FANUC

GE Fanuc Automation

Computer Numerical Control Products

Series 15 / 150 – Model B for Machining Center

Descriptions Manual

GFZ-62082E/04 April 1997

GFL-001

Warnings, Cautions, and Notes as Used in this Publication

Warning

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use.

In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.

Caution

Caution notices are used where equipment might be damaged if care is not taken.

Note

Notes merely call attention to information that is especially significant to understanding and operating the equipment.

DEFINITION OF WARNING, CAUTION, AND NOTE

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

WARNING

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

CAUTION

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

NOTE

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

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

PREFACE

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

Product Name	Abbreviations	
FANUC Series 15–MB	15–MB	Series 15
FANUC Series 15–MFB	15–MFB	
FANUC Series 15MEK–MODEL B–4 (*)	15MEK	
FANUC Series 15MEL–MODEL B–4 (*)	15MEL	
FANUC Series 150–MB	150–MB	Series 150

(*) The FANUC Series 15MEK/MEL–MODEL B–4 is a software–fixed CNC capable of 4 contouring axes switchable out of 8 axes for milling machines and machining centers.
Further the following functions can not be used in the 15MEK or 15MEL.

- Increment system D/E (Increment system C is an option function)
- Helical interpolation B
- Plane switching
- Designation direction tool length compensation
- 2 axes electric gear box
- Manual interruption of 3–dimensional coordinate system conversion
- 3–dimensional cutter compensation
- Trouble diagnosis guidance
- OSI/ETHERNET function
- High–precision contour control using RISC
- Macro compiler (self compile function)
- MMC–III, MMC–IV
- Smooth interpolation
- Connecting for personal computer by high–speed serial–bus

Manuals related to Series 15/150–MODEL B

Manuals related to FANUC Series 15/150–MODEL B are as follows. This manual is marked with an asterisk (*).

List of Manuals Related to Series 15/150–MODEL B

Manual Name	Specification Number	
FANUC Series 15–TB/TFB/TTB/TTFB DESCRIPTIONS	B–62072E	
FANUC Series 15/150–MODEL B For Machining Center DESCRIPTIONS	B–62082E	*
FANUC Series 15/150–MODEL B CONNECTION MANUAL	B–62073E	
FANUC Series 15/150–MODEL B CONNECTION MANUAL (BMI Interface)	B–62073E–1	
FANUC Series 15–MODEL B For Lathe OPERATOR’S MANUAL (Programming)	B–62554E	
FANUC Series 15–MODEL B For Lathe OPERATOR’S MANUAL (Operation)	B–62554E–1	
FANUC Series 15/150–MODEL B For Machining Center OPERATOR’S MANUAL (Programming)	B–62564E	
FANUC Series 15/150–MODEL B For Machining Center OPERATOR’S MANUAL (Operation)	B–62564E–1	
FANUC Series 15/150–MODEL B PARAMETER MANUAL	B–62560E	
FANUC Series 15/150–MODEL B MAINTENANCE MANUAL	B–62075E	
FANUC Series 15–MODEL B DESCRIPTIONS (Supplement for Remote Buffer)	B–62072E–1	
FANUC Series 15–MODEL B PROGRAMMING MANUAL (Macro Compiler / Macro Executer)	B–62073E–2	
PMC		
FANUC PMC–MODEL N/NA PROGRAMMING MANUAL (Ladder Language)	B–61013E	
FANUC PMC–MODEL NB/NB2 PROGRAMMING MANUAL (Ladder Language)	B–61863E	
FANUC PMC–MODEL N/NA PROGRAMMING MANUAL (C Language)	B–61013E–2	
FANUC PMC–MODEL NB PROGRAMMING MANUAL (C Language)	B–61863E–1	
FANUC PMC–MODEL N/NA PROGRAMMING MANUAL (C Language – Tool Management Library)	B–61013E–4	
Conversational Automatic Programming Function		
CONVERSATIONAL AUTOMATIC PROGRAMMING FUNCTION FOR MACHINING CENTER (Series 15–MF/MFB) PROGRAMMING MANUAL	B–61263E	
CONVERSATIONAL AUTOMATIC PROGRAMMING FUNCTION FOR MACHINING CENTER (Series 15–MF/MFB) OPERATOR’S MANUAL	B–61264E	
CONVERSATIONAL AUTOMATIC PROGRAMMING FUNCTION FOR LATHE (Series 15–TF/TTF/TFB/TTFB) OPERATOR’S MANUAL	B–61234E	
CONVERSATIONAL AUTOMATIC PROGRAMMING FUNCTION II FOR LATHE (Series 15–TFB/TTFB) OPERATOR’S MANUAL	B–61804E–2	
Tracing / Digitizing		
FANUC Series 15–MB DESCRIPTIONS (Supplement for Tracing / Digitizing)	B–62472E	
FANUC Series 15–MB CONNECTION MANUAL (Supplement for Tracing / Digitizing)	B–62473E	
FANUC Series 15–MB OPERATOR’S MANUAL (Supplement for Tracing / Digitizing)	B–62474E	
Gas, Laser Plasma Cutting Machine		
FANUC Series 15–MB DESCRIPTIONS (FOR GAS, LASER, PLASMA CUTTING MACHINE)	B–62082EN–1	
Multi–Teaching Function		
FANUC Series 15–MB CONNECTION MANUAL (Multi–Teaching Function)	B–62083E–1	
Multiple–axis and Multiple–path Control Function		
FANUC Series 15–TTB OPERATOR’S MANUAL (Supplement Explanations for Multiple–axis and Multiple–path Control Function)	B–62074E–1	

Table of Contents

APPENDIX

I. GENERAL

1 GENERAL

To achieve high–speed, highly accurate, and highly efficient processes required for future machining needs, the Series 15, an advanced industrial computer, was developed as the high–grade AI–CNC.
It uses surface–mounted electronic parts to enable the control unit to be made compact and the high–speed multi–master bus (FANUC BUS) to standardize all the printed– circuit boards for providing a series of systems in different sizes from small to large.
The sophisticated functions such as the world's fastest CNC function using a 32–bit microprocessor, high–speed and highly accurate digital servo system, and high–speed PMC function provided with the newly developed processor dedicated to PMC allow great enhancement of machining throughput. The high–grade AI–CNC has AI functions such as intelligent failure diagnosis guidance which allow full use of the user's know–how.
The 15–MB with the addition of a man–machine control (MMC) function which enables the incorporation of a high level man–machine interface.

2 LIST OF SPECIFICATIONS

Series 15 has the basic machine interface (BMI), FS3 interface and the FS6 interface and there are some limitations on functions depending on the machine interfaces.
The specification list also informs which functions are effective for each machine interface.
The features of CNCs mentioned here are classified as in the following table and the lists of specifications are written according to this classification.

Classification of specification	Table No. of specification list
Standard specification	Table 2 (a)
Optional specification	Table 2 (b)

Detailed explanations of each function is mentioned in an item of the text indicated in the specification list.

1) List of standard specification
   The list indicates the standard features.
2) List of optional specification
   The list indicated features which can be added to the standard features.

See DESCRIPTIONS (Supplement for Tracong/Digitizing) (B–62472E) of FANUC Series 15–MB for the following functions.

- Tracing functions
- Digitizing functions

See DESCRIPTIONS (B–62082EN–1) (For Gas, Laser, Plasma Cutting Machine) of FANUC Series 15–MB for the following functions.

- Automatic exact stop check
- Gradual curve cutting
- Torch swivel control function
- Error detect function
- Prallel axis control function
- Accelerating/decelerating signal
- Background graphic display

Table 2 (a) Standard specification (1/6)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Controlled axis	3 axes (2 axes also possible)	Same as left	Same as left	II 1.1
Simultaneous controllable axes	2 axes	Same as left	Same as left	II 1.3
Axis name	Optional from X, Y, Z, U, V, W, A, B, C	Same as left	Same as left	II 1.5
Increment system	0.01, 0.001, 0.0001 mm 0.001, 0.0001, 0.00001 inch	Same as left	Same as left	II 1.7
Interpolation unit	0.005, 0.0005, 0.00005 mm, 0.0005, 0.00005, 0.000005 inch	Same as left	Same as left	Appendix A
Maximum commandable value	±8 digits	Same as left	Same as left	Appendix A
High resolution detection interface	YES	YES	YES	
Positioning	Linear interpolation type positioning is also available	Same as left	Same as left	II 3.1
Linear interpolation	YES	YES	YES	II 3.3
Multi–quadrant circuit interpolation	YES	YES	YES	II 3.4
Rapid traverse rate	YES	YES	YES	II 5.1
Tangential speed constant control	YES	YES	YES	II 5.2.1
Cutting feedrate clamp	For each axis	Same as left	Same as left	II 5.2.2
Feed per minute	YES	YES	YES	II 5.2.3
Feedrate override	0–254% 1% step	0–150% 10% step	0–200% 10% step	II 5.3.1
Rapid traverse override	F0, F1, 50%, 100%	Same as left	Same as left	II 5.3.4
Function for overriding the rapid traverse feedrate in 1% units	YES	YES	YES	II 5.3.5
Automatic acceleration/ deceleration	Rapid traverse: Linear acceleration/deceleration Cutting feed: Exponential acceleration/ deceleration	Same as left	Same as left	II 5.4
Linear acceleration/ deceleration after cutting feed interpolation	YES	YES	YES	II 5.5

Table 2 (a) Standard specification (2/6)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Acceleration/deceleration before cutting feed interpolation	YES	YES	YES	II 5.7
Acceleration/deceleration prior to pre–read interpolation	YES	YES	YES	II 5.8
Bell–shaped acceleration/ deceleration after rapid traverse interpolation	YES	YES	YES	II 5.9
Cutting/rapid traverse position check function	YES	YES	YES	II 5.13
Exact stop, Exact stop mode Cutting mode Tapping mode	YES	YES (Tapping mode signal is not available)	YES	II 5.14–5.16
Dwell	Per second dwell and per revolution dwell	Same as left	Same as left	II 5.18
Reference position return	Manual, automatic (G27, G28, G29)	Same as left	Same as left	II 6.1 II 6.2 II 6.3
Reference position automatic setting function	YES	NO	NO	II 6.6
Dog–less reference position setting function	YES	YES	YES	II 6.7
Machine coordinate system selection (G53)	YES	YES	YES	II 7.1
Workpiece coordinate system selection (G54–G59)	YES	YES	YES	II 7.2
Local coordinate system setting (G52)	YES	YES	YES	II 7.3
Workpiece coordinate system change (G92)	YES	YES	YES	II 7.4
Workpiece coordinate system presetting	YES	YES	YES	II 7.7
Absolute/incremental programming	Can be combined in the same block	Same as left	Same as left	II 8.1
Decimal point input/pocket calculator type decimal point input	YES	YES	YES	II 8.4
Function for switching between diameter and radius programming	YES	YES	YES	II 8.6

Table 2 (a) Standard specification (3/6)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
S code output	S8–digit command (Binary output)	S2–digit command (BCD output)	Same as left	II 9.1
T code output	T8–digit command (Binary output)	T2–digit command (BCD output)	Same as left	II 10.1
Miscellaneous function	M8–digit command (Binary output)	M2–digit (BCD output)	M3–digit (BCD output)	II 11.1
High speed M/S/T/B interface	YES	NO	NO	II 11.3
Program number/program name	Program number: 4 digits Program name: 16 characters	Same as left	Same as left	II 12.1 II 12.2
Program number search	YES	YES	YES	II 12.1
Main program/subprogram	Subprogram: Fourfold nesting	Same as left	Same as left	II 12.4 II 12.5
Sequence number	5–digit	Same as left	Same as left	II 12.6
Sequence number search	YES	YES	YES	II 12.6
Tape code	EIA RS244, ISO840 automatic recognition	Same as left	Same as left	II 12.7
Command format	Word–address format	Same as left	Same as left	II 12.9
Label skip	YES	YES	YES	II 12.10
Control in/out	YES	YES	YES	II 12.11
Optional block skip	YES	YES	YES	II 12.12
Circular interpolation radius programming	YES	YES	YES	II 13.6
Circle cutting function	YES	YES	YES	II 13.10
Tool length compensation	YES	YES	YES	II 14.1
Tool offset amount memory A	Common to all tools	Same as left	Same as left	II 14.6.1
32 tool offsets	YES	YES	YES	II 14.7
Incremental offset input	YES	YES	YES	
Backlash compensation	Max. 9999 pulses	Same as left	Same as left	II 15.6
Tool length measurement	YES	YES	YES	III 17.7
Automatic corner deceleration	YES	YES	YES	II 19.3
Feedrate clamp by circular radius	YES	YES	YES	II 19.4
Advanced preview control function	YES	YES	YES	II 19.5

Table 2 (a) Standard specification (4/6)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Follow–up	YES	YES	YES	II 20.1
Follow–up for each axis	YES	YES	YES	II 20.2
Servo off and mechanical handle feed	YES	YES	YES	II 20.3 II 20.4
External mirror image	Possible on all axes	Possible on all axes	Possible on all axes	II 20.5
Controlled axis detach	YES	YES	YES	II 20.6
Roll–over function for a rotation axis	YES	YES	YES	II 20.16
Automatic operation	Tape operation/Memory operation/MDI operation	Same as left	Same as left	II 21
Cycle start/feed hold	YES	YES	YES	II 21.3.1 II 21.5.4
Buffer register	YES	YES	YES	II 21.4.1
Program stop/program end	YES	YES	YES	II 21.5.1 II 21.5.2
Reset and rewind	YES	YES	YES	II 21.2.4 II 21.5.5
Transverse inhibit limit	YES	NO	NO	II 21.10
Jog feed	YES	YES	YES	II 22.1
Incremental feed	×1, ×10, ×100, ×1000, ×10000, ×100000	×1, ×10, ×100, ×1000	×1, ×10, ×100, ×1000, ×10000, ×100000	II 22.2
Manual absolute on/off	YES	YES	YES	II 22.7
Manual operation stored stroke check	YES	NO	NO	II 22.9
Machine lock on all axes	YES	YES	YES	II 23.1
Machine lock on each axis	YES	YES	YES	II 23.2
Auxiliary function lock	YES	YES	YES	II 23.3
Dry run	YES	YES	YES	II 23.4
Single block	YES	YES	YES	II 23.5
Retrace program editing function	YES	NO	NO	II 23.6
Keyboards type manual data input (MDI), CRT character display	9″ monochrome (Note)	Same as left	Same as left	II 24.1

NOTE
The applicable display unit is limited.

Table 2 (a) Standard specification (5/6)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Clock function	YES	YES	YES	II 24.7
Run hour and parts number display	YES	YES	YES	II 24.8
Load meter display	YES	YES	YES	II 24.9
NC format guidance	YES	YES	YES	II 24.13
NC format guidance with figure	YES (Note)	YES (Note)	YES (Note)	II 24.14
Data protection key	3 types	1 type	1 type	II 24.16
Directory display and punching on each group	YES	YES	YES	II 24.19
Function for displaying multiple subscreens	YES	YES	YES	II 24.20
Help function	YES	YES	YES	II 24.21
Parameter setting (RS–232–C) screen	YES	YES	YES	II 24. 22
Screen for specifying high–speed and high–precision machining	YES	YES	YES	II 24.23
Operation history	YES	YES	YES	II 24.24
Waveform diagnosis function	YES	YES	YES	II 24.25
CRT screen saving function	YES	YES	YES	II 24.26
Workpiece origin manual setting	YES	NO	NO	II 24.28
Screen saver	YES	NO	NO	II 24.29
Part program storage & editing	YES	YES	YES	II 25
Expanded part program editing	YES	YES	YES	II 25.3
Background editing	YES	YES	YES	II 25.2
Part program storage length	80 m	Same as left	Same as left	II 25.5

NOTE
The applicable display unit is limited.

Table 2 (a) Standard specification (6/6)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Resisterable programs (Program name display is also possible)	100	100	100	II 25.4
High–speed part program registeration function	YES	YES	YES	II 25.9
Function selection with hard keys	YES	YES	YES	II 25.10
Multi–edit function	YES	YES	YES	II 25.11
Self–diagnosis functions	YES	YES	YES	II 26.1
Emergency stop	YES	YES	YES	II 28.1
Overtravel	YES	YES	YES	II 28.2.1
Stored stroke check 1	YES	YES	YES	II 28.2.2
Interlock	Each axis/all axes/all axes in automatic operation/ block start/cutting block start interlock	All axes or Z–axis only	Each axis, cutting block start	II 28.3
Status output	NC ready, servo ready, rewinding, alarm, distribution end, automatic operation, automatic operation start lamp, feed hold reset, im-position, rapid traversing, tapping, constant surface speed control, inch input and DI status	NC ready, servo ready, rewinding, alarm, distribution end, automatic operation start lamp, feed hold, reset	Same as left	II 29
Connectable servo motor	FANUC AC SERVO MOTOR series	Same as left	Same as left	II 36
Connectable servo unit	PWM transistor drive	Same as left	Same as left	II 36
Connectable position detector	Pulse coder/optical scale	Same as left	Same as left	II 37
Absolute position detector	YES	YES	YES	II 37
Connectable spindle motor	FANUC AC SPINDLE MOTOR series	Same as left	Same as left	II 38
Connectable spindle servo unit	PWM transistor drive	Same as left	Same as left	II 38
Power (About the CNC unit)	200 to 240 VAC +10%, –15% 50 to 60 Hz ±3 Hz	Same as left	Same as left	II 35.2

Table 2 (b) Optional specification (1/8)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Controlled axes expansion Name of axes: Select from X, Y, Z, A, B, C, U, V, W axis optionally	Max. 8 axes plus spindle control 2 axes	NO	Max. 5 axes plus spindle control	II 1.2
Simultaneous controllable axes expansion	Max. simultaneous controlled axes	Same as left	Same as left	II 1.4
Single direction positioning	YES	YES	YES	II 3.2
3–dimensional circular interpolation function	Basic 3 axes puls rotation 2 axes	Basic 3 axes plus rotation 1 axis	Basic 3 axes plus rotation 2 axes	II 3.5
Helical interpolation	Also applied to additional axes, Circular interpolation plus max. 2 axes linear interpolation	Also applied to additional axes	Also applied to additional axes	II 3.6
Helical interpolation B (*)	Circular interpolation plus max. 4 axes linear interpolation	NO	Circular interpolation plus max. 3 axes linear interpolation	II 3.7
Hypothetical axis	YES	YES	YES	II 3.8
Polar coordinate interpolation	YES	YES	YES	II 3.9
Cylindrical interpolation	YES	YES	YES	II 3.10
Exponential function interpolation	YES	YES	YES	II 3.11
Circular threading B	YES	YES	YES	II 3.12
Involute interpolation	YES	YES	YES	II 3.13
Helical involute interruption	Involute interpolation plus linear 4 axes	Involute interpolation plus linear 4 axes	Involute interpolation plus linear 4 axes	II 3.14
Spline interpolation	YES	YES	YES	II 3.15
Spiral interpolation and conical interpolation	YES	YES	YES	II 3.16
Smooth interpolation (*)	YES	NO	NO	II 3.17
Thread cutting, inch threading, continuous threading	YES	YES	YES	II 4
Per revolution feed	YES	YES	YES	II 5.2.4
Inverse time feed	YES	YES	YES	II 5.2.5
F–1 digit feed	YES	NO	YES	II 5.2.6
Second feedrate override	YES	NO	NO	II 5.3.2

Table 2 (b) Optional specification (2/8)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Second feedrate override B	YES	NO	NO	II 5.3.3
Bell–shaped acceleration/ deceleration after cutting feed interpolation	YES	YES	YES	II 5.6
Cutting point speed control function	YES	YES	YES	II 5.10
Acceleration/deceleration function for the constant speed specified by the PMC axis control function	YES	YES	YES	II 5.11
Automatic corner override	YES	YES	YES	II 5.17
Feed per rotation without a position coder	YES	YES	YES	II 5.19
2nd to 4th reference position return	YES	2nd reference position return	YES	II 6.4
Floating reference point return	YES	YES (Note 3)	YES (Note 3)	II 6.5
Programmable data input	G10, tool offset amount, Work zero pint offset amount can be changed by programming	Same as left	Same as left	II 7.5 II 14.8
Additional workpiece coordinate systems	YES	YES	YES	II 7.6
Plane switching (*)	YES	YES	YES	II 7.8
Polar coordinate command	YES	YES	YES	II 8.2
Inch/metric conversion	YES	YES	YES	II 8.3
Spindle speed binary/ analog output/spindle speed clamp (G92)	YES (S command should be within 60000 rpm) (Note 1)	YES (S command should be within 30000 rpm)	YES (S command should be with in 30000 rpm)	II 9.2 II 9.3 II 9.5
Constant surface speed control	YES	NO	YES	II 9.4
Actual spindle speed output	YES	NO	NO	II 9.6
Spindle positioning	YES	NO	NO	II 9.7
Spindle speed –fluctuation detection	YES	NO	NO	II 9.8
Tool life management	YES	NO	YES	II 10.2
Tool life management 512 pairs	YES	NO	NO	II 10.2

Table 2 (b) Optional specification (3/8)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
2nd auxiliary function (Select address from A, B, C, U, V, W other than controlled axes address)	8 digit (Binary output)	BCD 3 digits	BCD 3 digits	II 11.2
1 block plural M–command	YES	NO	NO	II 11.4
48–character program name	YES	YES	YES	II 12.3
Optional block skip addition	YES	NO	YES	II 12.13
Canned cycles (G73, G74, G76, G80–G89)	YES	YES	YES	II 13.1
Rigid tapping	YES	YES	YES	II 13.2
Optional angle chamfering/ corner R	YES	YES	YES	II 13.4 II 13.5
Programmable mirror image	YES	YES	YES	II 13.7
Index table indexing	YES	NO	YES	II 13.8
Figure copying	YES	NO	YES	II 13.9
Tool offset G45–G48	YES	YES	YES	II 14.2
Cutter compensation B	YES	YES	YES	II 14.3.1
Cutter compensation C	YES	YES	YES	II 14.3.2
Three dimensional tool compensation	YES	YES	YES	II 14.4
Tool offset by tool number	YES	YES	YES	II 14.5
Tool offset amount memory B	Geometry/wear memory, radius and length offset indistinguishable	Same as left	Same as left	II 14.6.2
Tool offset amount memory C	± 6 digits, geometry/wear memory, length/radius offset memory	Same as left	Same as left	II 14.6.3
Additional tool offset pairs	Total 99/200/499/999 pairs	Same as left	Same as left	II 14.7
Rotary table dynamic fixture offset	YES	YES	YES	II 14.9
Three–dimensional cutter comepensation (*)	YES	YES	YES	II 14.10
Designation direction tool length compensation (*)	YES	NO	NO	II 14.11

Table 2 (b) Optional specification (4/8)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Stored pitch error compensation	YES	YES	YES	II 15.1
Interpolating pitch error compensation	YES	YES	YES	II 15.2
The second cylindrical pitch error compensation method	YES	YES	YES	II 15.3
Inclination compensation	YES	YES	YES	II 15.4
Straightness compensation	YES	YES	YES	II 15.5
Programmable parameter entry	YES	YES	YES	II 15.7
Interpolation–type straightness compensation	YES	NO	NO	II 15.8
128–point straightness compensation	YES	NO	NO	II 15.9
Bidirectional pitch error compensation	YES	NO	NO	II 15.10
Axis switching	YES	YES	YES	II 16.1
Scaling	YES	YES	YES	II 16.2
Coordinate system rotation	YES	YES	YES	II 16.3
Three–dimensional coordinate conversion	YES	NO	NO	II 16.4
Skip function	YES	YES	YES	II 17.1
Multi–step skip function	YES	NO	NO	II 17.2
High–speed skip signal input	YES	YES	YES	II 17.3
Skipping the commands for several axes	YES	YES	YES	II 17.4
Automatic tool length measurement	YES	NO	NO	II 17.5
High–speed measuring position reach signal input	YES	YES	YES	II 17.6
Tool length/work zero point measurement B	YES	YES	YES	II 17.8
Torque limit skip	YES	NO	NO	II 17.9
Incremental system IS–D(*)	0.00001 mm 0.000001 inch	Same as left	Same as left	II 1.7

Table 2 (b) Optional specification (5/8)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Tool offset value digit expansion	YES	YES	YES	II 14.6
Custom macro	YES	YES	YES	II 18
Custom macro common variable	Total 100/200/300/600 variable	Same as left	Same as left	II 18.2
Read/punch function for custom macro common variable	YES	YES	YES	II 18.3
Interruption type custom macro	YES	NO	YES	II 18.4
Key and program encryption	YES	YES	YES	II 18.5
High–speed machining	YES	YES	YES	II 19.1
Multi–buffer	YES	YES	YES	II 19.2
High–precision contour control (*)	YES	YES	YES	II 19.6
Simple synchronous control	YES	YES	YES	II 20.7
Feed stop	YES	YES	YES	II 20.8
Arbitrary command multiply	YES	YES	YES	II 20.9
Twin table control	YES	YES	YES	II 20.10
Simple synchronization control positional deviation check function	YES	YES	YES	II 20.11
Normal direction control	YES	YES	YES	II 20.12
Chopping function	YES	NO	NO	II 20.13
Axis control by PMC	YES	NO	NO	II 20.14
Upgraded 5–axis control compensation parameter	YES	NO	NO	II 20.15
Two axes electronic gear box (*)	YES	NO	NO	II 20.17
EGB axis skip	YES	NO	NO	II 20.18
Electric gear box automatic phase matching	YES	NO	NO	II 20.19
Simple synchronous control	YES	NO	NO	II 20.7
Sequence number comparison and stop	YES	YES	YES	II 21.5.3

Table 2 (b) Optional specification (6/8)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Program restart	YES	YES	YES	II 21.6.1
Program restart function and output of M, S, T and B codes	YES	YES	YES	II 21.6.2
Restart of block	YES	YES	YES	II 21.6.3
Tool retract & recover	YES	NO	NO	II 21.6.4
Manual handle interruption	YES	YES	YES	II 21.7.1
Automatic/manual simultaneous operation	YES	NO	NO	II 21.7.2
Retrace	YES	NO	NO	II 21.8
Active block cancel	YES	NO	NO	II 21.9
Manual handle feed (1st)	YES	YES	YES	II 22.3
Manual handle feed (2nd, 3rd)	YES	YES	YES	II 22.4
Manual arbitrary angle feed	Unit of angle: 1/16°	NO	NO	II 22.5
Manual numerical command	YES	YES	YES	II 22.6
Manual interruption function for three–dimensional coordinate system conversion (*)	YES	NO	NO	II 22.8
Language selection	YES	YES	YES	II 24.6
Menu switch	YES	YES	YES	II 24.10
Software operator’s panel	YES	YES	YES	II 24.11
Graphic display	YES	YES	YES	II 24.12
Simple conversational automatic programming function	YES (Note 4)	YES (Note 4)	YES (Note 4)	II 24.15
Directory display of floppy cassette	YES	YES	YES	II 24.17
M–code grouping	YES	NO	NO	II 24.27
Rewinding of portable tape reader	YES	YES	YES	II 27.3.2
Machining time stamp function	YES	YES	YES	II 24.18
Additional programs	400/1000	Same as left	Same as left	II 25.4

Table 2 (b) Optional specification (7/8)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Part program storage length	Max. 5120 m	Same as left	Same as left	II 25.5
Playback	YES	YES	YES	II 25.6
Override playback	YES	YES	YES	II 25.7
External I/O device control	YES	NO	NO	II 25.8
Trouble diagnosis guidance (*)	YES	YES	YES	II 26.2
Tape reader without reels	YES	YES	YES	II 27.1.1
Tape reader with reels	YES	YES	YES	II 27.1.2
Reader/puncher interface	YES	YES	YES	II 27.2
Portable tape reader	YES	YES	YES	II 27.3.2
FANUC PROGRAM FILE Mate	YES	YES	YES	II 27.3.3
FANUC Handy File	YES	YES	YES	II 27.3.4
Stored stroke check 2	YES	YES	YES	II 28.2.3
Stroke check before move	YES	YES	YES	II 28.2.4
External deceleration	Applied to all axes	NO	Applied only X, Y, Z axes	II 28.4
Abnormal load detection function	YES	NO	NO	II 28.5
Moving signal output	YES	YES	YES	II 29.11
Moving direction signal output	YES	NO	NO	II 29.12
External data input/output	Input/output of tool offset amount, work zero offset value, machine coordinate system shift amount, alarm message, operator message, program number search, sequence number search and custom macro variables are available.	Input of tool offset amount and work zero offset value are available (with PMC) NO (without PMC)	Input of tool offset amount, work zero offset value, alarm message, operator message and program number search are available	II 30
External workpiece number search	31 points	15 points	31 points	II 31
CNC window	YES	NO	NO	II 33.2
CNC window B	YES	YES	YES	II 33.3

Table 2 (b) Optional specification (8/8)

Items	Functions			Reference item
	Basic machine interface (BMI)	3M interface	6M interface	
Multi–tap transformer	200/220/230/240/380/415/ 440/460/480/550 VAC	Same as left	Same as left	
Key input from PMC	YES	YES	YES	II 33.4
FS6M interface multi–handle	—	—	YES	II 32.3
Position switching function	YES	YES	YES	II 40

NOTE

1 S8–digit (Binary code output) is available as a standard feature in case of Basic Machine Interface (BMI).
2 T8–digit (Binary code output) is available as a standard feature in case of Basic Machine Interface (BMI).
3 The floating reference point return completion signal (output signal) is not provided.
4 The applicable display unit is limited.
5 Above functions with asterisk (*) can not be used in the 15MEK or 15MEL.

II. NC FUNCTIONS

This Part describes all the functions which will be realized throughout all models and all machine interfaces. For which functions are available on a specific machine interface in a specific model, refer to the list of specifications in Part I.

1 CONTROLLED AXES

1.1 BASIC CONTROLLED AXES

3 axes (2 axes possible)

1.2 CONTROLLABLE AXES EXPANSION

Max. 7 axes (Total max. 10 axes Cs axis: 2 axes)

1.3 BASIC SIMULTANEOUSLY CONTROLLABLE AXES

2 axes

1.4 SIMULTANEOUSLY CONTROLLABLE AXES EXPANSION

Simultaneously controllable axes:

Following are controlled all axes at a time. Positioning, Linear interpolation, jog feed and incremental feed.

1.5 NAME OF AXES

Name of axes can be optionally selected from A, B, C, U, V, W, X, Y, Z (Set by parameter).

1.6 PROGRAMMING AXIS NAME ADDITION

Nine alphabets A, B, C, U, V, W, X, Y, and Z can conventionally be used for program axis name. However, 9 or more axis names are required when 9 or more axes are to travel in the multi–axis machine with multiple heads. This function, adds 4 addresses I, J, K, and E further in addition to 9 axis names.

Axis name: A, B, C, E, I, J, K, U, V, W, X Y, and Z (Total 13)

The maximum number of digits is 8 and the decimal point programming is allowed.

However, if the I, J, K, and E are used as axis names, they cannot be used for uses other than axis names.

The conventional uses, and limitation of uses with this function are compared in the following:

Additional address	G–CODE etc.	Conventional uses	User for controlled axes	Comments
I, J, K	G02 G03	Center position of arc	Position vector of I, J, and K axes	The command R is used for the center.
	G41 G42	Three–dimensional offset vector	Same as the above	The three–dimensional tool compensation is not allowed
	G76 G87	Shift value in canned cycle	Same as the above	The shift value cannot be commanded.
	G22	One point of stroke limit	Same as left	The limit position cannot be commanded.
	G65 G66 G66.1	Argument	Argument	The decimal point position can be determined by increment system.
E	G33	Screw lead (number of threads in inch threading)	E axis position vector	The number of threads in inch threading cannot be specified with G33.
	#4108	Custom macro variable Model information of address ‘E’	No special meaning	The custom macro variable “#4108” cannot be used.

CAUTION
When this function is used, the second auxiliary function cannot be used.

1.7 INCREMENT SYSTEM

Least input increment		Least command increment		Maximum stroke		Code
0.01	mm	0.01	mm	999999.99	mm	IS–A
0.001	mm	0.001	mm	99999.999	mm	IS–B
0.0001	mm	0.0001	mm	9999.9999	mm	IS–C
0.00001	mm	0.00001	mm	9999.99999	mm	IS–D
0.000001	mm	0.000001	mm	999.999999	mm	IS–E
0.001	inch	0.001	inch	99999.999	inch	IS–A
0.0001	inch	0.0001	inch	9999.9999	inch	IS–B
0.00001	inch	0.00001	inch	999.99999	inch	IS–C
0.000001	inch	0.000001	inch	999.999999	inch	IS–D
0.0000001	inch	0.0000001	inch	99.9999999	inch	IS–E
0.001	deg	0.001	deg	99999.999	deg	IS–A
0.0001	deg	0.0001	deg	9999.9999	deg	IS–B
0.00001	deg	0.00001	deg	999.99999	deg	IS–C
0.000001	deg	0.000001	deg	999.999999	deg	IS–D
0.0000001	deg	0.0000001	deg	99.9999999	deg	IS–E

Five types of increment systems are provided. Increment system IS–A, IS–B, and IS–C can be specified for each axis by setting ISFx and ISRx of parameter No. 1004. Increment system IS–D can be specified for each axis by setting ISDx of parameter No. 1004. Increment system IS–E can be specified for each axis by setting ISEx of parameter No. 1009. Metric systems and inch systems, however, cannot be specified for a machine at the same time. Functions, such as circular interpolation and cutter compensation, cannot be used for axes using different increment systems. Increment systems IS–D and IS–E are optional.

For IS–B and IS–C, parameter IPPx (data No. 1004, input unit: multiplied by 10) sets the increment systems as follows. For the settings of the increment systems, refer to the manual provided by the machine tool builder.

Least input increment		Least command increment		Maximum stroke		Code
0.01	mm	0.001	mm	99999.999	mm	IS–B
0.001	inch	0.0001	inch	99999.9999	inch	
0.01	deg	0.001	deg	99999.999	deg	
0.001	mm	0.0001	mm	9999.9999	mm	IS–C
0.0001	inch	0.00001	inch	999.99999	inch	
0.001	deg	0.0001	deg	9999.9999	deg	

1.8 MAXIMUM STROKE

Maximum stroke = minimum command increment $\times$ 99999999 (999999999 for IS–D and IS–E)
See section 1.7.

2 PREPARATORY FUNCTIONS

The following G code are offered.

G code	Group	Function
G00	01	Positioning
G01		Linear interpolation
G02		Circular/Helical/spiral/conical interpolation CW
G03		Circular/Helical/spiral/conical interpolation CCW
G02.1		Circular threading B (CW)
G03.1		Circular threading B (CCW)
G02.2		Involute interpolation (CW)
G03.2		Involute interpolation (CCW)
G02.3		Exponential function interpolation (CW)
G03.3		Exponential function interpolation (CCW)
G02.4		3–dimensional circular interpolation
G03.4		3–dimensional circular interpolation
G04	00	Dwell
G05.1		Multi–buffer
G06.1	01	Spline interpolation
G07	00	Hypothetical axis interpolation
G07.1		Cylindrical interpolation
G09		Exact stop
G10		Data setting
G10.1		PMC data setting
G10.3		High–speed machining registration start
G10.6		Tool retract & recover
G11		Data setting mode cancel
G11.3		High–speed machining registration end
G12.1	26	Polar coordinate interpolation
G13.1		Polar coordinate interpolation cancel

G code	Group	Function	
G12.2	00	Full circle cutting (clockwise)	
G13.2		Full circle cutting (counterclockwise)	
G15	17	Polar coordinate command cancel	
G16		Polar coordinate command	
G17	02	Xp Yp plane	Xp: X axis or its parallel axis
G18		Zp Xp plane	Zp: Y axis or its parallel axis
G19		Yp Zp plane	Xp: Z axis or its parallel axis
G20	06	Inch input	
G21		Metric input	
G22	04	Stored stroke check on	
G23		Stored stroke check off	
G25	24	Spindle speed fluctuation detection off	
G26		Spindle speed fluctuation detection on	
G27	00	Reference position return check	
G28		Reference position return	
G29		Return from reference position	
G30		Return to 2nd, 3rd, 4th reference position	
G30.1		Floating reference position return	
G31		Skip function	
G31.1		Multi–step skip function 1	
G31.2		Multi–step skip function 2	
G31.3		Multi–step skip function 3	
G33	01	Tread cutting	
G37	00	Tool length automatic measurement	
G38		Cutter compensation C vector retention	
G39		Cutter compensation C corner rounding	
G40	07	Cutter compensation cancel/3 dimensional tool compensation cancel	
G40.1	19	Normal direction control cancel	
G41	07	Cutter compensation left/3 dimensional tool compensation	
G41.1	19	Normal direction control left on	

G code	Group	Function
G41.2	07	3–dimensional cutter compensation left
G41.3		Leading edge offset
G42		Cutter compensation right
G42.1	19	Normal direction control right on
G42.2	07	3–dimensional cutter compensation right
G43	08	Tool length compensation +
G43.1		Tool length compensation in tool axis direction
G44		Tool length compensation –
G45	00	Tool offset increase
G46		Tool offset decrease
G47		Tool offset double increase
G48		Tool offset double decrease
G49	08	Tool length compensation cancel
G50	11	Scaling cancel
G50.1	18	Programmable mirror image cancel
G51	11	Scaling
G51.1	18	Programmable mirror image
G52	00	Local coordinate system setting
G53		Machine coordinate system selection
G54	12	Workpiece coordinate system 1 selection
G54.1		Additional workpiece coordinate system selection
G54.2		Fixture offset 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 mode
G63		Tapping mode
G64		Cutting mode

G code	Group	Function
G65	00	Macro call
G65.3		High speed machining program 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
G69		Coordinate system rotation cancel
G72.1	00	Rotation copy
G72.2		Parallel copy
G73	09	Peck drilling cycle
G74		Counter tapping cycle
G76		Fine boring cycle
G80		Canned cycle cancel/external operation function cancel
G80		Electronic gear box synchronous cancel (Command for hobbing machine or 1 axis)
G81		Drilling cycle, spot boring /external operation
G81		Electronic gear box synchronous start (Command for hobbing machine or 1 axis)
G80.5		Electronic gear box synchronous cancel (Command for 2 axes)
G81.5		Electronic gear box synchronous start (Command for 2 axes)
G81.1	00	Chopping
G82	09	Drilling cycle, counter boring
G83		Peck drilling cycle
G84		Tapping cycle
G84.2		Rigid tapping cycle
G84.3		Counter rigid tapping cycle
G85		Boring cycle
G86		Boring cycle
G87		Back boring cycle
G88		Boring cycle
G89		Boring cycle

G code	Group	Function
G90	03	Absolute command
G91		Increment command
G92	00	Workpiece coordinates change/Maximum spindle speed setting
G92.1		Workpiece coordinate system presetting
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
G98	10	Canned cycle initial level return
G99		Canned cycle R point level return

CAUTION
G codes of group 00 are not modal.

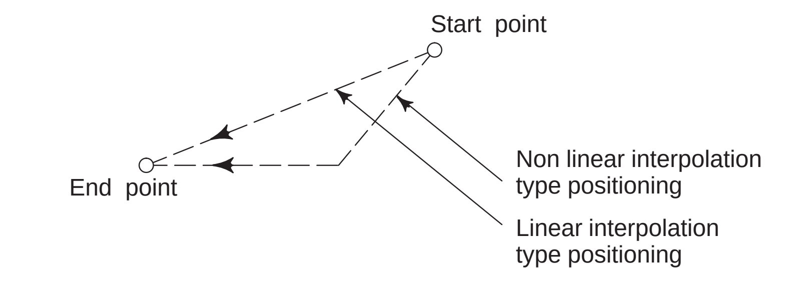
NOTE
A number of G codes can be specified in a single block if they are of different group each other.

3 INTERPOLATION FUNCTIONS

3.1 POSITIONING (G00)

The tool path can be selected by setting either of the following parameters.

- Linear interpolation type positioning
  Tool path is the same as linear interpolation (G01). Positioning is done in a speed which allows the minimum positioning time without exceeding rapid traverse rate of each axis.
- Non linear interpolation type positioning
  Positioning is done with each axis separately. Tool path generally does not became a line.



It is decelerated, to a stop at the end point, and imposition check is performed (checks whether the machine has come to the specified position).
Width of inposition can be set as a parameter.

Format

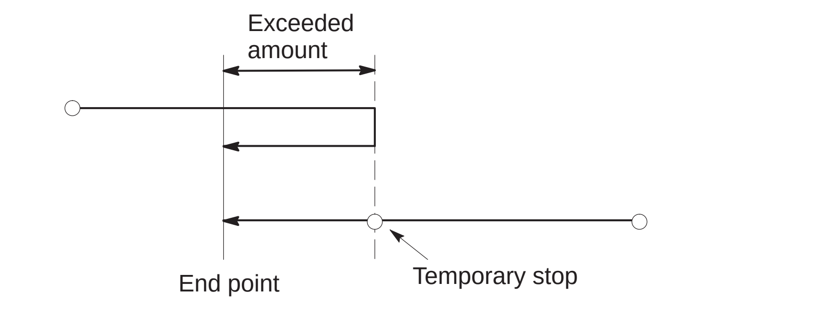
G00 IP_ _ ;
where
IP_ _ : Combination of optional axis address (of X, Y, Z, U, V, W, A, B, C) as X–Y–Z–A– . . .
This manual uses this notation hereinafter.
; : End of block (LF for ISO code / CR for EIA code)
This manual uses this notation hereinafter.

3.2 SINGLE DIRECTION POSITIONING (G60)

It is always controlled to perform positioning to the end point from a single direction, for better precision in positioning. If direction from start point to end point is different from the predecided direction, it once positions to a point past the end point, and the positioning is reperformed for that point to the end point.
Even if the direction from start point to end point is the same as predecided direction, the tool stops once before the end point.
Positioning in this case is always non–linear interpolation type positioning (this has no relations to the G00 parameter setting).

Format

G60 _ ;

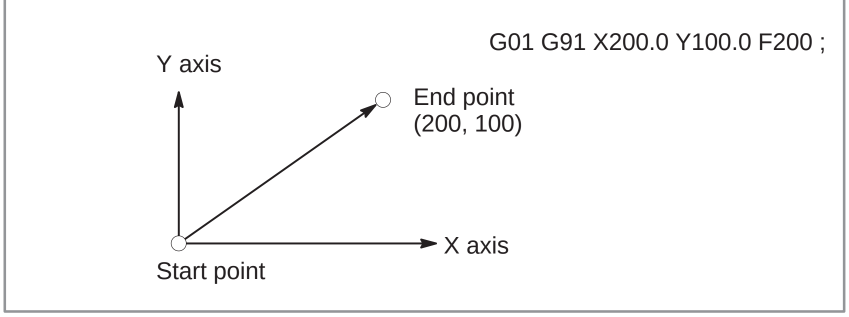


3.3 LINEAR INTERPOLATION

Linear interpolation is done with tangential direction feedrate specified by the F code.

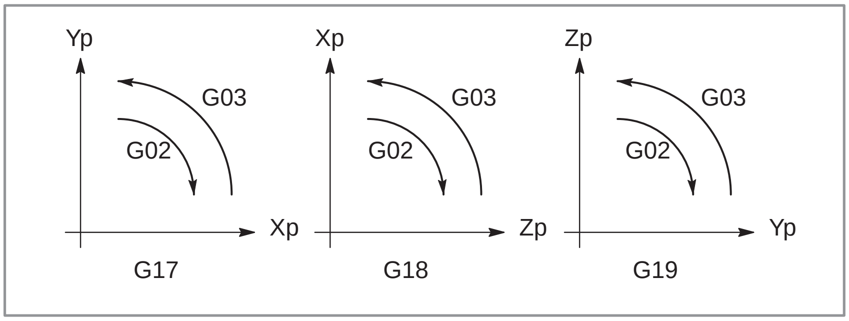
Format

G01 IP_ _ F_ _ ;
where
F : Feedrate



3.4 CIRCULAR INTERPOLATION (G02, G03)

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



Clockwise and counterclockwise

Feed rate 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
G18: Zp-Xp plane
G19: Yp-Zp plane
where
Xp: X axis or its parallel axis
Yp: Y axis or its parallel axis
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.

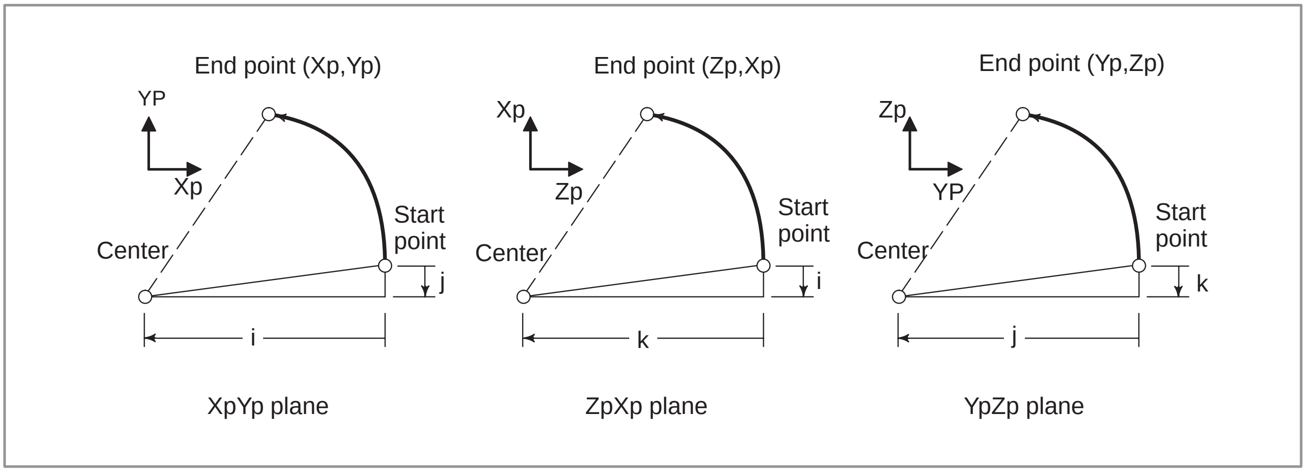
Format

G17 $\left\{\begin{matrix}\textbf{G02}\\ \textbf{G03}\end{matrix}\right\}$ **Xp_ _ Yp_ _ I_ _ J_ _ F_ _; Xp–Yp plane**

G18 $\left\{\begin{matrix}\textbf{G02}\\ \textbf{G03}\end{matrix}\right\}$ **Zp_ _ Xp_ _ K_ _ I_ _ F_ _; Zp–Xp plane**

G19 $\left\{\begin{matrix}\textbf{G02}\\ \textbf{G03}\end{matrix}\right\}$ **Yp_ _ Zp_ _ J_ _ K_ _ F_ _; Yp–Zp plane**

where
I_ _, J_ _, K_ _ Distance of tthe X, Y, Z, axes from the start point to the center of the circle



Circular interpolation command

3.5 3–DIMENSIONAL CIRCULAR INTERPOLATION FUNCTION

Spatial circular interpolation can be performed by specifying an intermediate point and end point of an arc.

```
G02.4   Xx1  Yy1  Zz1  Aa1  Bb1 ;
        Xx2  Yy2  Zz2  Aa2  Bb2 ;
or
G03.4   Xx1  Yy1  Zz1  Aa1  Bb1 ;
        Xx2  Yy2  Zz2  Aa2  Bb2 ;
```

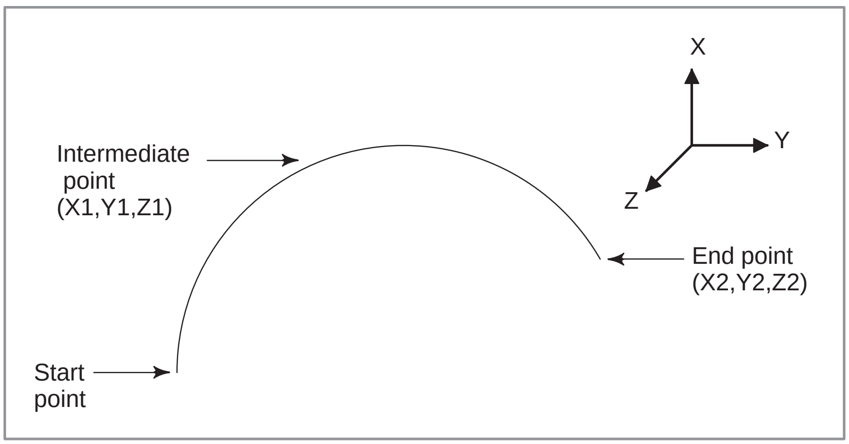
3-dimensional circular interpolation is performed by specifying one of the above commands.

In the above commands, the first block designates the intermediate point of an arc and the second block designates the end point.

In incremental specification, the intermediate point specified in the first block must be specified as a position relative to the start point. The end point specified in the second block must be specified as a position relative to the intermediate point.

Since this function does not distinguish between the directions of rotation, either G02.4 or G03.4 can be specified.

G02.4 and G03.4 fall within G code group 01. These commands are continuous–state commands. Therefore, Once G02.4 or G03.4 is specified, it is valid until another group 01 G code is specified.



As shown in the figure, an arc ending at a certain point cannot be obtained unless both an intermediate and end point are specified. Specify the intermediate point and end point in separate blocks.

In MDI operation, 3–dimensional circular interpolation starts when the start button is pressed after the blocks for the intermediate and end points are entered. If the start button is pressed immediately after the intermediate point block is entered, the end point of the arc is still unknown so only buffering is performed. In this case, to start 3–dimensional circular interpolation, enter the end point block, then press the start button again.

When the commands for 3–dimensional circular interpolation are specified successively, the end point is used as the start point for the next interpolation operation.

3.6 HELICAL INTERPOLATION (G02, G03)

Helical interpolation performs circular interpolation of a maximum of two axes, synchronizing with other optional two axes circular interpolation. Thread cutting of large radius threads or machining of solid cams are possible by moving a tool in a spiral.
The commanded speed is the speed of the tangential direction of the arc.

Format

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

Xp–Yp plane

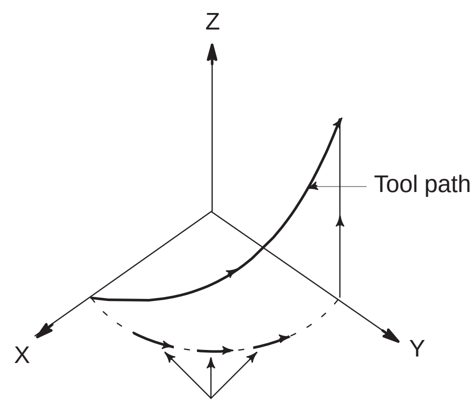
$$\mathbf{G18}\begin{Bmatrix}\mathbf{G02}\\\mathbf{G03}\end{Bmatrix}\ \mathbf{Zp\_\_\ \ Xp\_\_\ \ K\_\_\ \ I\_\_\ \ \alpha\_\_\ \ (\beta\_\_)\ \ F\_\_\ ;}$$

Zp–Xp plane

$$\mathbf{G19}\begin{Bmatrix}\mathbf{G02}\\\mathbf{G03}\end{Bmatrix}\ \mathbf{Yp\_\_\ \ Zp\_\_\ \ I\_\_\ \ J\_\_\ \ \alpha\_\_\ \ (\beta\_\_)\ \ F\_\_\ ;}$$

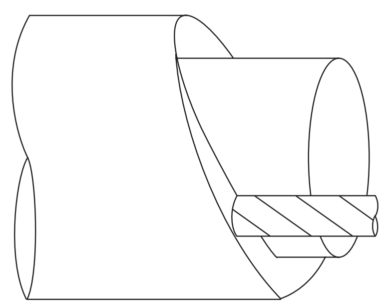
Yp–Zp plane

where
α, β : Optional axis other than the circular interpolation axes



Tangential speed along an arc by circular interpolation is the rate specified in programming.

Solid cam



3.7 HELICAL INTERPOLATION B (G02, G03)

Helical interpolation B performs circular interpolation of a maximum of four axes, synchronizing with other optional two axes linear interpolation.

The commanded speed is the speed of the tangential direction of the arc.

Format

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

Xp–Yp plane

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

Zp–Xp plane

$$\mathbf{G19}\left\{\begin{matrix}\mathbf{G02}\\ \mathbf{G03}\end{matrix}\right\}\ \mathbf{Yp\_\_\ \ Zp\_\_\ \ I\_\_\ \ J\_\_}\ \ \alpha\_\_\ \ (\beta\_\_\ \ \gamma\_\_\ \ \delta\_\_)\ ;$$

Yp–Zp plane

where
$\alpha, \beta, \gamma, \delta$: Optional axis other than the circular interpolation axes

3.8 HYPOTHETICAL AXIS INTERPOLATION (G07)

Format

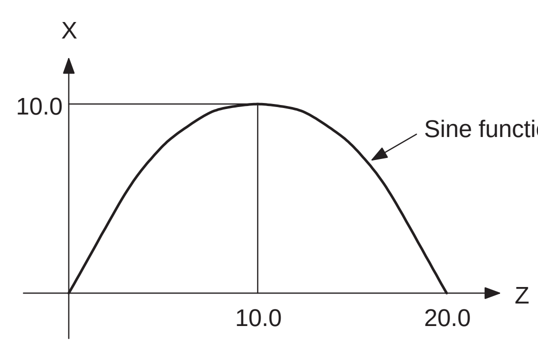
G07α0 ; Hypothetical axis setting
G07α1 ; Hypothetical axis cancel

where,
α : One of controlled axis address

Hypothetical axis interpolation can be used for the following applications:

(1) Sine function interpolation
Pulse distribution with one axis for the circular arc of helical interpolation as the hypothetical axis (Pulses are distributed but not output to the motor) allows the rest of the two axes to move as sine function interpolation. Which of the three axes is regarded as the hypothetical axis is commanded by G07.

Example :
G07 Y0 ; . . Determines the Y–axis as the hypothetical axis.
G91 G02 G17 X0 Y–20. R10.0 Z20.0 F50 ;
. . . . Sine interpolation is performed on the X– and Z–axes.
G07 Y1 ; . . Cancels the Y–axis as the hypothetical axis.

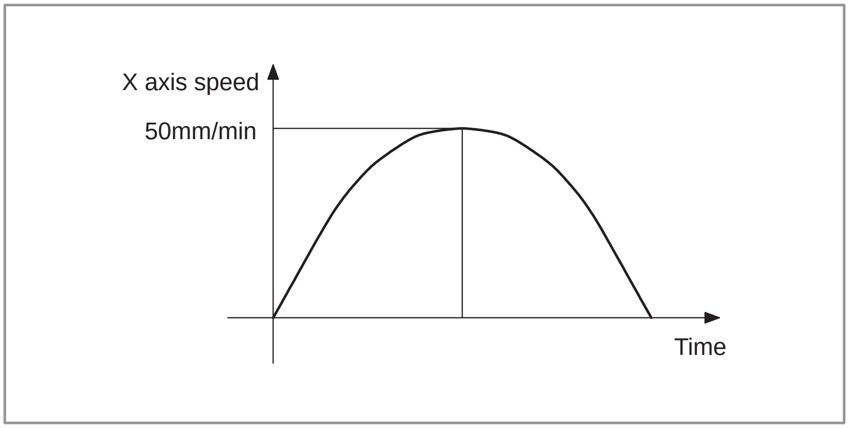


(2) Sine function change of moving speed

Pulse distribution with one axis for circular arc interpolation as the hypothetical axis allows the moving speed of the rest of one axis to change as a sine function.

Example :

G07 Y0 ; . . Determines the Y–axis as the hypothetical axis.
G91 G02 G17 X30.0 Y0 R15.0 F50 ;
. . . . Changes the feedrate on the X–axis as a sine function.
G07 Y1 ; . . Cancels the Y–axis as the hypothetical axis.



(3) Fraction lead threading

The long axis (the axis with the largest move distance) for threading is determined as the hypothetical axis to enable threading of the fraction lead.

Example :

G07 X0 ; . . Determines the X–axis as the hypothetical axis.
G91 G33 X1181.102 Z100.0 F100 ;
. . . . The Z–axis lead is expressed by the following formula.
G07 X1 ; . . Cancels the X–axis as the hypothetical axis.

$$\text{Z–axis lead} = 100 \times \frac{100.0}{1181.102} \doteq 8.4666$$

3.9 POLAR COORDINATE INTERPOLATION (G12.1, G13.1)

The function in which contour control is done in converting the 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) is the polar coordinate interpolation. It is an effective function when a straight line groove is cut on the outer diameter of a workpiece or when a cam shaft is ground.

Whether the polar coordinate interpolation is done or not is commanded by a G code.

G12.1; Polar coordinate interpolation mode
(Polar coordinate interpolation shall be done.)

G13.1; Polar coordinate interpolation cancel mode
(Polar coordinate interpolation is not done.)

These G codes shall be commanded in a single block.

1) Polar coordinate interpolation mode (G12.1)
   The axes (linear axis and rotary axis) on which polar coordinate interpolation is done are set beforehand by parameters.
   Change the mode to polar coordinate interpolation mode by commanding G12.1, and a plane (hereinafter referred to as polar coordinate interpolation plane) is selected in which linear axis is made to the first axis of the plane, and virtual axis being a right angle with the linear axis is made to the second axis of the plane. Polar coordinate interpolation is carried out on this plane.
   In the polar coordinate interpolation made, the command of linear interpolation (G01) and circular interpolation (G02, G03) is possible. And both absolute command (G90) and incremental command (G91) are possible.
   For the program command it is possible to apply cutter compensation. For the path after cutter compensation is done, polar coordinate interpolation can be made.
   As for feedrate, specify the tangential speed (relative speed between the workpiece and the tool) on the polar coordinate interpolation plane (cartesian coordinate system) with F.
2) Polar coordinate interpolation cancel mode (G13.1)
   The polar coordinate interpolation cancel mode is obtained by G13.1 command.

3) Example of a program
Polar coordinate interpolation by X axis (Linear axis) and C axis (Rotary axis)

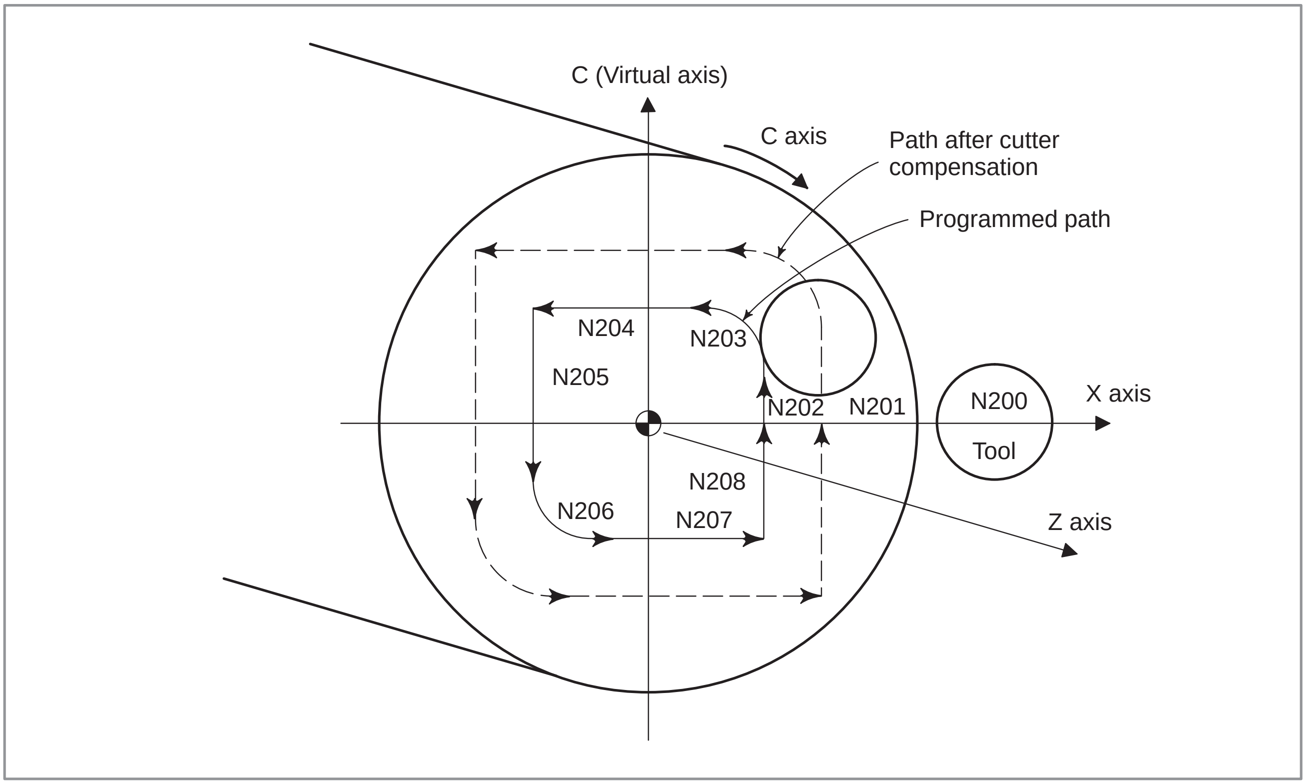


Fig. 3.9

(X axis is diameter programming and C axis is radius programming)

```
O0001;
   :
N100 G90 G00 X120.0 C0 Z_ ;          Positioning to the starting position
N200 G12.1;                          Starting polar coordinate interpolation
N201 G42 G01 X40.0 F_ D01;
N202 C10.0;
N203 G03 X20.0 C20.0 R10.0 ;
N204 G01 X-40.0 ;                    Contour program
N205     C-10.0 ;                    (Program in cartesian coordinate
N206 G03 X-20.0 C-20.0 I10.0 K0 ;    system of X-C plane)
N207 G01 X40.0 ;
N208     C0 ;
N209 G40 X120.0 ;
N210 G13.1 ;                         Canceling polar coordinate interpolation
N300 Z_ ;
N400 X_ C_ ;
   :
M30 ;
```

3.10 CYLINDRICAL INTERPOLATION (G07.1)

When the form on the expanded side view of a cylinder (from on the cylinder coordinate system) is commanded by a program command, the NC converts the form into a linear axis movement and a rotary axis movement then performs a contour control. This feature is called the cylindrical interpolation.
Cylindrical interpolation is commanded with G07.1.

G07.1 (Name of rotary axis) Radius value of cylinder ; : Cylindrical interpolation mode

G07.1 (Name of rotary axis) 0 ; : Cancellation mode of cylindrical interpolation

1) Cylindrical interpolation mode
   Cylindrical interpolation is made between the rotary axis specified in the block of G07.1 and the other optional linear axis.
   Circle interpolation command is allowed as well as linear interpolation, during cylindrical interpolation mode. Also, absolute command and incremental command can be made. Cutter compensation can be added to the program command. Cylindrical interpolation is made for the path after cutter compensation.
   Feed rate gives the tangential speed on the expanded plane of the cylinder with F.
2) Cancellation mode of cylindrical interpolation
   G07.1 (Name of rotary axis) 0;

   Cancellation mode of cylindrical interpolation is made when commanded as above.

3) An example of a program

```
O0001 (CYLINDRICAL INTERPOLATION);
  N1 G00 G90 Z100.0 C0;
  N2 G01 G18 Z0 C0;
  N3 G7.1 C57299;
  N4 G01 G42 Z120.0 D01 F250;
  N5 G40.0;
  N6 G02 Z90.0 C60.0 R30.0 ;
  N7 G01 Z70.0;
  N8 G03 Z60.0 C70.0 R10.0;
  N9 G01 C150.0;
  N10 G03 Z70.0 C190.0 R75.0;
  N11 G01 Z110.0 C230.0;
  N12 G02 Z120.0 C270.0 R75.0;
  N13 G01 G360.0;
  N14 G40 Z100.0;
  N15 G07.1 C0;
  N16 M30;
```

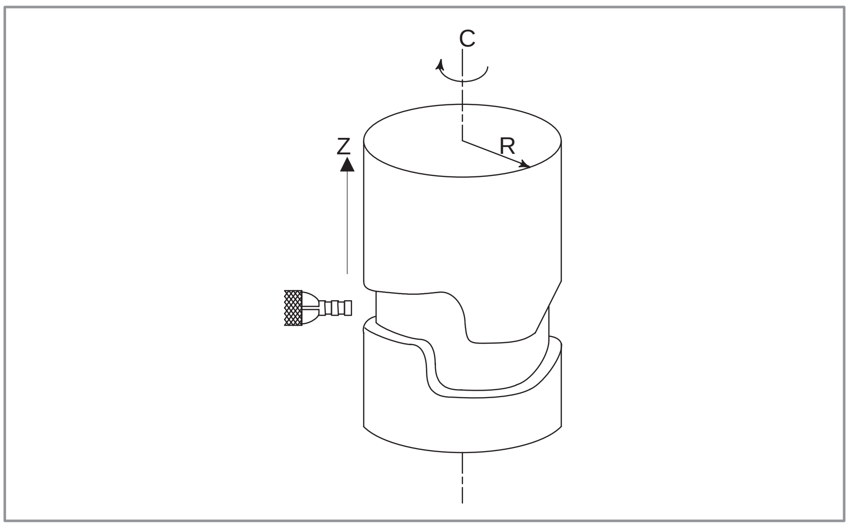


Fig. 3.10 (a)

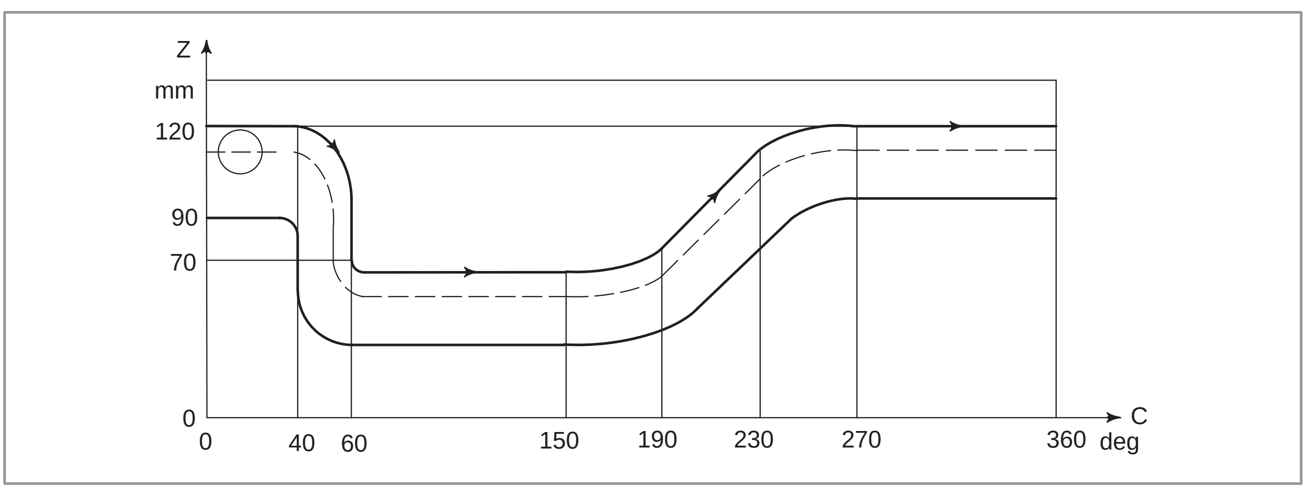


Fig. 3.10 (b)

3.11 EXPONENTIAL FUNCTION INTERPOLATION (G02.3, G03.3)

In synchronization with the travel of the rotary axis, the linear axis (X axis) performs the exponential function interpolation. With the other axes, the linear interpolation the X axis is performed.
This function is effective for the tapered constant helix machining in the tool grinding machine.

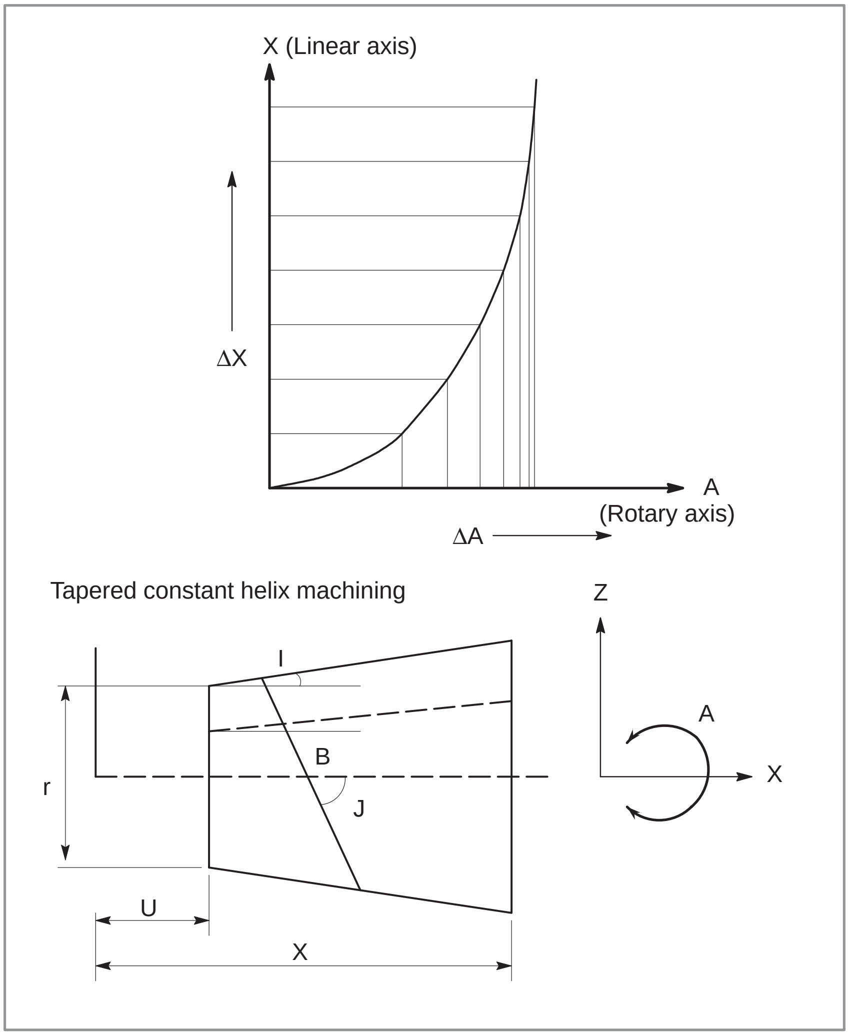


Fig. 3.11

The exponential function relation expression between the linear axis and the rotary axis is defined as in the following :

$$X(\theta) = R * (e^{\theta/R} - 1) * \frac{1}{\tan(I)} \quad \text{..... Travel of linear axis} \quad (1)$$

$$A(\theta) = (-1)^{\omega} * 360 * \frac{\theta}{2\pi} \quad \text{..... Travel of rotation axis} \quad (2)$$

where

$$K = \frac{\tan(J)}{\tan(I)}$$

$\omega = 0$ or 1 ..... Rotational direction

R, I, J are constant and θ is the angle (radian) of rotation.
Also from the equation (1),

$$\theta(X) = K * \ell n \left\{ \frac{X * \tan(I)}{R} + 1 \right\}$$

Equations (1) and (2) shall be specified by the following formats :

(Positive rotation) ω=0

G02.3 X_Y_ Z_ I_ J_ K_ R_ F_ Q_ ;

(Negative rotation) ω=1

G03.3 X_Y_ Z_ I_ J_ K_ R_ F_ Q_ ;

X_ ; Command terminal point by Absolute or incremental

Y_ ; Command terminal point by Absolute or incremental

Z_ ; Command terminal point by Absolute or incremental

I_ ; Command of angle I (The command unit is based on the reference axis.
The range of command is 1 to ±89°)

J_ ; Command of angle J (The command unit is based on the reference axis.
The range of command is 1 to ±89°)

K_ ; Amount of division of the linear axis in the exponential function interpolation (amount of span). (The command unit is based on the reference axis. The command range is a positive value.)

R_ ; Command of constant value R in the exponential function interpolation. (The command unit is based on the reference axis.)

F_ ; Command of initial feed rate.
The command is the same as the normal F code. The feed rate shall be given by the synthesized speed including the rotary axis.

Q_ ; Command of feed rate at terminal point.
The command unit is based on the reference axis. Within the NC, the tool is interpolated between the initial feedrate (F_) and final feedrate (Q_) depending on the amount of X axis travel.

3.12 CIRCULAR THREADING B (G02.1, G03.1)

Circular interpolation is made between two axes and simultaneously linear interpolation is made beween the optional two axes and the long axis of circle interpolation in the circular threading B. This circular thread cutting is not the one that the tool is moved in synchronization with rotation of the spindle (work) of the spindle motor, but the one that the sarvo motor controls the rotation of the workpiece. Therefore, it is effective for thread cutting in the same pitch on the barrel type surface, grooving, tool grinding, and etc. The speed along the long axis of the circle shall be specified as the feed rate.

Format

$$\mathbf{G17}\left\{\begin{array}{l}\mathbf{G02.1}\\ \mathbf{G03.1}\end{array}\right\}\ \mathbf{Xp\_\_}\ \ \mathbf{Yp\_\_}\ \ \alpha\_\_\ \ \beta\_\_\ \left\{\begin{array}{l}\mathbf{I\_\_}\ \ \mathbf{J\_\_}\\ \mathbf{R\_\_}\end{array}\right\}\ \mathbf{F\_\_}\ ;$$

Xp–Yp plane

$$\mathbf{G18}\left\{\begin{array}{l}\mathbf{G02.1}\\ \mathbf{G03.1}\end{array}\right\}\ \mathbf{Zp\_\_}\ \ \mathbf{Xp\_\_}\ \ \alpha\_\_\ \ \beta\_\_\ \left\{\begin{array}{l}\mathbf{K\_\_}\ \ \mathbf{I\_\_}\\ \mathbf{R\_\_}\end{array}\right\}\ \mathbf{F\_\_}\ ;$$

Zp–Xp plane

$$\mathbf{G19}\left\{\begin{array}{l}\mathbf{G02.1}\\ \mathbf{G03.1}\end{array}\right\}\ \mathbf{Yp\_\_}\ \ \mathbf{Zp\_\_}\ \ \alpha\_\_\ \ \beta\_\_\ \left\{\begin{array}{l}\mathbf{J\_\_}\ \ \mathbf{K\_\_}\\ \mathbf{R\_\_}\end{array}\right\}\ \mathbf{F\_\_}\ ;$$

Yp–Zp plane

where
α, β : Optional 2 axes other than circular interpolation

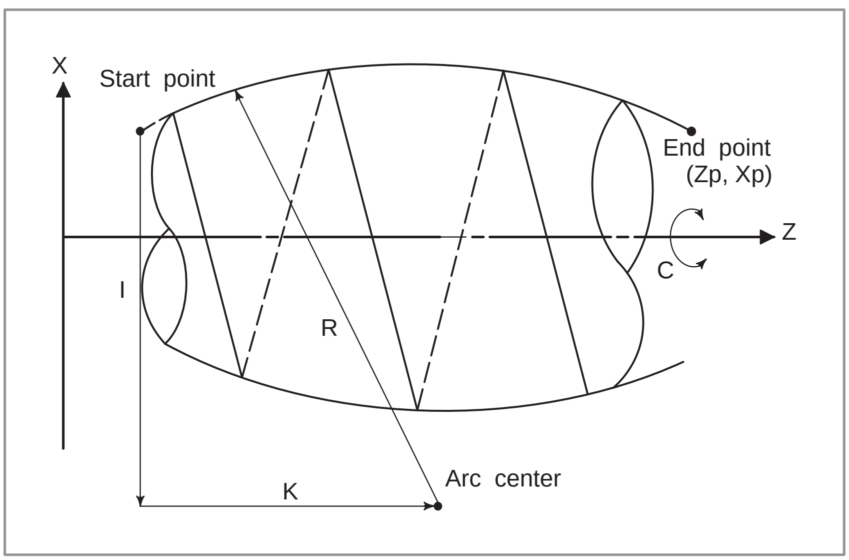


Fig. 3.12

3.13 INVOLUTE INTERPOLATION

With the following command, the involute curve machining can be performed. Approximate involute curve with a minute straight line or arc is not needed. Therefore, the programming becomes simple and reduces the tape length. The distribution of the pulse will not be interrupted during the continuous minute block high speed operation, so fast, smooth involute curve machining is possible.

Format

G17 $\begin{Bmatrix} \textbf{G02.2} \\ \textbf{G03.2} \end{Bmatrix}$ **Xp_ _ Yp_ _ I_ _ J_ _ R_ _ F_ _;**

Xp–Yp plane

G18 $\begin{Bmatrix} \textbf{G02.2} \\ \textbf{G03.2} \end{Bmatrix}$ **Zp_ _ Xp_ _ K_ _ I_ _ R_ _ F_ _;**

Zp–Xp plane

G19 $\begin{Bmatrix} \textbf{G02.2} \\ \textbf{G03.2} \end{Bmatrix}$ **Yp_ _ Zp_ _ J_ _ K_ _ R_ _ F_ _;**

Yp–Zp plane

G02.2	:	Clockwise involute interpolation
G03.2	:	Counterclockwise involute interpolation
Xp, Yp, Zp	:	End point coordinate value
I, J, K	:	Distance to the center of the basic circle of the involute curve from start point
R	:	Radius of basic circle
F	:	Cutting feedrate

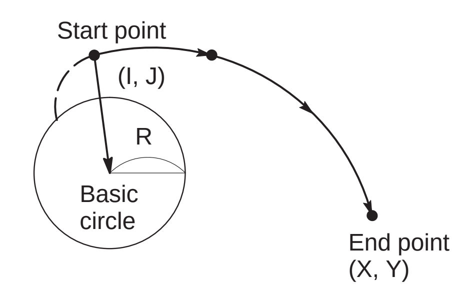


Fig. 3.13 (a) Clockwise involute interpolation

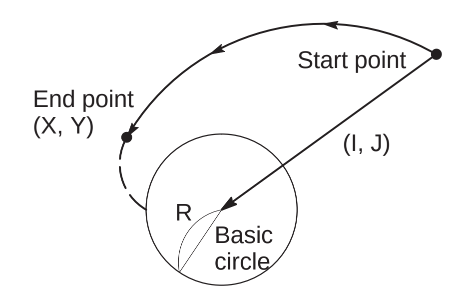


Fig. 3.13 (b) Counterclockwise involute interpolation

The cutter compensation can be applied to the commanded involute curve. The intersecting point vector of a straight line or circular arc and an involute curve is obtained and the offset involute curve is interpolated.

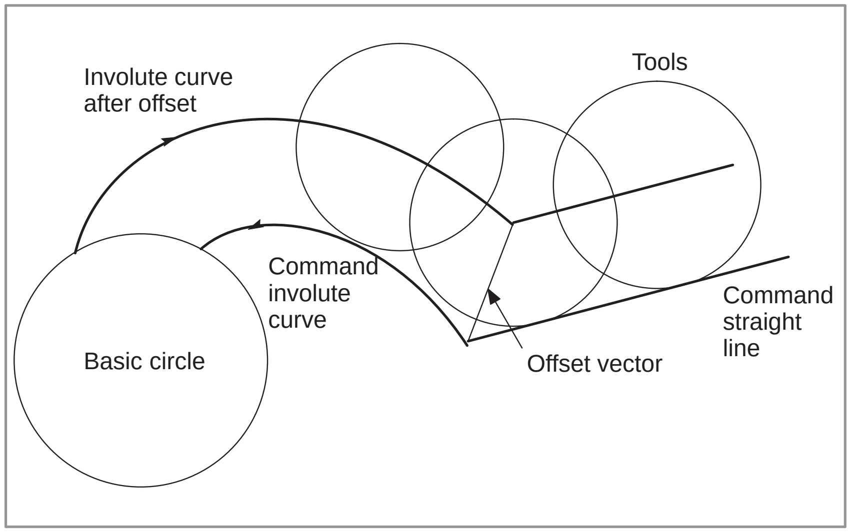


Fig. 3.13 (c) Cutter compensation and involute interpolation

3.14 HELICAL INVOLUTE INTERPOLATION

Helical involute interpolation is a similar to helical interpolation used for circular interpolation. Helical involute interpolation allows the manipulation of the tool along two axes for involute interpolation and along a maximum of four other axes concurrently.

3.15 SPLINE INTERPOLATION

Spline interpolation is prepared for machining of spline curve passing a specific dot–string. A smooth curve passing a dot–strings can be machined with this function.

The spline curve obtained by spline interpolation has the following characteristics.

(i) The spline curve passes through all command points.

(ii) The curves of the connecting line vector before and after, coincides at all command points except for start point and end point.

(iii) The curvature before and after coincided with the command point except for start point and end point.

Format

G06.1 X_ Y_ Z_ I_ K_ P_ Q_ R_ F_ ;

X : Mantissa of the X–axis component of primary differential vector at start point.

Y : Mantissa of the Y–axis component of primary differential vector at start point.

Z : Mantissa of the Z–axis component of primary differential vector at start point.

I : Mantissa of the X–axis component of secondary differential vector at start point.

J : Mantissa of the Y–axis component of secondary differential vector at start point.

K : Mantissa of the Z–axis component of secondary differential vector at start point.

P : Exponent of X–axis component of primary and secondary differential vectors.

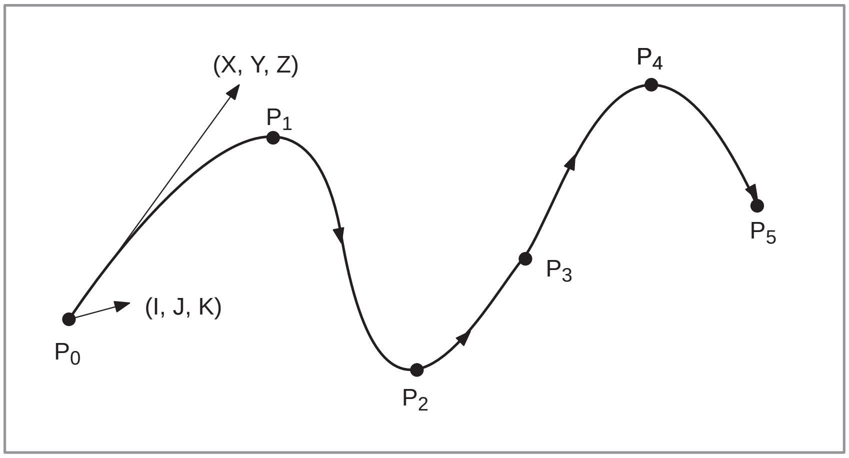
Q : Exponent of Y–axis component of primary and secondary differential vectors.

R : Exponent of Z–axis component of primary and secondary differential vector.

F : Feedrate

(Example)

G01 X – – Y – – Z – – F – – ; (P_0)
G06.1 X – – Y – – Z – – I – – J – – K – – P – – Q – – R – – F – – ;
X – – Y – – Z – – ; (P_1)
X – – Y – – Z – – ; (P_2)
X – – Y – – Z – – ; (P_3)
X – – Y – – Z – – ; (P_4)
X – – Y – – Z – – ; (P_5)
G00 X – – Y – – Z – – ;

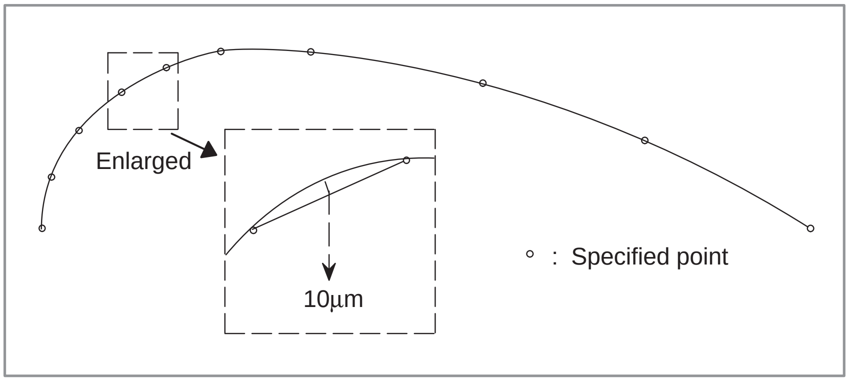


3.16 SPIRAL INTERPOLATION AND CONICAL INTERPOLATION

Spiral interpolation can be carried out when the circular interpolation command is specified together with the number of circles of the helix or a radius increment or decrement per circle.
Conical interpolation can be carried out when the spiral interpolation command is specified together with commands specifying a movement along another axis and an increment or decrement along the axis per circle of the helix.

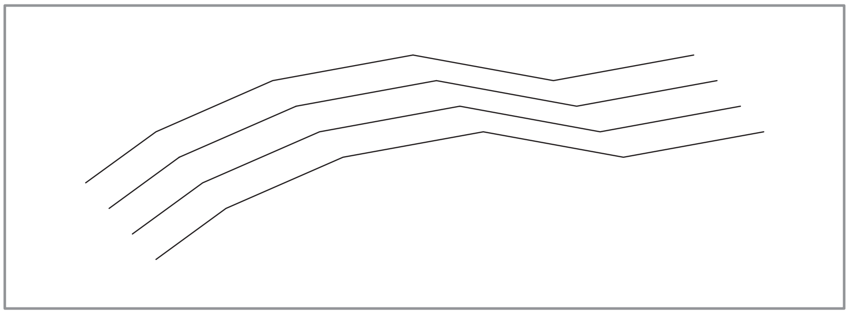
3.17 SMOOTH INTERPOLATION FUNCTION

To machine a part having sculptured surfaces, such as metal moldings used in automobiles and airplanes, a part program usually approximates the sculptured surfaces with minute line segments. As shown in the following figure, a sculptured curve is normally approximated using line segments with a tolerance of about 10µm.



When a program approximates a sculptured curve with line segments, the length of each segment differs between those portions that have mainly a small radius of curvature and those that have mainly a large radius of curvature. The length of the line segments is short in those portions having a small radius of curvature, while it is long in those portions having a large radius of curvature. The high-precision contour control of the FANUC Series 15 moves the tool along a programmed path thus enabling highly precise machining. This means that the tool movement precisely follows the line segments used to approximate a sculptured curve. This may result in a non-smooth machined curve if control is applied to machining a curve where the radius of curvature is large and changes only gradually. Although this effect is caused by high-precision machining, which precisely follows a pre- programmed path, the uneven corners that result will be judged unsatisfactory when smooth surfaces are required.

Profile	Portions having mainly a small radius of curvature	Portions having mainly a large radius of curvature
Example of machined parts	Automobile parts	Decorative parts, such as body side moldings
Length of line segment	Short	Long
Resulting surfaces produced using high-precision contour control	Smooth surface even when machining is performed exactly as specified by a program	Uneven surfaces may result when machining is performed exactly as specified by a program



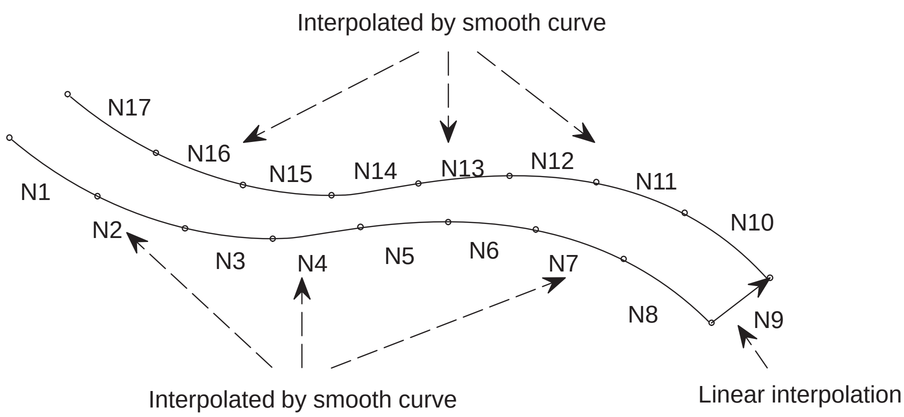
Example of uneven surfaces (polygon) resulting from machining that precisely follows the line segments

The smooth interpolation function enables high- speed, high- precision machining, as follows:
The CNC automatically selects either of two types of machining according to the program command.

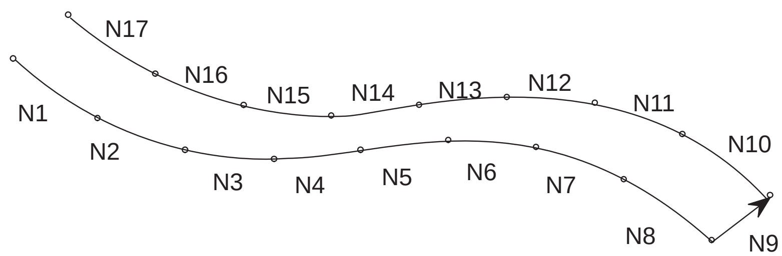
- For those portions where the accuracy of the figure is critical, such as at corners, machining is performed exactly as specified by the program command.
- For those portions having a large radius of curvature where a smooth figure must be created, points along the machining path are interpolated with a smooth curve, calculated from the polygonal lines specified with the program command (smooth interpolation).

Use the following command to specify smooth interpolation mode:
G05.1 Q2 X0 Y0 Z0 ;
The CNC automatically selects either of the above machining types, according to the program command. If a block specifies a travel distance or direction which differs greatly from that in the preceding block, smooth interpolation is not performed for that block, but linear interpolation is performed exactly as specified by the program command. Programming is thus very simple.

Example



Linear interpolation



4 THREAD CUTTING

4.1 EQUAL LEAD THREAD CUTTING (G33)

By feeding the tool synchronizing with the spindle rotation, thread cutting of the specified lead is performed. Specify lead of the long axis (an axis along which the tool travels longest distance) direction with the F code.

Table 4.1

	Increment		Allowable range of lead	
Input in millimeters	0.01	mm	0.0001 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
	0.00001	mm	0.0000001 to 5.0000000	mm/rev
	0.000001	mm	0.00000001 to 0.50000000	mm/rev
Input in inches	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
	0.000001	inch	0.00000001 to 0.50000000	inch/rev

The spindle must be equipped with a position coder.
Thread cutting start position (the starting point of the thread cutting, synchronizing with the spindle rotation) can be shifted. This is useful when cutting multiple thread. Specify the desired angle with Q.

Format

G33 IP_ _ F_ _ Q_ _ ;

where

F_ _ : Lead of the long axis

Q_ _ : Shift angle of thread start position (0° to 360°)

NOTE

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

4.2 INCH THREAD CUTTING (G33)

By specifying threads per inch of the long axis by the E code, inch thread cutting is performed. Thread cutting start position can be shifted.

Format

G33 IP_ _ E_ _ Q_ _ ;

where

E_ _ : Threads per inch of the long axis

4.3 CONTINUOUS THREAD CUTTING

Continuous thread cutting in which thread cutting command blocks are continuously commanded is available.

As it is controlled so that the spindle synchronism shift (occurred when shifting from one block to another) is kept to a minimum, special threads like threads which leads or shape change during the cycle can also be cut.

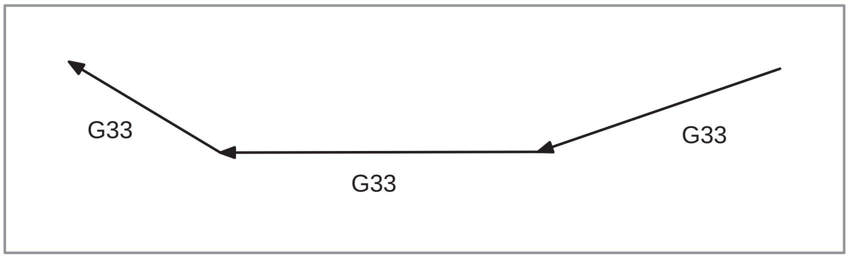


Fig. 4.3

5 FEED FUNCTIONS

5.1 RAPID TRAVERSE

Positioning of each axis is done in rapid motion by the positioning command (G00).
There is no need to program rapid traverse rate, because the rates are set in the parameter (per axis).

Table 5.1

	Least command increment	Rapid traverse rate range
Machine of mm system	0.01 mm	40 to 2400000 mm/min, deg/min
	0.001 mm	4 to 240000 mm/min, deg/min
	0.0001 mm	0.4 to 100000 mm/min, deg/min
	0.00001 mm	0.04 to 10000 mm/min, deg/min
	0.000001 mm	0.004 to 1000 mm/min, deg/min
Machine of inch system	0.001 inch	4 to 240000 inch/min
	0.0001 inch	0.4 to 24000 inch/min
	0.00001 inch	0.04 to 10000 inch/min
	0.000001 inch	0.004 to 1000 inch/min
	0.0000001 inch	0.0004 to 100 inch/min

5.2 CUTTING FEEDRATE

Feedrates of linear interpolation (G01), and circular interpolation (G02, G03) are commanded with numbers after the F code.

5.2.1 Tangential Speed Constant Control

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

5.2.2 Cutting Feedrate Clamp

Cutting feedrate upper limit can be set as parameters of each axis. If the actual cutting feedrate (feedrate with override) is commanded exceeding the upper limit, it is clamped to a speed not exceeding the upper limit.

5.2.3 Per Minute Feed (G94)

With the per minute feed mode G94, tool feedrate per minute is directly commanded by numerical value after F.

Table 5.2.3

	Least command increment	Cutting feedrate range
Input in mm Machine of mm system	0.01 mm	0.0001 to 2400000 mm/min, deg/min
	0.001 mm	0.0001 to 240000 mm/min, deg/min
	0.0001 mm	0.0001 to 100000 mm/min, deg/min
	0.00001 mm	0.0001 to 10000 mm/min, deg/min
	0.000001 mm	0.0001 to 1000 mm/min, deg/min
Input in inch Machine of inch system	0.001 inch	0.00001 to 240000 inch/min
	0.0001 inch	0.00001 to 24000 inch/min
	0.00001 inch	0.00001 to 10000 inch/min
	0.000001 inch	0.00001 to 1000 inch/min
	0.0000001 inch	0.00001 to 100 inch/min
Input in mm Machine of inch system	0.01 mm	0.0001 to 2400000 mm/min, deg/min
	0.001 mm	0.0001 to 240000 mm/min, deg/min
	0.0001 mm	0.0001 to 100000 mm/min, deg/min
	0.00001 mm	0.0001 to 10000 mm/min, deg/min
	0.000001 mm	0.0001 to 1000 mm/min, deg/min
Input in inch Machine of mm system	0.001 inch	0.00001 to 96000 inch/min
	0.0001 inch	0.00001 to 9600 inch/min
	0.00001 inch	0.00001 to 4000 inch/min
	0.000001 inch	0.00001 to 400 inch/min
	0.0000001 inch	0.00001 to 40 inch/min

5.2.4 Per Revolution Feed (G95)

With the per revolution feed mode G95, tool feedrate per revolution of the spindle is directly commanded by numeral after F. A position coder must be mounted on the spindle.

Table. 5.2.4

Least command increment		Cutting feedrate range			
0.01	mm or deg	0.0001	to	5000.0000	mm/rev or deg/rev
0.001	mm or deg	0.00001	to	500.00000	mm/rev or deg/rev
0.0001	mm or deg	0.000001	to	50.000000	mm/rev or deg/rev
0.00001	mm or deg	0.0000001	to	5.0000000	mm/rev or deg/rev
0.001	inch	0.00001	to	500.0000	inch/rev
0.0001	inch	0.000001	to	50.00000	inch/rev
0.00001	inch	0.0000001	to	5.000000	inch/rev
0.000001	inch	0.00000001	to	0.50000000	inch/rev

The above feedrates are limits according to the NC's interpolation capacity. When the whole system is considered, there are also limits according to the servo system.

5.2.5 Inverse Time Feed (G93)

Inverse time feed mode is commanded by G93, and inverse time by F code. Inverse time is commanded with the following value in a 1/min unit.

In linear interpolation F= Speed/distance
In circular interpolation F= Speed/radius

Command F0 for rapid traverse.

5.2.6 F1–digit Feed

When a 1-digit number from 1 to 9 is commanded after the F, the preset speed corresponding the 1-digit number commanded is set as feedrate.
Set the F1-digit feedrate change input signal on from the machine side, and rotate the manual pulse generator. Feedrate of the currently selected speed can be changed.
Feedrate set or changed will be memorized even after power is turned off.

5.3 OVERRIDE

5.3.1 Feedrate Override

The per minute feed (G94), per rotation feed (G95) and the inverse time feed (G93) can be overrided by:

0 to 254% (per every 1%).

In inverse time, feedrate converted to per minute feed is overridden. Feedrate override cannot be performed to F1-digit feed.

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

5.3.2 Second Feedrate Override

All cutting feedrate can be overrided by:

0 to 254% (per every 1%)

A second override can be performed on feed rats once overrided.

No override can be performed on functions as thread cutting and tapping in which override is inhibited.

This function is used for controlling feedrate in adaptive control, etc.

5.3.3 Second Feedrate Override B

This function selects the second feedrate override in the range from 0 to 655.34 with 0.01% increments.

5.3.4 Rapid Traverse Override

Rapid traverse rate can be overridden by:

F0, F1, 50, 100%

F0: A constant speed per axis can be set by parameter

F1: A constant % can be set by parameter

5.3.5 Function for Overriding the Rapid Traverse Feedrate in 1% Unit

This function overrides the rapid traverse feedrate with a value (1% units) entered from the machine operator’s panel in the range from 0% to 100%. By specifying the ROV8 bit of parameter 1402, the override selected by this function can be switched to and from the standard override for the rapid traverse feedrate (F0, Fn, 50%, 100%).

5.3.6 Override Cancel

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

5.4 AUTOMATIC ACCELERATION/ DECELERATION

Acceleration and deceleration is performed when starting and ending movement, resulting in smooth start and stop.

Automatic acceleration/deceleration is also performed when feedrate changes, so change in speed is also smoothly done.

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

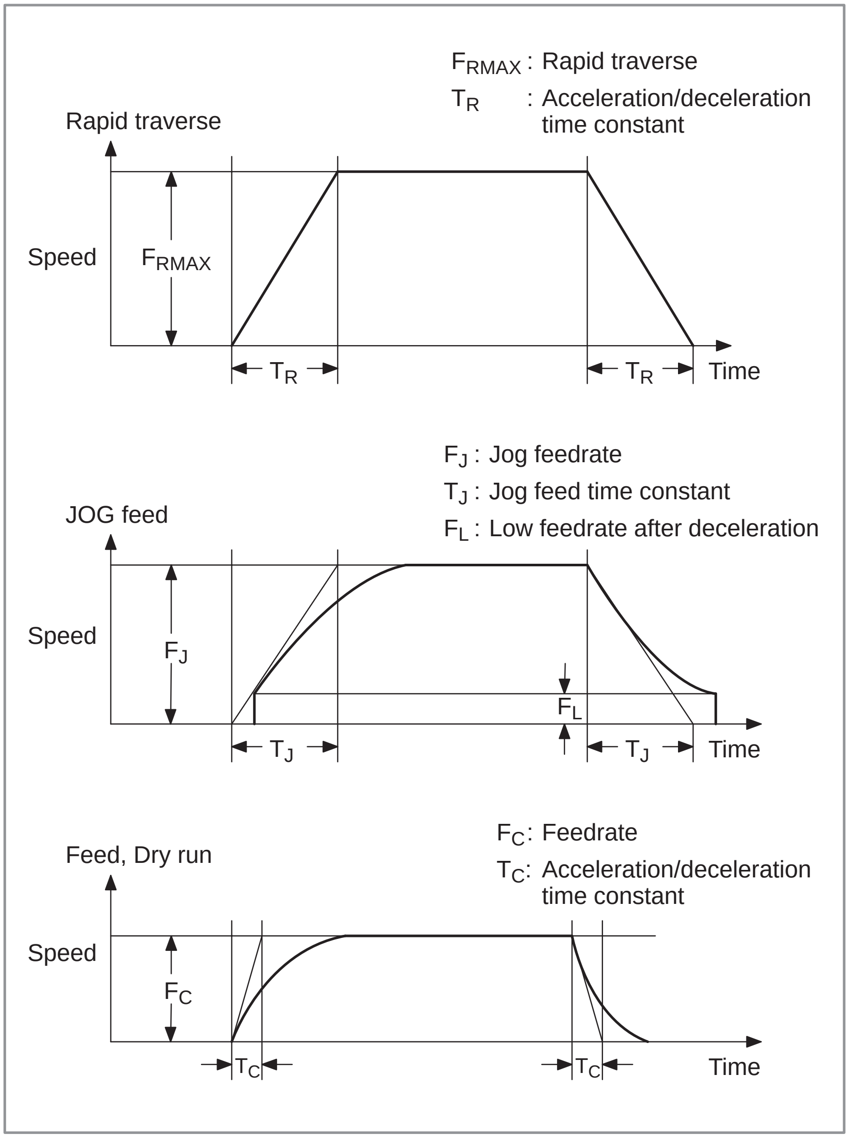


Fig. 5.4

5.5 LINEAR ACCELERATION/ DECELERATION AFTER CUTTING FEED INTERPOLATION

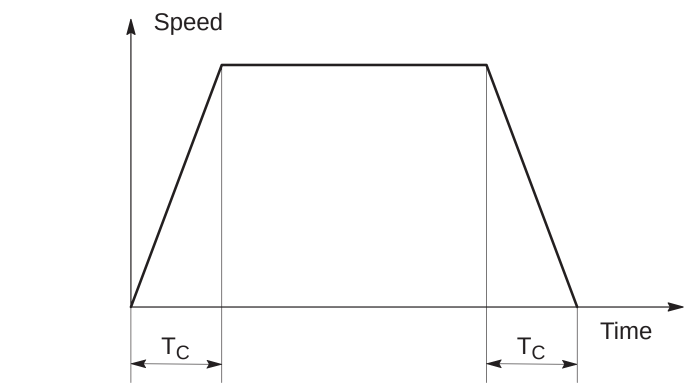


Fig. 5.5 (a)

In the linear acceleration/deceleration, the delay for the command caused by the acceleration/deceleration becomes 1/2 compared with that in exponential acceleration/deceleration, substantially reducing the time required for acceleration and deceleration.
Also, the radius direction error in the circular interpolation caused by the acceleration/deceleration is substantially reduced.

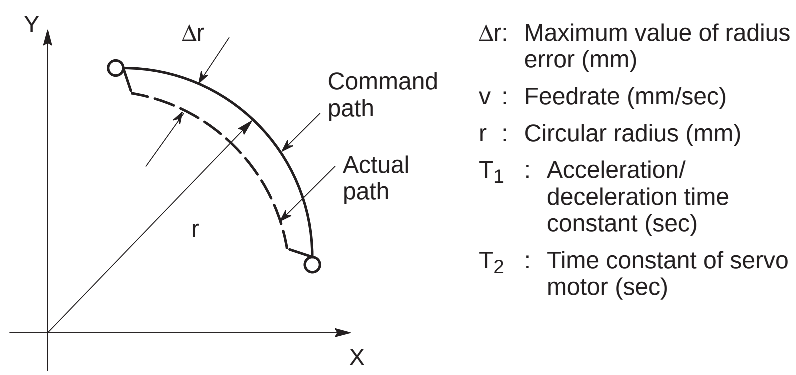


Fig. 5.5 (b)

The maximum value of error in this radius direction is obtained approximately by the following equation.

$$\Delta r = \left(\frac{1}{2}T_1{}^2 + \frac{1}{2}T_2{}^2\right)\frac{V^2}{r} \quad \ldots\ldots \text{ For exponential acceleration/deceleration}$$

$$\Delta r = \left(\frac{1}{24}T_1{}^2 + \frac{1}{2}T_2{}^2\right)\frac{V^2}{r} \quad \ldots\ldots \text{ For linear acceleration/deceleration after cutting feed interpolation}$$

Consequently, in case of the linear acceleration/deceleration after interpolation, if an error caused by the servo loop time constant is excluded, the radius directional error will be reduced to 1/12, compared with the exponential acceleration/deceleration.

5.6 BELL–SHAPED ACCELERATION/ DECELERATION AFTER CUTTING FEED INTERPOLATION

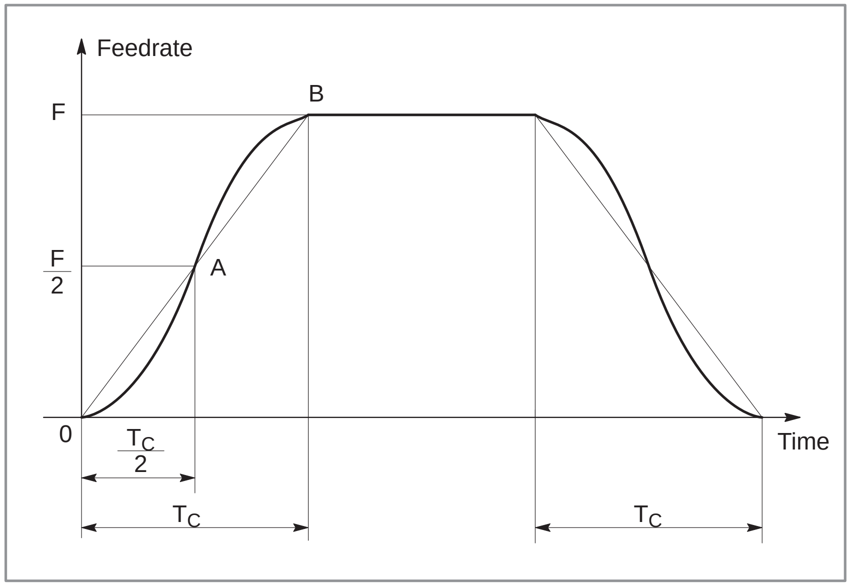


Fig. 5.6

As shown above in the quadratic curve, it is possible to accelerate and decelerate the cutting feedrate.
When the acceleration and deceleration section are connected, the composed curve shapes just like a hanging bell. That is why this kind of acceleration/deceleration is called bell–shaped acceleration/deceleration.
Considering a time constant as Tc (time spent to accelerate from feedrate 0 up to commanded feedrate F or time spent to decelerate from commanded feedrate F down to feedrate 0), feedrate accelerates up to 1/2 of the commanded feedrate (F/2) for 1/2 of the time constant (Tc/2). The acceleration/deceleration curve 0A shown in the figure above can be expressed by the following equation :

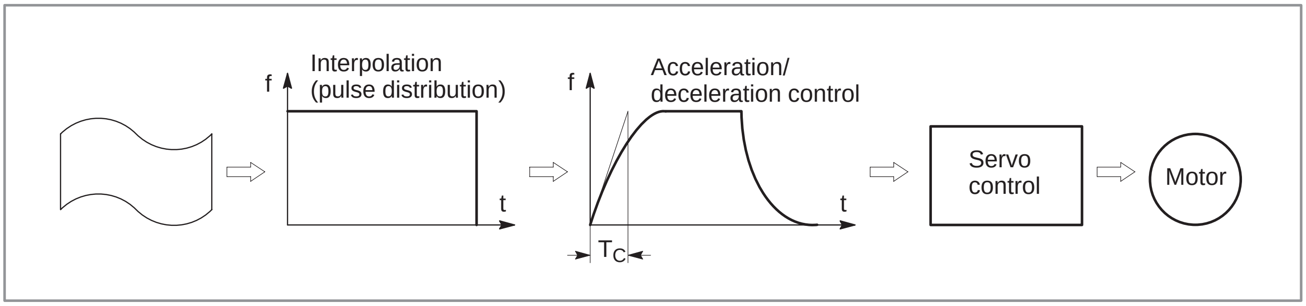
$$f(t) = \frac{2F}{{T_C}^2} t^2$$

The curve AB and 0A are symmetric with respect to point A.
The feature of this acceleration/deceleration is that the feedrate change is small near feedrate 0 and the commanded feedrate.

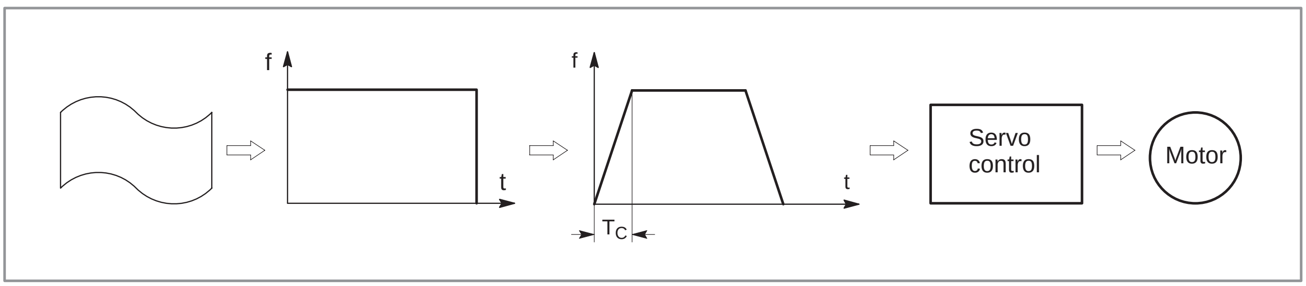
5.7 ACCELERATION/ DECELERATION BEFORE CUTTING FEED

In response to the cutting feed command, the feedrate before interpolation, that is, the command feedrate can be directly accelerated/ decelerated. This enables a machined shape error caused by the delay of acceleration/deceleration to be eliminated. However, the deceleration command (G09) needs to be given to the block requiring deceleration such as a corner by the program.

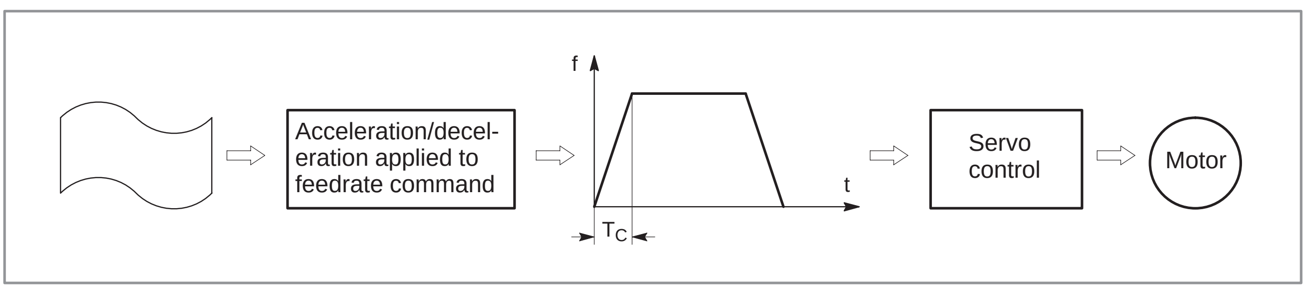
- **Exponential acceleration/deceleration after feed interpolation**



- **Linear acceleration/ deceleration after feed interpolation**



- **Linear acceleration/ deceleration before feed interpolation**

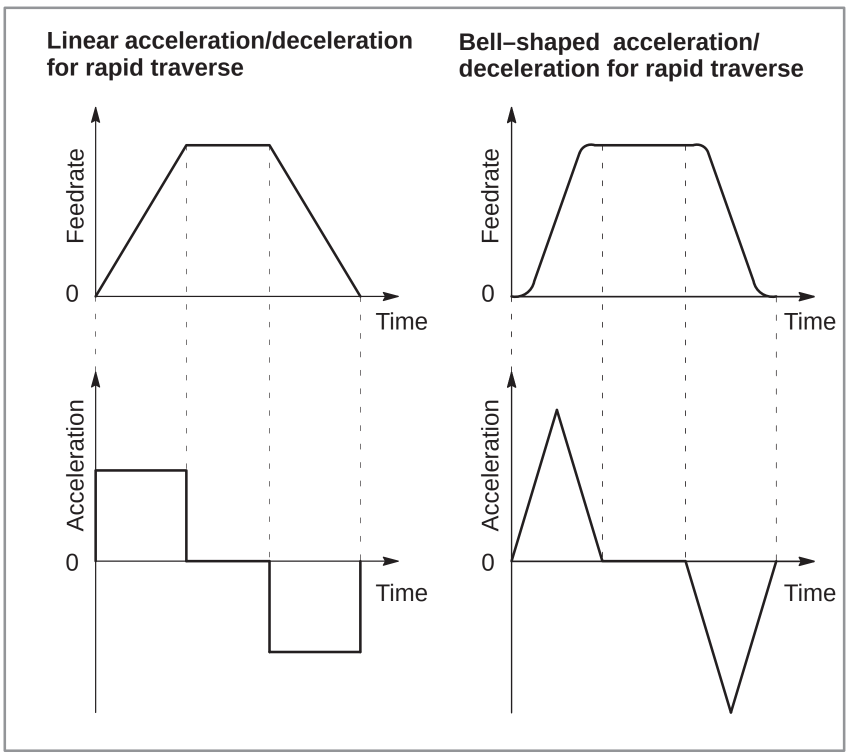


5.8 ACCELERATION/ DECELERATION BEFORE PRE–READ INTERPOLATION

Acceleration/deceleration before Pre–read Interpolation has the advantage that there is no machined shape error caused by the delay of acceleration/deceleration. However, the deceleration command (G09) must be given to the block such as a corner with a large speed change of either axis. Therefore, it has the disadvantage that acceleration/ deceleration needs to be considered by program commands. However, this function allows automatic judgement of whether the speed is decelerated by reading the command up to 15 blocks ahead, thus making it unnecessary to consider acceleration/deceleration during creation of program commands and eliminating any machined shape error caused by the delay of acceleration/deceleration.

5.9 BELL–SHAPED ACCELERATION/ DECELERATION AFTER RAPID TRAVERSE INTERPOLATION

The function for bell–shaped acceleration/deceleration after rapid traverse interpolation increases or decreases the rapid traverse feedrate smoothly.
This reduces the shock to the machine system due to changing acceleration when the feedrate is changed.
As compared with linear acceleration/deceleration, bell–shaped acceleration/deceleration allows smaller time constants to be set, reducing the time required for acceleration/deceleration.



5.10 CUTTING POINT SPEED CONTROL FUNCTION

The cutting point speed control function is used when circular interpolation is performed in the cutter compensation C mode. This function allows a programmed feedrate to be used as the feedrate at the cutting point rather than the feedrate at the center of the tool.
This function is enabled or disabled by setting the CAFC bit of parameter 1402.

5.11 ACCELERATION/ DECELERATION FUNCTION FOR THE CONSTANT SPEED SPECIFIED BY THE PMC AXIS CONTROL FUNCTION

This function accelerates and decelerates the machining feedrate to maintain the speed specified by the PMC axis control function throughout the cutting process.

This function linearly accelerates and decelerates the tool to ensure smooth operation during the entire cutting process. Also, when the feedrate changes in the middle of the cutting process, this function automatically accelerates or decelerates the speed to prevent irregular tool movement.

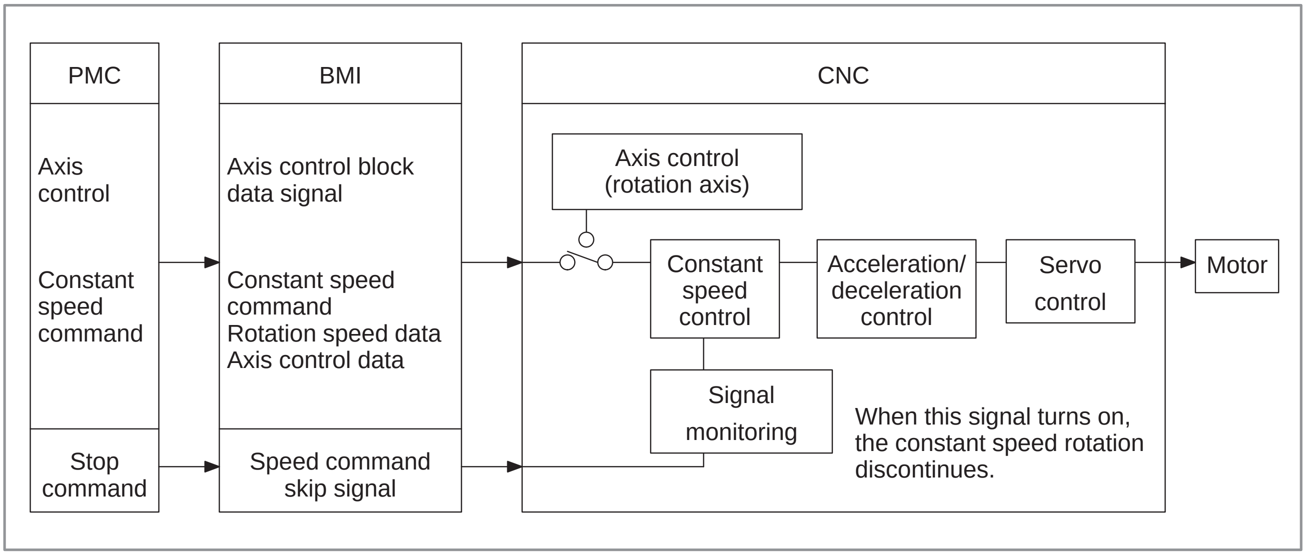


Fig. 5.11 Block diagram of system operation when the constant speed is specified

5.12 EXACT STOP (G09)

Move command in blocks commanded with G09 decelerates at the end point, and inposition check is performed. G09 command is not necessary for deceleration at the end point for positioning (G00) and inposition check is also done automatically. This function is used when sharp edges are required for workpiece corners in cutting feed.

5.13 CUTTING/RAPID TRAVERSE POSITION CHECK FUNCTION

In a block in which a positioning block or an exact stop command is specified, such as a cutting feed block, the cutting speed is decelerated at the end of the block to perform the position check. The cutting/rapid traverse position check function allows the operator to set the effective area size. Using this function, a small effective area can be specified for cutting feed blocks requiring accuracy and a large effective area can be specified for positioning blocks requiring a shorter positioning time.

5.14 EXACT STOP MODE (G61)

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

5.15 CUTTING MODE (G64)

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 till G61 (exact stop mode), G62 (automatic corner override), or G63 (tapping mode) is commanded.

5.16 TAPPING MODE (G63)

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. And in-tapping mode signal is issued during tapping operation. This G63 is valid till G61 (exact stop mode), G62 (automatic corner override), or G64 (cutting mode) is commanded.

5.17 AUTOMATIC CORNER OVERRIDE (G62)

When G62 is commanded during cutter compensation, cutting feedrate 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.

5.18 DWELL (G04)

With the G04 command, shifting to the next block can be delayed.
When commanded with a per minute feed mode (G94), shifting to the next block can be delayed for the commanded minutes.
When commanded with a per rotation feed mode (G95), shifting to the next block can be delayed till the spindle rotates for the commanded times.
Dwell may always be performed by time irrespective of G94 and G95 by parameter selection.

Format

Per second dwell

G94 G04 $\left\{ \begin{array}{l} \textbf{P\_ \_} \\ \textbf{X\_ \_} \end{array} \right\}$ **;**

P_ _ or X_ _: Dwell time commanded in seconds (0.001 to 99999.999 sec)

Per revolution dwell

G95 G04 $\left\{ \begin{array}{l} \textbf{P\_ \_} \\ \textbf{X\_ \_} \end{array} \right\}$ **;**

P_ _ or X_ _: Spindle rotation angle commanded in rev. (0.001 to 99999.999 rev)

5.19 FEED PER ROTATION WITHOUT A POSITION CODER

This function provided for machines that do not have (or use) a position coder. When a feedrate is specified in the feed–per–rotation mode, it is converted to a feedrate in the feed–per–minute mode on the assumption that the spindle turns according to the spindle speed command (S code). The tool is then moved along the feed axis at the converted feedrate.

Example)

G95 G01 F1. S1000 Z100. ;

When the above command is specified, the tool is moved along the Z–axis at F1000 in the feed–per–minute mode [mm/min], on the assumption that the spindle turns at 1000 revolutions per minute.

NOTE
In this function, S1 corresponds to one rpm.

6 REFERENCE POSITION

6.1 MANUAL REFERENCE POSITION RETURN

Positioning to the reference position can be done by manual operation.
With jogging mode (J), manual reference point return (ZRN) signals, and signal for selecting manual reference position return axis (±J1 to ±J6) on, the tool begins to move at rapid traverse. When deceleration limit switch mounted on the machine is turned on, it decelerates, and when it is turned off again, it stops at the first grid point, and reference position return end lamp lights.
This point is the reference position.
By performing manual reference position return, the machine coordinate system and the work coordinate system is established.
There are the following two methods to perform manual reference position return:

1) Grid method
   A certain grid of the position detection is appointed as the reference position. The reference position can be shifted by the grid shift function.

2) Magne–switch method
   The rise point of the proximity switch on the machine is appointed as the reference position.

6.2 AUTOMATIC REFERENCE POSITION RETURN (G28, G29)

1) Return to reference position (G28)
   With the G28 command, the commanded axis is positioned to the reference position via the commanded point. After positioning, the reference position return end lamp lights. If G28 was commanded when reference position return is not performed after power on, reference position return is done in the same sequence as the manual reference position return.

Format

G28 IP_ _ ;

IP_ _ : Command intermediate point

2) Return from reference position (G29)
   With the G29 command, the commanded axis is positioned to the point commanded by G29, via the intermediate point commanded by G28.

Format

G29 IP_ _ ;

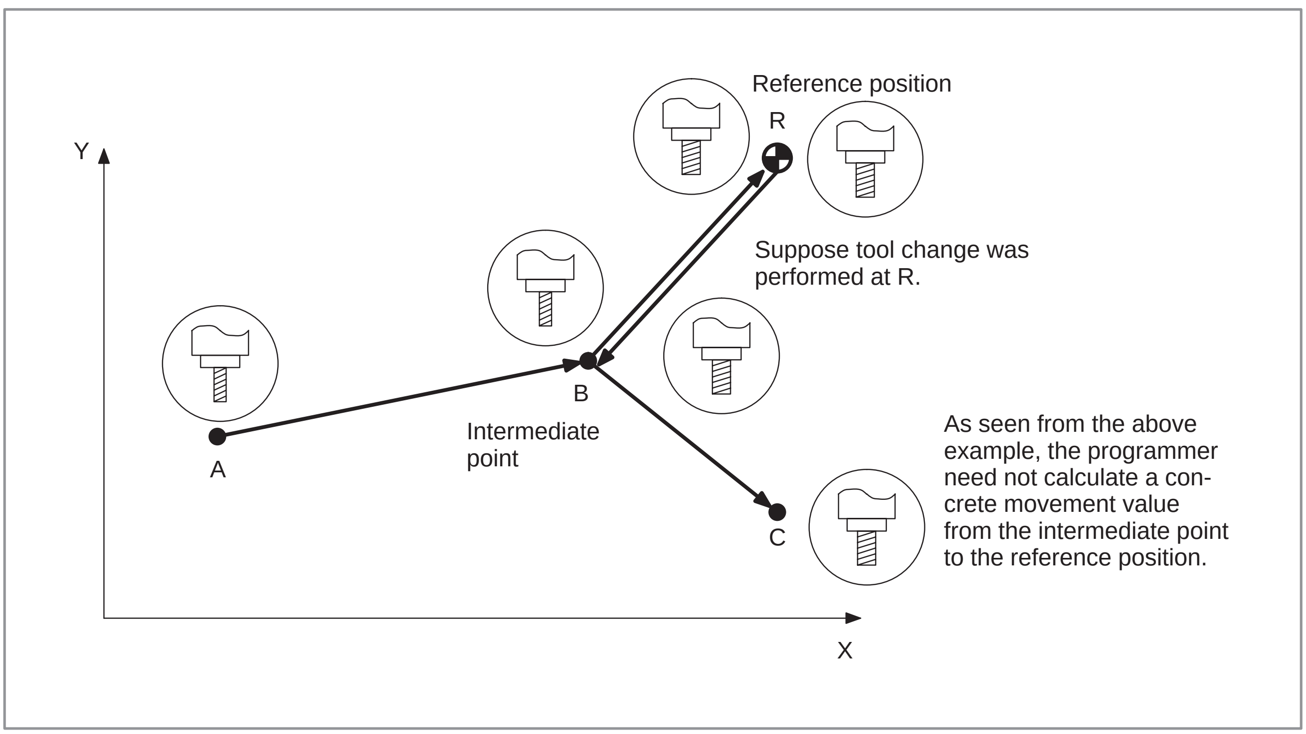


Fig. 6.2 Example of use of G28 and G29

6.3 REFERENCE POSITION RETURN CHECK (G27)

This function is used to check whether the reference position return command was performed correctly.
When G27 is commanded, the commanded axis is positioned to the specified position, reference position return end lamp lights if reference position return is performed to the correct position, and alarm arises it is not positioned correctly to the reference position.
This function is available after power is turned on and reference position return is performed.

Format

G27 IP_ _ ;

6.4 2ND, 3RD AND 4TH REFERENCE POINT RETURN (G30)

With the G30 command, the commanded axis is positioned to the 2nd, 3rd, or the 4th reference position, via the commanded point. 2nd, 3rd, or 4th reference position return lamp lights when positioning ends.
Set the 2nd, 3rd, and 4th reference position position as parameters.
This function is available after power is turned on and reference position return is performed.
G29 can be used to return from the 2nd, 3rd, and 4th reference point (same as reference position return, G28).

Format

$$\mathbf{G30} \left\{ \begin{array}{c} \mathbf{P2} \\ \mathbf{P3} \\ \mathbf{P4} \end{array} \right\} \mathbf{IP\_\ \_\ ;}$$

where
P2, P3, P4: Select from 2nd, 3rd, or 4th reference positions. If not selected, 2nd reference position return is automatically selected.

6.5 FLOATING REFERENCE POSITION RETURN (G30.1)

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 reference position of some sort of machine operation. It is not always a fixed position but may vary in some cases. The floating reference position can be set using the soft keys of MDI and can be memorized even if the power is turned off.
Generally, the position where the tools can be replaced on machining center, milling machine is a set position on top of the machinery. The tools cannot be replaced at any machine angle. Normally the tool replacement position is at any of the No. 1 to No. 4 reference position. The tool can be restored 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 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, tool replacement position must be changed for each workpiece shape and this feature can be easily realized by this function. Namely, the tool replacement position which is suitable for workpiece can be memorized as the floating reference position and it is possible to return the tool to the tool replacement position easily by commanding the G30.1.

Format

G30.1 IP_ ;

, however **IP_** : It is the intermediate point to the floating reference position and is commanded by an absolute value or an incremental value.

When the G30.1 is commanded, the axis commanded is set to the intermediate point with rapid traverse at first and then is set to the floating reference position from the intermediate point with rapid traverse. The positioning to the intermediate point or to the floating point is performed at rapid traverse for each axis (non-linear positioning). When the BMI interface is used, the floating reference position return completion signal is output after completing the floating reference position return.

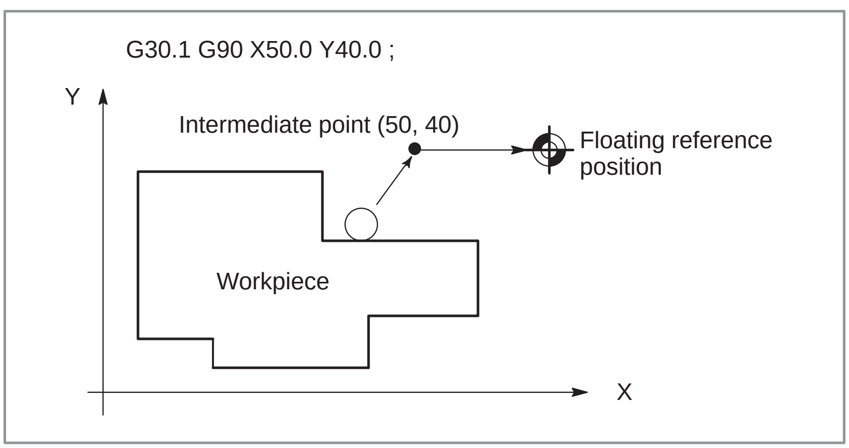
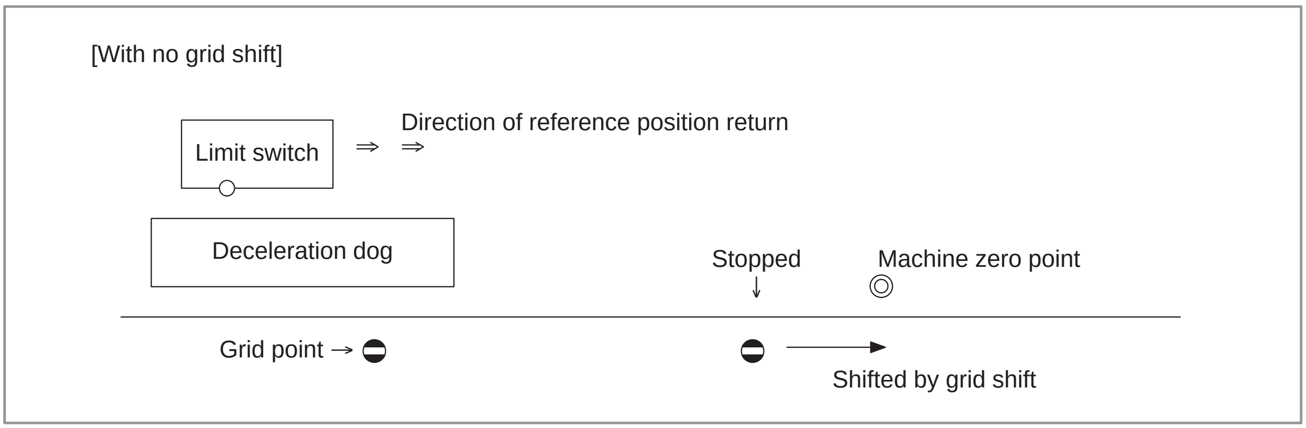


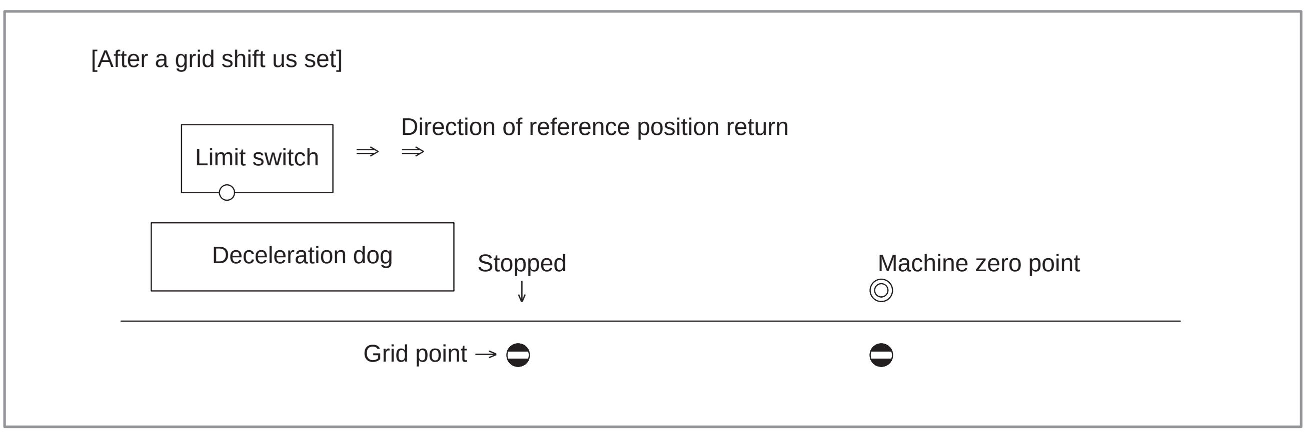
Fig. 6.5 Example of use G30.1

6.6 REFERENCE POSITION AUTOMATIC SETTING FUNCTION

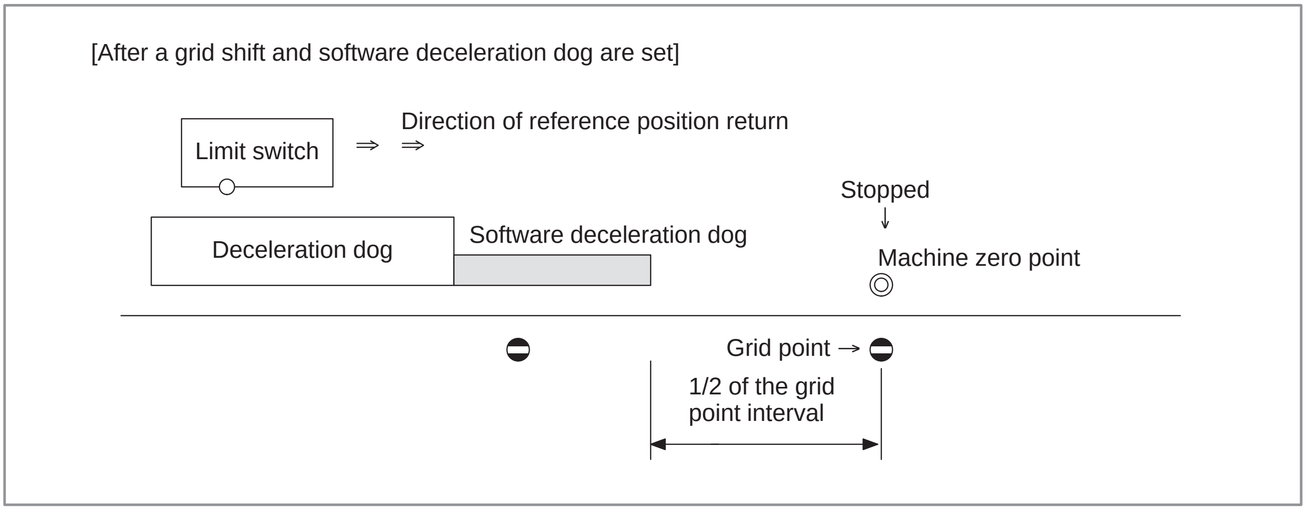
When adjusting a deceleration dog, the user should be able to adequately match an electrical grid point with the machine zero point. When the automatic reference position setting function is used, the grid shift and software deceleration dog amount can be set automatically by moving from a grid point where a stop is to take place, to the machine zero point. This movement is made by turning on and off the automatic reference position setting signals (RAST1, RAST2, RAST3, ...) and by manual operation (jog feed, manual handle feed).
In reference position return, the grid shift and software deceleration dog function as follows:
When reference position return is performed, the tool stops at the position where the first grid signal was detected after the limit switch of the deceleration dog was passed.
This position is the electrical stop position. It must match the machine zero point.



To match the electrical stop position with the machine zero point, a grid shift is set automatically. From the electrical stop position, the grid point can be shifted by +1/2 of a grid point interval.



Furthermore, a software deceleration dog is automatically set (software extension of the deceleration dog). The software deceleration dog can be used to match the electrical stop position with the machine zero point by turning off the deceleration dog 1/2 of grid point interval from the machine zero point.



6.7 DOG–LESS REFERENCE POSITION SETTING FUNCTION

The dog–less reference position setting function is used for cutting machines equipped with an absolute–position detector. This function allows the operator to set the reference point without the deceleration signal when manually moving the tool close to the reference point specified for each axis in the reference point return mode. Using this function, the reference point can be set at any desired position without using the reference point return deceleration signal.

7 COORDINATE SYSTEMS

7.1 MACHINE COORDINATE SYSTEM (G53)

Machine coordinate system is a coordinate system set with a zero point proper to the machine system.
A coordinate system in which the reference point becomes the parameter-preset coordinate value when manual reference point return is performed, is set.
With G53 command, the machine coordinate system is selected and the axis is moved in rapid traverse to the position expressed by the machine coordinates.

Format

G53 IP_ _ ;

7.2 WORKPIECE COORDINATE SYSTEM (G54 TO G59)

A coordinate system in which the zero point is set to a fixed point on the workpiece, to make programming simple. When actually machining, distance between machine coordinates' zero point and work coordinates' zero point is measured and set as workpiece zero point offset quantity via MDI. 6 type of workpiece coordinates can be set and selected with:
G54 – G59

Format

$$\left\{ \begin{array}{c} \textbf{G54} \\ \textbf{G55} \\ : \\ \textbf{G59} \end{array} \right\} \textbf{IP\_ \_ ;}$$

7.3 LOCAL COORDINATE SYSTEM (G52)

With G52 commanded, the local coordinate system with the commanded position as zero point can be set. 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 workpice coordinates' zero point.

Format

G52 IP_ ;

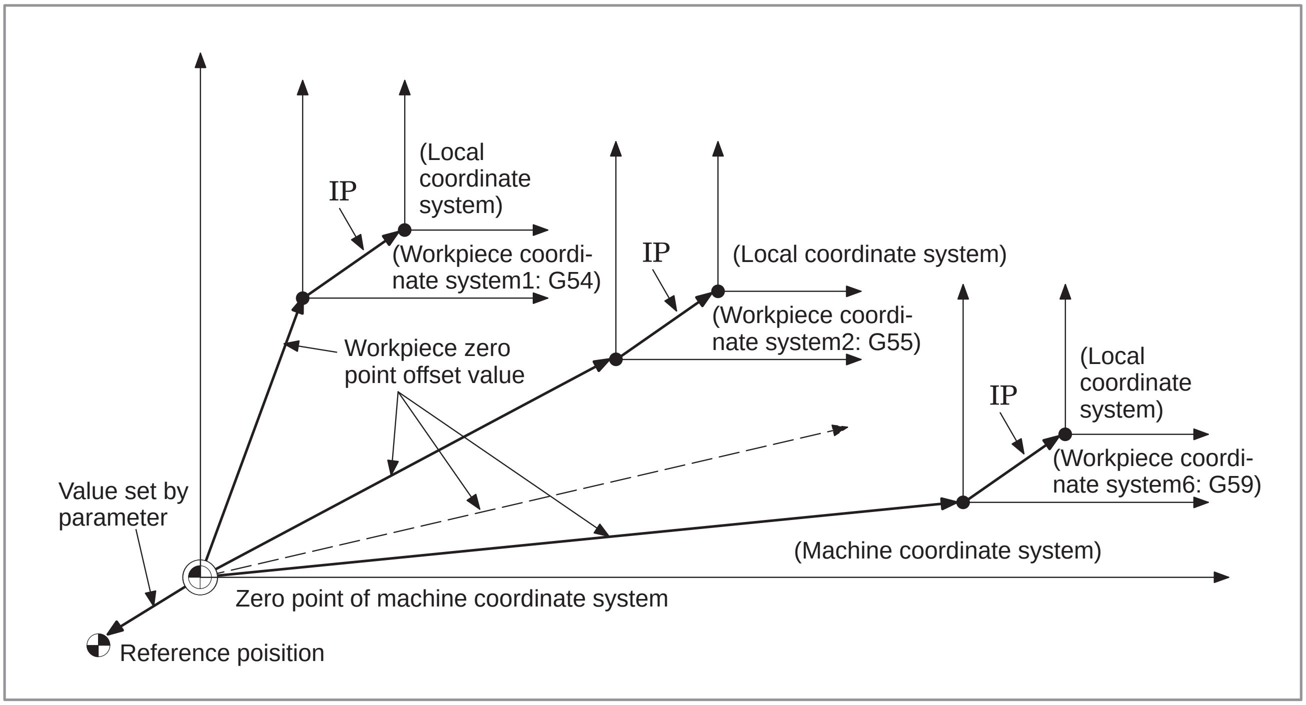


Fig. 7.3

When local coordinate system is set, local coordinate system 1 - 6, corresponding to workpiece coordinate system 1 - 6 is set. Distance between zero points are all the same preset value.
If G52 IP0; is commanded, local coordinate system is canceled.

7.4 WORKPIECE COORDINATES SYSTEM CHANGE (G92)

With the
G92 IP_ _ ;
command, workpiece coordinate system can be changed so that current position of the tool becomes the specified position.

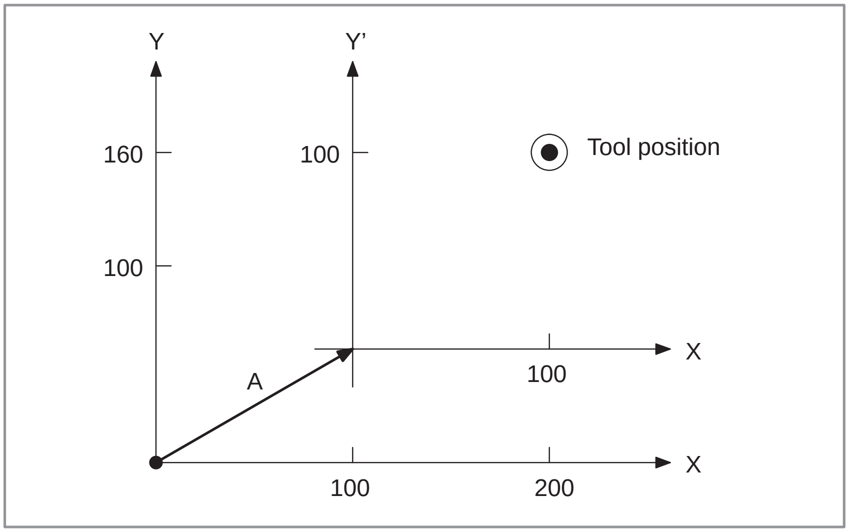


Fig. 7.4

If G92 X100 Z100 ; is commanded when the tool is positioned at (200, 160) in G54 mode, workpiece coordinate system 1 (X’ – Y’) displaced by vector A is created. At the same time, workpiece coordinate system 2 to 6 shift by vector A.

When creating a new workpiece coordinate system with the G92 command, since it is determined so that a certain point of the tool becomes a certain coordinate value, the new workpiece coordinate system can be determined irrespective of the old workpiece coordinate system. If the G92 command is used to determine a start point for machining based on workpieces, a new coordinate system can be created even if there is an error in the old workpiece coordinate system.

7.5 WORKPIECE ORIGIN OFFSET VALUE CHANGE (PROGRAMMABLE DATA INPUT) (G10)

Six workpiece coordinate systems can be set. But, when that is still not enough, or when workpiece origin offset value must be set by tape or changed, this G10 command is used to change workpiece origin offsets. When G10 is commanded in subsolute command (G90), the commanded workpiece origin offsets becomes the new workpiece origin offsets, and when G10 is commanded in incremental command (G91), the currently set workpiece origin offsets plus the commanded workpiece origin offset becomes the new workpiece offsets.

Format

G10 L2 P_P IP_ ;

where

P_P : Specifiy workpiece coordinates to which offsetts are changed
P : 1 to 6
IP : Workpiece origin offset value

7.6 ADDITIONAL WORKPIECE COORDINATE SYSTEMS (G54.1)

Forty-eight workpiece coordinate systems can be added when existing six workpiece coordinate systems (G54 - G59) are not enough for the operation. Make a command as follows for selection of workpiece coordinate system.

G54.1 P_P IP .....;

P: 1 – 48 Number of the additional workpiece coordinate system

The following are the methods of setting and changing of the workpiece origin offset value as well as those used for the existing workpiece coordinate systems of G54 to G59.

1) Method via CRT/MDI
2) Method via program
   - G10L20Pp;
   - Custom macro
3) Method of external workpiece coordinate system shift

The set workpiece origin offset value is displayed on the CRT. Also, the set workpiece origin offset can be punched out.

```
WORK ZERO OFFSET                O0001 N00100

   P: 01  (G54.1)    P: 03  (G54.1)
    X      123.456    X     –111.111
    Y      234.567    Y      222.222
    Z      345.678    Z     –333.333

   P: 02  (G54.1)    P: 04  (G54.1)
    X     –123.456    X      111.111
    Y      234.567    Y     –222.222
    Z     –345.678    Z      333.333

                                  (MM)
MDI  *** STOP **** *** *** 09:53:47  LSK
  INPUT  +INPUT  MEASURE TL_INP  INP_NO.+
```

Fig. 7.6

7.7 WORKPIECE COORDINATE SYSTEM PRESET (G92.1)

The workpiece coordinate system with its zero position away by the workpiece zero offset amount from the machine coordinate system zero position is set by returning the tool to the reference point 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 point return operation. The set workpiece coordinate may shift by any of the following commands or operation:

a) When manual interruption is performed with the manual absolute signal off
b) When the travel command is performed by the machine lock
c) When axis travel is performed by the handle interrupt or auto/manual simultaneous operation
d) When operation is performed by mirror image
e) When the setting of local coordinate system is performed by the G52 or change of workpiece coordinate system is performed by the G92
f) When origin setting of workpiece coordinate system is performed by the MDI operation

The workpiece coordinate system shifted by the above operation can be preset by the G code instruction or MDI operation the same as conventional manual reference point return.

1) Workpiece coordinate system preset by G code command
   The workpiece coordinate system can be preset by commanding the

Format

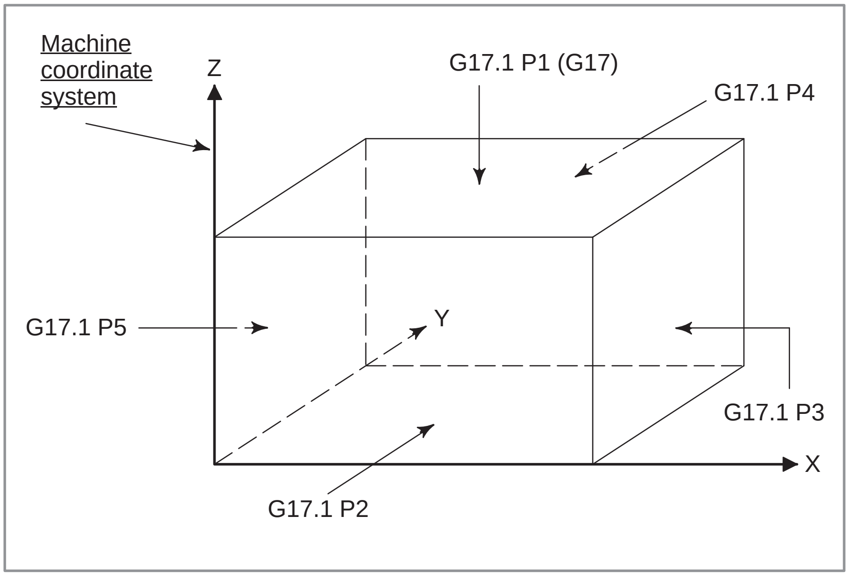
G92.1 IP 0 ;

IP 0 : The axis address to be preset the workpiece coordinate system
Uncommanded axis is not preset.

2) Workpiece coordinate system preset by MDI operation
   The workpiece coordinate system can be preset by the MDI operation with soft keys.

7.8 PLANE SWITCHING FUNCTION

This function switches a machining program created on the G17 plane in the right–hand Cartesian coordinate system to programs for other planes specified by G17.1Px commands, so that the same figure appears on each plane when viewed from the directions indicated by arrows.



8 COORDINATE VALUE AND DIMENSION

8.1 ABSOLUTE AND INCREMENTAL PROGRAMMING (G90, G91)

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

G90 : Absolute command
G91 : Incremental command

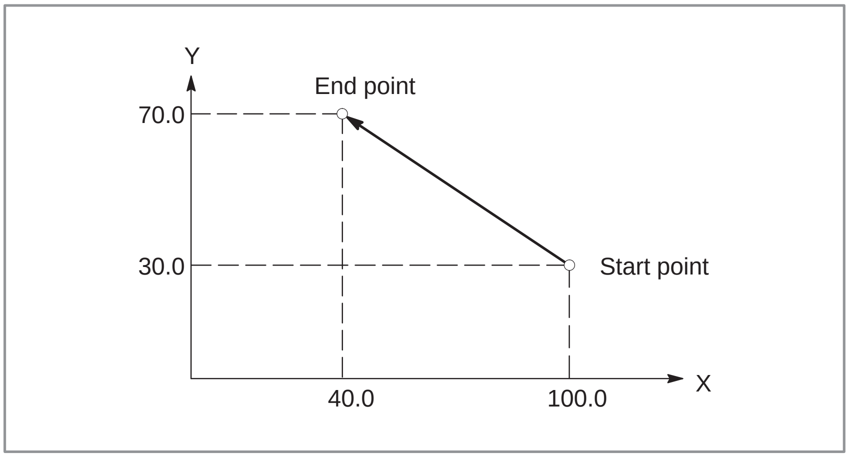


Fig. 8.1

For the above figure, incremental command programming results in:

G91X–60.0Y40.0 ;

while absolute command programming results in:

G90X40.0Y70.0 ;

8.2 POLAR COORDINATE COMMAND (G15, G16)

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

G15 : Polar coordinate system command cancel
G16 : Polar coordinate system command

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 command (G90, G91).

The center of the polar coordinates is the zero point of the local coordinates.

Example) Bolt hole cycle

N1 G17 G90 G16;	Polar coordinates command, X-Y plane
N2 G81 X100. Y30. Z-20. R-5. F200.;	100mm radius, 30° angle
N3 X100. Y150;	100mm radius, 150° angle
N4 X100. Y270;	100mm radius, 270° angle
N5 G15 G80;	Polar coordinates cancel

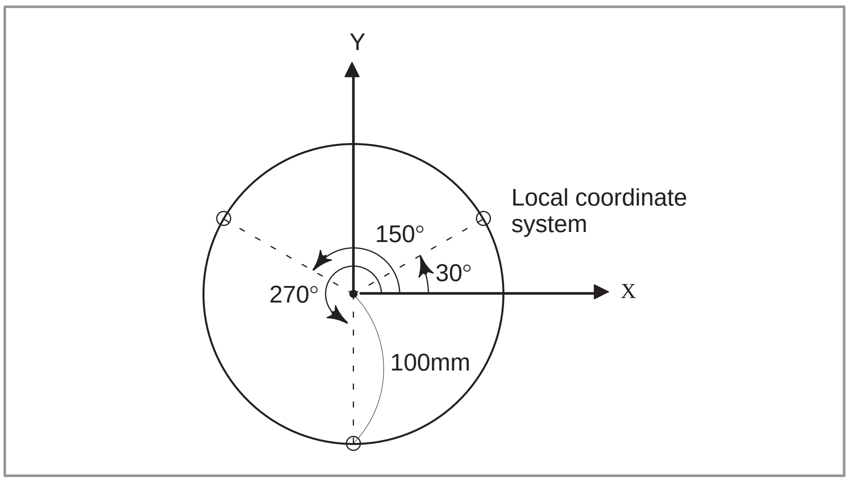


Fig. 8.2

8.3 INCH/METRIC CONVERSION (G20, G21)

Conversion of inch and metric input can be commanded by the G code command.

G20 : Inch input
G21 : Metric input

Whether the output is in inch system or metric system is parameter-set when the machine is installed.

Command G20, G21 at the head of the program.

Inch/metric conversation can also be done by MDI setting.

The contents of setting data differs depending on whether G20 or G21 is commanded.

8.4 DECIMAL POINT INPUT/POCKET CALCULATOR TYPE DECIMAL POINT INPUT

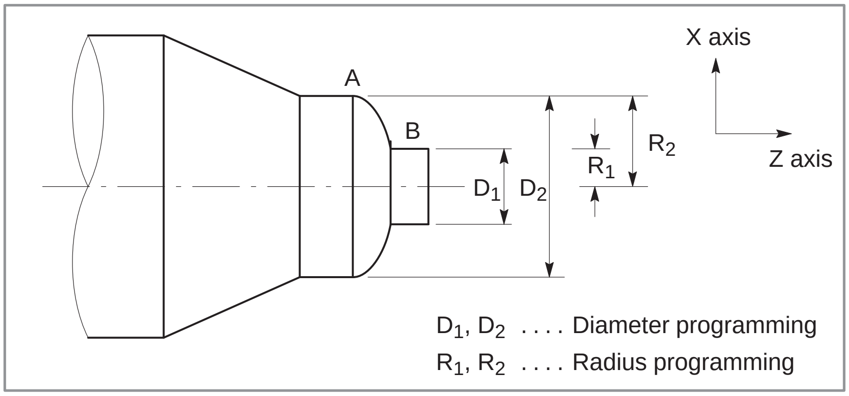
Numerals can be input with decimal points. Decimal points can be used basically in numerals with units of distance, speed, and angle. The position of the decimal point is at the mm, inch, deg position.
Use parameters to select input method; whether to input by pocket calculator type input, or by the former decimal point input.

Table 8.4

Program command	Pocket calculator type decimal point input	Format type decimal point input
X1000	1000 mm	1 mm
x1000	1000 mm	1000 mm

8.5 DIAMETER AND RADIUS PROGRAMMING

Since the work cross section is usually circular in latches, its dimensions can be specified in two ways when performing a thing:



When the diameter is specified, it is called diameter programming, and when the radius is specified, it is called radius programming.
The diameter programming or radius programming can be selected by parameter for each axis.

8.6 FUNCTION FOR SWITCHING BETWEEN DIAMETER AND RADIUS PROGRAMMING

In diameter/radius programming, the DIAx bit (bit 3 of parameter 1006) specifies whether to use diameter or radius programming for each controlled axis. With this function, the G code can switch between diameter and radius programming for axis commands.

9 SPINDLE FUNCTIONS

9.1 S CODE OUTPUT

The spindle speed is commanded in a 8–digit numeral with sign after the address S. The signed 8–digit numeral is output in 32–bit binary code. Minus numbers are expressed in two's complement. This code is kept till the next S code is commanded. Maximum input digits and use of the minus sign is commanded by parameters.

9.2 SPINDLE SPEED BINARY CODE OUTPUT

The commanded spindle speed is output in 16–bit binary code. For constant surface speed control (CSSC), the spindle speed after the CSSC is output.
This code is kept till the next S code is commanded, if it is not under constant surface speed control.

9.3 SPINDLE SPEED ANALOG OUTPUT

When output voltage to the spindle motor is input with sign + 13–bit binary codes, analog voltage corresponding the input is output.
<Maximum output voltage> ± 10V

9.4 CONSTANT SURFACE SPEED CONTROL (G96, G97)

Command G96, and for constant surface speed control.

Format

G96 : Constant surface speed control on
G97 : Constant surface speed control off

When constant surface speed control is on, directly commanding surface speed with the S code, spindle speed in which spindle speed makes surface speed constant to change of tool position (when absolute coordinate value in the work coordinates is regarded as the radius), is output in binary code.
Command surface speed in m/min unit for metric input, and feet/min unit for inch input. Commandable range is as follows:

– Without decimal point
  1 – 999999 m/min or feet/min
– With decimal point
  0.01 – 999999.99 m/min or feet/min

When constant surface speed control is off, spindle speed shall be commanded with an S code.
When constant surface speed control is on, a constant surface speed control on signal is output.
Command which axis to perform constant surface speed control with address P in the G96 block. If P is omitted, P1 (X axis) is regarded to be commanded.

P1 : X axis P2 : Y axis P3 : Z axis P4 : U axis
P5 : V axis P6 : W axis P7 : A axis P8 : B axis
P9 : C axis

9.5 SPINDLE SPEED CLAMP (G92)

Format

G92S _ _ ;

S_ _ : Maximum spindle speed (unit: rpm)

Maximum spindle speed is set with the above command. The maximum spindle speed set is output in 16–bit binary code. It is necessary to clamp to the spindle speed at the PMC side according to the maximum spindle speed output.

9.6 ACTUAL SPINDLE SPEED OUTPUT

Actual spindle speed calculated by the return pulses of the position coder on the spindle is output in 16–bit binary code.

9.7 SPINDLE POSITIONING

The spindle positioning can be done by the spindle motor, without adding an extra servo motor for the C axis.
The spindle position is detected by the position coder attached to the spindle for the per rotation feed and the thread cutting functions. The spindle motor speed command is output from the spindle speed analog output. It is not necessary to add new hardwares to the NC for spindle positioning. Whether to use the spindle motor for spindle positioning (spindle positioning mode) or to use the spindle motor for spindle rotation (spindle rotation mode) is command by special M codes (set by parameters).

1) Move command
   When commanded:
   G00 C_ _; ,
   The spindle is positioned to the commanded position by rapid traverse. Absolute (G90) and incremental (G91) command, as well as decimal point input is possible.

2) Increment system
   Least input increment: 0.001deg.
   Detection unit: $(360 \times N)/4096$ deg.
   N: Combination ratio of position coder and spindle (N=1,2,4)

9.8 SPINDLE SPEED FLUCTUATION DETECTION (G25, G26)

This function monitor spindle speed, detects a higher level of fluctuation than the commanded speed and signals an abnormality, if any, to the machine side, using an alarm, thereby preventing the spindle from seizure, for example. Whether the spindle speed fluctuation detection is done or not is specified by G code.

G25 : Spindle speed fluctuation detection is on.
G26 : Spindle speed fluctuation detection is off.

Format

G26 P_ Q_ R_ D_ ;

P_ : Time from the change of spindle speed to the start of the spindle speed fluctuation detection (Unit: msec)

Q_ : The ratio of spindle speed to the specified spindle speed where spindle speed fluctuation detection starts (Units: %)

R_ : Fluctuation ratio regarded as an alarm (Unit: %)

D_ : Fluctuation ratio which regarded as an alarm (Unit: rpm)

NOTE

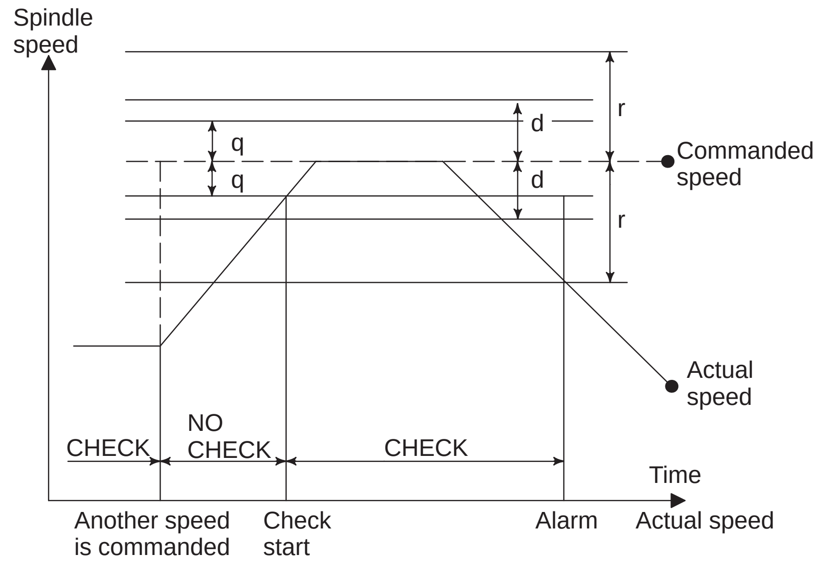
1 The value of P, Q, R and D remains after the power off.
2 The actual spindle speed is calculated by the return pulses generated from the position coder attached to the spindle.

There are two ways in generating an alarm:

An alarm is generated before the specified spindle speed reaches.

An alarm is generated after the specified spindle speed reaches.

1) When an alarm is generated after the spindle speed becomes the commanded speed.



Commanded speed : (Speed commanded by S) × (Spindle override)

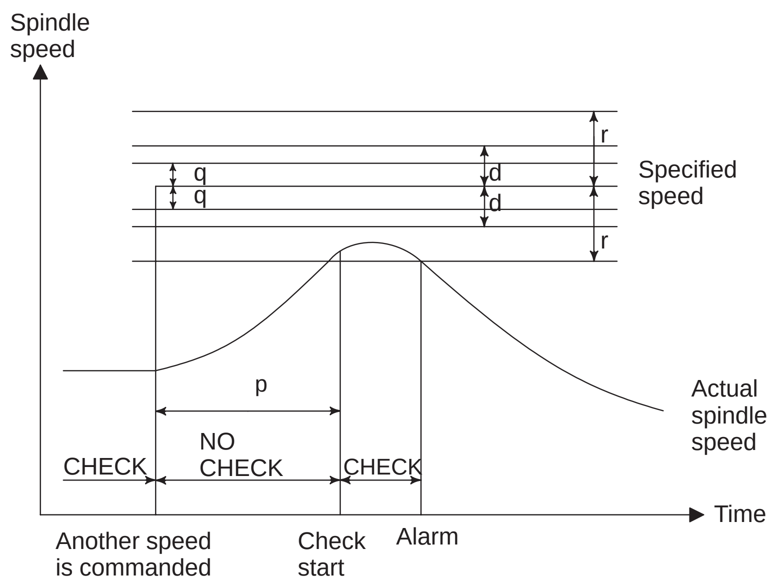
Actual speed : Speed detected by position coder

q : Allowable rate for starting checkup

r : Fluctuation rate in which an alarm is given

d : Fluctuation width in which an alarm is given

2) When an alarm is generated before the spindle speed becomes the commanded speed.



p : Time between changes in commanded speed and check start.

10 TOOL FUNCTIONS

10.1 T CODE OUTPUT

A tool is selected with a tool number commanded by a signed 8–digit number after address T.
The signed 8–digit number is output in 32–bit binary code. The minus numbers are expressed in two's complement. This code is valid till the next T code is commanded. Specify whether or not to use maximum input digits and minus sign with parameters.

10.2 TOOL LIFE MANAGEMENT

Tools are classified into groups, and tool life (hours and times of use) is set for each group. When use of the tool exceeds the preset hours or times of use, another tool in the same group which has not yet exceeded the preset life time is selected. If all the tool in a group exceeds the preset life time, a signal is output to inform the operator that the tools must be changed to new tools. With setting the cutter radius compensation number and the tool length compensation number of the tools, compensation corresponding to each tool can also be done.
With use of this function Factory Automation (FA) comes to a reach.
This function has the following features:

1) Tool life can be set in hours or times of use.
2) New tool select signal output
   This signal is output when a new tool is selected in a group. This can also be used for automatic measurement in compensations of the new tools.
3) Tool change signal
   When all the tools of a group has exceeded their life time, this signal is output to inform the operator.
4) Tool skip signal
   By inputting this signal, tools still not exceeding their life time, can also be changed.
5) Tool life management data is display/modification
   All tool life management data is displayed on the CRT screen, informing the operator of the condition of the tools at a single view. If necessary, the data can be modified via the MDI panel.
   Number of groups and number of tools per group is selected by parameter from the following.

Number of groups	Number of tools
16	16
32	8
64	4
128	2

The following selection is also possible.

Tool life management 512 groups		Tool life management 1024 groups	
Number of groups	Number of tools	Number of groups	Number of tools
64	32	128	32
128	16	256	16
256	8	512	8
512	4	1024	4

NOTE

1 The part program storage length becomes shorter by 6m when this function is provided.
2 In 15–MB, the expanded tool life management of the Series 10/11/12 is included in the tool life management.

11 MISCELLANEOUS FUNCTIONS

11.1 MISCELLANEOUS FUNCTIONS

When a signed 8–digit number after address M is commanded, a 32–bit binary code is output. The minus number is expressed in a two's complement. This code is kept till the next M code is output. Specify whether or not to use maximum input digits and minus sign with parameters.

This function is used for on/off at the machine side. A single M code can be commanded in one block. The following M codes are decoded and output:

M00 : Program stop
M01 : Optional stop
M02 : End of program
M30 : End of program and tape rewind

The above M codes can also be output in binary codes.

M98 (sub program call) and M99(return from sub program) and always processed in the NC so, signal will not be output.

11.2 SECOND MISCELLANEOUS FUNCTIONS

When a signed 8–digit number after address B is commanded, a 32–bit binary code is output. The minus number is expressed in a two's complement. This code is kept till the next B code is commanded. Specify whether or not to use maximum input digits and minus sign with parameters.

This function is used for on/off at the machine side. A single B code can be commanded in one block. By parameter setting, A, C, U, V or W can also be used in place of address B. However, the same address as the control axes cannot be used.

11.3 HIGH–SPEED M/S/T/B INTERFACE

The communication of execution command signal (strobe signal) and completion signal is 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 following describes an example of auxiliary function M code command. The same applies to the T, S, and B (second auxiliary function) functions.

If there is an M command, the NC side inverses the logic level of strobe signal MF. Namely, LOW signal is converted to HIGH signal, while HIGH signal is converted to LOW signal. After the NC side inverses the signal MF, it is considered that the operation of PMC side has been completed if the logic level of completion signal FIN from the PMC is the same as that of the signal MF.

In the conventional system, if the leading edge (from LOW to HIGH) of the completion signal FIN of M/S/T/B is received and then the trailing edge (from HIGH to LOW) of the signal FIN is received, it is considered that the operation has been completed. However, in this system, the operation is considered to have been completed by a single change of signal FIN.

Example) M10;
M20;

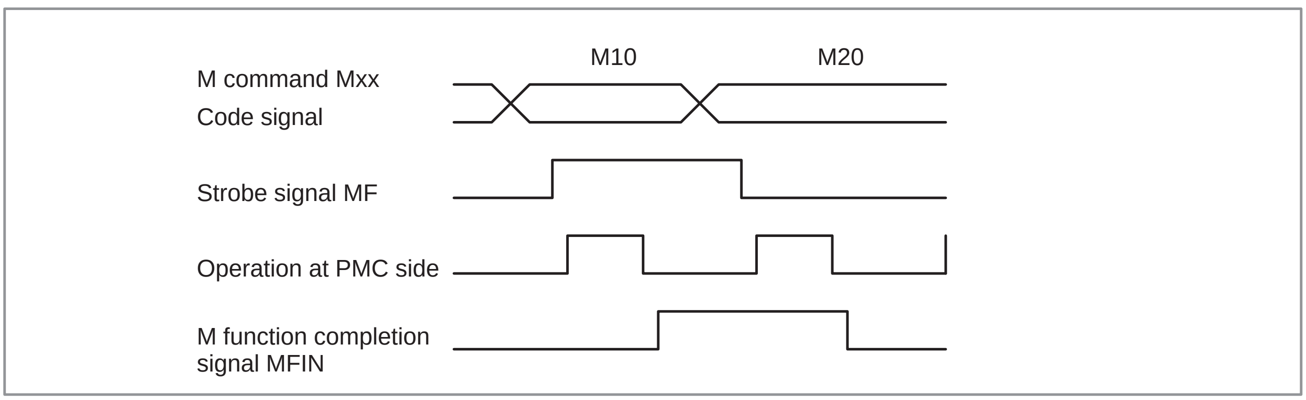


Fig. 11.3 (a) High–speed system time chart

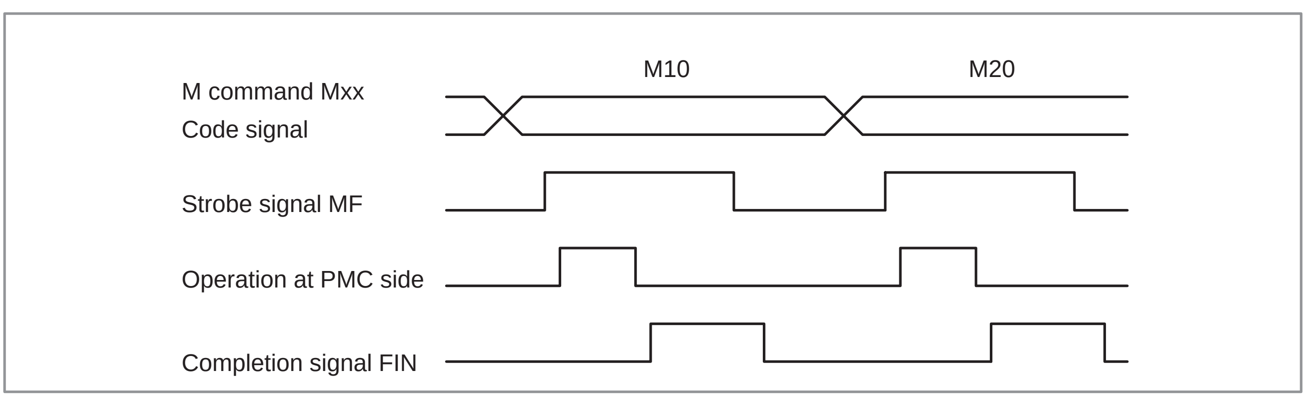


Fig. 11.3 (b) Conventional system time chart

NOTE

1. Either the conventional system or the high-speed system can be selected for communication of strobe signal and completion signal.
2. The high–speed system is valid only for the basic machine interface (BMI). It cannot be used for the FS3 and FS6 interfaces.
3. In the conventional system, only one completion signal is available for all functions of M/S/T/B. However, in the high–speed system, one completion signal is available for each of M/S/T/B functions.

11.4 1–BLOCK PLURAL M COMMAND

Up to five M codes can be simultaneously specified in one block.
As these M codes are simultaneously sent to PMC side, the machining cycle time compared with the conventional 1-block single M command is reduced.

Example)

(i) 1–block single M command

```
M40;
M50;
M60;
G28G91X0Y0Z0;
   :
```

(ii) 1–block plural M command

```
M40M50M60;
G28G91X0Y0Z0;
   :
```

CAUTION

1. The maximum input value of the first M code is ±99999999, while the maximum input values of the M codes from the second to fifth M codes are ±9999.
2. A strobe signal is provided for each of the first to fifth M codes (MF, MF2, MF3, MF4, and MF5).
   When all the operations for the first to fifth M codes are completed, completion signal FIN is output.

12 PROGRAM CONFIGURATION

12.1 PROGRAM NUMBER

A program number is given to each program to distinguish a program from other programs. The program number is given at the head of each program, with a 4-digit number after the address O.

Program number of the program currently under execution is always displayed on the CRT screen. Program search of programs registered in the memory is done with the program number. The program number can be used in various ways.

12.2 PROGRAM NAME

A program name can be given to the program to distinguish the program from other programs when displaying all the registered program on a screen. Register the name between the control-out and the control-in.

Any codes usable in the NC can be used for the program name.

The program name is displayed with the program number in the directory display of registered programs. Note that the program name displayed is within 16 characters.

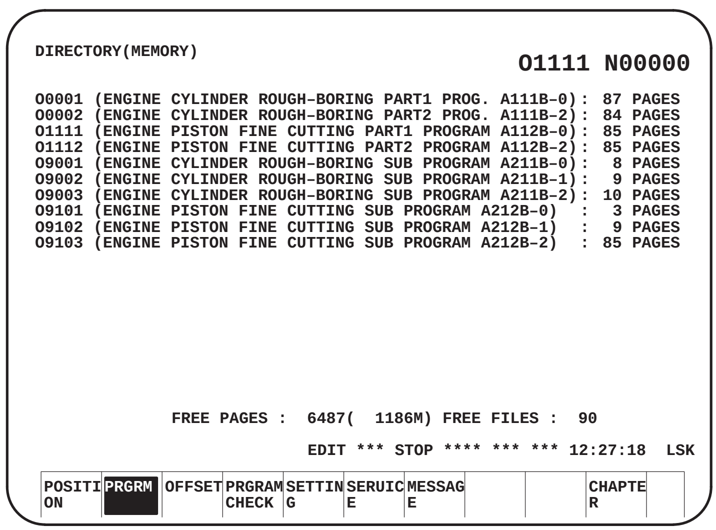
Example) O1234 (PROGRAM FOR ATC);

12.3 PROGRAM NAME (48 CHARACTERS)

A program name can be expanded to 48 characters.

1) Directory display of program names

   Program names can be displayed on the directory screen of program using a maximum of 48 characters.



Example of display for 14–inch CRT

2) Program search by program name from PMC

   A program search can be made by program name through the PMC window.

12.4 MAIN PROGRAM

A program is divided into the main program and the sub program. The NC normally operates according to the main program, but when a command calling a sub program is encountered in the main program, control is passed to the sub program. When a command indicating to return to the main program is encountered in the sub program, control is returned to the main program.

12.5 SUB PROGRAM

When there are sixed sequences or frequently repeated patterns in a program, programming can be simplified by entering these pattern as sub programs to the memory. Sub program is called by M98, and M99 commands return from the sub program. The sub program can be nested 8 folds.

Format

Sub program call

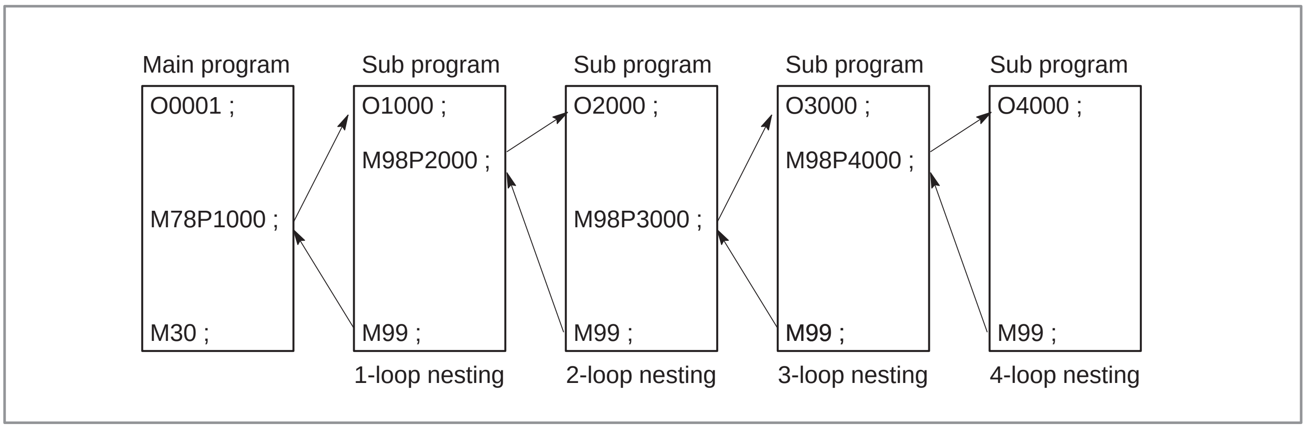
M98 P_ _ L_ _ ;

P_ _ : Program number of sub program to be called

L _ _ : How many times to repeat the sub program

Return from sub program

M99 ;



12.6 SEQUENCE NUMBER

Sequence number can be given in a 5-digit number after the address N at the head of the program block.
The sequence number of the program under execution is always displayed on the CRT screen. The sequence number can also be searched in the program by the sequence number search function.

12.7 TAPE CODES

Either the EIA or the ISO code can be used as tape code. The input program code is distinguished with the first end of block code (EIA: CR, ISO: LF). See the List of Tape Codes in Appendix C for tape codes used.

12.8 BASIC ADDRESSES AND COMMAND VALUE RANGE

Table 12.8 Basic addresses and command value range

Function	Address	Metric input	Inch input
Program number	O	1 to 9999	1 to 9999
Sequence number	N	1 to 99999	1 to 99999
Preparatory functions	G	0 to 99	0 to 99
Coordinates	X, Y, Z, U, V, W, A, B, C, I, J, K, R, Q		
Input unit IS–A		±999999.99 mm or deg	±99999.999 inch or deg
IS–B		±99999.999 mm or deg	±9999.9999 inch or deg
IS–C		±9999.9999 mm or deg	±999.99999 inch or deg
IS–D		±9999.99999 mm or deg	±999.999999 inch or deg
IS–E		±999.999999 mm or deg	±99.9999999 inch or deg
Feedrate per minute	F		
Input unit IS–A		0.0001 to 2400000.0 mm/min	0.0001 to 240000.00 inch/min
IS–B		0.0001 to 240000.00 mm/min	0.0001 to 24000.000 inch/min
IS–C		0.0001 to 24000.000 mm/min	0.0001 to 2400.0000 inch/min
IS–D		0.0001 to 2400.0000 mm/min	0.0001 to 240.00000 inch/min
IS–E		0.0001 to 240.00000 mm/min	0.0001 to 24.000000 inch/min
Feedrate per revolution	F		
Input unit IS–A		0.0001 to 5000.0000 mm/rev	0.00001 to 500.00000 inch/rev
IS–B		0.00001 to 500.00000 mm/rev	0.000001 to 50.000000 inch/rev
IS–C		0.000001 to 50.000000 mm/rev	0.0000001 to 5.0000000 inch/rev
IS–D		0.0000001 to 5.0000000 mm/rev	0.00000001 to 0.50000000 inch/rev
IS–E		0.00000001 to 0.50000000 mm/rev	0.000000001 to 0.050000000 inch/rev
Thread cutting lead	F	Same as per revolution feed	
Tool functions	T	0 to ±99999999	0 to ±99999999
Spindle functions	S	0 to ±99999999	0 to ±99999999
Miscellaneous functions	M	0 to ±99999999	0 to ±99999999
2nd Miscellaneous functions	B, A, C, U, V, W	0 to ±99999999	0 to ±99999999
Offset numbers	D, H	0 to 200	0 to 200
Per second dwell	P, X	0 to 99999.999 sec	0 to 99999.999 sec
Per rotation dwell	P, X	0 to 99999.999 rev	0 to 99999.999 rev
Repeated times	L	0 to 9999	0 to 9999

CAUTION

Coordinates maximum command value for inch input/metric output is limited to: ±39370.078 inch/±3937.0078 inch/ ±393.70078 inch.

NOTE

“:” can be used for O in ISO Code.

12.9 COMMAND FORMAT

The variable block word address format with decimal point is adopted as command format. See List of Command Format in Appendix B for details on command formats.

12.10 LABEL SKIP

Label skip function is valid in the following cases, and “LSK” is displayed on the CRT screen.
1) When power is put on.
2) When the NC is reset.
When label skip function is in valid, all codes to the first encountered end of block (EOB) code are ignored.
The ignored part is called “Reader part”, and section after the first end of block (EOB) code, “significant information”.

12.11 CONTROL–IN/ CONTROL–OUT

Information between the control–in and the control–out are regarded as notes and are ignored.
The reset codes (ISO code: %, EIA code: ER) cannot be used in this part.
The ignored part is called “Notes”.

	ISO code	EIA code
Control–out	(	Channel 2–4–5 on
Control–in	)	Channel 2–4–7 on

12.12 OPTIONAL BLOCK SKIP

When a slash and number (/n) is programmed at the head of a program, and when the machine is operated with the optional block skip switch n on the machine operator’s panel on, information in the block commanded with the /n corresponding to the switch number n is ignored.
If the optional block skip switch n is turned off, information in the /n commanded block will not be ignored. The block with /n commanded can be skipped by the operator’s selection.
1 can be used for n. The 1 to /1 can be omitted.
Example) /1 N12345 G00 X100.Y200.;

12.13 ADDITIONAL OPTIONAL BLOCK SKIP

2 to 9 can also be used for the n of the /n.

13 FUNCTIONS TO SIMPLIFY PROGRAMMING

13.1 CANNED CYCLES (G73, G74, G76, G80–G89, G98, G99)

Canned cycle is a function to simplify commands for machining (boring, drilling, or tapping, etc).
The canned cycle has the positioning plane and the drilling axis. The positioning plane is specified with the plane selection of G17, G18, and G19. The drilling axis is the basic axis X, Y or Z (that does not compose the positioning plane) or its parallel axis.

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 its parallel axis
Yp : Y axis or its parallel axis
Zp : Z axis or its parallel axis

The drilling axis address commanded in the same block as the G codes, G73 - G89, decides whether the drilling axis is the basic axis or its parallel axis. If the drilling axis address was not commanded, the basic axis becomes the drilling axis.
Axis other than the drilling axis becomes the positioning axis.

Example)
When U, V, W axes are set as parallel axes for X, Y, Z axes respectively.

```
G17 G81  . . .  Z_ _ ;   Drilling axis is Z axis.
G17 G81  . . .  W_ _;    Drilling axis is W axis.
G18 G81  . . .  Y_ _ ;   Drilling axis is Y axis.
G18 G81  . . .  V_ _ ;   Drilling axis is V axis.
G19 G81  . . .  X_ _ ;   Drilling axis is X axis.
G19 G81  . . .  U_ _ ;   Drilling axis is U axis.
```

It is not always necessary to command G17, G18, G19 in the same block as G73 - G89.

NOTE
Z axis can always be appointed the drilling axis by parameter setting.

Positioning can be commanded with optional axes other than the drilling axis. The drilling cycle starts after the positioning.
The following explanations are done on the XY plane, and Z axis as the drilling axis.
The following 12 types of canned cycles are available.

Table 13.1 Canned cycles (1/2)

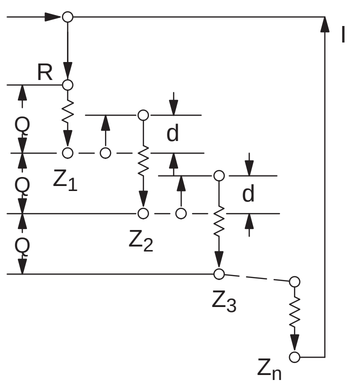
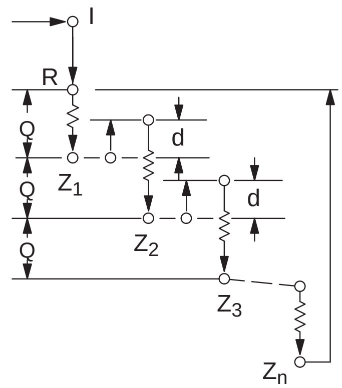
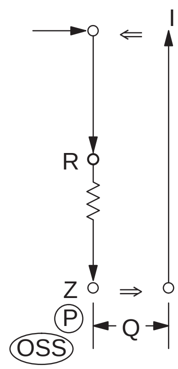
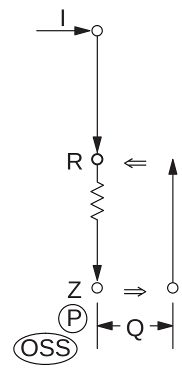
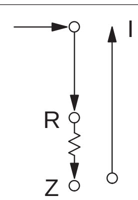
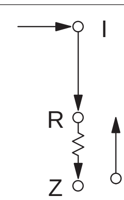
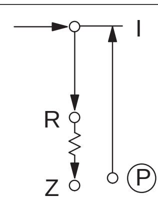
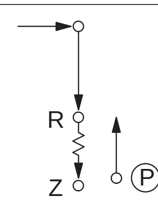
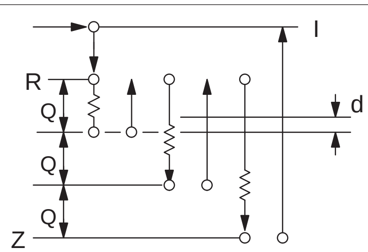
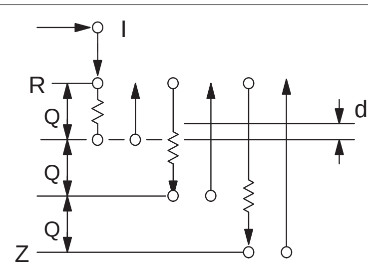
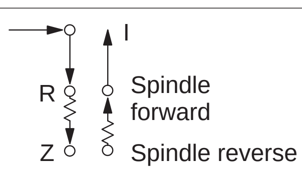
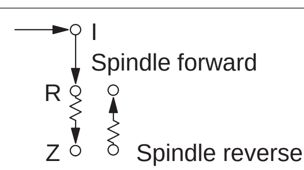
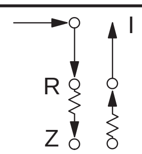
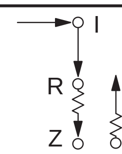
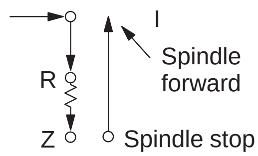
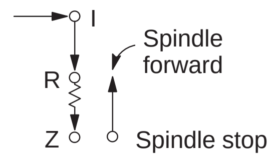
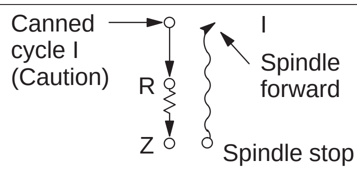
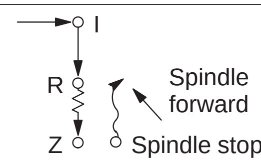
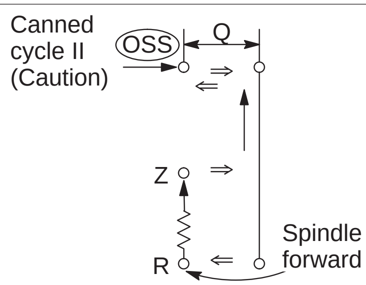
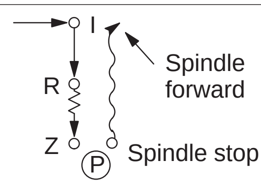
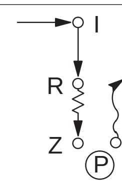
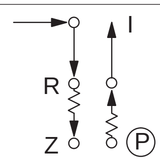
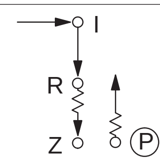
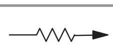
G code	Operation		Function
	G98 mode	G99 mode	
G73			Peck drilling cycle (Note)
G76			File boring cycle (for canned cycle II only)
G81			Drilling cycle (spot drilling)
G82			Drilling cycle (counter boring)
G83			Peck drilling cycle (Note)
G84 G74			Tapping cycle (G74 is CCW tapping cycle)

Table 13.1 Canned cycles (2/2)

G code	Operation		Function
	G98 mode	G99 mode	
G85			Boring cycle
G86			Boring cycle
G87			Boring cycle
		G99 mode cannot be used in canned cycle G87 (Canned cycle II)	Back boring cycle
G89			Boring cycle
G89			Boring cycle

 .... Cutting feed

 .... Rapid traverse

 .... Manual feed

p .... Dwell

Z .... Z point (Hole bottom position)

OSS .... Oriented spindle stop (Spindle stops at constant rotation position)

⇒ .... Shift

I .... Initial point

R .... R point

CAUTION

Set parameter whether to use signals (SRV, SSP) independent of the output signals from the NC (canned cycle I), or to use the M code (canned cycle II) for spindle CCW rotation and spindle stop.

NOTE

"d" of G73 and G83 is set by parameters.

When the drilling axis is Z axis, machining data in the canned cycle is commanded as follows:

G○○ X_ _ Y_ _ Z_ _ R_ _ Q_ _ P_ _ L_ _ F_ _ ;

Drilling mode G○○ . . . See previous table.

Drilling position data

X, Y . . . . . Command position of the hole.

Z . . . . . Specify hole end position shown in the table 13.1.

R . . . . . Specify R point position shown in the table 13.1.

Q . . . . . Specify cutting quantity with G73, G83, and shift quantity with G76, G87.

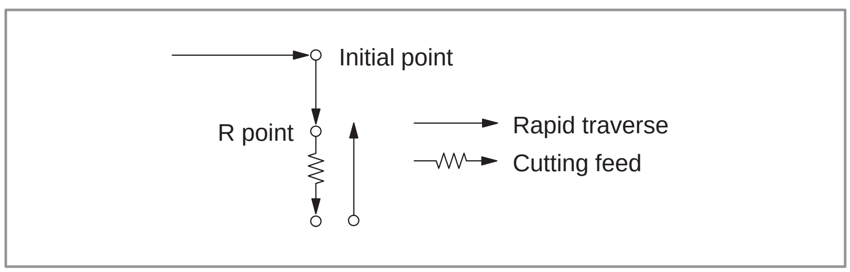
P . . . . . Specify dwell time at the hole bottom.

L . . . . . Specify how may times to repeat. When specified L0, drilling data will be set, but no drilling will be done.

F . . . . . Specify feedrate for cutting.

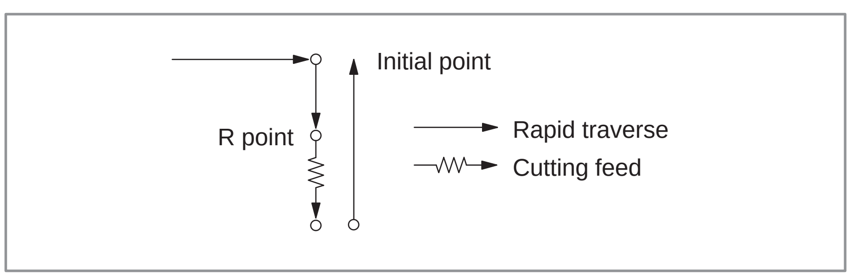
1) R point level return (G99)
By specifying G99, return point in canned cycle is specified to R point. The drilling starts from the end point of the previous block. If the previous block has ended in the initial point, it begins from the initial point and returns to the R point.

Example) When G81 was commanded under G99 mode



2) Initial level return (G98)
By specifying G98, return point in canned cycle is specified to the initial level. The drilling starts from the end point of the previous block. If the previous block has ended in the R point, it begins from the R point and returns to the initial point.

Example) When G81 was commanded under G98 mode



13.2 RIGID TAPPING (G84.2, G84.3)

In tapping, the feed amount of Z 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)
F : Feedrate of Z axis (mm/min)
S : Spindle speed (rpm)

The rotation of spindle and feed of Z axis are independently controlled in the tapping cycle (G84) and left–handed tapping cycle (G74). Therefore, the above conditions may not always be satisfied. Especially at the hole bottom, both the rotation of spindle and feed of Z 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 Z 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 Z 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

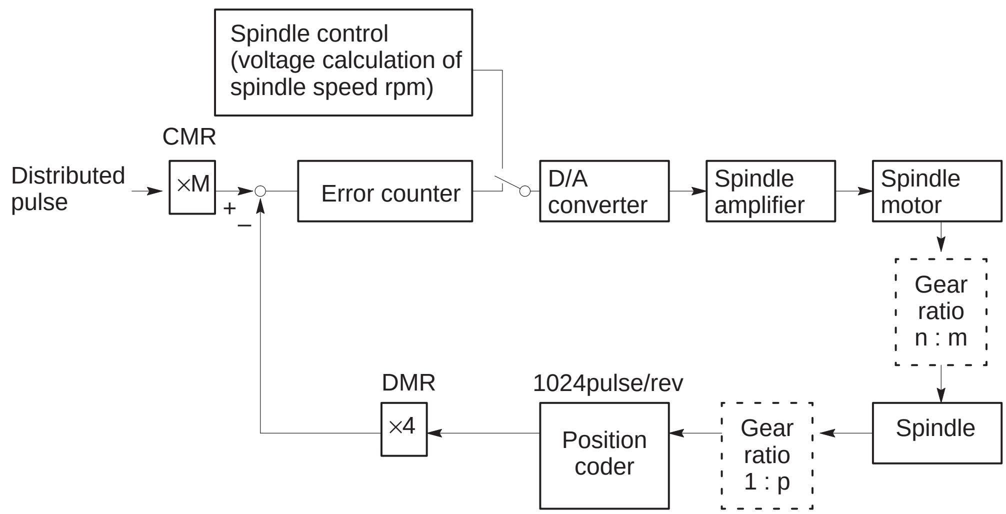
The rigid tapping cycle and rigid left–handled tapping cycle are commanded by G84.2 and G84.3, respectively.

Command format

$\left\{\begin{array}{l}\text{G84.2}\\\text{G84.3}\end{array}\right\}$ X_ _ Y_ _ Z_ _ R_ _ P_ _ F_ _ S_ _ L_ _ ;

- L_ _ : Number of repetitions
- S_ _ : Spindle speed rpm
- F_ _ : Cutting feedrate
- P_ _ : Dwell time on returning to points Z and R (however, when P is valid with parameter selection)
- R_ _ : Coordinate value of point R
- Z_ _ : Coordinate value of point Z
- X_ _ Y_ _ : Drilling position

When the G84.2 or G84.3 is commanded in the feed per revolution mode (G95), the unit of cutting feedrate F_ _ becomes mm/rev or inch/rev. Therefore, the pitch of screw tap can be directly specified.



Gear ratio of spindle to position coder (1 : p)	Least command increment (detection unit) deg
1 : 1	0.088 (1x360/4096)
1 : 2	0.176 (2x360/4096)
1 : 4	0.352 (4x360/4096)

Fig. 13.2 The Control system of spindle during rigid tapping

Even use of the spindle motor incorporating the position coder enables rigid tapping. In this case, the gear ratio of the spindle motor and the spindle is set by the parameter. If, however, there are multiple gears between the spindle motor and the spindle, i.e., if three speed gears (for high, medium, and low speeds) are mounted between them, only one of these gears enables rigid tapping.
In addition, use of the spindle motor incorporating the position coder enables rigid tapping but disables threading and per revolution dwell.

13.3 EXTERNAL OPERATION FUNCTION (G80, G81)

Format

G81 IP_ _ L_ _ ;

where

IP_ _ : Optional combination of axis address X, Y, Z, U, V, W. A, B, C

L_ _ : Times to repeat

With the above program, external operation signal is output after positioning. G80 command cancels the external operation function.

13.4 OPTIONAL ANGLE CORNER ROUNDING

By adding :

, R_ _

to the end of blocks commanding linear or circular interpolation, optional angle corner rounding can be automatically inserted.

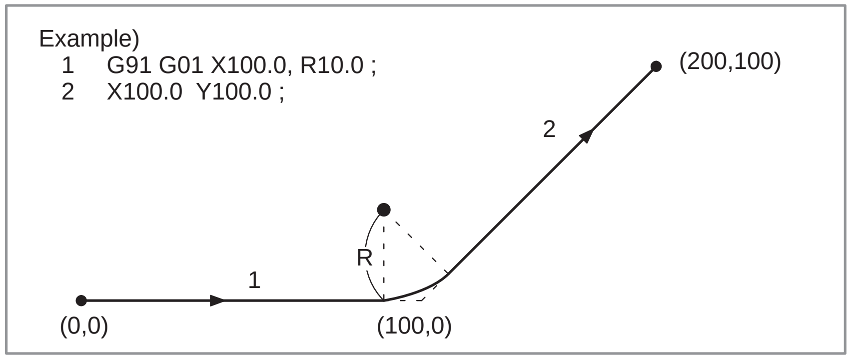


Fig. 13.4

13.5 OPTIONAL ANGLE CHAMFERING

By adding :

, C_ _

to the end of blocks commanding linear or circular interpolation, optional angle chamfering can be automatically inserted.
Specify a numeral following address C, which indicates the distance between the imaginary corner and start or end of chamfering.

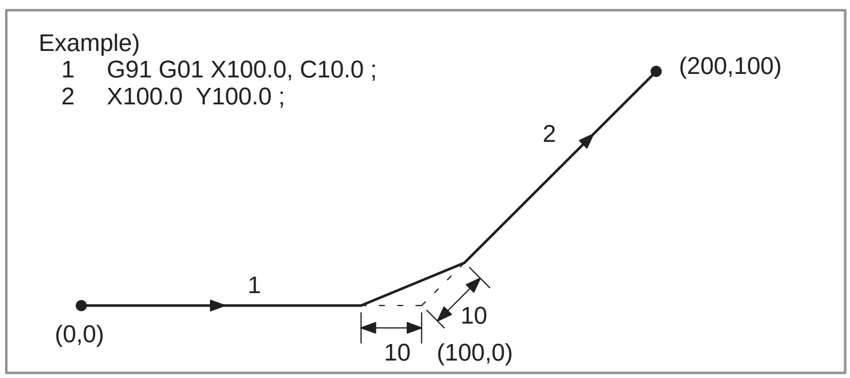


Fig. 13.5

13.6 CIRCULAR INTERPOLATION BY RADIUS PROGRAMMING

Radius value of an arc can be directly designated, instead of using I, J, K; thus simplifying programming.
For arc of 180° or more, designate a minus value to R. A whole circle cannot be commanded.

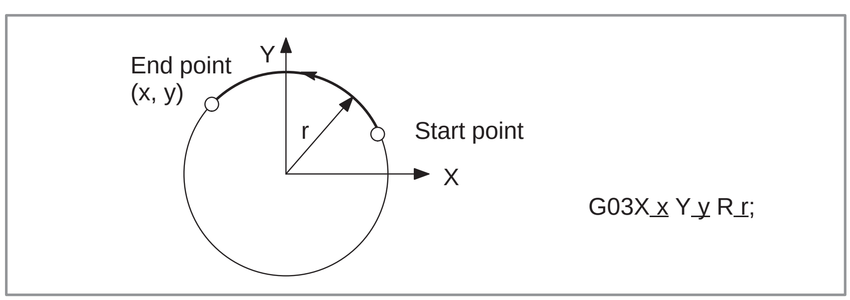
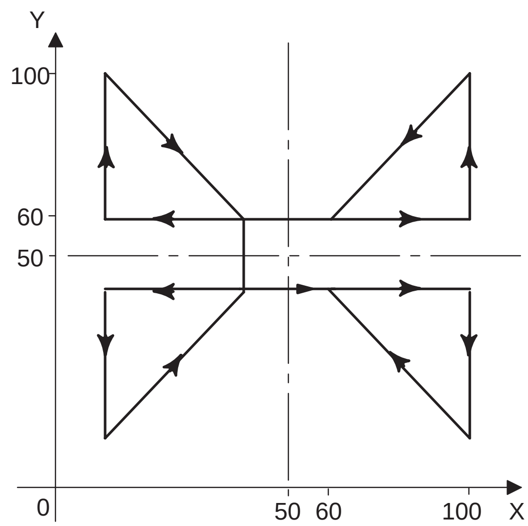


Fig. 13.6

13.7 PROGRAMMABLE MIRROR IMAGE (G50.1, G51.1)

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.

1) Setting of programmable mirror image
   G51.1 X_ _ Y_ _ Z_ _ ;
   is commanded and mirror image is commanded to each axis (as if mirror was set on the axis).
2) Programmable mirror image cancel
   G50.1 X0 Y0 Z0_ _;
   is commanded and the programmable mirror image is canceled.
   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.

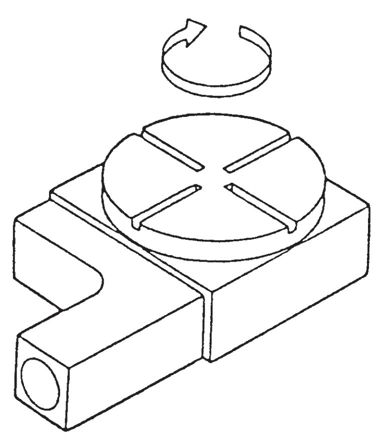



```
Main program
N10 G00 G90 ;
N20 M98 P9000 ;
N30 G51.1 X50.0 ;
N40 M98 P9000 ;
N50 G51.1 Y50.0 ;
N60 M98 P9000 ;
N70 G50.1 X0 ;
   (Cancel only X–axis)
N80 M98 P9000 ;
N90 G50.1 Y0 ;

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

13.8 INDEX TABLE INDEXING

The index table on the machining center is indexed by setting up the axis of indexing (arbitrary 1 axis).

To command for indexing, an indexing angle is only to be specified following a programmed axis (arbitrary 1 axis of X, Y, Z, A, B, C, U, V, W) assigned for indexing. It is not necessary to command the exclusive M code in order to clamp or unclamp the table and therefore programming will become easy.



13.9 FIGURE COPYING (G72.1, G72.2)

The repeat cutting can be made by the rotation or translation of a figure commanded with a sub program.
The plane for figure copying is selected by the plane selection commands of G17, G18, and G19.

1) Rotation Copy
   The repeat cutting can be made by the rotation of a figure commanded with a sub program using the following commands :

Format

G17 G72.1 P_ L_ Xp_ Yp_ R_ ; Xp–Yp plane
G18 G72.1 P_ L_ Zp_ Xp_ R_ ; Zp–Xp plane
G19 G72.1 P_ L_ Yp_ Zp_ R_ ; Yp–Zp plane

P : Sub program number
L : Number of repetitions
Xp : Xp axis center coordinate of rotation
(Xp : X axis or the axis which is parallel to X axis)
Yp : Yp axis center coordinate of rotation
(Yp : Y axis or the axis which is parallel to Y axis)
Zp : Zp axis center coordinate of rotation
(Zp : Z axis or the axis which is parallel to Z axis)
R : Rotation angle (+ = Counterclockwise direction)

2) Translation Copy
   The repeat cutting can be made by the translation of a figure commanded with a sub program using the following commands :
   Select the plane of translation copy with the plane selection commands G17, G18, and G19.

Format

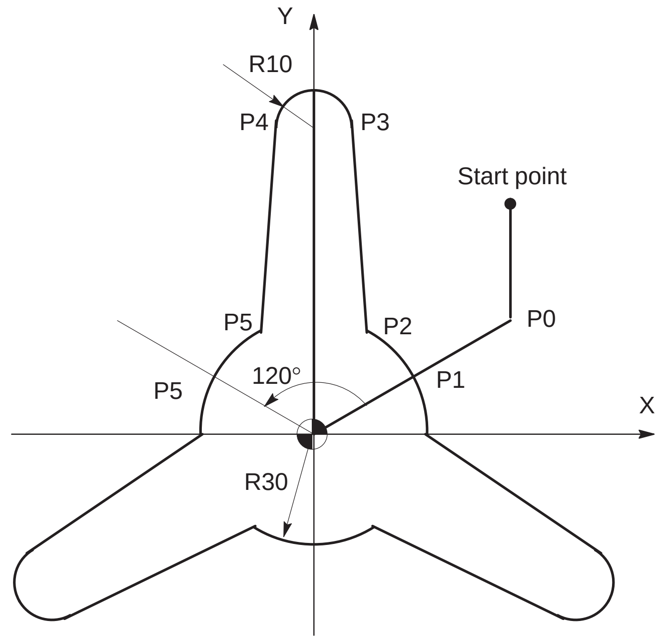
G17 G72.2 P_ L_ I_ J_ ; Xp–Yp plane
G18 G72.2 P_ L_ K_ I_ ; Zp–Xp plane
G19 G72.2 P_ L_ J_ K_ ; Yp–Zp plane

P : Sub program number
L : Number of repetitions
I : Shift amount in Xp direction
J : Shift amount in Yp direction
K : Shift amount in Zp direction

The rotation copy cannot be commanded in the subprogram which commanded a rotation copy. Similarly, the translation copy cannot be further commanded in a subprogram which commanded a translation copy.

However, the translation copy and rotation copy can be commanded in the subprograms which commanded the rotation copy and translation copy, respectively.

(Program example of rotation copy)



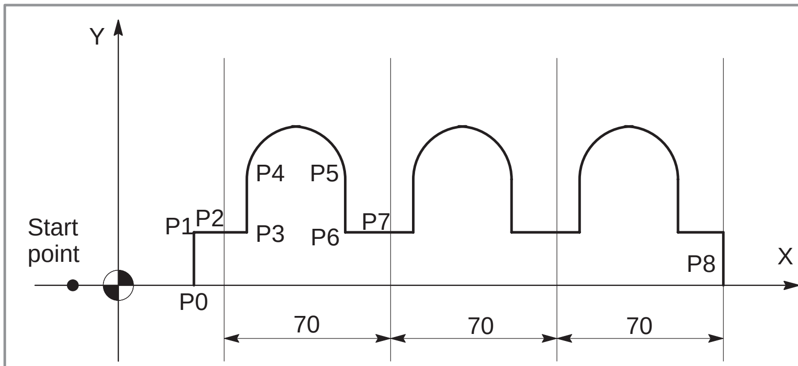
Main program

```
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_ ;              (P0)
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 ;
```

(Program example of translation copy)



Main program

```
O100 ;
N10 G92 X–20.0 Y0 ;
N20 G00 G90 X0 Y0;
N30 G01 G17 G41 X_ Y_ D01 F10 ;   (P0)
N40 Y_ ;                          (P1)
N50 X_ ;                          (P2)
N60 G72.2 P2000 L3 170.0 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 ;
```

13.10 CIRCLE CUTTING FUNCTION

During circle cutting, the tool moves from the center of a circle and cuts a workpiece along the circle as shown in Fig. 13.10. The tool first moves in a 45° direction, and then moves along an arc of a circle having half the radius of the target circle. The tool then comes into contact with the workpiece and starts cutting. The tool can cut the workpiece without leaving any marks on it.
A single block of G code can specify the series of movements described above. During circle cutting, cutter compensation can be performed.

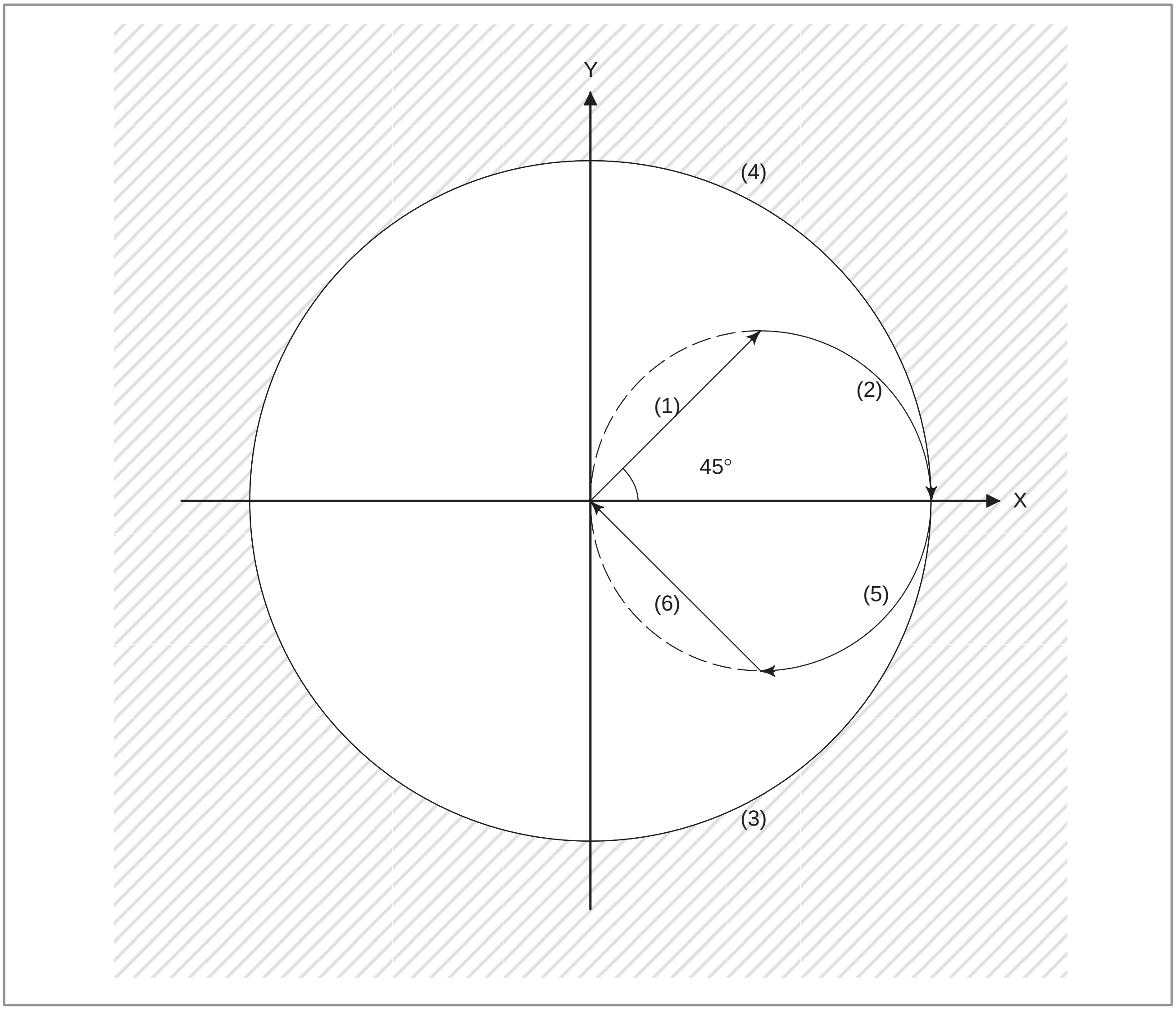


Fig. 13.10

14 COMPENSATION FUNCTIONS

14.1 TOOL LENGTH COMPENSATION (G43, G44, G49)

By setting the difference between tool length assumed when programming and the actual tool length as offsets, workpieces can be machined according to the size commanded by the program, without changing the program.

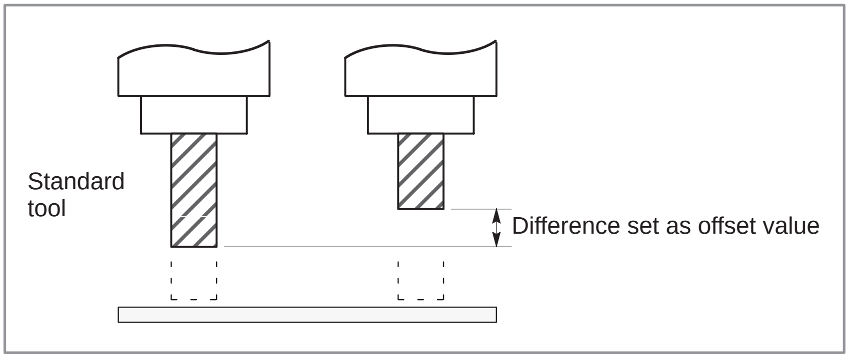


Fig. 14.1

1) Tool length compensation and its cancellation (G43, G44, G49)
   G43 : Tool length compensation +
   G44 : Tool length compensation –
   G49 : Tool length compensation cancel
   In G43 mode, the tool is offset to the + direction for the preset tool length offset amount. In G44 mode, it is offset to the - direction for the preset tool length offset amount. G49 cancels tool length compensation.
2) Tool length compensation axis
   Whether to perform tool length compensation always on the Z axis or on axis commanded in the G43, G44 block is selected by parameters. Movement command of only a single axis can be done when commanding tool length compensation axis in the G43, G44 block. An alarm arises if multiple axes are commanded. When movement command is omitted, tool length compensation is done on the Z axis. Tool length compensation can be performed on another axis (during tool length compensation on an axis). G49 cancels tool length compensation on all axes.

Format

$$\left\{\begin{matrix}\textbf{G43}\\ \textbf{G44}\end{matrix}\right\}\ \alpha\_\ \_\ \ \textbf{H}\_\ \_\ ;$$

where
α : One of X, Y, Z, U, V, W, A, B, C (optional axis address)

3) Assignment of offset amount (H code)
   The offset amount can be set in the tool length compensation memory. By setting a 3–digit number after address H as offset number, offset amount loaded in corresponding tool length compensation memory is used as tool length compensation amount.

14.2 TOOL OFFSET (G45, G46, G47, G48)

The programmed tool movement can be expanded or reduced for offset amount preset in the tool length compensation memory, by using this function.

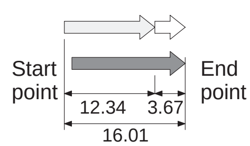
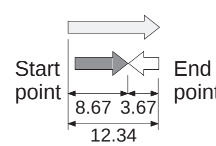
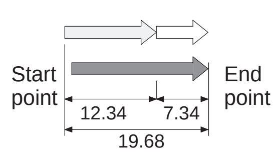
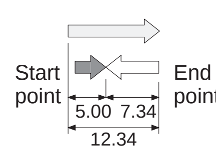
1) G45, G46, G47, G48
   G45: Tool offset expansion
   G46: Tool offset reduction
   G47: Tool offset double expansion
   G48: Tool offset double reduction

By commanding G45 – G48, expansion, reduction, double expansion, double reduction to axis move commanded in the program can be performed for the offset amount preset in the tool length compensation memory. The same offset amount is applied to all move command axes in the same block as G45 – G48.

2) Assignment of offset amount (D code)
   The offset amount can be set in the tool length compensation memory. By commanding an offset number with a 3–digit number after address D, offset amount corresponding to the number in the tool length compensation memory is used as tool offset amount.

> **CAUTION**
> It is also possible to assign the offset amount in H code, for common use with other NCs.

Table 14.2

G45	G46
Increase by set value	Decrease by set value
	
G47	**G48**
Double increase of set value	Double decrease of set value
	

Move command +12.34, offset value +3.67

14.3 CUTTER COMPENSATION

14.3.1 Cutter Compensation B (G40 – 42)

With cutter compensation B, inside of the sharp angle cannot be cut. If commanded, an alarm arises. In this case, an arc larger that the cutter radius can be commanded to the corner by programming. Other functions are same as cutter radius compensation C.

14.3.2 Cutter Compensation C (G40 – G42)

With this function, the programmed tool path can be offset when actually machining, for value of the tool radius set in the NC.
By measuring cutting radius for actual cutting, and setting the value in the NC as offset value, the tool can machine the programmed pattern, via the offset path. There is no need to change the program even when tool radius changes; just change the offset value.

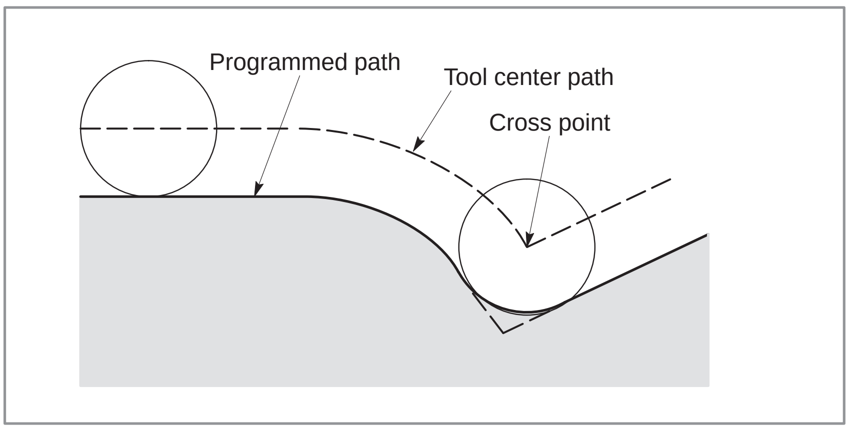


Fig. 14.3.2 (a)

Cross points of line and line, arc and arc, line and arc is automatically calculated in the NC to obtain offset actual tool path. So, programming becomes simple, because it is only necessary to program the machining pattern.

1) Cutter compensation and its cancellation (G40, G41, G42)
   G40 : Cutter radius compensation cancel
   G41 : Cutter radius compensation left
   G42 : Cutter radius compensation right
   G41 and G42 are commands for cutter radius compensation mode. The cutter is offset to the left forward in the cutter movement in G42 and right forward in G42. Cutter radius compensation is cancelled with G40.
2) Assignment of offset amount (D code)
   The offset amount can be set in the cutter radius compensation memory. When a 3–digit number after address D is commanded as offset number, corresponding offset amount in the tool compensation memory is applied as the offset amount for cutter radius compensation.

3) Plane selection (G17, G18, G19)

Cutter radius compensation is done on XY, ZX, YZ planes and on parallel axes of X, Y, Z axes.

Plane to perform cutter radius compensation is selected with G17, G18, G19.

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

Parameters are used to set which parallel axis of the X, Y, Z axes is to be the additional axis.

Plane to perform cutter radius compensation is decided in the axis address commanded in the G17, G18, or G19 block.

Example)
(U, V, W axes are parallel axes of X, Y, Z axes respectively)
G17 X_ ; XY plane
G17 U_ V_ ; UV plane
G19 Y_ W_ ; YW plane

If axis address of Xp, Yp, or Zp was omitted, compensation plane is decided regarding that X, Y, or Z was omitted.

4) Interference check

Tool overcutting is called ‘interference’. This function checks whether interference occurs, if cutter radius compensation is performed.

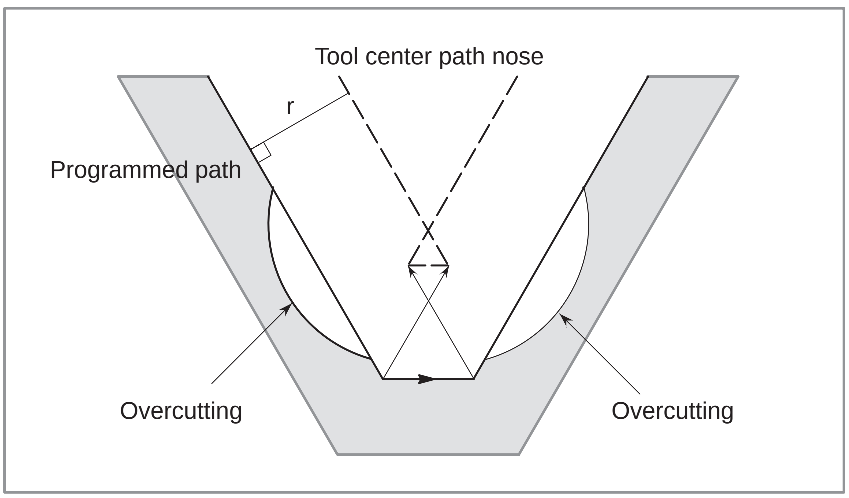


Fig. 14.3.2 (b)

14.4 3–DIMENSIONAL TOOL COMPENSATION (G40, G41)

With this function, tool can be offset to the 3–dimensional direction for the offset amount set in the tool compensation memory, when machining a 3–dimensional sculptured surface.

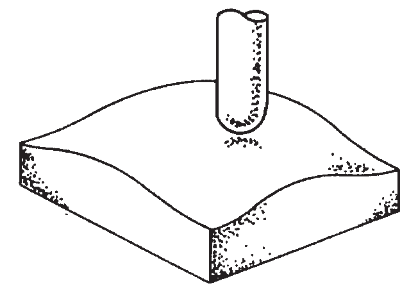


Fig. 14.4

1) 3–dimensional tool compensation and its cancellation (G40, G41)
   G40: 3–dimensional tool compensation cancel
   G41: 3–dimensional tool compensation
   G41 is for commanding 3–dimensional tool compensation mode, and G40 for its cancellation.
   Whether the tool compensation is 3–dimensional or not is distinguished in the block which G41 was commanded (I, J, K must all be commanded in the block). If all I, J, K are commanded in the G41 block, it is regarded as 3–dimensional tool compensation.

Format

G41 Xp_ _ Yp_ _ Zp_ _ I_ _ J_ _ K_ _ D_ _ ;

where

I_ _ J_ _ K_ _ :	Specifies offset direction
D_ _ :	Specifies offset amount

2) 3–dimensional tool compensation space
   3–dimensional tool compensation is not only possible in the XYZ space, but also in additional axes parallel to X, Y, Z axes.
   Space to perform 3–dimensional tool compensation is decided by the axis address commanded in the G41 block.
   Example) U, V, X, axes are parallel to X, Y, Z axes respectively.

G41 X_ _ I_ _ J_ _ K_ _ ;	XYZ space
G41 U_ _ V_ _ W_ _ I_ _ J_ _ K_ _ ;	UVW space
G41 W_ _ I_ _ J_ _ K_ _ ;	XYW space

   When axis address in Xp, Yp, or Zp axis was omitted, the compensation space is decided regarding that the X, Y, or Z was omitted.

3) Assignment of offset amount (D code)
   The offset amount can be set in the tool compensation memory.
   When a 3–digit number after address D is commanded as offset number, corresponding offset amount in the tool compensation memory is applied as the 3–dimensional tool compensation amount.

4) 3–dimensional tool compensation vector

Under the 3–dimensional tool compensation mode, a compensation vector is produced to the direction command by I, J, K at the end of each block to offset the tool by the compensation vector.

The compensation vector can be obtained by two methods, type A and type B, which can be selected by parameters.

Compensation vector (type A)	Compensation vector (type B)
$VXp = r \times i/p$	$VXp = r \times i/p$
$VYp = r \times j/p$	$VYp = r \times j/p$
$VZp = r \times k/p$	$VZp = r \times (1–k/p)$

where

- VXp : Compensation vector factor of X axis or its parallel axis direction
- VYp : Compensation vector factor of Y axis or its parallel axis direction
- VZp : Compensation vector factor of Z axis or its parallel axis direction
- r : Tool compensation amount selected by D code
- i, j, k : Numeral commanded by I, J, K
- p : Select by parameter whether to take SQRT ($i^2+j^2+k^2$) or constant preset by parameter.

If all I, J, K are not commanded, the same compensation vector as the precious block is produced at the end of the block.

14.5 TOOL OFFSET BY TOOL NUMBER

Cutter compensation value, tool length compensation value and tool pot number can be set corresponding to the tool number (T code). When a tool number is specified, a pot number corresponding to the tool number is output to the PMC as a T code. When a cutter compensation or a tool length compensation is specified, compensation is effected with the cutter compensation value or tool length compensation value being set corresponding to the tool number.

1) Setting tool data
   The tool data can be set by MDI or by program.
   Format of program

   a) When registering the tool data after clearing the tool data currently registered.

```
G10 L70 ;
T_ _  P_ _  R_ _  K_ _ ;
T_ _  P_ _  R_ _  K_ _ ;
_ _ _ _ _
T_ _  P_ _  R_ _  K_ _ ;
G11 ;
```

   G10 L70: Start of registration after clearing the tool data registered up to now
   T : Tool number (0 – 99999999)
   P : Pot number (0 – 9999)
   R : Cutter compensation value
   K : Tool length compensation value
   G11 : End of registration

   b) When additionally registering the tool data after the tool data currently registered.

```
G10 L71 ;
T_ _  P_ _  R_ _  K_ _ ;
T_ _  P_ _  R_ _  K_ _ ;
_ _ _ _ _
T_ _  P_ _  R_ _  K_ _ ;
G11 ;
```

2) Displaying tool data
The tool data can be displayed on the CRT.

```
TOOL DATA                        O0000 N00000
                                 E:001 S:002
NO.  T_CODE    POT#    LENGTH    RADIUS
001    1111    0010    10.000    20.000
002    3333    0020    11.000    21.000
003    5468    0021    12.000    22.000
004    5555    0033    13.000    23.000
005    3654    0051    14.000    24.000
006    2541    0014    15.000    25.000
007    6541    0024    16.000    26.000
008    1403    0015    17.000    27.000
009    7171    0061    18.000    28.000
010    6565    0034    19.000    29.000
                                   (MM)

MDI  *** STOP **** *** *** 10:19:28  LSK
POSITION PROGRAM OFFSET   PRG_CHK CHAPTER+
```

Fig. 14.5

3) Punching out tool data
The tool data being set can be punched out.

4) Outputting pot number
When a tool number is specified, the pot number corresponding to the tool number is output to the PMC as a T code.

5) Compensation value
When an M code for tool change (parameter setting) is specified, the offset value corresponding to the tool number being specified so far becomes effective.

NOTE
1 Part program length shortens by 14m.
2 Tool length cannot be measured by tool length/work zero point measured function B.

14.6 TOOL COMPENSATION MEMORY

There are three tool compensation memories, A, B, and C. One of the memories is selected according to the nature of the compensation.

The tool compensation amount can be set in the following range.

The valid range of tool compensation amount can be selected using ORG and OFN of parameter 6002, OUF of parameter 6004, and ONM of parameter 6007.

ONM	OUF	OFN	ORG	Geometric compensation	Geometric compensation	Wear compensation	Wear compensation
				input in mm	input in mm	input in inches	input in inches
0	0	0	1	±999.99 mm (±9999.99)	±99.999 inch (±99.999)	±99.99 mm (±999.99)	±9.999 inch (±99.999)
0	0	0	0	±999.999 mm (±9999.999)	±99.9999 inch (±999.9999)	±99.999 mm (±999.999)	±9.9999 inch (±99.9999)
0	0	1	0	±999.9999 mm (±999.9999)	±99.99999 inch (±999.99999)	±99.9999 mm (±999.9999)	±9.99999 inch (±99.99999)
0	1	0	0	±99.99999 mm (±99.99999)	±9.999999 inch (±999.999999)	±9.99999 mm (±999.99999)	±0.999999 inch (±99.999999)
1	0	0	0	±9.999999 mm (±999.999999)	±0.9999999 inch (±99.9999999)	±0.999999 mm (±999.999999)	±0.0999999 inch (±9.9999999)

(The value enclosed in parentheses are used when the extended option for the tool compensation amount is added.)

14.6.1 Tool Compensation Memory A

There is no difference between geometry compensation memory and tool wear compensation memory in this tool compensation memory A. Therefore, amount of geometry offset and tool wear offset together is set as the offset memory. There is also no differences between cutter radius compensation (D code) and tool length compensation (H code).

Table 14.6.1 Example of setting

Offset number	Compensation (geometry+wear)	D code/H code common
001	10.1	For D code
002	20.2	For D code
003	100.1	For H code

14.6.2 Tool Compensation Memory B

Memory for geometry compensation and tool wear compensation is prepared separately in tool compensation memory B. Geometry compensation and tool wear compensation can thus be set separately. There is no difference between cutter radius compensation (D code) and tool length compensation (H code).

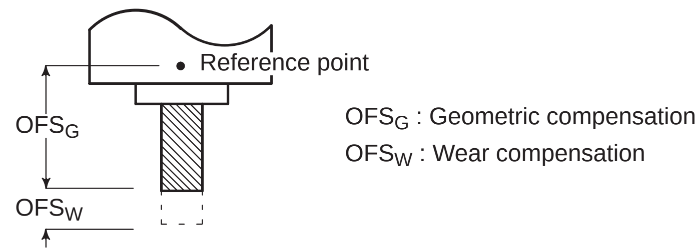


Fig. 14.6.2

Table14.6.2 Example of setting

Offset number	Geometry compensation	Wear compensation	D code/H code common
001	10.0	0.1	For D code
002	20.0	0.2	For D code
003	100.0	0.1	For H code

14.6.3 Tool Compensation Memory C

Memory for geometry compensation as well as tool wear compensation is prepared separately in tool compensation memory C. Geometry compensation and tool wear compensation can thus be set separately. Separate memories are prepared for cutter radius compensation (for D code) and for tool length compensation (for H code).

Table 14.6.3 Example of setting

Offset number	For D code		For H code	
	Geometry compensation	Wear compensation	Geometry compensation	Wear compensation
001	10.0	0.1	100.0	0.1
002	20.0	0.2	300.0	0.3

14.7 NUMBER OF TOOL OFFSETS

1) 32 tool offsets
   Offset numbers (D code/H code) 0 - 32 can be used.
   D00 - D32, or H00 - H32
2) 99 tool offsets
   Offset numbers (D code/H code) 0 - 99 can be used.
   D00 - D99, or H00 - H99
3) 200 tool offsets
   Offset numbers (D code/H code) 0 - 200 can be used.
   D00 - D200, or H00 - H200
4) 499 tool offsets
   Offset numbers (D code/H code) 0 - 499 can be used.
   D00 - D499, or H00 - H499
5) 999 tool offsets
   Offset numbers (D code/H code) 0 - 999 can be used.
   D00 - D999 or H00 - H999

14.8 CHANGING OF TOOL OFFSET AMOUNT (PROGRAMMABLE DATA INPUT) (G10)

Tool offset amount can be set/changed with the G10 command.
When G10 is commanded in absolute input (G90), the commanded offset amount becomes the new tool offset amount. When G10 is commanded in incremental input (G91), the current tool offset amount plus the commanded offset amount is the new tool offset amount.

Format

Tool compensation memory A

G10 L11 P_ R_ ;

where
P_ : Offset number
R_ : Tool offset amount

Tool compensation memory B

Setting/changing of geometry offset amount

G10 L10 P_ R_ ;

Setting/changing of tool wear offset amount

G10 L11 P_ R_ ;

Tool compensation memory C

Setting/changing of geometry offset amount for H code

G10 L10 P_ R_ ;

Setting/changing of geometry offset amount for D code

G10 L12 P_ R_ ;

Setting/changing of tool wear offset amount for H code

G10 L11 P_ R_ ;

Setting/changing of tool ware offset amount for D code

G10 L13 P_ R_ ;

NOTE
L1 may be used instead of L11 for the compatibility with the conventional NC's format.

14.9 ROTARY TABLE DYNAMIC FIXTURE OFFSET

The workpiece coordinate system is set after the position of a workpiece placed on a rotary table is measured. In a conventional system, however, if the rotary table rotates before cutting is started, the position of the workpiece must be measured again and the workpiece coordinate system must be reset accordingly.

The rotary table dynamic fixture offset function saves the operator the trouble of re–setting the workpiece coordinate system when the rotary table rotates before cutting is started. With this function the operator simply sets the position of a workpiece placed at a certain position on the rotary table as a reference fixture offset. If the rotary table rotates, the system automatically obtains a current fixture offset from the angular displacement of the rotary table and creates a suitable workpiece coordinate system. After the reference fixture offset is set, the workpiece coordinate system is prepared dynamically, wherever the rotary table is located.

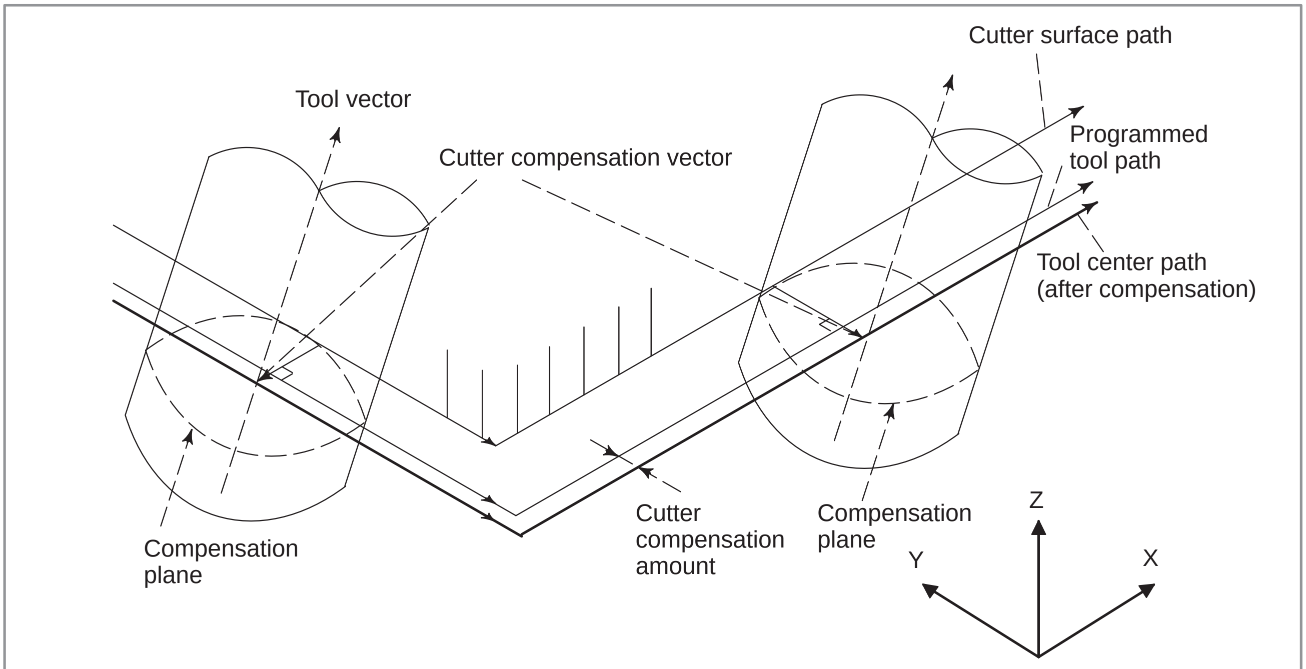
The zero point of the workpiece coordinate system is obtained by adding the fixture offset to the offset from the workpiece reference position.

14.10 THREE–DIMENSIONAL CUTTER COMPENSATION

The three–dimensional cutter compensation function is used with machines that can control the direction of tool axis movement by using rotation axes (such as the B– and C–axes). This function performs cutter compensation by calculating a tool vector from the positions of the rotation axes, then calculating a compensation vector in a plane (compensation plane) that is perpendicular to the tool vector. This function is also applicable to machines with inclined rotary heads.

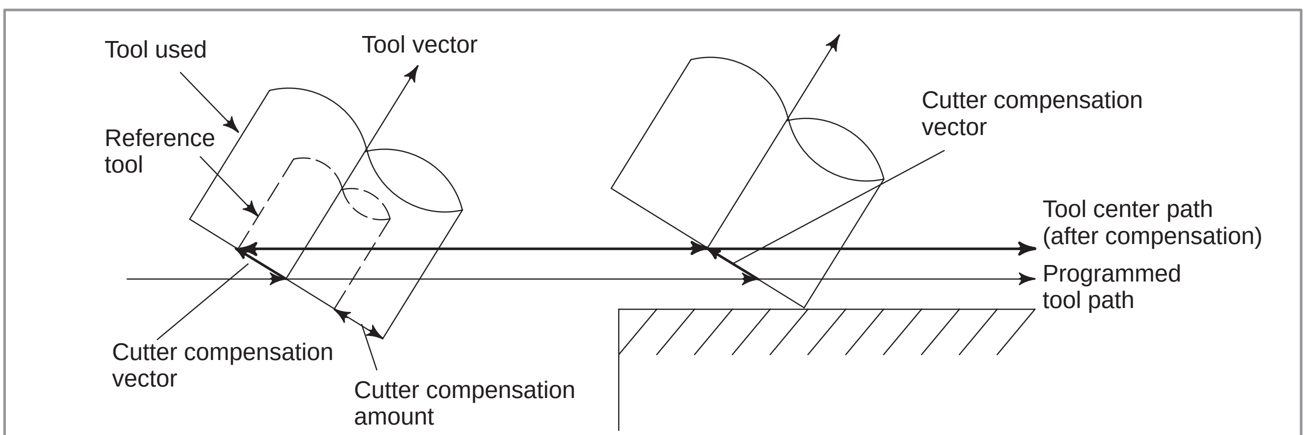
There are two types of cutter compensation: Tool side compensation and leading edge compensation. Which is used depends on the type of machining.

Example) Tool side compensation



Leading edge compensation is performed when a workpiece is machined by the edge of the tool. In leading edge compensation, the tool is shifted automatically by the distance of the tool radius along the line where the plane formed by the tool vector and the movement direction and a plane perpendicular to the tool axis direction intersect.

Example) Leading edge compensation

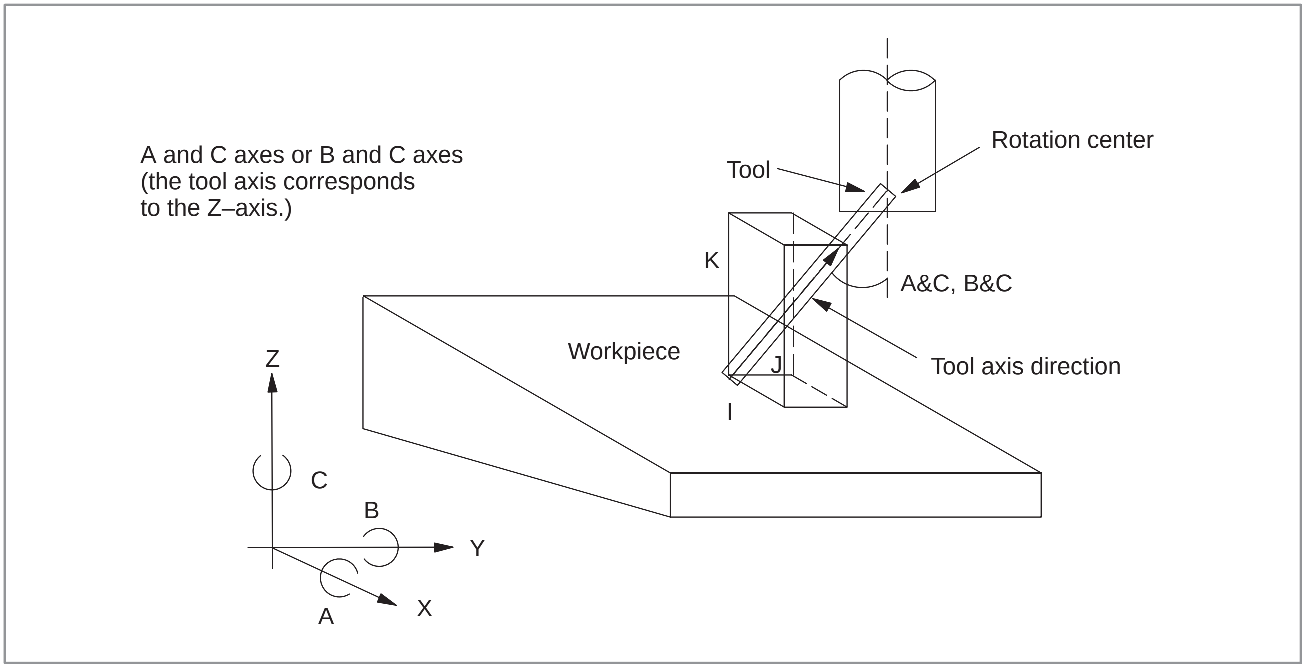


14.11 DESIGNATION DIRECTION TOOL LENGTH COMPENSATION

In a five–axis machine tool having three basic axes and two rotation axes for turning the tool, tool length compensation can be applied in the direction of the tool axis.
The tool axis direction is specified with I, J, and K; a move command for the rotation axes is not specified directly. When I, J, and K are specified in designation direction tool length compensation mode, the following opetation is performed automatically:

1. The basic three axes operate so that tool length compensation is applied using the offset specified by the D code in the direction specified by I, J, and K. (Compensation is applied in the same way as for the three–dimensional tool compensation function).
2. The two rotation axes operate so that the tool axis is oriented in the direction specified by I, J, and K. (This specifications manual explains this operation.)

<u>Machine configuration example</u>



15 ACCURACY COMPENSATION FUNCTION

15.1 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.

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 (for each axis).

Total offset points are:

Total offset points = 128 × controlled axes.

Optional distribution to each axis can be done by parameter setting. At each position:

Compensation pulse = (-7 to +7) × (magnification)

Where

Compensation pulse unit : same as detection unit

Magnification : 0 – 100 times, set by parameter (for each axis)

15.2 INTERPOLATION TYPE PITCH ERROR COMPENSATION

The stored pitch error compensation function output all the compensation pulses at each compensation point. The amount of output compensation pulses at each point covers the interval specified with a parameter. The interpolation type pitch error compensation function, however, outputs the compensation pulses evenly spaced between compensation points.

15.3 THE SECOND CYLINDRICAL PITCH ERROR COMPENSATION METHOD

When the rotary table is rotated using gears, pitch error can occur at two different intervals: one coinciding with the rotation of the rotary table and the other coinciding with the rotation of the gears that are rotating the table. To correct the pitch error occurring in this type of device, compensation is performed for both the table and the gears. This compensation method is know as the second cylindrical pitch error compensation.

When a single gear is mounted between the rotary table and the servo motor, as in Fig. 15.3 (a), pitch error compensation for wheel A is performed in the conventional manner, and compensation for wheel B is performed using the second cylindrical pitch error compensation method.

When multiple gears are mounted, as in Fig. 15.3 (b), compensation for pitch error caused by wheel A is performed in the conventional manner, and the second cylindrical pitch error compensation method is used to correct cyclic pitch errors occurring during the wheel A compensation interval.

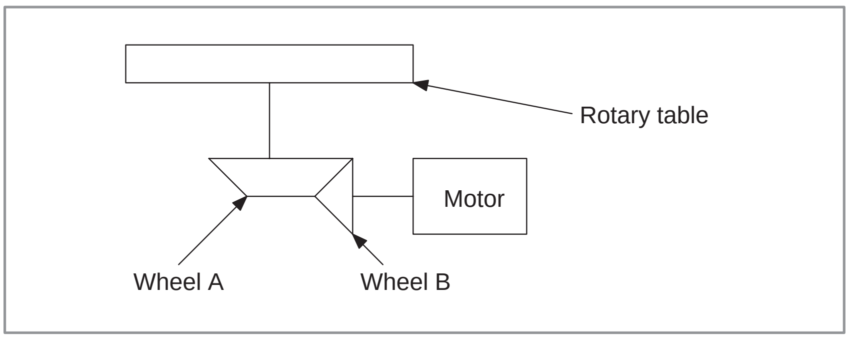


Fig. 15.3 (a) Application of second cylindrical pitch error compensation when a single gear in mounted

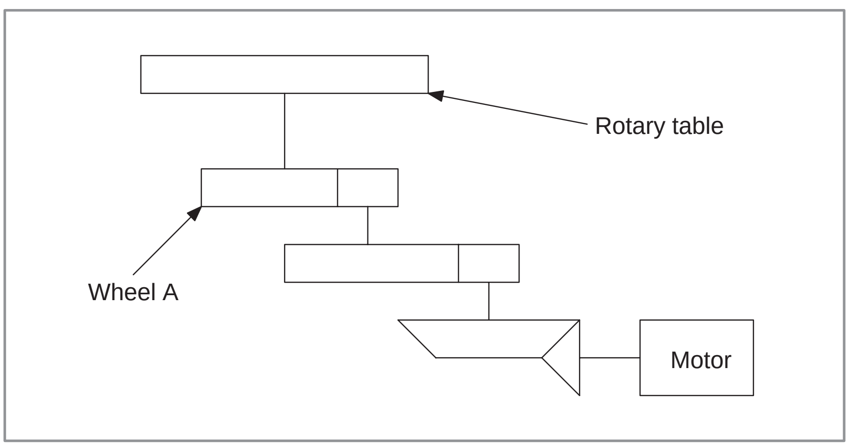


Fig. 15.3 (b) Application of second cylindrical pitch error compensation when multiple gears are mounted

Though the text in this manual cites the case where a rotary table is used, the second cylindrical pitch error compensation method can also be applied to error compensation for a gear–operated linear axis.
Fig. 15.3 (c) shows an example of such a linear axis. In this example, error compensation for the ball screw is performed in the conventional manner, and compensation for wheel A is performed using the second cylindrical pitch error compensation method.

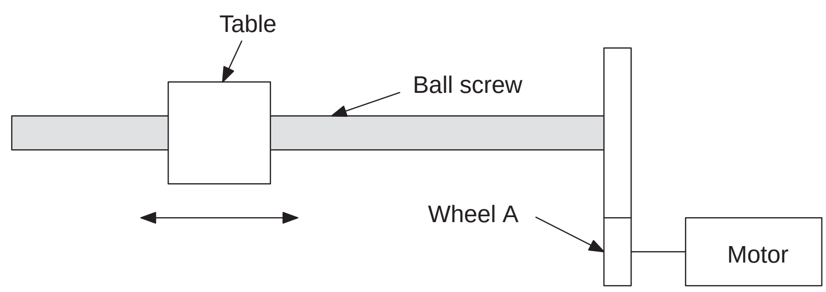
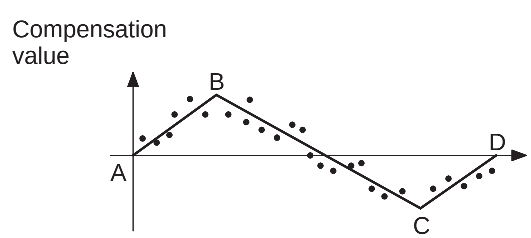


Fig. 15.3 (c) Application of second cylindrical pitch error compensation for a linear axis

15.4 INCLINATION COMPENSATION

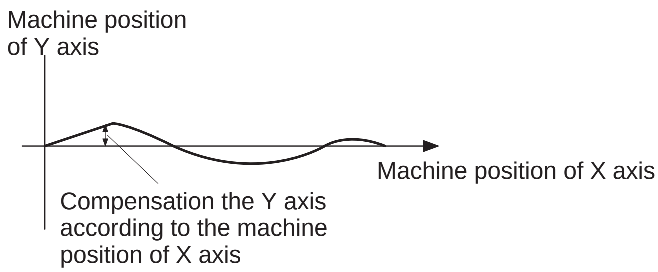
Error caused by machine position, as pitch error of the feed screw, can be compensated by making an approximate value of three lines. The stored pitch error compensation is used over sections where inclination compensation is not enough.
A smooth and high–precision compensation can be done with the inclination compensation. This compensation is also useful in decreasing setting points in the stored pitch error compensation data, so setting in stored pitch error compensation becomes easier. The three lines for inclination compensation can be set by parameters (for each axis).



Compensation data is approximated by lines AB, BC and CD.
All that is necessary is to set A, B, C and D.

15.5 STRAIGHTNESS COMPENSATION

To compensate straightness of the machine, other axes can be compensated according to the move of a certain axis. For example, the Z axis can be compensated according to the move of the X axis.



15.6 BACKLASH COMPENSATION

This function is used to compensate lost motions proper to the machine system. Offset amounts come in a range of 0 to ±9999 pulses per axis, and is set as parameter in detection units.

15.7 PROGRAMMABLE PARAMETER ENTRY (G10, G11)

Parameters and pitch errors data can be set by tape commands. Therefore, following uses can be done for example.

1) Parameter setting such as pitch errors compensation data, etc. when the attachment is replaced.
2) Parameters such as max. cutting speed and cutting feed time constant can be changed according to the machining conditions.

This function can be applied for various purposes.

<u>COMMAND FORMAT</u>

```
G10  L50 ;
N_ _  R_ _ ;
N_ _  P_ _  R_ _ ;
      ⋮
N_ _  R_ _ ;
G11 ;
```

where

G10 L50 ;	Parameter input mode
G11 ;	Parameter input mode cancel
N_ _ ;	Parameter No. (or pitch errors data No. plus 10000)
P_ _ ;	Axis NO. (in the case of axis type parameter)
R_ _ ;	Parameter setting value (or pitch errors data)

> **NOTE**
> Some parameters cannot be set.

15.8 INTERPOLATION–TYPE STRAIGHTNESS COMPENSATION

The conventional straightness compensation function compensates for the non–straightness of a machine by outputting the entire compensation amount for the interval of a pitch error compensation point, specified in a parameter, at one time. Unlike the conventional function, the interpolation–type straightness compensation function distributes the amount of compensation equally throughout the interval of a pitch error compensation point and outputs it as compensation pulses. Compensation data can be set for 128 points and can range between –7 and +7.

15.9 STRAIGHTNESS COMPENSATION AT 128–POINT

The conventional straightness compensation function compensates for the non–straightness of a machine by specifying compensation data for four arbitrary points, using a parameter, and by obtaining compensation data for up to 128 points along an approximate line connecting those four points. Unlike the conventional function, the 128–point straightness compensation function enables the specification of compensation data for up to 128 equally spaced points, in much the same way as the usual pitch error compensation function. By means of this method, the straightness compensation function assures precise compensation. In addition, this method supports up to five combinations of move and compensation axes for the straightness compensation function.

15.10 BI–DIRECTIONAL PITCH ERROR COMPENSATION FUNCTION

This function enables the setting of a pitch error compensation amount for each of the positive and negative movement directions and compensates for pitch errors in each direction. When the direction of axis movement is inverted, the required amount of compensation is automatically calculated from the pitch error compensation data and used to perform compensation similar to the conventional backlash compensation. This method can further reduce any difference between a route in the positive direction and a route in the negative direction.

16 COORDINATE SYSTEM CONVERSION

16.1 AXIS SWITCHING

The actual machine axes, x, y, and z that correspond to the axis addresses, X, Y, and Z specified in program command can be switched one another. Switching is made in six type by CRT/MDI setting or the external axis switching signal.

Table 16.1

	Programmed axis–addresses		
	X	**Y**	**Z**
Machine axes correspond with programmed axes X, Y and Z	x	y	z
	x	z	y
	y	x	z
	y	z	x
	z	x	y
	z	y	x

Axis switching can not be made at manual operation and reference point return (G28, G29 and G30).

16.2 SCALING (G50, G51)

Scaling can be commanded to figures commanded in the machining programs.

G51 IP_ _ P_ _ ;

where

IP : Combination of addresses of axes

P : Magnification

By this command, scaling of the magnification specified by P is commanded with the point commanded by I, J, K as its center. G50 cancels to scaling mode.

G50 : Scaling mode cancel

G51 : Scaling mode command

Commandable magnification is as follows :

0.00001 to 9.99999 times or 0.001 to 999.999 (whether to use magnification 0.00001 or 0.001 is according to parameter selection)

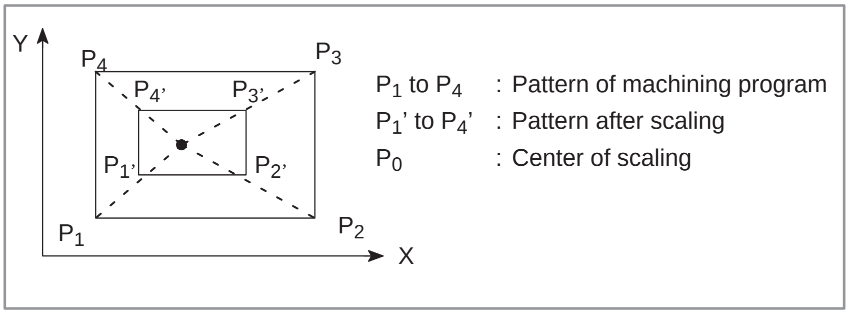


Fig. 16.2

If P was not commanded, the magnification set by the CRT/MDI is applied. A different scaling magnification can be set for each axis. Which of a different scaling magnification for each axis or the same magnification for all the axes can be selected by setting data. The scaling magnification can be set from the CRT/MDI but it cannot be set by a program command. Note that correct circular interpolation cannot be done between axes of different scaling magnifications.

When I, J, K are omitted, the point where G51 was commanded becomes the center of scaling.

Scaling cannot be done to offset amounts such as tool length compensation, cutter radius compensation, or tool offset.

16.3 COORDINATE SYSTEM ROTATION (G68, G69)

Parameters commanded by the program can be rotated. For example, by using this function, when the attached workpiece comes in a position which is somewhat rotated from the machine coordinates, a rotation can be performed to compensate the position.

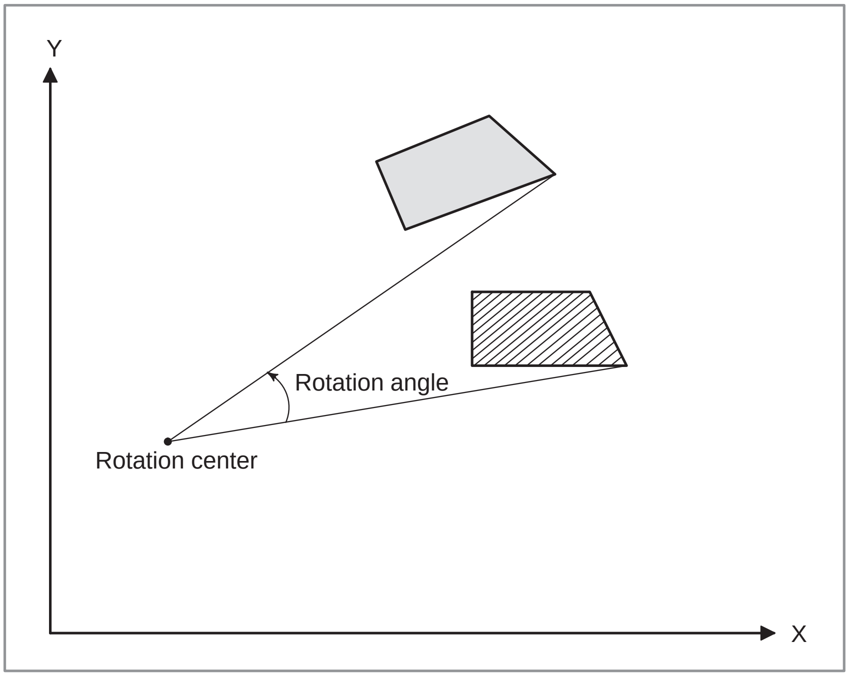


Fig. 16.3

Format

Command format

$$\left\{\begin{matrix}\mathbf{G17}\\ \mathbf{G18}\\ \mathbf{G19}\end{matrix}\right\}\ \mathbf{G68}\ \alpha\_\ \_\ \ \beta\_\ \_\ \mathbf{R}\_\ \_\ :$$

α, β : Specify two axes from X, Y, Z axes of plane G17, G18, G19. (G90/91 modes are recommended)

R : Rotation (+ for the counter clockwise direction. Commanded in absolute value.)

By this command, commands thereafter are rotated in the angle commanded by R, with the point commanded by α, β as the rotation center. Rotation angle is commanded in 0.00001° units in a range of :

$0 \leqq R \leqq 360.00000$

The rotation plane is the plane selected (G17, G18, G19) when G68 was commanded.

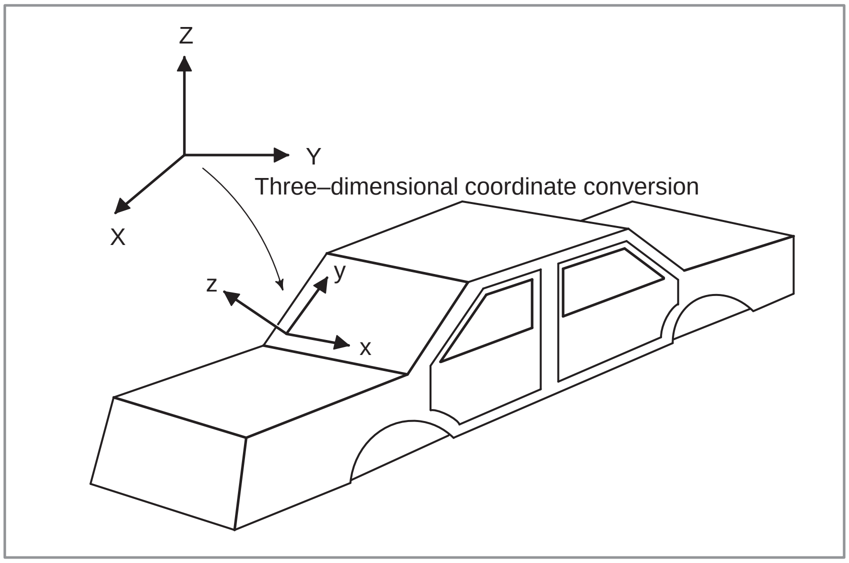
G17, G18 and G19 may not be commanded in the same block as G68.

When a, β is omitted, the point where G68 was commanded becomes the rotation center.

G69; Cancels the coordinate system rotation.

16.4 THREE–DIMENSIONAL COORDINATE CONVERSION

The coordinate system can be rotated about an axis by specifying the center of rotation, direction of the axis of rotation, and angular displacement. This coordinate conversion function is quite useful for three–dimensional machining using a diesinking machine. By applying three–dimensional coordinate conversion to a program generated for machining on the XY plane, identical machining can be executed on a desired plane.



G68 selects the three–dimensional coordinate conversion mode and G69 cancels it.

N1 G68 X x1 Y y1 Z z1 I i1 J j1 K k1 Rα ;
N2 G68 X x2 Y y2 Z z2 I i2 J j2 K k2 Rβ ;

X,Y,Z : Center of rotation (absolute)
I,J,K : Direction of the axis of rotation
R : Angular displacement

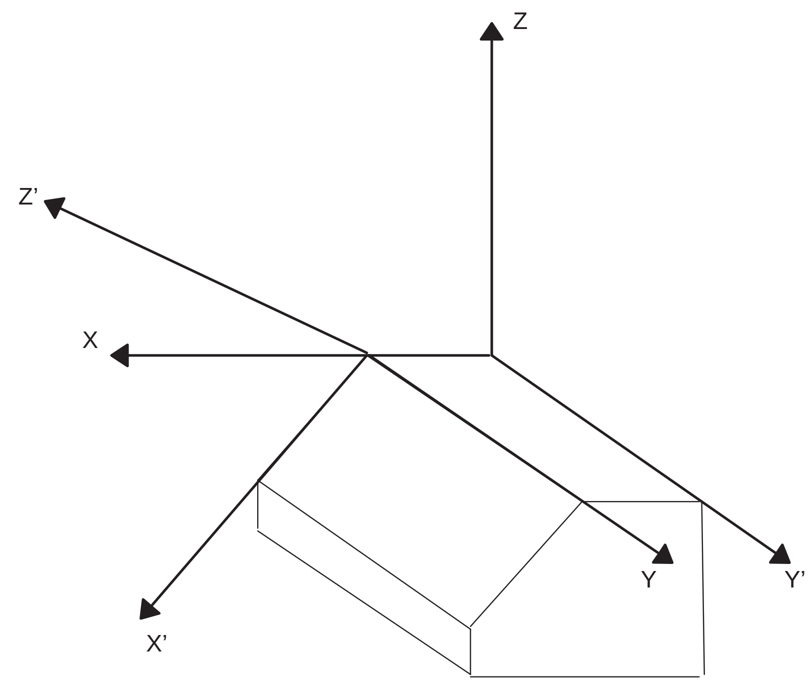
The center, axis, and angle of the first rotation are specified in the N1 block. The N1 block produces a new coordinate system, X’, Y’, Z’. Viewed from the original workpiece coordinate system, the new coordinate system is created by shifting the origin of the original coordinate system by (X1, Y1, Z1) and rotating the original coordinate system about vector (i1, j1, k1) by an angle α. In the N2 block, the center, axis, and angle of the second rotation are specified. The X, Y, Z, I, J, K, and R values specified in the N2 block indicate the values and angle on the coordinate system produced after coordinate conversion of the N1 block. The N2 block produces coordinate system X”, Y”, Z”. Viewed from X’ Y’ Z’, new coordinate system X”, Y”,Z” is created by shifting the center of X’, Y’, Z’ by (X2, Y2, Z2) and rotating X’, Y’, Z’ about vector (i2, j2, k2) by an angle β. The X, Y, and Z values specified in the N3 block are coordinates on X”, Y”, Z”. X’’, Y’’, Z’’ is called the program coordinate system.

If the X, Y, and Z values are not specified in the N2 block, the X, Y, and Z values specified in the N1 block are used as the center of the second rotation. This means that the N1 and N2 blocks have a common center of rotation. When only one rotation is required, the N2 block need not be specified. In the G68 block, specify X, Y, and Z using absolute values. Angular displacement R is positive when the coordinate system is rotated clockwise like a right–hand screw advancing in the direction of the axis of rotation. Bit RTR of parameter No. 6400 determines the unit of R.

– Bit 4 of parameter No. 6400 can specify that only the G69 command cancels the three-dimensional coordinate conversion mode (G68). With such a specification, a system reset, the ERS, ESP, or RRW input signal from the PMC does not cancel the three-dimensional coordinate conversion mode (G68).

– In the three- dimensional coordinate conversion mode (G68), making the M3R input signal from the PMC (address G031.3) high moves the tool in the direction of an axis selected in the coordinate system submitted to three-dimensional conversion (program coordinate system) during manual jog feed, manual incremental feed, or manual handle feed.

  When the M3R signal is low, three- dimensional conversion is not effective for the above three manual operations even in the three-dimensional coordinate conversion mode (G68).



X– Y– Z : Coordinate system before conversion (Workpiece coordinate system)

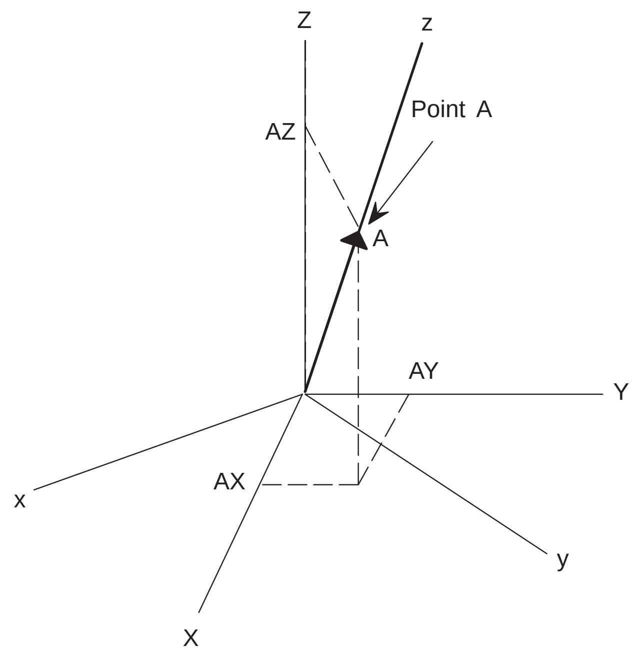
X'– Y'– Z' : Coordinate system after conversion (Program coordinate system)

Example) When the M3R signal is made high during the three-dimensional coordinate conversion mode, manual feed with the Z-axis selected causes a movement in the Z'-axis direction shown above.

– When the current tool position in the workpiece coordinate system is read using the custom macro system variables #5041 to #5055 (ABSOT), conventionally, the coordinates that are read are those in the coordinate system that has not be converted by coordinate conversion even in the three- dimensional coordinate conversion mode (G68). However, bit 5 of parameter No.6400 can specify that the coordinates that are read be those in the workpiece coordinate system that has been converted by three-dimensional coordinate conversion.

X, Y, Z Coordinate system before 3D conversion
x, y, z Coordinate system after 3D conversion

When reading point A using a system variableof custom macro, the following positions areread:
A (AX, AY, AZ) when parameter 6400#5=1
A (0, 0, Az) when parameter 6400#5=0



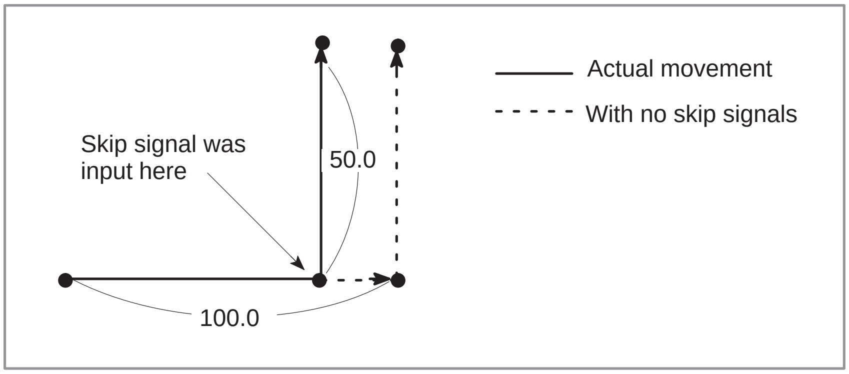
– The 3DROT output signal (address F159.3) informs the PMC that the system is in the three- dimensional coordinate conversion mode (G68). The 3DROT output signal is high during the three- dimensional coordinate conversion mode (G68).

– A status display on the CRT screen indicates that the system is in the three-dimensional conversion mode (G68).

17 MEASUREMENT FUNCTIONS

17.1 SKIP FUNCTION (G31)

By commanding axis move after G31, linear interpolation can be commanded like in G01. If an external skip signal is input during this command, the remainder of this command is cancelled, and program skips to the next block.
G31 is a one-shot command and is valid for the commanded block only.



The following two feed speed for the G31 block can be selected by parameter setting.

1) Feed speed commanded by F
2) Feed speed set by parameter

Coordinate value when skip signal is on, is stored in the system variables #5061 - #5066 of the customer macro, so this function can also be read with the customer macro function.

#5061 . . . . X axis coordinate value
#5062 . . . . Y axis coordinate value
#5063 . . . . Z axis coordinate value
#5064 . . . . Fourth axis coordinate value
#5065 . . . . Fifth axis coordinate value
#5066 . . . . Sixth axis coordinate value

As the skip function can be used when move amount is not clear, this function can be used for:

1) Constant feed in grinding machines
2) Tool measurement with tactile sensor.

17.2 MULTI–STEP SKIP FUNCTION (G31.1 – G31.3)

In blocks with either G31.1, G31.2, or G31.3 commanded, the coordinate value where skip signals (3 types) were input is stored in the custom macro variables, and at the same time, the remaining movement of the block is skipped. It is also possible to skip the remaining dwell with the skip signal by parameter, in a block where: G04 is commanded (dwell). Parameters decide which G code is valid to which of the three skip signals. The skip signal is not necessarily unique to a single G code; it is also possible to set a skip signal to multiple G codes.

Example)

In grinding, end point of machining is not commanded in the program, but is skipped by machining conditions signals from the machine side, and proceded to the next block.
Machining is done in the following procedure:

1. Feed in feedrate of 10mm/min, till machining condition 1 is satisfied.
2. Feed in feedrate of 3mm/min, till machining condition 2 is satisfied.
3. Dwell till machining condition 3 is satisfied.

Machining conditions, skip signals, and G codes in this case correspond as follows:
Machining condition 1 – Skip signal 1 – G31.1
Machining condition 2 – Skip signal 2 – G31.1, G31.2
Machining condition 3 – Skip signal 3 – G31.1, G31.2, G04

```
N1 G31.1 X100.0 F10.0 ; (Feed)
N2 G31.2 X100.0 F3.0 ; (Feed)
N3 G04 X100.0 ;  (Dwell)
```

In cases when machining condition 2 is already satisfied, machining is done actually from the N3 block (dwell).

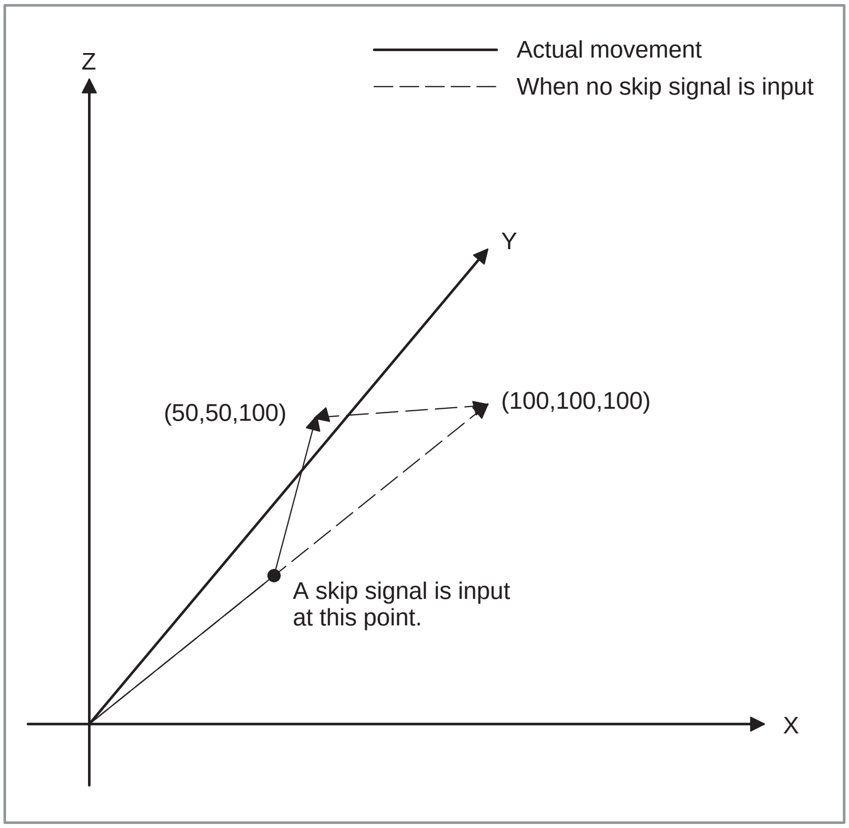
17.3 HIGH–SPEED SKIP SIGNAL INPUT

Delay and error of skip signal input is 0 – 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. This signal is connected directly to the NC; not via the PMC.

17.4 SKIPPING THE COMMANDS FOR SEVERAL AXES

Move commands can be specified for several axes at one time in a G31 block. If an external skip signal is input during such commands, the command is canceled for all specified axes and the next block is executed. The position for each specified axis where a skip signal is input is set in the macro variable for the axis (#5061 to #5066).

Example G31 G90 X100.0 Y100.0 Z100.0 ;
X50.0 Y50.0 Z100.0 ;



17.5 AUTOMATIC TOOL LENGTH MEASUREMENT (G37)

By Commanding:
G37Z_ _ ;

The tool starts moving to the measurement position, and keeps on moving till the measuring position reach signal from the measurement device is output. Moving of the tool is stopped when the tool head reaches the measurement position.

Difference between coordinate values when tool has reached the measurement position and coordinate value commanded by G37 is added to the tool length compensation amount currently used.

17.6 HIGH–SPEED MEASURING POSITION REACH SIGNAL INPUT

Delay and error of measuring position reach signal input is 0 – 2 msec at the NC side (not considering those at the PMC side).
This high–speed measuring position reach signal input function keeps this value to 0.1 msec or less, thus allowing high precision measurement. This signal is connected directly to the NC; not via the PMC.

17.7 TOOL LENGTH MEASUREMENT

Call offset value display screen on the CRT. Relative positions are also displayed on this screen. Reset the displayed relative position to zero. Set the tool for measurement at the same fixed point on the machine by hand. 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.

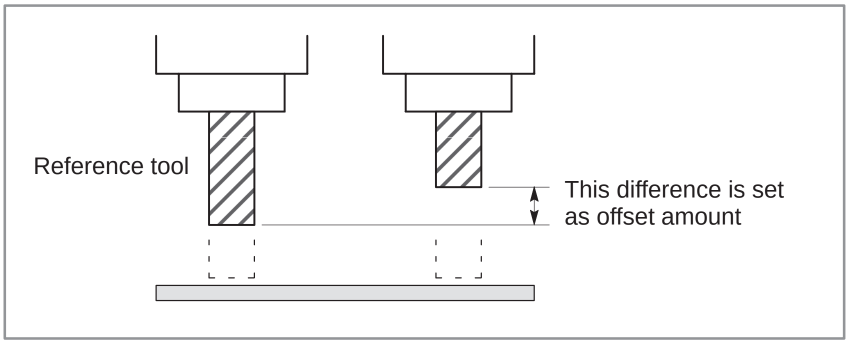


Fig. 17.7

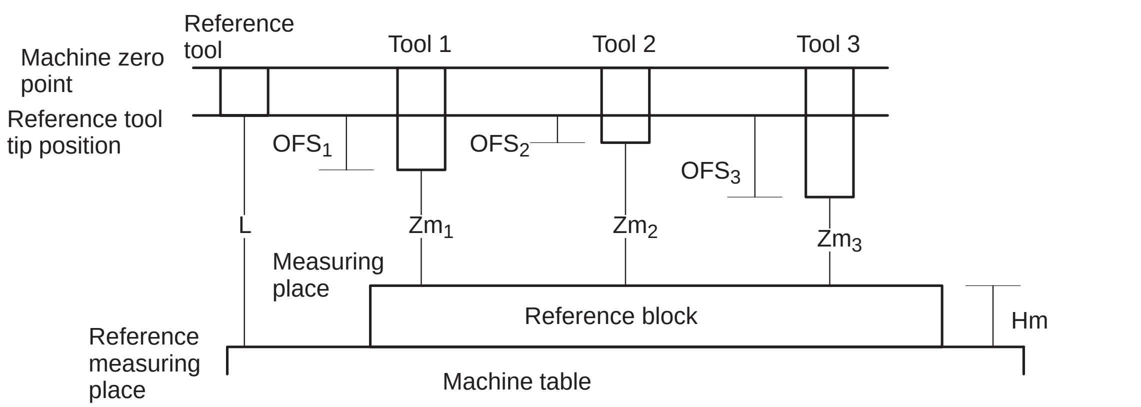
17.8 TOOL LENGTH/ WORKPIECE ZERO POINT MEASUREMENT B

Tool length can be measured only by touching the tool on the outer plane of workpiece or on the sensor, by a manual feed. The tool length is measured along any axis.

The center of reference hole can be made as the zero point of the workpiece coordinate system by applying a tool or touch probe to three points of the reference hole freely selected by a manual feed. Workpiece zero point of the axes X and Y is the center of the reference hole. Also, zero point of the workpiece coordinate system can be set by applying them to the edge of the workpiece instead of the reference hole. Workpiece zero point of freely selected axis can be measured.

The machining set up is done in short time securely because of an easy operation.

Example of tool length measurement



L: Travel distance of the reference tool from machine zero point to the reference measuring plane (Parameter)

Hm: Distance from the reference measuring plane to the actual measuring plane (Setting)

Zm: Travel distance of the measuring tool from the machine zero point to the measuring plane

Tool length compensation value (OFS_i) = Zm_i – Hm – L

Example of workpiece zero point measurement

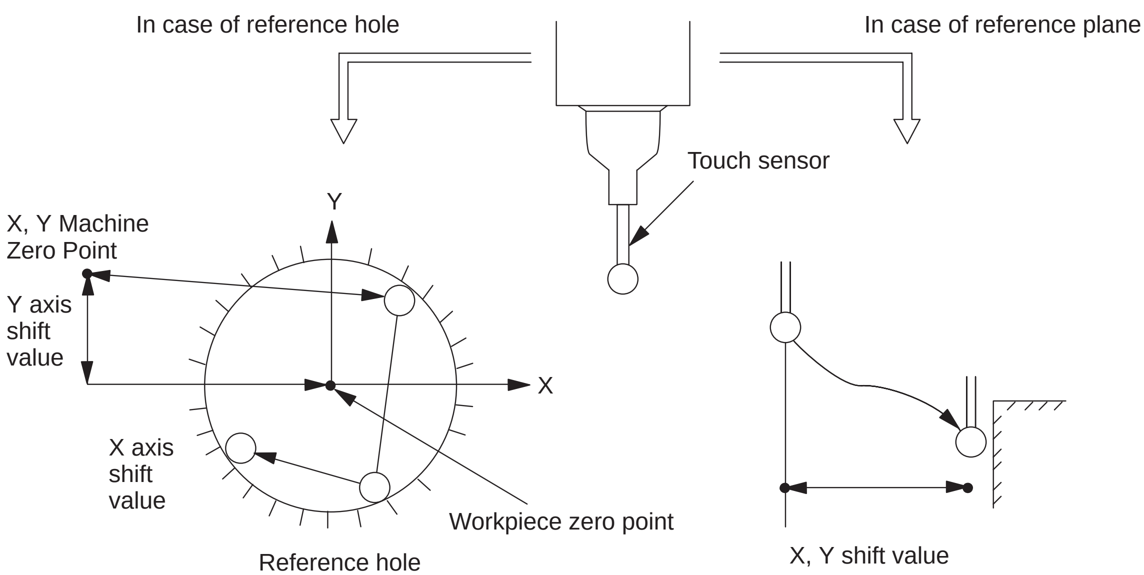


Fig. 17.8

17.9 TORQUE LIMIT SKIP

With this function, an axis moves with a torque limit applied for the feed motor. A skip operation is performed if the motor reaches the torque limit such as, for example, when the axis runs into the stopper.

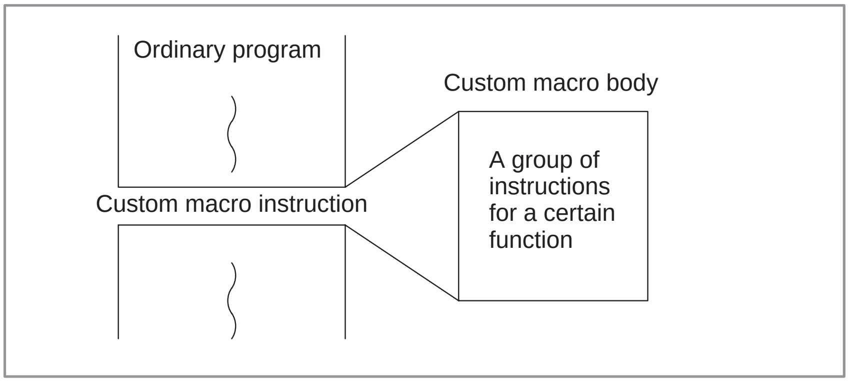
G31 P99 α [amount of movement] F [speed]; where α is an axis address
G31 P98 α [amount of movement] F [speed]; where α is an axis address

A cutting feed command like G01 can be realized by issuing a move command after G31 P99 (or G31 P98) with the motor torque limited (by, for example, executing a torque limit command in the PMC window). In this case, however, the move command is effective only for one axis at a time. When the motor torque reaches the limit, or if a skip signal (or high–speed skip signal) is received while G31 P99 is being executed, the rest of the command is skipped, and the next block is executed.

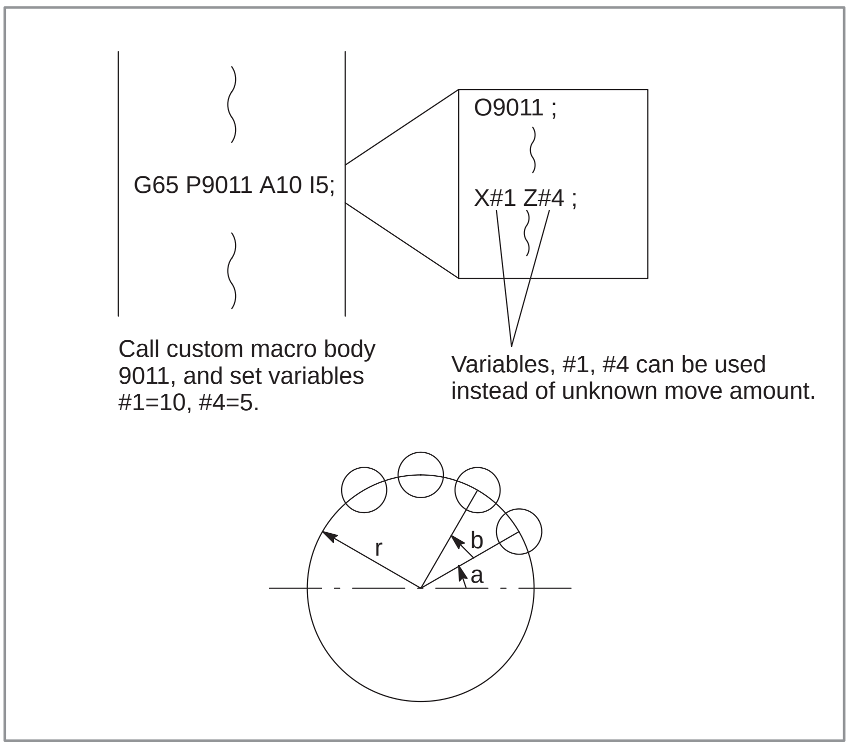
18 CUSTOM MACRO

18.1 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.



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.



Bolt hole circle as shown above 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 NC 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.

G65P p R r A a B b K k ;

- 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 NC 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.

1) Use of variables
   Variables: #i (i=1, 2, 3,.......)
   Quotation of variables: F#33 (#33: speed expressed by variables)

2) Operation between 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), COS (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)
   Example : #5 = SIN [[#2 + #4] * 3.14 + #4] * ABS (#10)

3) Variable naming
   A name with 8 characters or less can be given to variable (#500 to #519). Confirmation and setting of variables are made easier by naming variables, as these names are displayed on the CRT screen with the value of the variable.

4) Control command
Program flow in the custom macro body is controlled by the following command.

i) 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

ii) WHILE (<conditional expression>) DO m (m = 1, 2, 3)
to
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 ;

#120=#120+1 ;

Repeated 10 times.

END 1 ;

5) Format of custom macro body
The format is the same as the sub program.

O Macro number ;

Custom macro body

M99 ;

6) Custom macro instruction

i) 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 to Z) corresponds to which variable number.

ii) Modal call A

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 cancelled by G67.
This function is useful when drilling cycles are programmed as custom macro bodies.

iii) Modal call B

G66.1 P (macro number);
In this macro call mode, command values of each block are all regarded as arguments, and custom macro commanded by G66.1 is called without any execution. It can be regarded that G65P (macro number) is commanded at the head of each block.
This status is cancelled by G67.
In modal call B, command value of each block is once sent to the custom macro as arguments, so execution can be performed after various decisions and processes in the custom macro.
This function is useful, for example when automatically controlling the grind in the grinding machine rectangurally to the forward direction.

iv) Macro call by G codes

The macro can also be called by the parameter-set G codes. Instead of commanding:
N_ _ G65 PΔΔΔΔ <argument assignment> ;
macro can be called just by commanding:
N_ _ G× × <argument assignment> ;.
G code for calling the macro, and macro program number **** to be called, are coupled together and set as parameter.
Maximum ten G codes from G01 to G999 can be used for macro call (G00 cannot be used).
The G code macro call cannot be used in the macro which was called by a G code. It also cannot be used in sub programs called by sub program call with M codes or T codes.

v) Custom macro call by M code
Custom macros can be called by pre-determined M codes which are set by parameters.
The following command
N_ _ G65 P<u>ΔΔΔΔ</u> <Argument assignment> ;
is equivalent to the following command:
N_ _ M× × <Argument assignment> ;
The correspondence between M codes (Mxx) and program number (ΔΔΔΔ) of a macro shall be set by a parameter.
Signal MF and M code are not sent out the same as the subprogram call by M code.
Also when this M code is specified in a program called by a macro calling G code or a subprogram calling M, S, T or B code, the M code is regarded as a normal M code.
Up to ten M codes from M01 to M97 can be used for custom macro calling M codes.

vi) Sub program call by M code
An M code can be set by parameter to call a sub program. Instead of commanding:
N_ _ G_ _ X_ _ Y_ _ … M98 PΔΔΔΔ ; ,
the same operation can be performed simply by commanding:
N_ _ G_ _ X_ _ Y_ _ … M× × ;.
As for M98, M codes are not transmitted.
The M code × × for calling the sub program and the sub program number ΔΔΔΔ to be called are coupled together and set by parameter.
Maximum nine M codes from M03 to M97 can be used for macro call (M30 cannot be used).
Arguments cannot be transmitted. It also cannot be commanded in the same block as the block with M98 command.
When these M codes are commanded in macro called by G code or in subprogram called by M code or T code, they are regarded as ordinary M codes.

vii) Sub program call by 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.
This command cannot be done in the same block with a sub program calling M code, or with M98 command.
When T code is commanded in macros called by G code, or in sub programs called by M codes or T codes, the T code is treated as ordinary T codes.

viii) Sub–program call with S code

An S code can be set by a parameter to call a subprogram.

N_ _ G_ _ X_ _Z_ _ Ss; is equivalent tot the following two blocks.

#147 = s:

N_ _ G_ _ X_ _ Z_ _ M98 P9029;

S code s is stored as an argument in common variables #147.

The S code is not transmitted.

When this S code is specified in a macro called with a G code, or in a subprogram called with an M, S, T, or a B code, the subprogram is not called; but this S code is treated as ordinary code.

ix) Subprogram call with 2nd auxiliary function

A specified code dedicated for 2nd auxiliary function can be set by a parameter to call a subprogram.

N_ _ G_ _ X_ _ Z_ _ Bb; (where B is a 2nd auxiliary function code) is equivalent to the following two blocks.

#146 = b:

N_ _ G_ _ X_ _ Z_ _ M98 P9028;

B code b is stored as an argument in common variables #146.

2nd auxiliary function code is not transmitted.

When this 2nd function code is specified in a macro called with a G code, or a 2nd auxiliary function code, the subprogram is not called; but this 2nd auxiliary function code is treated as ordinary 2nd auxiliary function code.

7) 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.

i) Local variables #1 – #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.

ii) 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 #100 to #149 are cleared when power is turned off, but common variables #500 to #549 are not cleared after power is turned off.

NOTE

1 It is possible to increase number of common variables. For details, see “Number of common variables”.

2 It is possible to apply write protection for the common variables set by the parameter. Writing by the macro program and setting is prohibited.

iii) 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:

a) 128 points DI (for read only)
b) 128 points DO (for output only)
c) Tool offset amount, workpiece zero point offset amount
d) Position information (actual position, skip position, block end position, etc.)
e) Modal information (F code, G code for each group, etc.)
f) Alarm message (Set alarm number and alarm message, and the NC is set in an alarm status. The alarm number and message is displayed on the CRT.)
g) Operator's message (A message can be displayed on the CRT screen by setting an operator's message.)
Kanji, Katakana and Hiragana can be displayed in addition to usual alphanumeric character and special character as an operator message or an alarm message made by custom macro.
h) Clock (Time can be known. A time can also be preset.)
i) Single block stop, Miscellaneous function end wait hold
j) Feed hold, Feedrate override, Exact stop inhibition
k) Mirror image status

8) External output commands
Value of variables or characters can be output to external devices via the reader/puncher interface with custom macro command. Results in measurement is output using custom macro.

9) Limitations

i) Usable variables
#1 – #33, #100 – #149, #500 – #549, and system variables.

ii) Usable variable values
-10^{38} to -10^{-38}
10^{-38} to 10^{38}

iii) Constants usable in <expression>
–99999999 to –0.0000001
0.0000001 to 99999999

iv) Arithmetic precision
8-digit decimal number (in trigonometrical functions, some value may cause fall in precision).

v) Custom macro body call nesting
Maximum 4 folds.

vi) ( ) nesting
Maximum 5 folds.

vii) Repeated ID numbers
1 - 3

10) Example of custom macro

Pocket machining

Custom macro call command

G65 P9802 X x Y y Z z R r Q q I i J j K k T t D d F f E e ;

x, y : Start point (lower left of the pocket) absolute position of X, Y axes

z, r : Z point, R point absolute positions (R point must be at the plus side of the Z point)

q : Cut amount per cycle (a positive number)

i, j : X, Y direction length of the pocket (efficient when both is positive, and i ≧ j.)

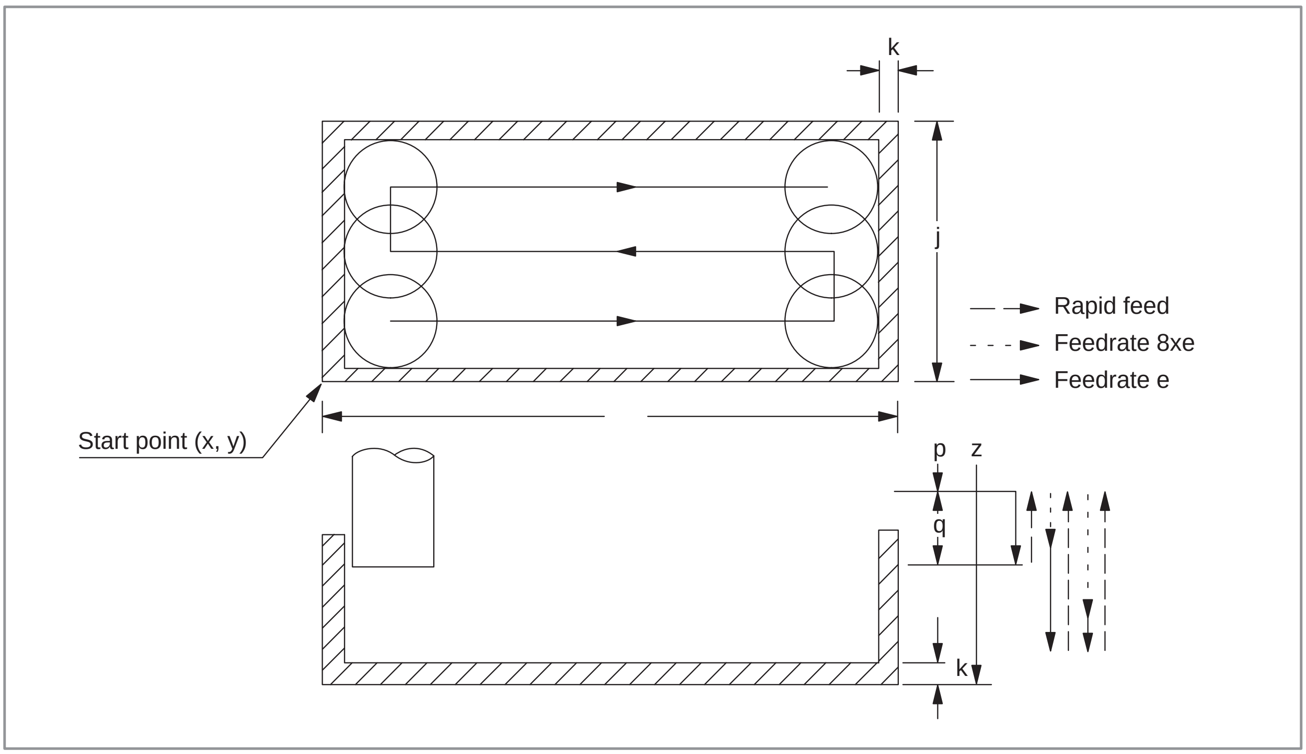
k : Finishing allowance (a positive number)

t : Machining is performed with constant cutting width less than max. cutting width (cutter diameter × t%)

d : Cutter radius compensation number (01 – 99)

f : Feedrate on XY plane

e : Feedrate for cutting. Feedrate up to 1mm above the cutting surface is 8 x e.



Custom macro body

```
O9802;
#27 = #[2000 + #7];
#28 = #6 + #27;
#29 = #5 – 2 * #28;
#30 = 2 * #27 * #20/100;
#31 = FUP [#29/#30];  (Fix up below decimal point)
#32 = #29/#31;
#10 = #24 + #28
#11 = #25 + #28;
#12 = #24 + #4 – #28;
#13 = #26 + #6;
G00X#10 Y#11;
Z#18;
#14 = #18;
DO 1;
#14 = #14 – #17;
IF [#14 GE #13] GOTO 1;
#14 = #13;
N1 G01 Z #14 F#8;
X#12F#9;
#15 = 1;
WHILE [#15LE #31] DO 2;
Y [#11 + #15*#32];
IF [#15 AND 1 EQ 0] GOTO 2;
X#10;
GOTO 3;
N2X#12;
N3#15 = #15 + 1;
END 2;
G00 Z#18;
X#10 Y#11;
IF [#14LE#13] GOTO 4;
G01 Z[#14 + 1] F[8 *#8];
END 1;
N4 M99;
```

18.2 NUMBER OF COMMON VARIABLES

Select common variables from the following:

1) Common variables A
   Common variables #100 – #149, #500 – #549 can be used.
   #100 – #149 will be cleared when power is turned off, but #500 – #549 will be kept after power off.
2) Common variables B
   Common variables #100 – #199, #500 – #599 can be used.
   #100 – #199 will be cleared when power is turned off, but #500 – #599 will be kept after power off.
3) Common variables C
   Common variables #100 – #199, #500 – #699 can be used.
   #100 – #199 will be cleared when power is turned off, but #500 – #699 will be kept after power off.

> **NOTE**
> Part program storage length will become short by 2.2m.

4) Common variables D
   Common variables #100 – #199, #500 – #999 can be used.
   #100 – #199 will be cleared when power is turned off, but #500 – #599 will be kept after power off.

> **NOTE**
> Part program storage length will become short by 7.4m.

18.3 READ/PUNCH FUNCTION FOR CUSTOM MACRO COMMON VARIABLES

The values and names of the common variables (#200 to #999) retained after the power is disconnected can be output to an output device in custom macro statement form.
As shown in the following example, output is in program format. Variable data can be set by executing this program.

Example)
```
%
;
#500=25600*65536/16777216;  . For a normal value
#501=#0; . . . . . . . . . . . . . . . . . . . When the value is empty
#502=0; . . . . . . . . . . . . . . . . . . . . When the value is 0
#503 =............;
............;
SETVN500[ABC,DEF,,,...]  ;  . . . For variable names
M2;
%
```

When this custom macro statement program is executed, values and names are set for the common variables.
As shown in #500 in the above example, the values of variables are generally expressed as mathematical expressions. Since macro variables are handled in floating–point form in the control unit, such mathematical expressions are used to accurately express the values stored internally. The user need not by concerned with this format.

18.4 INTERRUPTION TYPE CUSTOM MACRO

When custom macro interruption signal is input during automatic operation, the block currently under execution is interrupted and the specified custom macro is activated. After execution of this custom macro, it returns to the interrupted block and continues execution of the remaining commands.

```
.
.
.
M96P_ _ _ ;  When custom macro interruption signal is input
             between M96 block and M97 block, custom macro
             specified by P is activated.
.
.
.
M97;
.
.
.
```

With this function, custom macro interruption signal can be input on detection of tool break, tool change cycle can be executed by custom macro, and machining is continued.

18.5 KEY AND PROGRAM ENCRYPTION

To protect programs as custom macro developed uniquely by the users, the following functions are available.

– The registered programs can be locked in.

– The registered programs can be coded and punched.

– The coded and punched tapes (programs) can be registered.

1) Key
   For locked programs:
   – Part program editing cannot be done to prevent unauthorized access to knowhow.
   – Punching of the program cannot be done.
   – Display of the program cannot be done.
   – Uncoded programs cannot be registered.
   – Program umber search cannot be done.

2) Program Encryption
   – The registered programs can be encrypted and punched.
   – The coded and punched tapes (programs) can be registered.

   By coding the program, contents of the program can be kept secret. The coded tape can be attached to the NC.
   The first program number in the tape will not be coded, but characters thereafter will be punched in codes. A “%” will be punched at the end of the tape.

19 FUNCTIONS FOR HIGH SPEED CUTTING

19.1 HIGH SPEED MACHINING (G10.3, G11.3, G65.3)

The high–speed machining function allows the machining program, which is to be pre–processed and stored in the memory before the execution, to be called and executed at a high speed.
By this function, an interruption of the execution of blocks, in which blocks of minute move commands continue like in a three dimensional machining, is eliminated.

(Registration)

G10.3L1PpQq;
.
.
.
(NC commands)
.
.
.
G11.3

p: High speed machining data number

q: Cluster ID number

(Call)

G65.3PpQq;

19.2 MULTI–BUFFER (G05.1)

Normally the CNC calculates the next one block while executing a certain block and transforms it into executable data (execution format).
This is called buffering. By using the multi–buffer function, it is possible to increase the number of such buffering blocks up to fifteen.
This prevents stoppage between very small, consecutive blocks.
In other words, if the number of consecutive minute move blocks is 15 or less, the interruption of the execution between these blocks is eliminated.

Command Format

G05.1; Multi–buffer mode ON
.
.
.
(CNC command)
.
.
.
G05.1P1; Multi–buffer mode OFF

19.3 AUTOMATIC CORNER DECELERATION

Need of deceleration is automatically judged in order to prevent the large sag caused by the acceleration/deceleration and the servo delay on the junction of two blocks in cutting mode (G64). When the difference of speed component of each axis between two blocks is greater than the parameter setting value, deceleration is automatically made at the end point of the block, and move of the following block is started when the speed gets slower than the parameter setting value.

(Example)

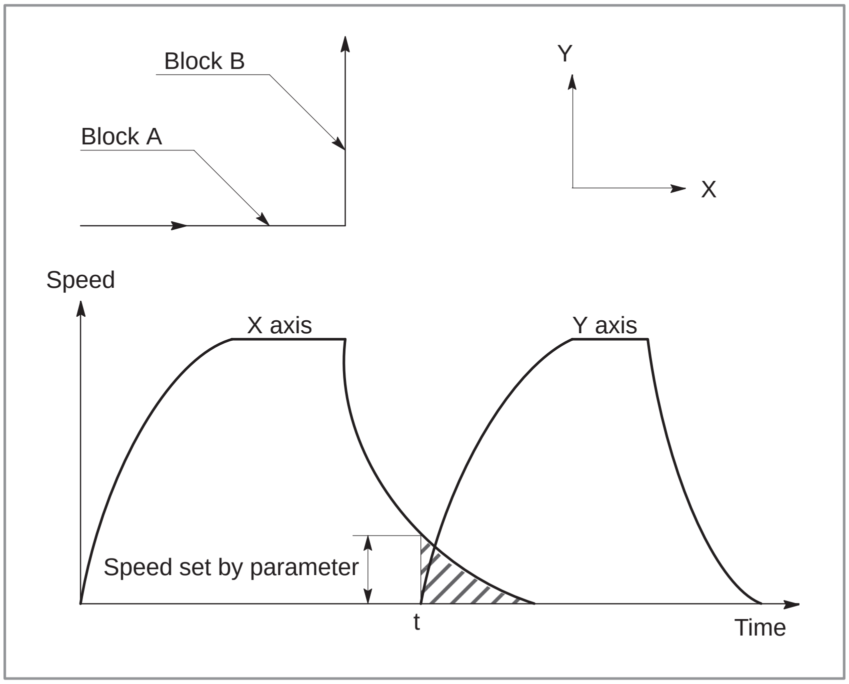


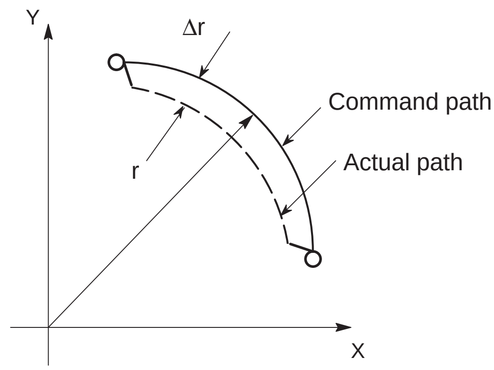
Fig. 19.3

19.4 FEEDRATE CLAMP BY CIRCULAR RADIUS

The machine is accelerated/decelerated automatically when the movement is started/stopped, so that the machine system should not be applied with any shock. When programming, therefore, no consideration needs to be made for acceleration/deceleration.

Especially when performing the high-speed arc cutting, however, the actual tool passage may bring about some error against the designated arc during circular interpolation due to this automatic acceleration/ deceleration.

This error can approximately be given by the following formula;



$$\Delta r = \frac{1}{2}(T_1{}^2 + T_2{}^2)\frac{V^2}{r} \quad \ldots\ldots\ \text{(Formula 1)}$$

Δr : Maximum value of radius error (mm)

v : Feedrate (mm/sec)

r : Circular radius (mm)

T_1 : Acceleration/deceleration time constant (sec)

T_2 : Time constant of servo motor (sec)

When performing the actual machining, the actual arc machining radius (r) and tolerance (Δr) are given, therefore, the maximum permissible speed v (mm/min.) can be given by the formula-(1).

“Feedrate clamp by circular radius” is such function that the circular cutting feed is automatically clamped when the feedrate designated may exceed the permissible tolerance to radial direction against the circular arc having optional radius designated by the program.

19.5 ADVANCED PREVIEW CONTROL FUNCTION

The advanced preview control function has been designed for high–speed, high–precision machining. This function reduces acceleration/deceleration delay and servo delay, which increase as the feedrate increases. When this function is used, the tool is moved as specified, and the machining error in circular or corner machining is reduced.

The advanced preview control function is implemented by the following functions:

1. Look–ahead acceleration/deceleration before interpolation (including advance feed–forward)
2. Multibuffer
3. Feedrate clamp by circular radius
4. Linear acceleration/deceleration after interpolation

19.6 HIGH–PRECISION CONTOUR CONTROL

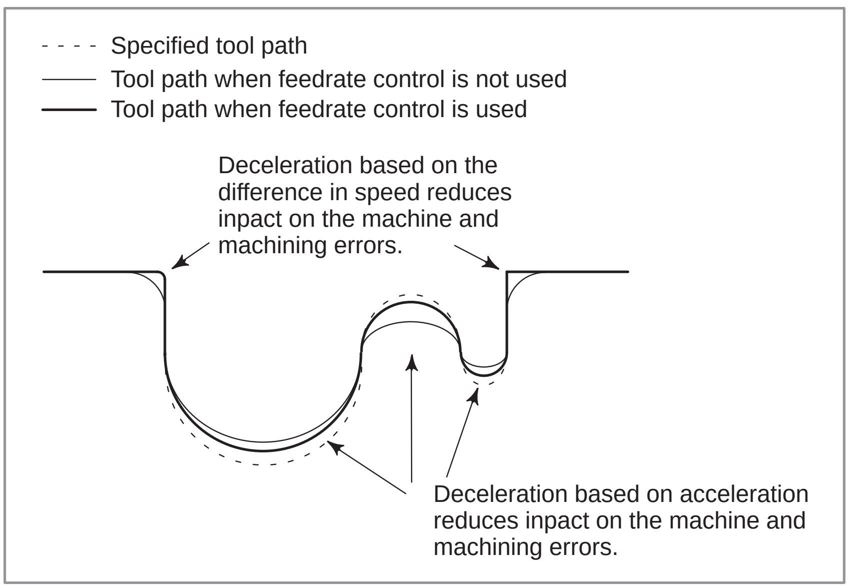
The high–precision contour control function allows precise high–speed machining when a free sculptured surface, such as a metal die, is machined using linear interpolation. To achieve greater speed and precision, the function calculates and controls the appropriate feedrate (automatic feedrate control) according to the machining profile.
This function includes the option multibuffer (J986).
To enter the automatic feedrate control mode, specify the following:

G05.1 Q1 ;

The feedrate is automatically controlled in automatic feedrate control mode by buffering the next 15 blocks (the next 60 blocks when the optional multi–buffer function for 60 blocks is provided). The feedrate is determined by the following conditions. If the specified feedrate exceeds the value determined by the conditions, acceleration/deceleration before interpolation is performed to reach the determined feedrate.

1) Change in speed on each axis at corners and specified allowable speed change
2) Expected acceleration on each axis and specified allowable acceleration
3) Expected variations in cutting load from movement along the Z–axis

If an appropriate feedrate is determined and acceleration/deceleration is performed according to these conditions, the impact on the machine and the machining errors liable to be produced when the direction in which the tool moves changes substantially are decreased. As a result, precise high–speed machining is enabled.
The feature of acceleration/deceleration before interpolation is used for automatic feedrate control. Since extending the time constant does not produce a machining error, machining can be done with small impact and high precision.
A specific time constant for acceleration/deceleration after interpolation is provided for the automatic feedrate control mode. By setting the time constant for the automatic feedrate control mode to a small value, the machining error due to acceleration/deceleration delay is reduced.



19.7 FEED FORWARD CONTROL

To reduce the machine shape error caused by the servo follow–up error (delay), the position loop gain (Kp) of the servo may be as high as possible. If, however, the position loop gain is too high, the servo system will oscillate.

The feed forward control enables reduction of the servo follow–up error without increasing the position loop gain.

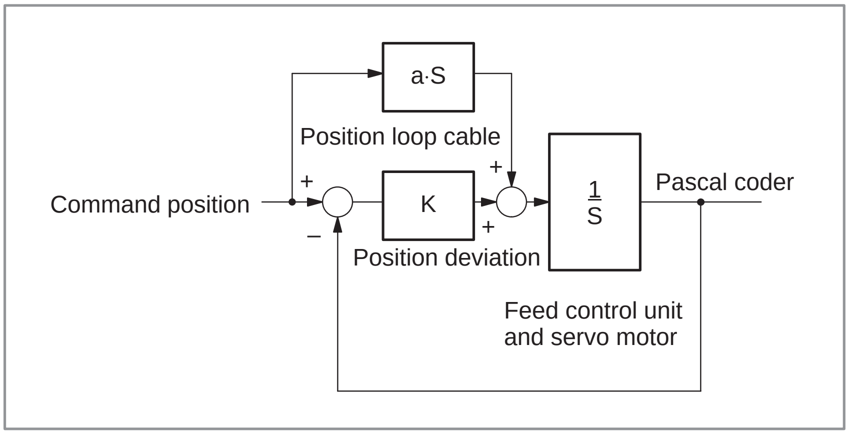


Fig. 19.7 Feed forward control

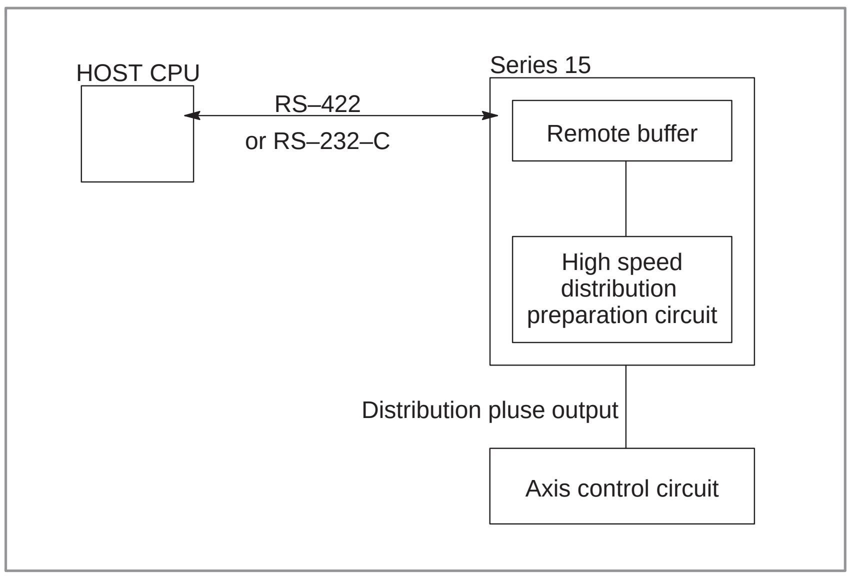
19.8 HIGH–SPEED DISTRIBUTION BY DNC OPERATION USING REMOTE BUFFER

A high–speed distribution can be executed by DNC operation using a remote buffer for the CNC with sub CPU.

After reading one block of data, the CNC first calculates the block data, generates the distribution pulse for each axis, and transfers it to the servo system to revolve the motor. In general, if the time required for generating distribution pulses for one block is shorter than the motor revolving time for one block, the pulse distribution intervals will be generated between blocks. That is, to execute the program having a series of minute blocks, the CNC may stop between blocks because generation of distribution pulses cannot catch up with the program execution speed. Therefore, the time for generating distribution pulses of one block (Block processing time) is one of the important factors to indicate the performance of the CNC.

The high–speed DNC operation (using the remote buffer in Series 15) allows the time required for generating distribution pulses for one block can be greatly reduced.

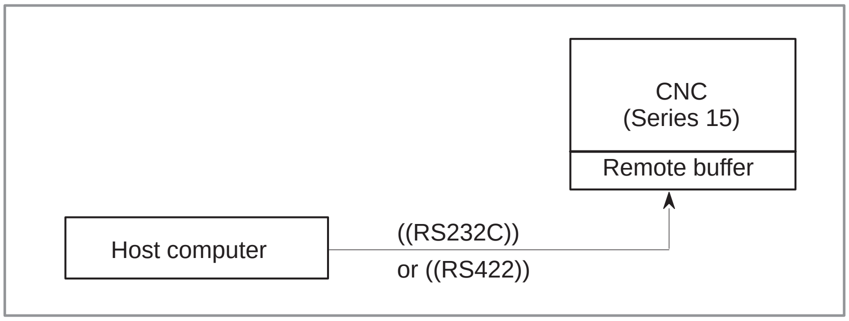
This function enables generation of the distribution pulses for one block in a short time, thus enabling execution of the program having a series of minute blocks at high speed without any stop between these blocks. For example, the program with a series of 1–mm blocks (for 3 axes simultaneous linear interpolation) can be executed at 15 m/minute during DNC operation.



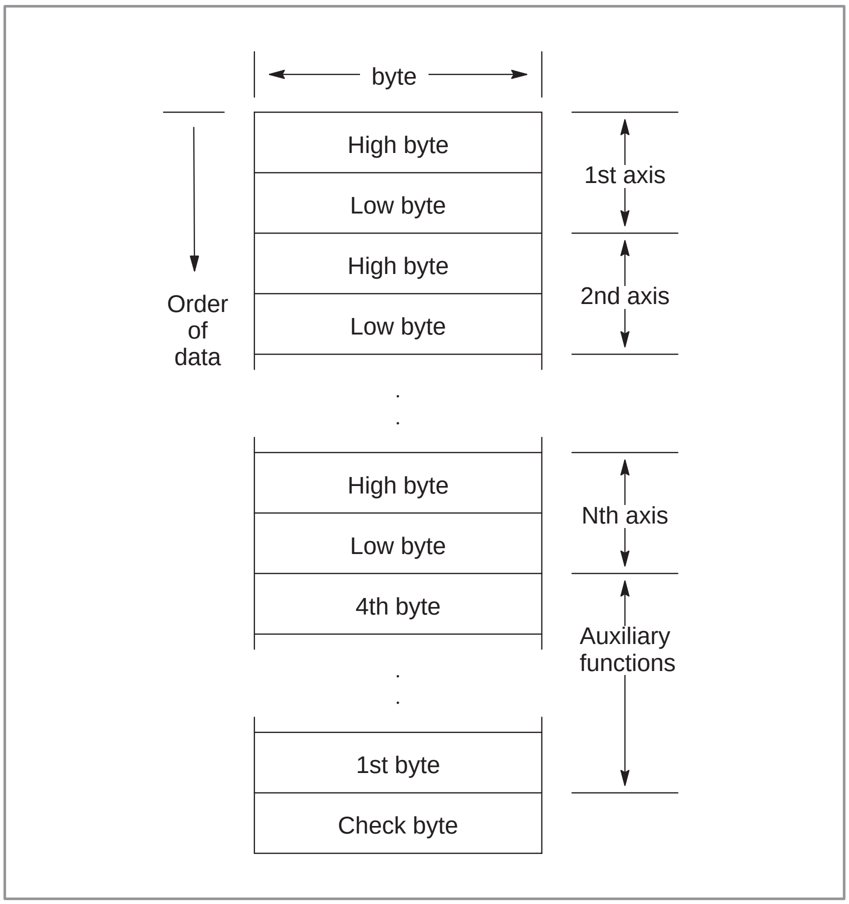
19.9 BINARY DATA INPUT OPERATION BY REMOTE BUFFER

A high–speed distribution can be executed by DNC operation using a remote buffer. Command the “G05;” by the normal NC command format without any other NC commands in the block, and then command the move data and auxiliary function using the following format to perform the binary input operation function. Set the “0” to both the move distance of all axes and auxiliary functions to return to the normal NC command format thereafter.

- Binary input operation On : G05;
- Binary input operation Off : Sets the move distance of all axes and auxiliary functions to “0”.



- Data format for binary input operation (All the data are binary)



The binary input operation data is in the format consisting of the move distance per unit time for each axis set in order.
The unit time can be selected from the following:

Unit time
2 ms
4 ms
8 ms
16 ms

NOTE
1 The following is required when performing binary input operation in units of "2 msec":
- Number of all controlled axes should be 3 or less.
- The system should be provided with the SUB CPU.

2 The following is required when performing binary input operation in units of "4 msec":
- Number of all controlled axes should be 6 or less.

19.10 DISTRIBUTION PROCESS BY REMOTE BUFFER

The distribution process is used to convert the NC program received from the host computer to the distribution data at the remote buffer side and to supply the converted distribution data to the CNC side. Use of this function allows the NC program where an extreme short move distance continues to be operated at high speed. The distribution process by the remote buffer is up to 3 axes.

The NC program format is the same as normal NC program. However, there are two sections, namely the one for performing distribution process and the one for passing to the CNC.

The section for performing distribution process is hereafter called the high–speed machining section.

The definition of high–speed machining section is commanded as in the followings:

Command	Meaning
G05 P1;	Start of high–speed machining section
G05 P0;	End of high–speed machining section

Commands which can be described in high–speed machining section are shown in the following table. The address not listed in the table is ignored during distribution process (distribution section) even if it is specified.

Command address	Meaning
G00	Interruption of distribution process (Note 1)
G01	Distribution process restart
First axis address	Move distance of first axis
Second axis address	Move distance of second axis
:	:
n–th axis address (n≦3)	Move distance of n–th axis (n≦3)
F	Cutting feedrate (Note 2)

NOTE
1 If the G00 is commanded even in the high–speed machining section, the distribution process is interrupted until the G01 is commanded.
2 F ≦ 15000 mm/min or F ≦ 600 inch/min
The number below the decimal point is ignored.

19.11 HIGH–PRECISION CONTOUR CONTROL USING 64–BIT RISC PROCESSOR

Machining error by acceleration/deceleration after interpolation is partly responsible for machining errors caused by the CNC. To eliminate this machining error, a RISC processor is used to enable the high–speed execution of the following functions.

(1) Acceleration/deceleration before interpolation based on the advance loading of multiple blocks, which are free of machining errors caused by acceleration/deceleration

(2) Automatic speed control function, which assures smooth acceleration/deceleration, where changes in the figure and feedrate and the maximum allowable machine speed are processed correctly by loading multiple blocks in advance

(3) NURBS interpolation

The smooth acceleration/deceleration achieved in this way increases the feed forward coefficient, thereby reducing servo–system follow–up control errors.

20 AXES CONTROL

20.1 FOLLOW UP FUNCTION

Normally, the machine is controlled to move to a commanded position. However, when the follow up function is applied, actual position in the NC is revised according to the move of the machine.
Follow up function is activated when:
– Emergency stop is on
– Servo alarm is on
Follow up is carried out and machine movement during the emergency stop and servo alarm is followed up in the NC, so actual position of the machine is reflected in the NC. Therefore, machining can be resumed after the emergency stop or the servo alarm has been deactivated, without performing the reference point return again.
However, when a trouble has generated in the position detection system, the system cannot follow up correctly. So present position in NC does not become correct value.
By parameter setting, follow up function can also be applied to:
– Servo off
status . It is also valid in cases when the machine is moved with a mechanical handle.

20.2 FOLLOW–UP FOR EACH AXIS

The conventional follow–up function performs follow–up for all the axes during the servo off state if parameter No.1800#2 (FVF) is set to 1.
The new follow–up function performs follow up for all the axes when parameter No.1800#2 (FVF) is set to 1. When the parameter is set to 0, however, it allows the operator to specify for each axis whether follow up is performed.
Parameter No.1802#3 is used to specify whether follow–up is performed during the servo off state. When the parameter is set to 0, follow–up is not performed. When it is set to 1, follow up is performed.

20.3 MECHANICAL HANDLE FEED

It is possible to move the machine by hand, using the mechanical handle installed on the machine; not by the NC (servo motor).
Move distance by the mechanical handle is followed up and actual position in the NC is revised. The mechanical handle feed is done by inputting the servo off signal of the axis fed. Parameter setting is necessary to command follow up function when servo off signal is on.

20.4 SERVO OFF

Servo on/off control per axis is possible by signals from machine side. This function is generally used with the machine clamp.

20.5 MIRROR IMAGE

The MDI-commanded or the program-commanded move direction of each axis can be reversed and executed.
Mirror image is set by MDI setting or by the switch on the machine side. Mirror image can be applied to all axes.

20.6 CONTROL AXIS DETACH

It is possible to detach or attach rotary tables and attachments with this function. Switch control axis detach signal according to whether the rotary tables and attachments are attached or detached. When this signal is on, the corresponding axis is excluded from the control axes, so the servo alarm applied to the axis are ignored. The axis is automatically regarded as being interlocked. This signal is not only accepted when power turned is on, so automatic change of attachments is possible any time with this function.
The same switching as with this signal can also be performed with the MDI setting.

20.7 SIMPLE SYNCHRONOUS CONTROL

The traveling command of master axis is given to two motors of master and slave axes in a simple synchronous control. However, no synchronous error compensation or synchronous error alarm is detected for constantly detecting the position deviation of the master and slave axes to compensate the deviation.

In the manual reference point return, the master and slave axes similarly move until the deceleration operation is performed. After that, the detection of grid is performed independently.

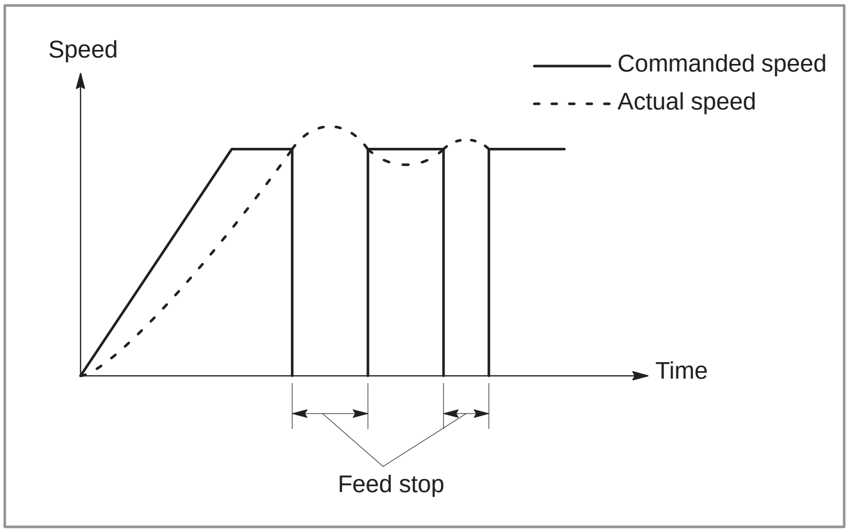
The pitch error and backlash compensation are independently performed for the master and slave axes.

An input signal from the machine side can be select whether the axis traveling is carried out based on the travelling command for that axis as in normal case or whether the axis travelling is carried out while synchronizing with the travelling of any other axis.

Simple synchronous operation is allowed in the automatic operation by tape command, manual data input, or memory command and in the manual operation such as manual continuous feed, manual handle feed, incremental feed, or manual reference position return.

20.8 FEED STOP

This function usually checks position deviation amount during motion. If the amount exceeds the parameter set "feed stop position deviation amount", pulse distribution and acceleration/deceleration control is stopped for the while exceeding, and move command to the positioning control circuit is stopped.
The overshoot at rapid feed acceleration is thus kept to a minimum.



20.9 ARBITRARY COMMAND MULTIPLY (CMR)

Arbitrary command multiply is used in case the detection unit is a special kind of value.
Range of the arbitrary command multiply is as follows.
1/1 – 1/27 (multiplication: 1/n, provided that n: 1 – 27)
1 – 48 (multiplication: n, provided that n: 1 – 48)
The following is the range of the standard command multiply.
Multiplication of 0.5 to 10 (multiplication unit: 0.25)

20.10 TWIN TABLE CONTROL

The synchronous operation, independent operation, and normal operation for two or more specified axes can be switched by an input signal from the machine side.

The following operations can be performed in the machine which has two tables which can be independently driven (for example Y and V axes):

1) Synchronous control

   It is used for cutting a large workpiece which requires two tables.

   The synchronous control of master and slave axes (V axis) is performed based on the travelling command of master axis (Y axis).

   The synchronous control gives the travelling command of master axis to two servo motors of master and slave axes. No synchronous error compensation is carried out for constantly detecting the deviation of two servo motors and for giving compensation for the servo motor of slave axis to minimize the deviation. Also, no synchronous error alarm can also be detected.

   Synchronous control of automatic operation, manual continuous feed, manual handle feed, and incremental feed can be made. However, no synchronous control of manual reference point return can be made.

2) Independent control

   It is used for cutting a small workpiece on one of two tables.

   It is the travelling command of master axis and is used for performing the travelling of master axis only or slave axis only.

   Even in the independent operation of slave axis, the program command may be the same command as in the master axis. Therefore, the same command program can be used when the workpiece is placed on either table.

   Independent control of automatic operation can be made. Manual operation is carried out the same as that of normal control.

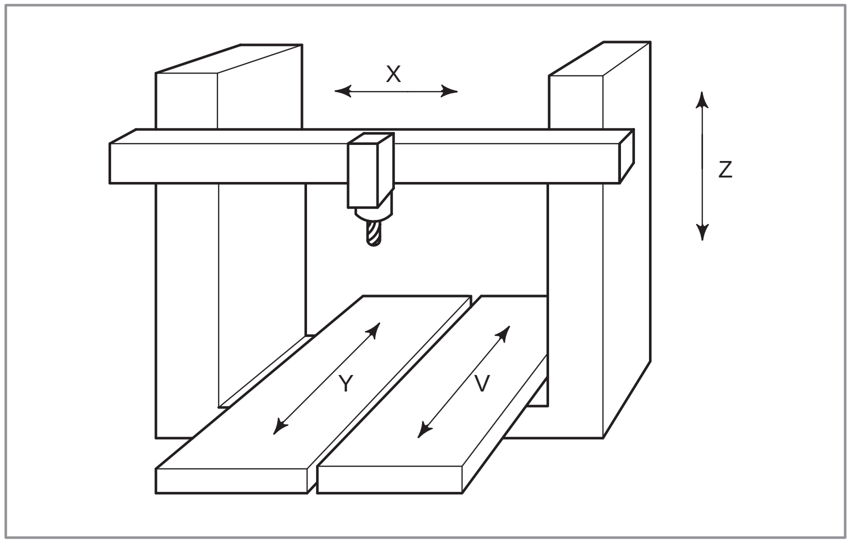
3) Normal operation

   It is used for cutting separate workpieces on each table.

   This control is the same as the normal CNC control. The master and slave axes are travelled by independent axis addresses (Y and V).

   The travelling commands of master and slave axes can be commanded in the same block. Both the automatic and manual operations are the same as the normal CNC control.

   The assignment of the master and slave axes to any axis is carried out by setting parameters.



20.11 SIMPLE SYNCHRONIZATION CONTROL POSITIONAL DEVIATION CHECK FUNCTION

During simple synchronization, an OT alarm is issued when the difference between the positional deviations for the synchronized axes (error counter value) is greater than the value set in a parameter. If this occurs in automatic operation, the movement is decelerated and stopped along all the axes. If this occurs in manual operation, the movement is decelerated and stopped only along the axes subject to simple synchronization. A signal is also issued in this case.

20.12 NORMAL DIRECTION CONTROL (G41.1, G42.1)

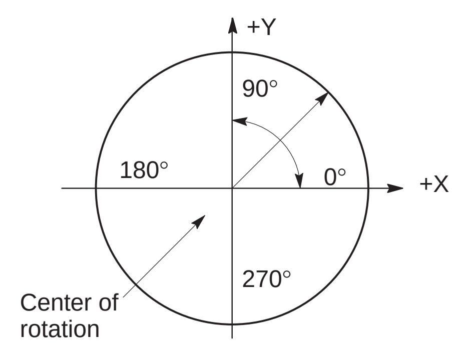
The rotation axis (C axis) can be controlled by commanding the G41.1 or G42.1 so that the tool constantly faces the direction perpendicular to the advancing direction during cutting.

G40.1: Normal direction control cancellation mode
(No normal direction control can be performed.)

G41.1: Normal direction control left side on
(Control is made to allow facing perpendicular to advancing direction to the left)

G42.1: Normal direction control right side on
(Control is made to allow facing perpendicular to advancing direction to the right)

In the normal direction control, control is made so that the tool may be perpendicular to the advancing direction on the X-Y plane.
With 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.
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 travelling 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 travelling direction.
Normal direction control is performed for the path after compensation during the cutter compensation mode. The feedrate of rotation of C axis inserted at the starting point of each block becomes the fede rate 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 command, 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 command.

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

20.13 CHOPPING FUNCTION (G81.1)

When contour grinding is performed, the side face of workpiece can be grinded by executing the contour program at other axes while the grinding axis (axis with a whetstone) is moved up and down. Chopping is commanded by the command of G81.1 and is inputted programming.

(1) Chopping by the program command

G80;

G80: Chopping mode command cancellation

G81.1 Z. . . Q. . . R. . . F. . . ;

Z: Top dead center position (Applicable to axes other than the Z axis.)

Q: Distance between the top and bottom dead center (Set by the incremental value based on the top dead point.)

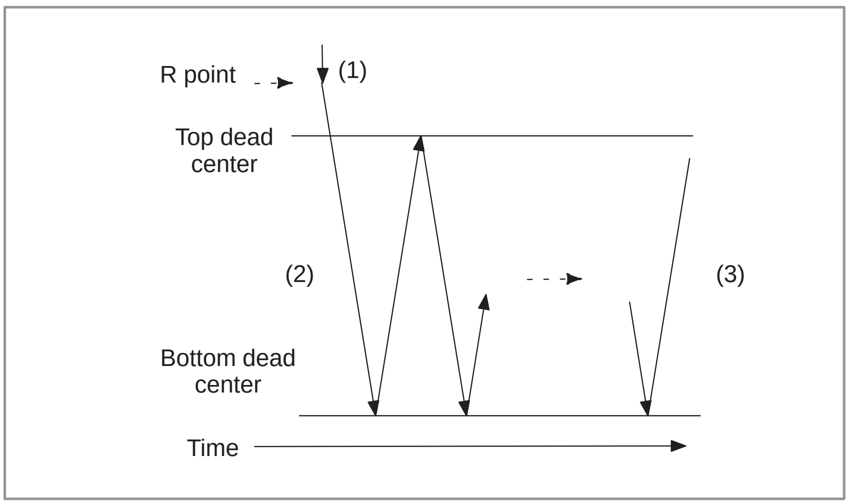
R: Distance from the top dead center to the R point (Set by the incremental value based on the top dead point.)

F: Chopping feedrate

(1) The positioning to the point R can be performed

(2) After that, the reciprocating motion continues at the commanded rate between the upper and lower dead points. Override can be applied to, the shopping rate by the override signal for chopping.

(3) The chopping operation is cancelled, returning to the point R by G80 command



Through the travelling command of chopping axis and canned cycle cannot be commanded during chopping mode, other NC commands can be commanded. Chopping operation continues in both manual mode and feed hold status. The chopping operation is suspended, returning to the point R by reset

(2) Chopping by the input signal

To start by the chopping start signal CHPST, previously set the data for the chopping axis, reference position, top dead point, bottom dead point and chopping rate. When the chopping start signal CHPST is operated from LOW to HIGH, the chopping operation is started. This chopping operation is independent of the operation mode selected. If the chopping axis is in the axis movement, the chopping operation is ignored.

(3) Servo delay compensation

When the grinding axis is operated at high–speed chopping operation, actual tool cannot reach the position commanded by program due to the servo delay and the delay on acceleration/deceleration.
The difference between the position commanded by programs and that of actual tool is measured. Then, this difference is automatically compensated. In order to compensate the shortage, increase the move command amount between the upper dead point and lower dead point and then perform chopping command at the rate where the number of choppings per unit time is equal to that commanded by programs.

20.14 AXIS CONTROL WITH PMC

Any axis can be released from the control of CNC and directly controlled from PMC. That is, input of commands such as moving distance and feedrate commands from PMC allows the axis to move independently of other axes moving under control of CNC. Therfore, use of an axis of CNC enables control of peripheral devices such as a turret, pallet, and index table. Which of CNC and PMC controls each axis can be selected by the input signal.
The following operations can be directly controlled from PMC:

(i) Rapid feed with the specified moving distance.

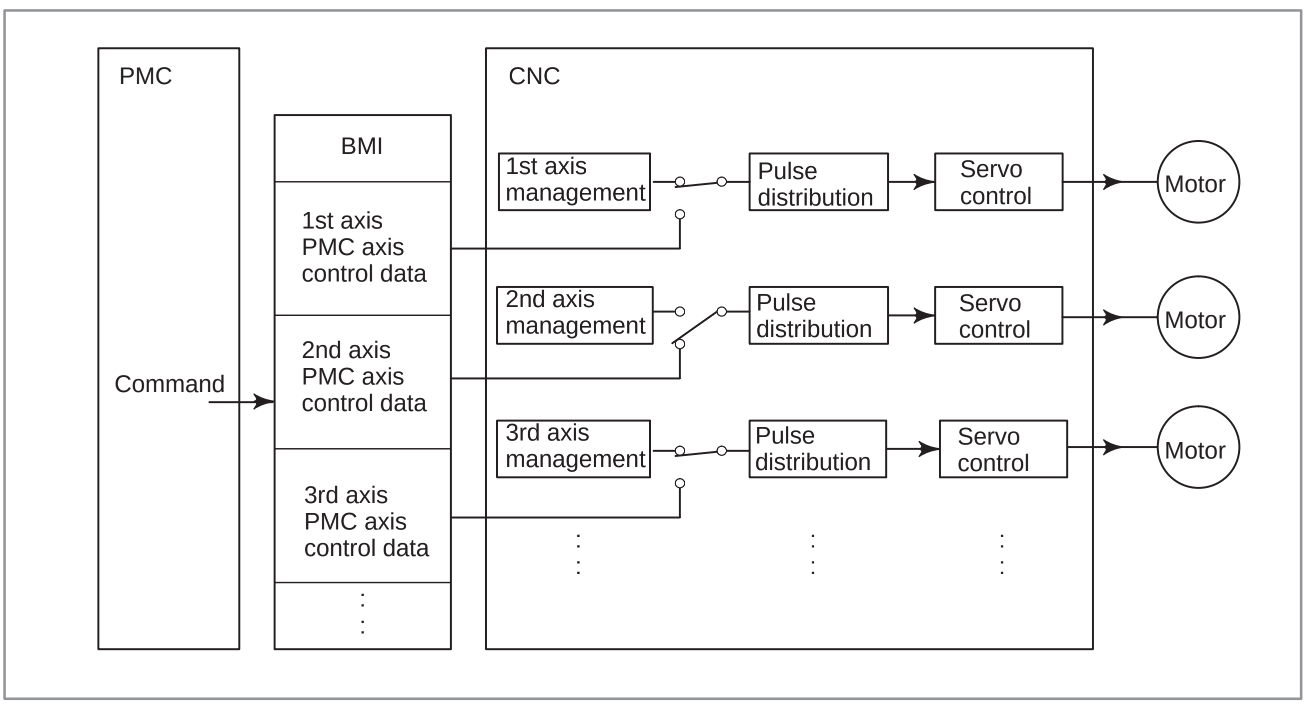
(ii) Cutting feed with the specified moving distance.

- The feedrate and override can also be specified.
- The cutting feed can be started simultaneously with and other PMC control axis.

(iii) Reference position return.

(iv) Positioning on a machine coordinate point.

(v) Dwell



20.15 UPGRADED 5–AXIS CONTROL COMPENSATION PARAMETER

(1) Specifying the coordinates

The 5–axis control functions automatically calculate the direction of the tool axis, which varies as the rotation axis (AC–axis, BC–axis, or AB–axis) moves, in order to manually move the tool with a handle or apply tool length compensation. The coordinates for the axis, which determine the direction of the tool axis, can be set in parameters No.7546 and 7547. The following 5–axis control functions can be used when the rotation axis is only mechanically operated an not an NC axis:

- Three–dimensional handle feed
- Tool length compensation along the tool axis (G43.1)

> **NOTE**
> The AC–axis means the A–axis and C–axis. The BC–axis and AB–axis also conform to this notation.

When the coordinates for the rotation axis are set in parameters for the above functions, the A–, B–, or C–axis can be used as an axis independent of the 5–axis control functions. For an axis independent of the 5–axis control functions, coordinates are updates as the axis moves but are not used for the functions.

(2) Display

The absolute coordinates are displayed by subtracting the tool length compensations along the tool axis.

(3) Display for three–dimensional coordinate conversion

The remaining distance the tool must be moved for three–dimensional coordinate conversion can be displayed about the program coordinate system and about the workpiece coordinate system.

20.16 ROLL–OVER FUNCTION FOR A ROTATION AXIS

The roll–over function for a rotation axis prevents a coordinate overflow for the corresponding rotation axis.

The improved roll–over function for a rotation axis can be enabled or disabled by using programmable parameter input (G10).

When the roll–over function for a rotation axis is executed, each absolute coordinate is kept within the range of 0 to 359.999 degrees.

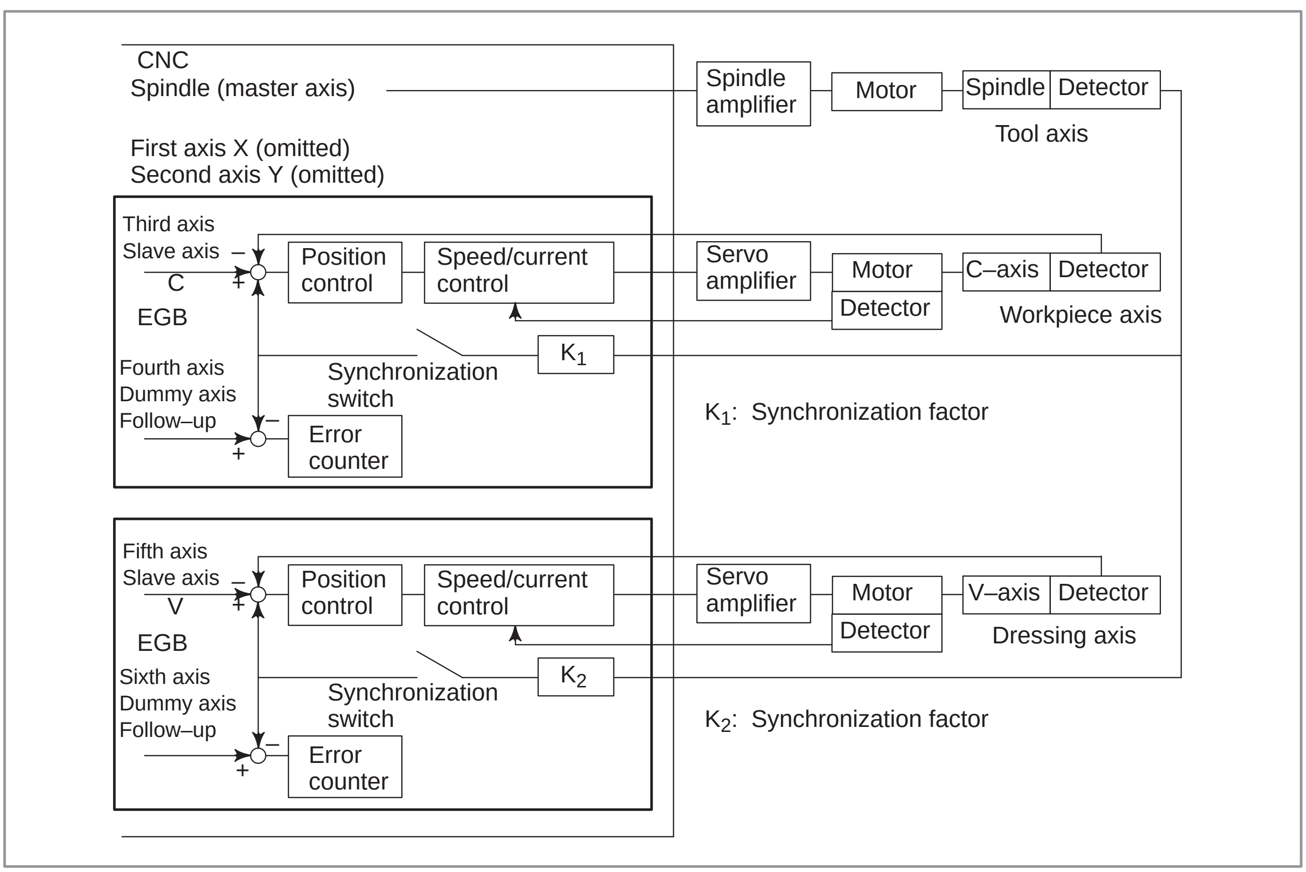
In the incremental mode, a specified value directly indicates an angular displacement. In the absolute mode, the specified value is converted to the remainder obtained by dividing the specified value by 360 degrees. The difference between the converted value and the current value indicates the angular displacement. The movement by angular displacement is always made in the shorter direction. That is, if the difference between the converted value and the current value is greater than 180 degrees, the movement to the specified position is made in the opposite direction. If the difference is 180 degrees, the movement is made in the normal direction.

20.17 TWO AXES ELECTRONIC GEAR BOX

This function rotates a workpiece in synchronization with a rotating tool, or moves a tool in synchronization with a rotating workpiece to produce high–precision gears, screws, and so forth. A desired synchronization ratio can be programmed. This function can implement an electronic gear box (EGB) that enables the user to reprogram the synchronization ratio between a workpiece and tool.
When the two axes electronic gear box option is selected, up to two groups of axes can be specified for synchronization. This means that on a gear grinder, for example, the user can use one axis to rotate a workpiece in synchronization with the tool, and can use the other axis to move the dressing axis in synchronization with the tool.

[Example of controlled axis configuration (gear grinder using the two axes electronic gear box)]

Spindle	:	EGB master axis: Tool axis
First axis	:	X
Second axis	:	Y
Third axis	:	C–axis (EGB slave axis: Workpiece axis)
Fourth axis	:	C–axis (EGB dummy axis: Not usable as an ordinary controlled axis)
Fifth axis	:	V–axis (EGB slave axis: Dressing axis)
Sixth axis	:	V–axis (EGB dummy axis: Not usable as an ordinary controlled axis)



20.18 SKIP FUNCTION FOR EGB AXIS

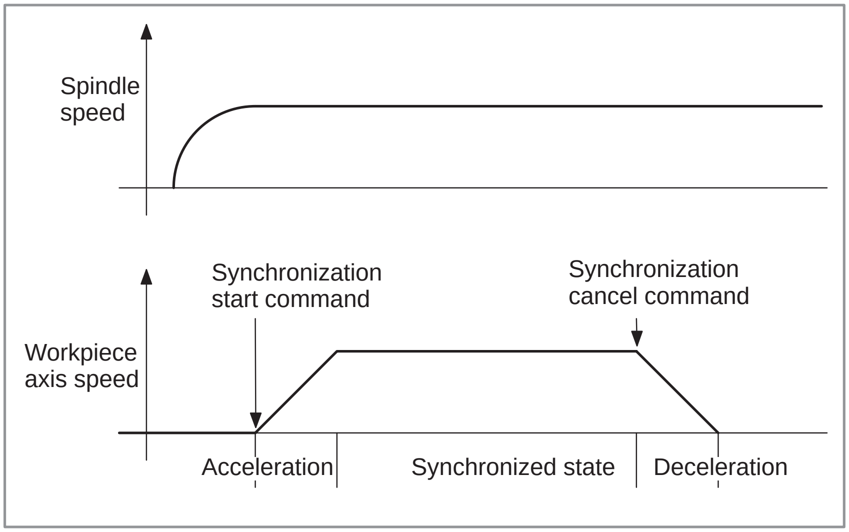
This function provides a skip or high–speed skip signal for an electronic gearbox (EGB) axis in synchronization mode under the control of the EGB function. Its main features are as follows:

1. If a skip signal is input while an EGB axis skip command block is being executed, the block will not end until the skip signal has been input a specified number of times.
2. Movement based on EGB is not stopped by a skip signal.
3. The machine coordinates when a skip signal was input, and the number of times that the skip signal has been input, are stored into specified custom macro variables.

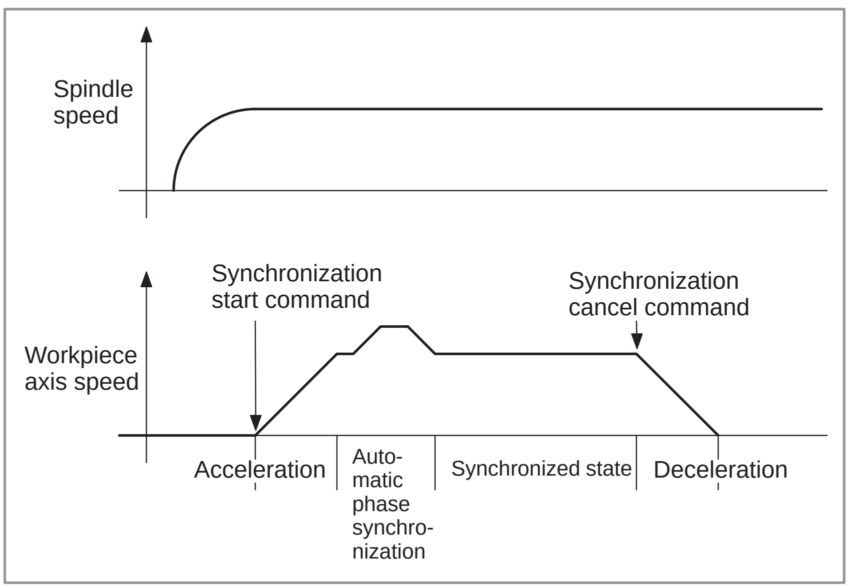
20.19 ELECTRONIC GEARBOX AUTOMATIC PHASE SYNCHRONIZATION

When a request is made to start or cancel synchronization, acceleration/deceleration can be performed before executing the request. Synchronization can be started or canceled while the spindle is rotating. When synchronization is applied, automatic phase synchronization can be performed so that the position of the C–axis when synchronization was started coincides with the position of the spindle one–rotation signal. This enables an operation similar to the one–rotation signal–based start operation of hob synchronization for the conventional hobbing machine functions.

(1) Acceleration/deceleration type



(2) Acceleration/deceleration and automatic phase matching type



21 AUTOMATIC OPERATION

21.1 OPERATION MODE

21.1.1 Tape Operation

The part program can be read and executed block by block from the control unit integral type tape reader or from the input device connected to the reader/puncher interface.

21.1.2 Memory Operation

Program registered in the memory can be executed.

21.1.3 MDI Operation

Multiple blocks can be input and executed by the CRT/MDI unit.

21.2 SELECTION OF EXECUTION PROGRAMS

21.2.1 Program Number Search

Program number currently in need can be searched from the programs registered in memory operating the CRT/MDI.

21.2.2 Program Search with Program Names

The name of a program among the programs registered in memory can be specified from the CRT/MDI to search and select the program.

21.2.3 Sequence Number Search

The sequence number of the currently selected program can be searched using the CRT/MDI unit.
When executing the program from half-way (not from the head) of the program, specify the sequence number of the half-way program, and the program can be executed from the half-way block by sequence number search.
The sequence number search function can be done on memory operation programs, and tape operation programs.

21.2.4 Rewind

After program execution has ended, the program in the memory or the tape reader can be rewinded to the program head, with this reset & rewind signal on.

21.3 ACTIVATION OF AUTOMATIC OPERATION

21.3.1 Cycle Start

Set operation mode to memory operation, MDI operation, or tape operation, press the cycle start button, and automatic operation starts.

21.4 EXECUTION OF AUTOMATIC OPERATION

21.4.1 Buffer Register

Buffer register equivalent to one block is available for program read and control of NC command operation intervals caused by preprocess time. The buffer register can be made for two blocks by selecting parameters.

21.5 AUTOMATIC OPERATION STOP

21.5.1 Program Stop (M00, M01)

Automatic operation is stopped after executing the M00 (program stop) commanded block. When the optional stop switch on the operator's panel is turned on, the M01 (optional stop) commanded block is executed and the automatic operation stops.
The automatic operation can be restarted by the cycle start button.

21.5.2 Program End (M02, M30)

The NC is reset after executing the M02 (end of program) or M30 (end of tape) commanded block.

21.5.3 Sequence Number Comparison and Stop

During program operation, when the block with a preset sequence number appears, operation stops after execution of the block, to a single block stop status. The sequence number can be set by the operator through the CRT/MDI panel.
This function is useful for program check, etc., because program can be stopped at optional block without changing the program.

21.5.4 Feed Hold

The NC can be brought to an automatic operation hold status by pressing the feed hold button on the operator's panel. When feed hold is commanded during motion, it decelerates to a stop.
Automatic operation can be restarted by the cycle start button.

21.5.5 Reset

The automatic operation can be ended in a reset status by the reset button on the CRT/MDI panel or by the external reset signal, etc. When reset is commanded during motion, it decelerates to a stop.

21.6 RESTART OF AUTOMATIC OPERATION

21.6.1 Program Restart

This function allows program restart by specifying the desired sequence number, for example after tool break and change, or when machining is restarted after holidays. The NC memorizes the modal status from the beginning of the program to the sequence number.
If there are M codes necessary to be output, output the M code by the MDI, press the start button, the tool automatically moves to the start position, and the program execution restarts.
CNC counts the number of blocks from the beginning of the program and displays it on the CRT screen. This number of blocks also includes the blocks made by CNC (such as the block to perform the operation of each fixed cycle).
Specifying the block counter value enables restart of the operation from the block with no sequence number or from the midpoint of the cycle operation.

21.6.2 Program Reset Function and Output of M, S, T, and B, Codes

The program restart function enables the following operations after searching for the block to be restarted:

(1) Before moving the tool to the machining restart position

(a) The program restart function automatically outputs the last M, S, T, and B codes to the PMC.
If the last S code is the S code (maximum spindle speed) specified in the block containing G92, the program restart function outputs this S code as the maximum spindle speed signal (MR0 to MR15).
If the S code is the other S code (specified spindle speed), it is output as the specified spindle speed signal (R0 to R15).
Only the S code specified last is displayed on the program restart screen regardless of whether it is in the same block as a G92.

(b) While searching for the block to be restarted, the program restart function automatically outputs all the sampled M codes and the last S, T, and B codes to the PMC. The function can sample up to 35 M codes. When the number of M codes sampled exceeds 35, the function outputs the latest 35 M codes to the PMC.

Specify whether the function performs operation (a) or (b) with the MOAL bot (bit 6 of parameter 7620).

(2) Before the machining restart position is reached.
On the program restart screen, M, S, and B codes can be specified from the MDI for output to the PMC while the system is still in the MEM or TAPE mode.

21.6.3 Restart of Block

Machining can be stopped half–way a block by feed hold, when for example tool breaks. The tool is then taken away from the workpiece for tool change, offsets of the new tool is set, and machining with the new tool is restarted from the point where machining was interrupted.

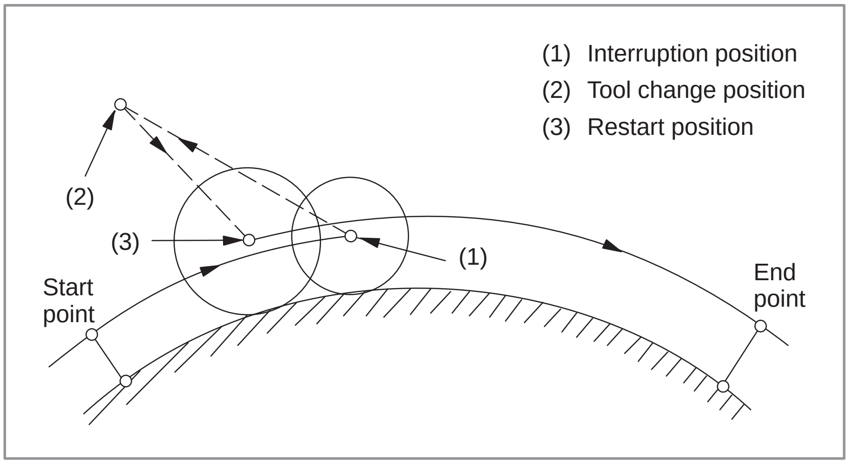


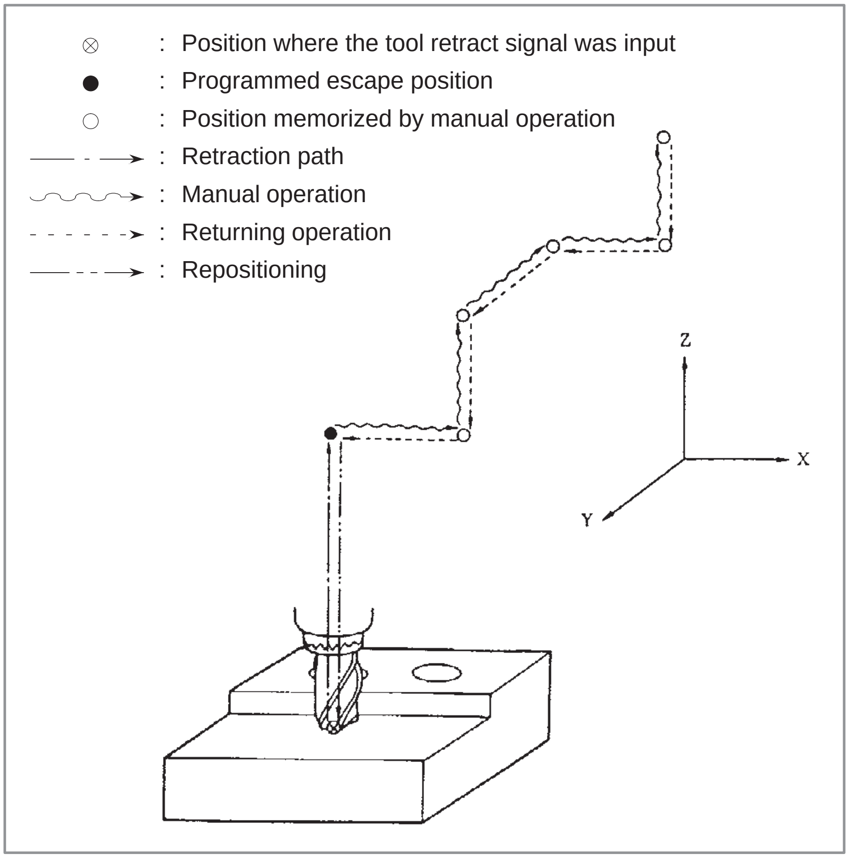
Fig. 21.6.3

21.6.4 Tool Retract & Recover

These functions are used for replacing tools damaged retraction of tools for confirming the cutting conditions, and recovering the tools efficiently to restart the cutting.

Also, the escape operation can be performed with the tool retract signal by previously setting the escape amount (position) with a program. This can be used for retraction for detecting tool damage.

1) Input the tool retract signal during executing the automatic operation. Then, the escape operation (retraction) is performed to the escape position commanded by the program.
2) Input the tool retract signal to initiate the retract mode.
3) After that, switch the automatic mode to the manual mode to move tools with manual operation such as the jog feed, incremental feed, handle feed, and manual numeric command. A maximum of 10 points can be automatically memorized as travel path.
4) Input the tool recovery signal to return the tool to the retraction position in the opposite direction along the path moved by manual operation automatically (recovery operation).
5) Perform the cycle start to return the tool to the position where the tool retract signal was entered (repositioning).



Command the escape amount using the G10.6.

G10.6 IP_ _ ; ;

The escape data sorted by G10.6 is valid until the next G10.6 is commanded. Command the following to cancel the escape amount:

G10.6; (Signal command)
where
The G10.6 is the one-shot G code.

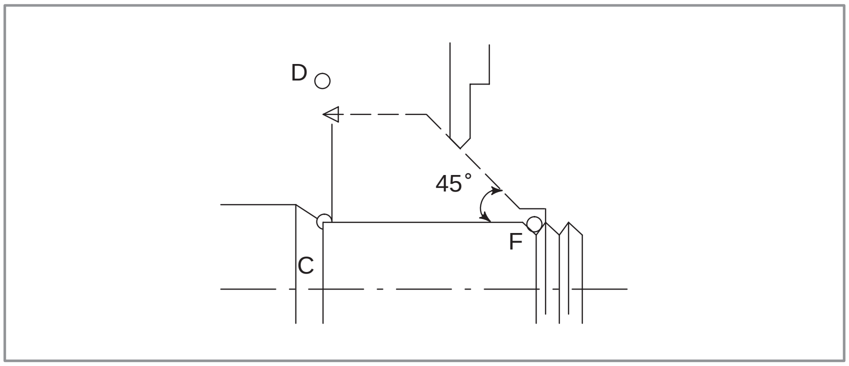
The tool can be retracted to a special location of workpiece coordinate system when the escape amount is command by the ABSOLUTE (G90). When the escape amount is commanded by the INCREMENTAL (G91), the tool can retract by only the commanded escape amount.

Also, it can always be regarded as the incremental command regardless of the Absolute/Incremental commands (G90/G91) by parameter setting.

1) Thread cutting and retract

   The chamfering direction and distance are to be commanded as escape amount during thread cutting.

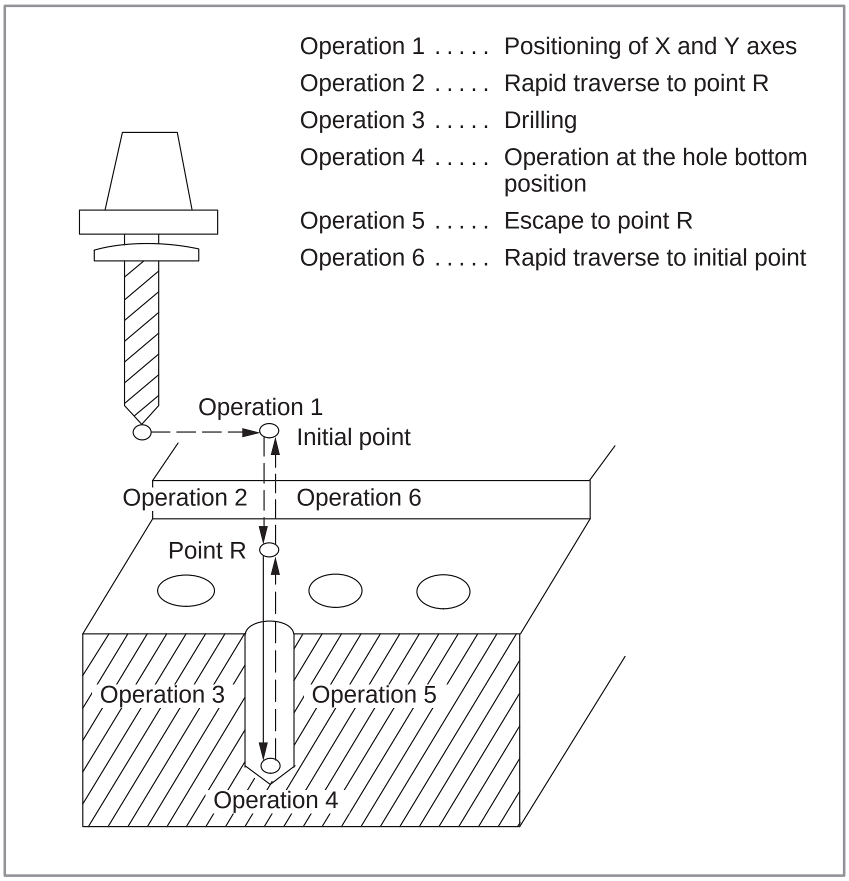
   When the retract signal is input, chamfering is performed in the commanded direction of 45 degrees by the commanded distance. After chamfering is completed, thread cutting continues and stops when it is completed.



The operation after stop is the same as that of normal retract.

2) Command cycle and retract

The following tool retract is performed during the canned cycle for drilling (canned cycle) :



The following retract is performed by inputting retract signal during the canned cycle:

a) During operation 1
Retract is executed in the similar manner as in the normal retract function. (Traveling is carried out by the escape amount (position) set by the G10.6.)

b) During operation 2
The operation 2 is suspended, travelling to the initial point and then it stops.

c) During operation 3
The operation 3 is suspended, the remaining cycle operations d), e), and f) are executed, travelling to the initial point is made, and then it stops.

d) During operation 4, 5, or 6
The operation 4, 5, or 6 continues and then it stops after travelling to the initial point.

The travelling by G10.6 is not performed even if the retract signal is input between the cases b) to d) above.
Also, the retract mode is initiated after travelling to the initial point.

21.7 MANUAL INTERRUPTION DURING AUTOMATIC OPERATION

21.7.1 Handle Interruption

During automatic operation, tool can be adjusted by the manual pulse generator without changing the mode. The pulse from the manual pulse generator is added to the automatic operation command and the tool is moved for the recommended pulses.
The workpiece coordinate system thereafter is shifted for the pulse commanded value. Movement commanded by handle interruption can be displayed on the CRT screen.

21.7.2 Automatic/Manual Simultaneous Operation

When auto/manual simultaneous operation selection signal is set on, automatic operation (tape, MDI, memory, or tape editing) and manual operation (manual feed, incremental feed, or manual handle feed) are simultaneously performed.
This function allows, for example, staging of the next workpiece during automatic operation.

21.8 RETRACE

By turning on retract signal, it is possible to retract the tool path which so far has been passed. By turning off trace signal, it is possible to advance along the retraced path. When the path up to the position where retrace was started is traced again, cutting continues according to program commands.

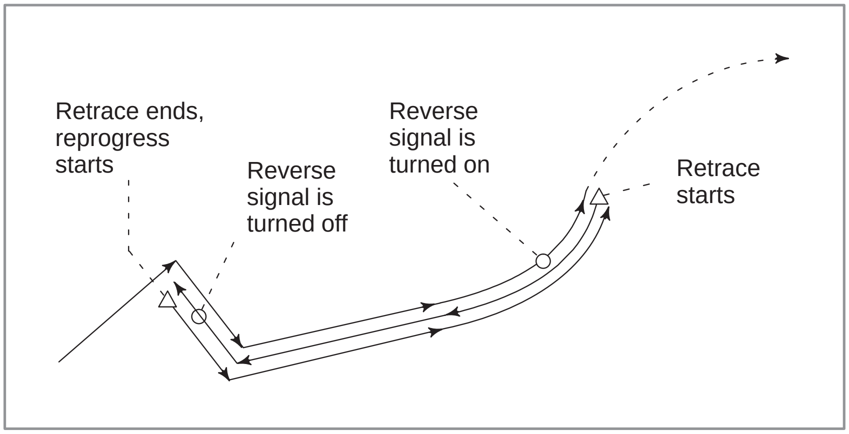


Fig. 21.8 Retrace and reprogress

When retrace signal is turned on and the path so far passed is traced back, it is called “retrace”. When retrace signal is turned off and the retraced path is progressed again up to the point where retrace was started, it is called “readvance”.

Approximately 40 to 80 blocks which were previously executed in the automatic operation mode such as memory, tape, and MDI operations can be retraces.

A block created inside the CNC is also counted as one block on retracing.

When all 40 to 80 blocks are retraced or there are no blocks to be retraced, the tool retraces the last block, and the tool stops.

It is possible to select whether the feedrate on retracing is set to the commanded rate or to the rate set by a parameter.

NOTE

1 When the following functions are added, no reverse function can be mounted:
   - ○ FS3/6 interface
   - ○ Interrupt–type custom macro
   - ○ Multi–buffer

2 The blocks including the following commands cannot be retraced:
   - ○ Inch/metric conversion
   - ○ Reference position return function
   - ○ Thread cutting
   - ○ Remote buffer

3 No retrace can be performed during execution of the following functions:
   - ○ Circular thread cutting B
   - ○ Polar coordinate interpolation
   - ○ Cylindrical interpolation
   - ○ High–speed cutting
   - ○ Exponential function interpolation

4 The M, S, T and the second auxiliary functions are also output during retrace.
   When these functions are executed, some sort of countermeasures are required at the machine side.

21.9 ACTIVE BLOCK CANCEL

Automatic operation can be stopped by inputting signal BCAN to the control unit. After the automatic operation enters the STOP state, output signals STL and OP go low. All modal data is maintained.
Automatic operation is restarted the block after stopped block by cycle start.

21.10 TRANSVERSE INHIBIT LIMIT FUNCTION

If an absolute coordinate value exceeds a transverse inhibit limit value (specified in setting parameter No. 5251) during automatic operation, the movement of the axis is stopped, but automatic operation continues, and the absolute coordinate value is updated. In other words, the machine behaves as if a machine lock were in effect. When the absolute coordinate value returns to within the transverse inhibit limit, movement of the axis is resumed. Also, during manual operation, if an absolute coordinate value exceeds a transverse inhibit limit value, movement of the axis is stopped. In other words, the machine behaves as if a machine lock were in effect. When the absolute coordinate value returns to within the transverse inhibit limit, movement of the axis is resumed.

22 MANUAL OPERATION

22.1 MANUAL FEED

1) 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 - 655.34%, 0.01% step.
   The parameter set speed can be set to each axis.
2) Manual rapid feed
   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.

22.2 INCREMENTAL FEED

Specified move amount can be positioned to the + or - direction with the button. Move amount of:
0 – (least command increment) $\times$ 99999999
can be specified. The feed rate is that of manual feed.
It is also possible to specify (least command increment) $\times$ (magnification) (not optional move amount) by selecting parameters:
The possible magnifications to be specified are as follows.
$\times$1, $\times$10, $\times$100, $\times$1000 $\times$10000, $\times$100000.

Table 22.2

Increment system	Metric input	Inch input
IS–A	0.01, 0.1, 1.0, 10.0, 100.0, 1000.0 mm	0.001, 0.01, 0.1, 1.0, 10.0, 100.0 inch
IS–B	0.001, 0.01, 0.1, 1.0, 10.0, 100.0 mm	0.0001, 0.001, 0.01, 0.1, 1.0, 10.0 inch
IS–C	0.0001, 0.001, 0.01, 0.1, 1.0, 10.0 mm	0.00001, 0.0001, 0.001, 0.01, 0.1, 1.0 inch
IS–D	0.00001, 0.0001, 0.001, 0.01, 0.1, 1.0 mm	0.000001, 0.00001, 0.0001, 0.001,,0.01, 0.1 inch

22.3 MANUAL HANDLE FEED (1ST)

By rotating the manual pulse generator, the axis can be moved for the equivalent distance. Manual handle feed is controlled 1 axis at a time. The manual pulse generator generates 100 pulses per rotation. Move amount per pulse can be specified from the following magnifications:
$\times$1, $\times$10, $\times$M.
M is parameter set value of 1 - 1000. Move distance is :
(Least command increment) $\times$ (magnification)

Table 22.3

Increment system	Metric input	Inch input
IS–A	0.01, 0.1, M/100 mm	0.001, 0.01, M/1000 inch
IS–B	0.001, 0.01,M/1000 mm	0.0001, 0.01, M/10000 inch
IS–C	0.0001, 0.001, M/10000 mm	0.00001, 0.0001, M/100000 inch
IS–D	0.00001, 0.0001, M/100000 mm	0.000001, 0.00001, M/100000 inch

It is also possible to specify the following magnifications by selecting parameter.
$\times$1, $\times$10, $\times$100, $\times$M

22.4 MANUAL HANDLE FEED (2ND, 3RD)

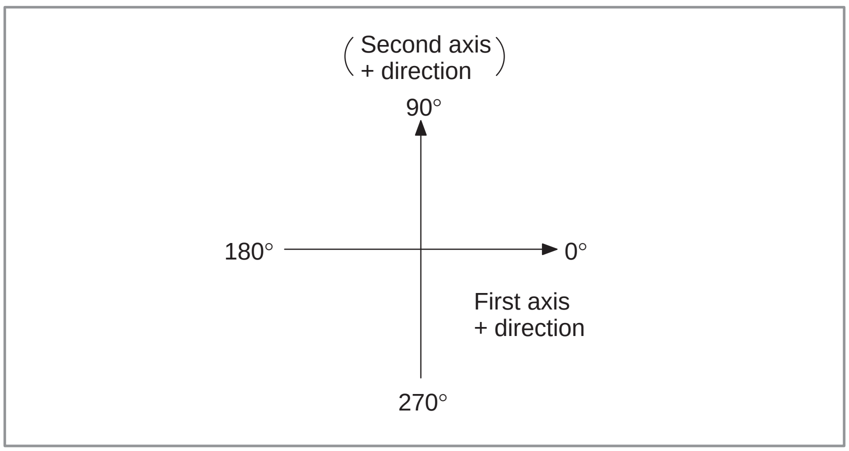
A 2nd, as well as 3rd manual pulse generator can be rotated to move the axis for the equivalent distance. Manual handle feed of 3 axes can be done at a time. Multiplier is common to 1st, 2nd and 3rd manual pulse generators.

22.5 MANUAL ARBITRARY ANGLE FEED

The tool can be moved to an optional direction on an optional plane by manual operation. Simple plane cutting can be performed with this function because feedrate, feed direction, feed plane can always be changed.

1) Plane selection
   Specify 1st and 2nd axis of the plane to perform manual optional angle feed by external signals.
2) Feed direction assignment
   Specify feed direction of manual arbitrary angle feed by external signals. Every 1/16 degrees between 0° – 360° can be specified.
   The angles are as follows.

   Angles of 360° or more can be specified, for the NC can convert the angle. Feed direction can be optionally changed during manual arbitrary angle feed. Inposition check will not be performed even when feed direction changes, and move command to the new direction is immediately executed.



3) Feedrate assignment
   Feedrate of manual arbitrary angle feed (tangential direction speed) is specified with the jogging speed set dial. Feedrate can be freely changed during manual arbitrary angle feed.
4) Manual arbitrary angle feed start, stop command
   Manual arbitrary angle feed is performed while manual arbitrary angle feed signals is on.
   Manual arbitrary angle feed signal for forward direction feed and reverse direction (180° reversed direction) feed are available.

22.6 MANUAL NUMERIC COMMAND

Program format data commanded via the MDI can be executed in JOG feed mode.
Manual numerical command can be executed any time the JOG feed is available.
The following commands can be executed:

- – Positioning (G00)
- – Linear interpolation (G01)
- – Automatic reference position return (G28)
- – 2nd/3rd/4th reference position return (G30)
- – M, S, T, B (2nd miscellaneous function)

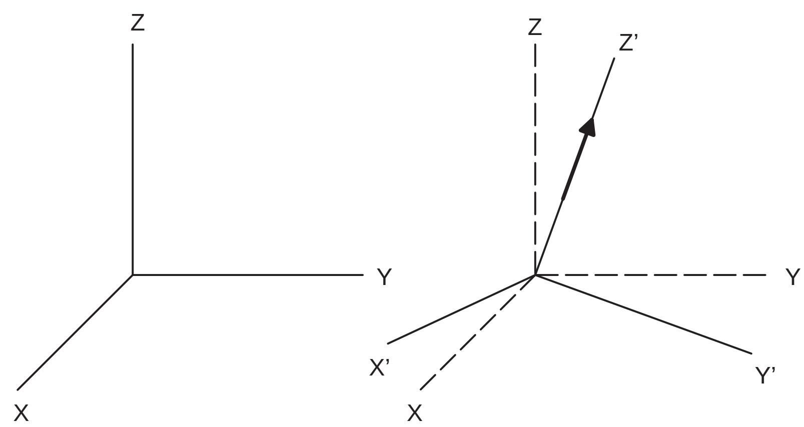
Activation of the commanded data is the same as cycle start in automatic operation.
When feed hold is commanded during manual numeric command execution, the move command will stop but execution will continue till the M, S, T, B (2nd miscellaneous function) ends.
Move command once stopped cannot be continued.

22.7 MANUAL ABSOLUTE ON/OFF

When tool is moved by manual operation when input signal ABS is on, the move distance is added to the absolute coordinate value.
When tool is moved by manual operation when input signal ABS is off, the move distance is ignored, and is not added to the absolute coordinate value. In this case, the workpiece coordinates is shifted for the amount that the tool was move by manual operation.

22.8 MANUAL INTERRUPTION FUNCTION FOR THREE–DIMENSIONAL COORDINATE SYSTEM CONVERSION

When the handle of the manual pulse generator is rotated in the three–dimensional coordinate system conversion mode, this function adds the travel distance specified by the manual pulse generator to the travel distance during automatic operation.



Coordinate system to be converted

Converted coordinate system (When the Z–axis is selected, the travel distance is added to the Z'–axis as shown in the figure above.)

22.9 STORED STROKE LIMIT CHECK IN MANUAL OPERATION

If a request is made to move an axis beyond stored stroke limit 1 during a manual operation (manual rapid traverse, jog feed, handle feed, or incremental feed), the request is rejected, and a warning message, rather than an alarm, is displayed. The warning message appears just before the axis enters the forbidden area set for stored stroke limit 1, and is cleared automatically when the axis starts moving in the other direction.

23 PROGRAM TEST FUNCTIONS

23.1 ALL AXES MACHINE LOCK

In machine lock condition, the machine does not move, but the position display is updated as if the machine were moving. Machine lock is valid even in the middle of a block.

23.2 MACHINE LOCK ON EACH AXIS (Z AXIS COMMAND CANCEL)

Machine lock can be commanded per axis.
Not only the Z axis, but also optional axis command can be cancelled.

23.3 AUXILIARY FUNCTION LOCK

This function inhibits transmitting of M, S, T, B function code signals and strode signals to the machine side. The decoded DM00, DM01, DM02, and DM30 signals can be transmitted under this miscellaneous function lock.

23.4 DRY RUN

In this dry run mode, commanded cutting feedrate is ignored and axis is fed at feedrate specified with the jogging rate dial. Rapid feed command (G00) is done in rapid feedrate, and rapid traverse override is valid.
Dry run can also be commanded to rapid feed command (G00) by parameter setting.

23.5 SINGLE BLOCK

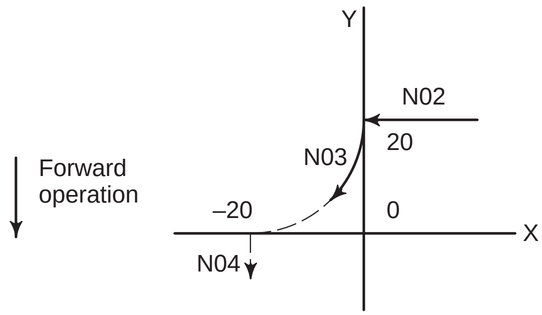
The program can be executed block by block under automatic operation.

23.6 RETRACE PROGRAM EDITING FUNCTION

This function checks a machining program while executing it in the test mode. If it finds an error in the program, the error can be corrected the error at that time and the corrected program can be executed immediately. It can be used to check and correct a machining program without stopping operation.

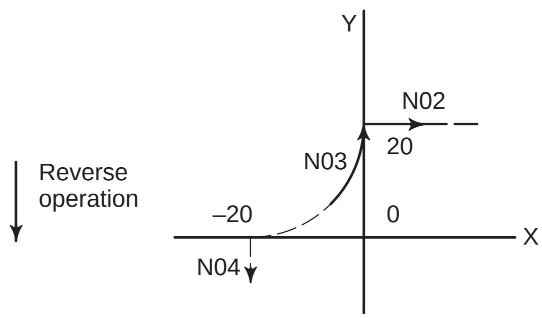
1. While the tool was moving according to the N03 block, an error was found in the tool path.

```
PROGRAM(MEMORY)

N01 G90 G00 X20. Y20. ;
N02 G01 X0. Y20. F500 ;
N03 G02 X–20. Y0. R20. ;
N04 G01 X–20. Y–10. ;
```



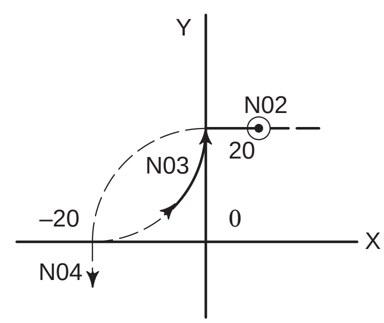
2. The tool is stopped and reverse operation is started immediately.

```
PROGRAM(MEMORY)

N01 G90 G00 X20. Y20. ;
N02 G01 X0. Y20. F500 ;
N03 G02 X–20. Y0. R20. ;
N04 G01 X–20. Y–10. ;
```



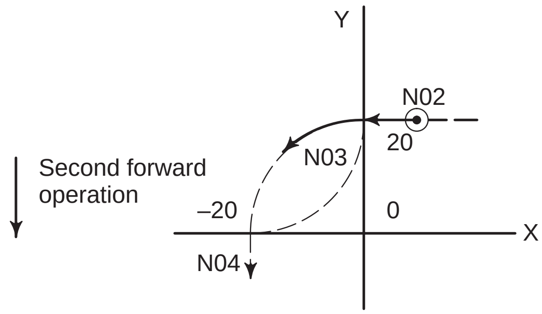
3. After the N03 block to be corrected is executed in reverse, the tool is stopped and the program is edited.

```
PROGRAM(MEMORY)

N01 G90 G00 X20. Y20. ;
N02 G01 X0. Y20. F500 ;
N03 G03 X–20. Y0. R20. ;
N04 G01 X–20. Y–10. ;
```



4. When operation is restarted, the corrected program is executed.

```
PROGRAM(MEMORY)

N01 G90 G00 X20. Y20. ;
N02 G01 X0. Y20. F500 ;
N03 G02 X–20. Y0. R20. ;
N04 G01 X–20. Y–10. ;
```



24 SETTING AND DISPLAY UNIT

24.1 SETTING AND DISPLAY UNIT

The following Setting and Display units are available.

9″ Monochrome CRT/MDI (Small Type)

9″ Monochrome CRT/MDI (Standard Type)

9″ Monochrome PDP/MDI (Standard Type)

9″ Monochrome CRT (Separate Type)

9″ Monochrome PDP (Separate Type)

9.5″ Color LCD/MDI (Horizontal Type)

9.5″ Color LCD/MDI (Vertical Type)

14″ Color CRT/MDI (Horizontal Type)

14″ Color CRT/MDI (Vertical Type)

10.4″ Color LCD (Separate Type)

Separate Type MDI for 9″ CRT/PDP

Separate Type MDI for Intelligent Terminal

Separate Type MDI for 10.4″ LCD (Vertical Type)

Separate Type MDI for 10.4″ LCD (Horizontal Type)

NOTE

The examples in this chapter use the 9″ CRT/MDI unit (standard type) for displaying set values.

24.2 EXPLANATION OF THE KEYBOARD

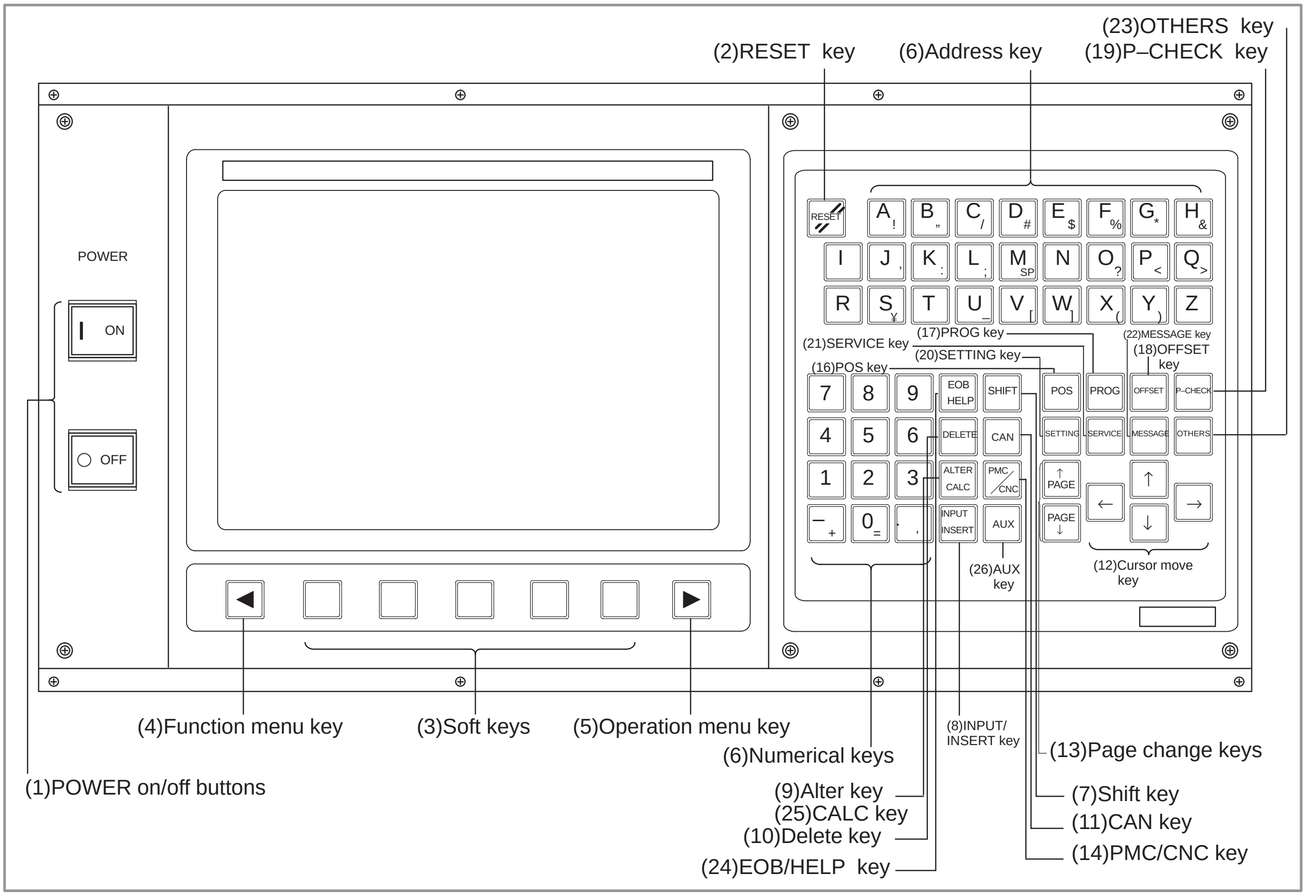


Fig. 24.2 (a) 9″ CRT/MDI unit

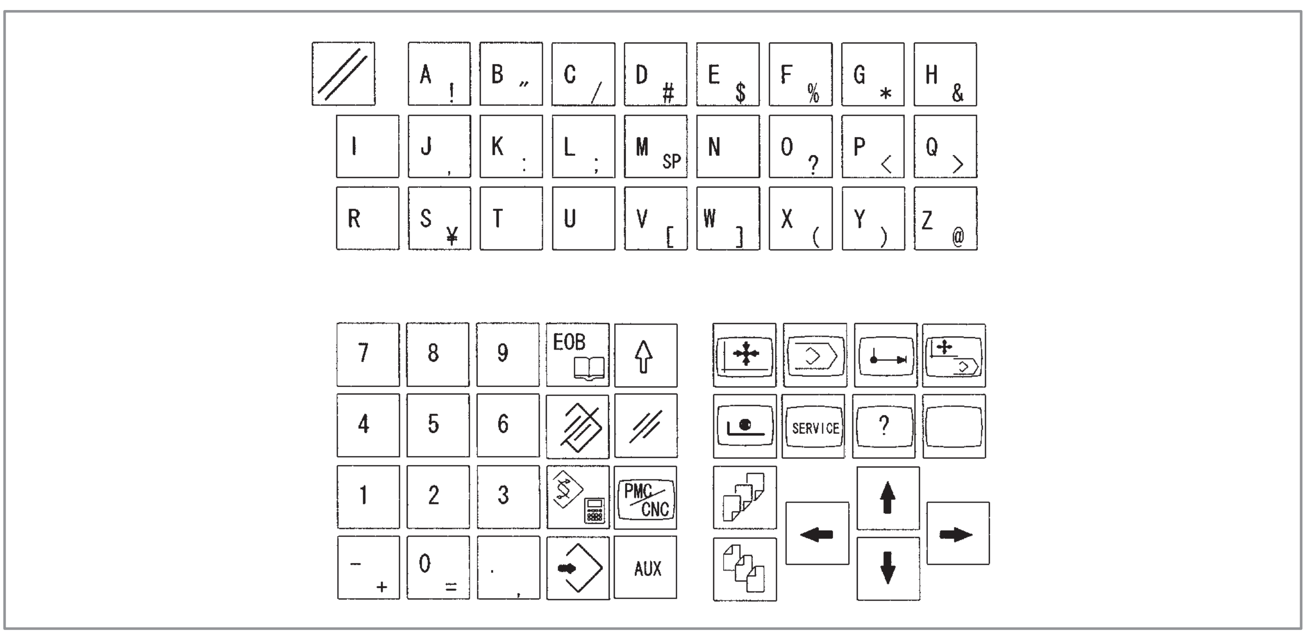


Fig. 24.2 (b) Key arrangement (with pictorial indications)

Table 24.2 MDI Keyboard functions (1/3)

No.	Name	Functions
(1)	<Power> ON/OFF button	Press this button to turn CNC power ON and OFF.
(2)	<RESET> key	Press this key to reset the CNC, to cancel an alarm, etc.
(3)	Soft key	The soft key has various functions, according to the Applications. The soft key functions are displayed at the bottom of the CRT screen.
(4)	Function menu key	Pressing this key when the soft keys are not function selection keys returns the soft keys to the states of the function selection keys. Pressing the key when the soft keys are function selection keys changes the soft keys to the function selection keys that do not fit on the screen. (The 9″ CRT/MDI panel has five soft keys. The 14″ CRT/MDI panel has ten soft keys. However, these soft keys are not sufficient for some applications. In this case, a plus sign (+) is displayed at the extreme right of the bottom line on the CRT. The plus sign indicates that some soft keys do not fit on the screen.)
(5)	Operation menu key	The functions of the soft keys vary according to the applications. Pressing this key when the soft keys are not operation selection keys changes the soft keys to the operation selection keys that are effective on the selected CRT screen. Pressing this key when the soft keys are operation selection keys changes the soft keys to the operation selection keys that do not fit on the screen. (The 9″ CRT/MDI panel has five soft keys. The 14″ CRT/MDI panel has ten soft keys. However, these soft keys are not sufficient for some applications. In this case, a plus sign (+) is displayed in the rightmost frame of the bottom line on the CRT. The plus sign indicates that some soft keys do not fit on the screen.)
(6)	Address/numerical key	Press these keys to input alphabetic, numeric, and other characters.
(7)	<SHIFT> key	Some address keys are marked with two characters. To enter the lower right character, press the shift key first. When the shift key is pressed, ^ is displayed in the key input buffer. This indicates that pressing the address key enters the lower right character. !, ”, $, \, –, <, >, :, ;, %, ’ These characters can be used on the MMC screen.
(8)	<INPUT/INSERT> key	When an address or numeric key is pressed, the data is entered in the key input buffer, then displayed on the CRT. Press the input key to store the data entered in the key input buffer in the offset register. The input key is equivalent to an <INPUT> soft key. Either may be used. Pressing this key on the program editing screen inserts the contents of the key input buffer after the position where the cursor is located. The INSERT soft key has the equivalent function. Either key can be used.
(9)	<ALTER>key	Pressing this key on the program editing screen replaces the word where the cursor is located with the contents of the key input buffer. The ALTER soft key has the equivalent function. Either key can be used.
(10)	<DELETE>key	Pressing this key on the program editing screen deletes the word where the cursor is located. The DELETE WORD soft key has the equivalent function. Either key can be used.
(11)	Cancel <CAN> key	Pressing this key deletes a character or symbol input to the key input buffer. The contents of the buffer are displayed on the CRT. The position where a new entry is to be input is displayed with an underscore (_). Pressing the CAN (cancel) key deletes the character immediately before the underscore. Example) When the contents of the key input buffer are displayed as shown below, >N001X100Z_ pressing the CAN key deletes Z and the displayed contents change as follows: >N001Z100_

Table 24.2 MDI Keyboard functions (2/3)

No.	Name	Functions
(12)	Cursor move keys	The following four cursor keys are provided: → :This key moves the cursor forward in small increments. The cursor is moved in the direction of order. ← :This key moves the cursor backward in small increments. The cursor is moved to the opposite direction. ↓ : This key moves the cursor for the key input buffer forward when a character is entered in the key input buffer. The position of the cursor for the key input buffer is indicated by an underscore. Pressing an address or numeric key enters an address or number at the cursor position. Pressing the cancel <CAN> key deletes the character before the cursor position. This key moves the cursor on the CRT screen forward in large increments when no data is entered in the key input buffer. The cursor is moved in the direction of order. ↑ : This cursor key moves the cursor for the key input buffer backward. This key moves the cursor on the CRT screen backward in large increments when no data is entered in the key input buffer. The cursor is moved to the opposite direction.
(13)	Page change keys	Two kinds of page change keys are described below. < ↓ >: This key is used to changeover the page on the CRT screen in the forward direction. < ↑ >: This key is used to changeover the page on the CRT screen in the reverse direction. The small 9″ monochrome CRT/MDI panel is not provided with page keys. Pressing the → and ↓ cursor keys simultaneously is equivalent to pressing the ↓ page key. Pressing the ← and ↑ cursor keys simultaneously is equivalent to pressing the ↑ page key.
(14)	<PMC/CNC>switch key	This key is used to determine whether the CRT/MDI panel is used for the CNC or PMC.
(15)	<MMC>key	Pressing this key enables the CRT/MDI to be used in the MMC. This key is valid only when a CNC having the MMC is used.
(16)	<POS>key	Pressing this key selects the current position display screen. The POSITION soft key has the equivalent function. Either key can be used.
(17)	<PROG>key	Pressing this key selects the part program display screen. The PROGRAM soft key has the equivalent function. Either key can be used.
(18)	<OFFSET>key	Pressing this key selects the tool offset display screen or the screen displaying offset from the workpiece reference position. The OFFSET soft key has the equivalent function. Either key can be used.
(19)	<P–CHECK>key	Pressing this key selects the program check screen. The P CHECK soft key has the equivalent function. Either key can be used.
(20)	<SETTING>key	Pressing this key selects the setting screen. The SETTING soft key has the equivalent function. Either key can be used.
(21)	<SERVICE>key	Pressing this key selects the parameter and diagnosis screen. The MAINTENANCE soft key has the equivalent function. Either key can be used.
(22)	<MESSAGE>key	Pressing this key selects the screen for alarm messages and operator messages. The MESSAGE soft key has the equivalent function. Either key can be used.
(23)	<OTHERS>key	Pressing this key selects and displays the screen specified with parameter No. 2215.

Table 24.2 MDI Keyboard functions (3/3)

No.	Name	Functions
(24)	<HELP>key (SHIFT/EOB)	Pressing this key displays the help window on a screen. Alarm help, soft key help, and G code guide can be displayed. For details, see Help Functions.
(25)	Arithmetic <CALC> key <SHIFT>+<ALTER>	Press this key to execute operation commands in the key input buffer. Example) When the data in the key input buffer is [10 + 20 x 30 + 400/8], the arithmetic key is pressed. Then, the data in the key input buffer is changed to 660. The arithmetic key is standard.
(26)	<AUX> key	Auxiliary key

24.3 SOFT KEYS AND CALCULATION KEYS

Soft keys

The 14–inch CRT has 10+2 keys (10 soft keys and a "Function Menu" key and a "Operation Menu" key at both sides of the 10 keys) . The "Function Menu" and "Operation Menu" keys are used to select functions in the soft keys.
The 9–inch CRT has 5+2 soft keys.
These soft keys can be assigned with various functions, according to the needs.

Calculation key (CALC)

Characters input via keys are once input in the key–input buffer and displayed on the lower part of the CRT screen.
When the CALC key is pressed, operation command in the buffer is executed.
The following operator and formula can be used.

(i) Operator

+ (Sum), − (Difference) * (Product), / (Quotient)

(ii) Function

SIN (Sine), COS(cosine), TAN(tangent), ATAN (arc tangent), SQRT (square root), ABS (absolute value), ACOS (arc cosine), ASIN (arc sine), LN (natural logarithm), EXP (exponent)

Example 1

Key–in buffer data

X [100*5 + 200/5]

When the CALC key is pressed, the key–input buffer data becomes as: X540.

Example 2

Key–input buffer data

[10 + 20 * 30 +400/8]

When the CALC key is pressed, the key–input buffer data becomes as: 660

The following functions at are mainly available via the CRT/MDI panel:

1) Actual position display and actual position presetting
2) Contents of program display, program directory display (display of program number, program name, part program storage length left, number of programs left)
3) Program editing
4) Offset amount display and setting
5) Commanded value display, MDI input
6) Parameter setting and display
7) Alarm message/operator message display
8) Custom macro variables display and setting
9) Tool life management data display and setting
10) Diagnosis
11) Others

24.4 MANUAL DATA INPUT (MDI)

The following data can be input via the MDI panel.

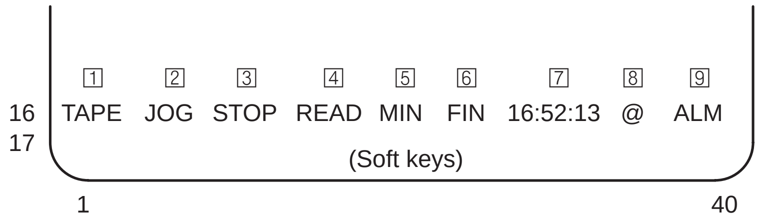
a) Program input (multiple–block command is possible)
b) Setting data input (for tool compensation data, etc.)
c) Parameter input (rapid traverse rate, acceleration/deceleration time constants, etc.)
d) Diagnosis data input
e) Tape storage and editing operation
f) Other operations

24.5 DISPLAY

The following data are displayed on the CRT screen. One 9″ CRT screen can display maximum 680 characters (40 × 17 lines) and one 14″ CRT screen can display maximum 1998 (74 × 27 lines).

1) Status display

Status of the control unit (alarm, editing) is displayed.
Status display can be seen one line above the soft key display on the CRT screen.



The above figure is the 9″ CRT. The same display is done on the 14″ CRT (second line from the bottom, from the right).
The display has ten fields from $\boxed{1}$ to $\boxed{9}$, and the following is displayed in all types of FANUC NCs.

$\boxed{1}$ Automatic operation mode selection (MEM, MDI, TAPE, EDIT, or ****)

The currently selected automatic operation mode is displayed. When automatic operation is not selected, the “****” is displayed.

$\boxed{2}$ Manual operation mode selection (JOG, HND, INC, AGJ, J+H, REF, or ***)

The currently selected manual operation mode is displayed. When manual operation is not selected, the “***” is displayed.

$\boxed{3}$ Automatic operation status (RSET, STOP, HOLD, STRT, MSTR, or SRCH)

Displays what status the automatic operation is.

RSET – Resetting
STOP – Automatic operation stop
HOLD – Automatic operation hold
STRT – Automatic operation start
MSTR – Manual numeric command start
SRCH – Sequence number searching

$\boxed{4}$ Program editing status (READ, PNCH, VRFY, SRCH, COND, EDIT, or ****)

Displays what status the program editing is.

READ – Registering
PNCH – Punching
VRFY – Verifying
SRCH – Searching
COND – Arranging memory
EDIT – Other editing operation (INSERT, ALTER, etc.)
**** – No editing done

5 Axis move, dwell status (MTN, DWL, or ****)

"MTN" is displayed when axis in moving, "DWL" when dwelling, and "***" in other cases.

6 M, S, T, B, functions' status (FIN or ****)

When miscellaneous function as M, S, T, B, functions are under execution (waiting for end signal from the PMC), "FIN" is displayed, and "***" in other cases.

5' 6' Emergency stop status (EMG)

When emergency stop status commanded, display of 5 and 6 escapes but "– –EMG– –" is displayed with inverted.

7 Current time display

The current time is displayed in units of hour, minute and second.

Example) 16 : 52 : 13

8 Non–volatile memory write status (@ or space)

"@" is displayed when data is being written in the non–volatile memory for parameter, tool offset and NC part program etc..

9 Alarm or label skip status (ALM, BAT, WRN, LSK or ***)

"ALM (inverted blinking display)" is displayed when an alarm occurs.
"WRN (blinking display)" is displayed when a warning message is issued on the CRT.
"BAT" is displayed when a signal predicting battery down is sent out. After changing the battery and pushing the RESET key, "BAT" is erased.
"LSK" is displayed when tape reader is under label skip status, and when all of alarm warning and battery alarm does not occur.
"***" is displayed in other cases :

2) Key input display

Data input via the address keys or the numerical keys are displayed at the left lower part of the screen.

3) Program number, sequence number display

Program number, sequence number is displayed on the right upper part of the screen.

4) Alarm display

Alarm number and its contents are displayed briefly.

5) Alarm message display

Alarm message contents are displayed.

6) Present position display

Relative position and position in the workpiece coordinates are displayed in 3–times magnified characters.

7) Total position display

Relative position, position in the workpiece coordinates, position in the machine coordinate, and remaining move distance are displayed in one screen.

8) Command value display

The following two displays are performed.

i) Previously commanded modal value (LAST)

ii) Command value to be executed (ACTIVE)

9) Setting (parameter set by the operator) display
   Displays setting value.
10) Tool offset amount display
   Displays offset value. Relative position is also displayed at the same time.
11) Program display
   i) Display of program for editing.
   ii) Display of program currently under execution.
   iii) Display of program list.
   A list of program number, program name, and size of programs stored in the memory is displayed.
   Remaining memory size is also displayed.
12) Parameter display
13) Self diagnosis result display
14) Custom macro variables display
15) Operator message, external operator message, external alarm message display
16) Actual speed display
   i) Actual feedrate per minute (mm/min or inch/min)
   ii) Actual spindle speed (rpm)
17) Program check screen
   The following are displayed on one screen.
   i) Program number on execution
   ii) Sequence number on execution
   iii) Program text on execution
   iv) Current position
   v) Modal G codes
   vi) Modal M codes
   vii) T code
   viii) Actual feedrate and spindle speed
   ix) Status

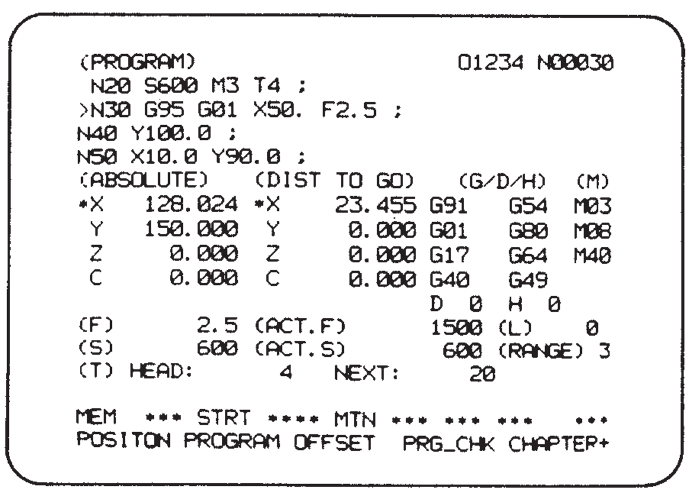
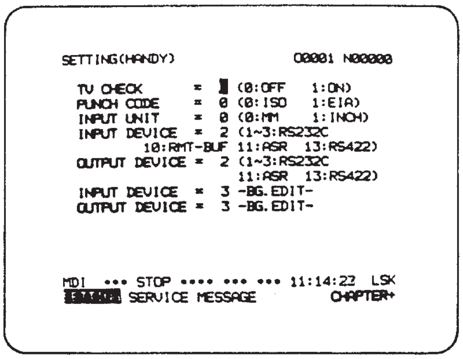


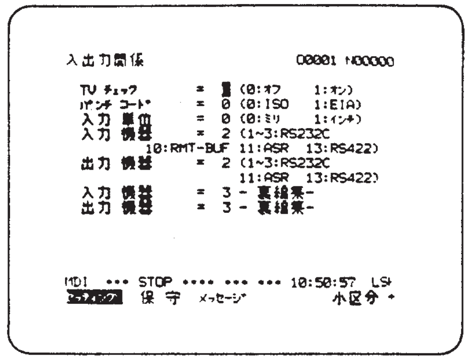
Fig. 24.5

24.6 LANGUAGE SELECTION

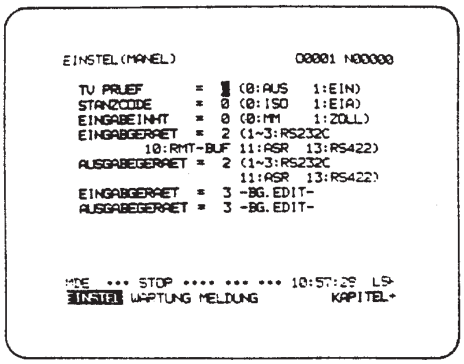
The Japanese, English, German, French, Italian, Spanish and Swedish are prepared as display languages. Select the language to be displayed by parameters.



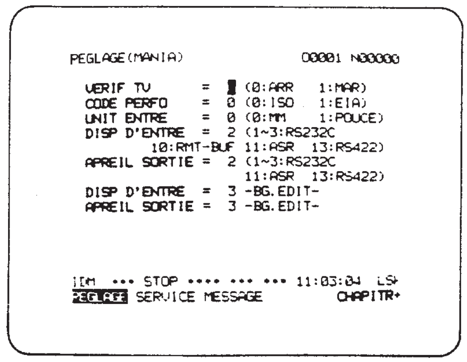
English



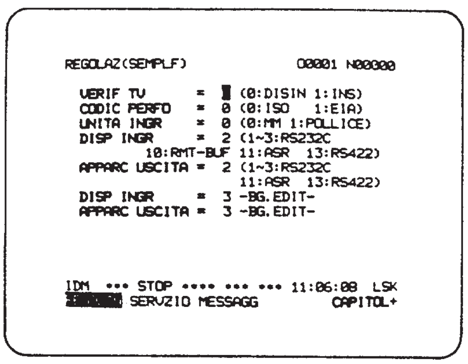
Japanese



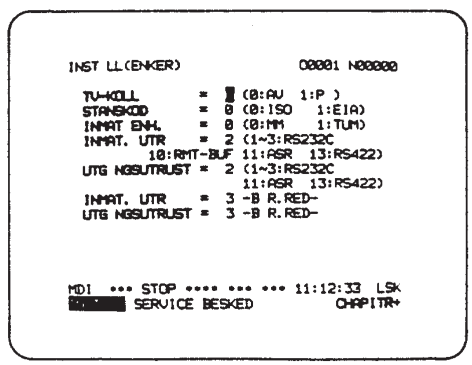
German



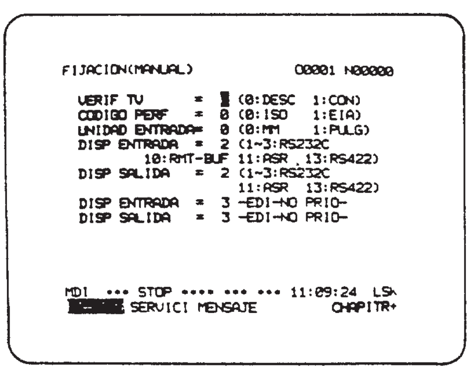
French



Italian



Spanish



Swedish

24.7 CLOCK FUNCTION

Series 15 incorporates a clock to display the time in the hour/minute/second format on each display screen. Some screens allows display of the year, month, and day.

The custom macro system variable can be used to read the time. The time will be told through the window at PMC side.

24.8 RUN HOUR & PARTS NUMBER DISPLAY

This function displays the integrated power–on time, the integrated cycle operation time, the integrated cutting time and timer on the CRT display screen. The integrated cycle operation time, the integrated cutting time and timer can be altered and preset, using the MDI.

In addition to the above, this function displays the count of the total number of parts machined, the number of parts required and the number of parts on the CRT screen. Each time M02, M30 or a parameter set M code is executed, the count of the total in memory is incremented by 1.

If a program is prepared so as to execute M02, M30 or a parameter set M code each time one part machining is completed, the number of parts machined can be counted automatically.

If the count of the number of parts reaches the number of parts required, a signal is output to the PMC side.

It is possible to change and preset the number of parts required and the number of parts counted, using MDI. The number of parts required and the number of parts counted can be read and written, using external data input/output function and custom macro variables.

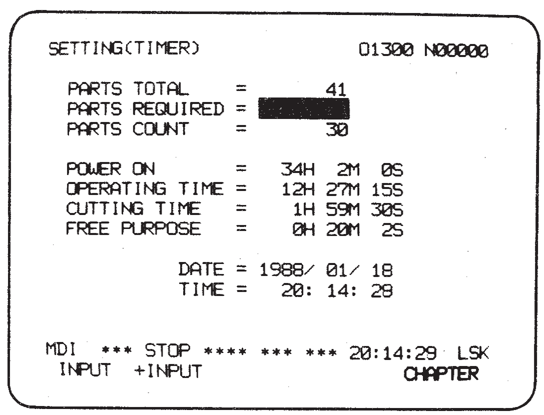


Fig. 24.8

24.9 LOAD METER DISPLAY

The load values (torque values) of spindle motor and servo motor are displayed in bar chart on the CRT. When the 14–inch CRT is used, fluctuation waveform of load value is also graphically displayed.

The most recent sampling values and fluctuation status (for one minute) are displayed in bar chart display and waveform display, respectively. Set the rated load value of motor corresponding to each load meter to parameters. The load meter displays 100% when the load value is the rated load value.

The load meter, position in the workpiece coordinate system, command rate, real rate, override value, number of cutting parts, and run hour are displayed on the load meter display screen.

The load meter can be displayed up to three servo motor axes and a parameter can be used to select any one of three axes.

It is required that the load current of spindle motor should be informed to the A/D converter of CNC to display the load of spindle motor.

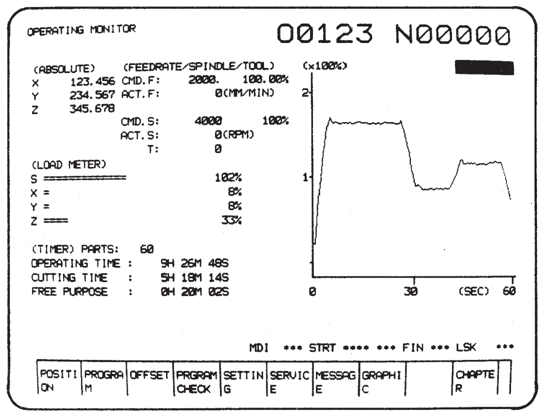


Fig. 24.9 In case of 14″ CRT

24.10 MENU SWITCH

Instead of the switches on the machine operator's panel, on/off commands of the functions will be made possible via setting on the CRT/MDI. This function will vastly decrease number of switches on the machine operator's panel.
On/off commands of the following function are available on the CRT screen.

1) Single block (SBK)
2) Machine lock (MLK)
3) Display lock (DLK)
4) Auxiliary functions lock (AFL)
5) Dry run (DRN)
6) Optional block skip (BDT 1 – 9)
7) Mirror image (MIX, MIY, MIZ, ...)
8) Z axis ignore (ZNG)
9) Absolute switching (ABS)

Signals from the machine side is still valid with this function. When corresponding sign commanded "1", the function will be "on".

24.11 SOFTWARE OPERATOR'S PANEL

In this function, functions of switches on the machine operator's panel is done by operation on the CRT/MDI panel. Mode selection and jogging override, etc. can be operated by setting operation via the CRT/MDI panel with this function, thus allowing commitance of corresponding switches on the machine operator's panel.
This function is valid only when the screen is displayed with operator's panel. Mode cursor with the cursor operation keys, and select various operations, viewing the screen.
The following operations can be done via the CRT/MDI panel :

A Mode selection

B Manual pulse generator feed axis selection
Move distance selection per pulse of manual pulse generator

C Rapid traverse override
Jog feedrate speed override
Feedrate override

D Optional block skip (Block delete)
Single block
Machine block
Dry run

E Memory protect

F Feed hole

G Jog/incremental feed axis direction selection
Manual rapid traverse selection
selection

H General–purpose switch : Eight general–purpose switches are provided and each of these switches can be named by up to eight alphanumeric characters.

There is a parameter per groups A – G shown above, which decides validity of operation function by CRT/MDI panel.
It is possible to disable the display of a switch which is disabled by this parameter (by parameter setting).

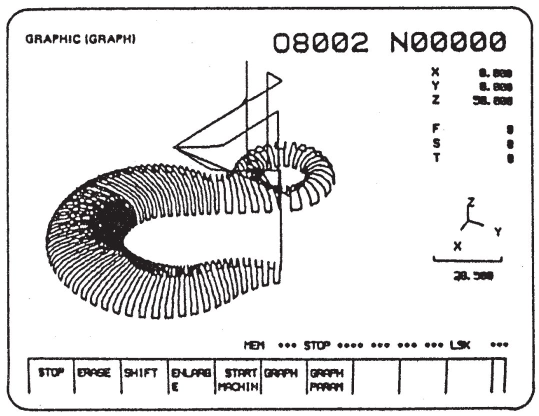
NOTE
With system using two manual pulse generators, axis selection via the software operator's panel cannot be performed.

24.12 GRAPHIC DISPLAY FUNCTION

This function allows display of tool path on the CRT screen, making program check easier. The following functions are offered.

1) Tool path of the machining program can be displayed. Machining process can be checked just by viewing the tool path drawing on the CRT screen.
   Program check before machining can be done by displaying the programmed locus on the CRT screen.
2) Display is possible with the XY plane, YZ plane, ZX plane, or isometric ; scaling of the screen is also possible.

> **NOTE**
> 14″color CRT/MDI is required.

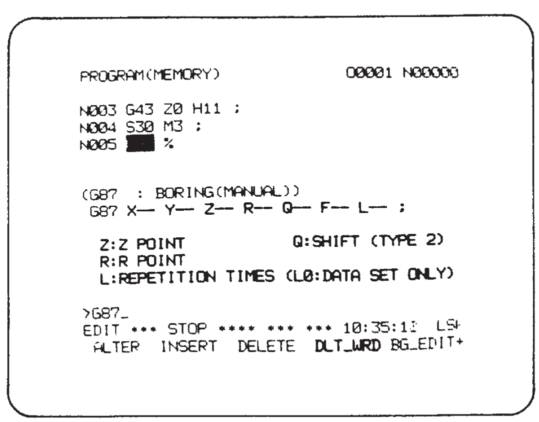


24.13 NC FORMAT GUIDANCE

A guidance for programming with NC format.

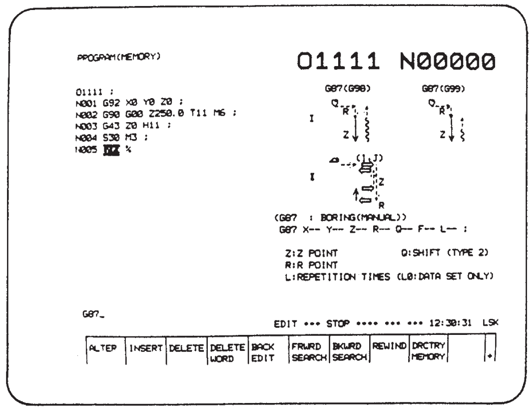
i) List of G code.
ii) Standard format of 1 block for G code.

Above guidance can be displayed on the CRT screen. In case if you have forgotten the G code or G code format, for example, by referring to this guidance, it eliminates the trouble of referring to the operator's manual. Thus, it reduces the programming time.



24.14 NC FORMAT GUIDANCE WITH PICTURE

Standard format of 1 block for G code guidance can be displayed on CRT screen with figure.



> **NOTE**
> This function is only available for 14″CRT/MDI.

24.15 SIMPLE CONVERSATIONAL AUTOMATIC PROGRAMMING FUNCTION

The NC cycle program can be created by selecting the menu displayed on the CRT or inputting data according to the menu instead of programming by using the NC format. Namely the programmer selects in the processing order those required for the actual cycle from among the menus each representing such turning processes for example drilling processes as boring, tapping, etc. Furthermore, the data required for each process, for example the hole position, the hole depth, etc. is asked in the menu. The programmer can create the program by simply inputting numeric values in response to these questions.

Basically this function is realized by the custom macro to be created by each machine tool builder. Since the machine tool builder can freely decide on the menu of which processes to prepare or how to prepare the menu of the data required for each cycle, he can utilize a function incorporating his original processing know–how.

NOTE

1 To order this function, it is also necessary to order the following options:
   i) Custom macro
   ii) 80 m of part program storage length
   Of the part program storage length, 35 m are used for the simple conversational programming program registration area, etc.
   Therefore, the part program storage length of a normal NC format is 35m less than the total length. As these require the part program storage length of the custom macro for the cycle which was created by the machine tool builder, the part program storage length which the end user can program in the NC format is obtained by subtracting such a part program storage length.

2 This function is applicable only to the 9″ CRT/MDI. It is not applicable to the 14″ CRT/MDI.

24.16 DATA PROTECTION KEY

A data protection key can be installed on the machine side for protection of various NC data. The following three input signals are offered, according to type of data to be protected.

1) KEY 1
   Allows input of tool compensation amount and workpiece zero point offset amount.
2) KEY 2
   Allows setting data input, and absolute coordinate value preset.
3) KEY 3
   Allows part program input and editing.

24.17 DIRECTORY DISPLAY OF FLOPPY CASSETTE/ PROGRAM FILE

File names in the floppy cassette (FANUC CASSETTE F1) and program file (FANUC PROGRAM FILE Mate can be listed on the CRT display (directory display). Each file name of up to 17 letters can be displayed in directory display.
Files in the floppy cassette are :

> NC command program, NC parameter/pitch error compensation data, tool compensation data, and etc.

When NC program in part program memory is written into the floppy cassette, program number can be given to it as a file name. When NC parameter/pitch error compensation data is written into the floppy cassette, “PARAM AND PITCH” is given them as a fixed name. When tool compensation data is written into the floppy cassette, “OFFSET” is given to it as a fixed name.

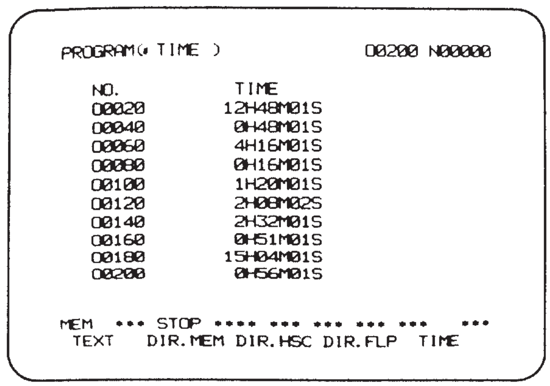
24.18 MACHINING TIME STAMP FUNCTION

Up to 10 machining times counted each main program are displayed on the program machining time display screen in time, minute, and second. When more than ten programs are operated, programs are discarded in the order of older ones.

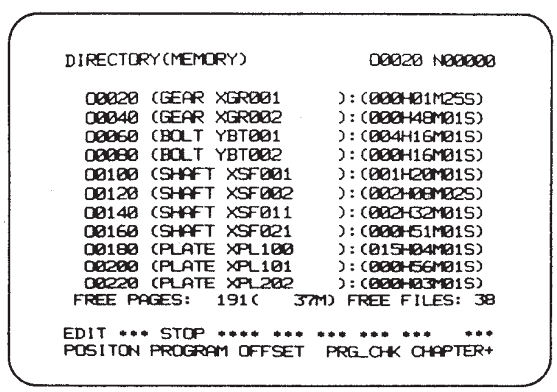
In the memory operation mode, the time from the initial start to the next reset or to the M02/M30 can be counted after reset. The execution time for M, S, T and B functions are added, but the time during operation stop is not added.

The machining time being displayed can be inserted (stamped) in the program stored into the memory as comments. The machining time is inserted after the program number as a comment.

The machining time inserted after the program number can be displayed instead of the tape length of program (amount of used memory) on the program directory screen. The display between machining time and tape length can be selected by the setting.
Since the machining time for each program can be known, it becomes a valid reference data on process planning at factories.



Machining time display screen



Program directory screen

24.19 DIRECTORY DISPLAY AND PUNCHING ON EACH GROUP

Programs can be classified and display in each groups such as in workpiece unit in addition to conventional program directory display (directory display) which shows the program numbers and names of all registered programs.

It is required that the program name of the same group should begin with the same character string.

With the directory display on each group, when a group name (character string) to be directory displayed is specified from the MDI, only the programs whose program names start with the specified group are directory displayed.
Also, when a group name to be punched is specified from the MDI, the programs which start with the specified group name are collectively punched.

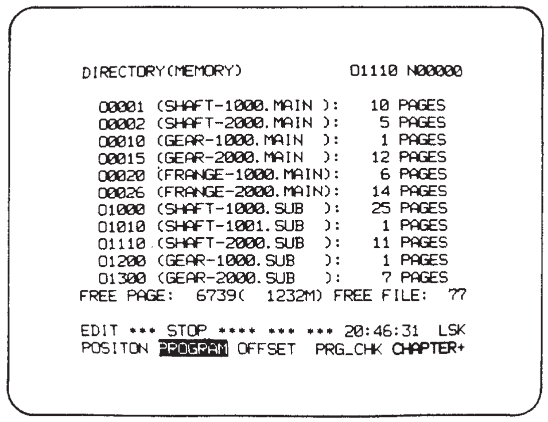


Fig. 24.19 (a) Directory display of registered programs

When the program as Fig. 24.19 (a) are registered, if “SHAFT” is set, a directory shown in Fig. 24.19 (b) is displayed.

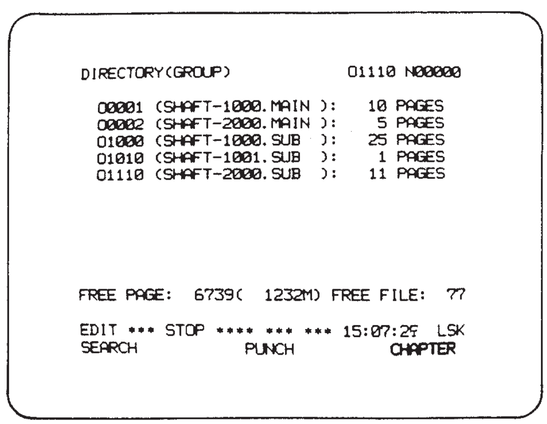


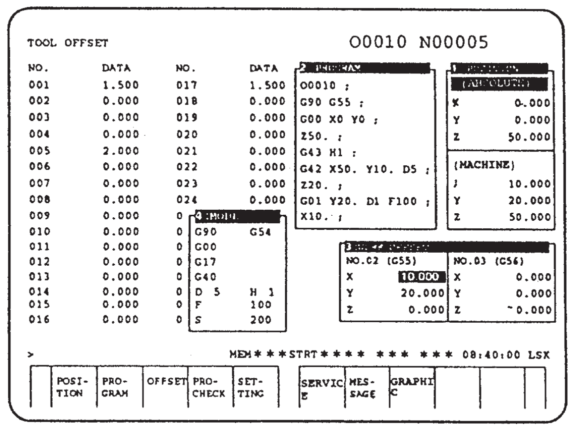
Fig. 24.19 (b) Directory display on each group

24.20 FUNCTION FOR DISPLAYING MULTIPLE SUBSCREENS

At times, the operator may want to see the data of two or more CNC screens at a time. For example :

- When the operator wants to check the current position simultaneously on both the screen for cutter compensation and the screen for the offset from the workpiece reference point
- When the operator wants to check the current program on the graphic screen

This function displays subscreens on the main screen.
The current position, current program, cutter compensation value, alarm, and other data can be displayed on a subscreen. There are very few restrictions on the data items that can be displayed on a subscreen and on the position and size of the subscreen. Simple input and editing are permitted for data on a subscreen. (On the position subscreen, for example, the origin/preset operation is permitted. On the cutter compensation subscreen, a compensation value can be input. Program editing and other complicated operations are not allowed.)
Multiple subscreens are displayed as shown below :



On a single main screen, up to five subscreens can be displayed. Information describing a displayed subscreen (data items, displayed position, and size) is stored. These items need not be specified each time the power is turned on. (Information on subscreens for up to all 30 main screens can be stored.)

24.21 HELP FUNCTION

The help function displays detained information about the alarm state of the CNC unit and soft key operation in a window on the CRT screen. This function can display the following:

(1) Alarm help
Alarms are issued when the operator makes an error in operating the CNC unit or a failure occurs in the CNC unit. The help function explains the cause and location of the error in detail. It also explains action to be taken to cancel the alarm condition.

(2) Soft key help
The soft keys displayed on the CRT screen depend on the operating state of the CNC unit. The help function explains in detail the function of each soft key that is currently displayed. It describes when a soft key needs to be pressed and what happens to the CNC unit when the key is pressed.
While operating the CNC unit, the operator can obtain necessary information from the window on the CRT screen without referring to the manuals.

24.22 PARAMETER SETTING (RS–232–C) SCREEN

It is possible to specify the parameters for the RS–232–C interface, remote buffer, and RS–422 interface the same screen.

24.23 SCREEN FOR SPECIFYING HIGH–SPEED AND HIGH–PRECISION MACHINING

The parameter for high–speed and high–precision machining can be specified on this screen. Since data entered in the parameter fields according to the units of measurement displayed on the screen, it is unnecessary to specify the units used for each parameter.
These parameters can be easily specified using the automatic setting function and automatic tuning function.
Up to three different patterns can be set for these parameters: finishing, medium, and roughing. The pattern to be used for actual machining can be specified by a program.

24.24 OPERATION HISTORY

This function always collects history data for keys pressed by the NC operator, the states of signals set by the NC operator, and alarms that occurred. In addition, this function enables the operator to monitor history data when necessary.

The main features of this function are:

(1) Collecting the following history data items:

1. Procedures in which the NC operator presses MDI keys
2. Changes in the states (on or off) of the input/output signals
3. Alarm data
4. Time stamp (time and date)

(2) Searching for the following data items:

1. Input/output signals
2. Alarm data
3. Time and date

(3) Outputting the following data items (punch–out):

1. All history data items
2. A selected range of history data items

(4) Selecting signals

Up to 20 input/output signals can be selected for history data collection.

(5) Alarm history data check

Details of alarm history data, including the time and date of alarm occurrence, can be checked.

24.25 WAVEFORM DIAGNOSIS FUNCTION

Waveform diagnosis functions are classified into the following two types:

(1) Single–shot type

This type of waveform diagnosis function enables graphic display of waveforms that represent variations in the following data items.
This function can generate triggers for sampling data when a machine control signal rises or falls. This facilitates the adjustment of servo and spindle motors.
The collected data can be output via a reader/punch interface.

[Data items for servos]

Servo errors, number of pulses to be generated, torque, electric current commands, heat simulation, and composite speed for all axes

[Data items for spindles]

The speed of each spindle and the value of the load meter for each spindle

[Data items for signals]

On or off states of machine control signals specified by signal addresses

(2) Servo–alarm type

This type of waveform diagnosis function triggers terminating data sampling when a servo alarm occurs or when the specified machine signal rises or falls. The termination of data sampling can be delayed by a specified time since the trigger is generated, thus facilitating detection of faults.

Recorded data items can be input or output to or from an external device using the reader/punch interface.

[Data items for servos]

Servo errors, number of pulses to be generated, torque, electric current commands, heat simulation, and composite velocity for all axes

[Data items for spindles]

None

[Data items for signals]

None

24.26 CRT SCREEN SAVING FUNCTION

The CRT screen saving function clears all data items on the CRT screen when the power is on and the CRT has not used within a certain period of time. This function is effective for extending the life of the screen.

(1) Small CRT/MDI Panel

To clear all data items on the CRT screen, press the <SHIFT> and <CAN> keys simultaneously. To subsequently redisplay data items on the CRT screen, press any key.

(2) Panels Other than the Small CRT/MDI Panel

To clear all data items on the CRT screen, press any function key and the <CAN> key simultaneously. To subsequently redisplay data items on the CRT screen, press any key.
The following function keys can be used.

POS	PROG	OFFSET	P–CHECK
SETTING	SERVICE	MESSAGE	OTHERS

24.27 M–CODE GROUP FUNCTION

M codes that have been executed, or which are being executed, are displayed in groups by specifying the group number (up to 127) for each M code and their function name, either on the M–code group setting screen or in a program. This function enables the display, on the screen, of the M function being used by the machine. In addition, the last M code is output for each group by the program restart M, S, T, and B code output function. The M–code group function also checks the validity of a combination of up to five M codes, specified in a block.

24.28 WORKPIECE ZERO POINT MANUAL SETTING FUNCTION

A workpiece origin offset value can be specified on the workpiece origin offset screen, so that the current position becomes a new workpiece origin or a specified position.

24.29 SCREEN SAVER FUNCTION

If the operator does not enter anything from the keyboard for a preset period, the screen saver function automatically erases the current display and calls the saver screen. The previous screen is retrieved if:

– The operator presses a key on the keyboard or on the PMC.

– An alarm is newly issued.

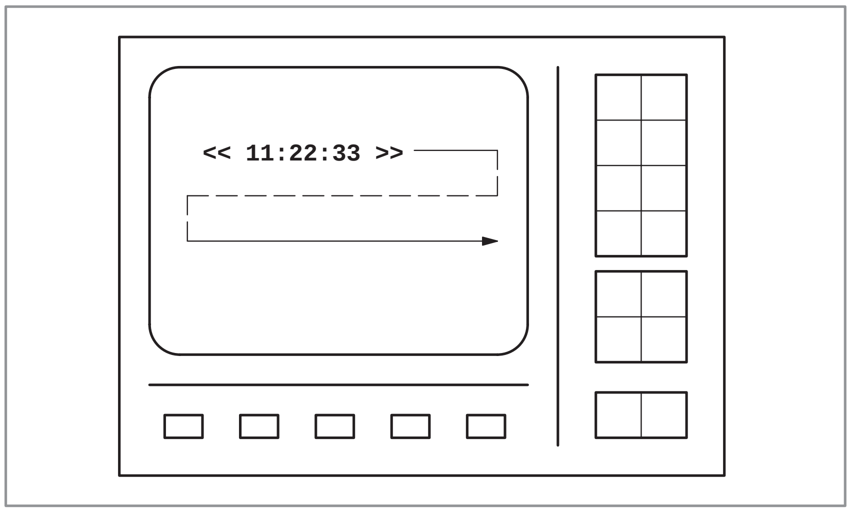
– An operator message is newly issued.

– The operation mode is switched.

– The saver return signal goes high.

Saver screen

The screen that appears when the screen saver function operates (saver screen) is shown below. The display of the current time moves from left to right and from top to bottom, after erasing the current screen, which remains unupdated.



25 PART PROGRAM STORAGE AND EDITING

25.1 FOREGROUND EDITING

The following part program storage and editing is possible

1) **Program tape registration to the memory**
   - Single program registration
   - Multi program tape registration
   - Additional program registration to registered program
2) **Program input via MDI**
3) **Program deletion**
   - Single program deletion
   - All programs deletion
4) **Program punching**
   - Single program punching
   - All programs punching
5) **Program editing**
   a) Change
      - Word change
      - Change of 1-word to multi-words

   b) Insertion
      - Word insertion
      - Multi words, and multi blocks insertion

   c) Deletion
      - Word deletion
      - Deletion to EOB
      - Deletion to the specified word
6) **Part program collation**
   Collation of program stored in the memory and program on the tape can be done.

25.2 BACKGROUND EDITING

Part program storage and editing can be done during machining. The same functions as foreground editing can be performed.

25.3 EXPANDED PART PROGRAM EDITING

The following editing is possible.

1) **Conversion**

   a) Address conversion
      An address in the program can be converted to another address. For example address X in the program can be converted to address Y.

   b) Word conversion
      A word in the program can be converted to another word. For example, a programmed M03 can be converted to M04.

2) **Program copy**
   A program can be copied to make a new program.

   a) Copy of all the program

   b) Copy of part of a program

3) **Program move**
   A program can be moved to make a new program.

4) **Program merge**
   A new program can be created by merging two programs.

5) **Copy and move to the key–in buffer**
   A part of a program can be copied or moved to the key–in buffer.

6) **Sequence number automatic insertion**
   The sequence number, where a certain increment value is added to the sequence number of the previous block can be automatically inserted at the head of each block in preparation of programs by the part program editing.

25.4 NUMBER OF REGISTERED PROGRAMS

Number of registered programs can be selected from the following: 100/400/1000

25.5 PART PROGRAM STORAGE LENGTH

The following part program storage length can be selected:
80/160/320/640/1280/2560/5120m

NOTE
Part program storage length may decrease according to options selected.

1) Custom macro common variables

Table 25.5 (a) Common variables and shortened part program length

Common variable	Shortened part program length	Remarks
Common variable A	0m	#100 to #149, #500 to #549
Common variable B	0m	#100 to #199, #500 to #599
Common variable C	2.2m	#100 to #199, #500 to #699
Common variable D	7.4m	#100 to #199, #500 to #599

2) Number of offset pairs

Table 25.5 (b) Offset pairs and shortened part program length

Offset memory	Offset pairs	Shortened part program length	Notes
A	32	0m	No differences between geometry/wear No differences between cutter diameter/tool length
	99	0m	
	200	1.5m	
	499	4.4m	
	999	10.3m	
B	32	0m	Differences between geometry/wear No differences between cutter diameter/tool length
	99	1.5m	
	200	3.7m	
	499	10.3m	
	999	22.0m	
C	32	0m	Differences between geometry/wear Differences between cutter diameter/tool length
	99	3.7m	
	200	8.1m	
	499	22.0m	
	999	44.7m	

3) Tool life management
   – Part program length shortens by 5.9m. In case of 512 groups of tool life management, the part program length shortens by 45m.
4) Tool offset by tool number
   The part program length shortens by 14m.
5) Additional workpiece coordinate system
   The part program storage length shortens by 4.4m.

25.6 PLAY BACK

Program can be prepared by storing machine position obtained by manual operation in the memory as program position. Data other than the coordinate value (M codes, G codes, feedrates, etc.) are registered in the memory by the same operation as part program storage and editing.

25.7 OVERRIDE PLAY BACK

It is possible to memorize the cutting feedrate override and spindle speed override effected during execution of a program and to operate the FANUC 10/11/12 according to the override memorized.

The storage of override is called teaching and the operation with memorized override is called playback.

The teaching can be made during memory operation only. When the teaching of override is commanded during memory operation, the override value in the following format can be stored after the command of block which is currently being executed in memory.

(Command of block which is currently executed)
————————, LIF —– R —–, LIS —– ;
or
————————, LIF —– R —– S —– ;

, where

, L1 : Command of override value

F —–: Cutting feed rate override value (0 to 254, unit of 1%)

R—–: Distance of long axis to the end point for linear line
unit: Least input increment of standard axis unit
Center angle to the end point of an arc Unit: angle (standard axis unit IS–A: 0.01 deg, IS–B: 0.001 deg, IS–C: 0.0001 deg)
If the remaining distance of the long axis or the remaining center angle of a circular arc becomes the value commanded by R or less during the playback operation, the commanded feedrate override value become valid.

S—–: Spindle speed override value (50 to 120, unit of 1%)
During the playback operation, the spindle override value commanded from the start of the block become valid.

When the override is memorized for several times during executing one block, the last override value is memorized.

The override value added to the last of block can be edited by the part program edit operation in the similar manner as other commands.

When the memorized (teaching) program is operated by override, the operation (playback) can be performed by the memorized feedrate override and spindle speed override. The memorized override value is valid immediately until the next override is received. In the override playback operation, set the override switch on the operator's panel to 100%.
In other setting, the override value commanded by the program is multiplied by the override value of the switch on the operator panel.

The override playback can be performed in all automatic (MEMORY, MDI, TAPE) operations.

The teaching can be performed using the MDI keys or using the switches and buttons added to the machine operator's panel. When it is performed on the machine operator's panel, it is required that the BMI interface be used.

The spindle speed binary output or analog output function is required to perform the teaching/playback of spindle speed override.

25.8 EXTERNAL I/O DEVICE CONTROL

Part program registration and punch can be commanded externally.

1) Program registration
A part program can be registered in memory through the input device selected for foreground editing (in case of part program edit mode) or through the input device selected for background editing (in case of other than part program edit mode) using the external read start signal.

2) Program punch
A part program can be punched through the output device selected for foreground editing (in case of part program edit mode) or through the output device selected for background editing (in case of other than part program edit mode) using the external punch start signal.

25.9 HIGH–SPEED PART PROGRAM REGISTRATION FUNCTION

This function speeds up the registration of part programs in the foreground mode (EDIT mode). In the background mode, part programs are registered at normal speed.

25.10 FUNCTION SELECTION WITH HARD KEYS

Function selection can be performed with not only soft keys, but also with the following hard keys:

POS	PROG	OFFSET	P–CHECK
SETTING	SERVICE	MESSAGE	OTHERS

These hard keys correspond to the following function selection items:

POS	:	Current position
PROG	:	Program
OFFSET	:	Offset
P–CHECK	:	Program check
SETTING	:	Setting
SERVICE	:	Service
MESSAGE	:	Message
OTHERS	:	Screen selected by parameter No.2215

25.11 MULTI–EDIT FUNCTION

The multi–edit function allows two programs to be edited at the same time by displaying them in the left right halves of the program text screen.

26 DIAGNOSIS FUNCTIONS

26.1 SELF DIAGNOSIS FUNCTIONS

The NC checks the following itself.

1) Abnormality of detection system
2) Abnormality of position control unit
3) Abnormality of servo system
4) Overheat
5) Abnormality of CPU
6) Abnormality of ROM
7) Abnormality of RAM
8) Abnormality in data transfer between CRT/MDI
9) Abnormality of part program storage memory
10) Abnormality in tape reader read function
11) Abnormality in data transfer between PMC

Signals from position coder, input/output signals, and inner status of the NC can be displayed on the CRT screen.

26.2 TROUBLE DIAGNOSIS GUIDANCE

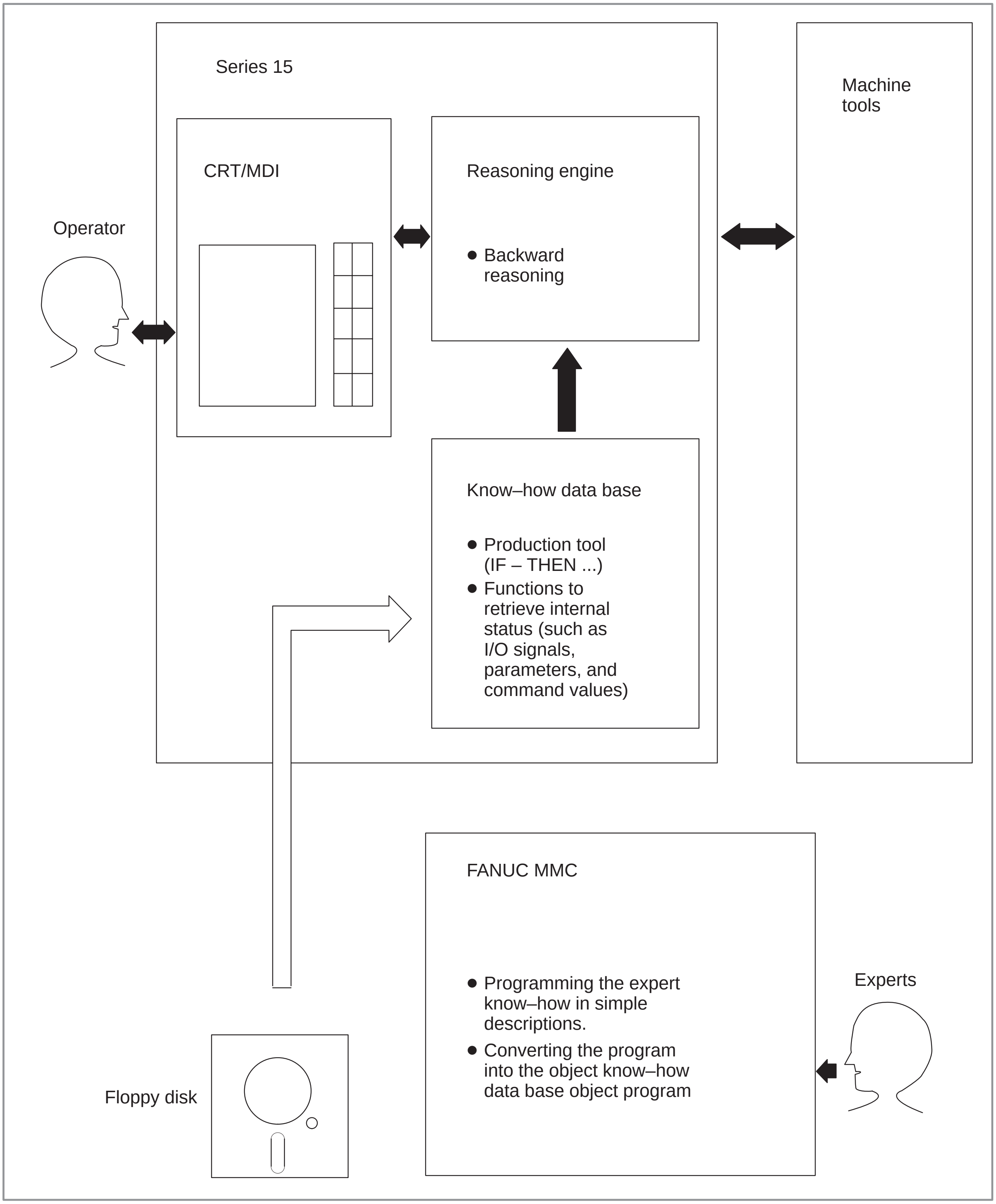
As computer technology has developed, research and development of Artificial Intelligence (AI) has greatly progressed in various fields. There are many AI applications such as automatic translation, picture recognition, audio recognition, and intelligent robots. Among these applications, expert systems have already reached the most practical level.

Series 15 introduced this expert system as a numeric controller for the first time in the world. This expert system is used for trouble diagnosis of the CNC machining tool. The trouble diagnosis guidance function of Series 15 has various features as shown below.

1) By storing know–how of experts who master troubleshooting and measures against various troubles occurring in the machine in the CNC memory, the reasoning engine built into the CNC troubleshoots the trouble cause of the machine based on the stored know–how in the same process used by the experts.
2) The operator may only perform simple conversational operations such as inputting the trouble phenomena through the CRT/MDI unit and answering questions given from the CNC if necessary, while the CNC diagnoses the trouble instead of an expert.
3) The functions for reading data in the control unit which are the parameter contents and the command values such as the feedrate and G code and the data transferred between the machine and the control unit, can be built into the know–how data base to allow establishment of the advanced and realtime trouble diagnosis.
4) Various commands for graphic display can also be built into the know–how data base. The questions and the instruction of the trouble cause given to the operator can be made easy to understand by illustrating the machine parts on the screen with these graphic display commands.
5) The expert know–how can be programmed in easy descriptions using the FANUC MMC. The rule for expressing the know–how is prescribed in an easy–to–understand format called a production rule “If...THEN...” The know–how described in the program is stored in the CNC know–how data base after being converted into the object format on the FANUC MMC.

6) Each know–how can be handled as an independent module to allow easy addition, correction, and deletion of know–how.

7) The trouble diagnosis contents can be changed by changing the know–how data base contents only to enable construction of the original trouble diagnosis experts system meeting the needs of each machine.



27 DATA INPUT/OUTPUT

The NC has the following input/output data.
These data are input/output via various input/output devices as CRT/MDI, tape reader, etc.

1) Input data

   The NC has the following input data.
   - Part program
   - Tool compensation amount, Workpiece zero point offset amount
   - Tool life management data
   - Setting data
   - Parameters

2) Output data

   The NC has the following output data.
   - Part program
   - Tool compensation amount, Workpiece zero point offset amount
   - Setting data
   - Parameters

27.1 TAPE READER

27.1.1 Tape Reader without Reels

1) Reading speed 300 ch/sec (60Hz) or 250 ch/sec (50Hz)
2) Reading method Opto–electrical (LED)
3) Tape capacity 20m
(When installed inside the control unit cabinet)

27.1.2 Tape Reader with Reels

1) Reading speed 300 ch/sec ±10% (50/60Hz)
2) Winding speed 600 ch/sec ±10% (50/60Hz)
3) Read method Opto–electrical (LED)
4) Reel capacity Reel radius 187mm dia.
150m of tape (tape thickness 0.108mm) windable
5) Tape capacity 20m
(When installed inside the control unit cabinet)
6) Tape rewinding function
Automatically rewinds up to % (ISO code) or ER (EIA code) by M30 command.
(This feature is effective with reels.)

In case of free standing type cabinet or built–in type 2 cabinet, a tape reader can be mounted in the cabinet.
In case of built–in type 1, unbundled type or panel mount type cabinet, it must be installed on the machine side.
The following tape can be used in this NC.

Table 27. 1. 2 Specification of paper tapes used in the CNC

	Item		Tape reader with/without reels
1	Kind of tape		8–channel paper tape (Mylar tape cannot be used)
2	Light transmission rate $\frac{\text{(Transmission light)}}{\text{(including light)}} \times 100\%$		40% or less
3	Color of tape		Any color is usable as long as light transmission percentage is 40% or less (black, grey, blue, pink, white).
4	Standard	Material	JIS C 6243 or EIA RS–227–A or ISO 1729 (Provided that light transmission rate of item 2 is satisfied).
		Dimensions and locations of hole	JIS C 6246 or EIA RS–227–A or ISO 1154

27.2 READER/PUNCHER INTERFACES

The following can be input/output via the reader/puncher interface.

a) Part program registration
b) Tool offset amount, workpiece zero point offset amount, tool life management data input
c) Parameter input
d) Part program punch
e) Tool offset amount punch
f) Parameter punch

27.3 INPUT/OUTPUT DEVICES

The following Input/Output devices are prepared, which are connectable to the reader/puncher interface.

27.3.1 FANUC FLOPPY CASSETTE

Data can be stored in this floppy cassette. NC data can also be input from this cassette. Used with FANUC Cassette Adaptor 3.

Outer dimensions	:	$90 \times 94 \times 33$ (mm)
Weight	:	24g
Memory Capacity	:	Equivalent to 770m of tape length

27.3.2 Portable Tape Reader

The portable tape reader is a carrying type paper tape reader. Used to load program, data, and parameter to the NC. The main feature of the portable tape reader is as follows: For the outline dimensions, refer to Appendix F.

Read speed :	300 ch/sec (60Hz), 250 ch/sec (50Hz)
Read method :	Opto–electrical (LED)
Interface with the NC :	Reader/puncher interface

27.3.3 FANUC PROGRAM FILE Mate

The built–in hard disk enables data to be stored and it can be connected to the reader/puncher interface to input data to CNC. This hard disk has a large storage capacity of approximately 50,000m of paper tape data, so it can register maximum 1024 command programs.
It can be connected to the remote buffer to achieve high–speed transfer of maximum 86.4 kbps.
The hard disk is sealed to be continuously used under the factory environment.

27.3.4 FANUC Handy File

The FANUC Handy File is a compact multifunctional input/output floppy disk unit for use with various types of FA equipment. Programs can be transferred or edited through operations performed directly on the Handy file or through remote operation from connected equipment.

Compared with media such as paper tape, a 3.5″ floppy disk is both compact and durable, and eliminates noise during input/output. Programs with a total capacity of up to 1.44 MB (equivalent to about 3600 m paper tape) can be saved on a single floppy dick.

28 SAFETY FUNCTIONS

28.1 EMERGENCY STOP

With the emergency stop, all commands stops, and the machine stops immediately. Connect the “emergency stop” signal both to the control unit and to the servo unit side.

When emergency stop is commanded, servo excitation is also reset, and servo ready signal will also turn off. Move distance of the machine will still be reflected in the actual position and machine position will not be lost (Follow up function). After resetting the emergency stop, operation can thus be continued without need of another reference point return.

Whether to reset the NC by emergency stop or to rise an alarm without resetting, is selected by parameter.

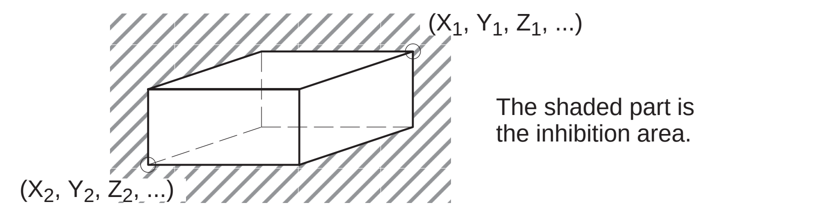
28.2 OVERTRAVEL FUNCTIONS

28.2.1 Overtravel

When the movable section has gone beyond the stroke end, a signal is output, the axis decelerates to a stop, and overtravel alarm is displayed. All directions on all axes has overtravel signals.

28.2.2 Stored Stroke Check 1

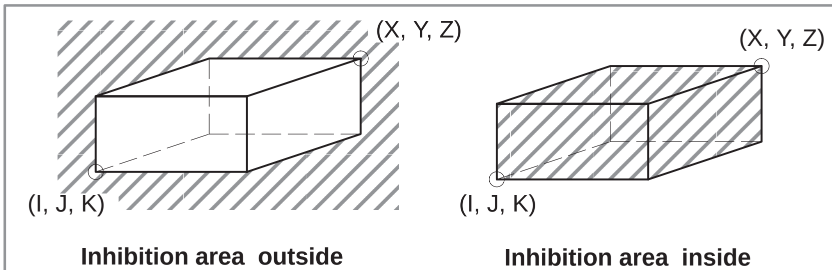
The movable section of the machine is parameter set in machine coordinates value. If the machine moves beyond the preset range, it decelerates to a stop and alarm is displayed. (This function is valid after manual reference position return at power on.)
This function can be used instead of hardware overtravel limit switch. When both is equipped with, both are valid.



This function can be used instead of hardware overtravel limit switch. When both is equipped with, both are valid.

28.2.3 Stored Stroke Check 2 (G22, G23)

An inhibition area can be specified inside or outside an area set by setting data or by program. Command distance from the machine coordinates zero point for limit positions. This function is valid after manual reference position return right after the power on. When specifying the limits with program, limits or axes X, Y, Z can be set.
The inhibition area can be changed according to the workpiece. The parameter decides whether the inhibition area is outside or inside the specified area.



On/off of stored stroke check 2 is commanded by program as follows:
G22: Stored stroke check function on
G23: Stored stroke check function off

Format

G22 X_ Y_ Z_ I_ J_ K_ ;

28.2.4 Stroke Check before Move

Before starting block move, end point coordinate value is checked according to actual position of the machine and commanded move distance, to check whether machine will move in the inhibition area of stored stroke check 1 or 2. If machine will invade the inhibition area, the machine is stopped right after move in the block starts and an alarm is displayed.

This function checks whether the end point of the block invades the inhibition area, but checking of the whole path is not done. When the machine enters the inhibition area in the half way, an alarm will arise by stored stroke check 1 or 2.

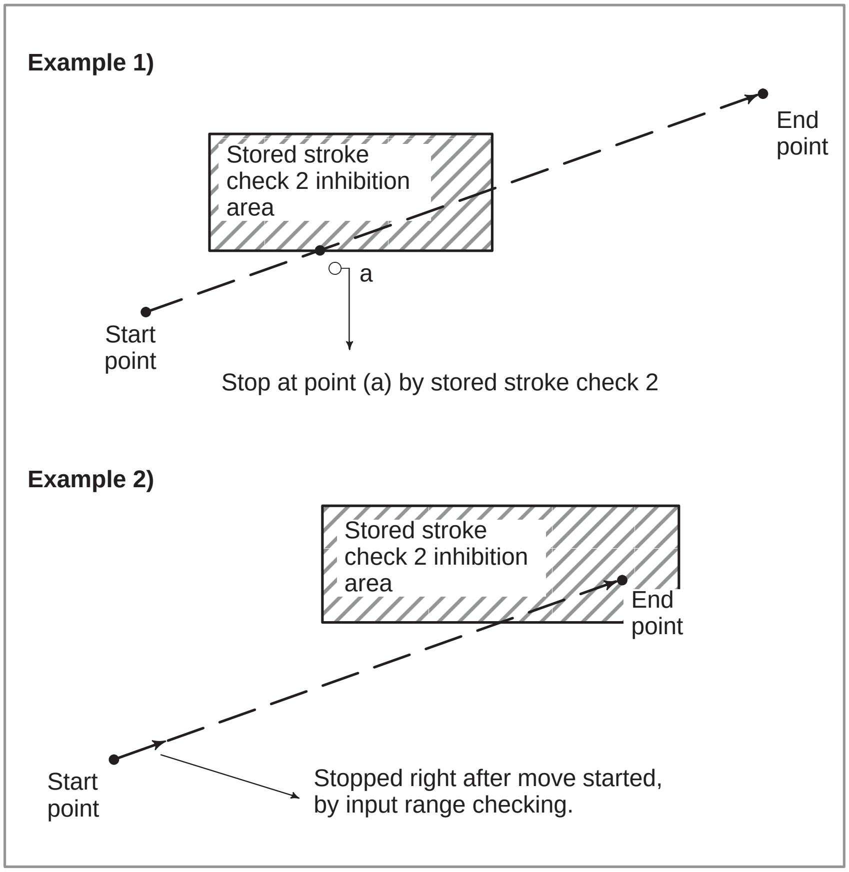


Fig. 28.2.4

28.3 INTERLOCK

28.3.1 Interlock per Axis

Axis feed commanded to each axis can be stopped separately. If interlock is commanded to any of the moving axis during cutting feed, all axes of the machine movement will decelerate to a stop. When interlock signal is reset, the moving starts.

28.3.2 All Axes Interlock

Feed of all axes can be inhibited. When all axes interlock is commanded during move, it decelerates and stops. When all axes interlock signal is reset, the moving restarts.

28.3.3 Automatic Operation All Axes Interlock

Feed of all axes in the automatic operation can be inhibited. When automatic operation all axes interlock is commanded during move, it decelerates to a complete stop. When the automatic operation all axes interlock is reset, move restart .

28.3.4 Block Start Interlock

Start of the next block can be inhibited during automatic operation. Block already started will continue to execute to the end. When block start interlock is reset, execution starts from the next block.

28.3.5 Cutting Block Start Interlock

Start of blocks with move commands other than positioning can be inhibited.

When cutting block start interlock is reset, execution of the next block is restarted.

In case when spindle rotation was activated, or when spindle speed was changed, cutting can be done in the correct speed by commanding cutting block start interlock till the spindle accelerates to the commanded speed.

28.4 EXTERNAL DECELERATION

Feedrate can be decelerated by an external deceleration signal from the machine side. A feedrate after deceleration can be set by parameter.
External deceleration is prepared every axis and every direction.
When the tool is to be moved in the reverse direction ,futile time may not be wasted since no external deceleration is applied.
Conditions to make this signal effective are set by parameters:

1) Whether this signal is applied to rapid traverse only or to all feeds.
2) Whether external deceleration in the + direction is made effective (each axis).
3) Whether external deceleration in the – direction is made effective (each axis).

This function allows the maximum of valid strokes and keeps shock to the machine to a minimum, to stops at stroke end. This function is also useful when overtravel of the machine from the stroke end in, for example, ATC reference points, must e kept to a minimum.

28.5 UNEXPECTED DISTURBANCE TORQUE DETECTION FUNCTION

This function is divided into two parts, as described below.

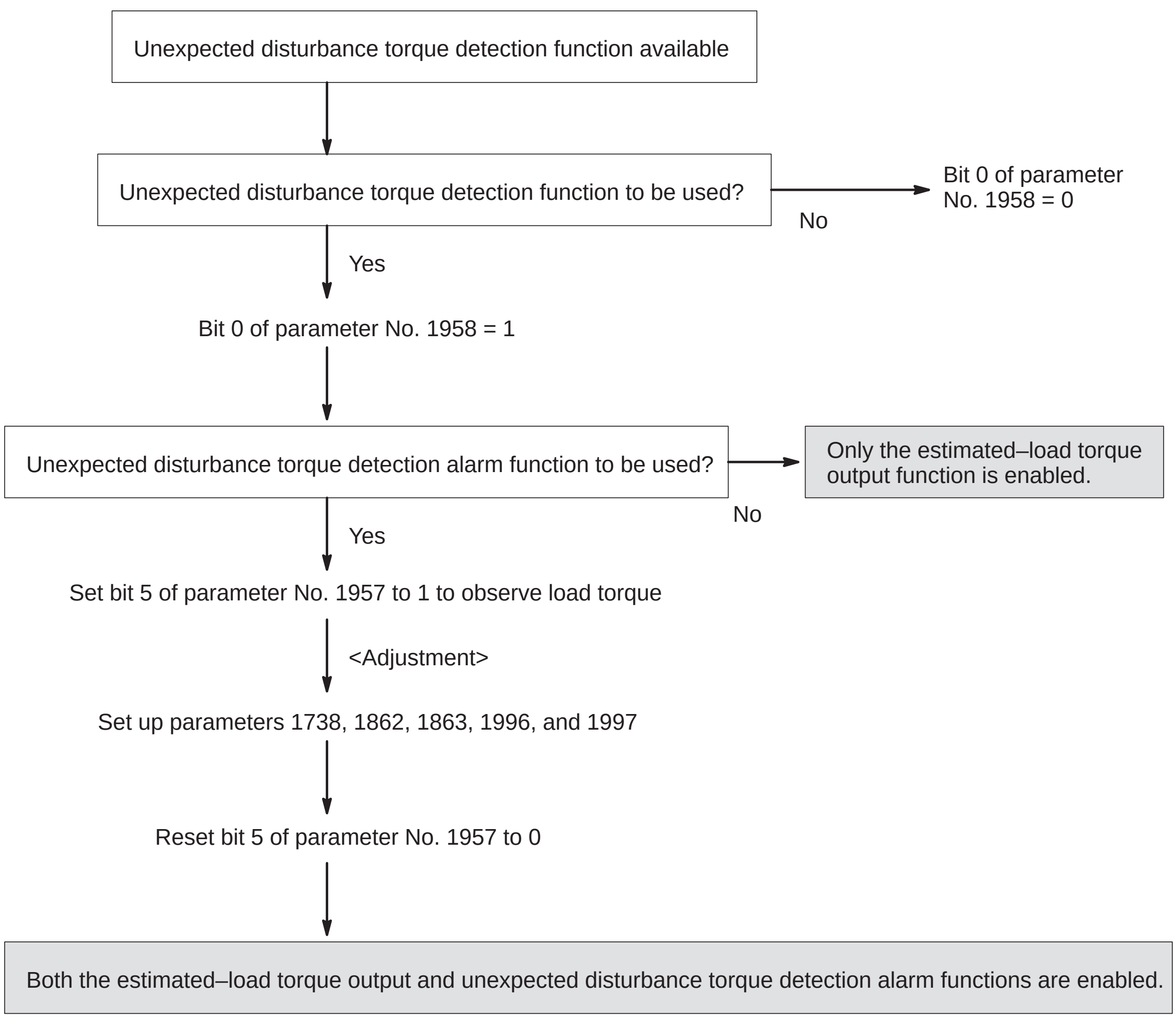
- Unexpected disturbance torque detection function
  1. Estimated–load torque output function

     The CNC is constantly estimating the load torque, which does not include the motor torque necessary for acceleration/deceleration. Enabling the estimated–load torque output function makes it possible to read this data from the PMC by using a window function.
  2. Unexpected disturbance torque detection alarm function

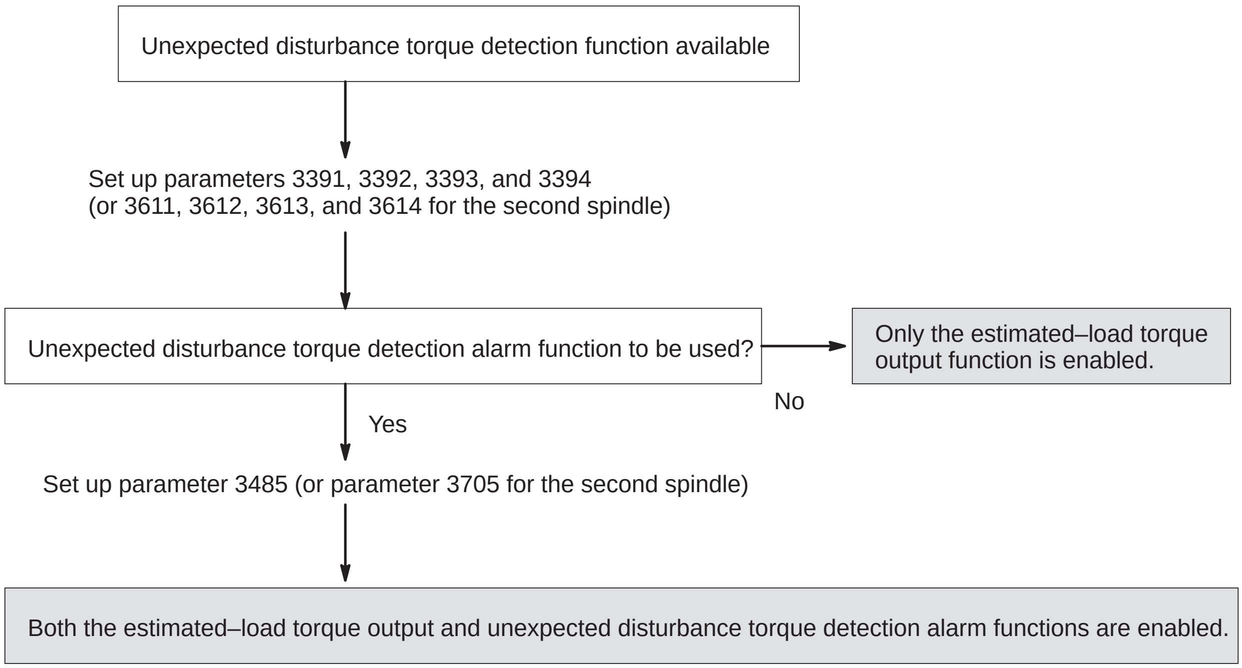
     If a load torque obtained by using the estimated–load torque detection function is larger than a value specified in a parameter, the unexpected disturbance torque detection alarm function causes the motor to stop or reverse by an amount specified in a parameter, then causes the CNC to output an alarm; reversing is possible only in the case of servo motors.

The parameters used with the unexpected disturbance torque detection function are described below.

(1) Servo axis



(2) Spindle axis



29 STATUS OUTPUT

29.1 NC READY SIGNAL

This signal is sent to the machine side when NC power is on and control becomes possible. Sending of this signal will be stopped when NC power is turned off.

29.2 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.

29.3 REWINDING SIGNAL

This signal shows that tape reader or main program in memory is rewinding.

29.4 ALARM SIGNAL

This signal is transmitted when the NC comes under an alarm status. Alarm type signal is also sent out.
By this alarm type signal, differences between the “overtravel alarm”, “over heat alarm”, “servo alarm”, “program error alarm” or “control unit abnormality alarm” are distinguished.

29.5 DISTRIBUTION END SIGNAL

This signal is sent out when pulse distribution of the M, S, T, or B functions has ended, so that they can be used after move of the commanded block ends.

29.6 AUTOMATIC OPERATION SIGNAL

This signal is sent out when it is under automatic operation.

29.7 AUTOMATIC OPERATION START LAMP SIGNAL

This signal is sent out when automatic operation is being activated.

29.8 FEED HOLD SIGNAL

This signal is sent out when automatic operation is held by feed hold.

29.9 RESET SIGNAL

This signal is sent out to show that the NC has been reset.

29.10 INPOSITION SIGNAL

This signal shows that an axis is under inposition status. This signal is output for all axes.

29.11 MOVE SIGNAL

This signal shows that an axis is moving. This signal is sent out for every 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.

29.12 AXIS MOVE DIRECTION SIGNAL

This signal is output to show move direction of each axis. This signal is output for each axis.

29.13 RAPID TRAVERSING SIGNAL

This signal shows that the move command is done under rapid traverse.

29.14 TAPPING SIGNAL

This signal is output to show that the machine is under tapping mode (G63) or tapping cycle (G74,G84) is under operation.

29.15 THREAD CUTTING SIGNAL

This signal shows that the machine is under thread cutting mode (G33).

29.16 CONSTANT SURFACE SPEED CONTROL SIGNAL

This signal shows that the machine is under constant surface speed control mode (G96).

29.17 INCH INPUT SIGNAL

This signal shows that input is done under inch input mode (G20).

29.18 DI STATUS OUTPUT SIGNAL

To inform the exterior of the states of menu switch, software operator's panel,which are set via CRT/MDI, and machine operator's panel, following DI state output signals are sent.

– Mode selection signal
– Signal block signal
– Manual absolute on/off signal
– Dry run signal
– Machine lock signal
– Display lock signal
– Auxiliary function lock signal
– Option block skip signal
– Mirror image signal

30 EXTERNAL DATA INPUT/OUTPUT

The external data input/output function allows NC operation by data sent from outside the NC (for example from the machine side).
The following external data input/output are available.

1) External tool compensation
2) External program number search
3) External sequence number search
4) External workpiece coordinate system shift
5) External machine coordinate system compensation
6) External alarm message
7) External operator message
8) External custom macro variable value input
9) External tool offset amount output
10) External program number output
11) External sequence number output
12) External workpiece coordinate system shift amount output
13) External machine coordinate system compensation amount output
14) External custom macro variable value output

30.1 EXTERNAL TOOL COMPENSATION

In this function, offset number is specified from outside to change tool offset amount.
The input signal designates whether the input tool offset amount is:

– absolute or incremental
– geometry offset or tool wear offset
– cutter radius compensation amount or tool length compensation amount

When 0 was specified for offset number, the offset number currently selected for use is selected.
If the machine is equipped with automatic measurement devices of tools and workpiece, error can be input to the NC with this function.
External tool compensation amount range is:

0 – ± 999999

in least command increment.

30.2 EXTERNAL PROGRAM NUMBER SEARCH

A program number from 1 – 9999 can be given from outside to the NC to call the corresponding program from the NC memory.
In machines with automatic loading function of various workpiece, this function can be used to automatically select and execute program suitable to the workpiece.

30.3 EXTERNAL SEQUENCE NUMBER SEARCH

A sequence number from 1 – 99999 is given from outside to the NC, and the sequence number is searched from the current selected program.

30.4 EXTERNAL WORKPIECE COORDINATE SYSTEM SHIFT

The workpiece coordinate system can be shifted for the shift amount given from outside.
The input signal specifies whether the input shift amount is:

– absolute or increment
– for which axis
– for which workpiece coordinate system (G54 – G59), or for all coordinate systems

This shift amount is not lost by power off.
The shift amount range is:

0 – ± 99999999

in least command increment.

30.5 EXTERNAL MACHINE COORDINATE SYSTEM COMPENSATION

The machine coordinate system is compensated by offset amount given from outside. This offset amount always take absolute value; never an increment value.
The offset amount range is:

$0 - \pm 9999$

in detection unit.
When offset amount is input, the actual machine move distance is the difference between the previous offset amount and current offset amount.
This function is used to compensate the machine coordinate system error caused by mechanical deformation.

30.6 EXTERNAL ALARM MESSAGE

By sending alarm number from outside, the NC is brought to an alarm status; an alarm message is sent to the NC, and the message is displayed on the CRT screen of the NC. Reset of alarm status is also done with external data.
Up to 4 alarm numbers and messages can be sent at a single time. Alarm numbers 0 – 999 can be sent to the NC, though the NC will display the number with an“EX” to distinguish from the internal numbers.
Up to 30 characters of alarm message can be sent with a single alarm number.

30.7 EXTERNAL OPERATORS MESSAGE

Message to the operator is given from outside the NC, and the message is displayed on the CRT screen.
The message is sent after the operator message number (0 – 999). Up to 4 messages with message numbers can be sent at a single time.
The message numbers 0 – 99 are displayed on the CRT screen along with the message. An “EX” is put before the number when displayed on the screen.
Message numbers 100 – 999 will not be displayed on the CRT screen; only the messages will be displayed on the CRT screen without the numbers.
Maximum 128 characters can be used for a single message. An external data will clear the operator messages.

30.8 EXTERNAL CUSTOM MACRO VARIABLE VALUE INPUT

By Specifying Custom Macro Variable Value Input

By specifying custom macro common variable number from outside, the variable value can be changed. Variable value of $0 - \pm 99999999$ can be input.

30.9 EXTERNAL TOOL OFFSET AMOUNT OUTPUT

A specified offset number of the tool offset amount can be output by a request from outside the NC. When 0 is specified as offset number, the currently selected offset number will be output.

30.10 EXTERNAL PROGRAM NUMBER OUTPUT

Main program number currently under execution can be output by an external request.

30.11 EXTERNAL SEQUENCE NUMBER OUTPUT

Sequence number of the currently executed block can be output by an external request.

30.12 EXTERNAL WORKPIECE COORDINATE SYSTEM SHIFT AMOUNT OUTPUT

Offset amount of the specified axis in the specified workpiece coordinate system can be output by an external request. When 0 is specified as the workpiece coordinate system number, the currently selected coordinate system number will be output.

30.13 EXTERNAL MACHINE COORDINATE SYSTEM COMPENSATION AMOUNT OUTPUT

Compensation amount of the specified axis in the specified machine coordinate system can be output by an external request.

30.14 EXTERNAL CUSTOM MACRO VARIABLE VALUE OUTPUT

Variable value of the specified common variable number can be output by an external request.

31 EXTERNAL WORKPIECE NUMBER SEARCH

By specifying workpiece numbers of 01 – 31 externally (from the machine side, etc.), program corresponding to the workpiece number can be selected.
The workpiece number and the program is corresponded in either of the following methods according to parameter selection.

1) The workpiece number equals the program number.
   For example when workpiece number 21 is specified, program, O0021 is selected.
2) The first 2 digits of the program number is optional, and the last 2 digits, the workpiece number.
3) The first 2 digits of the program number is the parameter set number, and the last 2 digits the workpiece number.

32 MACHINE INTERFACE

32.1 BASIC MACHINE INTERFACE (BMI)

All functions of the Series 15 can be used with this interface.

32.2 3M INTERFACE

This interface is compatible with the SYSTEM 3M.
There are limits to functions.

32.3 6M INTERFACE

This interface is compatible with the SYSTEM 6M.
There are limits to functions. When used with connection unit, physical interface will be the same as the SYSTEM 6M.

33 PROGRAMMABLE MACHINE CONTROLLER (PMC–NA/NB)

Magnetic sequence circuit of the machine side can be incorporated in the CNC.
With Series 15, maximum input of 1024 points and output of 1024 points can be processed.

<table>
<tr><th colspan="2"></th><th>I/O Link dealing connection unit</th><th>I/O Link dealing I/O Unit</th><th>I/O Link dealing operator's panel connection unit</th></tr>
<tr><td colspan="2">Maximum input point</td><td>192 points</td><td>1024 points</td><td>96 points</td></tr>
<tr><td colspan="2">Maximum output point</td><td>128 points</td><td>1024 points</td><td>64 points</td></tr>
<tr><td rowspan="5">User program steps (PMC–NA)</td><td rowspan="5">Select one of these options.</td><td colspan="3">Ladder; 8000 steps</td></tr>
<tr><td colspan="3">Ladder; 16000 steps, Memory only for C language/ PASCAL; 40 KB</td></tr>
<tr><td colspan="3">Ladder; 16000 steps, Memory only for C language/ PASCAL; 168 KB</td></tr>
<tr><td colspan="3">Ladder; 16000 steps, Memory only for C language/ PASCAL; 424 KB</td></tr>
<tr><td colspan="3">Ladder; 16000 steps, Memory only for C language/ PASCAL; 936 KB</td></tr>
<tr><td rowspan="5">User program steps (PMC–NB)</td><td rowspan="5">Select one of these options.</td><td colspan="3">Ladder; 8000 steps</td></tr>
<tr><td colspan="3">Ladder; 24000 steps</td></tr>
<tr><td colspan="3">Ladder; 24000 steps, Memory only for C language; 128 KB</td></tr>
<tr><td colspan="3">Ladder; 24000 steps, Memory only for C language; 384 KB</td></tr>
<tr><td colspan="3">Ladder; 24000 steps, Memory only for C language; 896 KB</td></tr>
</table>

In addition to ladder diagrams, program sequences can also be written in C or Pascal, thus allowing more complex control. Note that Pascal can be used for PMC–NA only.

NOTE
For the dedicated Pascal area, specify PMC control B function as the basic option.

33.1 PMC INSTRUCTION

There are two types of PMC instructions, basic and functional.

1) Basic instruction
   Basic instructions are used most extensivly in the design of sequence program and command to perform one–bit operations, such as AND, or OR, there are 12 types.
   The mnemonic language is as follows:

   RD, RD.NOT, WRT, WRT.NOT,
   AND, AND.NOT, OR, OR.NOT,
   RD.STK, RD.NOT.STK,
   ADN.STK, OR.STK
   SET, RST (PMC–NB only)

2) Functional instruction
   Functional instructions ease programming of machine interfaces that are difficult to program with basic instructions. 55 kinds of function instruction is prepared in PMC–NA. 68 kinds of function instruction is prepared in PMC–NB. See the following table.

No	Symbol	Function
1	END 1	First level program end
2	END 2	Second level program end
3	END 3	Third level program end
4	TMR	Timer processing
5	TMRB	Fixed timer processing
6	TMRC	Timer processing
7	DEC	Decoding
8	DECB	Binary decoding
9	CTR	Counter processing
10	CTRC	Counter processing
11	ROT	Rotation control
12	ROTB	Binary rotation control
13	COD	Code conversion
14	CODB	Binary code conversion
15	MOVE	Data transfer after AND
16	MOVOR	Data transfer after OR
17	COM	Common line control
18	COME	Common line control end
19	JMP	Jump
20	JMPE	Jump end
21	PARI	Parity check

No	Symbol	Function
22	DCNV	Data conversion
23	DCNVB	Extended data conversion
24	COMP	Comparison
25	COMPB	Binary comparison
26	COIN	Coincidence check
27	SFT	Shift register
28	DSCH	Data search
29	SDCHB	Binary data search
30	XMOV	Indexed data transfer
31	XMOVB	Binary indexed data transfer
32	ADD	Addition
33	ADDB	Binary addition
34	SUB	Subtraction
35	SUBB	Binary subtraction
36	MUL	Multiplication
37	MULB	Binary multiplication
38	DIV	Division
39	DIVE	Binary division
40	NUME	Constant definition
41	NUMEB	Binary constant definition
42	DISP	Message display
43	DISPB	Extended message display
44	EXIN	External data input
45	POS1 (*1)	Simple positioning module control
46	POS2 (*1)	Positioning module control
47	POSDP (*1)	Positioning module status data read
48	POSDO (*1)	Positioning module control data output
49	SPCNT	Spindle control
50	MONI (*1)	Monitor control
51	WINDR	Window data read
52	WINDW	Window data write
53	LIBRY (*1)	Library

No	Symbol	Function
54	LEND (*1)	Library end
55	FNC9X	optional function command (X=0 * 7)
56	MOVB (*2)	Transfer of 1 byte
57	MOVW (*2)	Transfer of 2 byte
58	MOVN (*2)	Transfer of an arbitrary number of bytes
59	JMPB (*2)	Label jump1
60	JMPC (*2)	Label jump2
61	LBL (*2)	Label
62	MMC3R (*2)	MMC3 window data read
63	MMC3W (*2)	MMC3 window data write
64	MMCWR (*2)	MMC window data read
65	MMCWW (*2)	MMC window data write
66	DIFU (*2)	Rising edge detection
67	DIFD (*2)	Falling edge detection
68	EOR (*2)	Exclusive OR
69	AND (*2)	Logical AND
70	OR (*2)	Logical OR
71	NOT (*2)	Logical NOT
72	END (*2)	End of a subprogram
73	CALL (*2)	Conditional subprogram call
74	CALLU (*2)	Unconditional subprogram call
75	SP (*2)	Subprogram
76	SPE (*2)	End of a subprogram

(*1) : It is effective only in PMC–NA.

(*2) : It is effective only in PMC–NB.

33.2 NC WINDOW

In addition to the former PMC functions, a large window between the PMC and the NC is offered, for the machine tool builders to make software and incorporate abundant new know–hows. The following functions are available in the PMC through the window.

– Read of MDI key data.

– Display of data on the CRT.

– Use of the non–volatile memory.

– Read of NC data.
  Machine position, skip position, servo delay amount, acceleration/deceleration delay amount, custom macro variables, parameter value, feedrate, diagnosis value, alarm number, tool offset data, modal data.

– Change of NC data
  Manual pulse generator interruption amount, custom macro variables, parameter value, feedrate, tool offset data.

– Format conversion of CNC command programs

– Input/output of CNC command programs

– Input/output of data through reader/puncher interfaces

33.3 NC WINDOW B

The PMC can process the following operations through the window.

1) Read out of tool management data

2) Graphic drawing

33.4 KEY INPUT FROM PMC

By allowing input signals from the PMC side to the CNC side to turn to and off, the same operation as operating keys on the CRT/MDI panel can be performed.
The following applications are possible, for instance. After allowing to travel the tool at an arbitrary machining position by using the playback function (option), when to store its positions as the program command, X, Y, Z, INSERT, etc. must be input via key operations. However, these operations can be realized simply by depressing a switch on the operator's panel at the machine side. Namely, just like same effects can be obtained by allowing key input signals such as X, Y, Z, INSERT, etc. to turn ON/OFF at the PMC side, when a switch is depressed.

33.5 OUTPUT AND SETTING OF PMC PARAMETERS

The PMC parameters can be output through the reader/puncher interface. Also, he output tape can be read for setting the parameter again.

34 MAN MACHINE CONTROL (MMC) (ONLY 150–MB)

Machine tool builders can incorporate highly advanced man–machine interface functions such as conversational automatic programming or conversational operation based on much knowhow.

34.1 HARDWARE SPECIFICATIONS

Item	Specifications	
Processor	32–bit microprocessor. Arithmetic processing unit can be optionally mounted.	
Main memory (RAM)	512 KB (including O/S area). It can be expanded up to a total of 832 KB optionally.	
Auxiliary memory	ROM file	512 KB/1 MB/2 MB (including O/S area)
	Data file	Bubble memory 512 KB/1 MB Battery backup RAM 128 KB/256 KB/512 KB
	Floppy disk (option for developing softwares)	Both–sided high density (format capacity 1 MB) 5–1/4 inch floppy disk x 2
Display	14 inch color CRT (commonly used with the CRT/MDI of CNC and PMC)	
	Character display	Alphanumeric/Kana (Japanese alphabets) characters. 80 characters x 27/21 lines JIS first–level Kanji (Chinese Character) 40 characters x 27/21 lines Color (x7), inversion, and blinking can be specified for can be specified for each character.
	Graphic display	640 x 432 dots Color (x7) can be specified for each dot.
Keyboard	Built–in keyboard (commonly used with the CRT/MDI of CNC and PMC)	
	ASCII configuration full keyboard (option for developing softwares)	
Interface	Reader/puncher interface	Serial port (x 3)
	Printer interface	Centronics specifications parallel port (x 1)

34.2 SOFTWARE SPECIFICATIONS

<table>
<tr><th>Item</th><th colspan="2">Specifications</th></tr>
<tr><td rowspan="3">Operating system (O/S)</td><td colspan="2">Single user multi–task O/S, editor, assembler, and debugger are also included.</td></tr>
<tr><td>Multi–task</td><td rowspan="2">It is supported by the O/S. These functions can be utilized by any language for development.</td></tr>
<tr><td>Graphics</td></tr>
<tr><td rowspan="5">Language for development</td><td>BASIC interpreter</td><td>Standard attachment. Hardware–dependent portion such as graphics is also supported by the language level.</td></tr>
<tr><td>C compiler</td><td rowspan="4">Purchased separately.

Nearly all languages for development marked for the above O/S are available.</td></tr>
<tr><td>PASCAL compiler</td></tr>
<tr><td>Relocatable assembler</td></tr>
<tr><td>Otehrs</td></tr>
<tr><td>FANUC library</td><td colspan="2">The following functions can be supported:
○ MMC/CNC window
○ MMC/PMC window
○ Expanded graphics function</td></tr>
</table>

34.3 MMC/CNC WINDOW

A large window is prepared between CNC and MMC.
The following operation can be performed at the MMC side via the window. For details, refer to the MMC Operator's Manual.

- ○ CNC system data input
- ○ Output of CNC command data for operation
- ○ Output of CNC command data for registration
- ○ Output of CNC command data for verification
- ○ CNC command data input
- ○ Specified program search
- ○ Specified program delete
- ○ All programs delete
- ○ Tool offset input
- ○ Tool offset output
- ○ Parameter input
- ○ Parameter output
- ○ Setting data input
- ○ Setting data output
- ○ Custom macro variable input
- ○ Custom macro variable output
- ○ Skip position input
- ○ Servo delay input
- ○ Acceleration/deceleration delay input
- ○ Model data input
- ○ Diagnosis input
- ○ A/D conversion data input
- ○ Alarm status
- ○ Program No. under execution
- ○ Sequence No. under execution
- ○ Actual speed
- ○ Spindle speed
- ○ Absolute position
- ○ Machine position

34.4 MMC/PMC WINDOW

The following functions are available at the MMC side through the MMC and PMC window:

- ○ DI/DO image input
- ○ Data input
- ○ Data output

> **NOTE**
> The MMC cannot be installed partially (simple conversational automatic programming, CRT/MDI 2 control, and 9″ CRT etc..)

35 CONTROL UNIT

35.1 CONTROL UNIT

There can be the following four kinds of control units of Series 15 and the one of best be selected according to the system configuration.

1) Kind and size of control unit

   3 slots : 202 (W) × 380 (H) × 172 (D) mm

   4 slots : 254 (W) × 380 (H) × 172 (D) mm

   6 slots : 366 (W) × 380 (H) × 172 (D) mm

   8 slots : 478 (W) × 380 (H) × 172 (D) mm

35.2 POWER SUPPLY

The input power source of Series 15 (CNC unit) is as follows.

200 to 240 V +10%, –15%
50/60 Hz ± 3 Hz

35.3 ENVIRONMENTAL CONDITIONS

1) Ambient temperature

   0°C to 45°C when operating

   –20°C to 60°C when stored or delivering

2) Change in temperature

   Max. 1.1°C/min

3) Humidity

   75% or less (relative humidity) generally

   Max. 95% for a short time (no condensation)

4) Vibration

   0.5 G or less when operating

5) Circumstances

   When using in places with thick dust, cutting oil, or organic solvents consult us.

36 SERVO

A connectable servo motor and the servo amplifier are as follows.

Servo motor	:	FANUC AC servo motor (With serial interface pulse coder)
Servo amplifier	:	FANUC AC servo amplifier (Digital servo)

37 POSITION DETECTOR

A connectable position detector is as follows.

For semi–closed control	:	Serial interface pulse coder (Servo motor built–in type)
For full–closed control	:	Pulse coder/Optical scale (2–phase pulse interface)

38 SPINDLE

A connectable spindle motor and the spindle amplifier are as follows.

Spindle motor	:	FANUC AC spindle motor, etc.
Spindle amplifier	:	FANUC AC spindle amplifier, etc.

39 MACHINE INTERFACE

Series 15 has the interface to connect FANUC I/O Link.
The device such as I/O Unit–MODEL A with FANUC I/O Link can be connected.

40 POSITION SWITCHING FUNCTION

This function outputs a signal when machine coordinates along a control axis are in the range specified by a parameter.
Specify in parameters a control axis and the range for machine coordinates in which the position switching signal is output.
The position switching signal can be output on up to ten lines.

APPENDIX

A RANGE OF COMMAND VALUE

Table A (a) Linear axis (in case of metric thread for feed screw and metric input)

	Increment system				
	IS–A	**IS–B**	**IS–C**	**IS–D**	**IS–E**
Least input increment	0.01 mm	0.01 mm or *1 0.001 mm	0.001 mm or *1 0.0001 mm	0.0001 mm or 0.00001 mm	0.00001 mm or 0.000001 mm
Least command increment	0.01 mm	0.001 mm	0.0001 mm	0.00001 mm	0.000001 mm
Interpolation unit	0.005 mm	0.0005 mm	0.00005 mm	0.000005 mm	0.0000005 mm
Max. programmable dimension *2	±999999.99 mm	±99999.999 mm	±9999.9999 mm	±9999.99999 mm	±999.999999 mm
Max. rapid traverse *4	2400000 mm/min	240000 mm/min	100000 mm/min	10000 mm/min	1000 mm/min
Feedrate range *4	0.0001 – 2400000 mm/min	0.0001 – 240000 mm/min	0.0001 – 100000 mm/min	0.00001 – 10000 mm/min	0.000001 – 1000 mm/min
Incremental feed *5	0.01, 0.1, 1, 10, 100, 1000 mm/step	0.001, 0.01, 0.1, 1, 10, 100 mm/step	0.0001, 0.001, 0.01, 0.1, 1, 10 mm/step	0.00001, 0.0001, 0.001, 0.01, 0.1, 1.0 mm/step	0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1 mm/step
Tool compensation	0 – ±999.99 mm	0 – ±999.999 mm	0 – ±999.9999 mm	0 – ±9999.99999 mm	0 – ±999.999999 mm
Backlash compensation *6	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses
Dwell time *3	0 – 999999.99 sec	0 – 99999.999 sec	0 – 9999.9999 sec	0 – 9999.99999 sec	0 – 999.999999 sec

Table A (b) Linear axis (in case of metric thread for feed screw and inch input)

	Increment system				
	IS–A	**IS–B**	**IS–C**	**IS–D**	**IS–E**
Least input increment	0.001 inch	0.001 inch or *1 0.0001 inch	0.0001 inch or *1 0.00001 inch	0.00001 inch or 0.000001 inch	0.000001 inch or 0.0000001 inch
Least command increment	0.01 mm	0.001 mm	0.0001 mm	0.00001 mm	0.000001 mm
Interpolation unit	0.0005 inches	0.00005 inches	0.000005 inches	0.0000005 inches	0.00000005 inches
Max programmable dimension *2	±39370.078 inches	±3937.0078 inches	±393.70078 inches	±39.3700787 inches	±3.93700787 inches
Max. rapid traverse *4	2400000 mm/min	240000 mm/min	100000 mm/min	10000 mm/min	1000 mm/min
Feedrate range *4	0.00001 – 96000 inch/min	0.00001 – 9600 inch/min	0.00001 – 4000 inch/min	0.00001 – 400 inch/min	0.000001 – 40 inch/min
Incremental feed *5	0.001, 0.01, 0.1, 1, 10, 100 inch/step	0.0001, 0.001, 0.01, 0.1, 1, 10 inch/step	0.00001, 0.0001, 0.001, 0.01, 0.1, 1 inch/step	0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1 inch/min	0.0000001, 0.000001, 0.00001, 0.0001, 0.001, 0.01 inch/min
Tool compensation	0 – ±99.999 inches	0 – ±99.9999 inches	0 – ±99.99999 inches	0 – ±999.999999 inches	0 – ±99.9999999 inches
Backlash compensation *6	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses
Dwell time *3	0 – 99999.999 sec	0 – 9999.9999 sec	0 – 999.99999 sec	0 – 9999.99999 sec	0 – 999.999999 sec

Table A (c) Linear axis (in case of inch thread for feed screw and inch input)

	Increment system				
	IS–A	**IS–B**	**IS–C**	**IS–D**	**IS–E**
Least input increment	0.001 inch	0.001 inch or *1 0.0001 inch	0.0001 inch or *1 0.00001 inch	0.00001 inch or 0.000001 inch	0.000001 inch or 0.0000001 inch
Least command increment	0.001 inch	0.0001 inch	0.00001 inch	0.000001 inch	0.0000001 inch
Interpolation unit	0.0005 inches	0.00005 inches	0.000005 inches	0.0000005 inches	0.00000005 inches
Max. programmable dimension *2	±99999.999 inches	±9999.9999 inches	±999.99999 inches	±999.999999 inches	±99.9999999 inches
Max. rapid traverse *4	240000 inch/min	24000 inch/min	10000 inch/min	1000 inch/min	100 inch/min
Feedrate range *4	0.00001 – 240000 inch/min	0.00001 – 24000 inch/min	0.00001 – 10000 inch/min	0.000001 – 1000 inch/min	0.000001 – 100 inch/min
Incremental feed *5	0.001, 0.01,0.1, 1, 10, 100 inch/step	0.0001, 0.001, 0.01, 0.1, 1, 10 inch/step	0.00001, 0.0001, 0.001, 0.01, 0.1, 1 inch/step	0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1 inch/min	0.0000001, 0.000001, 0.00001, 0.0001, 0.001, 0.01 inch/min
Tool compensation *6	0 – ±99.999 inches	0 – ±99.9999 inches	0 – ±99.99999 inches	0 – ±999.999999 inches	0 – ±99.9999999 inches
Backlash compensation	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses
Dwell time *3	0 – 99999.999 sec	0 – 9999.9999 sec	0 – 999.99999 sec	0 – 9999.99999 sec	0 – 999.999999 sec

Table A (d) Linear axis (in case of inch thread for feed screw and metric input)

	Increment system				
	IS–A	**IS–B**	**IS–C**	**IS–D**	**IS–E**
Least input increment	0.01 mm	0.01 mm or *1 0.001 mm	0.001 mm or *1 0.0001 mm	0.0001 mm or 0.00001 mm	0.00001 mm or 0.000001 mm
Least command increment	0.001 inch	0.0001 inch	0.00001 inch	0.000001 inch	0.0000001 inch
Interpolation unit	0.005 mm	0.0005 mm	0.00005 mm	0.000005 mm	0.0000005 mm
Max programmable dimension *2	±999999.99 mm	±99999.999 mm	±9999.9999 mm	±9999.99999 mm	±999.999999 mm
Max. rapid traverse *4	240000 inch/min	24000 inch/min	10000 inch/min	1000 inch/min	100 inch/min
Feedrate range *4	0.0001 – 2400000 mm/min	0.0001 – 240000 mm/min	0.0001 – 100000 mm/min	0.000001 – 10000 mm/min	0.0000001 – 1000 mm/min
Incremental feed *5	0.01, 0.1, 1, 10, 100, 1000 mm/step	0.001, 0.01, 0.1, 1, 10, 100 mm/step	0.0001, 0.001, 0.01, 0.1, 1, 10 mm/step	0.00001, 0.0001, 0.001, 0.01, 0.1, 1.0 mm/step	0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1 mm/step
Tool compensation	0 – ±999.99 mm	0 – ±999.999 mm	0 – ±999.9999 mm	0 – ±9999.99999 mm	0 – ±999.999999 mm
Backlash compensation *6	0 –±9999 pulses	0 – ±9999 pulses	0 –±9999 pulses	0 –±9999 pulses	0 –±9999 pulses
Dwell time *3	0 – 999999.99 sec	0 – 99999.999 sec	0 – 9999.9999 sec	0 – 9999.99999 sec	0 – 999.999999 sec

Table A (e) Rotary axis

	Increment system				
	IS–A	**IS–B**	**IS–C**	**IS–D**	**IS–E**
Least input increment	0.01 deg	0.01 deg or *1 0.001 deg	0.001 deg or *1 0.0001 deg	0.0001 deg or 0.00001 deg	0.00001 deg or 0.000001 deg
Least command increment	0.01 deg	0.001 deg	0.0001 deg	0.00001 deg	0.000001 deg
Interpolation unit	0.005 deg	0.0005 deg	0.00005 deg	0.000005 deg	0.0000005 deg
Max. programmable dimension *2	±999999.99 deg	±99999.999 deg	±9999.9999 deg	±9999.99999 deg	±999.999999 deg
Max. rapid traverse *4	2400000 deg/min	240000 deg/min	100000 deg/min	10000 deg/min	1000 deg/min
Feedrate range *4	0.0001– 2400000 deg/min	0.0001– 240000 deg/min	0.0001 – 100000 deg/min	0.00001 – 10000 deg/min	0.000001 1000 deg/min
Incremental feed *5	0.01, 0.1, 1, 10, 100, 1000 deg/step	0.001, 0.01, 0.1, 1, 10, 100 deg/step	0.0001, 0.001, 0.01, 0.1, 1, 10 deg/step	0.00001, 0.0001, 0.001, 0.01, 0.1, 1.0 deg/step	0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1 deg/step
Tool compensation	0 – ±999.99 deg	0 – ±999.999 deg	0 – ±999.9999 deg	0 – ±9999.99999 deg	0 – ±999.999999 deg
Backlash compensation *6	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses	0 – ±9999 pulses
Dwell time *3	0 – 999999.99 sec	0 – 99999.999 sec	0 – 9999.9999 sec	0 – 9999.99999 sec	0 – 999.999999 sec

*1 Selected by parameters for each axis. Certain functions are not applicable for axes with different increment systems (e.g. circular interpolation, tool nose radius compensation, etc.)

*2 When given commands for axes of different increment system in the same block, limitations are set by the smallest value.

*3 Will depend on the unit system of the axis on address X.

*4 The feedrate ranges shown above are limitations depending on CNC interpolation capacity. When regarded as a whole system, limitations, depending on the servo system, must also be considered.

*5 In case of BMI interface, incremental feed amount can be specified by setting amount (parameter setting.)

*6 The unit of backlash compensation is detection unit.

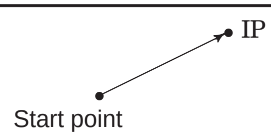
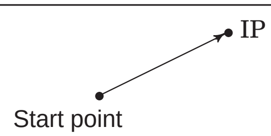
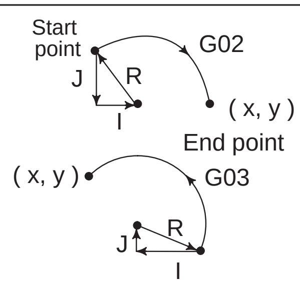
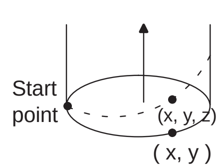
B FUNCTIONS AND COMMAND FORMAT LIST

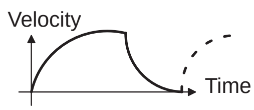
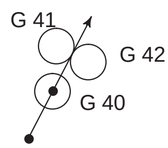
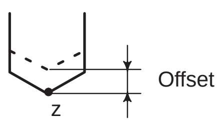
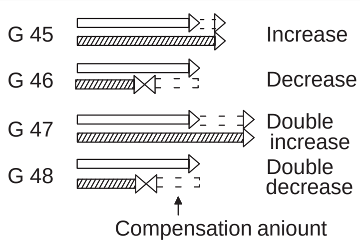
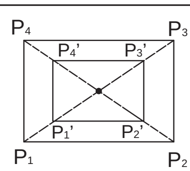
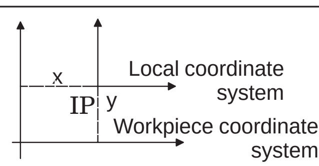
The symbols in the list represent the followings.

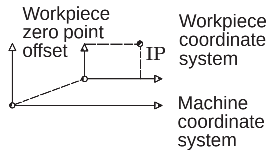
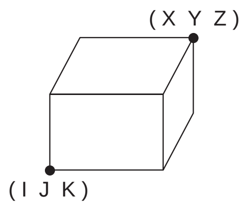
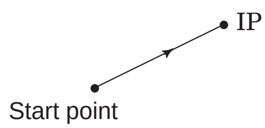
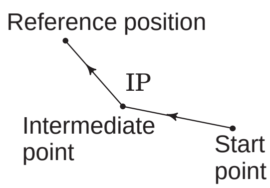
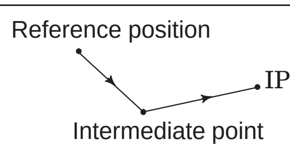
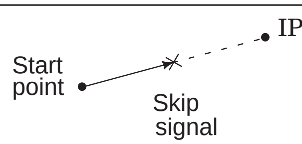
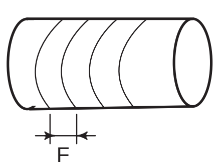
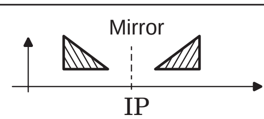
IP _______ : X _______ Y _______ Z _______ A . . . _______

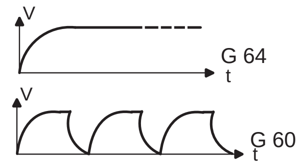
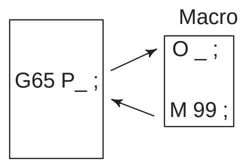
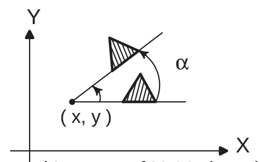
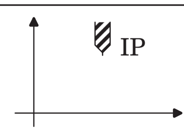
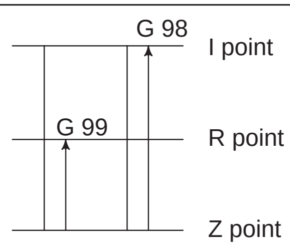
As seen above, the format consists of a combination of arbitary axis addresses among X, Y, Z, A, B, C, U, V, and W.

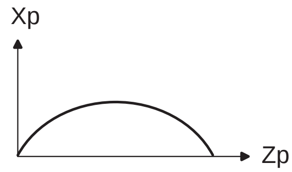
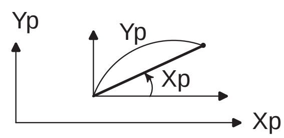
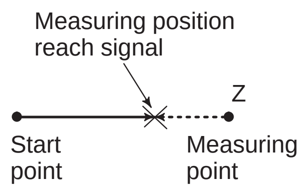
x : First basic axis (X usually)
y : Second basic axis (Y usually)
z : Third basic axis (Z usually)
α : One of arbitrary addresses
β : One of arbitrary addresses
Xp : X axis or its parallel axis
Yp : Y axis or its parallel axis
Zp : Z axis or its parallel axis

Functions	Illustrations	Command format
Positioning (G00)		G00IP___:
Linear interpolation (G01)		G01IP___F___:
Circular interpolation (G02, G03)	 (In case of X–Y plane)	G17 $\begin{Bmatrix} \text{G02} \\ \text{G03} \end{Bmatrix}$ Xp___ Yp___ $\begin{Bmatrix} \text{R\_\_\_} \\ \text{I\_\_\_ J\_\_\_} \end{Bmatrix}$ F___ ; G18 $\begin{Bmatrix} \text{G02} \\ \text{G03} \end{Bmatrix}$ Xp___ Zp___ $\begin{Bmatrix} \text{R\_\_\_} \\ \text{I\_\_\_ K\_\_\_} \end{Bmatrix}$ F___ ; G19 $\begin{Bmatrix} \text{G02} \\ \text{G03} \end{Bmatrix}$ Yp___ Zp___ $\begin{Bmatrix} \text{R\_\_\_} \\ \text{J\_\_\_ K\_\_\_} \end{Bmatrix}$ F___ ;
Helical interpolation (G02, G03)	 (In case of X–Y plane, G03)	G17 $\begin{Bmatrix} \text{G02} \\ \text{G03} \end{Bmatrix}$ Xp___ Yp___ $\begin{Bmatrix} \text{R\_\_\_} \\ \text{I\_\_\_ J\_\_\_} \end{Bmatrix}$ α___ F___ ; G18 $\begin{Bmatrix} \text{G02} \\ \text{G03} \end{Bmatrix}$ Xp___ Zp___ $\begin{Bmatrix} \text{R\_\_\_} \\ \text{I\_\_\_ K\_\_\_} \end{Bmatrix}$ α___ F___ ; G19 $\begin{Bmatrix} \text{G02} \\ \text{G03} \end{Bmatrix}$ Yp___ Zp___ $\begin{Bmatrix} \text{R\_\_\_} \\ \text{J\_\_\_ K\_\_\_} \end{Bmatrix}$ α___ F___ ; α: Any axis other than circular interpolation axes.

Functions	Illustrations	Command format
Dwell (G04)		Per second dwell $G04 \left\{\begin{matrix} X\_\_\_ \\ P\_\_\_ \end{matrix}\right\};$ Per revolution dwell $G95 \quad G04 \left\{\begin{matrix} X\_\_\_ \\ P\_\_\_ \end{matrix}\right\};$
Exact stop (G09)		$G09 \left\{\begin{matrix} G01 \\ G02 \\ G03 \end{matrix}\ \_\_\_\_\_\_ \right\};$
Change of offset value by program (G10)		Geometry offset amount G10 L10 P___ R___ ; Wear offset amount G10 L11 P___ R___ ; Work zero point offset amount G10 L2 P___ IP___ ;
Cutter compensation (G40–G42)		$\left\{\begin{matrix} G17 \\ G18 \\ G19 \end{matrix}\right\} \left\{\begin{matrix} G40 \\ G41 \\ G42 \end{matrix}\right\}$ IP__ D___ ; D: Tool offset No.
Tool length compensation (G43, G44, G49)		$\left\{\begin{matrix} G43 \\ G44 \end{matrix}\right\}$ α__ H __ ; $\left\{\begin{matrix} G43 \\ G44 \end{matrix}\right\}$ H __ ; H: Tool offset No. G49; … Cancel
Tool offset (G45–G48)		$\left\{\begin{matrix} G45 \\ G46 \\ G47 \\ G48 \end{matrix}\right\}$ IP__ D __ ; D: Tool offset number
Scaling (G50, G51)		G51 IP___ P___ ; P : Scaling magnification G50; … Cancel
Setting of local coordinate system (G52)		G52 IP___ ;
Command in machine coordinate system (G53)		G53 IP___ ;

Functions	Illustrations	Command format
Selection of workpiece coordinate system (G54 – G59)		$\left\{\begin{array}{c}\text{G54}\\ :\\ \text{G59}\end{array}\right\}$ IP___ ;
Single direction positioning (G60)		G60 IP___ ;
Inch/metric conversion (G20, G21)		Inch input G20; Metric input G21;
Stored stroke check (G22, G23)		G22 X___ Y___ Z___ I___ J___ K___ ; G23; … Cancel
Reference position return check (G27)		G27 IP__ ;
Reference position return (G28) 2nd, 3rd, 4th reference position return (G30)		G28 IP___ ; G30 P $\left\{\begin{array}{c}2\\ 3\\ 4\end{array}\right\}$ IP___ ;
Return from reference position (G29)		G29 IP__ :
Skip function (G31) Multiple skip function (G31.1 – G31.3)		$\left\{\begin{array}{l}\text{G31}\\ \text{G31.1}\\ \text{G31.2}\\ \text{G31.3}\end{array}\right\}$ IP__ F __ ;
Thread cutting (G33)		Even lead thread cutting G33 IP___ F___ Q___ : Q : Thread cutting start point shift angle Inch thread cutting G33 IP___ E___ Q___ ; E : Threads per inch
Programmable mirror image (G50.1, G51.1)		G51.1 IP___ ; G50.1; … Cancel

Functions	Illustrations	Command format
Cutting mode/Exact stop mode, Tapping mode, Automatic corner override		G64 ___ ; Cutting mode G60 ___ ; Exact stop mode G62 ___ ; Automatic corner override mode G63 ___ ; Tapping mode
Custom macro (G65, G66, G66.1, G67)		One–shot call G65 P___ <Argument assignment>; P : Program No. Modal call {G66 / G66.1} P___ <Argument assignment>; G67; … Cancel
Coordinate system rotation (G68, G69)	 (In case of X–Y plane)	G68 {G17 Xp __ Yp __ / G18 Zp __ Xp __ / G19 Yp __ Zp __} R α ; G69 ; … Cancel
Canned cycles (G73, G74, G76, G80 – G89)	See “Canned cycle”.	G80; … Cancel {G73 / G74 / G76 / G81 / : / G89} x__ y__ z__ P__ Q__ R__ F__ L__ ;
Absolute/incremental programming (G90/G91)		G90 __________; Absolute G91 __________; Incremental G90 ___ G91___ ; Combined use
Change of workpiece coordinate (G92)		G92 IP___ ;
Inverse time/Per–minute feed/Per–revolution feed (G93, G94, G95)	1/min mm/min inch/min mm/rev inch/rev	G93___ F___ ; Inverse time G94___ F___ ; Feed per minute G95___ F___ ; Feed per revolution
Initial point return/R point return (G98, G99)		G98 ____________ ; G99 ____________ ;
Constant surface speed control (G96, G97)	m/min or feet/min.	G96 S __ ; G97; … Cancel

Functions	Illustrations	Command format
Hypothetical axis interpolation (G07)		G07 α0; {G17 G18 G19} {G02 G02} Xp___ Yp___ Zp___ ; G07 α1; α:Hypothetical axis
Polar coordinate (G15, G16)	Local coordinate  Workpiece coordinate system	G17 G16 Xp___ Yp____ ; G18 G16 Zp___ Xp____ ; G19 G16 Yp___ Zp____ ; G15; … Cancel
Tool length measurement (G37)		G37 Z____ ;

C LIST OF TAPE CODE

ISO code										EIA code										Meaning
Character	8	7	6	5	4		3	2	1	Character	8	7	6	5	4		3	2	1	
0			○	○		∘				0			○			∘				Numeral 0
1	○		○	○		∘			○	1						∘			○	Numeral 1
2	○		○	○		∘		○		2						∘		○		Numeral 2
3			○	○		∘		○	○	3				○		∘		○	○	Numeral 3
4	○		○	○		∘	○			4						∘	○			Numeral 4
5			○	○		∘	○		○	5				○		∘	○		○	Numeral 5
6			○	○		∘	○	○		6				○		∘	○	○		Numeral 6
7	○		○	○		∘	○	○	○	7						∘	○	○	○	Numeral 7
8	○		○	○	○	∘				8					○	∘				Numeral 8
9			○	○	○	∘			○	9				○	○	∘			○	Numeral 9
A		○				∘			○	a		○	○			∘			○	Address A
B		○				∘		○		b		○	○			∘		○		Address B
C	○	○				∘		○	○	c		○	○	○		∘		○	○	Address C
D		○				∘	○			d		○	○			∘	○			Address D
E	○	○				∘	○		○	e		○	○	○		∘	○		○	Address E
F	○	○				∘	○	○		f		○	○	○		∘	○	○		Address F
G		○				∘	○	○	○	g		○	○			∘	○	○	○	Address G
H		○			○	∘				h		○	○		○	∘				Address H
I	○	○			○	∘			○	i		○	○	○	○	∘			○	Address I
J	○	○			○	∘		○		j		○		○		∘			○	Address J
K		○			○	∘		○	○	k		○		○		∘		○		Address K
L	○	○			○	∘	○			l		○				∘		○	○	Address L
M		○			○	∘	○		○	m		○		○		∘	○			Address M
N		○			○	∘	○	○		n		○				∘	○		○	Address N
O	○	○			○	∘	○	○	○	o		○				∘	○	○		Address O
P		○		○		∘				p		○		○		∘	○	○	○	Address P
Q	○	○		○		∘			○	q		○		○	○	∘				Address Q
R	○	○		○		∘		○		r		○			○	∘			○	Address R
S		○		○		∘		○	○	s			○	○		∘		○		Address S
T	○	○		○		∘	○			t			○			∘		○	○	Address T
U		○		○		∘	○		○	u			○	○		∘	○			Address U
V		○		○		∘	○	○		v			○			∘	○		○	Address V
W	○	○		○		∘	○	○	○	w			○			∘	○	○		Address W
X	○	○		○	○	∘				x			○	○		∘	○	○	○	Address X
Y		○		○	○	∘			○	y			○	○	○	∘				Address Y
Z		○		○	○	∘		○		z			○		○	∘			○	Address Z

ISO code										EIA code											Meaning
Character	8	7	6	5	4		3	2	1	Character	8	7	6	5	4		3	2	1		
DEL	○	○	○	○	○	∘	○	○	○	Del		○	○	○	○	∘	○	○	○	x	Delete (cancel an error punch).
NUL						∘				Blank						∘				x	Not punched. Can not be used in significant section in EIA code.
BS	○				○	∘				BS			○		○	∘		○		*	Back space
HT					○	∘			○	Tab			○	○	○	∘	○	○		*	Tabulator
LF or NL					○	∘		○		CR or EOB	○					∘					End of block
CR	○				○	∘	○		○											*	Carriage return
SP	○		○			∘				SP				○		∘				*	Space
%	○		○			∘	○		○	ER					○	∘		○	○		Absolute rewind stop
(			○		○	∘				(2–4–5)				○	○	∘		○			Control out (a comment is started)
)	○		○		○	∘			○	(2–4–7)		○			○	∘		○			Control in (the end of a comment)
+			○		○	∘		○	○	+		○	○	○		∘					Positive sign
–			○		○	∘	○		○	–		○				∘					Negative sign
:			○	○	○	∘		○													Colon (Address O)
/	○		○		○	∘	○	○	○	/			○	○		∘			○		Optional block skip
.			○		○	∘	○	○		.		○	○		○	∘		○	○		Period (A decimal point)
#	○		○			∘		○	○												Sharpe
$			○			∘	○													*	Dollar sign
&	○		○			∘	○	○		&					○	∘	○	○		*	Ampersand
’			○			∘	○	○	○											*	Apostrophe
*	○		○		○	∘		○													Asterisk
,	○		○		○	∘	○			,			○	○	○	∘		○	○		Comma
;	○		○	○	○	∘		○	○											*	Semicolon
<			○	○	○	∘	○													*	Left angle bracket
=	○		○	○	○	∘	○		○											*	Equal
>	○		○	○	○	∘	○	○												*	Right angle bracket
?			○	○	○	∘	○	○	○											*	Question mark
@	○	○				∘														*	Commercial at mark
”			○			∘		○												*	Quotation
[	○	○		○	○	∘		○	○												Left brace
]	○	○		○	○	∘	○		○												Right brace

NOTE

1 *: When read in the comment zone, the codes are read into the memory.
When read in the significant datazone, the codes are ignored.

2 x: Ignored.

3 When a custom macro option is used, the following codes can also be used in the significant data zone.
+. [,], #, *, @, ? in ISO code.
+ in EIA code and codes set by parameter.

4 Codes not in this table are ignored if their parity is correct.

5 Codes with incorrect parity cause the TH alarm; however, when they are in the comment zone, they are ignored without generating the TH alarm.

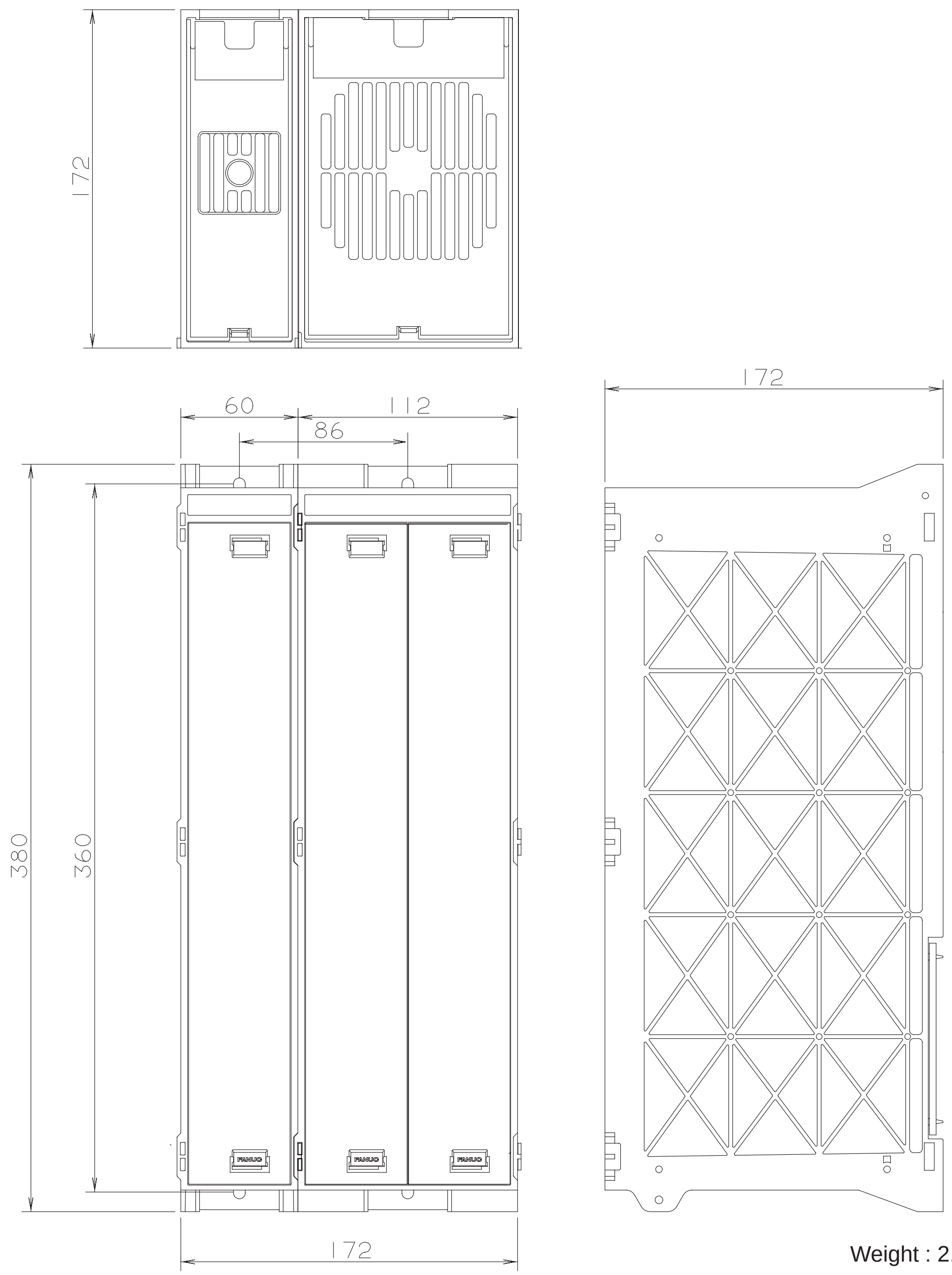
6 A character with all eight holes punched does not generate the TH alarm even if EIA code.

D EXTERNAL DIMENSIONS BASIC UNIT

Name	Number of Figure
3–slot control unit	Fig.1
4–slot control unit	Fig.2
6–slot control unit	Fig.3
8–slot control unit	Fig.4
ISA extension unit	Fig. 5

Fig. 1 3–SLOT CONTROL UNIT

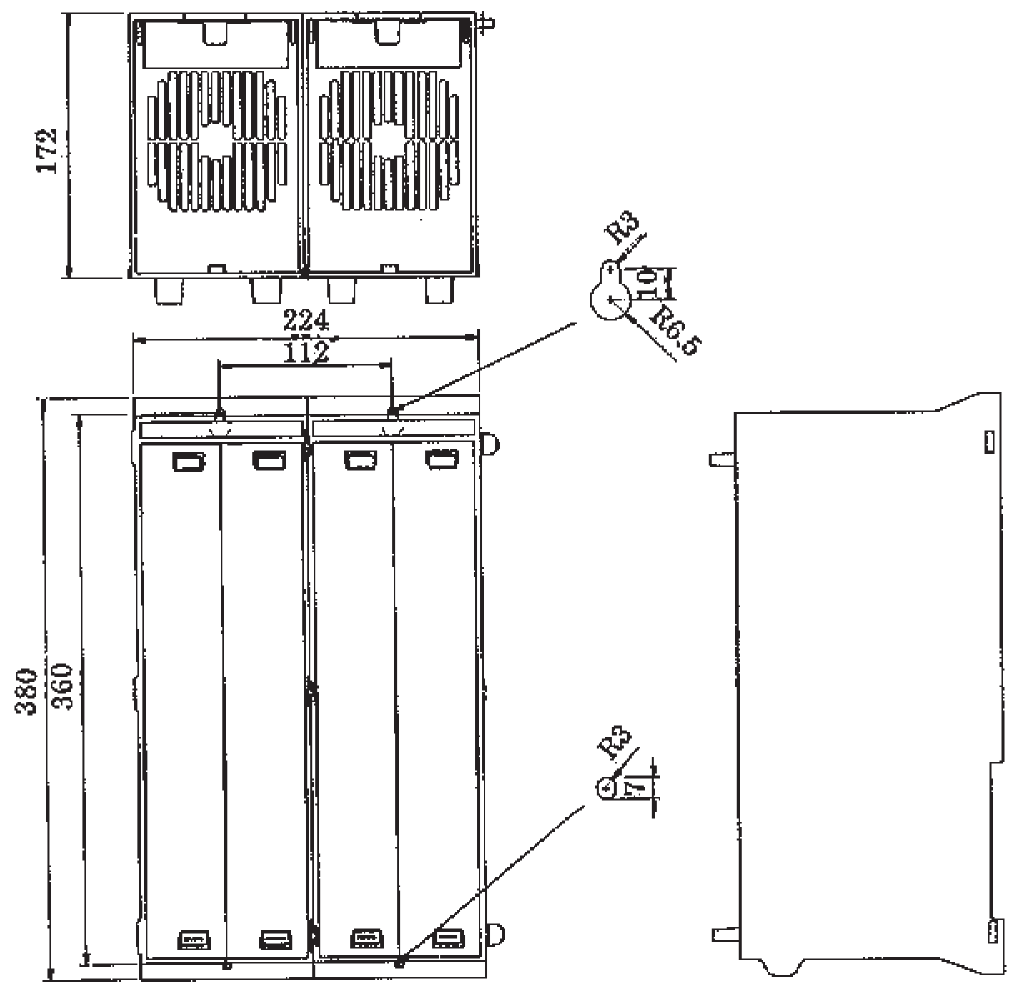
Specification : A02B–0162–B503
A02B–0162–B513



Weight : 2.3kg

Fig. 2 4–SLOT CONTROL UNIT

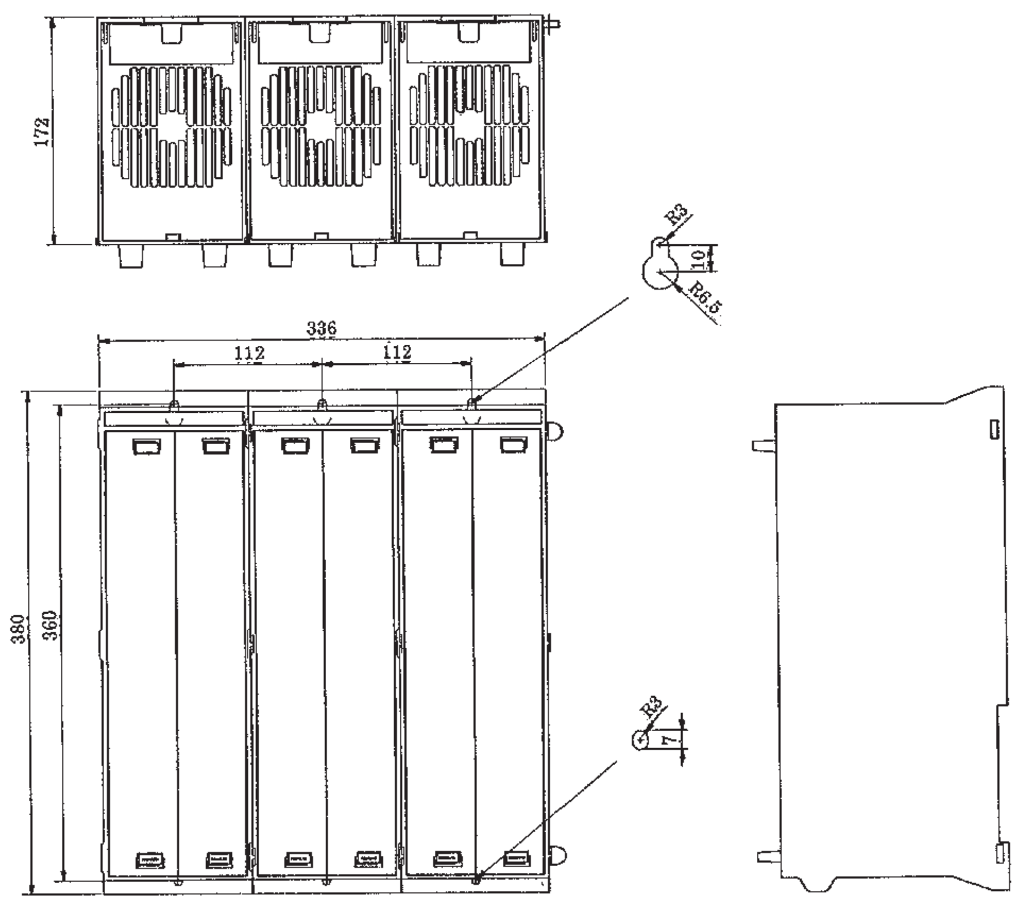
Specification : A02B–0162–B504
A02B–0162–B514
A02B–0162–B524
A02B–0162–B534



Weight : 2.5kg

Fig. 3 6–SLOT CONTROL UNIT

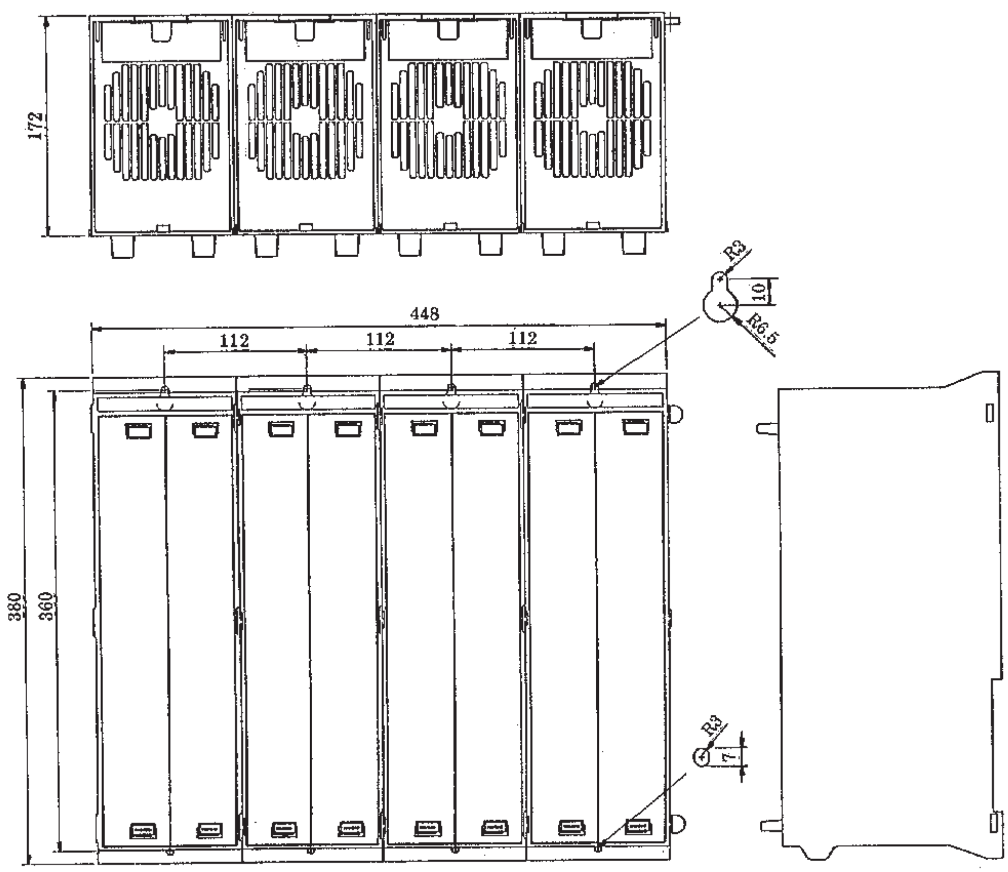
Specification : A02B–0162–B506
A02B–0162–B516
A02B–0162–B526
A02B–0162–B536
A02B–0162–B546
A02B–0162–B556
A02B–0162–B566
A02B–0162–B576



Weight : 3.8kg

Fig. 4 8–SLOT CONTROL UNIT

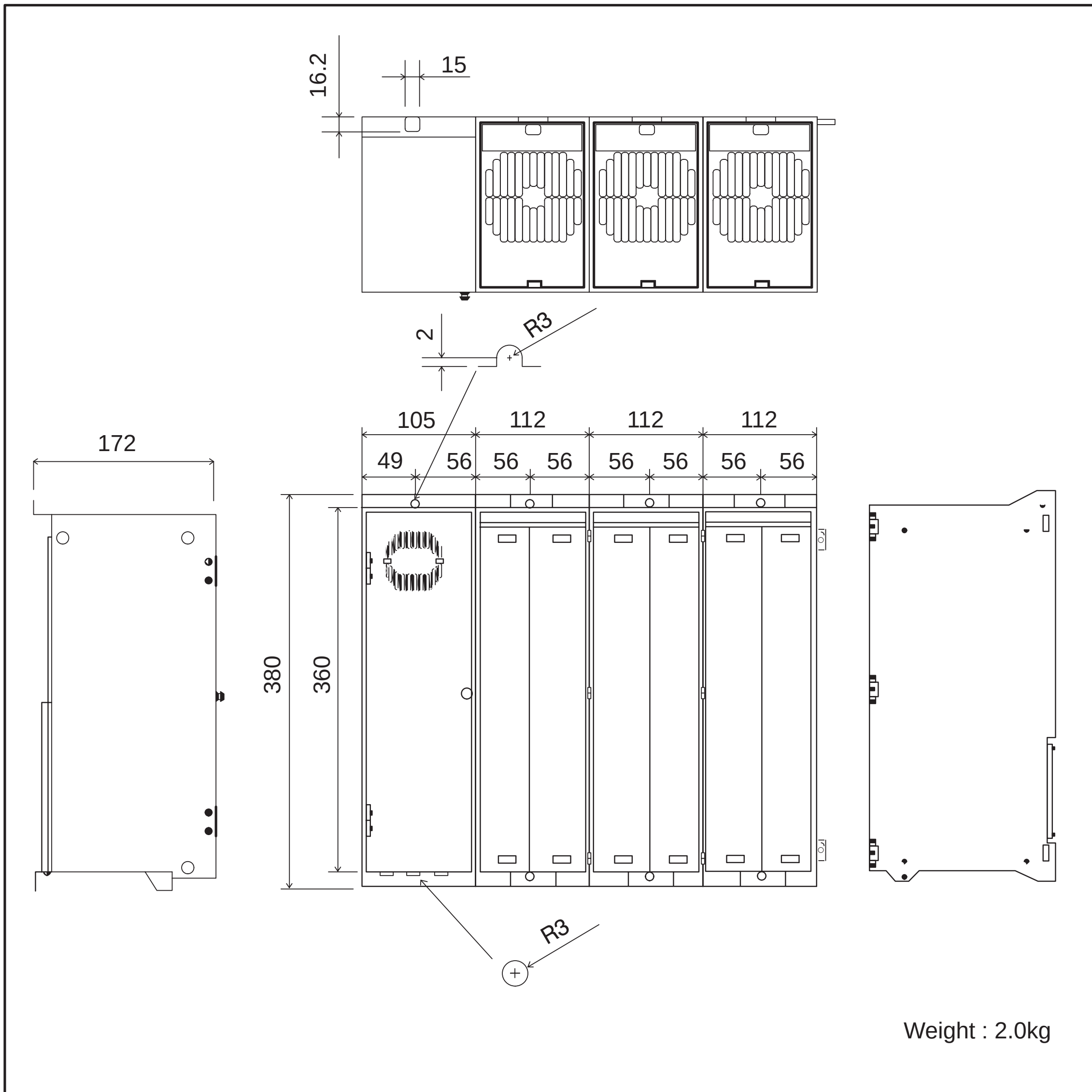
Specification : A02B–0162–B508 A02B–0240–B508 A02B–0244–B508
A02B–0162–B518 A02B–0240–B518 A02B–0244–B518
A02B–0162–B528 A02B–0241–B508
A02B–0162–B538 A02B–0241–B518
A02B–0162–B548 A02B–0242–B508
A02B–0162–B558 A02B–0242–B518
A02B–0162–B568 A02B–0243–B508
A02B–0162–B578 A02B–0243–B518



Weight : 5.0kg

Fig. 5 ISA EXPANSION UNIT

Specification : A02B–0207–C030



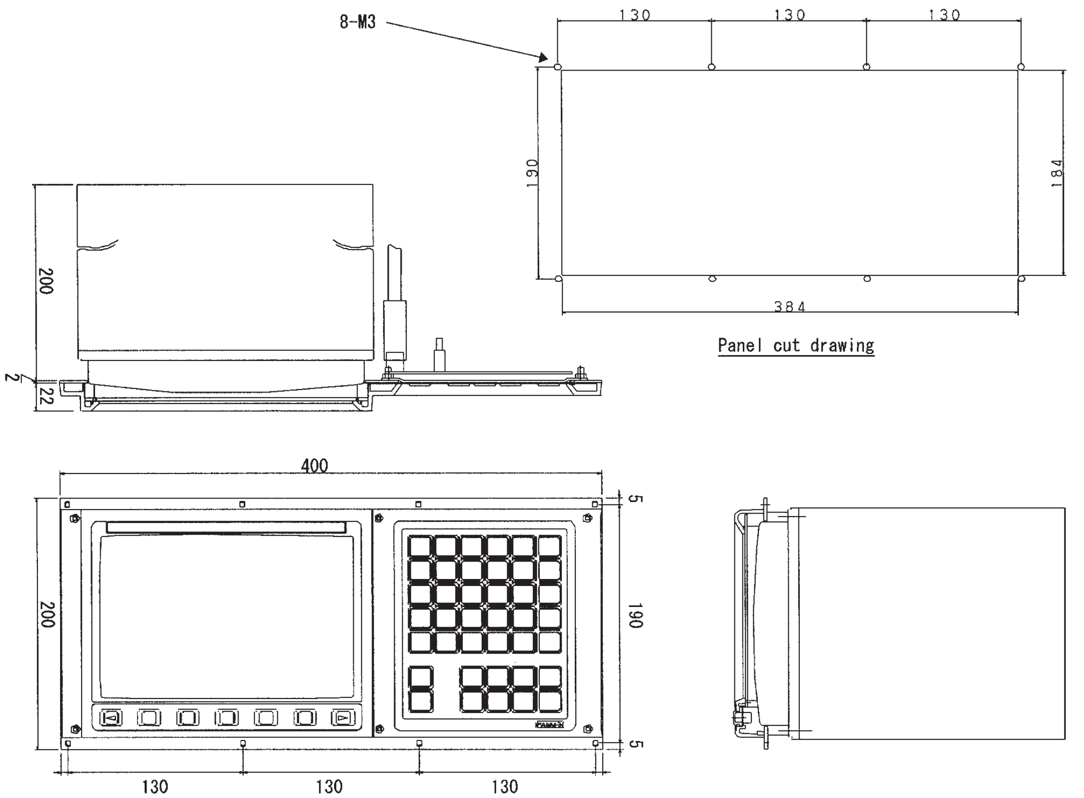
NOTE
1 The above figure shows a configuration in which an ISA expansion unit is added to a 6–slot control unit.
2 An ISA expansion unit is always added to the left side of the control unit.

E EXTERNAL DIMENSIONS CRT/MDI UNIT

Name	Number of Figure
9″ monochrome CRT/MDI (small size, horizontal type)	Fig.1
9″monochrome CRT/MDI (standard size, vertical type)	Fig.2
9″ monochrome CRT/MDI (standard size, horizontal type)	Fig.3
9″ monochrome CRT (separate type)	Fig.4
9″ PDP/MDI (small size)	Fig. 5
9″ PDP/MDI (standard size)	Fig. 6
9″ PDP (separate type)	Fig. 7
9.5″ LCD/MDI (vertical type), 10.4″ LCD/MDI (vertical type)	Fig. 8
9.5″ LCD/MDI (horizontal type), 10.4″ LCD/MDI (horizontal type)	Fig.9
14″ CRT/MDI (vertical type)	Fig. 10
14″ CRT/MDI (horizontal type)	Fig. 11
Separate MDI (for 9″ CRT or 9″ PDP)	Fig. 12
Separate MDI (vertical type for 10.4″ LCD)	Fig. 13
Separate MDI (horizontal type for 10.4″ LCD)	Fig. 14
10.4″ color LCD (separate type)	Fig. 15
9.5″ monochrome LCD (separate type)	Fig. 16

Fig. 1 9″ MONOCHROME CRT/MDI (SMALL SIZE, HORIZONTAL TYPE)

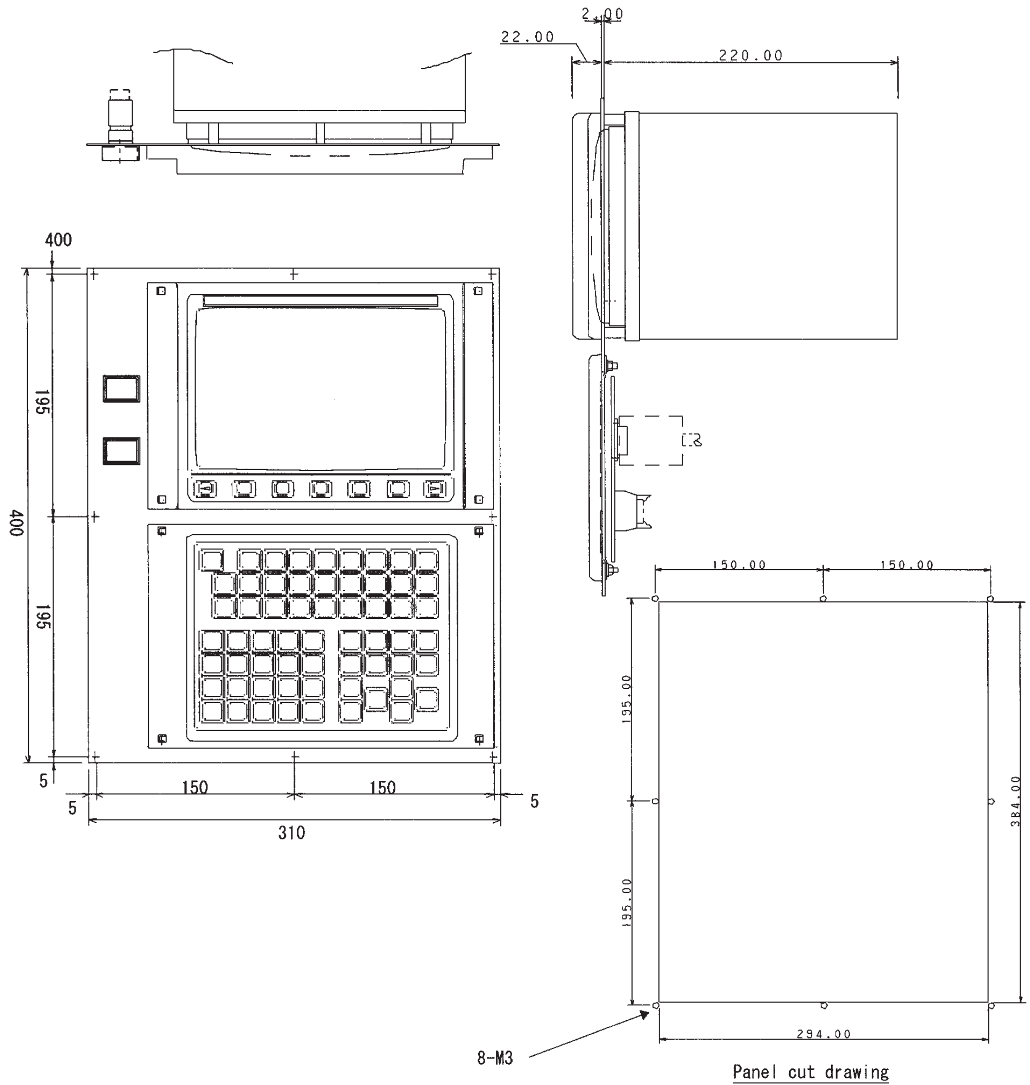
Specification : A02B–0163–C301 (M series, English key)
A02B–0163–C302 (T series, English key)



Weight : 4.5kg

Fig. 2 9″ MONOCHROME CRT/MDI (STANDARD SIZE, VERTICAL TYPE)

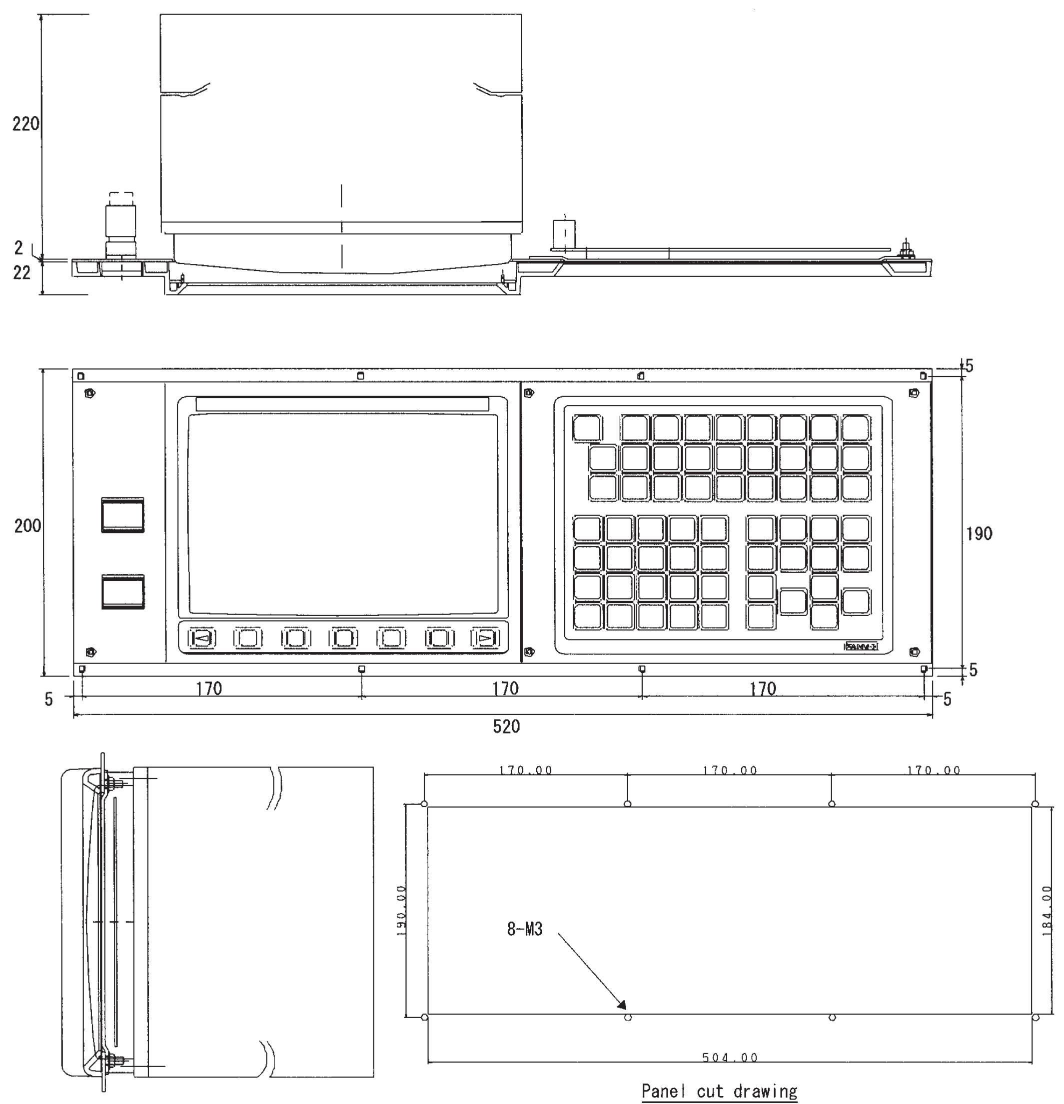
Specification : A02B–0163–C244 (English key)
A02B–0163–C444 (Symbolic key)



Weight : 5 kg

Fig. 3 9″ MONOCHROME CRT/MDI (STANDARD SIZE, HORIZONTAL TYPE)

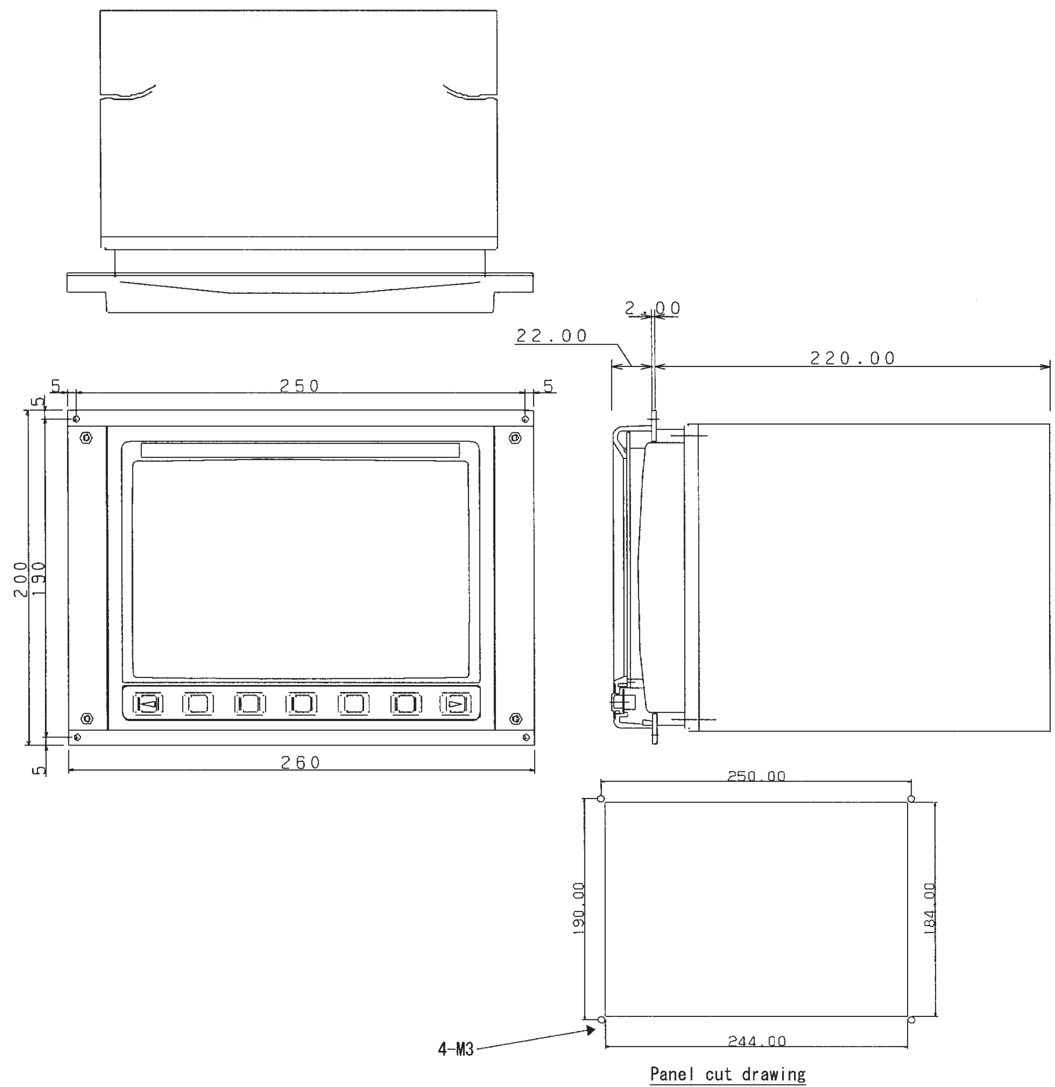
Specification : A02B–0163–C245 (English key)
A02B–0163–C445 (Symbolic key)



Weight : 4.5 kg

Fig. 4 9″ MONOCHROME CRT (SEPARATE TYPE)

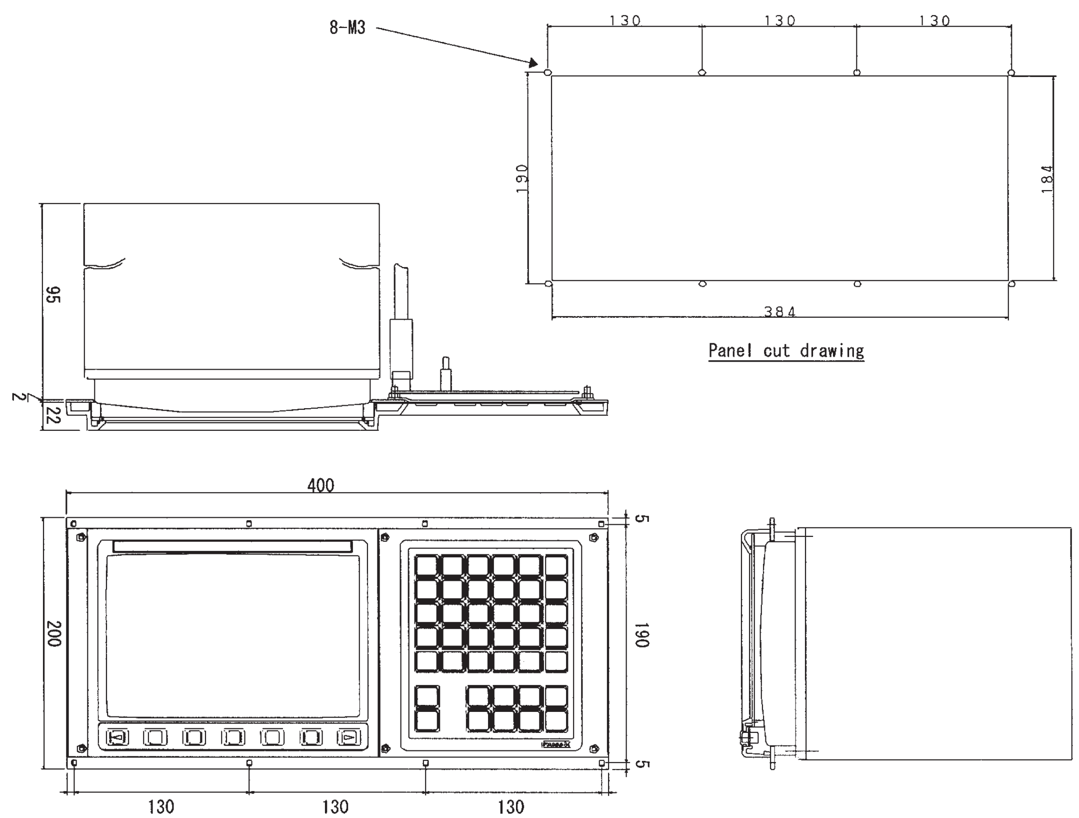
Specification : A02B–0162–C046



Weight : 3.5 kg

Fig. 5 9″ PDP/MDI (SMALL SIZE)

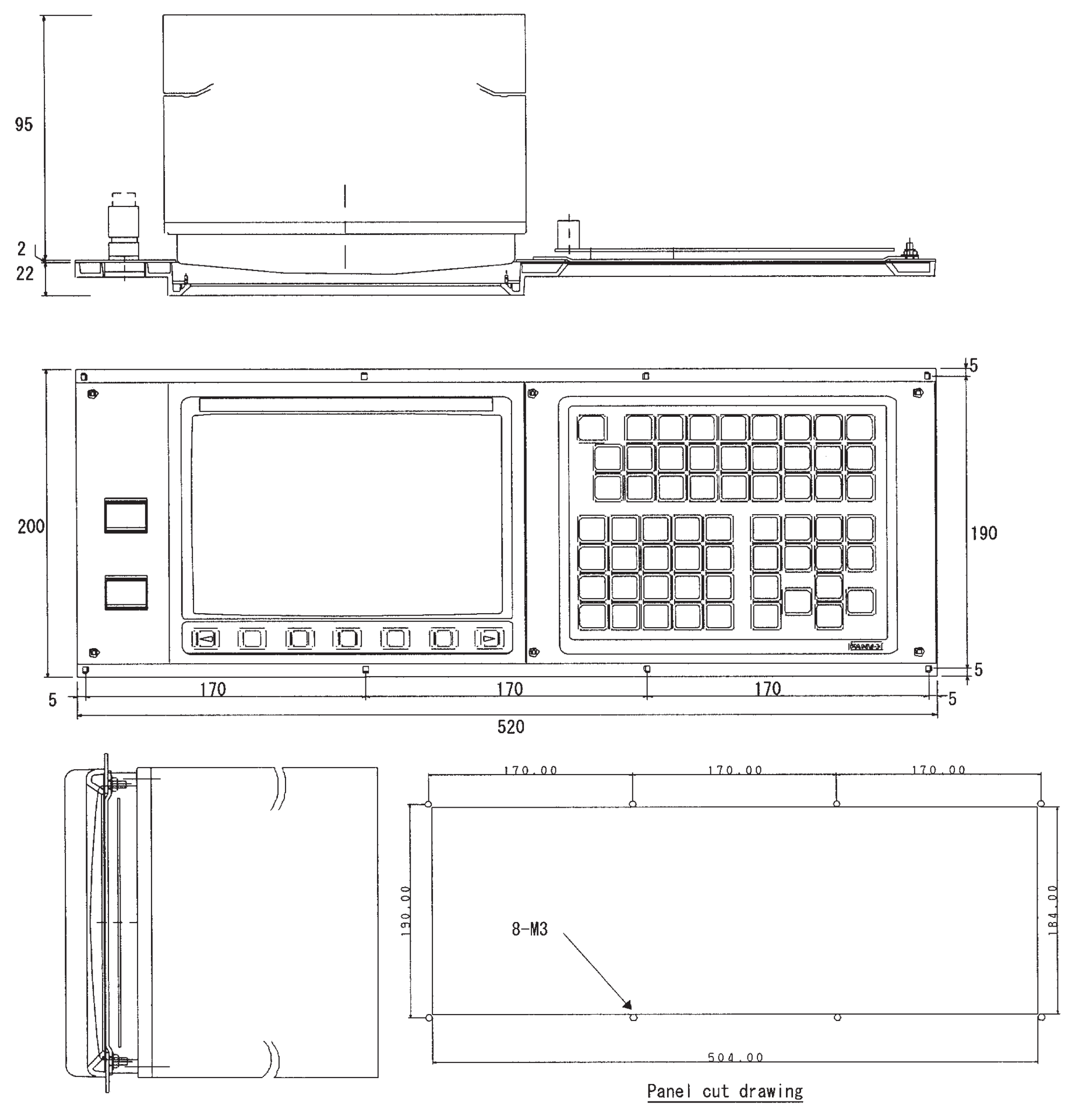
Specification : A02B–0163–C305 (M series, English key)
A02B–0163–C306 (T series, English key)



Weight : 3 kg

Fig. 6 9″ PDP/MDI (STANDARD SIZE)

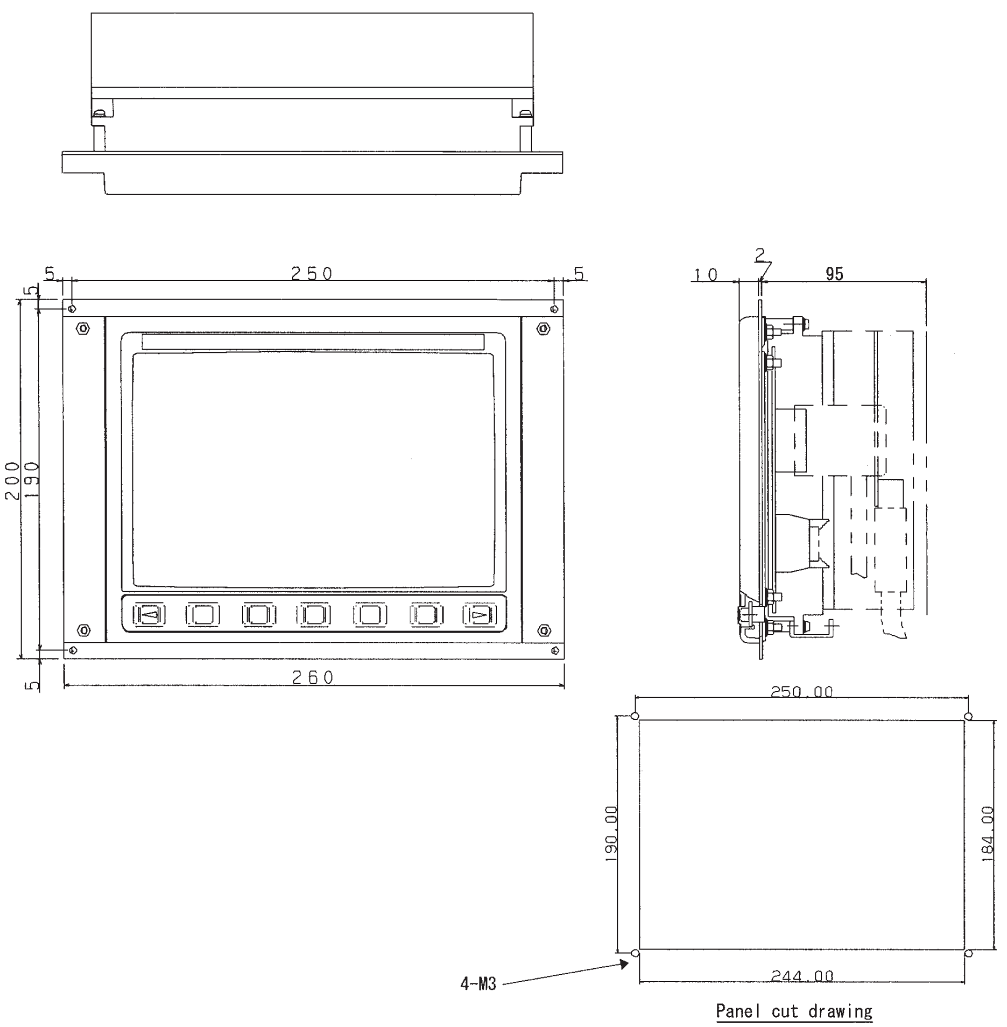
Specification : A02B–0163–C265 (English key)
A02B–0163–C465 (Symbolic key)



Weight : 3.5 kg

Fig. 7 9″ PDP (SEPARATE TYPE)

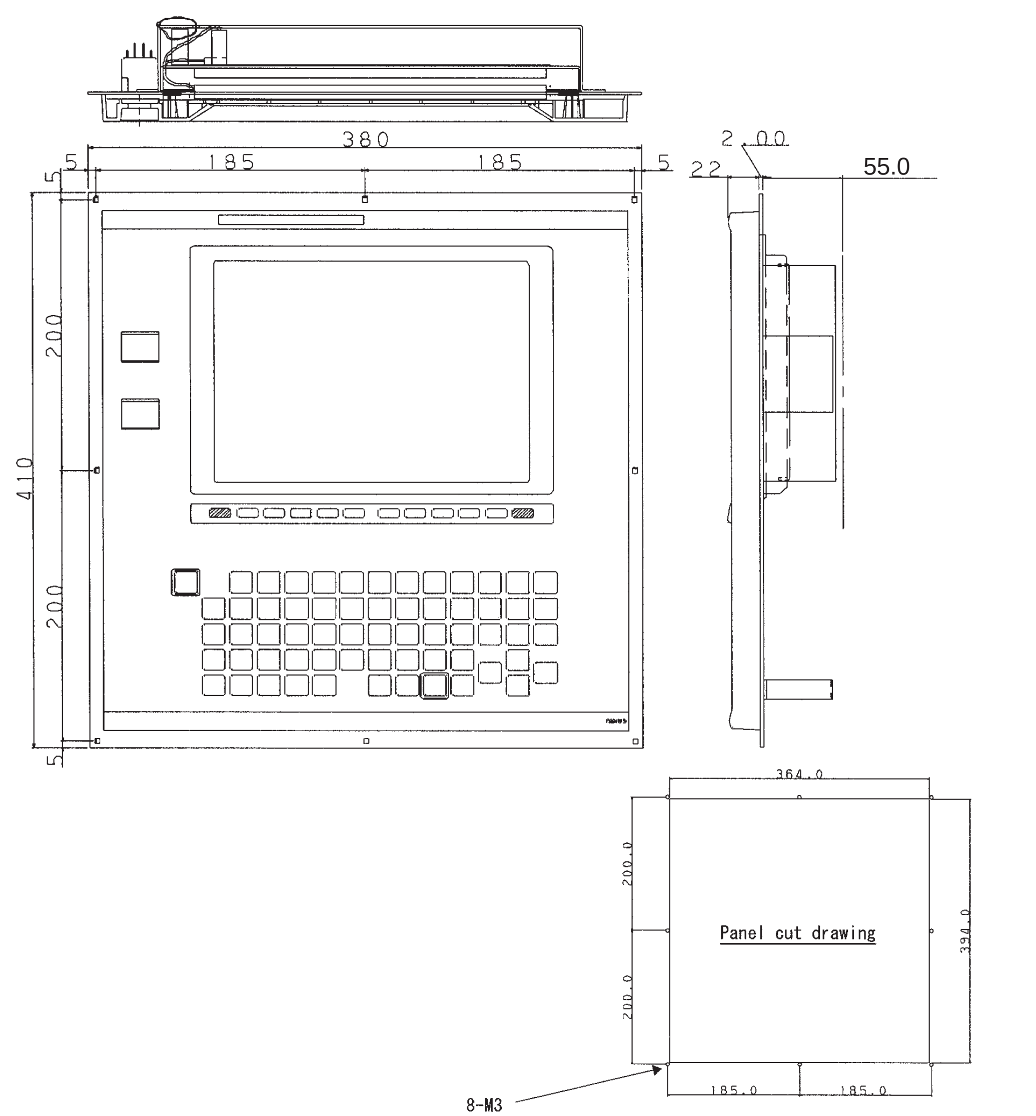
Specification : A02B–0163–C268



Weight : 2.5 kg

Fig. 8 9.5″ LCD/MDI (VERTICAL TYPE), 10.4″ LCD/MDI (VERTICAL TYPE)

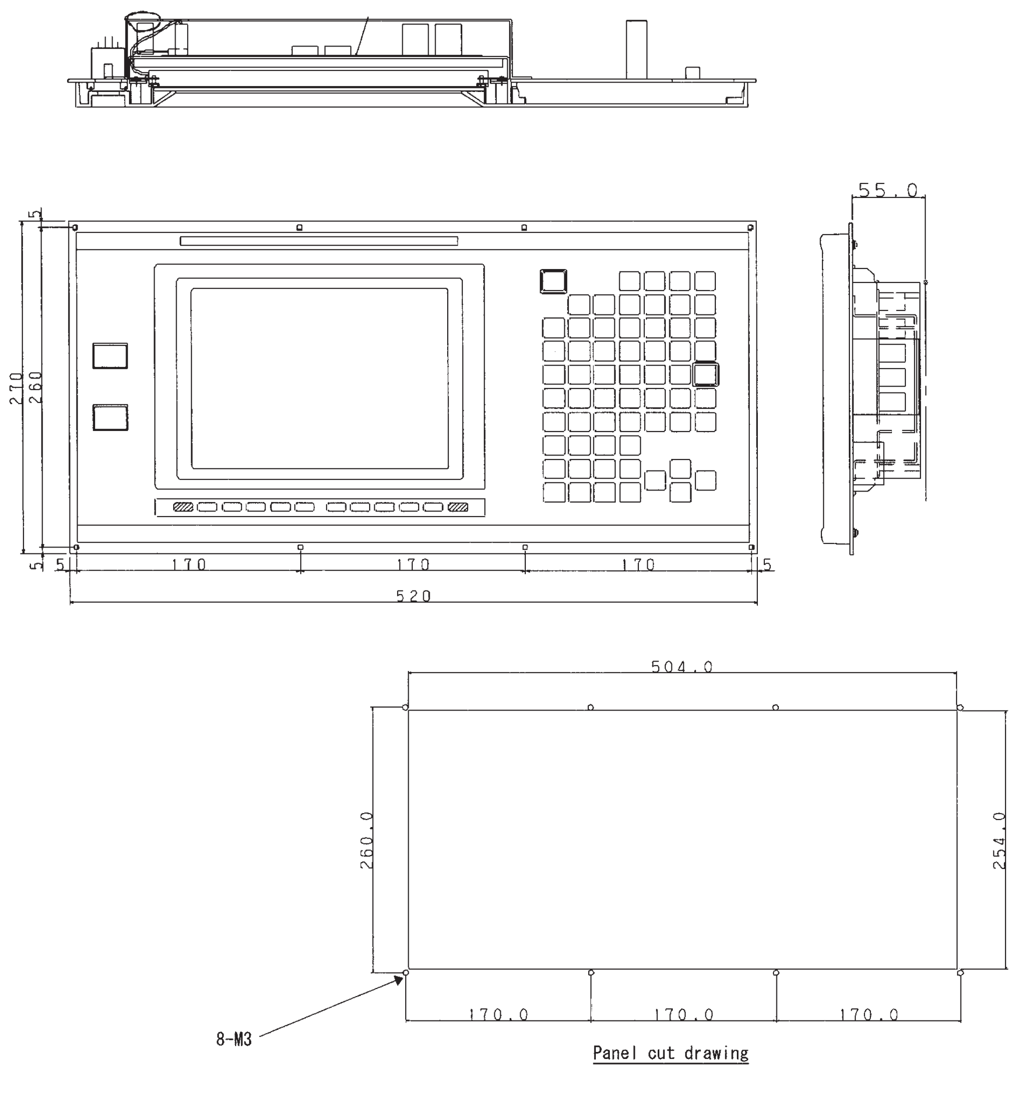
Specification : A02B–0163–C331 (English key for MB/MFB/TB/TTB/MEL/TEE)
A02B–0163–C333 (English key for TFB/TTFB/TEF)
A02B–0163–C341 (English key for MMC–IV)
A02B–0163–C371 (English key for MB/MFB/TB/TTB/MEL/TEE)
A02B–0163–C571 (Symbolic key for MB/MFB/TB/TTB/MEL/TEE)
A02B–0163–C381 (English key for MMC–IV)
A02B–0163–C581 (Symbolic key for MMC–IV)



Weight : 5.5 kg

Fig. 9 9.5″ LCD/MDI (HORIZONTAL TYPE), 10.4″ LCD/MDI (HORIZONTAL TYPE)

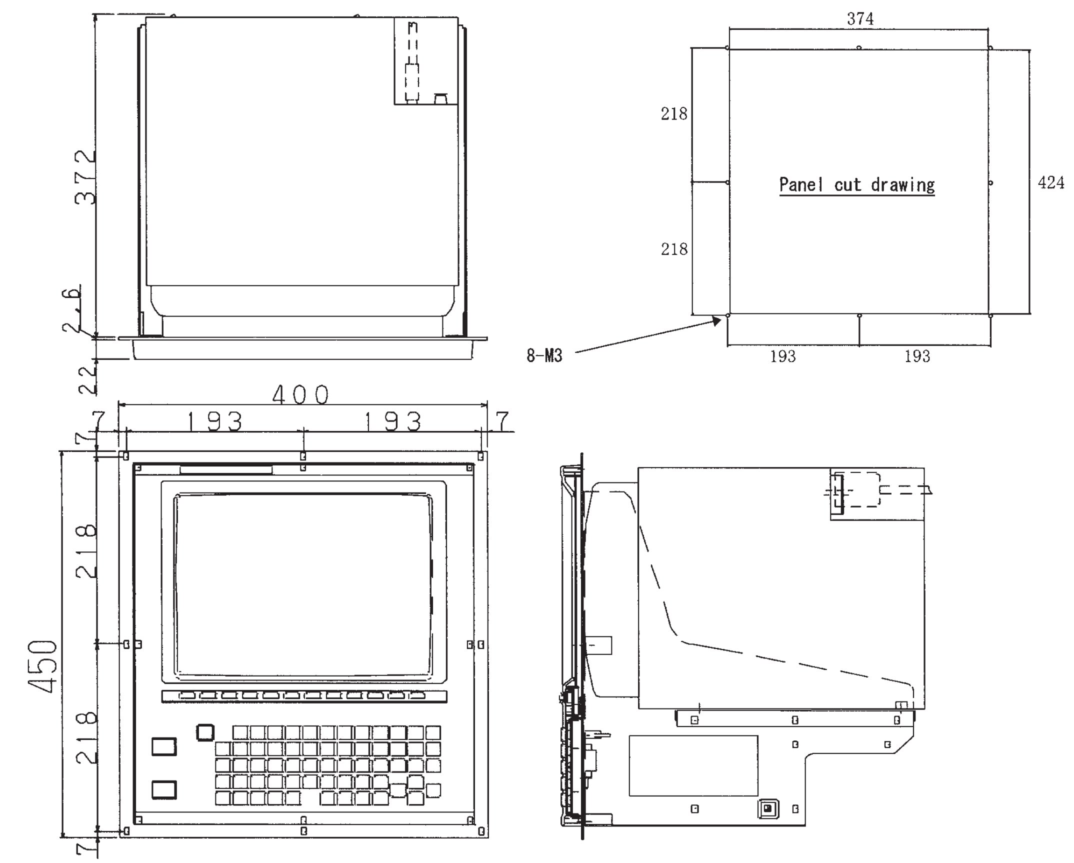
Specification : A02B–0163–C332 (English key for MB/MFB/TB/TTB/MEL/TEE)
A02B–0163–C334 (English key for TFB/TTFB/TEF)
A02B–0163–C342 (English key for MMC–IV)
A02B–0163–C372 (English key for MB/MFB/TB/TTB/MEL/TEE)
A02B–0163–C572 (Symbolic key for MB/MFB/TB/TTB/MEL/TEE)
A02B–0163–C382 (English key for MMC–IV)
A02B–0163–C582 (Symbolic key for MMC–IV)



Weight : 5.0 kg

Fig. 10 14″ CRT/MDI (VERTICAL TYPE)

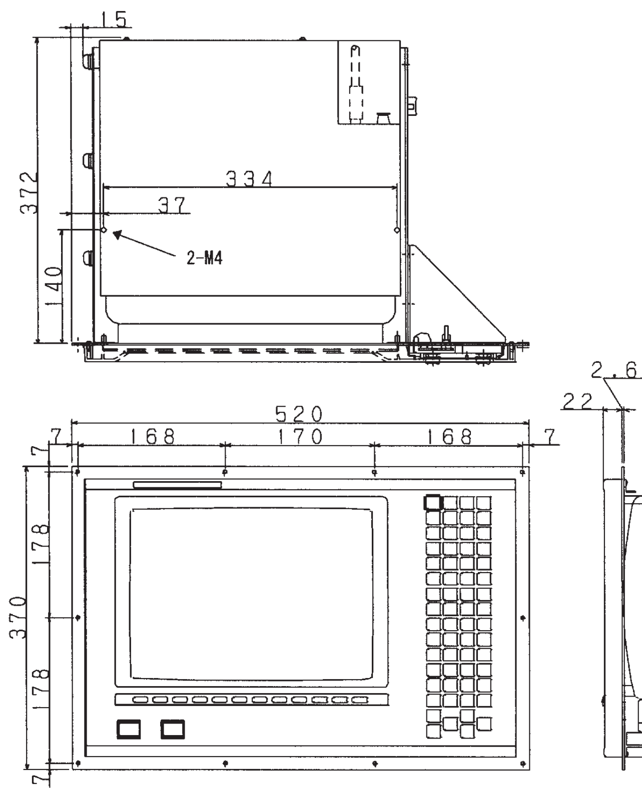
Specification : A02B–0163–C321 (English key for MB/MFB/TB/TTB/MEL/TEE)
A02B–0163–C323 (English key for TFB/TTFB/TEF)
A02B–0163–C523 (Symbolic key for TFB/TTFB/TEF)



Weight : 20.5 kg

Fig. 11 14″ CRT/MDI (HORIZONTAL TYPE)

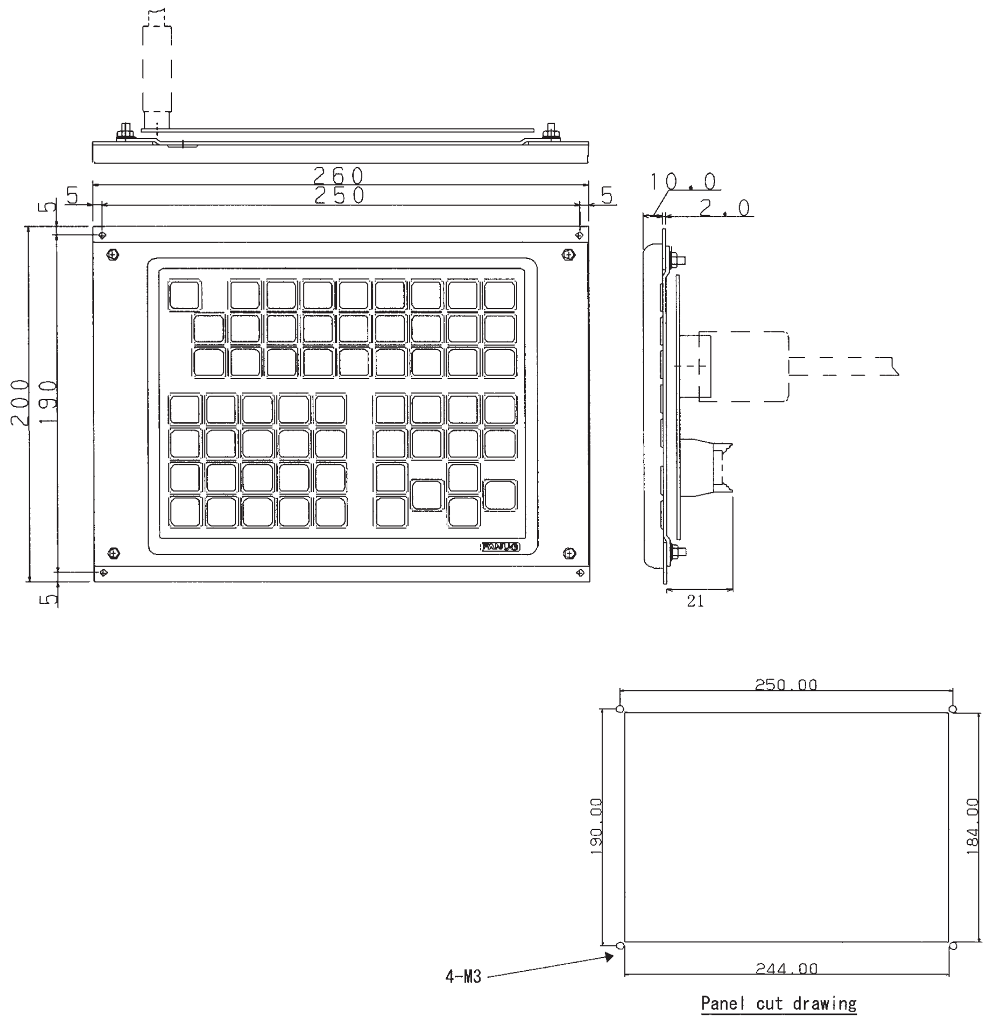
Specification : A02B–0163–C322 (English key for MB/MFB/TB/TTB/MEL/TEE)
A02B–0163–C324 (English key for TFB/TTFB/TEF)
A02B–0163–C522 (Symbolic key for MB/MFB/TB/TTB/MEL/TEE)



Weight : 20.5 kg

Fig. 12 SEPARATE MDI (FOR 9″ CRT OR 9″ PDP)

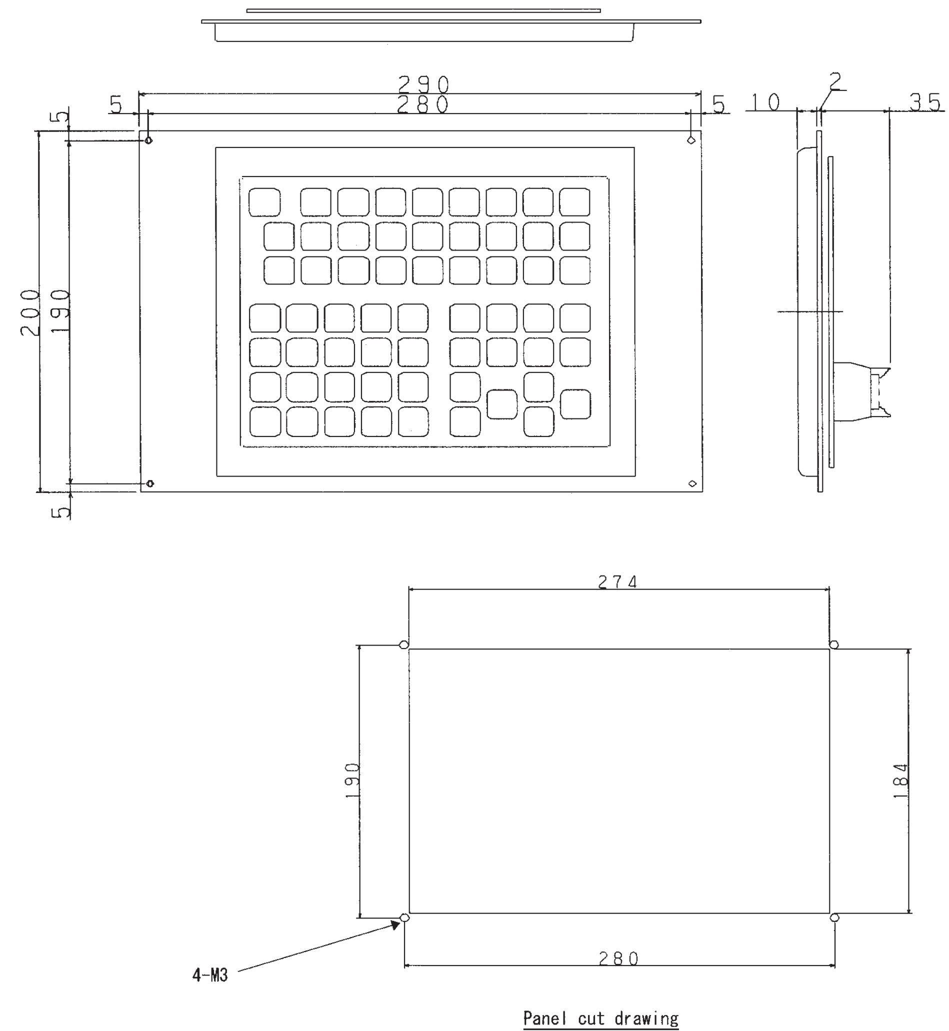
Specification : A02B–0163–C312 (English key)
A02B–0163–C313 (Symbolic key)



Weight : 1 kg

Fig. 13 SEPARATE MDI (VERTICAL TYPE FOR 10.4″ LCD)

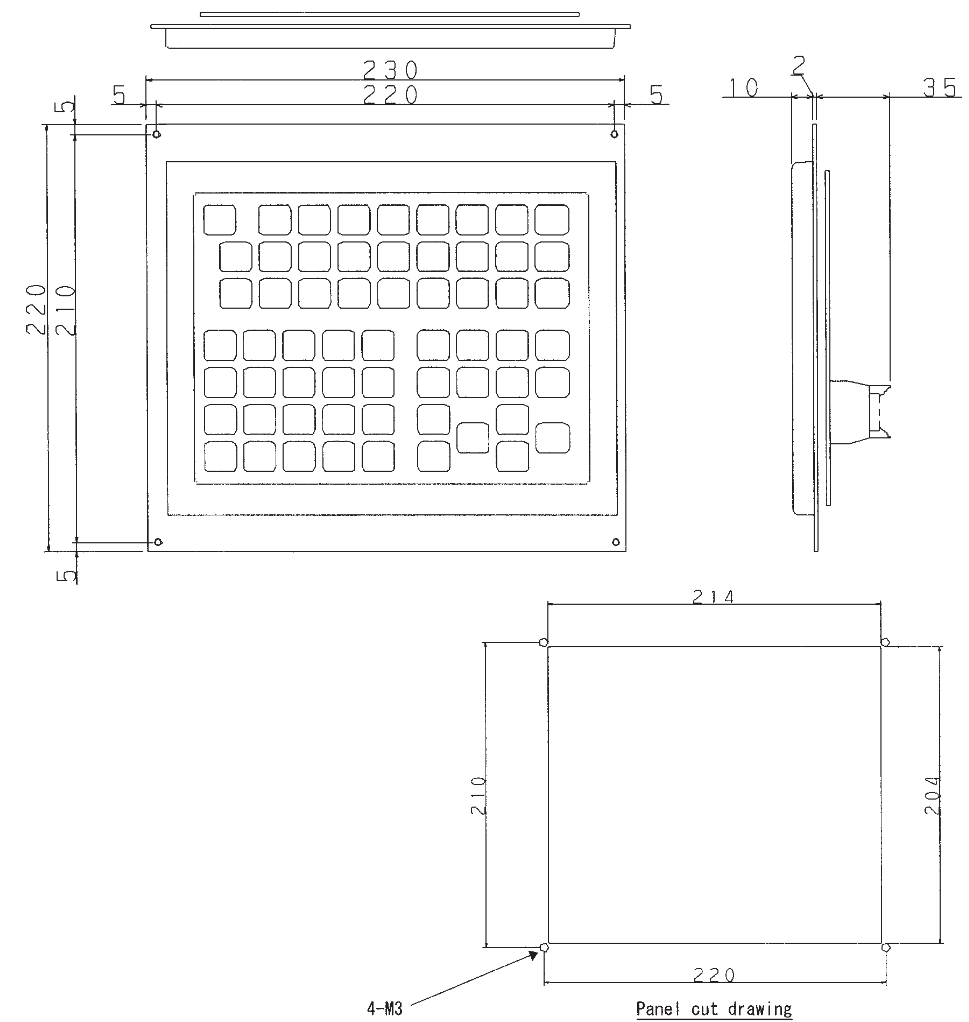
Specification : A02B–0163–C316 (English key)
A02B–0163–C318 (Symbolic key)



Weight : 1 kg

Fig. 14 SEPARATE MDI (HORIZONTAL TYPE FOR 10.4″ LCD)

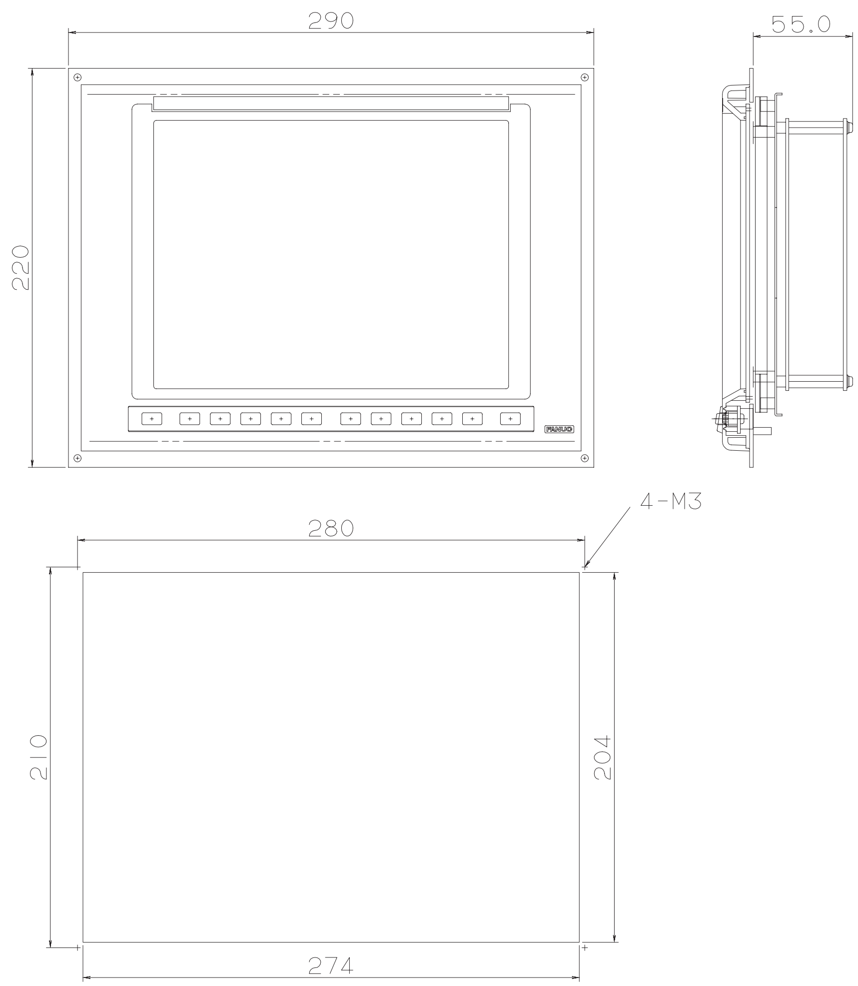
Specification : A02B–0163–C317 (English key)
A02B–0163–C319 (Symbolic key)



Weight : 1 kg

Fig. 15 10.4″ COLOR LCD (SEPARATE TYPE)

Specification : A02B–0222–C150

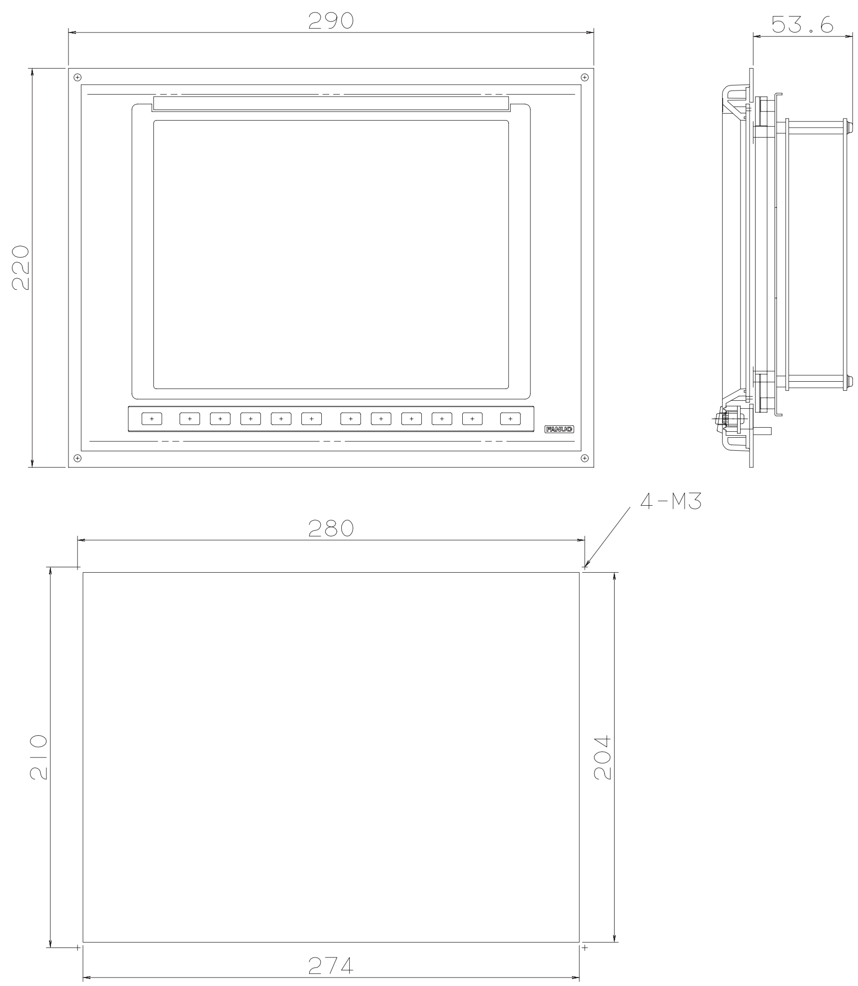


Panel cut drawing

Weight : 2.5 kg

Fig. 16 9.5″ MONOCHROME LCD (SEPARATE TYPE)

Specification : A02B–0222–C110



Panel cut drawing

Weight : 2.5 kg

F EXTERNAL DIMENSIONS OF EACH UNIT

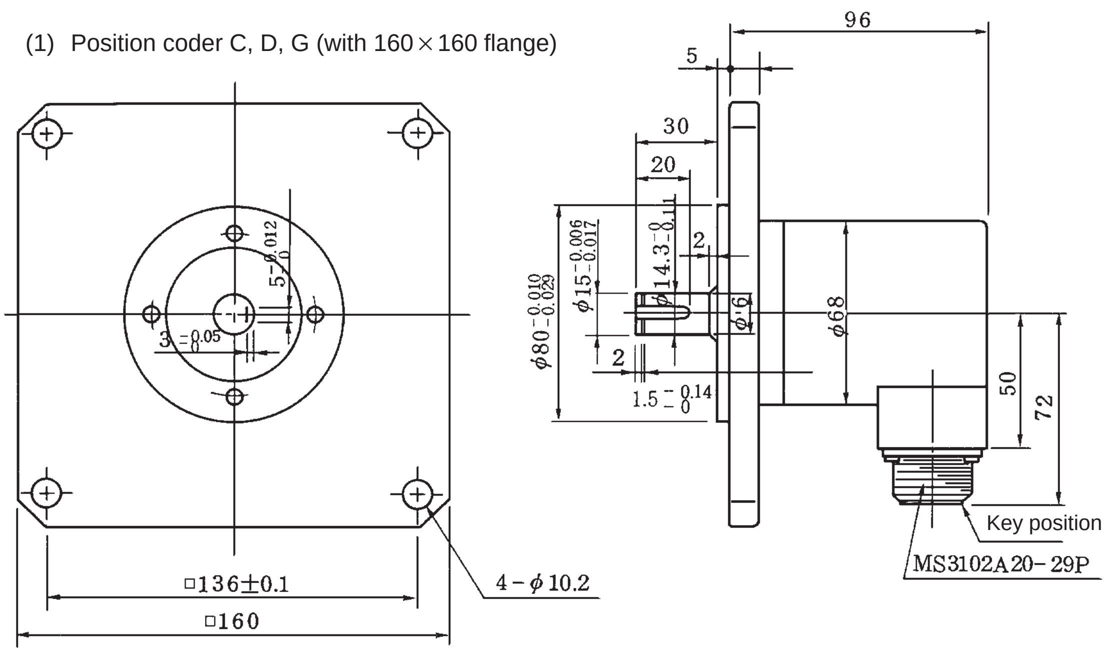
Name	Number of Figure
Position coder	Fig.1
Manual pulse generator	Fig.2
Pendant type manual pulse generator	Fig.3
Battery case for separate type absolute pulse coder	Fig.4
Punch panel (wide width type)	Fig. 5
Punch panel (narrow width type)	Fig. 6
Portable tape reader without reels	Fig. 7
Portable tape reader with reels	Fig. 8
Separate type tape reader without reels	Fig.9
Separate type tape reader with reels	Fig. 10

Fig. 1 POSITION CODER

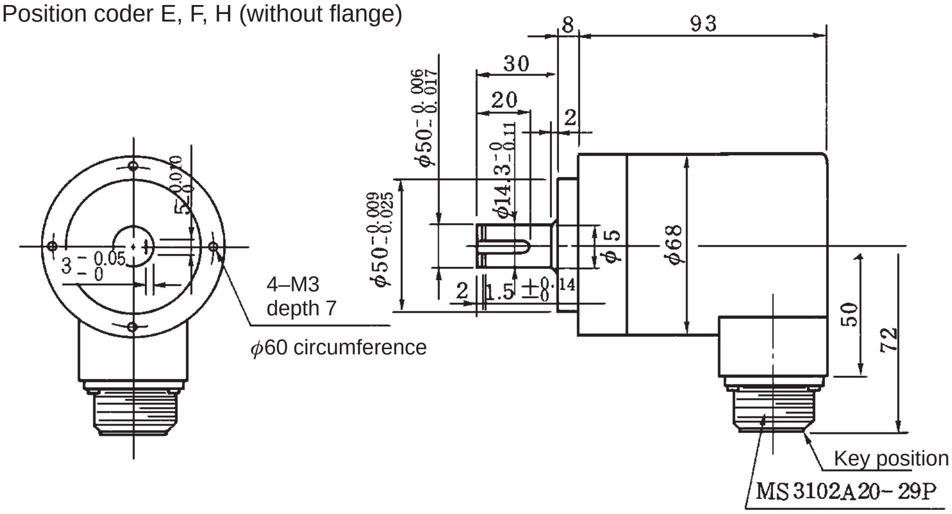
Specification :

Position coder C A76L–0027–0001#101 (Max. 4000rpm, with 160 × 160 flange)
Position coder D A76L–0027–0001#001 (Max. 6000rpm, with 160 × 160 flange)
Position coder G A76L–0027–0001#201 (Max. 8000rpm, with 160 × 160 flange)
Position coder J A76L–0027–0001#102 (Max. 4000rpm, with 68 × 68 flange)
Position coder K A76L–0027–0001#002 (Max. 6000rpm, with 68 × 68 flange)
Position coder L A76L–0027–0001#202 (Max. 8000rpm, with 68 × 68 flange)
Position coder E A76L–0027–0001#103 (Max. 4000rpm, without flange)
Position coder F A76L–0027–0001#003 (Max. 6000rpm, without flange)
Position coder H A76L–0027–0001#203 (Max. 8000rpm, without flange)

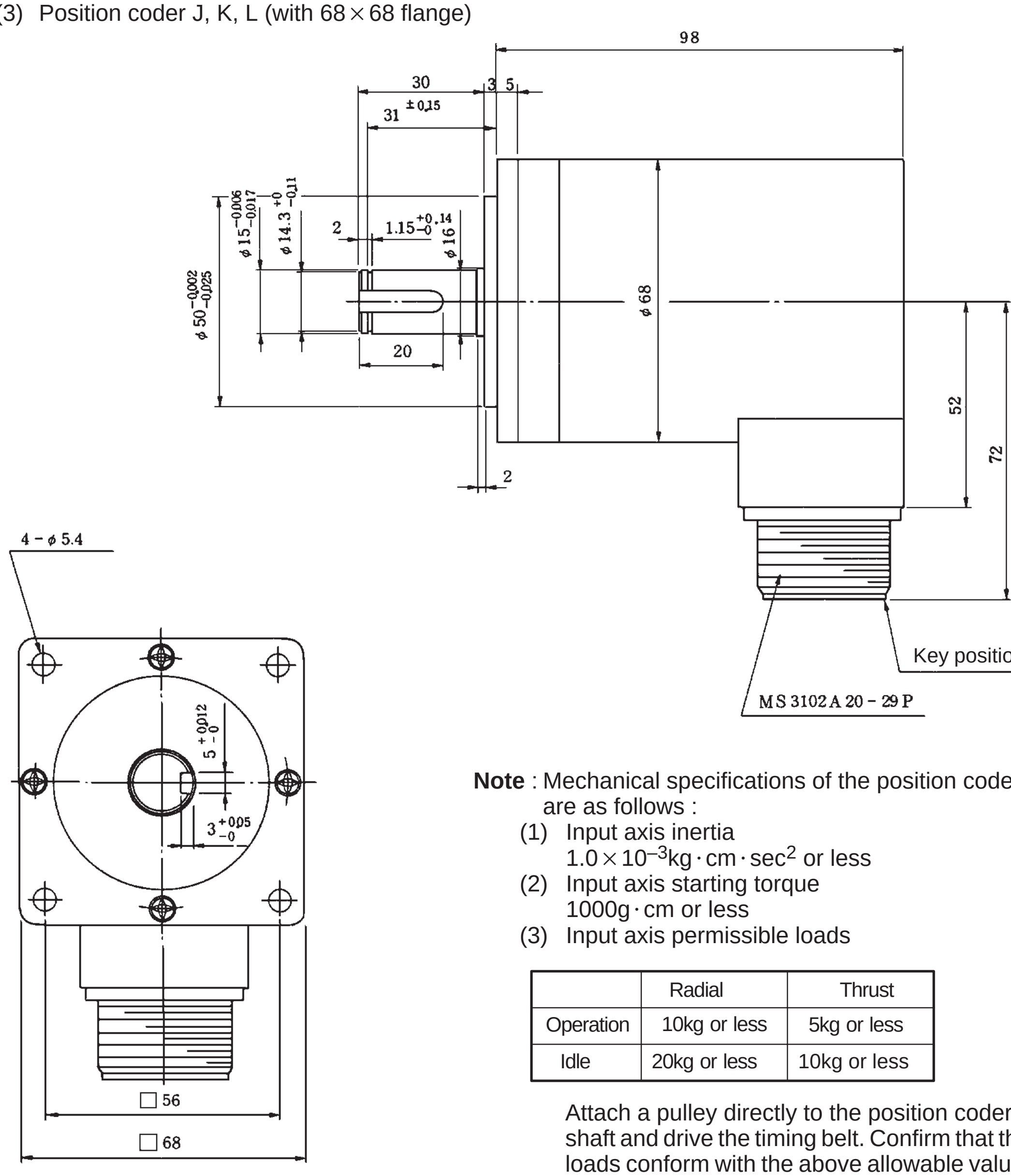
(1) Position coder C, D, G (with 160 × 160 flange)



(2) Position coder E, F, H (without flange)



(3) Position coder J, K, L (with 68 × 68 flange)



Note : Mechanical specifications of the position coder are as follows :

(1) Input axis inertia
1.0×10^{-3}kg · cm · sec^2 or less

(2) Input axis starting torque
1000g · cm or less

(3) Input axis permissible loads

	Radial	Thrust
Operation	10kg or less	5kg or less
Idle	20kg or less	10kg or less

Attach a pulley directly to the position coder shaft and drive the timing belt. Confirm that the loads conform with the above allowable value.

Fig. 2 MANUAL PULSE GENERATOR

Specification : A860–0202–T001

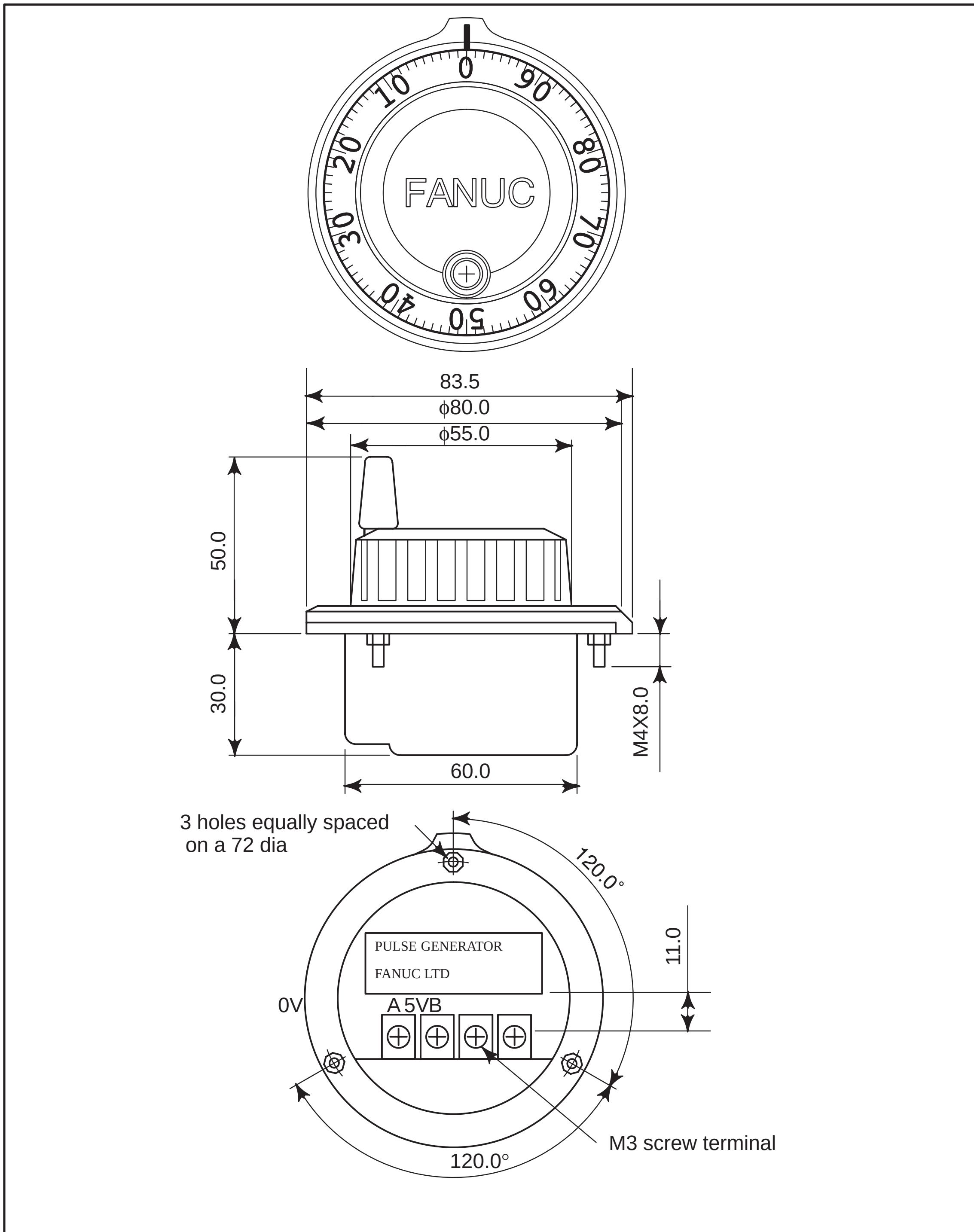
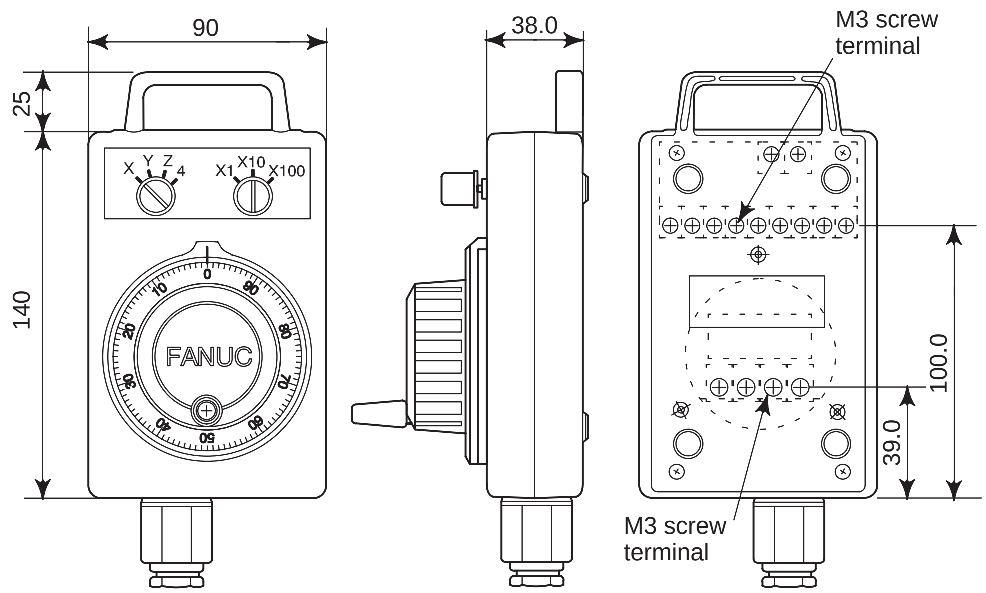


Fig. 3 PENDANT TYPE MANUAL PULSE GENERATOR

Specification : A860–0202–T004 to T015

A860–0202–T004 to T009



A860–0202–T010 to T015

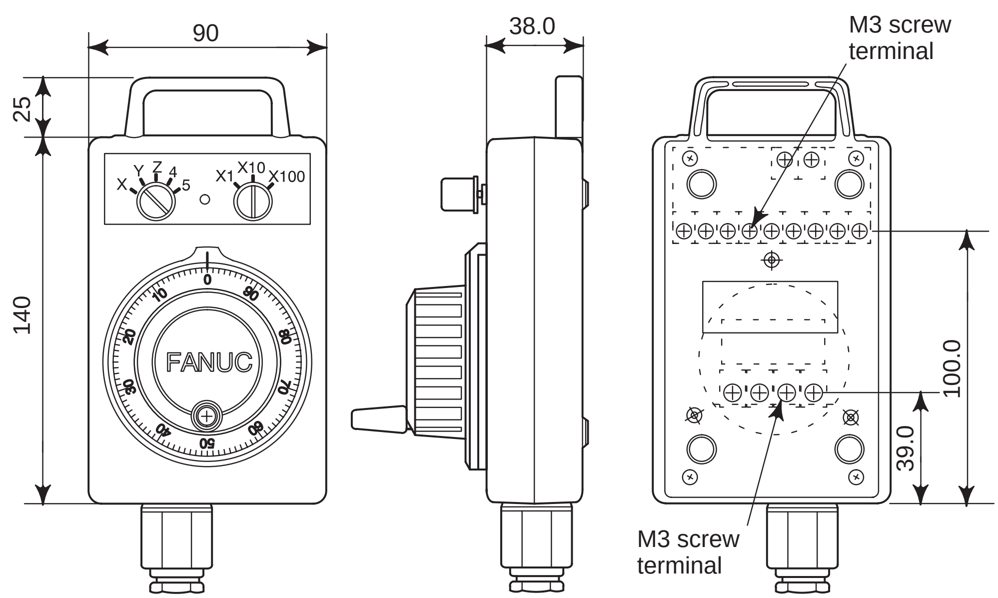
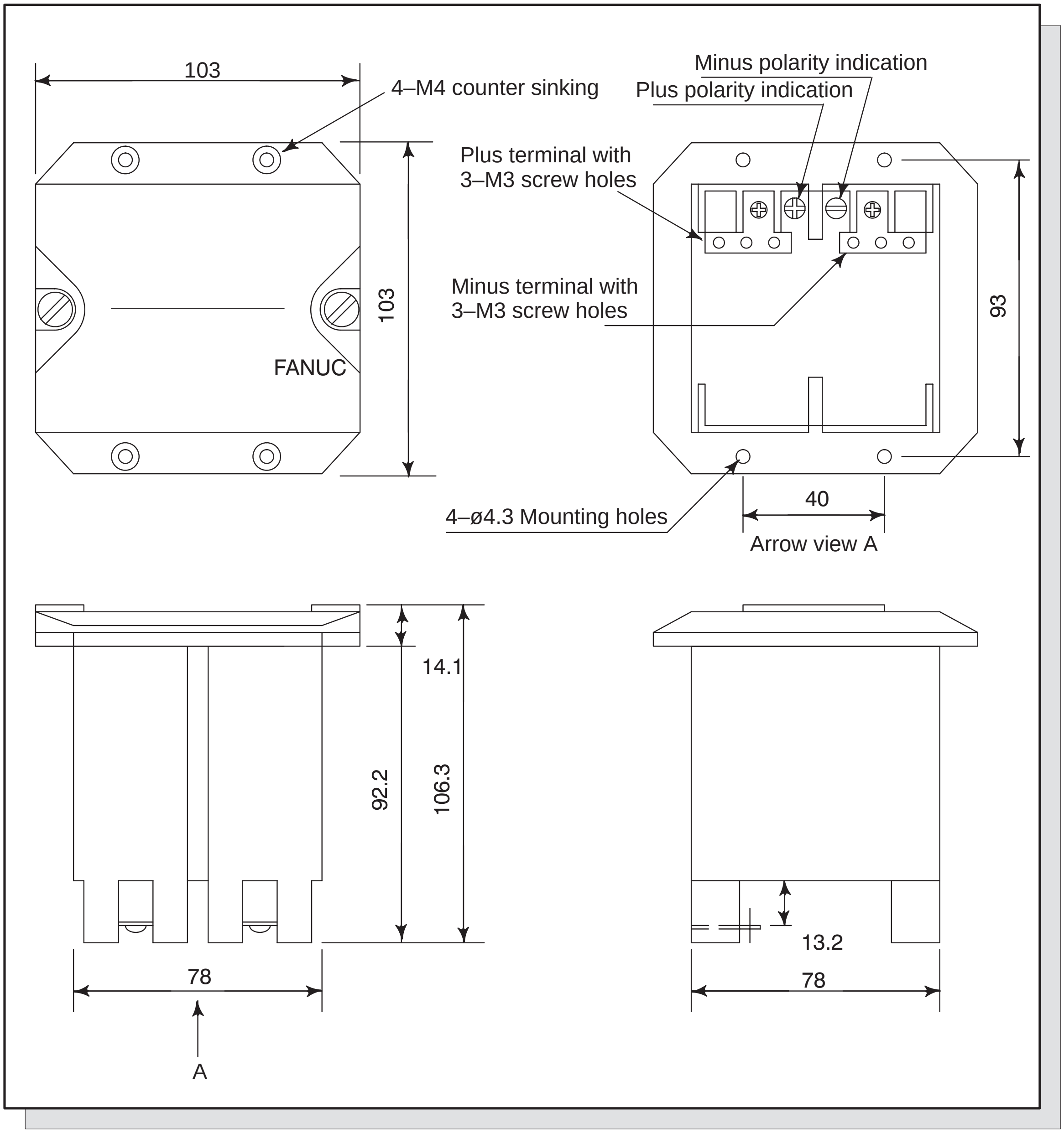


Fig. 4 BATTERY CASE FOR SEPARATE TYPE ABSOLUTE PULSE CODER

Specification : A06B–6050–K060



> **NOTE**
> Power supply is not included.

Fig. 5 PUNCH PANEL (WIDE WIDTH TYPE)

Specification : A02B–0120–C181 (Cable length : 1m)
A02B–0120–C182 (Cable length : 2m)
A02B–0120–C183 (Cable length : 5m)
A08B–0047–C051 (Cable length : 1m)
A08B–0047–C052 (Cable length : 2m)
A08B–0047–C053 (Cable length : 5m)

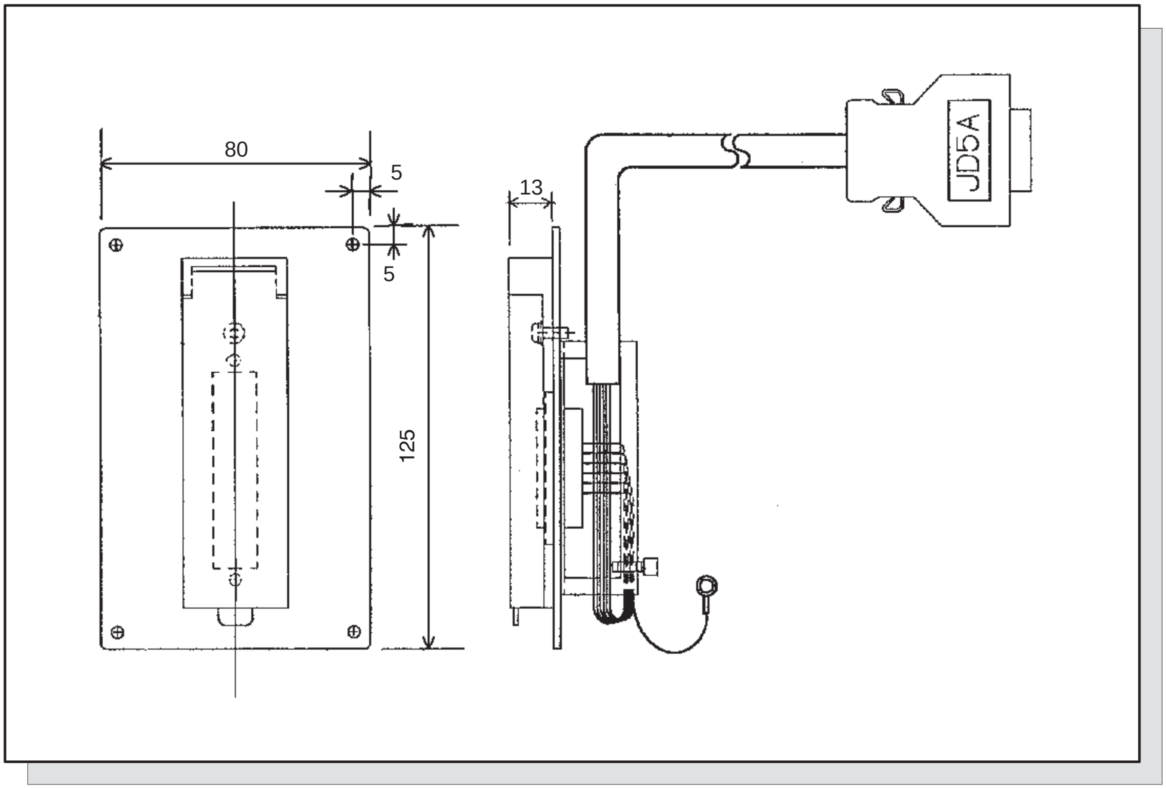


Fig. 6 PUNCH PANEL (NARROW WIDTH TYPE)

Specification : A02B–0120–C191 (Cable length : 1m)
A02B–0120–C192 (Cable length : 2m)
A02B–0120–C193 (Cable length : 5m)

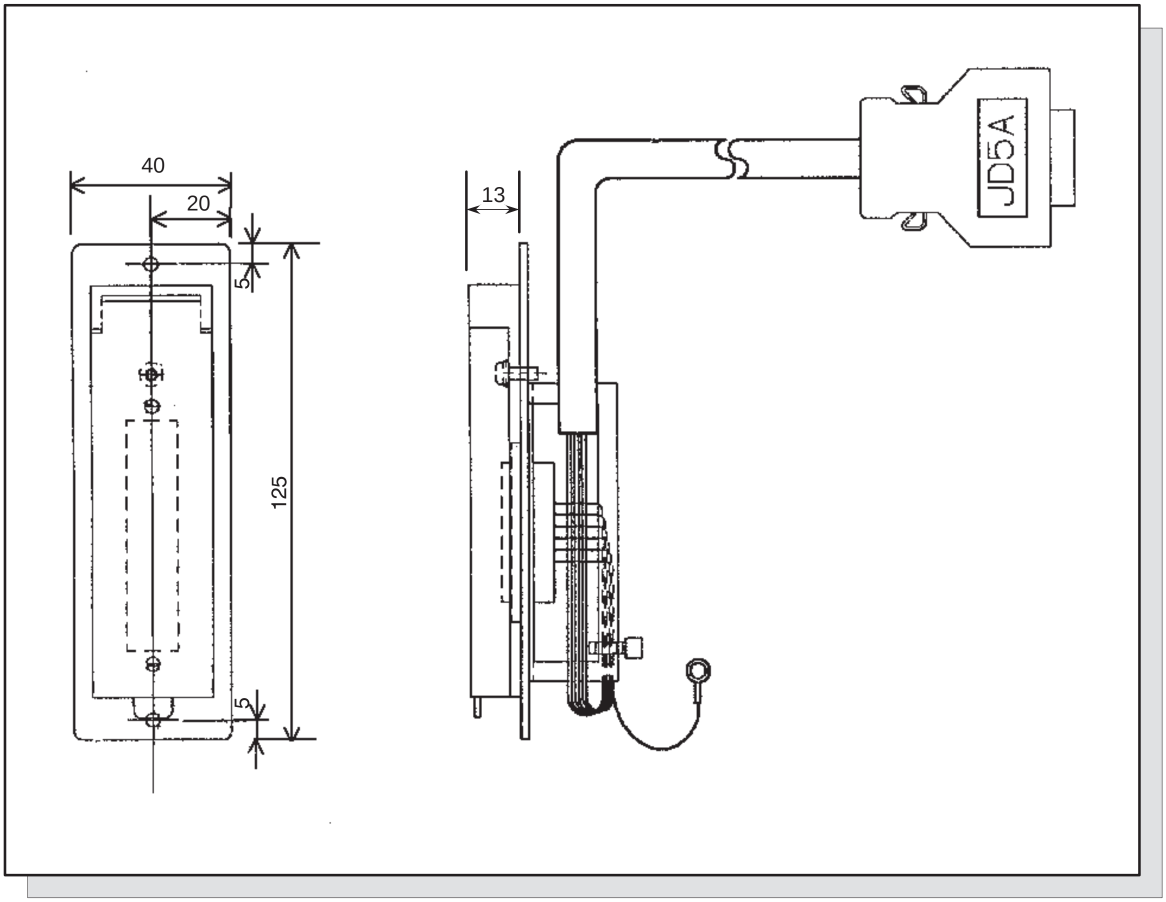
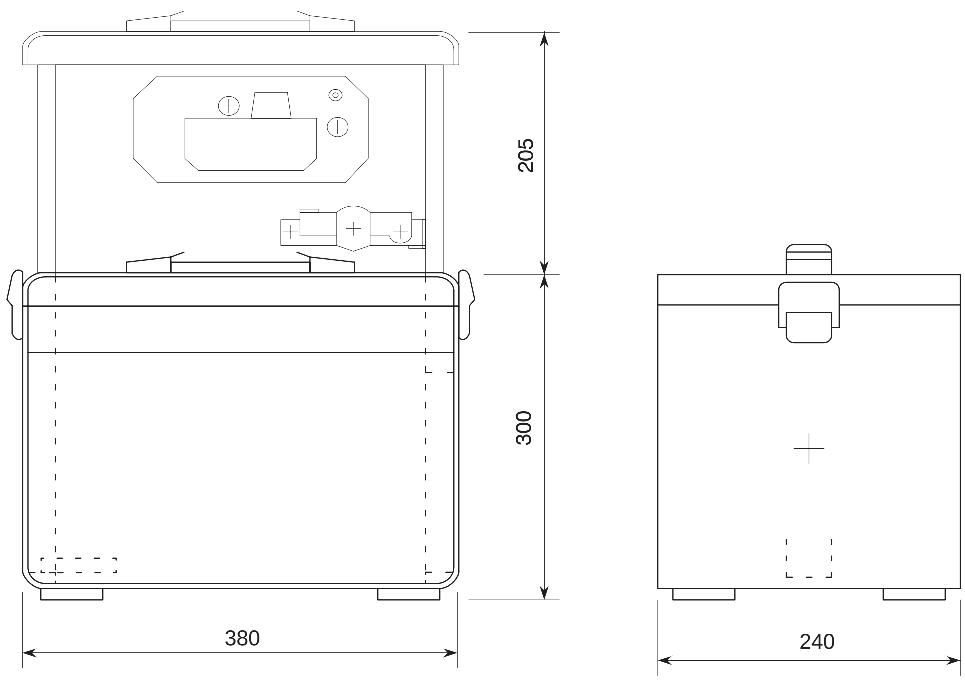


Fig. 7 PORTABLE TAPE READER WITHOUT REELS

Specification : A13B–0074–B001



Paint : Munsell No. 5GY3.5/0.5 leather tone finish
Weight : Applox. 15kg

Fig. 8 PORTABLE TAPE READER WITH REELS

Specification : A13B–0087–B001

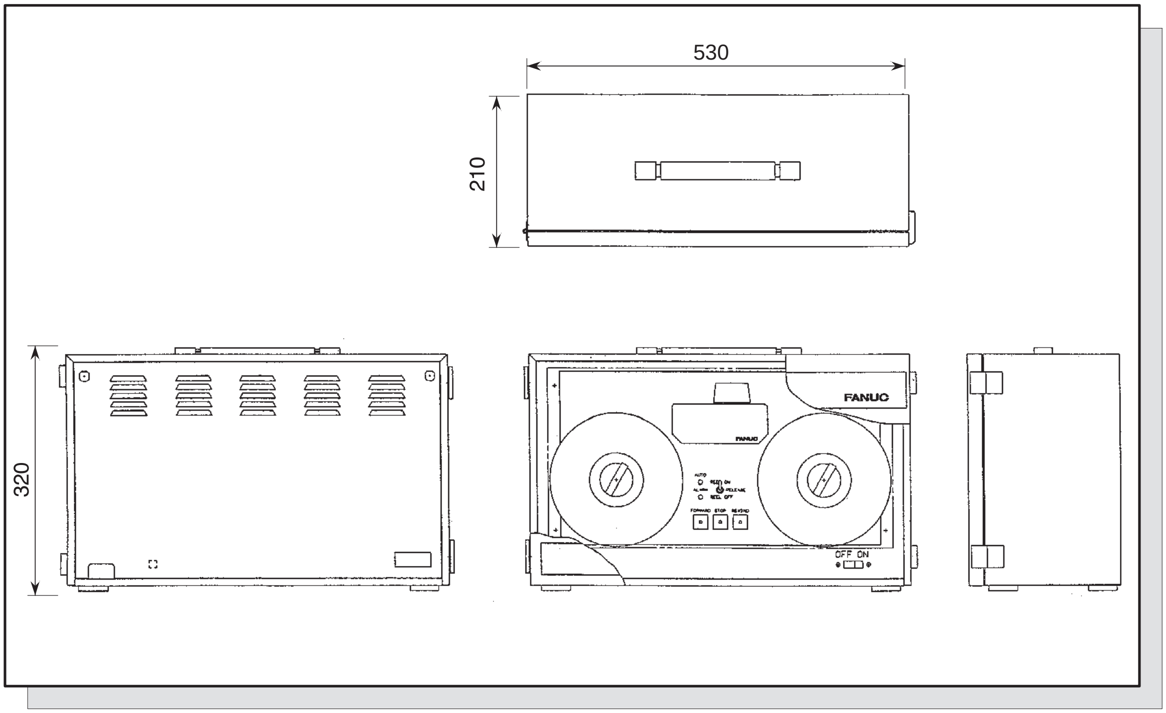
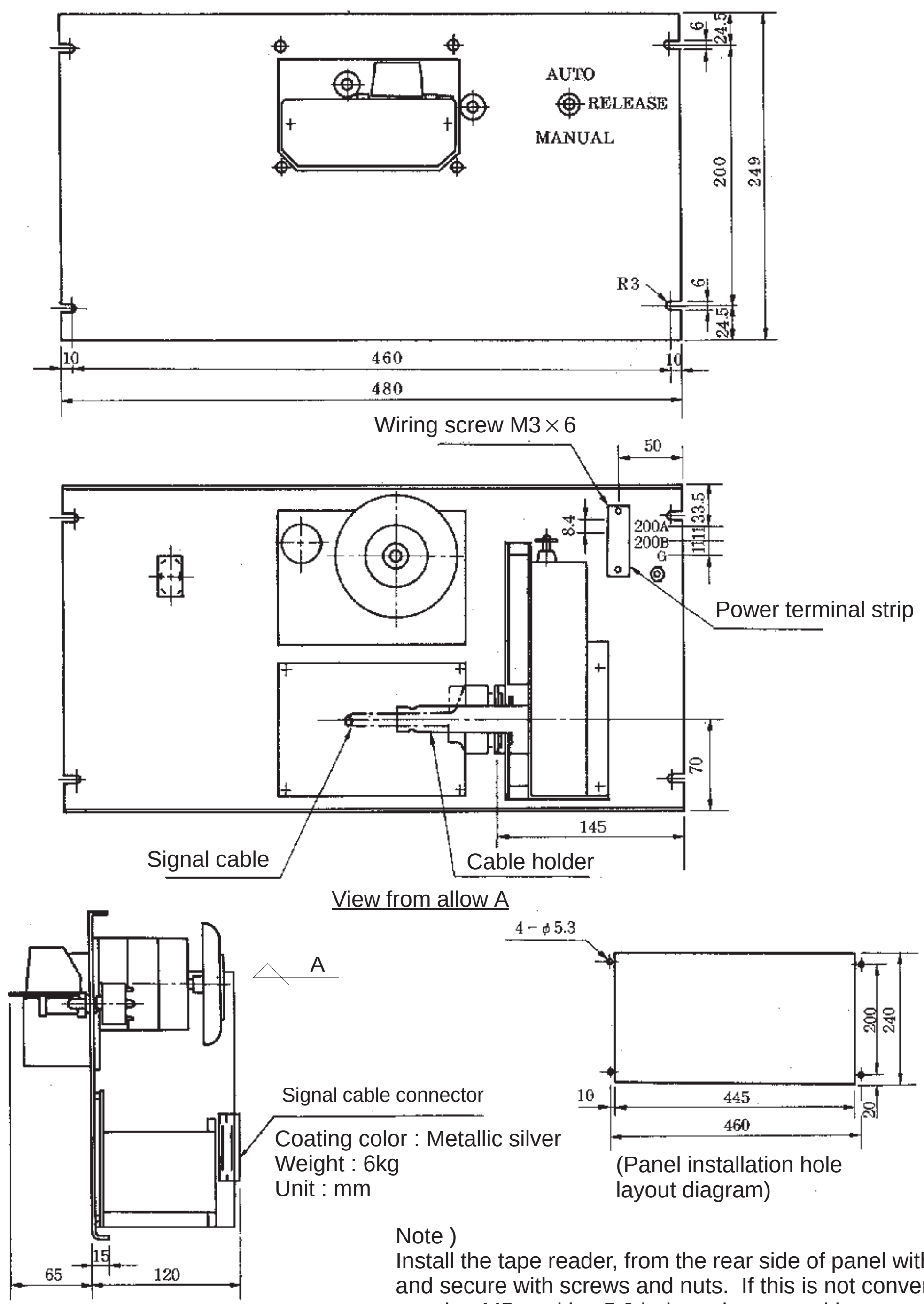


Fig. 9 SEPARATE TYPE TAPE READER WITHOUT REELS

Specification : A13B–0073–B001



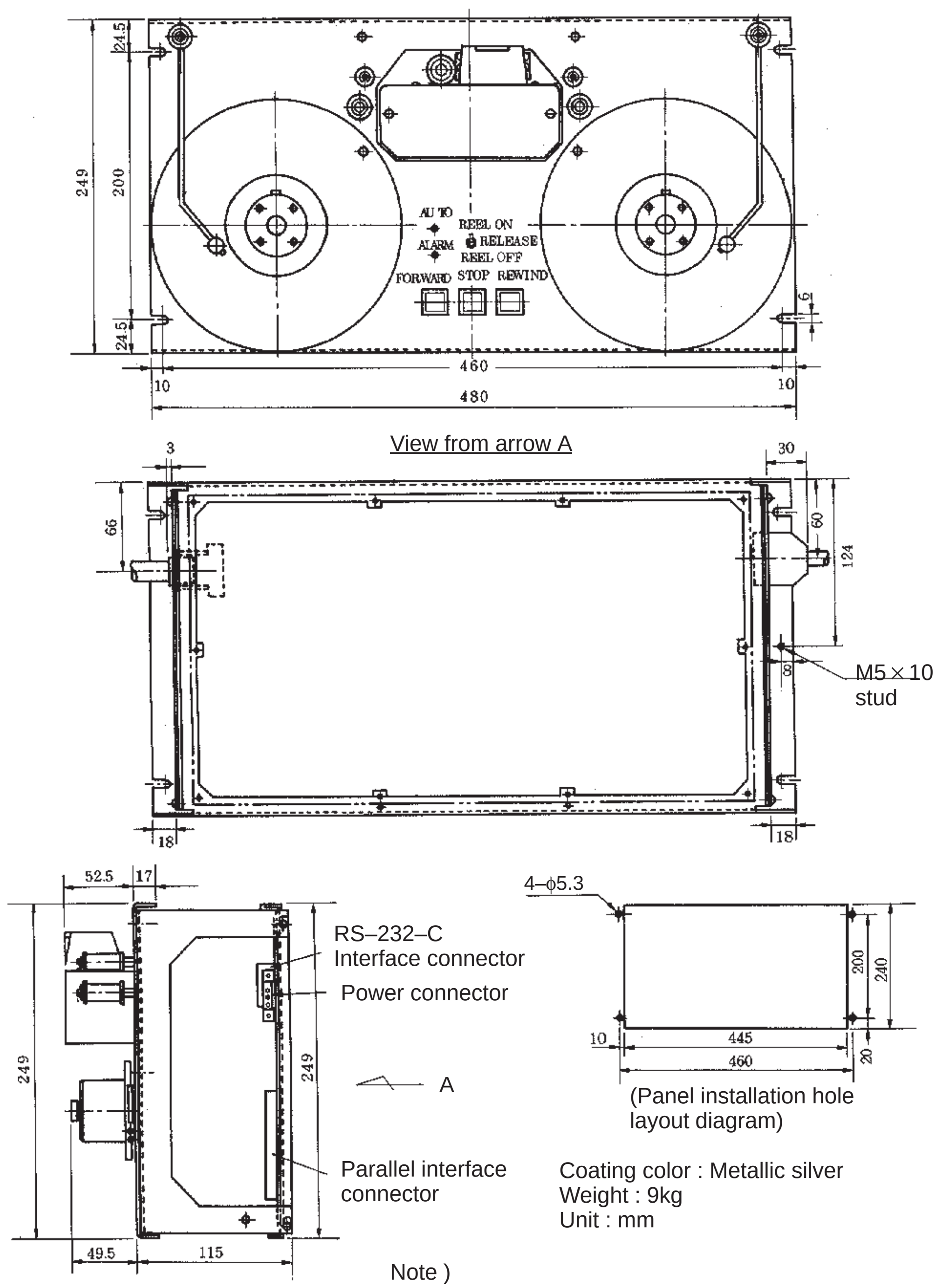
Coating color : Metallic silver
Weight : 6kg
Unit : mm

(Panel installation hole layout diagram)

Note)
Install the tape reader, from the rear side of panel with holes and secure with screws and nuts. If this is not convenient, attach a M5 stud in ϕ5.3 hole and secure with a nut.

Fig. 10 SEPARATE TYPE TAPE READER WITH REELS

Specification : A13B–0080–B001



Coating color : Metallic silver
Weight : 9kg
Unit : mm

Note)
Install the tape reader, from the rear side of panel with holes and secure with screws and nuts. If this is not convenient, attach a M5 stud in ϕ5.3 hole and secure with a nut.

G POWER SUPPLY AND HEAT LOSS

Unit	Power supply voltage	Power supply
3 slot control unit + PSU AI	170 to 264VAC	2A
4 slot control unit + PSU AI		2A
4 slot control unit + PSU BI		5A
6 slot control unit + PSU BI		5A
8 slot control unit + PSU BI		5A
14″ CRT/MDI unit	170 to 264VAC	0.6A
9″ CRT/MDI unit	24VDC ± 10% ± 10% includes momentary surges and ripples	0.8A
9″ PDP/MDI unit		2.0A
9.5″ LCD/MDI unit		0.8A
10.4″ LCD unit		0.8A
I/O Unit–A		Depends on the type and number of modules. Refer to “I/O Unit–MODEL A connection and Maintenance Manual” (B–61813E)

(PSU : Power supply unit)

Name		Heat-loss	
Control unit	Basic unit (4–slot)	60W	Power AI
	Basic unit (4–slot)	80W	Power BI
	Basic unit (6–slot)	80W	Power BI
	Basic unit (8–slot)	80W	Power BI
	Main CPU board	20W	
	PMC board	18W	
	Sub CPU board	18W	This board can not be used in 15MEK or 15MEL.
	Option 1 board	15W	
	RISC board	18W	This board can not be used in 15MEK or 15MEL.
	Buffer board (for multiple axis)	6W	
	Axis CPU	15W	
	MMC–II CPU board	20W	This board can not be used in 15MEK or 15MEL.
	MMC–II graphic board	20W	This board can not be used in 15MEK or 15MEL.
	MMC–III CPU board	20W	This board can not be used in 15MEK or 15MEL.
	MMC–IV CPU board	15W	This board can not be used in 15MEK or 15MEL.
	OSI/ethernet board	18W	This board can not be used in 15MEK or 15MEL.
	Data server board	18W	
CRT/MDI	9″ monochrome CRT/MDI	14W	For small and standard size
	9″ color CRT/MDI	38W	For small and standard size
	9″ monochrome PDP/MDI	40W	For small and standard size
	10.4″ color LCD/MDI	20W	
	9.5″ color LCD/MDI	20W	
	14″ color CRT/MDI	70W	
Connection unit	Connection unit 1	35W	
	Connection unit 1+2	60W	
Operator’s panel	Operator’s panel connection unit	30W	
Control transformer		51W	

Index

≪Numbers≫

≪A≫

≪B≫

≪C≫

≪D≫

≪E≫

≪F≫

≪G≫

≪H≫

Revision Record

FANUC Series 15/150 For Machining Center DESCRIPTIONS (B–62082E)

Edition	Date	Contents
04	Apr., ’97	15MEK and 15MEL were added.
03	Sep., ’96	The explanations of the following functions were added: Smooth interpolation function, Designation direction tool length compensation, Three–dimensional coordinate conversion, Interpolation–type straightness compensation, Straightness compensation at 128 points, Bi–directional pitch error compensation function, Torque limit skip, High–precision contour control using 64–bit RISC processor, Skip function for EGB axis, Electronic gearbox automatic phase synchronization, Transverse inhibit limit function, Stored stroke check in manual operation function, M–code group function, Workpiece zero point manual setting function, Screen saver function, Unexpected disturbance torque detection function. Appendixes D to G were added. 150–MB was added.
02	Dec., ’93	The parameters of the following functions were added. 3–dimensional circular compensation, Helical involute interpolation, Spiral interpolation and Conical interpolation, Second feedrate override B, Function for overriding the rapid traverse feedrate in 1% units, Bell–shaped acceleration/deceleration after rapid traverse interpolation, Cutting point speed control function, Acceleration/deceleration function for the constant speed specified by the PMC control function, Cutting/rapid traverse position check function, Feed per rotation without a position coder, Automatic reference position setting function, Dog–less reference position setting function, Plane conversion function, Function for switching between diameter and radius programming, Circle cutting function, Rotary table dynamic fixture offset, Advance control function, Three–dimensional cutter function, The second cyclical pitch error compensation method, Skipping the commands for several axes, Read/punch function for custom macro common variables, Advance control function, High–precision contour control function, Follow–up for each axis, Simple synchronization control Positional deviation check function, Upgraded 5–axis control function, Roll–over function for a rotation axis, two–axis electronic gear box, Program restart function and output of M, S, T, and B, codes Active block cancel, Manual interpolation function for three–dimensional coordinate conversion, Retrace program editing function, Function for displaying multiple subscreens, Help function, Parameter setting (RS–232–C) screen, Screen for specifying high–speed and high–precision machining, Operation history, Wave form diagnosis function, CRT screen saving function, High–speed part program registration function, Function selection by hard keys, Multi edit function,FANUC Handy File, Position switching function Correction of errors.
01	Feb., ’93	————