V10905452#2
G94  G97  G13.1 S                   SV SOFT EDITION   9H0003.0
G21  G54  G50.1                     PS SPEC.          A06B-6200-H015
G40  G64  G54.2                     PS S/N            V10902140
G49  G69  G80.5                     PS SOFT EDITION   9G0003.0
                                    A>_
                                    MEM  STOP *** ***   12:00:00
```

Fig. 12.35.1 (b) Servo information screen (Editing)

12.35.2 Spindle Information Screen

- Displaying the spindle information screen



SPINDLE INFORMATION

S1

*MAINSP MOTOR SPC A06B-1401-B105

*MAINSP MOTOR S/N V00000001

SP SPEC. A06B-6220-H002#H600

SP S/N V10919202

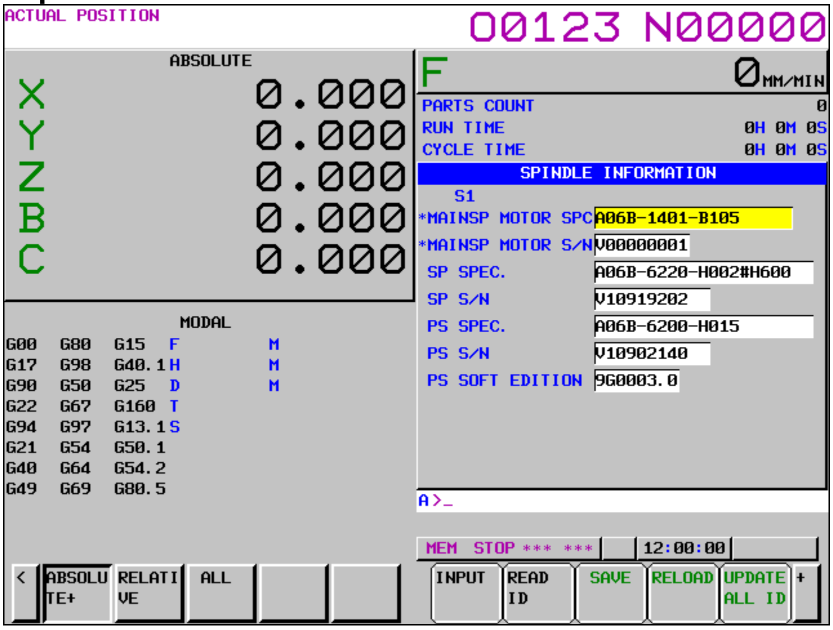
PS SPEC. A06B-6200-H015

PS S/N V10902140

PS SOFT EDITION 9G0003.0

Fig. 12.35.2 (a) Spindle information screen

- Editing the spindle information screen



SPINDLE INFORMATION

S1

*MAINSP MOTOR SPC A06B-1401-B105

*MAINSP MOTOR S/N V00000001

SP SPEC. A06B-6220-H002#H600

SP S/N V10919202

PS SPEC. A06B-6200-H015

PS S/N V10902140

PS SOFT EDITION 9G0003.0

Fig. 12.35.2 (b) Spindle information screen (Editing)

12.36 GRAPHIC DISPLAY

The tool path of a program during machining can be drawn. So, the progress of machining and the current tool position can be checked. The following functions are available:

- The current tool position in the workpiece coordinate system is displayed.
- Graphic coordinates can be set freely.
- Rapid traverse and cutting feed can be drawn using a different color for each.
- The values of F, S, and T in the program during drawing are displayed.
- Graphic enlargement or reduction is possible.

12.37 DYNAMIC GRAPHIC DISPLAY

The dynamic graphic display function has two features:

- Path Drawing
  The path of coordinates specified in a program is drawn on the screen. By displaying a travel path on the screen, the path can be checked easily before performing machining actually.
- Animation
  The figure of a workpiece to be machined by a programmed tool movement is drawn. By drawing the 3-dimensional figure of a workpiece to be machined in an animation-like way, the intermediate machining process and final figure can be grasped easily.

When machining is performed according to a program, this function can draw a tool path with another program. This function performs drawing much faster than the graphic display function based on automatic operation, so that a program check can be made more swiftly.

NOTE

This function is an optional function.

12.38 TOUCH PANEL CONTROL

A display unit with a touch panel enables you to operate soft keys by touching the screen.

Moreover, an application using a touch panel can be created with the C Language Executor.

NOTE

1 This function is an optional function.
2 With a CNC of LCD-mounted type, RS-232C serial port 2 (JD36A) is occupied.
3 Touch panel pressing information is read at intervals of 32 msec.
4 A positional precision of ±2.5 mm is provided.

12.39 EXTERNAL TOUCH PANEL INTERFACE

External Touch Panel(called "ETP" below) of SNP-X protocol can be connected with CNC. ETP has functions that can read out/ write in from/to PMC such control signals as input signal(X), output signal(Y), internal relay(R), keep relay(K), data table(D), extra relay (E), timer(T), counter(C), and the function is almost the same as operating panel of machine.

The remarkable function of ETP is drawing function. Assignment between drawing and address(signal) can be specified freely. For example, the data in data table can be set with the switch on the screen which is designed to assign the setting of data table.

NOTE
This function is an optional function.

12.40 SPEED DISPLAY FUNCTION OF A MILLING TOOL WITH SERVO MOTOR

Any servo motor axis can be selected to display its speed considering gear ratio.

12.41 SYSTEM ALARM HISTORY

Up to two previous system alarms can be stored and displayed on the system alarm history screen. In addition, system alarm information can be output to an external device.

```
SOFTWARE INFORMATION 1
  CONTROL REGISTERS
     8010C000 00003030 111EED50 00003930 44000200 20000000 111EEC4C 102A6F44
     70000202 16062840 42000000 820040D0 00000000 00000000 00000000 00000000
     00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
  GENERAL PURPOSE REGISTERS
     111EEC4C 10901DD0 103FB820 00000000 0000001D 10F65B3C 10F65B24 00000003
     10901D54 00000002 00380000 00000000 00000000 108B8F54 00000001 10F42D84
     00000001 10F42AFC 14400000 16062840 00000000 10EA3788 10F42AFC 108B0FF4
     10F42C84 108B0FF4 00380000 10EA3928 FFFFFFFF 108F29B0 108B0FF4 10EA3788
  FLOATING POINT REGISTERS
     00000A00 FFF80000 00000000 00000000 00000000 41E00000 00000000 00000000
     00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
     00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
     00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
     00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
     00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
     00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
     00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
```

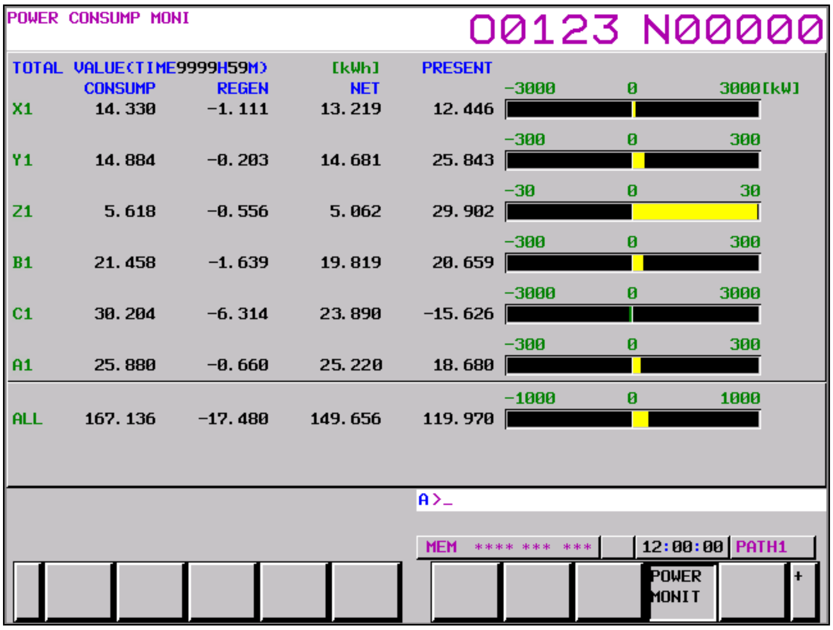
Fig. 12.41 (a) System alarm history screen

12.42 ROBOT CONNECTION FUNCTION

For a machine tool or machining line with a robot, this function allows the operator to check the status of the robot on the CNC screen without entering the inside of the safety fence for the robot. This function also allows the operator to change both robot and CNC programs on the CNC screen.
For details, refer to the Robot and Machine Tool Integration Function OPERATOR’S MANUAL (B-75114EN).

12.43 POWER CONSUMPTION MONITOR

The electric power consumption and integral power consumption of servo motor and spindle motor are stored to diagnosis data. User application can use these data via PMC window or FOCAS2 function. Moreover, these values can be confirmed on the CNC screen.



TOTAL VALUE(TIME9999H59M) [kWh]

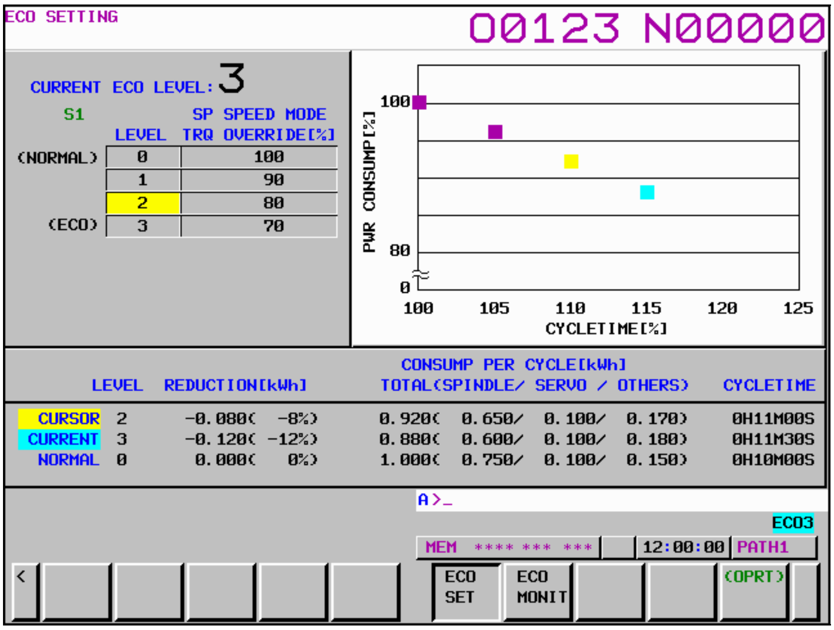
	CONSUMP	REGEN	NET	PRESENT
X1	14.330	-1.111	13.219	12.446
Y1	14.884	-0.203	14.681	25.843
Z1	5.618	-0.556	5.062	29.902
B1	21.458	-1.639	19.819	20.659
C1	30.204	-6.314	23.890	-15.626
A1	25.880	-0.660	25.220	18.680
ALL	167.136	-17.480	149.656	119.970

Fig.12.43 (a) Power consumption monitoring screen

12.44 ENERGY SAVING LEVEL SELECTING FUNCTION

Suppressing torque at acceleration/deceleration in spindles, where there is large power consumption, can reduce whole power consumption of a machine, though machining time gets longer.

Energy saving level selecting function makes it possible to switch between machining with shorter time and that with less power consumption. Effects of energy saving can be observed on Eco setting screen and Eco monitoring screen.

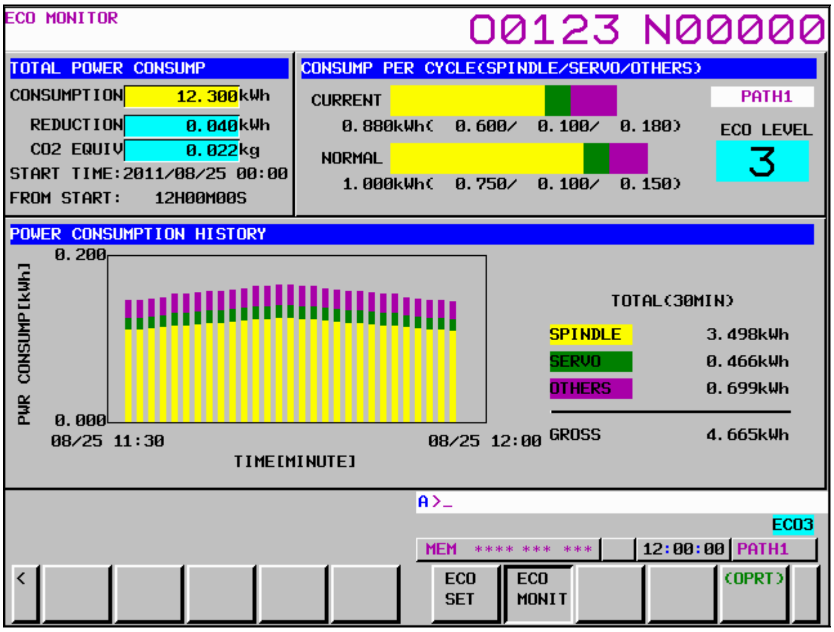


CURRENT ECO LEVEL:3

S1	LEVEL	SP SPEED MODE TRQ OVERRIDE[%]
(NORMAL)	0	100
	1	90
	2	80
(ECO)	3	70

	LEVEL	REDUCTION[kWh]	CONSUMP PER CYCLE[kWh] TOTAL(SPINDLE/ SERVO / OTHERS)	CYCLETIME
CURSOR	2	-0.080(-8%)	0.920(0.650/ 0.100/ 0.170)	0H11M00S
CURRENT	3	-0.120(-12%)	0.880(0.600/ 0.100/ 0.180)	0H11M30S
NORMAL	0	0.000(0%)	1.000(0.750/ 0.100/ 0.150)	0H10M00S

Fig.12.44 (a) Eco setting screen



CONSUMPTION 12.300kWh
REDUCTION 0.040kWh
CO2 EQUIV 0.022kg
START TIME:2011/08/25 00:00
FROM START: 12H00M00S

CURRENT
0.880kWh(0.600/ 0.100/ 0.180)
NORMAL
1.000kWh(0.750/ 0.100/ 0.150)

ECO LEVEL 3

TOTAL(30MIN)

SPINDLE	3.498kWh
SERVO	0.466kWh
OTHERS	0.699kWh
GROSS	4.665kWh

Fig.12.44 (b) Eco monitoring screen

NOTE

This function is an optional function.

12.45 MACHINE STATE MONITORING FUNCTION

This function monitors state of machine.
When trouble of the machine occurs, CNC information such as the operation history, the position and the feedrate at that time is saved in the CNC memory.
The saved CNC information can be used to investigate the cause of machine trouble.

Unexpected disturbance torque detection function, servo/spindle alarm, and a ladder program made by machine tool builders monitor the state of machine.
In the ladder program, the shock value is monitored by the Multi-Sensor Unit (MSU).When trouble of the machine such as the spindle collision occurs, the ladder program notifies CNC that the trouble has occurred.

When CNC is notified the trouble occurrence, CNC saves the information on that time in the CNC memory.
Machine tool builders or maintenance members investigate the cause of generation of trouble by using the saved CNC information.
The maximum save number of CNC information is 100. So, the past trouble can be investigated.

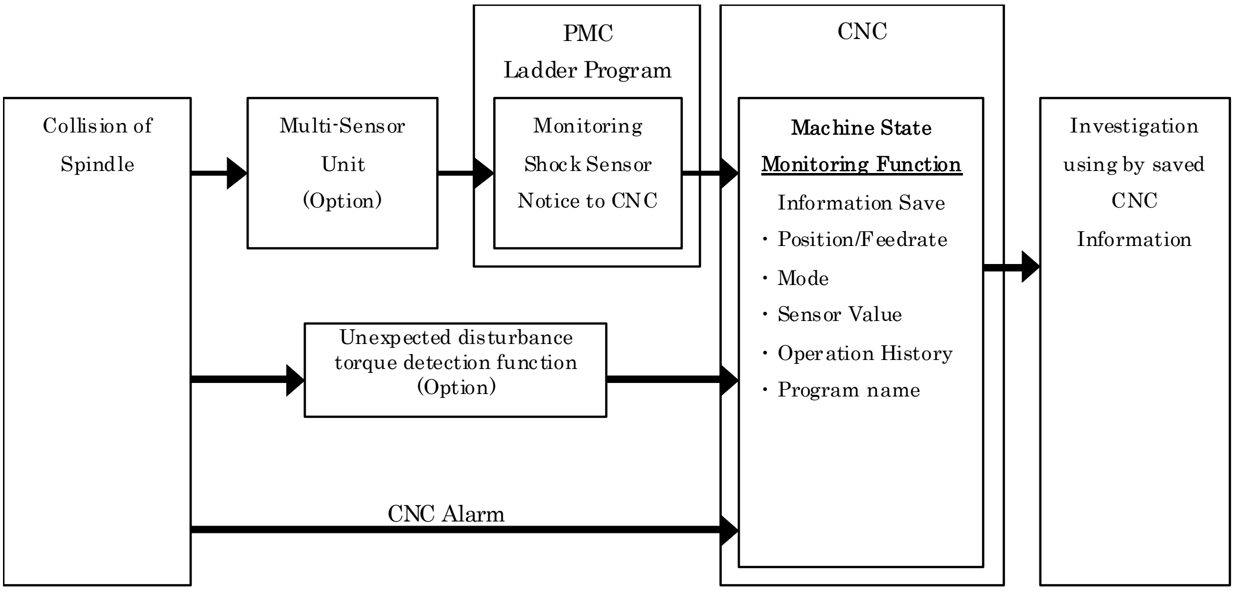


Fig.12.45 Summary of machine state monitoring function (Ex. collision of spindle)

Machine state monitoring function has the following two screens.
(1) Machine state monitoring screen
(2) Machine state history screen

In the machine state monitoring screen, four bytes PMC signals that set by parameter and the signal of the Multi-Sensor Unit can be monitored.

In the machine state history screen, saved CNC state and signal value can be displayed. So, the signal of Multi-Sensor Unit and the CNC state such as position and program at the time when a spindle collides can be checked.
In addition, displayed information can be output to external devices.

When CNC saved information exceeded 100, CNC information is overwritten from the old one. CNC Information is remained for eight days or more, so saving information is restricted.

NOTE

1 This function is an optional function.
2 Multi-Sensor Unit and unexpected disturbance torque detection function are the optional functions.
Please refer to the following manuals for these functions.
  - Multi-Sensor Unit:
    CONNECTION MANUAL (HARDWARE) B-64603EN
  - Unexpected Disturbance Torque Detection Function:
    CONNECTION MANUAL (FUNCTION) B-64603EN-1
    “2 PREPARATIONS FOR OPERATION 2.9UNEXPECTED DISTURBANCE TORQUE DETECTION FUNCTION”

NOTE

Multi-Sensor Unit and unexpected disturbance torque detection function are the optional functions.
Please refer to the following manuals for these functions.

- Multi-Sensor Unit:
  CONNECTION MANUAL (HARDWARE) B-64603EN
- Unexpected Disturbance Torque Detection Function:
  CONNECTION MANUAL (FUNCTION) B-64603EN-1
  “2 PREPARATIONS FOR OPERATION 2.9UNEXPECTED DISTURBANCE TORQUE DETECTION FUNCTION”

12.46 SERVO/SPINDLE WAVEFORM DATA OUTPUT FUNCTION

When something trouble occurs with machine, servos and spindles waveform data before and after the trouble is automatically output to Data Server, which is sampled by Trouble diagnosis function.
Since long-termed data is automatically saved, it is easier to detect the cause of the trouble.

CNC always samples waveform data.
If something trouble occurs with machine, CNC detects the trouble through alarms or signals and output waveform data before and after the trouble to Data Server automatically.

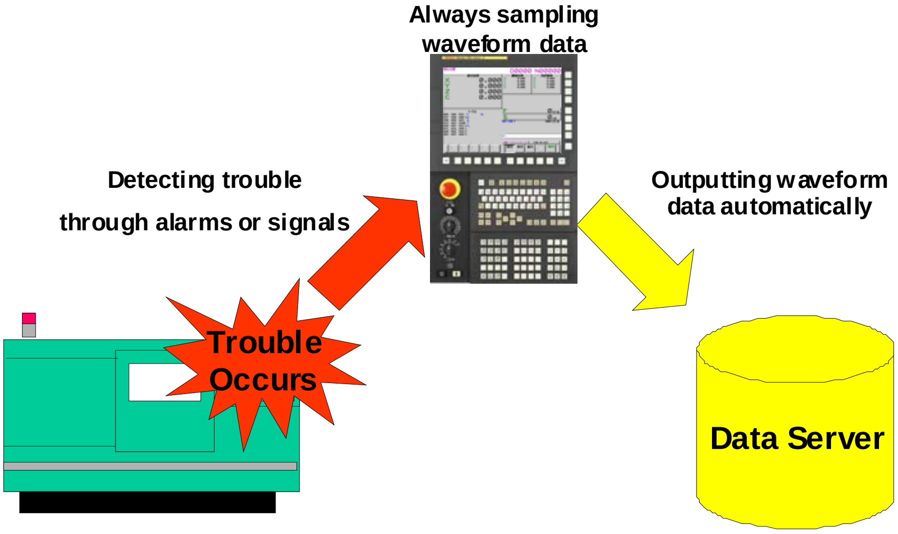


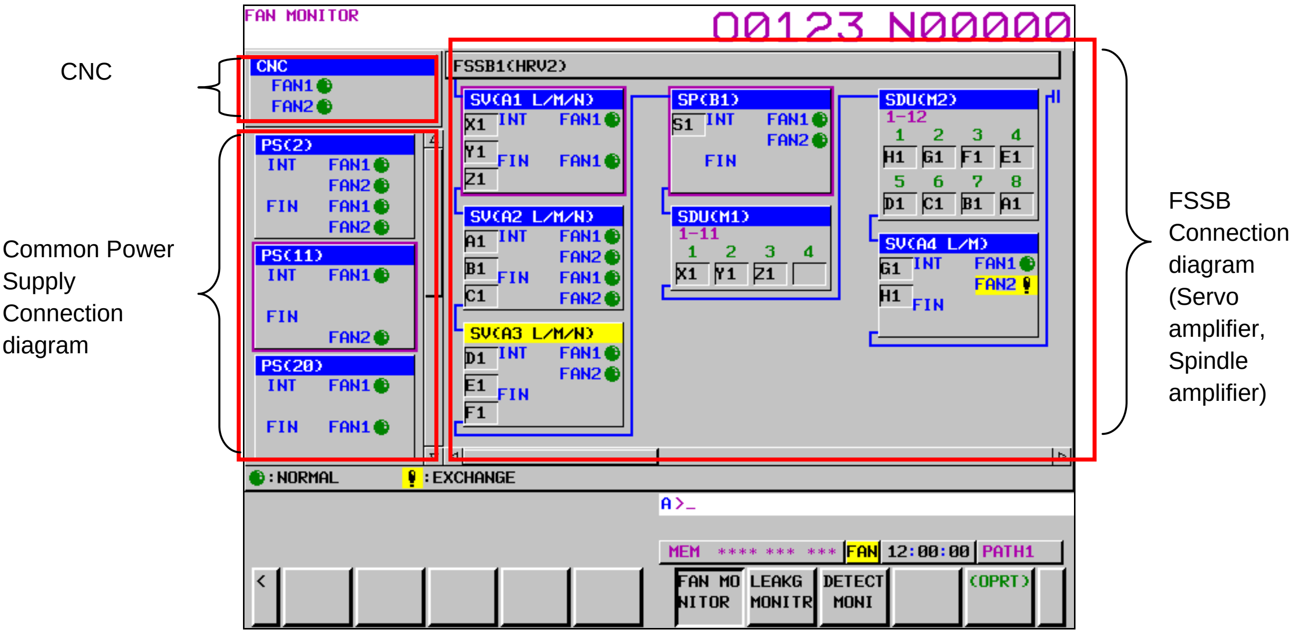
Fig.12.46 (a) Outline of Servo / spindle waveform data output function

12.47 MAINTENANCE MONITOR

Status of CNC, amplifiers and power supplies necessary for maintenance can be confirmed on this screen.

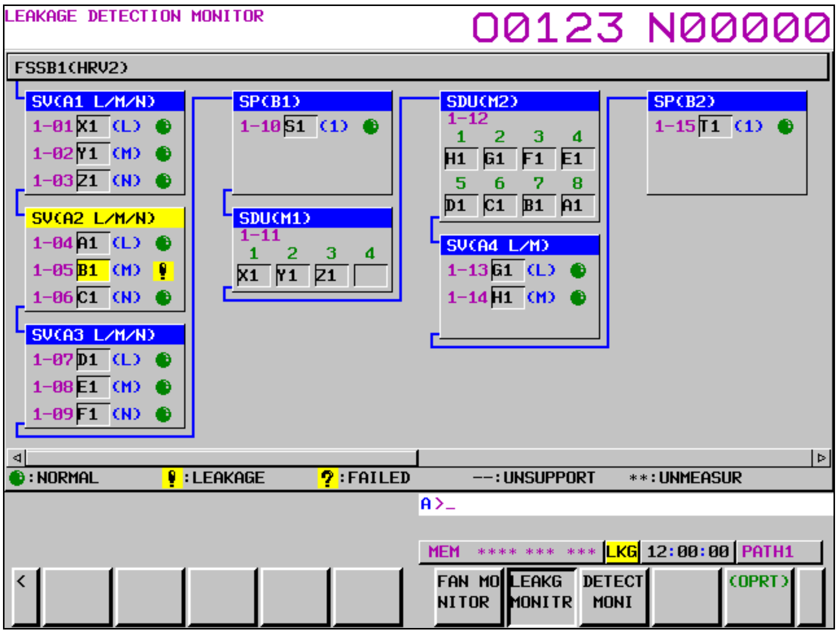
12.47.1 Fan Monitor Screen

This screen is possible to confirm it bringing the state of the fan of CNC, servo amplifier, spindle amplifier, and power supply. Confirm this screen when "FAN" is displayed in status display, and examine which fan is abnormal.



12.47.2 Leakage Detection Monitor Screen

The leakage status of servo amplifier and spindle amplifier can be confirmed on this screen. When "LKG" is displayed in status display, confirm this screen and examine the part where axis fault occurred.



12.48 TROUBLE DIAGNOSIS

Investigating the cause of Servo/Spindle/CNC alarms becomes easier by diagnosis according to the guidance message of this function.

And when the thermal simulation or disturbance level of servo axis exceeds the trouble forecast level, the machine can be safely stopped by detecting the breakdown beforehand by the use of the trouble forecast signal.

12.49 MACHINE ALARM DIAGNOSIS

It is necessary to notify the operator by the alarm display and warning message in the machine tool when a peculiar problem to the machine such as the tool breakage and coolant shortage occurs.

The machine tool builder customizes by using the external alarm, the macro alarm, and the operator message function and a peculiar alarm and message to such a machine can be displayed.

In this function, the diagnostic information to specify the factor in addition can be added to the alarm and the message that the machine tool builder customized.

When a peculiar alarm to the machine etc. are occurred, the added diagnostic information is displayed on the guidance screen of the trouble diagnosis function and can be used to specify the cause.

13 DATA INPUT/OUTPUT

Chapter 13, "DATA INPUT/OUTPUT", consists of the following sections:

13.1 RS232C INTERFACE

The data shown below can be input/output through RS232C interface.

- Program
- Tool compensation value
- Parameter
- Pitch error compensation data
- Custom macro common variable
- Workpiece coordinate system setting data
- Operation history data (output only)
- Tool management data

Usually, the screen is switched according to the type of data to be input from or output to an external device; for example, a parameter screen is used for parameter input/output, and a program screen is used for program input/output. However, a single ALL I/O screen can be used to input and output programs, parameters, offset data, and macro variables.

13.2 DATA SERVER FUNCTION

The data server function performs machining by storing NC programs in the built-in memory card.

- Machining programs can be transferred between the personal computer and data server.
- Memory operation using macro statements and sub programs called from the built-in memory card of the data server can be performed. DNC operation from the personal computer is also possible.
- Programs stored in the built-in memory card of the data server can be edited.

While operation is performed using the data server operation, other Ethernet functions can be used.

NOTE

This function is an optional function.

13.3 BUFFER MODE OF DATA SERVER

In this mode, the host computer connected to the data server is selected as an external I/O device.
Unlike the FTP mode, the buffer mode allows the area on the built-in memory card of the data server to be used as an intermediate buffer. In the buffer mode, I/O (input/output) operations for an NC program are performed as if the storage mode were set.

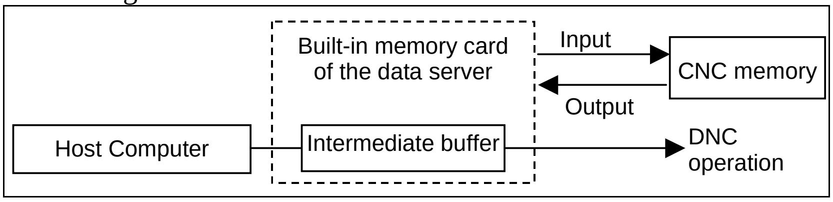


Fig. 13.3 (a)

> **NOTE**
> This function is included in the option "Data server function".

13.4 DATA SERVER EXPLORER CONNECTION

Data server Explorer connection extends the number of interfaces that can be connected concurrently from 5 to 10 when the fast data server is used as an FTP server.
This function makes FTP communication using Explorer on personal computers easy.
It also allows operations such as checking the FTP communication connection status and manual disconnection from the FTP server maintenance screen.

> **NOTE**
> This function is an optional function.
> To use this function, the both options for "Data server function" and this function are required.

13.5 EXTERNAL DATA INPUT

The external data input function sends data to the CNC from an external unit such as a machine to perform desired operation.
The external data input function includes the following capabilities:

- External tool compensation
- External program number search
- External workpiece coordinate system shift
- External machine zero point shift
- Extended external machine zero point shift
- External alarm message
- External operator message
- Assignment of machined parts count and required parts count data

13.5.1 External Tool Offset

These signals provide for changing the tool compensation value via the PMC. When the offset number is specified by a part program, data input from the PMC is added to the offset value. The offset value can also be used as input data itself by specifying the input signal.
If the tool compensation value is externally input when offset number 0 is specified in a part program (an offset cancel) in the lathe turning machine, the workpiece coordinate system shifts by the entered quantity.

The external tool offset range is ±79999999.
The unit and setting range are the same as the Tool offset.

13.5.2 External Program Number Search

A program number (1 to 9999) is specified from an extended source and is selected in the CNC memory.
Data for the external program number search is accepted regardless of CNC mode, but the search execution can be made only in the reset state in MEM mode.

13.5.3 External Workpiece Coordinate System Shift

In the external workpiece coordinate system shift, the shift value can be externally modified by the signal of PMC.
Each axis has this shift value (setting the parameter), and this shift value is added to all the workpiece coordinate systems in common.
The shift value is not lost by cutting off the power supply.
The shift value can not only be added to the current work coordinate system shift but also be substituted for the current value.
The value range is 0 to ±79999999.

13.5.4 External Machine Zero Point Shift

The machine coordinate system can be externally shifted by inputting a shift value.
When the shift value is input, compensation is immediately applied to the corresponding axis and the axis moves. The position accuracy can be improved by combining this function with sensors.
The specification to shift the axis is the same as the external workpiece coordinate system shift.
The compensation value is specified in signals ED0 to ED15 using a binary code ranging from 0 to ±9999. This compensation value must be specified in absolute value. The value which the machine actually moves at input is the difference from the previously stored value.
When a large value of compensation is applied at one time, an alarm such as "excessive error on stop" may occur. In this case, input the compensation in several smaller increments.

13.5.5 Extended External Machine Zero Point Shift

The conventional external machine zero point shift value function cannot make shifts on multiple axes simultaneously.
With this extended function, external machine zero point shifts can be performed on all controlled axes.
An external machine zero point shift value is to be set in a parameter-set R area. A shift value must be specified using a binary code, and the absolute value of a number from -32767 to 32767 must be specified.

13.5.6 External Alarm Message

By sending alarm number from PMC, the CNC is brought to an alarm status; an alarm message is sent to the CNC, and the message is displayed on the screen of the CNC.
Reset of alarm status is also done with external data.
Up to 4 alarm numbers and messages can be sent at a same time. Up to 32 characters can be sent in an alarm message.
Also the alarm is set separately for one.
It is available to select the range of alarm number and the display form by setting the parameter.

- When the parameter is set to 0.
  Alarm number 0 to 999 can be sent. To distinguish these alarms from other alarms, the CNC displays them by adding 1000 to an alarm number.
- When the parameter is set to 1.

Alarm number 0 to 4095 can be sent. The CNC displays them with prefix characters “EX” to an alarm number for display.

13.5.7 External Operator Message

The external operator message function allows transfer of messages for the operator from the outside to the CNC to display them on the CNC screen.
The operator messages can be cleared by external data.
Up to 4 message numbers can be sent at a same time. Up to 256 characters can be sent in an operator message.
It is available to select the range of message number and the display form by setting the parameter.

- When the parameter is set to 0.
  The message numbers 0 to 999 can be sent.
  The message numbers 0 to 99 are displayed along with the message.
  To distinguish these alarms from other alarms, the CNC displays them by adding 2000 to an alarm number.
  When a message from 100 to 999 is displayed, the message number is not displayed; only its text is displayed.
- When the parameter is set to 1.
  The message numbers 0 to 4095 can be sent.
  The message numbers 0 to 99 are displayed along with the message.
  The CNC displays them with prefix characters "EX" to an alarm for display.
  When a message number from 100 to 4095 is displayed, the message number is not displayed; only its text is displayed.

13.5.8 Assignment of Machined Parts Count and Required Parts Count

Each of the required parts count and machined parts count can be externally preset to a value in the range 0 to 9999.

13.6 EXTERNAL KEY INPUT (KEY INPUT FROM THE PMC)

By turning input signals from the PMC to CNC on and off, code signals corresponding to keys on the MDI panel are input from the PMC to CNC. This produces the same effect as when keys on the MDI panel are operated.
For example, the following application is found:
To store a machining position as a programmed command after moving the tool to that position by using the playback function (option), you had to type keys such as X, Y, Z, and INSERT. These key operations can be replaced by just one press of a switch on the machine operator's panel. Therefore, when the switch is pressed, the PMC can input code signals corresponding to the keys such as X, Y, Z, and INSERT to the CNC to produce the same effect as when key operations take place.

13.7 EXTERNAL WORKPIECE NUMBER SEARCH

By specifying a workpiece number from 0001 to 9999 through an external unit such as a machine, the program corresponding to the workpiece number can be selected.
A workpiece number is used as a program number without modification. For example, when workpiece number 12 is specified, program number O0012 is selected.

13.8 MEMORY CARD INPUT/OUTPUT

By using the memory card interface located on the left side of the LCD display unit, information written on the memory card can be read into the CNC, or data can be written to the memory card.
The following types of data can be input/output:
(1) Program
(2) Offset data
(3) Parameter
(4) Pitch error compensation data
(5) Custom macro common variable
(6) Workpiece coordinate system setting data
(7) Operation history data
(8) Tool management data and so on

If an attempt is made to use an existing file name when NC data, such as programs and parameters, are written to the memory card, it is possible to select whether to write the NC data to the existing file or cancel the attempt.
Whether to enable this function can be specified using the bit 1 (COW) of parameter No. 11308.
For detailed explanations about how to output data, refer to the OPERATOR'S MANUAL (Common to Lathe System/Machining Center System) (B-64604EN).

13.9 USB MEMORY INPUT/OUTPUT

By using the USB memory interface located on the left side of the LCD display unit, information written on a USB memory can be read into the CNC, or data can be written to the USB memory.
The following types of data can be input/output:
(1) Program
(2) Offset data
(3) Parameter
(4) Pitch error compensation data
(5) Custom macro common variable
(6) Workpiece coordinate system setting data
(7) Operation history data
(8) Tool management data, and so on

When a USB memory is used, the following functions that support memory cards cannot be used:
(1) DNC operation
(2) Schedule operation
(3) External sub program call (M198)
(4) Memory card program edit and operation

If an attempt is made to use an existing file name when NC data such as programs and parameters are written to a USB memory, it is possible to select whether to write the NC data to the existing file or cancel the attempt.
Whether to enable this function can be specified using bit 1 (COW) of parameter No. 11308.
For details of USB memory specifications, refer to the OPERATOR'S MANUAL (Common to Lathe System/Machining Center System) (B-64604EN).

13.10 SCREEN HARD COPY

The information displayed on the screen of the display unit can be output as bit-mapped format data to the memory card or USB memory. Note, however, that only freeze-screen information can be output.

Bit-mapped format data created with this function can be referenced using a device such as a personal computer.

13.11 POWER MATE CNC MANAGER

When the β amplifier (with I/O Link interface) is used as an additional (slave) axis of the CNC, the power mate CNC manager enables the display and setting of data from the CNC.

Up to eight slaves can be connected per I/O Link channel.

The power mate CNC manager supports the following functions:

(1) Current position display (absolute/machine coordinate)
(2) Parameter display, setting, I/O (memory card, program area on the CNC)
(3) Diagnosis display
(4) System configuration screen display
(5) Alarm display

13.12 EXTERNAL I/O DEVICE CONTROL

The registration or output of a program can be specified externally.

- Registration
  External input start signal EXRD can be used to register a program in the part program memory from an external input device with the background edit function.
- Output
  External output start signal EXWT can be used to output all programs registered in part program memory to an external output device with the background edit function.

13.13 ONE TOUCH MACRO CALL

By pressing a switch on the machine, the following three operations can be performed with minimum ladder modifications:

- Switching to the MEM mode
- Execution of macro programs stored in memory
- Return to the mode before execution. The program selected before execution is automatically selected.

This function is enabled only in the reset state. This means that this function cannot be used during automatic operation (during automatic operation halt and automatic operation stop periods as well).

NOTE

This function is an optional function.

13.14 AUTOMATIC DATA BACKUP

It is possible to back up data held in the CNC's FROM/SRAM by storing it automatically in the FROM, which requires no battery and to restore the baked-up data as required. If data is lost from the CNC due to unforeseen circumstances, this function can be used to restore the data easily. Also, it is possible to hold up to three occurrences of backup data. With this function, the CNC data can be quickly switched to a post-machine adjustment state or an arbitrary backup state.

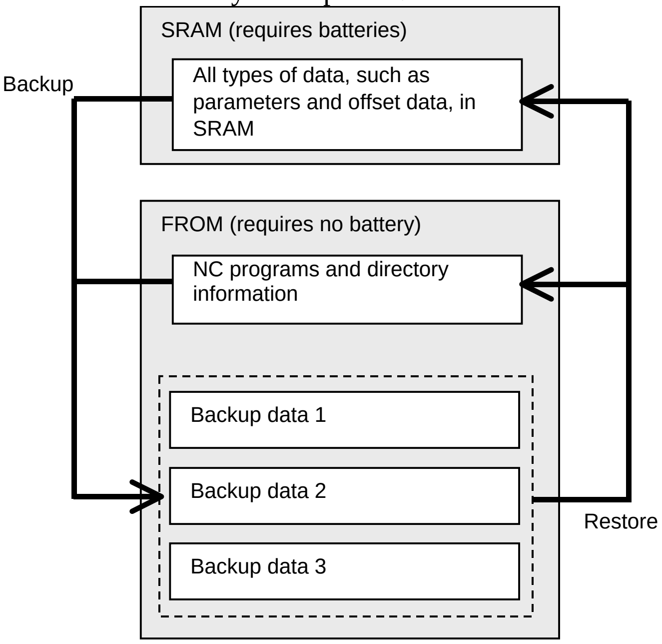


Fig. 13.14 (a) Flow chart when automatic data backup is performed

14 INTERFACE FUNCTION

Chapter 14, "INTERFACE FUNCTION", consists of the following sections:

14.1 EMBEDDED ETHERNET

The 100Mbps Embedded Ethernet is supported as standard. By connecting the CNC with a personal computer, NC programs can be transferred, the machine can be controlled, machine operation status can be monitored, and the machine can be adjusted and maintained.

- Using the FTP file transfer function in transferring NC programs
  By CNC screen operations, NC programs can be transferred. The personal computer uses FTP server software, so that NC programs can be transferred to and from a host computer operating not in the Windows environment.

- Using FOCAS2/Ethernet function in controlling and monitoring machines
  User-specific application software that controls and monitors the machine can be created by using the FOCAS2/Ethernet function.
  In addition, the unsolicited messaging function enables the CNC to send messages (CNC/PMC data) voluntarily to an application on the PC in response to a command from an NC program or a ladder program.

- Machine adjustment, maintenance ladder program maintenance, and servo motor adjustment using FANUC LADDER-III and the servo guide can be performed online.
  In addition, by using the CNC Screen Display function, allows performing display and operation similar to a CNC on the Windows PC.

- By using the CNC FTP transfer function, data can be transferred using an application of the C Language Executor.

- CNC screen Web server function
  The same screen on a standard CNC Display unit can be displayed on a Web browser of a Tablet-type device that connects with CNC through Ethernet. This function enables screen display and screen switch of CNC on a Web browser.

- CNC Status Notification function
  An e-mail of the information about CNC alarm status can be delivered to the e-mail software of a portable terminal or a personal computer within Intranet.

- DNC operation
  The DNC operation can also be performed while NC programs are being FTP-transferred from the PC.

NOTE
1 DNC operation cannot be performed for multiple paths simultaneously.
2 DNC operation using the FOCAS2/Ethernet function cannot be performed.

14.2 FAST ETHERNET / FAST DATA SERVER

To use Fast Ethernet / Fast data server, an option board is required.

- Using FOCAS2/Ethernet function in controlling and monitoring machines
  User-specific application software that controls and monitors the machine can be created by using the FOCAS2/Ethernet function.
  In addition, the unsolicited messaging function enables the CNC to send messages (CNC/PMC data) voluntarily to an application on the PC in response to a command from an NC program or a ladder program.
  Fast Ethernet uses a dedicated CPU on the option board to perform communication processing, so it enables high-speed and simultaneous data transfer to and from more than one personal computer.
  This function is suitable for building a production system that exchanges information with the machining lines and host computer of a plant.

- Using the FTP file transfer function in transferring NC programs
  By CNC screen operations, NC programs can be transferred. The personal computer uses FTP server software, so that NC programs can be transferred to and from a host computer operating not in the Windows environment.

- Data Server function
  A large program such as a program for molding can be stored on the built-in memory card of the fast data server to allow high-speed machining. While the data server is operating, another Ethernet function can be used at the same time.

  High-speed transfer

  Machining programs can be transferred between a personal computer and the data server at a high speed. CNC parameters and tool information files can also be sent and received.

  Memory operation

  Memory operation that calls macro statements and subprograms from the built-in memory card of the fast data server can be performed.

  DNC operation

  The DNC operation of an NC program stored on the built-in memory card of the fast data server can be performed. In addition, the DNC operation of an NC program stored in the PC can also be performed.

NOTE

DNC operation cannot be performed for multiple paths simultaneously.

Program editing

NC programs stored on the built-in memory card of the fast data server can be edited.

14.2.1 Functional Differences between the Embedded Ethernet and the Fast Ethernet

There are the following differences between the embedded Ethernet function and Fast Ethernet function using an option board.

Table 14.2.1 (a)

	Embedded Ethernet	Fast Ethernet
FOCAS2/Ethernet function (Note 1)	Available	Available
DNC operation (Note 2)	Available	Available
FTP file transfer function	Available	Available
Data Server function	Not available	Available
CNC screen display function (Note 3)	Available	Available
Machine remote diagnosis function (Note 3)	Available	Available
Unsolicited messaging function	Available	Available
DNS/DHCP client function	Available	Available
CNC screen Web server function	Available	Not available
CNC Status Notification function	Available	Not available
Modbus/TCP Server function (Note 4)	Available	Available
Industrial Ethernet (Except Modbus/TCP Server function)	Not available	Available

NOTE

1 With the embedded Ethernet function, the number of FOCAS2/Ethernet clients that can be connected concurrently is smaller as compared with the Fast Ethernet.

	Embedded Ethernet	Fast Ethernet
Number of clients that can be connected concurrently	Up to 5	Up to 20
Number of personal computers that can be connected concurrently	1 (recommended)	Up to 20

2 DNC operation using the FOCAS2/Ethernet function cannot be performed on embedded Ethernet function.

3 CNC screen display function and Machine remote diagnosis function can be used on Embedded Ethernet. When high-load machining such as high-speed and high-precision machining is performed or the software options that require the CPU power are specified, the screen update interval might be slower. Even if a higher-performance personal computer is used, the screen update interval is not improved.

On Embedded Ethernet, when other network functions (FOCAS2/Ethernet function etc.) are used simultaneously with these functions, it affects the screen update interval. Therefore, please do not use other network functions at the same time on Embedded Ethernet when using these functions.

4 For detailed explanations about Modbus/TCP Server function operated on the embedded Ethernet, refer to the CONNECTION MANUAL (FUNCTION) (B-64603EN-1).

For detailed explanations about Modbus/TCP Server function operated on Fast Ethernet, refer to the Industrial Ethernet CONNECTION MANUAL (B-64013EN).

NOTE

5 Communication using the embedded Ethernet is processed by the CPU on the CNC. This means that the CNC operation status may affect the communication performance of the embedded Ethernet, and, conversely, communication using the embedded Ethernet may affect CNC processing.
The embedded Ethernet function operates with a priority lower than processing performed during automatic operation and processing performed for each axis during manual operation. For this reason, the communication rate may become lower when automatic operation is performed or when many axes are controlled.
Meanwhile, the embedded Ethernet function operates with a priority higher than CNC screen display processing, C Language Executor (excluding high-level tasks), and other processing. For this reason, communication using the embedded Ethernet may degrade the performance of the above processing.

6 Note that if the embedded Ethernet is connected with a system in which a large amount of broadcast data is used, such as an intra-office network, it takes time to process broadcast data, which may affect the performance of processing including CNC screen display.

⚠ **CAUTION**

Do not use other network functions (CNC Screen Display function etc.) at the same time on embedded Ethernet when using DNC operation from the embedded Ethernet.
In case of running a program of miniature line segments such as die machining on DNC operation from the embedded Ethernet, it may affect the machining speed or the machining quality level.
If a high working accuracy is necessary, please use DNC operation of Data Server.

14.3 FIELD NETWORKS

The following field networks are supported, which allows DI/DO signals assigned to PMC addresses to be transferred to other CNCs and other vendors’ devices that comply with the same communication standards.

- PROFIBUS-DP functions (master/slave)
  PROFIBUS-DP is a communication function defined by the PROFIBUS & PROFINET International.
  PROFIBUS-DP has a master function and a slave function, and the CNC supports both functions.
  DI/DO signals can be transferred between the CNC and a device provided with PROFIBUS-DP.
  To use the PROFIBUS-DP functions (master/slave) requires an option board.

NOTE

As a personal computer tool related to the PROFIBUS-DP master function, the PROFIBUS Setting Tool is provided.
The PROFIBUS Setting Took is included in the CNC Setting Tool (drawing number: A08B-9510-J540).
The PROFIBUS Setting Tool allows the user to create parameters for the PROFIBUS-DP master function on a personal computer easily.
It is recommended that the user who uses the PROFIBUS-DP master function purchase this tool.

- DeviceNet functions (master/slave)
  DeviceNet is a communication function defined by Open DeviceNet Vendor Association, Inc. (ODVA).
  DeviceNet has a master function and a slave function, and the CNC supports both functions. DI/DO signals can be transferred between the CNC and a device provided with the DeviceNet function.
  To use the DeviceNet functions (master / slave) requires an option board.

- CC-Link function (remote device station)
  CC-Link is a communication function defined by the CC-Link Partner Association.
  CC-Link has a master function and a slave function, and the CNC supports remote device stations of the slave function. DI/DO signals can be transferred between the CNC and a device provided with the CC-Link master function.
  To use the CC-Link function (remote device station) requires an option board.

14.4 INDUSTRIAL ETHERNET

This section refers to industrial communication functions that use standard Ethernet technologies.
The CNC can perform IO communication with communication partner devices by using Industrial Ethernet.
In this manual, it refers specifically to the FL-net function, the EtherNet/IP function, the Modbus/TCP Server function, and PROFINET function.
The Fast Ethernet board (option board) is required to use the Industrial Ethernet functions.
Moreover, these functions and the Ethernet function can be operated on the same option board (Fast Ethernet board) simultaneously.(exclusive of the PROFINET IO Controller function)

- FL-net functions
  FL-net is a communication function set up by The Japan Electrical Manufacturers' Association.
  DI/DO signals can be transferred between devices equipped with this communication function.
  With a master-less method, the function enables data to be exchanged among all involved devices, and its high transfer performance and guaranteed cycle period are best suited for line control.

  Safety function by FL-net allows you to transfer safety signals between multiple CNCs connected by the FL-net.
  To use Safety function by FL-net, FL-net function and Dual check safety function are required.

NOTE

As a personal computer tool related to the FL-net function the FL-net Setting Tool is provided.
The FL-net Setting Tool is included in the CNC Setting Tool (drawing number: A08B-9510-J540).
The FL-net Setting Tool allows the user to create parameters for the FL-net function on a personal computer easily.
This tool also allows the user to manage the FL-net settings for multiple machines together and check whether I/O and PMC area are assigned correctly among the related devices.

- EtherNet/IP functions (Scanner/Adapter/Adapter Safety)
  EtherNet/IP is a communication function set up by Open DeviceNet Vendor Association, Inc. (ODVA). CNC supports Scanner function, Adapter function and Adapter Safety function.
  DI/DO signals can be transferred between devices equipped with this communication function.
  EtherNet/IP Adapter Safety function allows you to transfer safety signals with the device that supports EtherNet/IP Scanner Safety function.
  To use EtherNet/IP Adapter Safety function, EtherNet/IP Adapter function and Dual check safety function are required.

- Modbus/TCP Server function
  Modbus/TCP is a communication function defined by the Modbus Organization.
  Modbus/TCP has a Client function and a Server function, and the CNC supports the Server function.
  DI/DO signals can be transferred between the CNC and the Modbus/TCP Client device.
  The Modbus/TCP server function can be used also on Embedded Ethernet or the option board.
  However, the Modbus/TCP server function cannot be used on Embedded Ethernet and the option board at the same time.

- PROFINET functions (IO Controller / IO Device)
  PROFINET is a communication function defined by the PROFIBUS & PROFINET International.
  PROFINET has IO Controller function and IO Device function, and the CNC supports both functions.
  DI/DO signals can be transferred between the CNC and the PROFINET device.

NOTE

PROFINET IO Controller function and the Ethernet function can not be operated on the same option board.

15 PMC

Chapter 15, "PMC", consists of the following sections:

15.1 PMC BASIC SPECIFICATIONS

Table 15.1 (a) Basic specifications of the PMCs

Function	1st to 3rd path PMC	DCS PMC (Note1)
PMC Memory Type (Note2)	1st path PMC : - PMC Memory-B - PMC Memory-C - PMC Memory-D 2nd to 3rd path PMC : - PMC Memory-A - PMC Memory-B - PMC Memory-C - Shared with 1st path PMC	(Note3)
Programming language	Ladder Step sequence (Note4) Function block	Ladder Function block
Divided ladder program ・Number of programs ・File number	 16(Note5) 1 to 99	 None
Number of ladder levels	3	2 (Note6)
Level 1 execution period (Note7)	4ms or 8 ms	8 ms
Processing power • Basic instruction processing speed (transition contact) (Note8)	 18.2ns/step	 1 μs/step
Program capacity (Note9) - Ladder - Symbol / Comment - Message	 Up to about 100,000 steps 1 KB or more 8 KB or more	 Up to about 5,000 steps 1 KB or more 8 KB or more
Instructions - Basic instructions - Functional instructions (Note10)	 24 218 (230)	 24 210 (230)
Instructions (when the extended PMC ladder instruction function is invalid) - Basic instructions - Functional instructions (Note10)	 14 93 (105)	 14 85 (105)

Function	1st to 3rd path PMC	DCS PMC (Note1)
CNC interface - Inputs (F) - Outputs (G)	 768 bytes × 10 (Note11) 768 bytes × 10 (Note11)	 768 bytes 768 bytes
DI/DO I/O Link (Note 12, 13) I/O Link *i* (Notes 14 to 17) - Inputs (X) - Outputs (Y)	 Up to 2,048 points (Note18) Up to 2,048 points (Note18)	 Up to 896 points (Note19) Up to 896 points (Note19)
Symbol / Comment (Note20) - Number of symbol characters - Number of comment characters (Note21)	 40 255	 40 255
Program storage area (Flash ROM) (Note22)	Up to 2 MB (Total size of sequence programs of all PMC paths and PMC message multi-language data)	128 KB

NOTE

1 This PMC is used for Dual Check Safety (option) and handles the safety related signals.
2 As for the setting of the PMC memory type, refer to Section, "PMC MEMORY TYPE SELECTION" in this chapter.
3 There is no variation of PMC Memory Type in DCS PMC.
4 The Step Sequence is unavailable in 2nd to 3rd path PMC and the divided ladder program.
5 Maximum number of the main ladder program and divided ladder program is 16 in all of PMC system. The number of available programs depends on the configuration of specified option and each data size.
6 A program can be created on level 3 to maintain source-level compatibility with programs for other models, but it is not executed.
7 CNC parameter No. 11930 is used to specify a level-1 execution period. Note, however, that it is impossible to specify a level-1 execution period for each PMC separately.
8 It is the processing speed of contact other than Positive/Negative transition contact.
9 The maximum overall program size (including the maximum number of ladder steps, symbols/comments, and messages) varies depending on option settings. Refer to Section, "TOTAL NUMBER OF LADDER STEPS IN MULTI-PATH PMC" and "CAPACITY OF MEMORY FOR STORING SEQUENCE PROGRAMS AND MESSAGE MULTI-LANGUAGE DISPLAY FUNCTION DATA" in this chapter for details.
10 For the number of functional instructions, each parenthesized number indicates the number of all functional instructions, and each non-parenthesized number, the number of valid functional instructions.
11 It is possible to specify which program is used to control a specific CNC system.
12 You can use up to three I/O Link channels (2,048 input points and 2,048 output points).
13 The transfer cycle of the signals from I/O Link depends on the combination with each PMC and each I/O Link channel.
14 You can use up to two I/O Link i channels (2,048 input points and 2,048 output points).
15 I/O Link i can assign I/O devices for plural PMC paths in one channel.

NOTE

16 I/O Link i can be used for the channel 1 and the channel 2.
17 When you use the I/O Link i, you can select either the normal mode (2ms) or the high-speed mode (0.5ms) of the transfer cycle of signals for every group unit.
18 You can use both I/O Link and I/O Link i in a CNC system. In the case of the system including DCSPMC, you can use up to 2,048 input points and 2,048 output points.
19 When using I/O Link, you can use up to 64 input points and 64 output points.
20 These are the number for extended symbol and comment character. The number of basic symbol character is 16 and the number of comment character is 30.
21 This number is the number of single-byte characters. When you use double-byte characters as a comment, the number becomes half.
22 The capacity of the program storage area varies depending on option settings. Refer to Section, "CAPACITY OF MEMORY FOR STORING SEQUENCE PROGRAMS AND MESSAGE MULTI-LANGUAGE DISPLAY FUNCTION DATA" in this chapter for details.

Table 15.1 (b) Basic specifications of each PMC Memory Type

Function	1st to 3rd path PMC				DCS PMC (Note 1)
	PMC Memory-A	PMC Memory-B	PMC Memory-C	PMC Memory-D	
PMC Memory					
- Internal relay (R)	1,500 bytes	8,000 bytes	16,000 bytes	60,000 bytes	1,500 bytes
- System Relay (R9000 or Z)	500 bytes	500 bytes	500 bytes	500 bytes	500 bytes
- Extra relay (E) (Note 2)	10,000 bytes	10,000 bytes	10,000 bytes	10,000 bytes	(Note 3)
- Message display (A)					
· Display requests	2,000 points	2,000 points	4,000 points	6,000 points	(Note 4)
· Status displays	2,000 points	2,000 points	4,000 points	6,000 points	(Note 4)
- Nonvolatile memory					
- Timer (T)					
- Variable timer	80 bytes (40 pieces)	500 bytes (250 pieces)	1,000 bytes (500 pieces)	1,000 bytes (500 pieces)	80 bytes (40 pieces)
- Variable timer precision	80 bytes (40 pieces)	500 bytes (250 pieces)	1,000 bytes (500 pieces)	1,000 bytes (500 pieces)	80 bytes (40 pieces)
- Counter (C)					
- Variable counter	80 bytes (20 pieces)	400 bytes (100 pieces)	800 bytes (200 pieces)	1200 bytes (300 pieces)	80 bytes (20 pieces)
- Fixed counter	40 bytes (20 pieces)	200 bytes (100 pieces)	400 bytes (200 pieces)	600 bytes (300 pieces)	40 bytes (20 pieces)
- Keep relay (K)					
- User area	20 bytes	100 bytes	200 bytes	300 bytes	20 bytes
- System area	100 bytes	100 bytes	100 bytes	100 bytes	100 bytes
- Data table (D)	3,000 bytes	10,000 bytes	20,000 bytes (Note 5)	60,000 bytes (Note 5)	3,000 bytes
- Step sequence					
- Step number (S)	(None)	2,000 bytes	2,000 bytes	2,000 bytes	(None)
Functional instructions					
- Variable timers (TMR)	40 pieces	250 pieces	500 pieces	500 pieces	40 pieces
- Fixed timers (TMRB/TMRBF)	100 pieces	500 pieces	1,000 pieces	1,500 pieces	100 pieces
- Variable counters (CTR)	20 pieces	100 pieces	200 pieces	300 pieces	20 pieces
- Fixed counter (CTRB)	20 pieces	100 pieces	200 pieces	300 pieces	20 pieces
- Rising/Falling edge detection (DIFU/DIFD)	256 pieces	1,000 pieces	2,000 pieces	3,000 pieces	256 pieces
- Labels (LBL)	9,999 pieces	9,999 pieces	9,999 pieces	9,999 pieces	9,999 pieces
- Subprograms (SP)	512 pieces	5,000 pieces	5,000 pieces	5,000 pieces	512 pieces

NOTE

1. This PMC is used for Dual Check Safety function (option).
2. The extra relay is common memory for the multi-PMC function. To put it another way, its size covers all of the 1st to the 3rd PMCs. And, it can be used as nonvolatile memory by option.
3. No extra relay is available for DCS PMC.
4. The message display relay is ineffective in DCS PMC because the message display function is unavailable in it.
5. When using two or more paths of PMC memory C or one path of PMC memory D, specify the option "Nonvolatile PMC data table area expansion (40KB)". If this option is not specified in these configurations, data at D10000 and subsequent addresses is not saved. Refer to Section, "PMC MEMORY TYPE SELECTION" in this chapter for details.

15.2 MULTI-PATH PMC FUNCTION (3-PATHS)

Independent sequence programs of each PMC path are executed.
The sequence programs of each PMC path basically have their independent memory space. The E addresses represent a memory space shared by the PMC paths and can be used as the interface among the PMC paths.
The M, N addresses can be also used as the interface among the PMC paths.

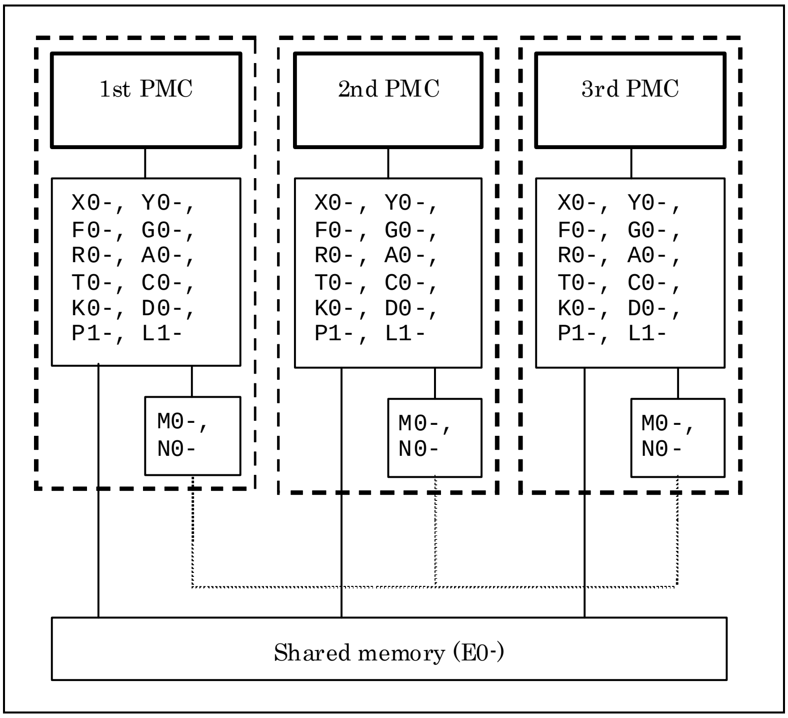


Fig. 15.2

Sequence programs and PMC parameters for a PMC path can be updated and preserved independent of other PMC paths.

NOTE
The first path PMC function is the basic function.
The second to the third path PMC functions are optional functions.

15.3 TOTAL NUMBER OF LADDER STEPS IN MULTI-PATH PMC

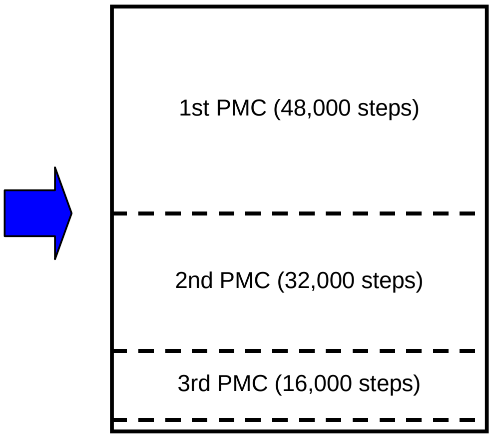
For a multi-path PMC, select an option according to the total number of steps for all PMC paths.

Option name
PMC ladder function 24,000 steps (Basic)
PMC ladder function 32,000 steps
PMC ladder function 64,000 steps
PMC ladder function 100,000 steps

For example, for a 3-path PMC, to use 48,000 steps for the 1st PMC, 32,000 steps for the 2nd PMC, and 16,000 steps for the 3rd PMC, select option PMC ladder function 100,000 steps since the total number of steps is 96,000.

Number of steps for each PMC path

	Number of steps
1st PMC	48,000 steps
2nd PMC	32,000 steps
3rd PMC	16,000 steps
(Total)	96,000 steps



Option PMC ladder function 100,000 steps

NOTE

1. If the total number of steps for all PMC paths exceeds the number of steps selected with the option, PMC alarm “ER03 PROGRAM SIZE ERROR (OPTION)“ is issued for the PMC path for which operation was performed when the number of steps was exceeded.
2. The total number of steps does not include ladder steps for the dual check safety function.

15.4 PMC MEMORY TYPE SELECTION

There are four types of PMC memory: A to D, which differ in the PMC address size and number of available function instructions. The 2nd to the 3rd PMC path can share PMC memory with the 1st PMC path. For the dual check safety PMC, there is no kind of PMC memory type.

For details of each PMC memory type, refer to Section, "PMC BASIC SPECIFICATIONS" in this chapter. For CNC parameters for selecting a PMC memory type, refer to the PMC Programming Manual (B-64513EN)”.

PMC memory type

The following tables list PMC memory types that can be selected for each PMC path.

1st path PMC	2nd to 3rd path PMC	Remark
PMC-memory B (default) PMC-memory C	PMC-memory A (default) PMC-memory B PMC-memory C Shared with 1st path PMC	You can specify up to three paths both of PMC-memory B and C in total.
PMC-memory D	Shared with 1st path PMC	

Data table area for each PMC memory type

The following table lists the data table area for each PMC memory type.

PMC memory type	Data table	Basic nonvolatile area
PMC-memory A	3,000 bytes	3,000 bytes
PMC-memory B	10,000 bytes	10,000 bytes
PMC-memory C	20,000 bytes	20,000 bytes (In case of using one path of PMC-memory C) 10,000 bytes (In case of using two or more paths of PMC-memory C)
PMC-memory D	60,000 bytes	10,000 bytes

NOTE

To use two or more paths of PMC memory C or one path of PMC memory D, specify the option “Nonvolatile PMC data table area expansion (40KB)”. If this option is not specified, data at D10000 and subsequent addresses is not saved.

15.5 PMC MESSAGE MULTI-LANGUAGE DISPLAY FUNCTION

The PMC message multi-language display function manages the language of alarm message and operator message, switching the language according to the language setting of CNC using message data defined in various languages. The message data for this function is stored into a memory card format data, separated from the data of ladder program, and loaded into CNC individually.

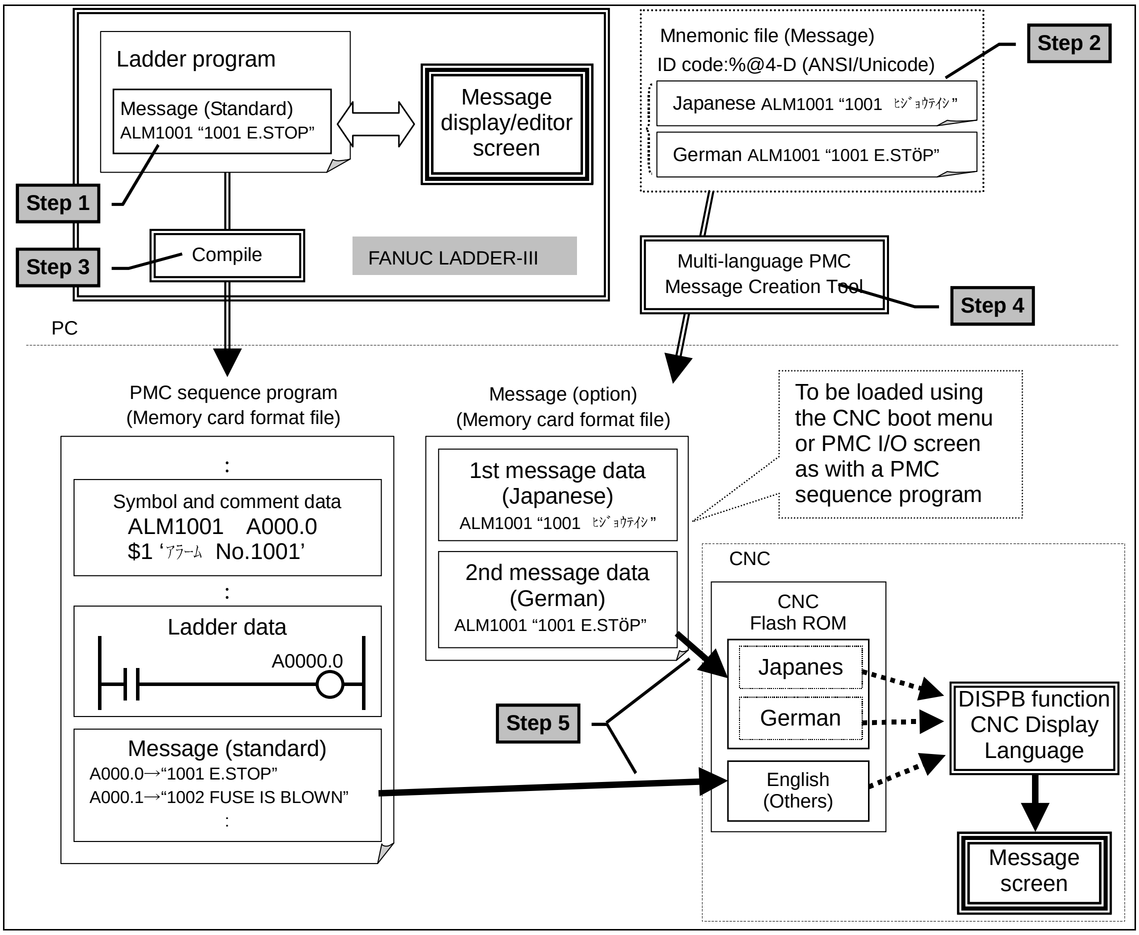


Fig. 15.5

15.6 CAPACITY OF MEMORY FOR STORING SEQUENCE PROGRAMS AND MESSAGE MULTI-LANGUAGE DISPLAY FUNCTION DATA

For memory for storing sequence programs and PMC message multi-language display function data, specify the total capacity for all PMC paths with a combination of the following two options. Calculate the size of each data in 128-KB units.

(1) PMC ladder step option

Name	Memory capacity
PMC ladder function 24,000 steps (Basic)	256 KB
PMC ladder function 32,000 steps	384 KB
PMC ladder function 64,000 steps	768 KB
PMC ladder function 100,000 steps	1 MB (1,024 KB)

(2) PMC symbol, comment, and message capacity expansion option

Name	Memory capacity
PMC symbol, comment, and message capacity expansion 512KB	512KB
PMC symbol, comment, and message capacity expansion 1MB	1MB (1,024KB)

Sample configuration

- The sequence program of the 1st PMC:	Ladder 48,000 steps, Memory size 640KB
- The sequence program of the 2nd PMC:	Ladder 32,000 steps, Memory size 384KB
- The sequence program of the 3rd PMC:	Ladder 16,000 steps, Memory size 128KB
- The multi-language message data of the 1st PMC:	Memory size 256KB
- The multi-language message data of the 2nd PMC:	Memory size 128KB
Total:	Ladder 96,000 steps, Memory size 1,536KB

For the above configuration, specify the following options:

(1) “Multi-Path PMC Function (3-Path)”
Specify the path number according to using PMC path.

(2) “PMC Ladder Function 100,000 Steps”
Specify total steps of all PMC paths.

(3) “PMC Symbol, Comment and Message capacity expansion 512KB”
Specify memory capacity expansion option in order to add to “PMC Ladder Function Step Option”. The memory size of “PMC Ladder Function 100,000 steps” is 1,024KB. Therefore, this option is necessary because it is 512KB short of memory.

NOTE

1 If the total size exceeds the capacity specified by options, alarm “ER02 PROGRAM SIZE OVER”, or “WN64 MESSAGE FILE SIZE OVER“ is issued only for the PMC path for which the capacity is exceeded.

2 When plural data are edited, inputted or outputted at the same time using CNC screen or FANUC LADDER-III, the data may not be updated even if the total size is under the specified memory capacity. In this case, simultaneous operations should be stopped and retry.

3 The above memory capacity does not include memory for the dual check safety PMC. The memory capacity of sequence programs for the dual check safety PMC is fixed to 128 KB.

15.7 I/O Link *i* and I/O Link

For the high-speed serial interface which passes input/output signals between the PMC and each I/O devices, there are two-communication method, i.e. the FANUC I/O Link *i* and the FANUC I/O Link.

For channels 1 and 2, an I/O Link type can be selected using a parameter. For channel 3, only the I/O Link is available.

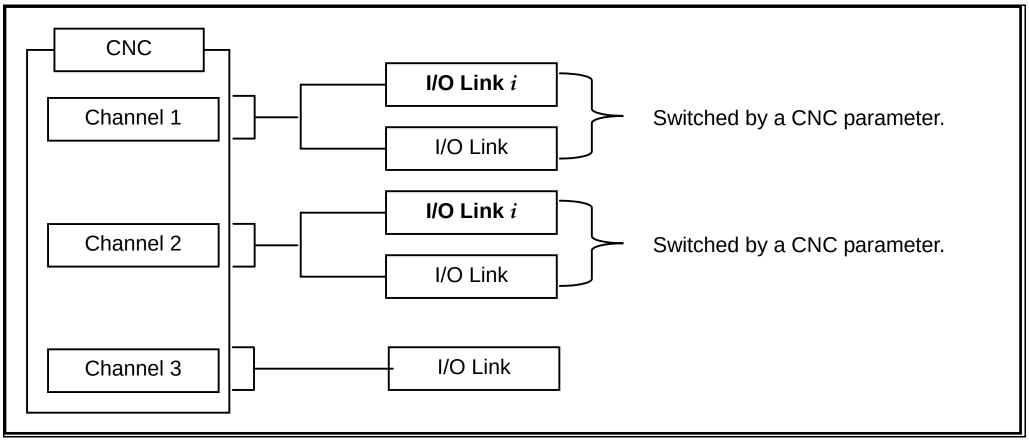


Fig. 15.7 (a) Setting of the I/O Link type for each channel

The maximum number of I/O points of I/O Link *i* is 2048/2048 for each channel. The maximum number of I/O points of the I/O Link is 1024/1024 for each channel. Up to 2048/2048 I/O points can be used in total for the entire system. Within this range of the total number of points, I/O Link *i* or I/O Link can be selected for each channel.

[The example of the combination of I/O Link *i* and I/O Link]

Channel 1	Channel 2	Channel 3	Total number of points (DI / DO)
I/O Link *i*	I/O Link *i*	—	2048 / 2048
I/O Link *i*	I/O Link	I/O Link	2048 / 2048
I/O Link *i*	I/O Link	—	2048 / 2048
I/O Link	I/O Link	I/O Link	2048 / 2048
I/O Link *i*	—	—	2048 / 2048
I/O Link	I/O Link	—	2048 / 2048
I/O Link		I/O Link	2048 / 2048
I/O Link	—	—	1024 / 1024

15.8 NONVOLATILE PMC EXTRA RELAY FUNCTION

The contents of the extension relay area (E address area) of the PMC memory are preserved. The contents of the memory are not lost even when the power to the CNC is turned off.
When the multi-path PMC function is used, the E address area represents a shared memory space, so that the values of E addresses updated by all PMC paths are preserved.

NOTE

The values of the E addresses of the PMC for the dual check safety function are not preserved.

15.9 FUNCTION BLOCK FUNCTION

A “function block” is a ladder program for implementing a process (function) that is defined as a block in advance.

A defined function block can be placed in another ladder program to execute the defined function with required input/output parameter setting.

A frequently used function can be defined as a function block to reuse the function easily, which saves many programming steps and increases the efficiency of development.

This function also allows program diagnosis without displaying details of the program in the function block, which is effective in decreasing maintenance ladder diagrams.

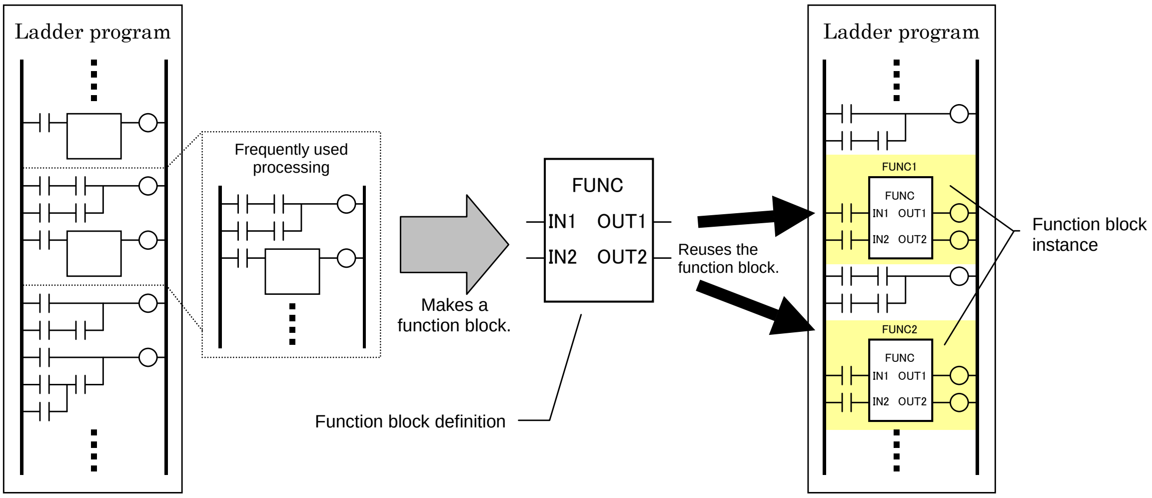


Fig. 15.9 (a) Reusing a program using a function block

16 OTHERS

Chapter 16, "OTHERS", consists of the following sections:

16.1 STATUS OUTPUT SIGNAL

16.1.1 CNC Ready Signal

This signal is sent to the machine side when CNC power is on and control becomes possible. Sending of this signal will be stopped when CNC power is turned off.

16.1.2 Servo Ready Signal

This signal is sent to the machine side when the servo system becomes operatable. Axes necessary to be braked must be braked when this signal is not sent.

16.1.3 Automatic Operation Signal

This signal is sent out when it is under automatic operation.

16.1.4 Cycle Operation Start Lamp Signal

This signal is sent out when automatic operation is being activated.

16.1.5 Feed Hold Signal

This signal is sent out when operation is held by feed hold.

16.1.6 Reset Signal

This signal is sent out to show that the CNC has been reset.

16.1.7 NC Alarm Signal

This signal is transmitted when the CNC comes under an alarm status.

16.1.8 Distribution End Signal

This signal is sent out when pulse distribution of the M, S, T, or second auxiliary function has ended, so that they can be used after move of the commanded block ends.

16.1.9 Rewinding Signal

This signal shows that program in memory is rewinding.

16.1.10 Inch Input Signal

This signal shows that input is done under inch input mode (G20).

16.1.11 Cutting Feed Signal

This signal shows that the move command is done under cutting.

16.1.12 In-position Signal

This signal shows that an axis is under in-position status. This signal is output for all axes.

16.1.13 Threading Signal

This signal shows that the machine is under threading mode (G33) or threading cycle (T series).

16.1.14 Tapping Signal

This signal is output to show that the machine is under tapping mode (G63) or tapping cycle (G74 or G84 for M series), (G84 or G88 for T series) in the canned cycle is under operation.

16.1.15 Axis Moving Signal

This signal shows that an axis is moving. This signal is sent out for each axis.
This move signal can be combined with the interlock signal to automatically clamp and unclamp the machine, or control on/off of the lubricating oil.

16.1.16 Axis Moving Direction Signal

This signal is output to show move direction of each axis. This signal is output for each axis.

16.1.17 Overtravel Alarm Signal

Indicates whether the tool was about to enter the parameter-specified forbidden area (stored stroke limits).

16.1.18 Rapid Traverse Signal

This signal shows that the move command is done under rapid traverse.

16.1.19 Constant Surface Speed Signal

This signal shows that the machine is under constant surface speed control mode (G96).

16.1.20 DI Status Output Signal

To inform the exterior of the states of software operator's panel, which are set via MDI panel, and machine operator's panel, following DI state output signals are sent.

- Mode-select check signal
- Single-block check signal
- Manual absolute check signal
- Dry-run check signal
- Machine-lock check signal
- Display-lock check signal
- Auxiliary-function-lock check signal
- Optional block-skip check signal
- Mirror-image check signal

APPENDIX

A RANGE OF COMMAND VALUE

Linear axis

- In case of millimeter input, feed screw is millimeter

	Increment system		
	IS-A	IS-B	IS-C
Least input increment (mm)	0.01	0.001	0.0001
Least command increment (mm)	0.01	0.001	0.0001
Max. programmable dimension (mm)	±999,999.99	±999,999.999	±99,999.9999
Max. rapid traverse (mm/min)*1	999,000	999,000	100,000
Feedrate range (mm/min)*1	0.01 to 999,000	0.001 to 999,000	0.0001 to 100,000
Incremental feed (mm/step)	0.01 0.1 1.0 10.0	0.001 0.01 0.1 1.0	0.0001 0.001 0.01 0.1
Tool compensation amount (mm)*2	0 to ±9,999.99	0 to ±9,999.999	0 to ±9,999.9999
Backlash compensation amount (pulses)*3	0 to ±9,999	0 to ±9,999	0 to ±9,999
Dwell (sec)*4	0 to 999,999.99	0 to 999,999.999	0 to 99,999.9999

- In case of inch input, feed screw is millimeter

	Increment system		
	IS-A	IS-B	IS-C
Least input increment (inch)	0.001	0.0001	0.00001
Least command increment (inch)	0.001	0.0001	0.00001
Max. programmable dimension (inch)	±39,370.078	±39,370.0787	±3,937.00787
Max. rapid traverse (mm/min)*1	999,000	999,000	100,000
Feedrate range (inch/min)*1	0.001 to 96,000	0.0001 to 9,600	0.00001 to 4,000
Incremental feed (inch/step)	0.001 0.01 0.1 1.0	0.0001 0.001 0.01 0.1	0.00001 0.0001 0.001 0.01
Tool compensation amount (inch)*2	0 to ±999.999	0 to ±999.9999	0 to ±999.99999
Backlash compensation amount (pulses)*3	0 to ±9,999	0 to ±9,999	0 to ±9,999
Dwell (sec)*4	0 to 999,999.99	0 to 999,999.999	0 to 99,999.9999

- In case of inch input, feed screw is inch

	Increment system		
	IS-A	IS-B	IS-C
Least input increment (inch)	0.001	0.0001	0.00001
Least command increment (inch)	0.001	0.0001	0.00001
Max. programmable dimension (inch)	±99,999.999	±99,999.9999	±9,999.99999
Max. rapid traverse (inch/min)*1	96,000	9,600	4,000
Feedrate range (inch/min)*1	0.001 to 96,000	0.0001 to 9,600	0.00001 to 4,000
Incremental feed (inch/step)	0.001 0.01 0.1 1.0	0.0001 0.001 0.01 0.1	0.00001 0.0001 0.001 0.01
Tool compensation amount (inch)*4	0 to ±999.999	0 to ±999.9999	0 to ±999.99999
Backlash compensation amount (pulses)*3	0 to ±9,999	0 to ±9,999	0 to ±9,999
Dwell (sec)*4	0 to 999,999.99	0 to 999,999.999	0 to 99,999.9999

- In case of millimeter input, feed screw is inch

	Increment system		
	IS-A	**IS-B**	**IS-C**
Least input increment (mm)	0.01	0.001	0.0001
Least command increment (mm)	0.01	0.001	0.0001
Max. programmable dimension (mm)	±999,999.99	±999,999.999	±99,999.9999
Max. rapid traverse (inch/min)[*1]	96,000	9,600	4,000
Feedrate range (mm/min)[*1]	0.01 to 999,000	0.001 to 999,000	0.0001 to 100,000
Incremental feed (mm/step)	0.01 0.1 1.0 10.0	0.001 0.01 0.1 1.0	0.0001 0.001 0.01 0.1
Tool compensation amount (mm)[*2]	0 to ±9,999.99	0 to ±9,999.999	0 to ±9,999.9999
Backlash compensation amount (pulses)[*3]	0 to ±9,999	0 to ±9,999	0 to ±9,999
Dwell (sec)[*4]	0 to 999,999.99	0 to 999,999.999	0 to 99,999.9999

- Rotary axis

	Increment system		
	IS-A	**IS-B**	**IS-C**
Least input increment (deg)	0.01	0.001	0.0001
Least command increment (deg)	0.01	0.001	0.0001
Max. programmable dimension (deg)	±999,999.99	±999,999.999	±99,999.9999
Max. rapid traverse (deg/min)[*1]	999,000	999,000	100,000
Feedrate range (deg/min)[*1]	0.01 to 999,000	0.001 to 999,000	0.0001 to 100,000
Incremental feed (deg/step)	0.01 0.1 1.0 10.0	0.001 0.01 0.1 1.0	0.0001 0.001 0.01 0.1
Tool compensation amount (deg)[*2]	0 to ±9,999.99	0 to ±9,999.999	0 to ±9,999.9999
Backlash compensation amount (pulses)[*3]	0 to ±9,999	0 to ±9,999	0 to ±9,999
Dwell (sec)[*4]	0 to 999,999.99	0 to 999,999.999	0 to 99,999.9999

NOTE

*1 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.

*2 If the mode of input is switched between inch input and metric input, the maximum compensation value that can be set at inch input time is (maximum compensation value) × 1/25.4. If a value exceeding this value is specified at inch input time, the compensation value is not converted to a metric value correctly when the mode of input is switched to metric input.

*3 The unit is the detection unit.

*4 Depends on the increment system of the axis at in address X.

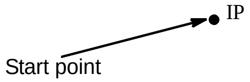
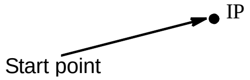
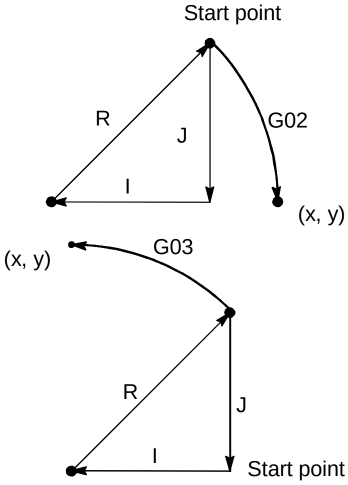
B LIST OF FUNCTIONS AND PROGRAM FORMAT

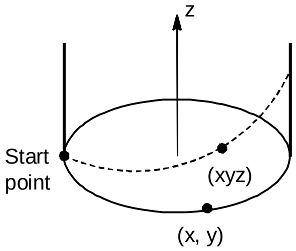
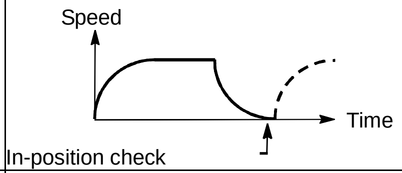
With some functions, the format used for specification on the machining center system differs from the format used for specification on the lathe system. Moreover, some functions are used for only one of the control types for the machining center system and lathe system.
Some functions cannot be added as options depending on the model.
For details of command formats, see the relevant sections or subsections.

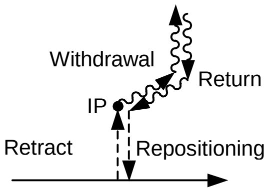
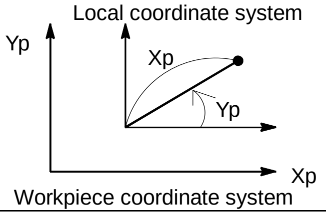
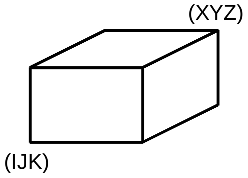
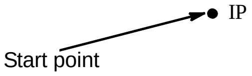
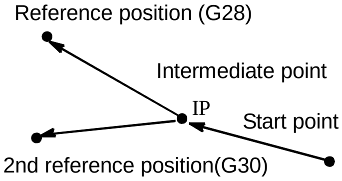
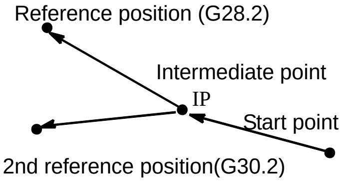
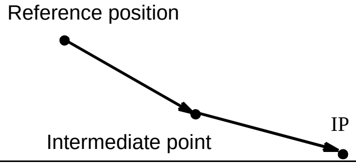
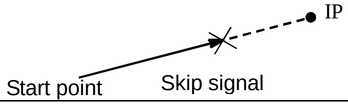
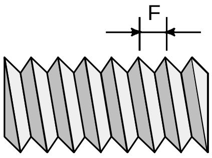
In the list, the following symbols are used:

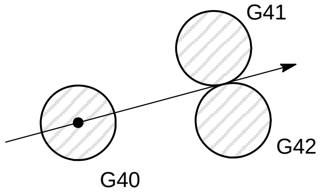
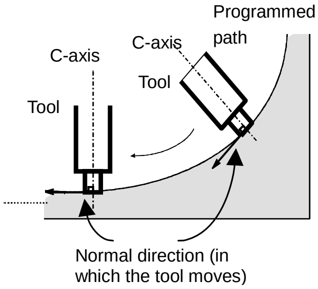
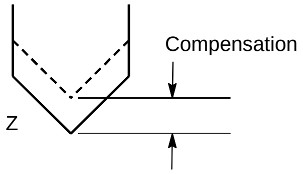
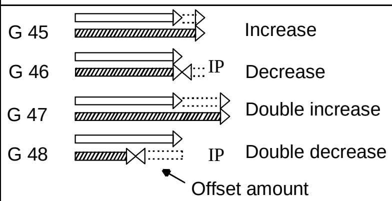
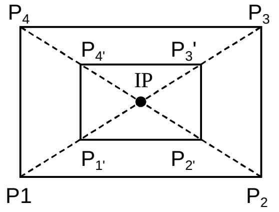
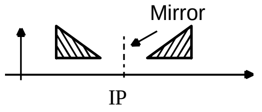
- For machining center system
  x : 1st basic axis (X), y : 2nd basic axis (Y), z : 3rd basic axis (Z)
- For lathe system
  x : 1st basic axis (X), z : 2nd basic axis (Z), coded using G code system A

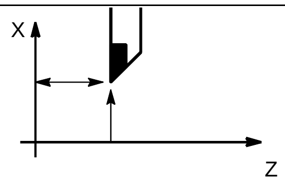
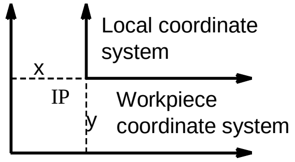
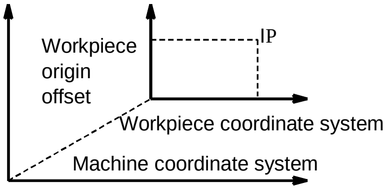
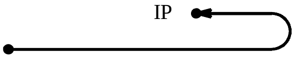
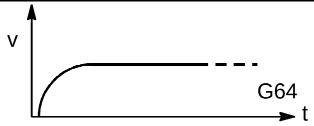
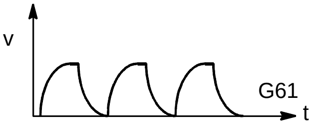
IP_ : presents a combination of arbitrary axis addresses using X, Y, Z, A, B, C, U, V, and W (such as X_ Y_ Z_ A_).
α : One arbitrary address
β : One arbitrary address
Xp : X axis or its parallel axis
Yp : Y axis or its parallel axis
Zp : Z axis or its parallel axis

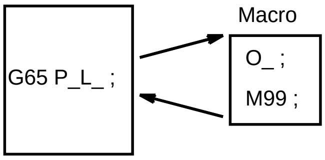
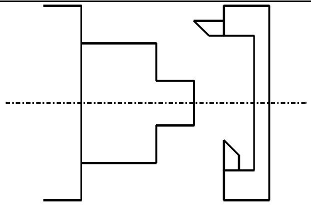
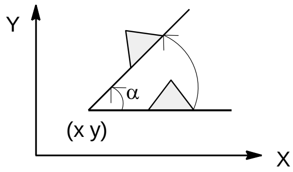
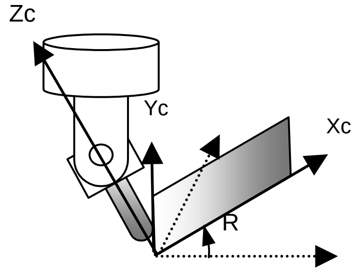
Functions	Illustration	Program format
Positioning (G00)		G00 IP_ ;
Linear interpolation (G01)		G01 IP_ F_;
Circular interpolation (G02, G03)		**• For machining center** $G17 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X\_ \ Y\_ \begin{Bmatrix} R\_ \\ I\_ \ J\_ \end{Bmatrix} F\_ ;$ $G18 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X\_ \ Z\_ \begin{Bmatrix} R\_ \\ I\_ \ K\_ \end{Bmatrix} F\_ ;$ $G19 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Y\_ \ Z\_ \begin{Bmatrix} R\_ \\ J\_ \ K\_ \end{Bmatrix} F\_ ;$ **• For lathe** $\begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X\_ \ Z\_ \begin{Bmatrix} R\_ \\ I\_ \ K\_ \end{Bmatrix} F\_ ;$

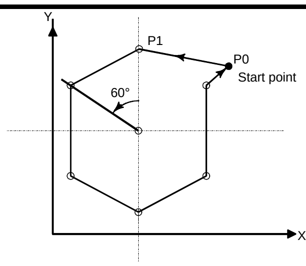
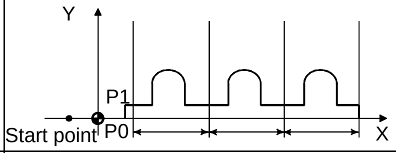
Functions	Illustration	Program format
Helical interpolation (G02, G03)	 In case of G03 on X-Y plane	$G17 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X\_ \ Y\_ \begin{Bmatrix} R\_ \\ I\_ \ J\_ \end{Bmatrix} \alpha\_ \ F\_ ;$ $G18 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X\_ \ Z\_ \begin{Bmatrix} R\_ \\ I\_ \ K\_ \end{Bmatrix} \alpha\_ \ F\_ ;$ $G19 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Y\_ \ Z\_ \begin{Bmatrix} R\_ \\ J\_ \ K\_ \end{Bmatrix} \alpha\_ \ F\_ ;$ α: Arbitrary address except the circular interpolation axis
Dwell (G04)		$G04 \begin{Bmatrix} X\_ \\ P\_ \end{Bmatrix} ;$
G code preventing buffering (G04.1)		G04.1 (P_) ; P : The operation mode of G04.1. P1: Compatible operation to preventing buffering by command of only G31. P2: Compatible operation to preventing buffering by command of only G53.
AI advanced preview control (M Series) / AI contour control (G05.1)		G05.1 Q1 ; AI advanced preview control (M Series) / AI contour control mode on G05.1 Q0 ; AI advanced preview control (M Series) / AI contour control mode off
Nano smoothing (G05.1)		G05.1 Q3 IP0 ; Nano smoothing mode on G05.1 Q0 ; Nano smoothing mode off
Cylindrical interpolation (G07.1)		G07 IP_ r_; Cylindrical interpolation mode r : Cylinder radius G07 IP 0; Cylindrical interpolation mode cancel
AI advanced preview control (M Series) / AI contour control (Advanced preview control compatible command) (G08)		G08 P1 ; AI advanced preview control (M Series) / AI contour control mode on G08 P0 ; AI advanced preview control (M Series) / AI contour control mode off
Exact stop (G09)		$G09 \begin{Bmatrix} G01 \\ G02 \\ G03 \end{Bmatrix} IP\_ ;$
Programmable data input (G10)		• **For machining center** Tool compensation memory A G10 L01 P_ R_ ; Tool compensation memory C G10 L10 P_ R_ ; (Geometry offset amount/H) G10 L11 P_ R_ ; (Wear offset amount/H) G10 L12 P_ R_ ; (Geometry offset amount/D) G10 L13 P_ R_ ; (Wear offset amount/D) • **For lathe** Geometry offset amount G10 P_ X_ Z_ R_ Q_ ; P = 10000 + Geometry offset number Wear offset amount G10 P_ X_ Z_ C_ Q_ ; P = Wear offset number

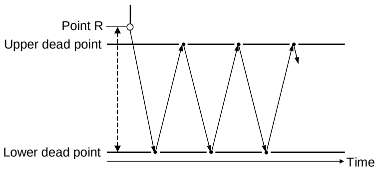
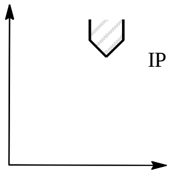
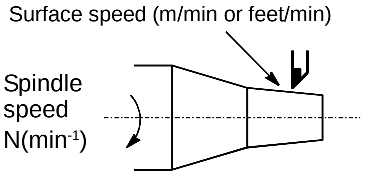
Functions	Illustration	Program format
Tool retract and recover (G10.6)		G10.6 IP_ ; Specify the amount of retraction G10.6 (as a single block containing no other commands) ; Cancel the amount of retraction
Polar coordinate interpolation (G12.1, G13.1)		G12.1 ; Polar coordinate interpolation mode on G13.1 ; Polar coordinate interpolation cancel
Polar coordinate command (G15, G16)		G17 G16 Xp_ Yp_ . . . ; G18 G16 Zp_ Xp_ . . . ; G19 G16 Yp_ Zp_ . . . ; G15 ; Cancel
Plane selection (G17, G18, G19)		G17 ; Xp Yp-plane selection G18 ; Zp Xp-plane selection G19 ; Yp Zp-plane selection
Inch/metric conversion (G20, G21)		Inch input G20 ; Metric input G21 ;
Stored stroke check (G22, 23)		G22 X_ Y_ Z_ I_ J_ K_ ; G23 ; Cancel
Reference position return check (G27)		G27 IP_ ;
Reference position return (G28) 2nd Reference position return (G30)		G28 IP_ ; G30 IP_ ;
In-position check disable reference position return (G28.2) In-position check disable 2nd reference position return (G30.2)		G28.2 IP_; G30.2 IP_;
Movement from reference position (G29)		G29 IP_ ;
Skip function (G31)		G31 IP_ F_ ;
Threading (G33)		• **For machining center** G33 IP_ F_ ; F : Lead
Threading (G32)		• **For lathe** Equal lead threading G32 P_ F_ ;

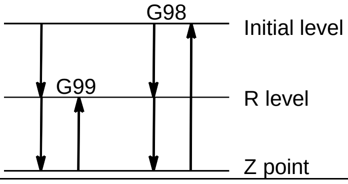
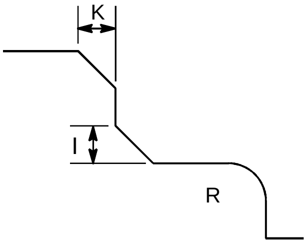
Functions	Illustration	Program format
Tool radius/tool nose radius compensation (G38, G39, G40 to G42)		• **For machining center** $\left\{\begin{matrix}G17\\G18\\G19\end{matrix}\right\} \left\{\begin{matrix}G41\\G42\end{matrix}\right\}$ D_ ; D : Tool compensation number G40 : Cancel
Tool radius/tool nose radius compensation (G40 to G42) (G38, G39)		$\left\{\begin{matrix}G41\\G42\end{matrix}\right\}$ IP_ ; G40 : Cancel
Normal direction control (G40.1, G41.1, G42.1)		• **For machining center** G41.1 ; Normal direction control on : left G42.1 ; Normal direction control on : right G40.1 ; Normal direction control cancel
Tool length compensation (G43, G44, G49)		• **For machining center** $\left\{\begin{matrix}G43\\G44\end{matrix}\right\}$ Z_ H_ ; $\left\{\begin{matrix}G43\\G44\end{matrix}\right\}$ H_ ; H : Tool compensation number G49 : Cancel
Tool offset (G43.7)		
Tool offset (G45 to G48)		• **For machining center** $\left\{\begin{matrix}G45\\G46\\G47\\G48\end{matrix}\right\}$ IP_ D_ ; D : Tool offset number
Scaling (G50, G51)		• **For machining center** G51 X_ Y_ Z_ $\left\{\begin{matrix}P\_\\I\_\ J\_\ K\_\end{matrix}\right\}$; P, I, J, K : Scaling magnification X, Y, Z : Control position of scaling G50 : Cancel
Programmable mirror image (G50.1, G51.1)		G51.1 IP_ ; G50.1 ; . . . Cancel

Functions	Illustration	Program format
Synchronous, composite, and superimposed control by program command (G50.4, G51.4, G50.5, G51.5, G50.6, G51.6)		G51.4 P_Q_(L_) ; Start synchronous control (L_ can be omitted.) G50.4 Q_ ; Cancel synchronous control. P : Number to identify synchronous master axis Q : Number to identify synchronous slave axis L : Parking start command G51.5 P_Q_ ; Start composite control G50.5 P_Q_ ; Cancel composite control P : Number to identify composite axis 1 Q : Number to identify composite axis 2 G51.6 P_Q_ ; Start superimposed control G50.6 Q_ ; Cancel superimposed control P : Number to identify superimposed master axis Q : Number to identify superimposed slave axis
Coordinate system setting Maximum spindle speed clamp (G50)		• **For lathe only** G50 IP_ ; (Coordinate system setting) G50 S_ ; (Maximum spindle speed clamp)
Local coordinate system setting (G52)		G52 IP_ ;
Command in machine coordinate system (G53)		G53 IP_ ;
Tool axis direction control (G53.1)		• **For machining center** G53.1 ; Tool axis direction control
Selection of workpiece coordinate system (G54 to G59)		{G54 : G59} IP_ ;
Single direction positioning (G60)		• **For machining center** G60 IP_ ;
Cutting mode (G64) Exact stop mode (G61) Tapping mode (G63)		G64_ ; Cutting mode G61_ ; Exact stop mode G63_ ; Tapping mode
Automatic corner override (G62)		G62_ ; Automatic corner override

Functions	Illustration	Program format
Custom macro (G65, G66, G66.1, G67)		• **One-shot call** G65 P_ L_ <Argument assignment> ; P : Program number L : Number of repetition • **Modal call** G66 P_ L_ <Argument assignment> ; Call after the move command G66.1 P_ L_ <Argument assignment> ; Each block call G67 ; Cancel
Mirror image for double turret (G68, G69)		• **For lathe only** G68 : Mirror image for double turret on G69 : Mirror image cancel
Coordinate system rotation, 3-dimensional coordinate conversion (G68, G69) (G68.1, G69.1)	 In case of X-Y plane	• **For machining center** G68 $\left\{ \begin{array}{l} \text{G17 X\_ Y\_} \\ \text{G18 Z\_ X\_} \\ \text{G19 Y\_ Z\_} \end{array} \right\}$ R $\underline{\alpha}$; G69 ; Cancel • **For lathe** G68.1 $\left\{ \begin{array}{l} \text{G17 X\_ Y\_} \\ \text{G18 Z\_ X\_} \\ \text{G19 Y\_ Z\_} \end{array} \right\}$ R $\underline{\alpha}$; G69.1 ; Cancel
Tilted working plane indexing (G68.2, G68.4)		• **For machining center** G68.2/G68.4 P_ X_ Y_ Z_ I_ J_ K_ ; G68.2 : Tilted working plane indexing G68.4 : Tilted working plane indexing (incremental multiplexed command) G69 ; Tilted working plane indexing cancel Without P : Euler's angle P1 : Roll, pitch, yaw P2 : Three points P3 : Two vectors P4 : Projection angle X, Y, Z : Feature coordinate system origin I, J, K : Euler angles for determining the orientation of the feature coordinate system
Tilted working plane indexing by tool axis direction (G68.3)	 Feature coordinate system Xc-Yc-Zc	• **For machining center** G68.3 X_ Y_ Z_ R_ ; G69 ; Cancel X,Y, Z : Feature coordinate system origin R : Feature coordinate system rotation angle about the Z-axis

Functions	Illustration	Program format
Figure copy (G72.1, G72.2)	 	● **For machining center** ● **Rotational copy** $\left\{\begin{matrix}\text{(G17)}\\ \text{(G18)}\\ \text{(G19)}\end{matrix}\right\}$ G72.1 P_ L_ $\left\{\begin{matrix}\text{X\_ Y\_}\\ \text{Z\_ X\_}\\ \text{Y\_ Z\_}\end{matrix}\right\}$ R_ ; ● **Linear copy** $\left\{\begin{matrix}\text{(G17)}\\ \text{(G18)}\\ \text{(G19)}\end{matrix}\right\}$ G72.2 P_ L_ $\left\{\begin{matrix}\text{I\_ J\_}\\ \text{K\_ I\_}\\ \text{J\_ K\_}\end{matrix}\right\}$;
Canned cycle for drilling (G73, G74, G80 to G89)		● **For machining center** G80 ; Cancel G73 $\left.\begin{matrix}\text{G74}\\ \text{G76}\\ \text{G81}\\ :\\ \text{G89}\end{matrix}\right\}$ X_Y_Z_P_Q_R_F_K_ ;
Canned cycle (G71 to G76) (G90, G92, G94)		● **For lathe only** N_ G70P_ Q_ ; G71U_ R_ ; G71P_ Q_ U_ W_ F_ S_ T_ ; G72W_ R_ ; G72P_ Q_ U_ W_ F_ S_ T_ ; G73U_ W_ R_ ; G73P_ Q_ U_ W_ F_ S_ T_ ; G74R_ ; G74X(u)_ Z(w)_ P_ Q_ R_ F_ ; G75R_ ; G75X(u)_ Z(w)_ P_ Q_ R_ F_ ; G76P_ Q_ R_ ; G76X(u)_ Z(w)_ P_ Q_ R_ F_ ; $\left\{\begin{matrix}\text{G90}\\ \text{G92}\end{matrix}\right\}$ X_ Z_ I_ F_ ; G94X_ Z_ K_ F_ ;
Canned grinding cycle (for grinding machine) (G71 to G75, G77 to G79)		● **For machining center** G75 I_ J_ K_ α_ R_ F_ P_ L_ ; G77 I_ J_ K_ α_ R_ F_ P_ L_ ; G78 I_ (J_) K_ α_ F_ P_ L_ ; G79 I_ J_ K_ α_ R_ F_ P_ L_ ; α : Arbitrary axis address of the grinding axis ● **For lathe** G71 A_ B_ W_ U_ I_ K_ H_ ; G72 P_ A_ B_ W_ U_ I_ K_ H_ ; G73 A_ (B_) W_ U_ K_ H_ ; G74 P_ A_ (B_) W_ U_ K_ H_ ;

Functions	Illustration	Program format
High precision oscillation function (G81.1, G80)		G81.1 Z_ Q_ R_ F_ ; Z : Upper dead point Q : Distance between the upper dead point and lower dead point R : Distance from the upper dead point to point R F : Feedrate during oscillation base feedrate G80 ; Cancels oscillation
Absolute/incremental programming (G90/G91)		• **For machining center** G90_ ; Absolute programming G91_ ; Incremental programming : G90_ G91_ ; Programming in both modes • **For lathe** X_ Z_ C_ : Absolute programming U_ W_ H_ : Incremental programming Distinguished by an address specified in combined use with a G function such as G00 and G01.
Maximum incremental command value check (G91.1)		G91.1 IP_ ; IP_ ; Maximum incremental value Set 0 to cancel maximum incremental value check.
Change of workpiece coordinate system (G92) Maximum spindle speed clamp (G92)		• **For machining center** G92 IP_ ; Change of workpiece coordinate system G92 S_ ; Constant surface speed control : Maximum spindle speed clamp
Workpiece coordinate system preset (G92.1)		• **For machining center** G92.1 IP 0 ;
Inverse time feed (G93)		• **For machining center** G93 ; Inverse time setting mode
Feed per minute, Feed per revolution (G94, G95)	mm/min inch/min	• **For machining center** G94 F_ ; Feed per minute G95 F_ ; Feed per revolution
(G98, G99)	mm/rev inch/rev	• **For lathe** G98 F_ ; Feed per minute G99 F_ ; Feed per revolution
Constant surface speed control (G96, G97)		G96 S_ ; Constant surface speed control on (surface speed specification) G97 S_ ; Constant surface speed control off (spindle speed specification)

Functions	Illustration	Program format
Speed display function of a milling tool with servo motor (G96.1,G96.2,G96.3,G96.4)		G96.1 P_R_; The next block starts operating upon completion of spindle indexing (the SV speed control mode is off). G96.2 P_R_; The next block starts operating without waiting for spindle indexing to complete. G96.3 P_; The next block starts operating after confirming the completion of spindle indexing (the SV speed control mode is off). G96.4 P_; The SV speed control mode is on.
Initial point return / R point return (G98, G99)		• **For machining center** G98_ ; G99_ ;
Optional chamfering/corner R		• **For machining center** ,C_ : Chamfering ,R_ : Corner R
Chamfering/corner R		• **For lathe only** $X\_\left\{\begin{matrix}C\pm K\\R\_\end{matrix}\right\}P\_;$ $Z\_\left\{\begin{matrix}C\pm K\\R\_\end{matrix}\right\}P\_;$
In-feed control (for grinding machine) (G160, G161)		• **For machining center** G161 R_ ; Figure program (G01, G02, G03) G160 ;

C PROGRAM CODE LIST

Character name	ISO code		EIA code		Custom macro		Usable as file name
	Character	Code (hexadecimal)	Character	Code (hexadecimal)	Without custom macro	With custom macro	
Number 0	0	30	0	20			*
Number 1	1	B1	1	01			*
Number 2	2	B2	2	02			*
Number 3	3	33	3	13			*
Number 4	4	B4	4	04			*
Number 5	5	35	5	15			*
Number 6	6	36	6	16			*
Number 7	7	B7	7	07			*
Number 8	8	B8	8	08			*
Number 9	9	39	9	19			*
Address A	A	41	a	61			*
Address B	B	42	b	62			*
Address C	C	C3	c	73			*
Address D	D	44	d	64			*
Address E	E	C5	e	75			*
Address F	F	C6	f	76			*
Address G	G	47	g	67			*
Address H	H	48	h	68			*
Address I	I	C9	i	79			*
Address J	J	CA	j	51			*
Address K	K	4B	k	52			*
Address L	L	CC	l	43			*
Address M	M	4D	m	54			*
Address N	N	4E	n	45			*
Address O	O	CF	o	46			*
Address P	P	50	p	57			*
Address Q	Q	D1	q	58			*
Address R	R	D2	r	49			*
Address S	S	53	s	32			*
Address T	T	D4	t	23			*
Address U	U	55	u	34			*
Address V	V	56	v	25			*
Address W	W	D7	w	26			*
Address X	X	D8	x	37			*
Address Y	Y	59	y	38			*
Address Z	Z	5A	z	29			*
Delete	DEL	FF	Del	7F	×	×	
Back space	BS	88	BS	2A	×	×	
Tabulator	HT	09	Tab	2E	×	×	
End of block	LF or NL	0A	CR or EOB	80			
Carriage return	CR	8D			×	×	
Space	SP	A0	SP	10	□	□	
Absolute rewind stop	%	A5	ER	0B			
Control out (start of comment)	(	28	(2-4-5)	1A			
Control in (end of comment)	)	A9	(2-4-7)	4A			

Character name	ISO code		EIA code		Custom macro		Usable as file name
	Character	Code (hexadecimal)	Character	Code (hexadecimal)	Without custom macro	With custom macro	
Plus sign	+	2B	+	70			*
Minus sign	-	2D	-	40			*
Colon (address O)	:	3A					
Optional block skip	/	AF	/	31			
Period (decimal point)	.	2E	.	6B			*
Sharp	#	A3	Parameter (No.6012)				
Dollar sign	$	24			□	□	
Ampersand	&	A6	&	0E			
Apostrophe	‘	27			□	□	
Asterisk	*	AA	Parameter (No.6010)				
Comma	,	AC	,	3B			
Semicolon	;	BB			□	□	
Left angle bracket	<	3C					
Equal sign	=	BD	Parameter (No.6011)				
Right angle bracket	>	BE					
Question mark	?	3F			△	○	
Commercial at mark	@	C0			△	△	
Quotation mark	“	22			□	□	
Left square bracket	[	DB	Parameter (No.6013)		△		
Right square bracket	]	DD	Parameter (No.6014)		△		
Underscore	_	6F	Parameter (No.6018)		△	△	*
Lowercase letter a	a	E1			△	△	*
Lowercase letter b	b	E2			△	△	*
Lowercase letter c	c	63			△	△	*
Lowercase letter d	d	E4			△	△	*
Lowercase letter e	e	65			△	△	*
Lowercase letter f	f	66			△	△	*
Lowercase letter g	g	E7			△	△	*
Lowercase letter h	h	E8			△	△	*
Lowercase letter i	i	69			△	△	*
Lowercase letter j	j	6A			△	△	*
Lowercase letter k	k	EB			△	△	*
Lowercase letter l	l	6C			△	△	*
Lowercase letter m	m	ED			△	△	*
Lowercase letter n	n	EE			△	△	*
Lowercase letter o	o	6F			△	△	*
Lowercase letter p	p	F0			△	△	*
Lowercase letter q	q	71			△	△	*
Lowercase letter r	r	72			△	△	*
Lowercase letter s	s	F3			△	△	*
Lowercase letter t	t	74			△	△	*
Lowercase letter u	u	F5			△	△	*
Lowercase letter v	v	F6			△	△	*
Lowercase letter w	w	77			△	△	*
Lowercase letter x	x	78			△	△	*

Character name	ISO code		EIA code		Custom macro		Usable as file name
	Character	Code (hexadecimal)	Character	Code (hexadecimal)	Without custom macro	With custom macro	
Lowercase letter y	y	F9			△	△	*
Lowercase letter z	z	FA			△	△	*

NOTE

1 The symbols used in the "Custom macro" column have the following meanings.

(Space) : The character will be registered in memory and has a specific meaning. It is used incorrectly in a statement other than a comment, an alarm occurs.

× : The character will not be registered in memory and will be ignored.

△ : The character will be registered in memory, but will be ignored during program execution. The character that has "*" in its "Usable as file name" field and is used as a file name is not ignored. Lowercase alphabetic characters in an NC program follow the setting of bit 0 (ESL) of parameter No. 3459.

○ : 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 The symbol used in the column "Usable as file name" has the following meaning:

* : Capable of being coded between "<" and ">" as a file name.

3 Codes not in this table are ignored if their parity is correct.

4 Codes with incorrect parity cause the TH alarm. But they are ignored without generating the TH alarm when they are in the comment section.

D OUTLINE DRAWINGS OF UNITS AND CONNECTORS

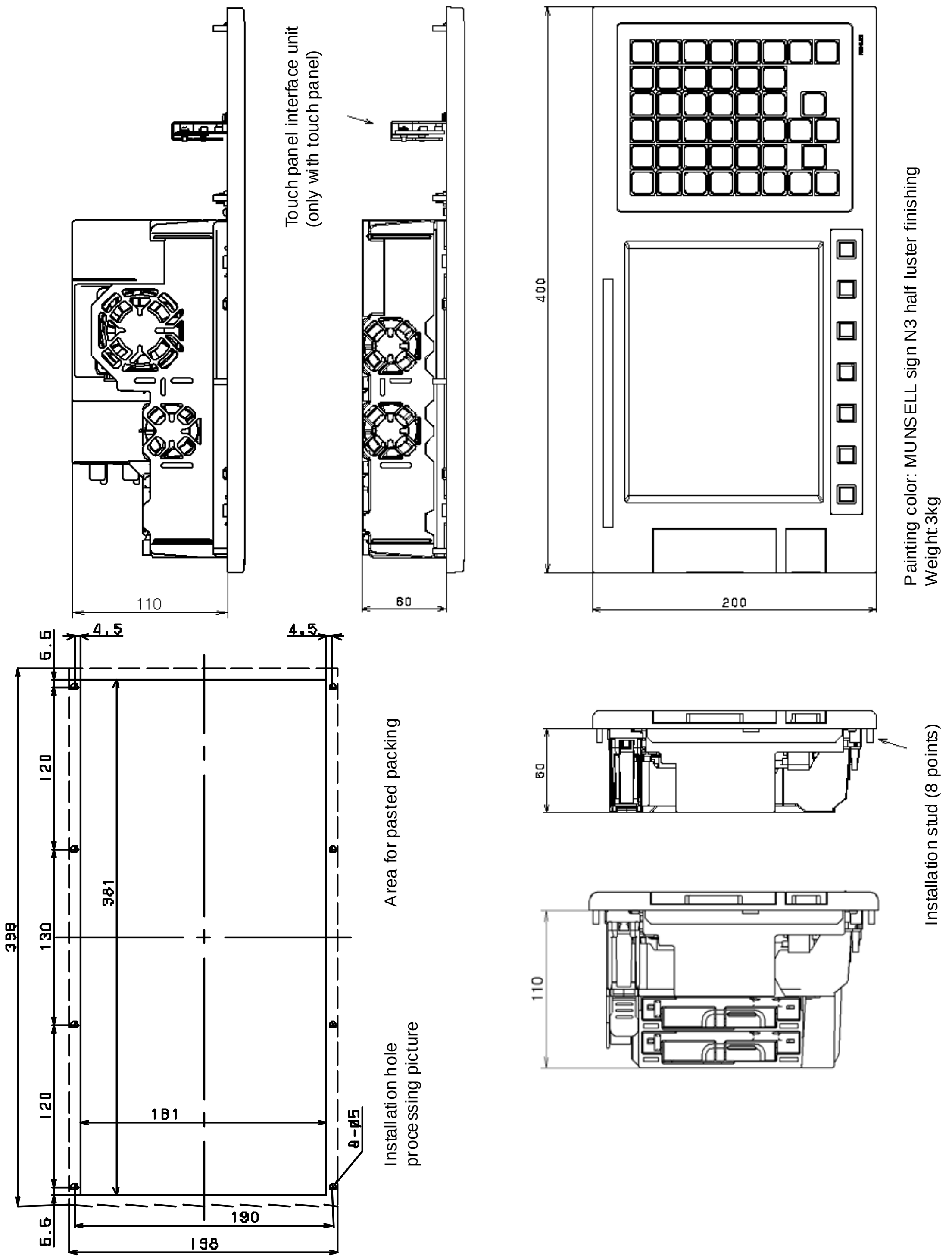
Outline drawings for LCD-mounted type control unit

Name		See
Control unit	8.4" color LCD/MDI (horizontal)	Fig. U1
	8.4" color LCD/MDI (vertical)	Fig. U2
	10.4" color LCD	Fig. U3
	15" color LCD	Fig. U4

Other outline drawings

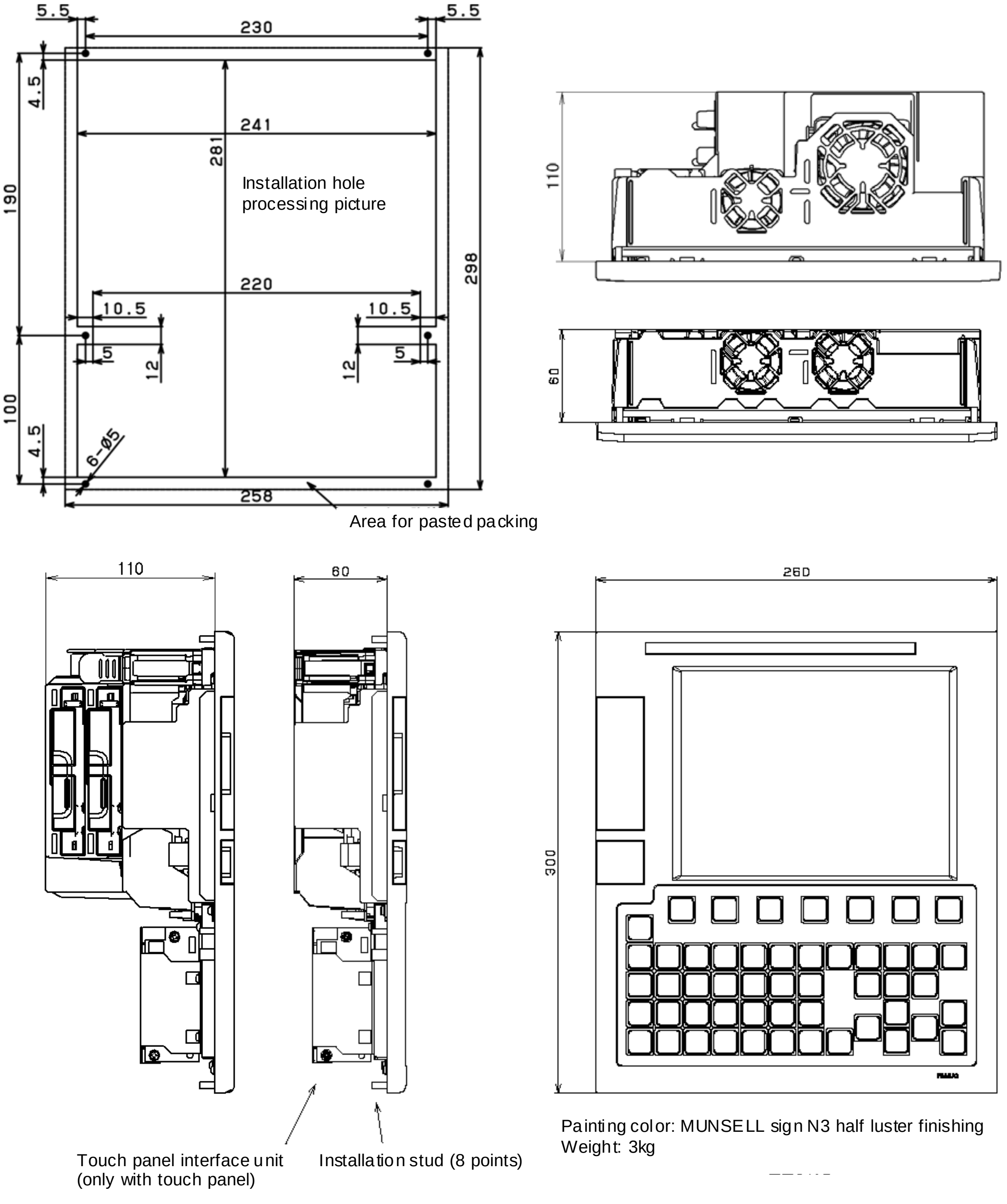
Name	See
MDI unit (small type, 200x140mm) (See Figs. U18(b),(d) for key layout.)	Fig. U5
MDI unit (ONG, 200x260mm) (See Figs. U18(a),(c) for key layout.)	Fig. U6
MDI unit (ONG, 220x230mm) (See Figs. U18(a),(c) for key layout.)	Fig. U7
MDI unit (ONG, 220x290mm) (See Figs. U18(a),(c) for key layout.)	Fig. U8
MDI unit (QWERTY TYPE A, 160x290mm) (See Figs. U18(e) for key layout.)	Fig. U9
MDI unit (QWERTY TYPE B, 145x400mm) (See Figs. U18(f) for key layout.)	Fig. U10
Manual pulse generator	Fig. U11
Pendant type manual pulse generator	Fig. U12
Separate detector interface unit or analog input separate detector interface unit	Fig. U13
Absolute pulse coder battery case for a separate detector	Fig. U14
Battery case for external installation	Fig. U15
Punch panel	Fig. U16
Ethernet Connector Panel	Fig. U17
MDI key layout	Fig. U18

Fig. U1 Control unit (8.4” color LCD/MDI horizontal)



NOTE: The unit is installed from outside cabinet.
Fix with M4 nut from the reverse side. The tightening torque is 2.0 N ·m.

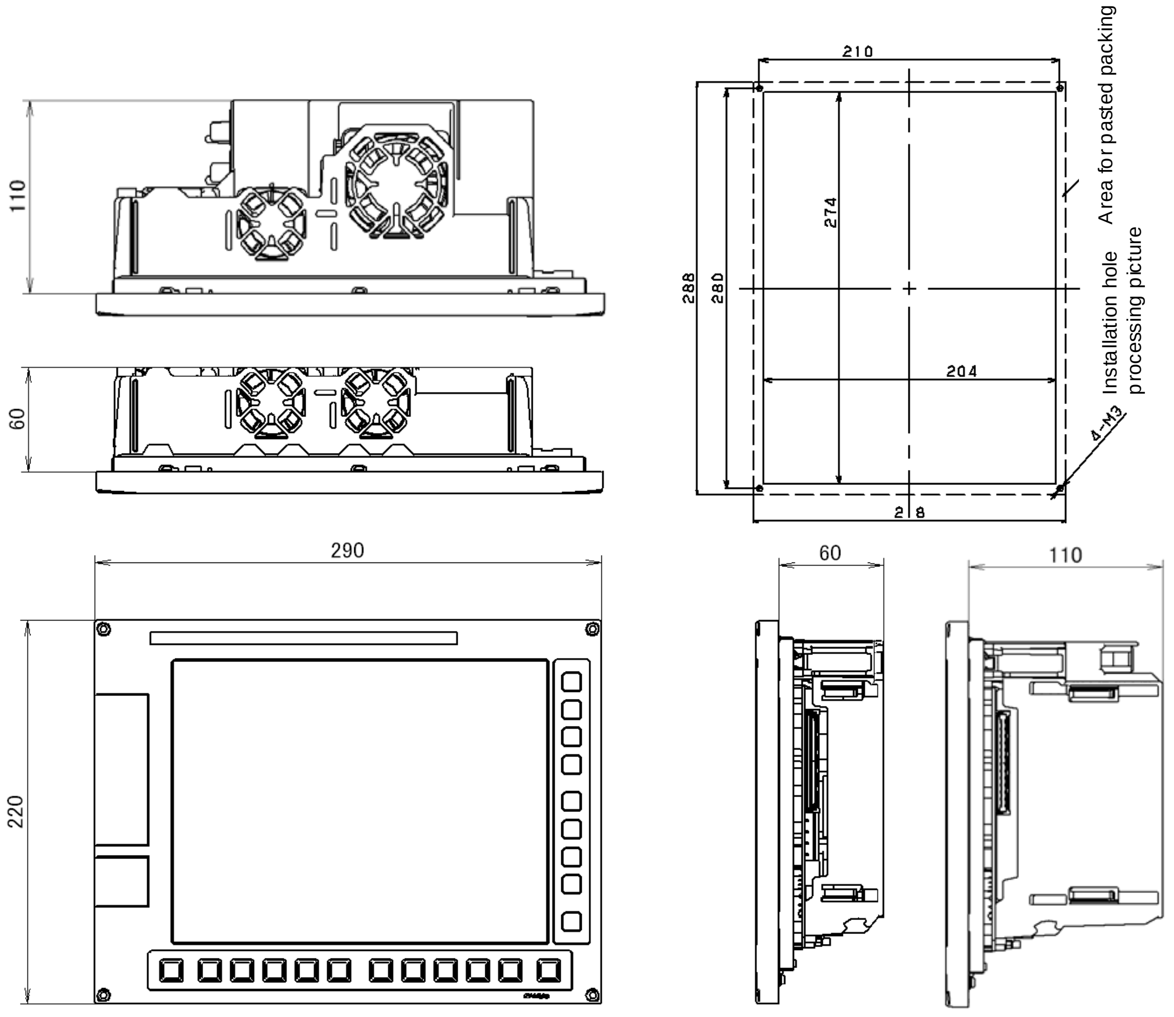
Fig. U2 Control unit (8.4” color LCD/MDI vertical)



Painting color: MUNSELL sign N3 half luster finishing
Weight: 3kg

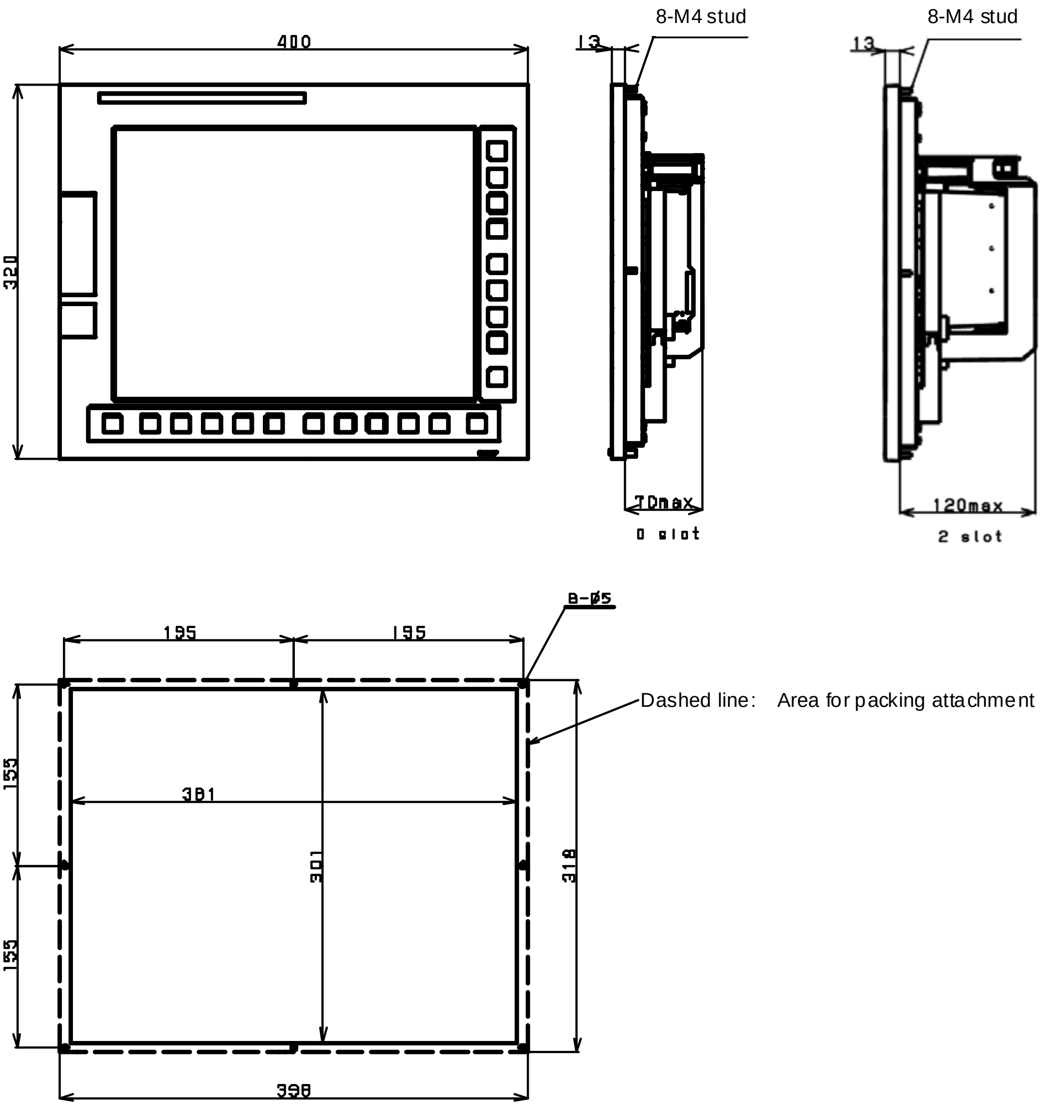
NOTE: The unit is installed from outside cabinet.
Fix with M4 from the reverse. The tightening torque 2.0 N·m.

Fig. U3 Control unit (10.4” color LCD)



Painting color: MUNSELL sign N3 half luster finishing
Weight: 3kg

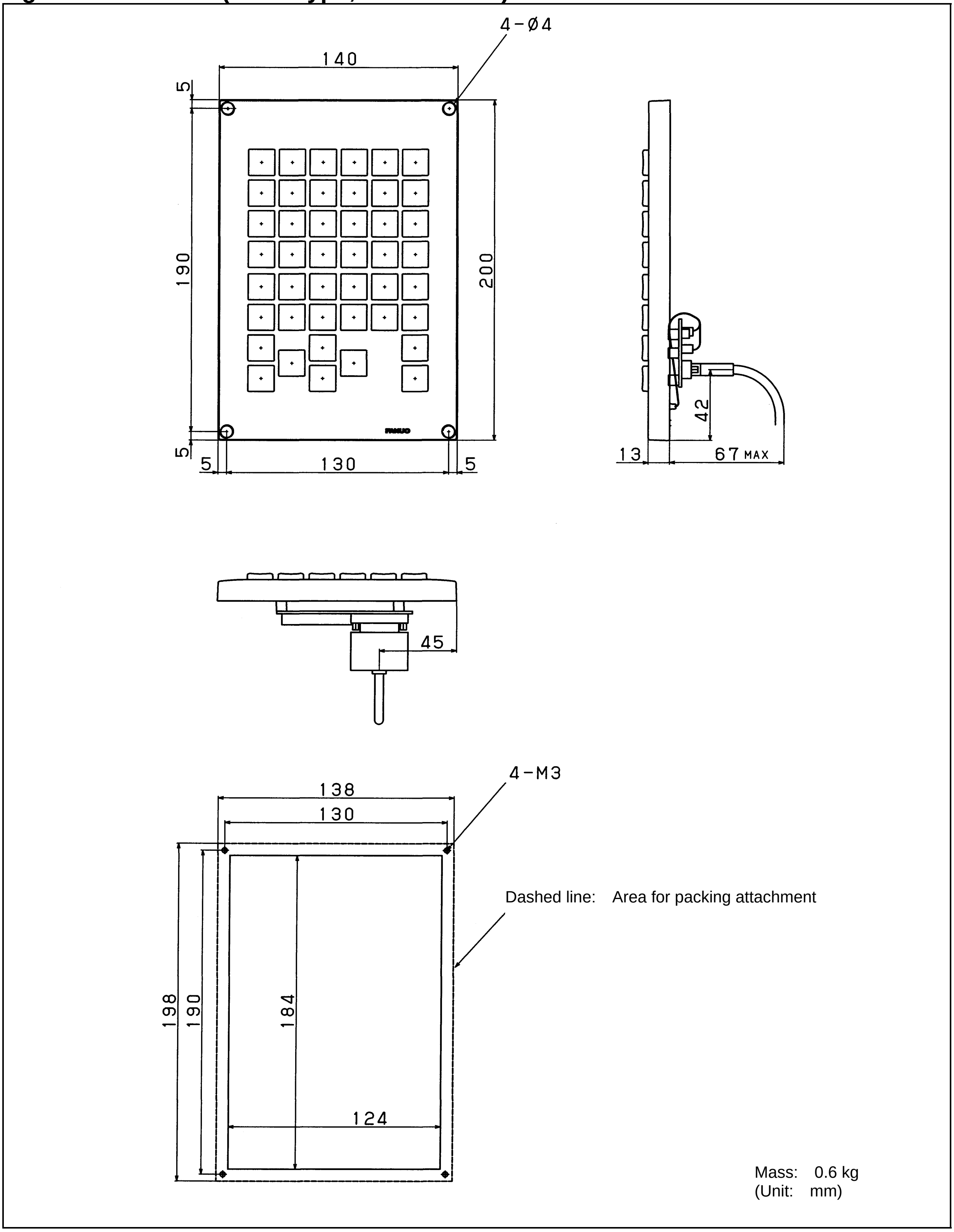
Fig. U4 Control unit (15” color LCD)



Mass: 5 kg
(Unit: mm)

15" unit mounting hole machining drawing
(Mount the unit onto the outside of the cabinet then secure the unit with nuts from the inside.)

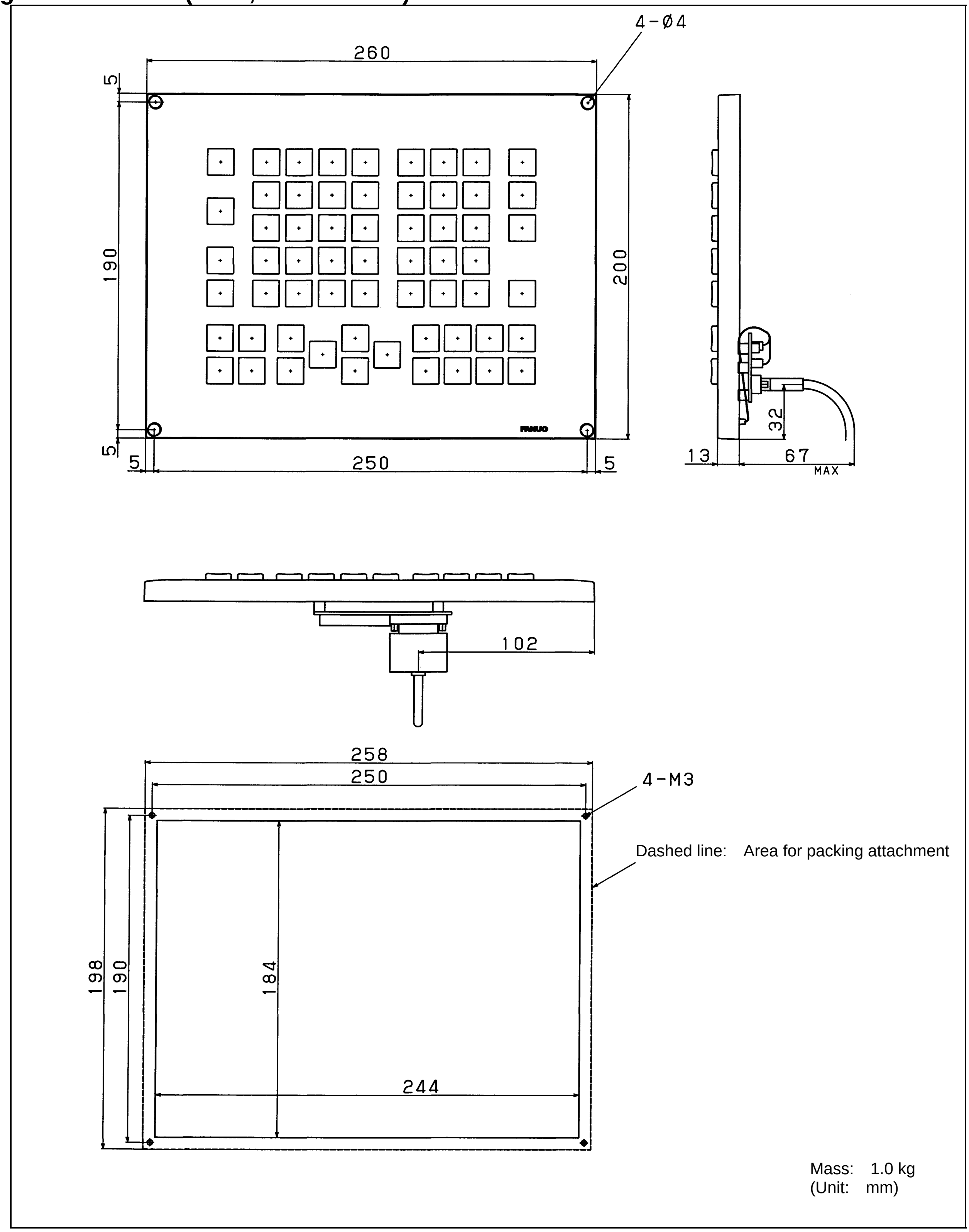
Fig. U5 MDI unit (small type, 200x140mm)



Dashed line: Area for packing attachment

Mass: 0.6 kg
(Unit: mm)

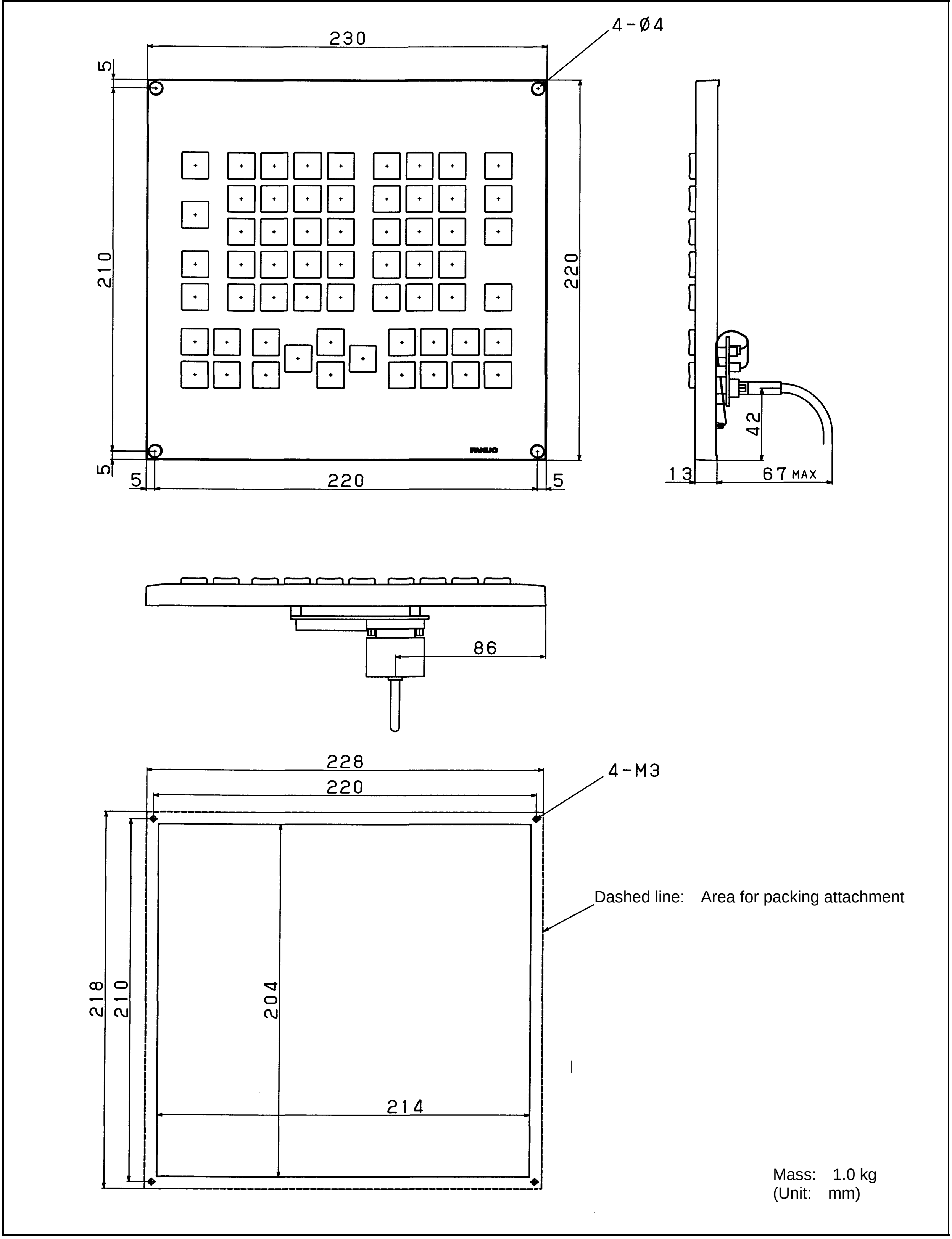
Fig. U6   MDI unit (ONG, 200x260mm)



Dashed line: Area for packing attachment

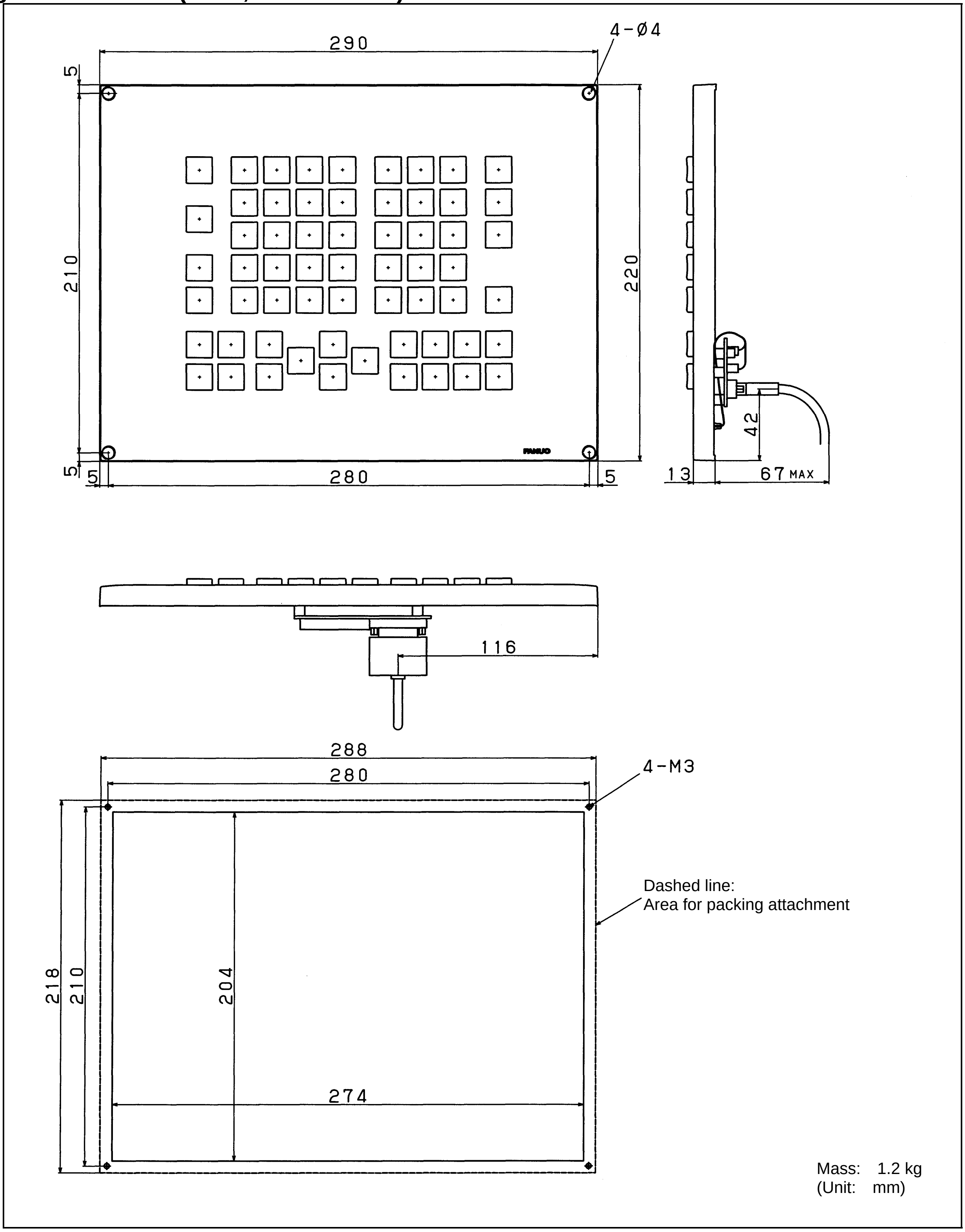
Mass: 1.0 kg
(Unit: mm)

Fig. U7 MDI unit (ONG, 220x230mm)



Mass: 1.0 kg
(Unit: mm)

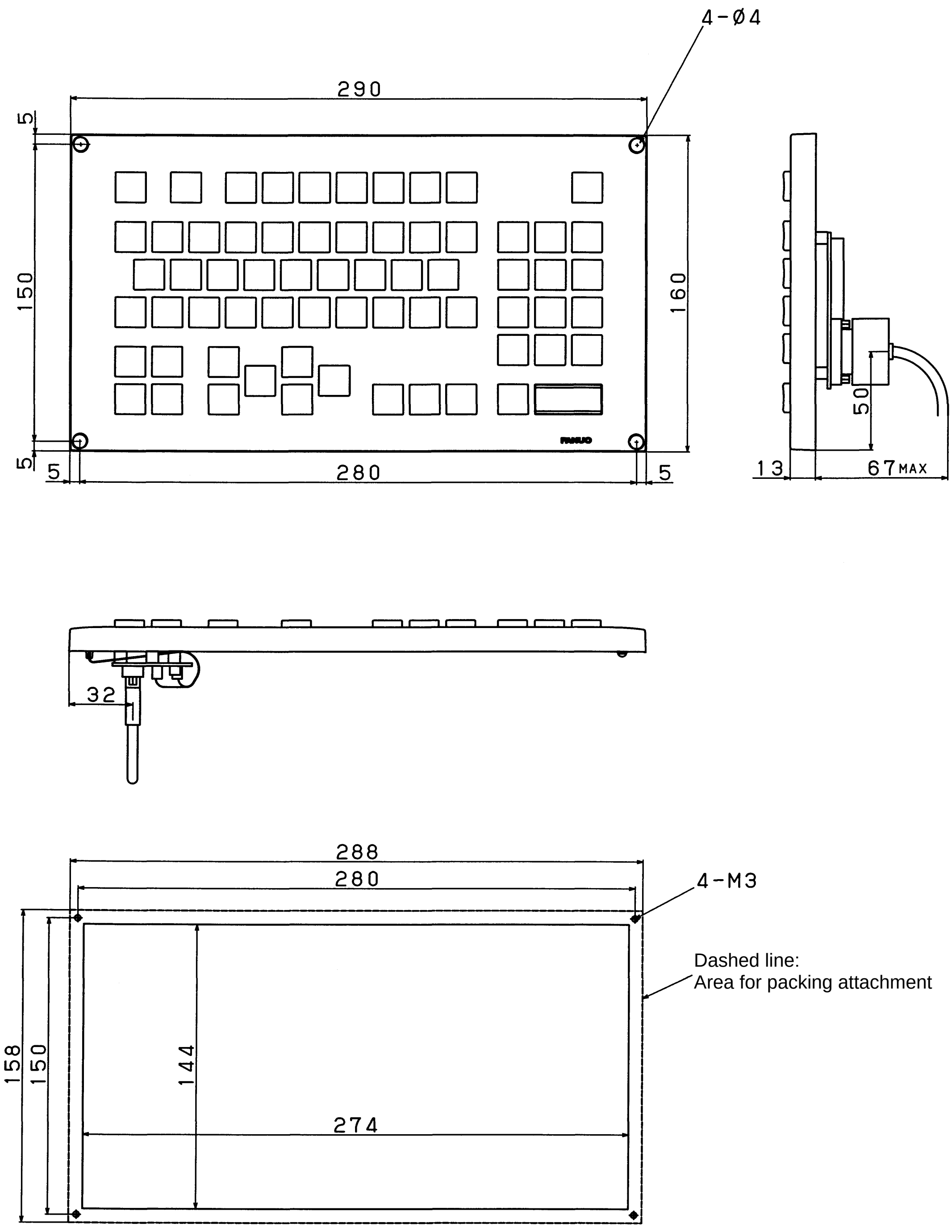
Fig. U8 MDI unit (ONG, 220x290mm)



Dashed line:
Area for packing attachment

Mass: 1.2 kg
(Unit: mm)

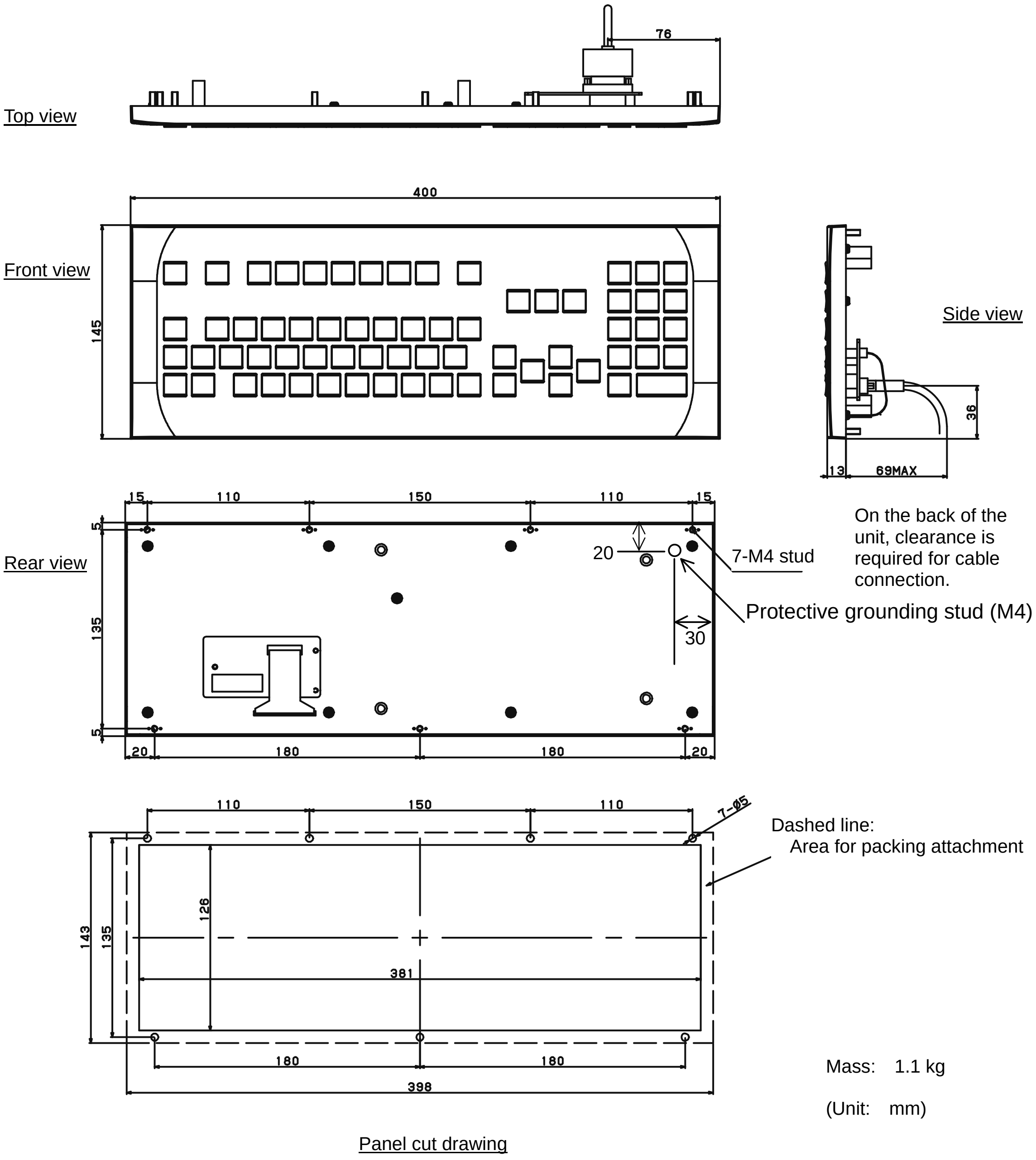
Fig. U9 MDI unit (QWERTY TYPE A, 160x290mm)



Dashed line:
Area for packing attachment

Mass: 1.0 kg
(Unit: mm)

Fig. U10 MDI unit (QWERTY TYPE B, 145x400mm)



On the back of the unit, clearance is required for cable connection.

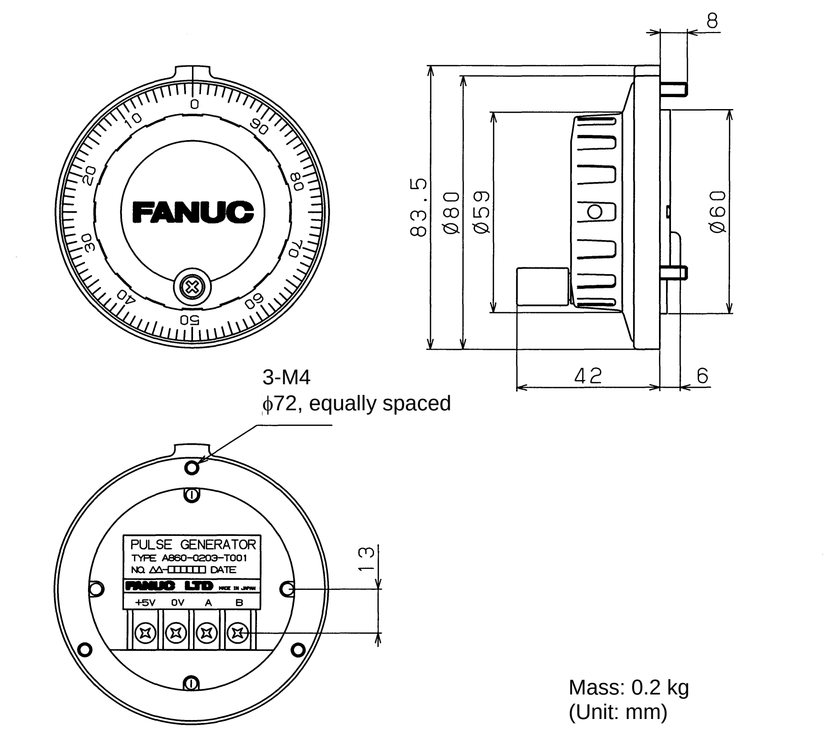
Dashed line:
Area for packing attachment

Mass: 1.1 kg

(Unit: mm)

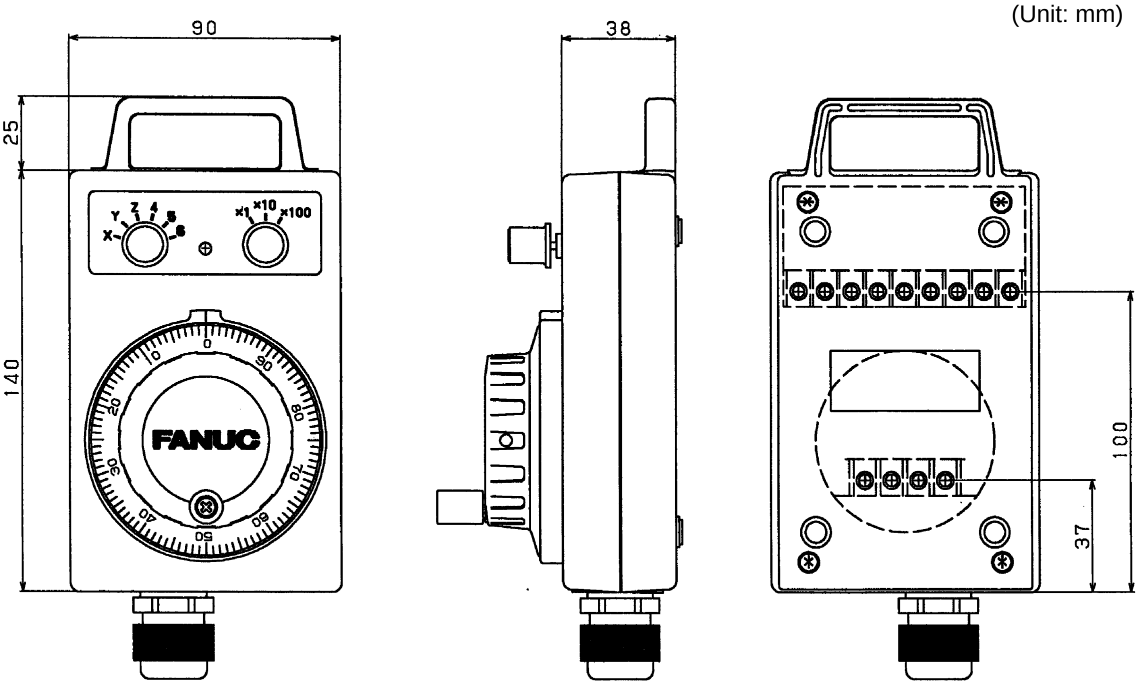
(Mount the unit onto the outside of the cabinet then secure the unit with nuts from the inside.)

Fig. U11 Manual pulse generator



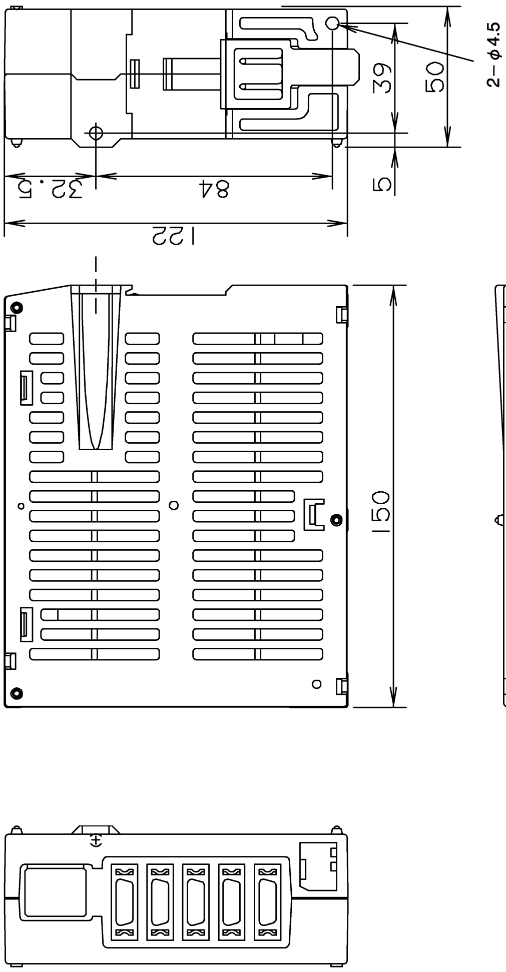
Mass: 0.2 kg
(Unit: mm)

Fig. U12 Pendant type manual pulse generator



Mass: 0.4kg
(Unit: mm)

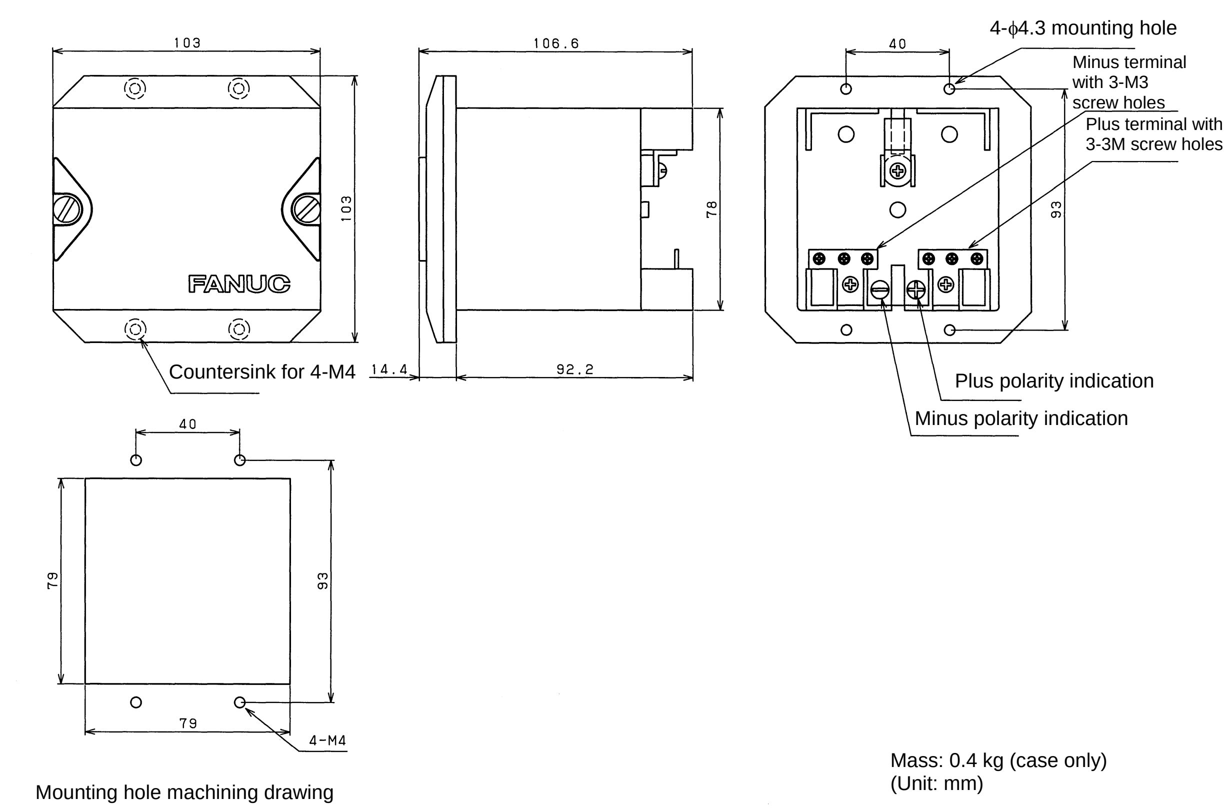
Fig. U13 Separate detector interface unit or analog input separate detector interface unit



Mass: 0.4kg

(Unit: mm)

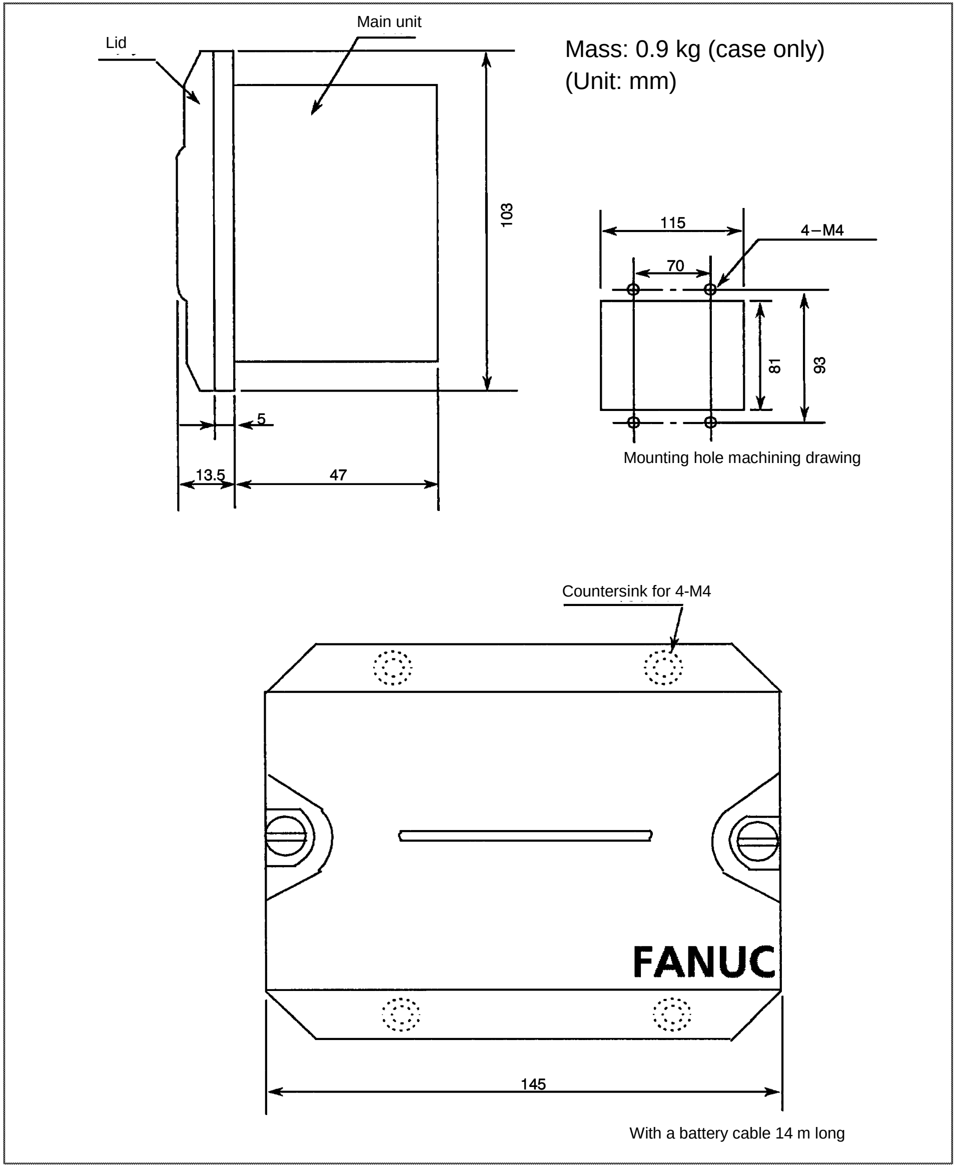
Fig. U14 Absolute pulse coder battery case for a separate detector



Mounting hole machining drawing

Mass: 0.4 kg (case only)
(Unit: mm)

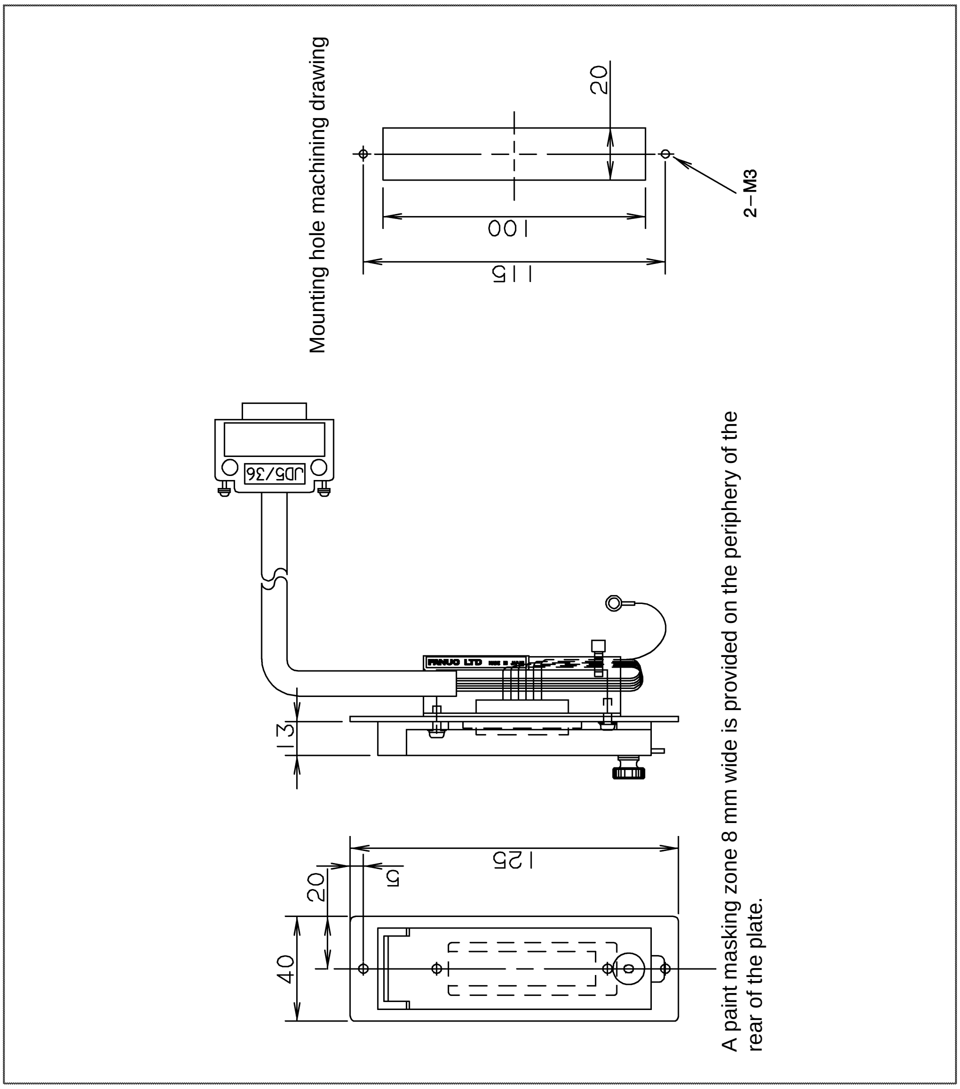
Fig. U15 Battery case for external installation



Mass: 0.9 kg (case only)
(Unit: mm)

With a battery cable 14 m long

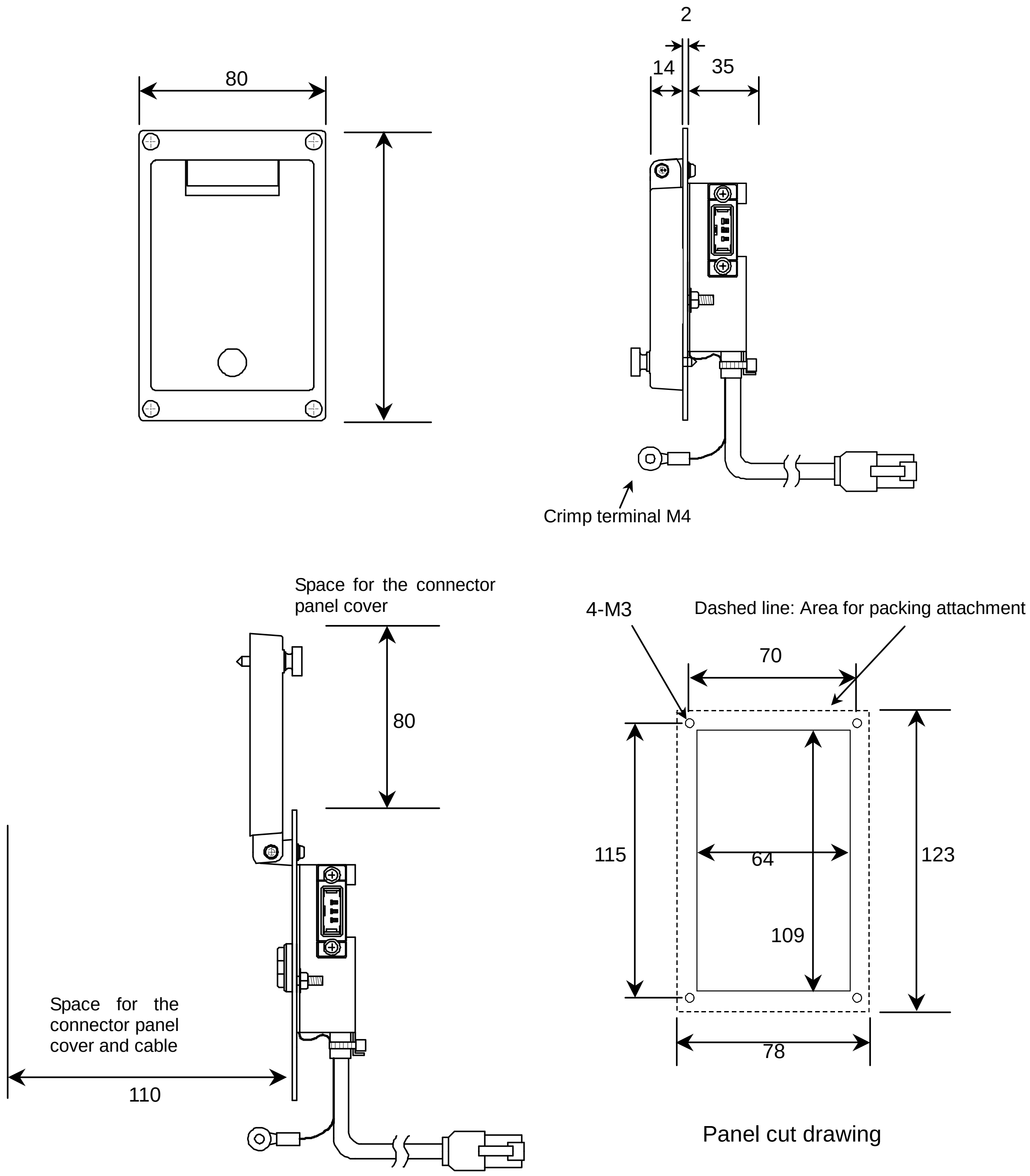
Fig. U16 Punch panel (for LCD-mounted type control unit)



Mounting hole machining drawing

A paint masking zone 8 mm wide is provided on the periphery of the rear of the plate.

Fig. U17 Ethernet Connector Panel



Weight: 0.4kg

(Unit: mm)

Fig. U18 MDI key layout

Fig. U18(a) MDI unit (ONG) with lathe system (T series)

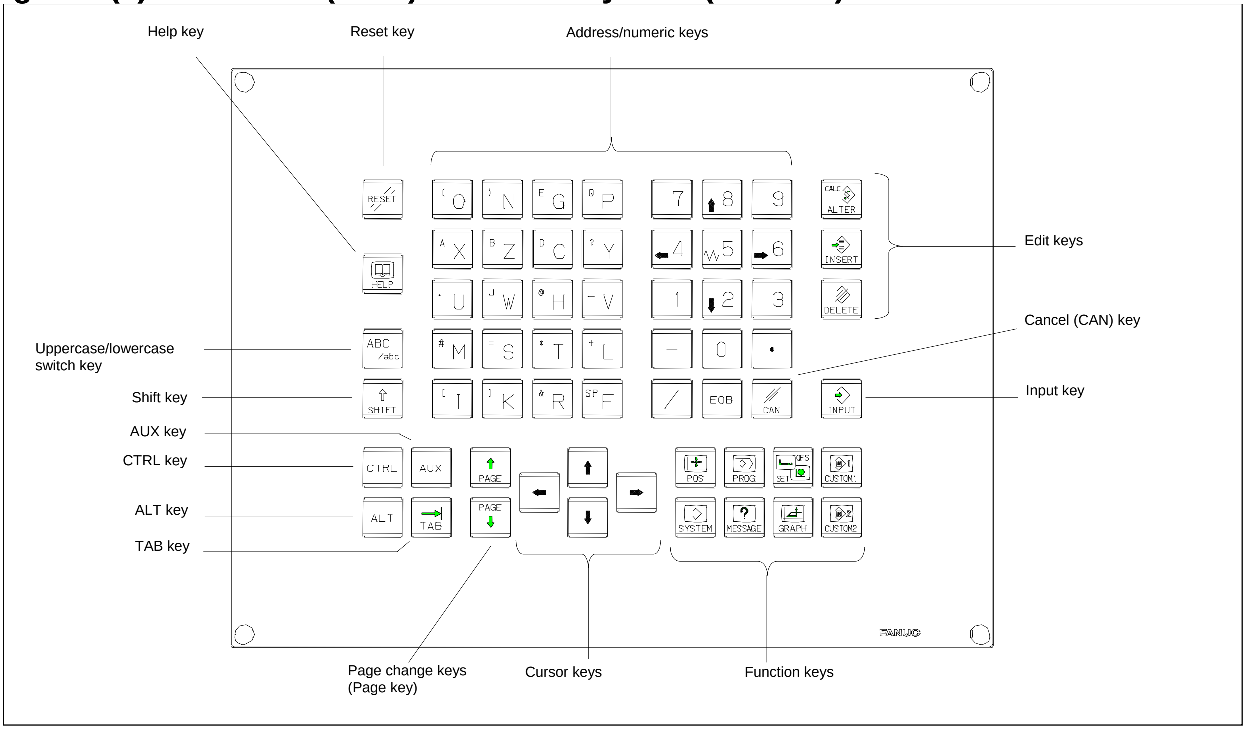


Fig. U18(b) MDI unit (small type) with lathe system (T series)

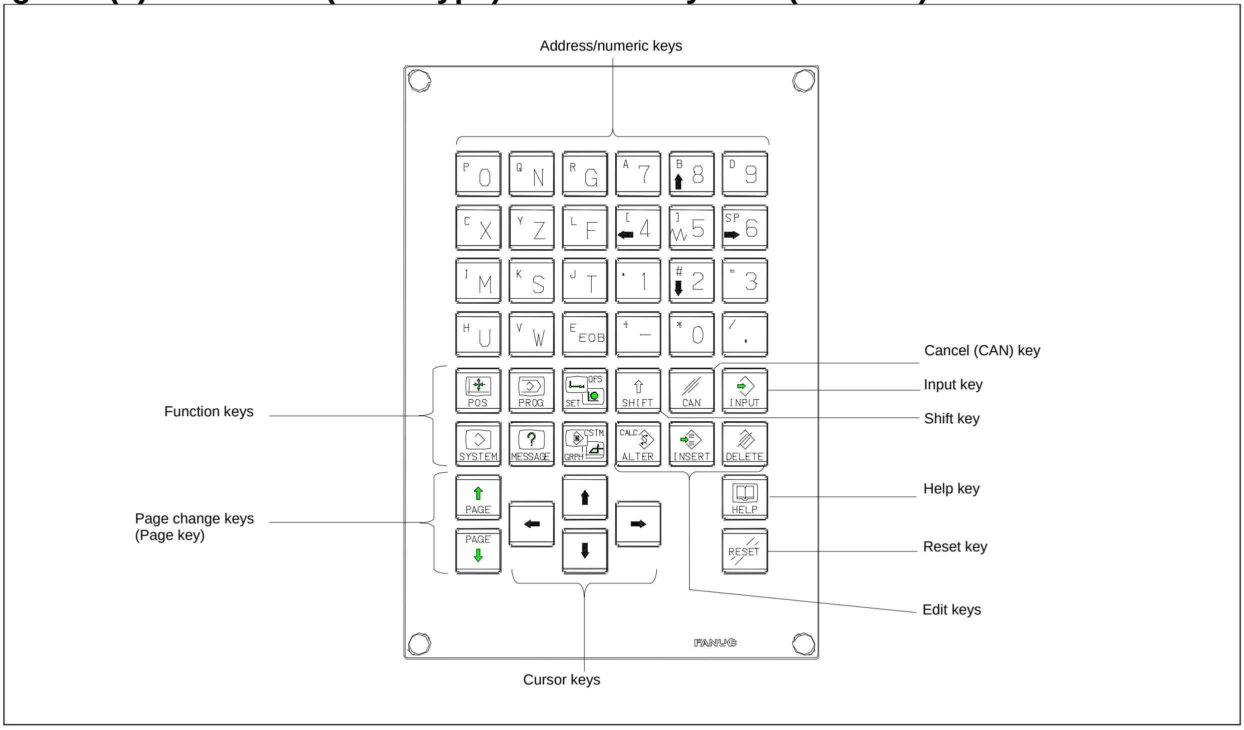


Fig. U18(c) MDI unit (ONG) with machining center system (M series)

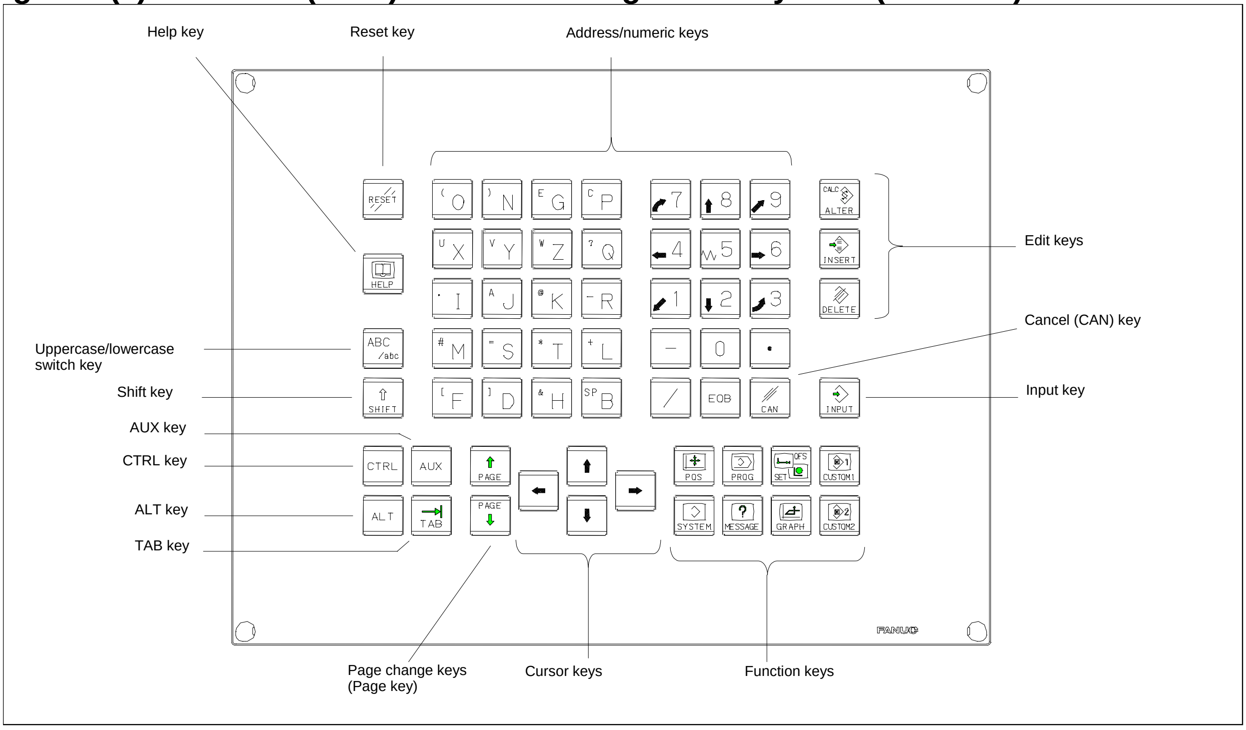


Fig. U18(d) MDI unit (small type) with machining center system (M series)

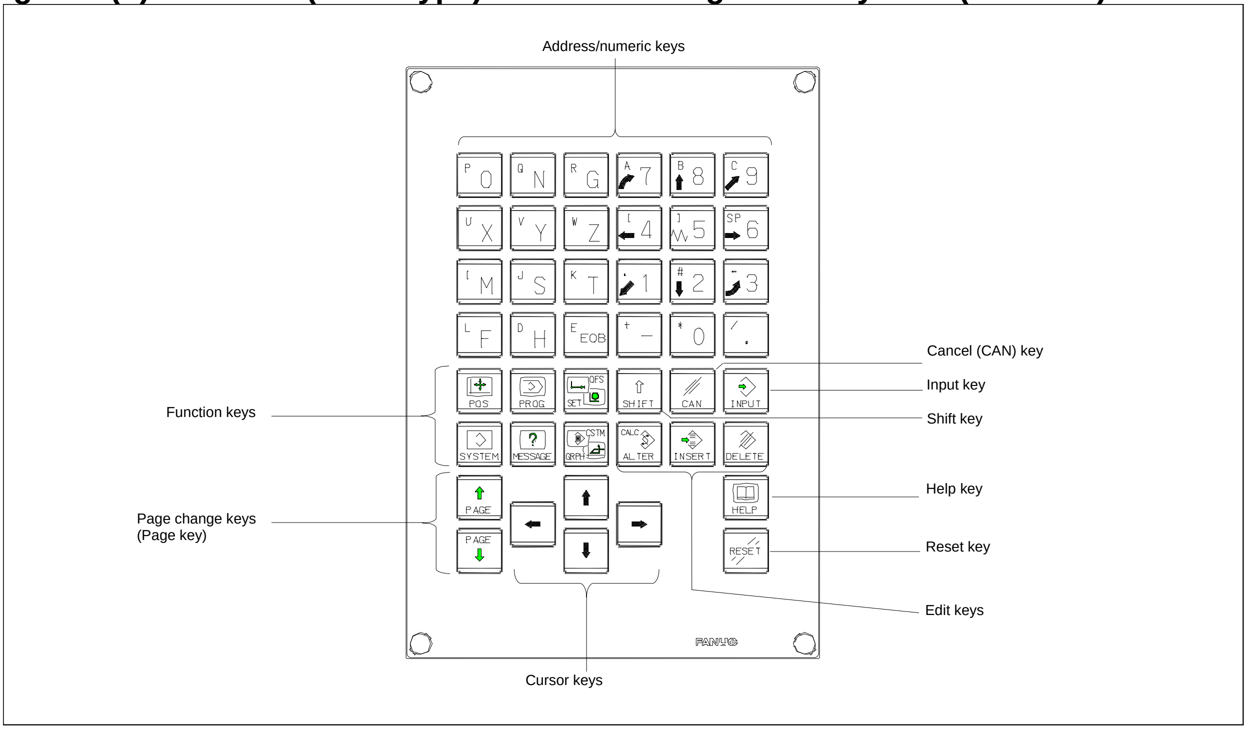


Fig. U18(e) MDI Unit (QWERTY TYPE A) for common to lathe system/machining center system

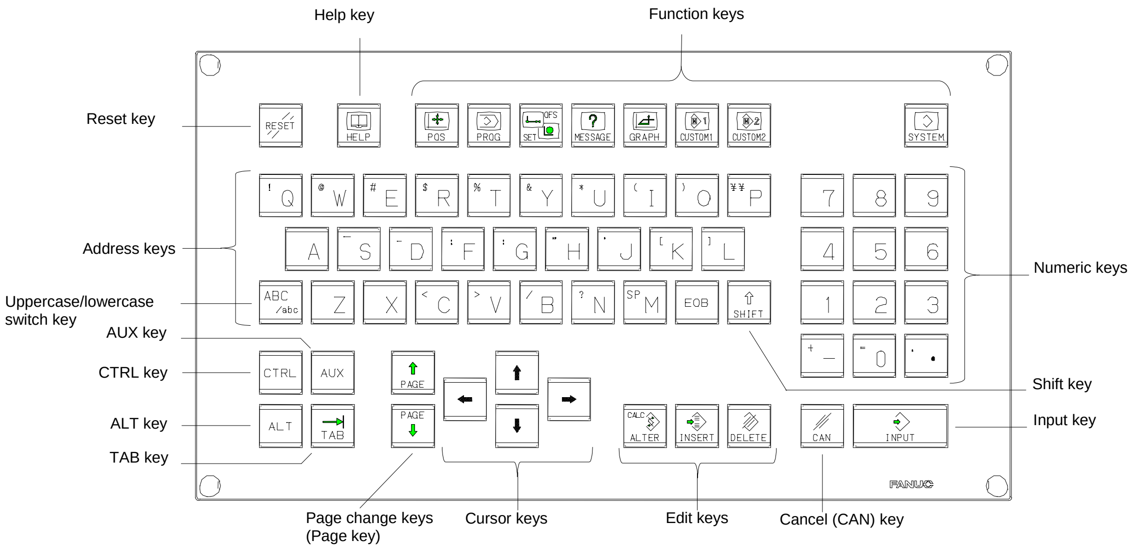
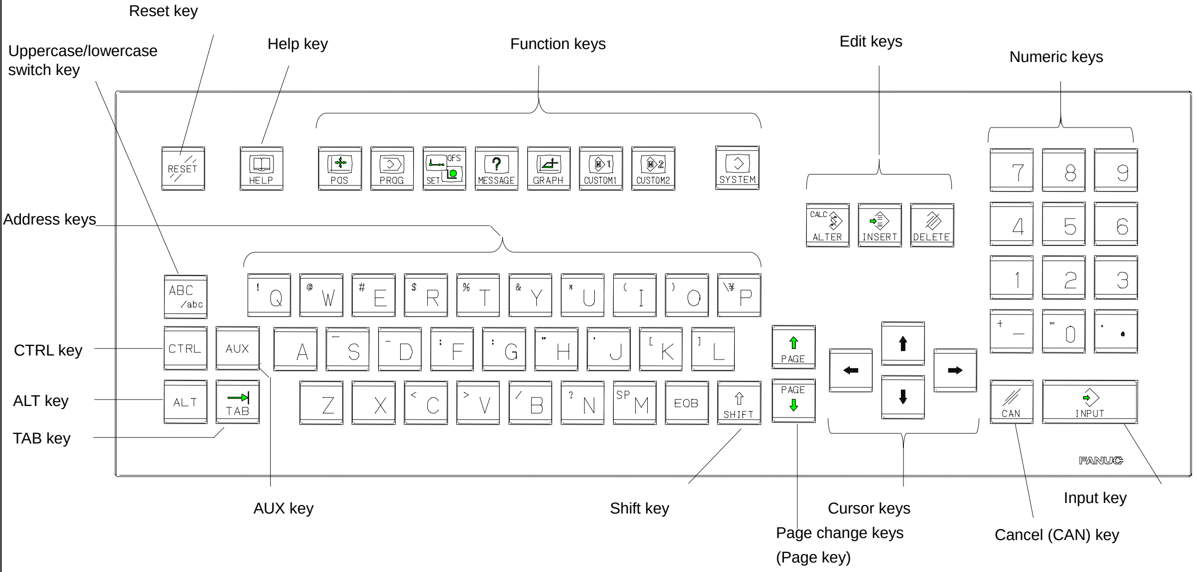


Fig. U18(f) MDI Unit (QWERTY TYPE B) for common to lathe system/machining center system

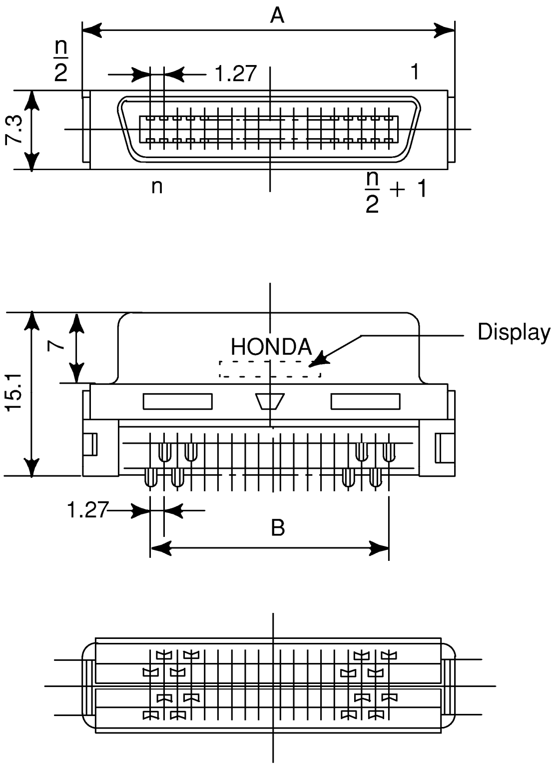


Connectors

Fig. title	Specification No.	Fig. No.
PCR connector (soldering type)	PCR-E20FS	Fig. C1 (a)
FI40 connector	FI40-2015S	Fig. C1 (b)
Connector case (PCR type manufactured by HONDA TSUSHIN KOGYO)	PCR-V20LA/PCR-V20LB	Fig. C2 (a)
Connector case (FI type manufactured by HIROSE ELECTRIC)	FI-20-CV	Fig. C2 (b)
Connector case (Component type manufactured by FUJITSU）	FCN-240C20-Y/S	Fig. C2 (c)
Connector case (PCR type manufactured by HIROSE ELECTRIC)	FI-20-CV7	Fig. C2 (d)
Connector (1) for servo side manufactured by Tyco Electronics	AMP1-178128-3	Fig. C3 (a)
Connector (2) for servo side manufactured by Tyco Electronics	AMP2-178128-3	Fig. C3 (b)
Connector (3) for +24 V power supply manufactured by Tyco Electronics	AMP1-178288-3	Fig. C3 (c)
Connector (4) for +24 V power supply manufactured by Tyco Electronics	AMP2-178288-3	Fig. C3 (d)
Contact for connector manufactured by Tyco Electronics	AMP1-175218-2/5 AMP1-175196-2/5	Fig. C3 (e)
Connector (case) manufactured by HONDA TSUSHIN KOGYO		Fig. C4 (a)
Connector (angled case) manufactured by HONDA TSUSHIN KOGYO		Fig. C4 (b)
Connector (male) manufactured by HONDA TSUSHIN KOGYO		Fig. C4 (c)
Connector (female) manufactured by HONDA TSUSHIN KOGYO		Fig. C4 (d)
Connector (terminal layout) manufactured by HONDA TSUSHIN KOGYO		Fig. C4 (e)
Connector (3 pins/brown) manufactured by SOURIAU Japan KK	SMS3PN-5	Fig. C5
Connector for flat cable manufactured by HIROSE ELECTRIC	HIF3BB-50D-2.54R HIF3BB-34D-2.54R	Fig. C6
Connector (for MDI (CA55)) manufactured by Japan Aviation Electronics	LY10-DC20	Fig. C7 (a)
Contact (for MDI (CA55)) manufactured by Japan Aviation Electronics	LY10-C2-3	Fig. C7 (b)
Punch panel connector for RS232C interface		Fig. C8 (a)
Locking plate for RS232C interface connector		Fig. C8 (b)
Connector (for distribution I/O connection printed circuit board) manufactured by HONDA TSUSHIN KOGYO	MRH-50FD	Fig. C9
Faston terminal	A02B-0166-K330	Fig. C11

Fig. C1 (a) PCR connector (soldering type)

TYPE	:	HONDA PCR–E20FS (SOLDERING TYPE)
USAGE	:	GENERAL
MATING		
HOUSING	:	HONDA PCR–V20LA (PLASTIC)



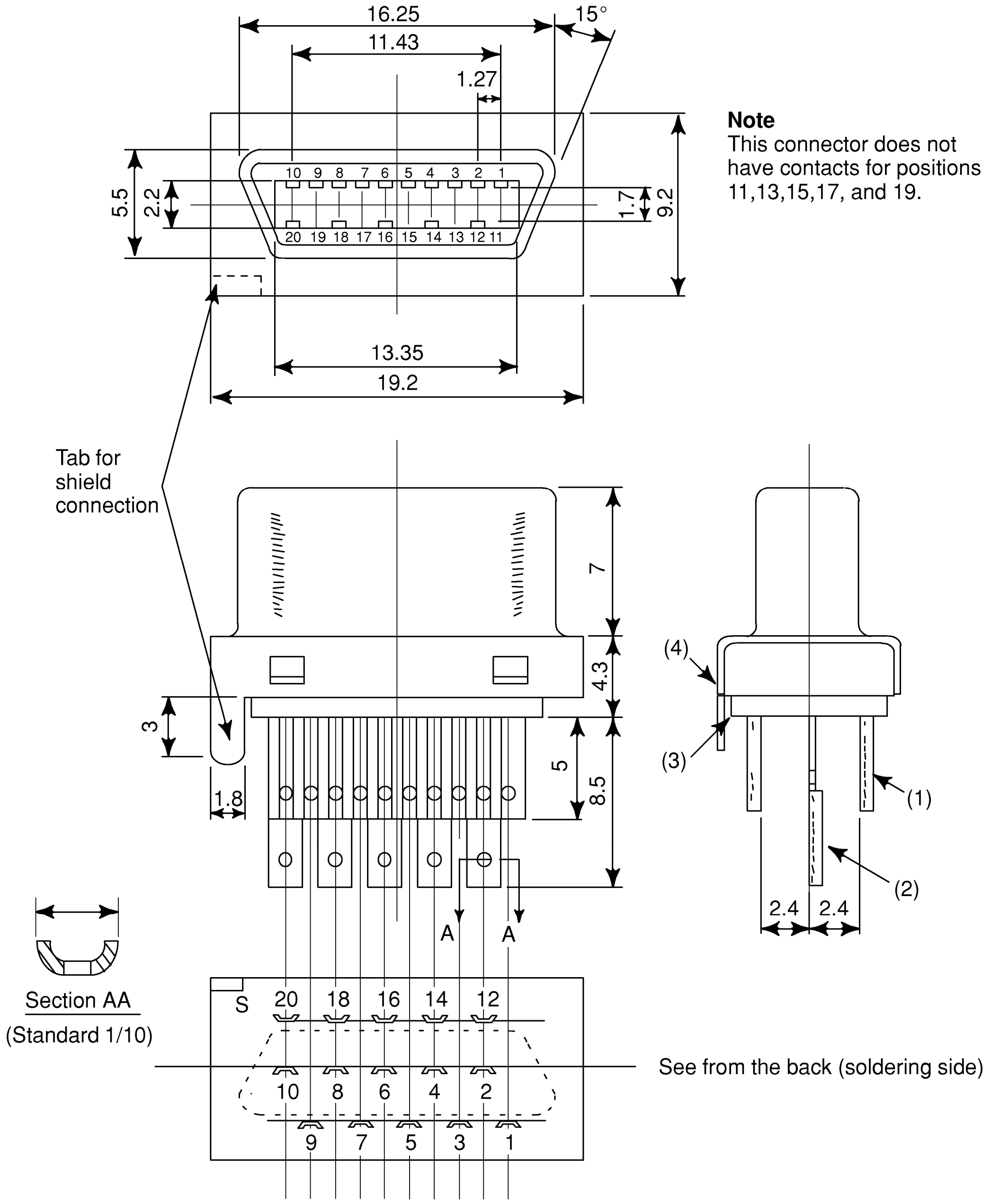
	A	B
PCR–E20FS	21.65	11.43

Fig. C1 (b) FI40 connector

TYPE : HIROSE FI40–2015S

USAGE : PULSE CODER INTERFACE
LINEAR SCALE INTERFACE
MPG INTERFACE

MATING/HOUSING : HIROSE FI–20–CV



Note
This connector does not have contacts for positions 11,13,15,17, and 19.

Fig. C2 (a) Connector case (PCR type manufactured by HONDA TSUSHIN KOGYO)

TYPE	:	HONDA PCR–V20LA (for 6 dia. cable)
USAGE	:	GENERAL

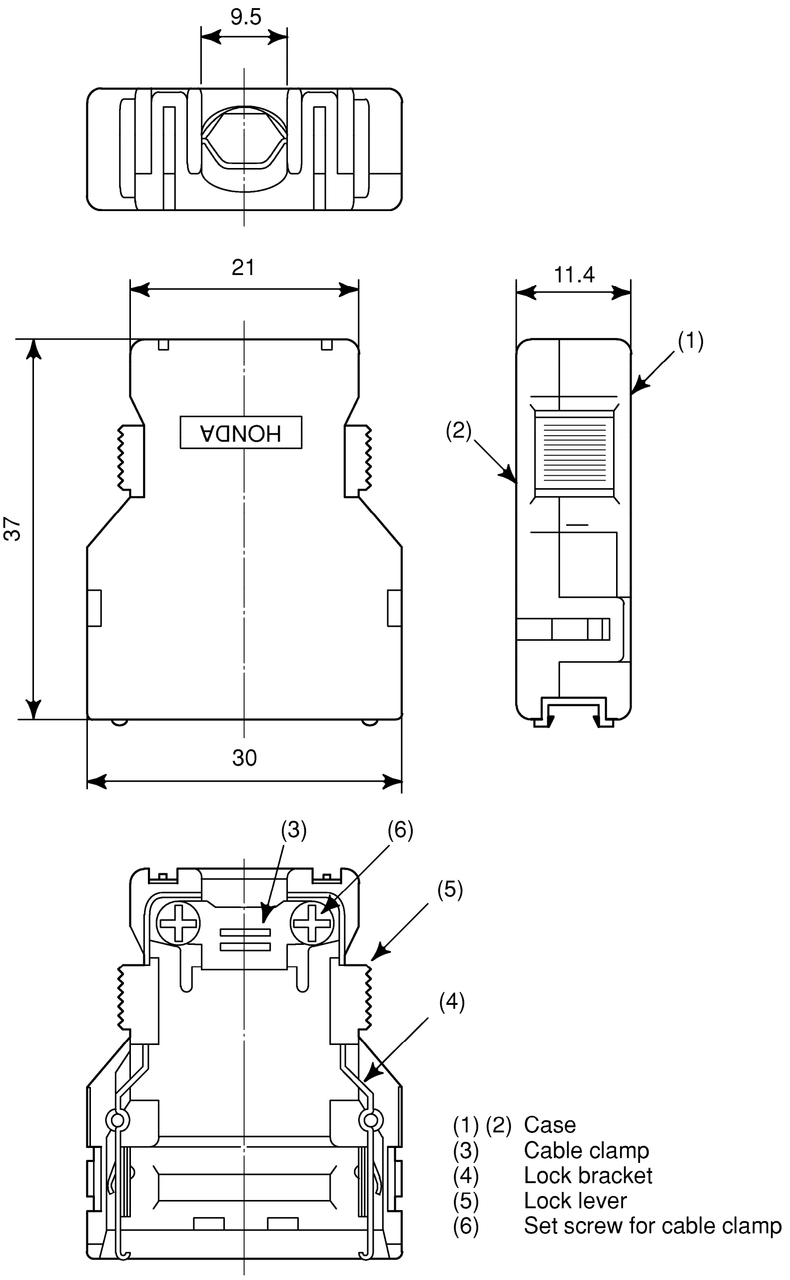


Fig. C2 (b) Connector case (FI type manufactured by HIROSE ELECTRIC)

TYPE : HIROSE FI–20–CV
USAGE : PULSE CODER INTERFACE
LINEAR SCALE INTERFACE
MANUAL PULSE GENERATOR INTERFACE

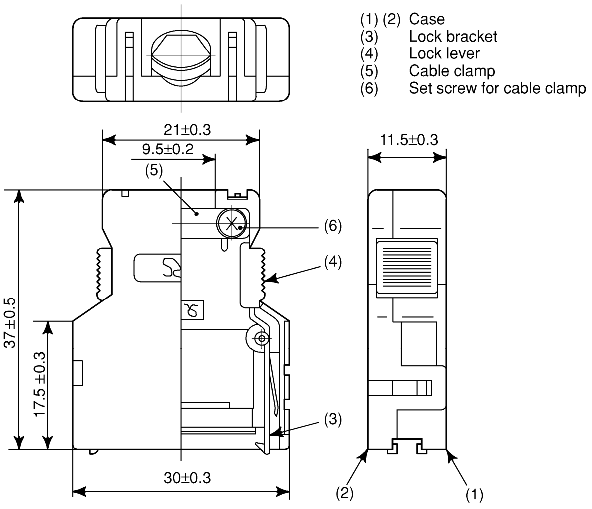


Fig. C2 (c) Connector case (Component type manufactured by FUJITSU)

TYPE : FUJITSU FCN–240C20–Y/S (for 5.8 dia. cable)
USAGE : GENERAL

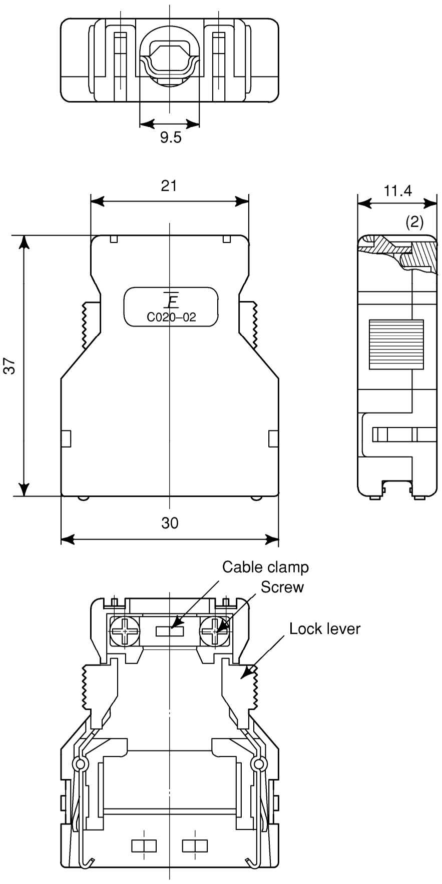


Fig. C2 (d) Connector case (PCR type manufactured by HIROSE ELECTRIC)

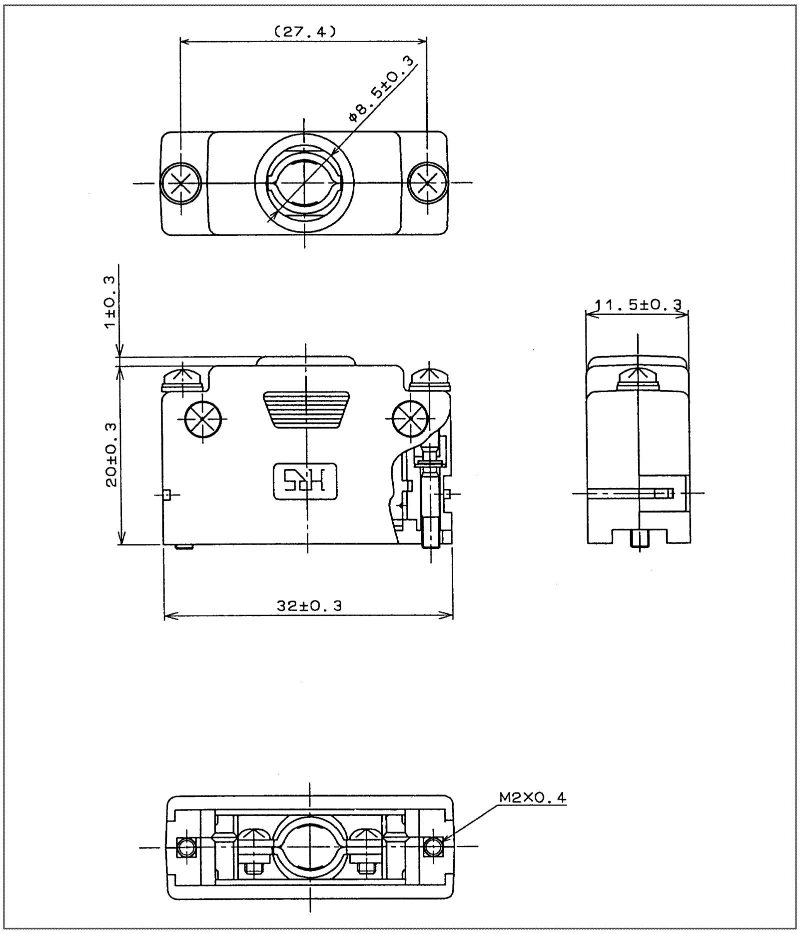


Fig. C3 (a) Connector (1) for servo side manufactured by Tyco Electronics

TYPE : AMP1–178128–3

DIMENSION

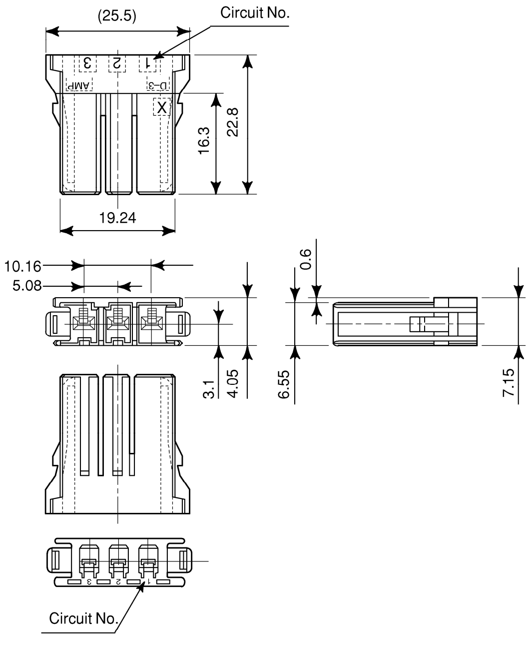


Fig. C3 (b) Connector (2) for servo side manufactured by Tyco Electronics

TYPE : AMP2–178128–3

DIMENSION

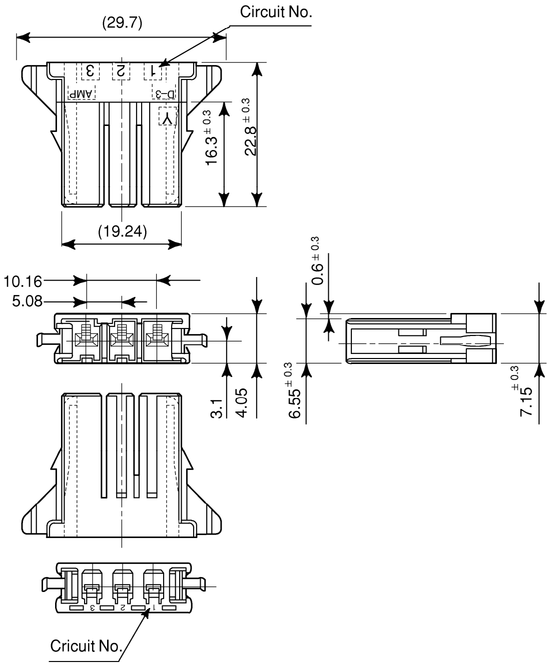


Fig. C3 (c) Connector (3) for +24 V power supply manufactured by Tyco Electronics

TYPE : AMP1–178288–3

USAGE : POWER SUPPLY UNIT CP1A
+24V INPUT

3	
2	0V
1	+24V

DIMENSION

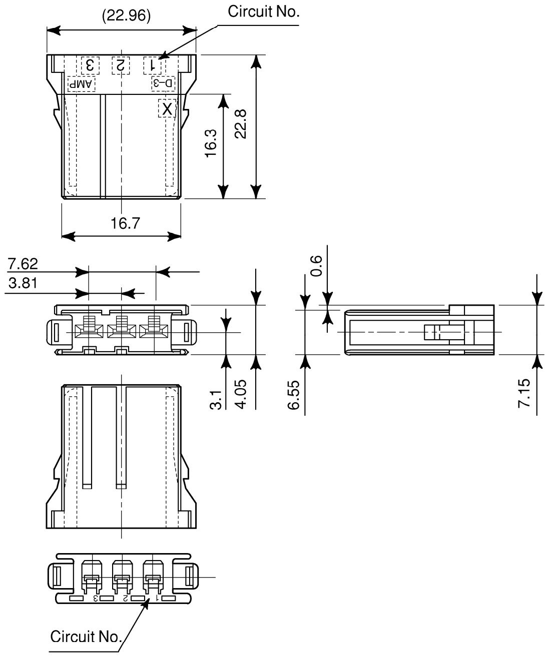


Fig. C3 (d) Connector (4) for +24 V power supply manufactured by Tyco Electronics

TYPE : AMP2–178288–3

USAGE : POWER CP1B
+24V OUTPUT

3	
2	0V
1	+24V

DIMENSION

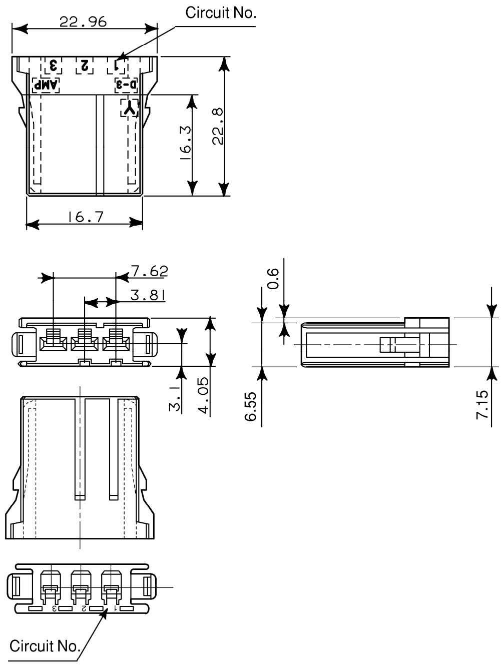


Fig. C3 (e) Contact for connector manufactured by Tyco Electronics

SEPARATE : AMP1–175218–2(Gold coated)
AMP1–175218–5(Tin coated)

REEL : AMP1–175196–2(Gold coated)
AMP1–175196–5(Tin coated)

WIRE : AWG 16, 18, 20

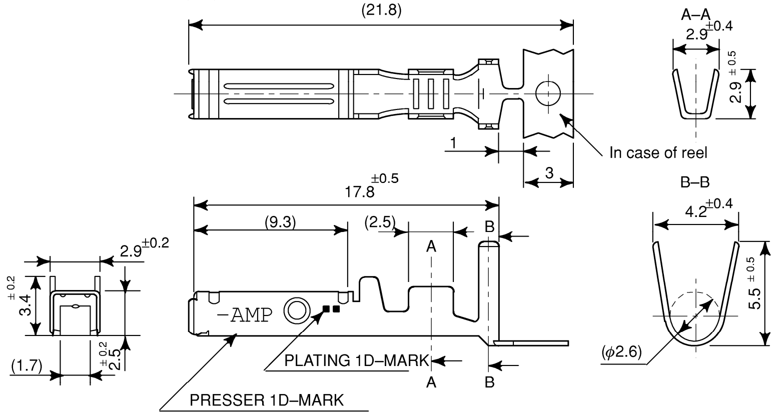
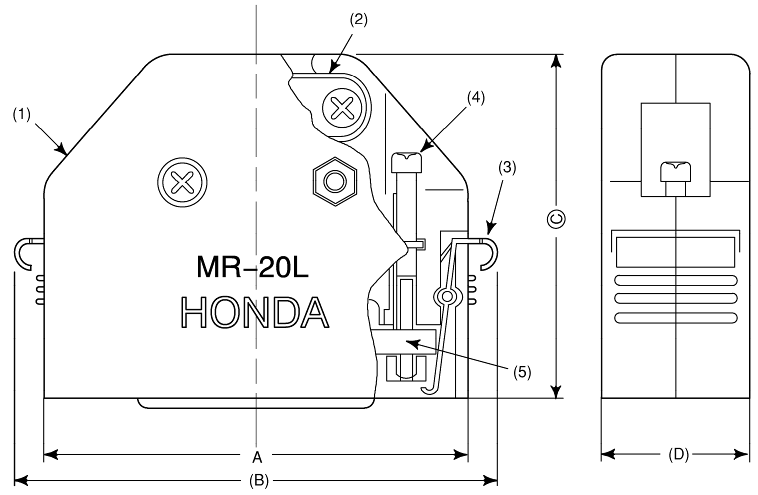


Fig. C4 (a) Connector (case) manufactured by HONDA TSUSHIN KOGYO

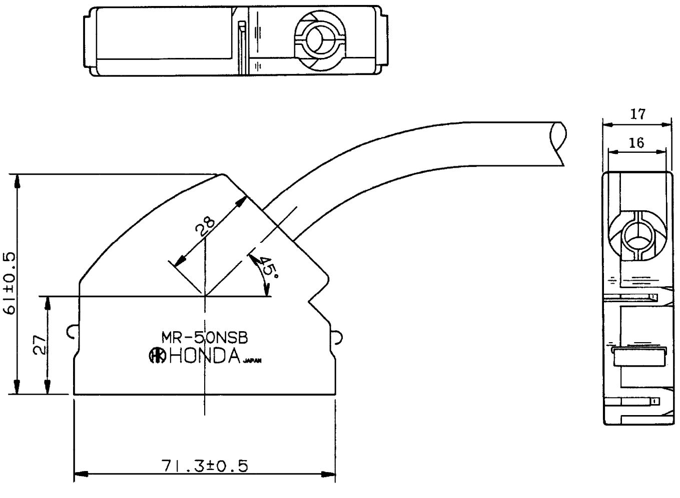


Symbol Specification	A	(B)	C	(D)	Number of terminals
MR–20LMH (Plug)	39.3	44.9	39.8	17	20
MR–20LFH (Jack)					
MR–50LMH (Plug)	67.9	73.5	44.8	18	50
MR–50LFH (Jack)					

Symbol	Name
1	Connector cover
2	Cable clamp
3	Stopper
4	Screw for cable clamp
5	Plug (MR–20, 50MH)
	Jack (MR–20, 50FH)

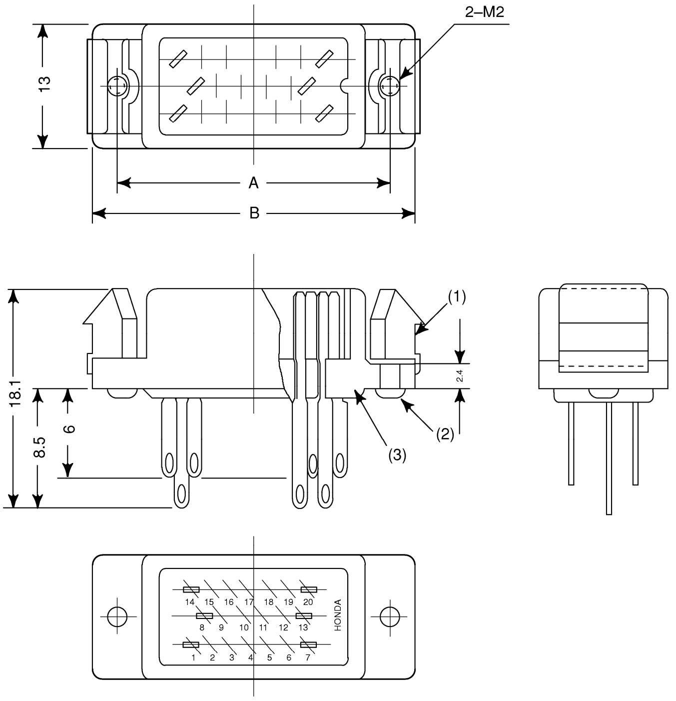
Outer diameter of the cable
MR–20L dia.10mm max
MR–50L dia.16mm max

Fig. C4 (b) Connector (angled case) manufactured by HONDA TSUSHIN KOGYO



External dimensions of 50–pin connector

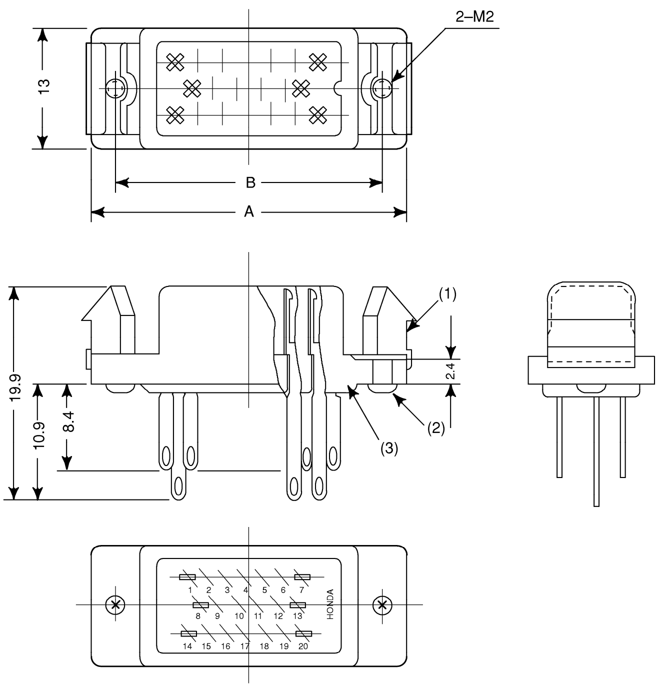
Fig. C4 (c) Connector (male) manufactured by HONDA TSUSHIN KOGYO



	A	B	Number of terminals
MR–20RMH	32.8	27.8	20
MR–50RHF	61.4	56.4	50

Symbol	Name
1	Cable clamp
2	Screw 2.6dia.×8
3	Connector (MR–20,–50MH)

Fig. C4 (d) Connector (female) manufactured by HONDA TSUSHIN KOGYO



	A	B	Number of terminals
MR–20RMH	32.8	27.8	20
MR–50RMH	61.4	56.4	50

Symbol	Name
1	Cable clamp
2	Screw 2.6dia.×8
3	Connector (MR–20,– 50FH)

Fig. C4 (e) Connector (terminal layout) manufactured by HONDA TSUSHIN KOGYO

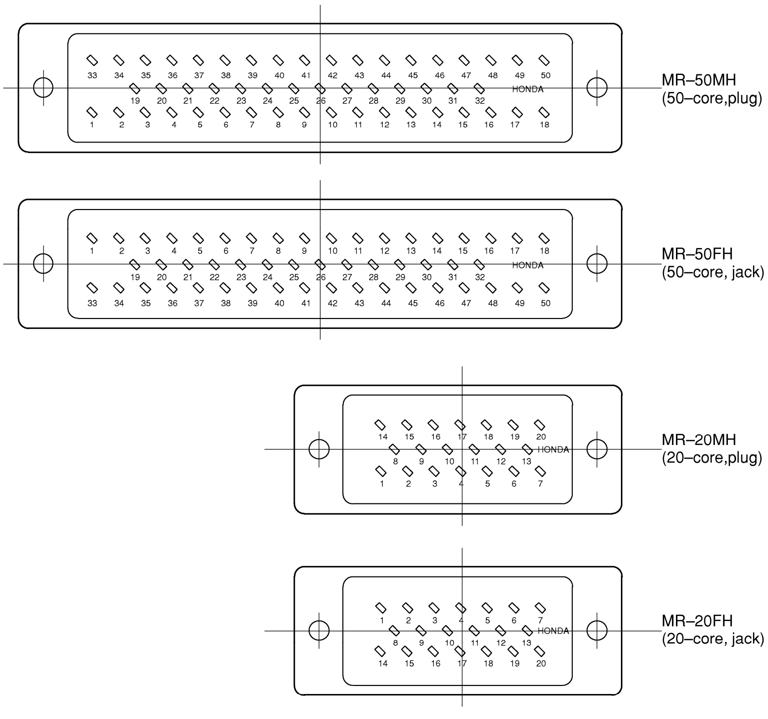
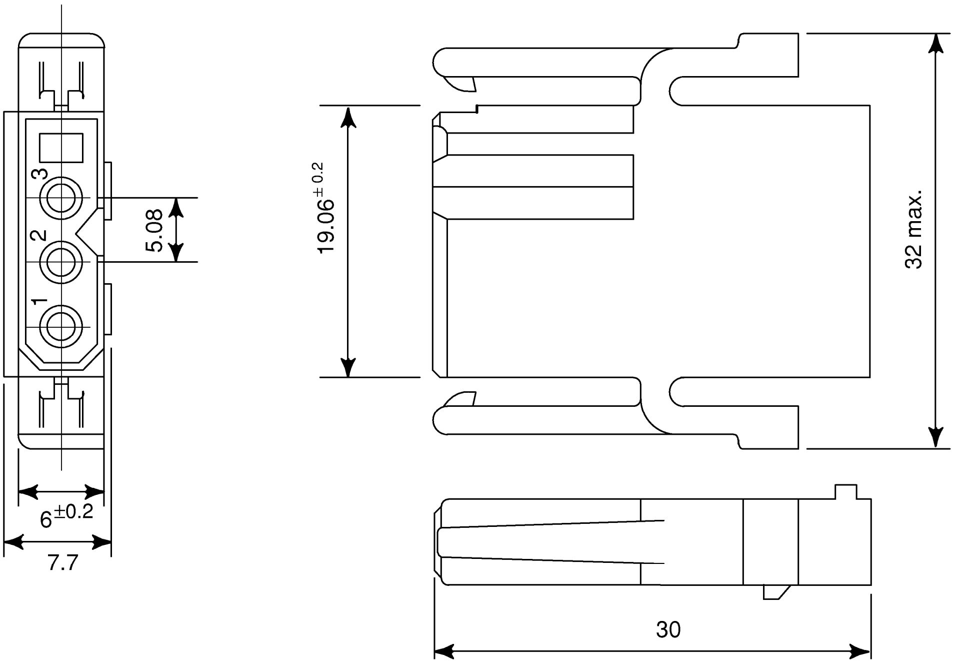


Fig. C5 Connector (3 pins/brown) manufactured by SOURIAU Japan KK



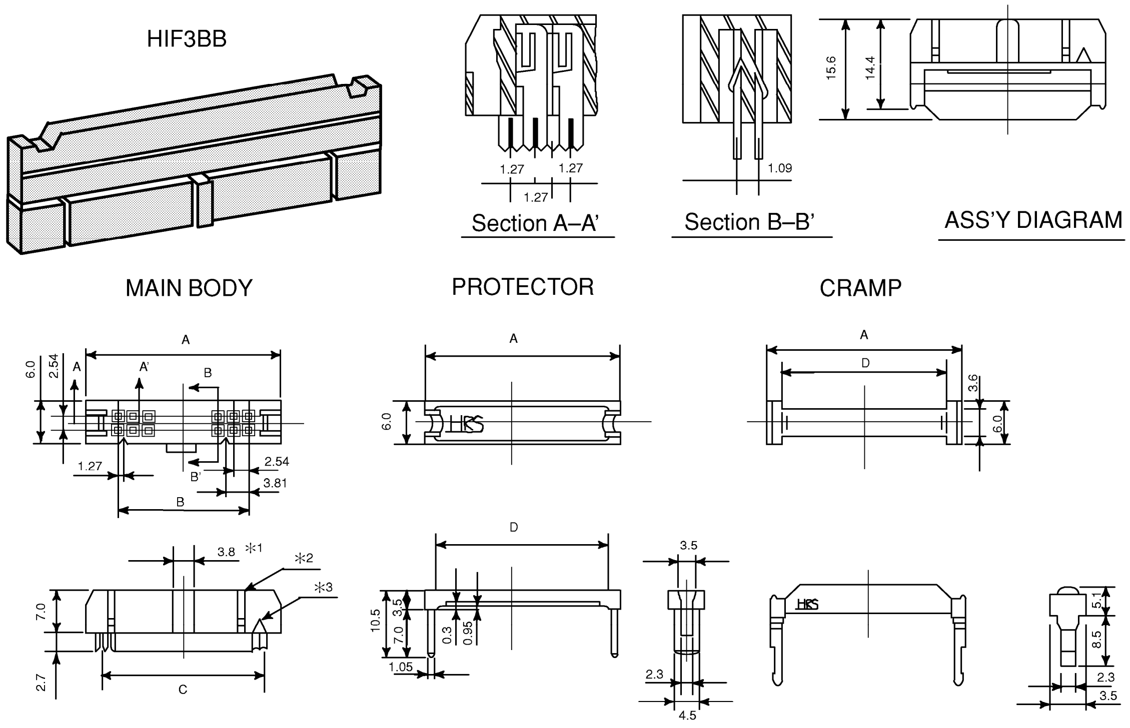
Manufacturer : Burndy Japan Corp.

Name		Specification (Connector maker number)	Remarks
Connector housing for cable		SMS3PNS–5	Brown
Contact	(Crimp type)	RC16M–23T3	For details on tools required for crimp terminals,contact the manufacturer.
	(Solder type)	RC16M–SCT3	

Cables : Cross sectional area : 0.75mm^2(30/0.18)
Insulation diameter : 2.8mm max
Peeling length : 7.2mm

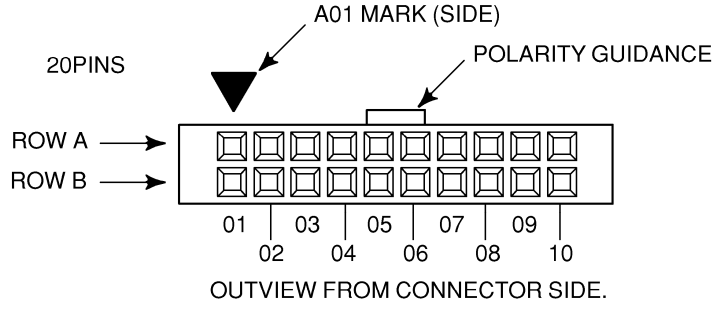
Fig. C6 Connector for flat cable manufactured by HIROSE ELECTRIC

CONNECTOR FOR FLAT CABLE (HIROSE ELEC. CO.)

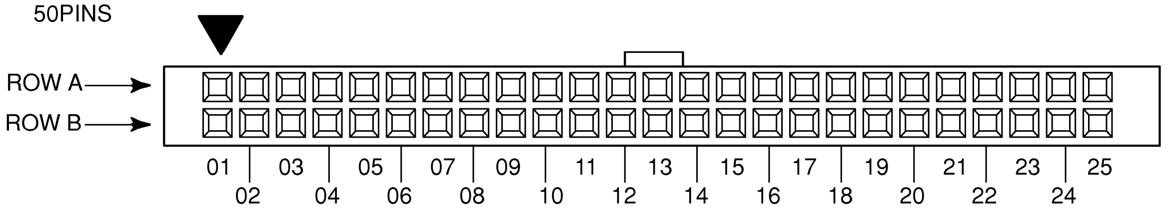


Dimensions

Description	No.of contact	A	B	C	D
HIF3BB–34D–2.54R	34	47.75	40.64	41.91	43.23
HIF3BB–50D–2.54R	50	68.07	60.96	62.23	63.6



OUTVIEW FROM CONNECTOR SIDE.



OUTVIEW FROM CONNECTOR SIDE.

Fig. C7 (a) Connector (for MDI (CA55)) manufactured by Japan Aviation Electronics

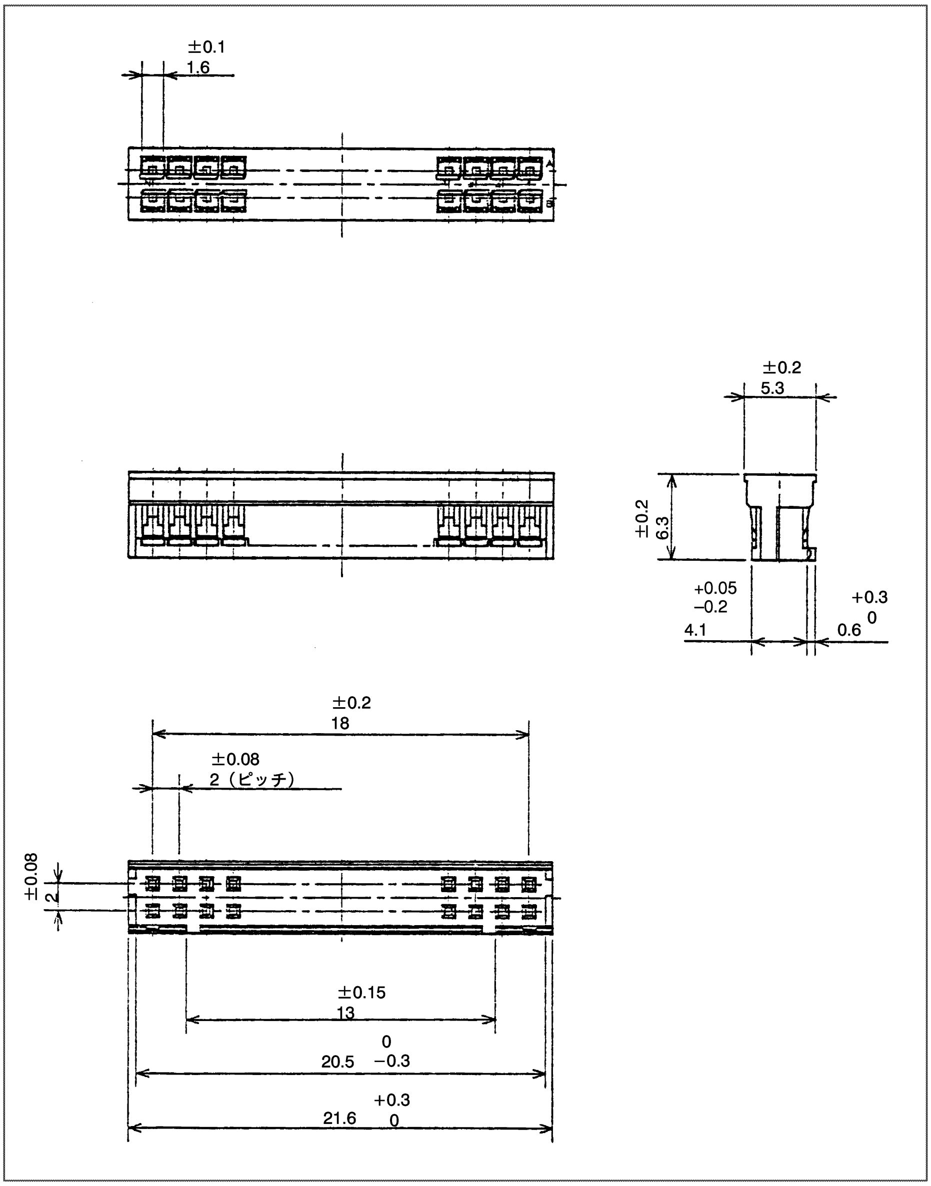


Fig. C7 (b) Contact (for MDI (CA55)) manufactured by Japan Aviation Electronics

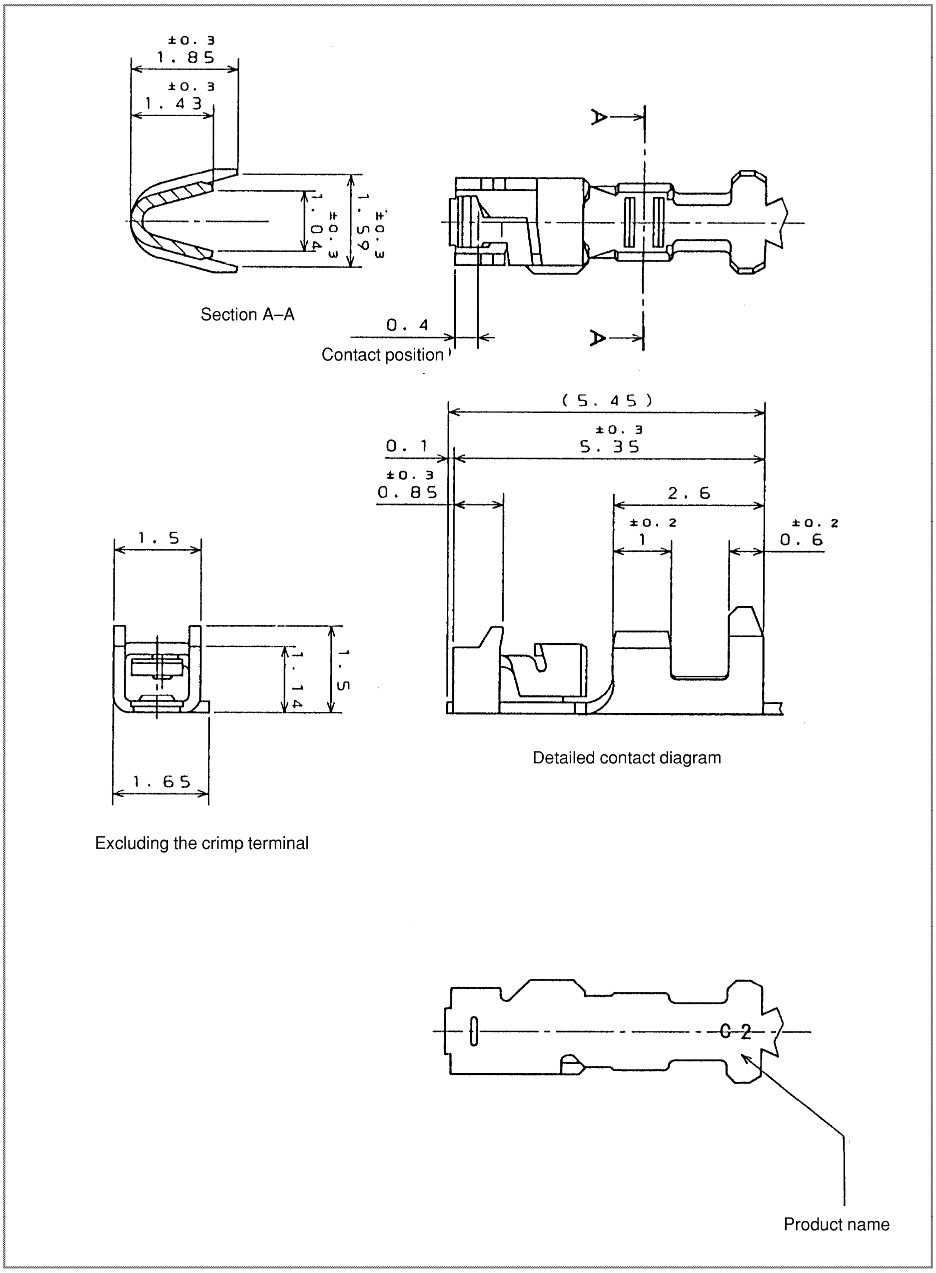


Fig. C8 (a) Punch panel connector for RS232C interface

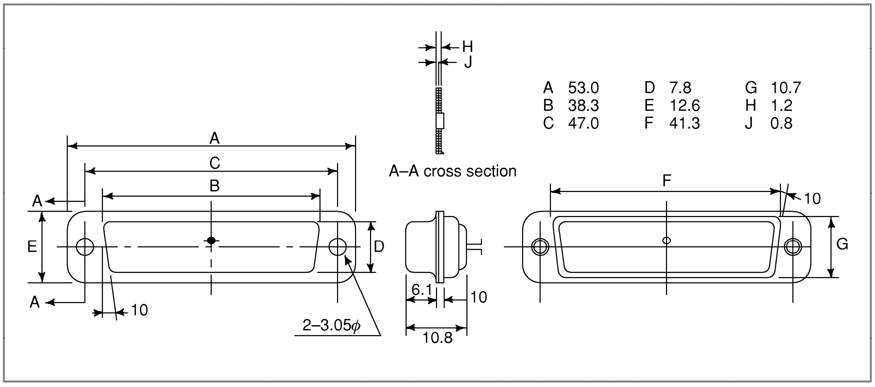


Fig. C8 (b) Locking plate for RS232C interface connector

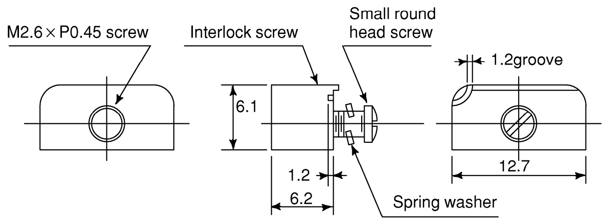
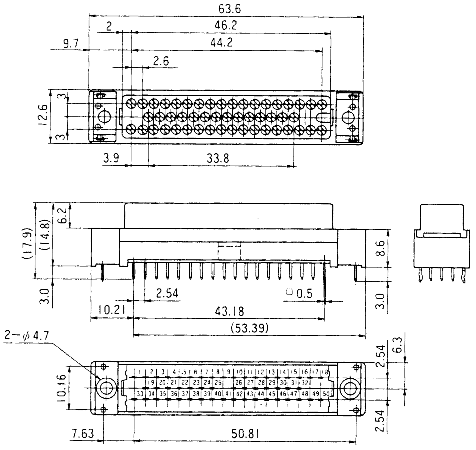


Fig. C9 Connector (for distribution I/O connection printed circuit board) manufactured by HONDA TSUSHIN KOGYO

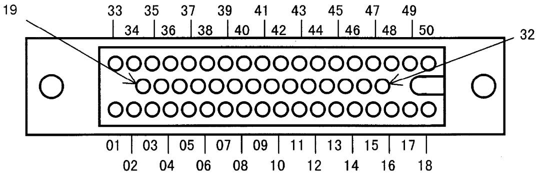
Honda MR type, 50 pins, male, connection printed circuit board soldering type connector

Type No. Honda Tsushin Kogyo Co., Ltd.
MRH–50FD



MRH-50FD

Pin configuration of Honda MR connector, 50 pins, male

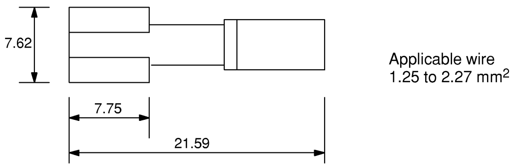


Viewed from the connector side

Fig. C11 Faston terminal

Type:	Faston terminal
Use:	Frame grounding
Applicable housing:	17604-1 manufactured by Tyco Electronics or FVDDF2-250B TYPE Ⅰ manufactured by J.S.T. Mfg.

Dimensions:



INDEX

REVISION RECORD

Edition	Date	Contents
01	Sep., 2014	

B-64602EN/01