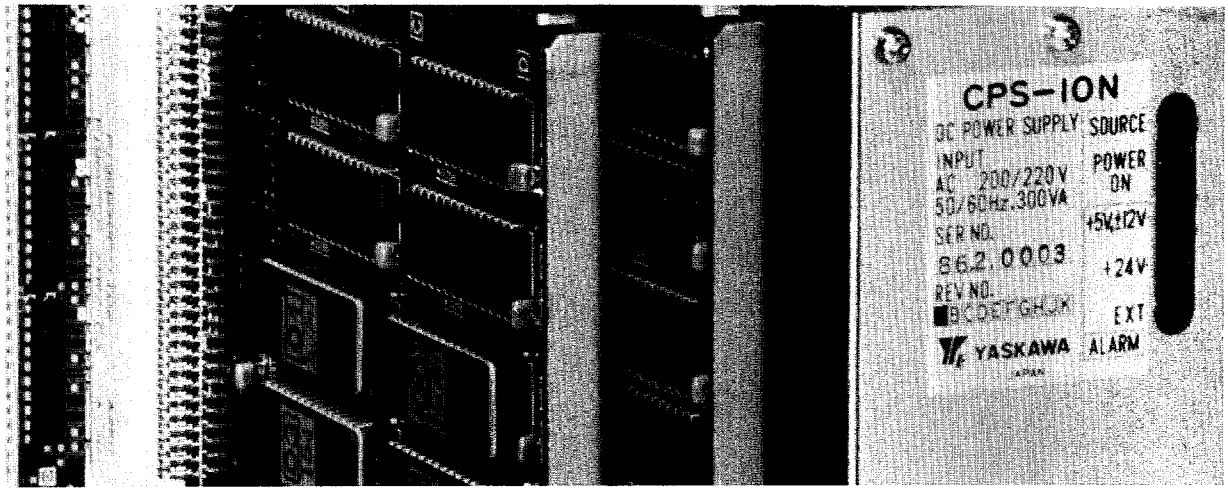
YASNAC LX3

CNC SYSTEM FOR TURNING APPLICATIONS

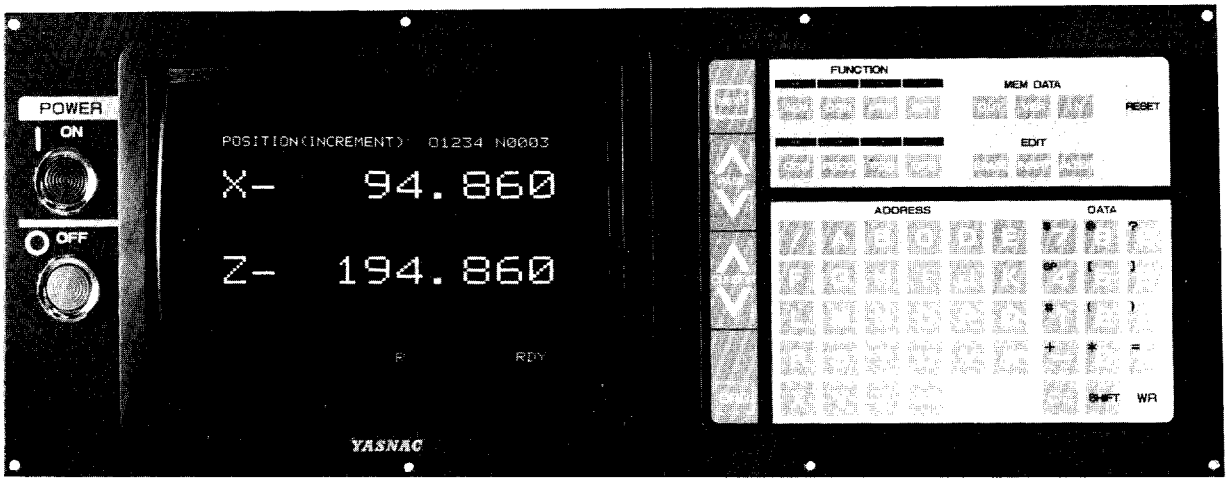


Before initial operation read these instructions thoroughly, and retain for future reference.

YASKAWA

This manual is primarily intended with 9" CRT character display (basic) to give operators instructions for YASNAC LX3 programming, and operation.

This manual applies to the basic and optional features of YASNAC LX3. The optional features are marked with a dagger. For the specifications of your YASNAC LX3, refer to the machine tool builder's manual.



586-175

YASNAC LX3 OPERATOR'S STATION

PREFACE

When reading this manual keep in mind that the information contained herein does not cover every possible contingency which might be met during the operation. Any operation not described in this manual should not be attempted with the control.

The functions and performance as NC machine are determined by a combination of machine and the NC control. For operation of your NC machine, the machine tool builder's manual shall take priority over this manual.

The illustration of machine control station should be used for your reference in understanding the function. For detailed array of operator's devices and names, refer to machine tool builder's manual.

Unless otherwise specified, the following rules apply to the description of programming examples shown in this manual.

- Feed Function Selection: G99 (mm/rev)
- Reference Zero Point
  (Return to reference zero by manual and automatic return):
- Absolute Zero Point:
- Work Coordinate Zero Point
- Dimensions: in MM

TABLE OF CONTENTS

1. INTRODUCTION *1*

2. PROGRAMMING *1*

2.1 Tape Format *1*

2.2 Program Number and Sequence Number *6*

2.3 Coordinate Words *7*

2.4 Rapid Traverse Rate *10*

2.5 Spindle-Speed Function (S-Function) *13*

2.6 Tool Function (T-Function) *14*

2.7 Miscellaneous Functions (M-Function) *19*

2.8 Preparatory Functions (G-Function) *23*

3. NC TAPE PUNCHING *148*

3.1 Tape Code *148*

3.2 Programming *148*

3.3 NC Tape *150*

3.4 NC Tape Handling *150*

4. STANDARD NC OPERATOR'S STATION WITH CRT CHARACTER DISPLAY *151*

4.1 Pushbuttons, Keys, and Lamps *151*

4.2 Power ON/OFF Operation *155*

4.3 Display and Writing Operation *156*

4.4 Loading Part Programs and NC Data into Memory (in) *174*

4.5 Tape Verifying *178*

4.6 Edit *180*

4.7 Part Program and NC Data Output Operations *183*

4.8 Summary of Storing and Editing Operations *186*

5. MACHINE CONTROL STATION *187*

5.1 Switching Units on the Control Station *187*

5.2 Operation Procedure *198*

6. OPERATION PROCEDURE *213*

6.1 Inspection before Turning on Power *213*

6.2 Turning on Power *213*

6.3 Manual Operation *213*

6.4 Preparation for Stored Leadscrew Error Compensation and Stored Stroke Limit *214*

6.5 Preparations for Automatic Operation *214*

6.6 Operation in Tape and Memory Mode *215*

6.7 Manual Operation Interrupting Automatic Operation *216*

6.8 Automatic Operation in MDI Mode *216*

6.9 MDI Operation Interrupting Automatic Operation *216*

6.10 Preparation for Turning off Power *217*

6.11 Turning off Power *217*

APPENDIX 1 LIST OF SETTING NUMBERS *A−1*

APPENDIX 2 LIST OF PARAMETER NUMBERS *A−7*

APPENDIX 3 STORED LEADSCREW ERROR COMPENSATION *A−25*

APPENDIX 4 LIST OF STANDARD INPUT/OUTPUT SIGNALS *A−26*

APPENDIX 5 LIST OF ALARM CODES *A−35*

APPENDIX 6 LIST OF DATA *A−54*

INDEX

	Subject	Chapter	Par.	Page
A	ABSOLUTE AND INCREMENTAL INPUTS	2	2.3.5	8
	ABSOLUTE/INCREMENTAL PROGRAMMING (G90, G91)	2	2.8.31	146
	Acceleration and Deceleration of Rapid Traverse and Manual Feed	2	2.4.3.1	12
	ADDING PART PROGRAMS	4	4.6.4	181
	ADDRESS KEYS	4	4.1.4	153
	Address Search	4	4.3.3.4	161
	ALARM CODE (ALM) DISPLAY	4	4.3.9	172
	Alarm Code Display	4	4.3.9.1	172
	Alarm Number of Macro Programs	2	2.8.23.10	88
	Argument Designation	2	2.8.23.2	68
	AUTO MODE HANDLE OFFSET	5	5.2.7	212
	AUTO MODE HANDLE OFFSET SWITCH†	5	5.1.28	195
	AUTOMATIC ACCELERATION AND DECELERATION	2	2.4.3	12
	AUTOMATIC COORDINATE SYSTEM SETTING†	5	5.2.2	199
	Automatic Nose R Function	2	2.8.19.1	56
	AUTOMATIC OPERATION IN MDI MODE	6	6.8	216
	AUTOMATIC RETURN TO REFERENCE POINT (G28)	2	2.8.11	35
	Automatic Threading Cycle (G76)	2	2.8.25.8	109
	Automatic Writing into the Tool Coordinate Memory	5	5.2.3.3	203
	Automatic Writing into the Work Coordinate System Shift Memory	5	5.2.3.4	204
B	BUFFER REGISTER	2	2.1.5	6
	BUFFERING FUNCTION (M93, M92)†	2	2.7.3	19
C	CANNED CYCLES (G90, G92, G94)	2	2.8.26	116
	CHECKING REGISTERED PART PROGRAM NUMBER	4	4.6.1	180
	Circular Arc Multiple Cornering (G112)	2	2.8.30.2	143
	CIRCULAR INTERPOLATION (G02, G03)	2	2.8.4	28
	CIRCULAR PATH MODE ON/OFF ON TOOL RADIUS COMPENSATION (G97, M96)†	2	2.7.4	20
	COMMAND DATA DISPLAY	4	4.3.2	157
	Command Data Display	4	4.3.2.1	158
	Command Pulse Accumulation Register Display (COMMAND PULSE)	4	4.3.4.9	165
	Conditions for the Automatic Nose R Compensation Function to be Enabled	2	2.8.19.2	56
	Conditions of the Specifications to Perform FS Editing	5	5.1.31.7	198
	Considerations and Remarks for Macro Programs	2	2.8.23.9	84
	CONSTANT DISPLAY	4	4.3.1	156
	CONSTANT SURFACE SPEED CONTROL (G96, G97)†	2	2.8.27	123
	Control Commands	2	2.8.23.6	81
	Coordinate System Setting (G50 X_. Z_.)	5	5.2.3.5	205
	COORDINATE WORDS	2	2.3	7
	COORDINATE WORDS	2	2.3.1	7
	CORNERING (G11, G12)†	2	2.8.7	31
	CRT CHARACTER DISPLAY	4	4.1.2	152
	CURRENT POSITION DISPLAY	4	4.3.4	162
	CURSOR KEYS	4	4.1.8	154
	CUTTING DEPTH OVERRIDE SWITCH† FOR G71 AND G72	5	5.1.29	195
	CYCLE START PUSHBUTTON AND LAMP	5	5.1.2	188
D	DATA KEYS	4	4.1.5	153
	DECIMAL POINT PROGRAMMING	2	2.1.3	6
	DELETING PART PROGRAM BLOCKS	4	4.6.5	182
	Display and Deleting of Registered Program Number (PROGRAM NO. TABLE□□)†	4	4.3.9.3	173
	Display and Write of Local and Common Variables	2	2.8.23.8	83
	DISPLAY AND WRITING OPERATION	4	4.3	156
	Display of Subprogram Run Status (SUB PROG. NESTING)	4	4.3.2.2	158
	Display of Tool Life Control Use Status (TOOL LIFE CONTROL)	4	4.3.2.3	159
	Display in EDIT Mode	4	4.3.3.3	161
	DISPLAYING AND CHECKING STORED PART PROGRAMS	4	4.6.2	180

INDEX (Cont'd)

	Subject	Chapter	Par.	Page
D	DISPLAYING AND WRITING PARAMETERS	4	4.3.7	170
	Displaying and Writing Parameters	4	4.3.7.3	171
	DISPLAYING AND WRITING SETTING DATA (SETTING)	4	4.3.6	168
	DISPLAYING AND WRITING TOOL OFFSET DATA†	4	4.3.5	166
	DISPLAYING STATUS INPUT/OUTPUT SIGNALS	4	4.3.8	172
	Display in Memory Run Mode (PROGRAM [MEM])	4	4.3.3.2	160
	DISPLAY LOCK/MACHINE LOCK SWITCH	5	5.1.19	193
	DRY RUN SWITCH	5	5.1.18	193
	DWELL (G04)	2	2.8.5	29
E	EDIT	4	4.6	180
	EDIT KEYS	4	4.1.10	154
	EDIT LOCK SWITCH	5	5.1.22	194
	EIA/ISO AUTO RECOGNITION	3	3.1.2	148
	EMERGENCY STOP PUSHBUTTON	5	5.1.4	189
	ERROR DETECT OFF POSITIONING (G06)	2	2.8.2.2	27
	Example of FS Editing	5	5.1.31.6	197
	Example of High-speed M Function Processing	2	2.7.8.4	23
	Exercises of Macro Programs	2	2.8.23.11	89
F	Facing Cycle B (G94)	2	2.8.26.3	121
	FEED FUNCTION (F- AND E-FUNCTION)	2	2.4.2	10
	FEED HOLD PUSHBUTTON AND LAMP	5	5.1.3	188
	FEED/MINUTE AND FEED/REVOLUTION SWITCHOVER	2	2.8.28	124
	Feed Per Minute (G98 Mode)	2	2.4.2.2	12
	Feed Per Revolution (G99 Mode)	2	2.4.2.1	11
	FEEDRATE OVERRIDE CANCEL SWITCH	5	5.1.12	192
	Finishing Cycle (G70)	2	2.8.25.5	106
	FS AUTOMATIC EDITING FUNCTION	5	5.1.31	196
	FUNCTION KEYS	4	4.1.3	152
	Functions	2	2.6.5.2	17
	Functions	5	5.1.31.3	196
G	General	2	2.8.25.1	94
	General	5	5.1.8.2	190
	General	5	5.1.31.1	196
	General	5	5.2.3.10	206
	GENERAL PROGRAM FORM	3	3.2.2	148
	G50 POINT RETURN SWITCH†	5	5.1.24	195
	G50 POINT RETURN†	5	5.2.4	208
	GROOVE WIDTH COMPENSATION (G150, G151)*	2	2.8.32	147
	Grooving in X-Axis (G75)	2	2.8.25.7	108
H	HANDLE AXIS SELECT SWITCH†	5	5.1.6	189
	HANDLE DIAL† (MANUAL PULSE GENERATOR)	5	5.1.5	189
	HANDLE DIALS FOR SIMULTANEOUS CONTROL OF UP TO TWO AXES†	5	5.1.8	189
	Handle Interpolation Function	5	5.1.8.1	190
	HIGH-SPEED M FUNCTION	2	2.7.8	22
I	Improved Multiple Repetitive Cycle Function	2	2.8.25.10	115
	INCH/METRIC DESIGNATION BY G CODE (G20, G21)†	2	2.8.8	33
	Input/Output	5	5.1.31.2	196
	I/O Channel	2	2.7.8.1	22
	INPUTTING SETTING DATA AND PARAMETER DATA	4	4.4.5	178
	INPUTTING TOOL OF OFFSET DATA INTO MEMORY	4	4.4.4	177
	Input Unit and 10 Times Input Unit	2	2.3.3.1	7
	INSPECTION BEFORE TURNING ON POWER	6	6.1	213
	INTERLOCK INPUT (INTERLOCK)	5	5.1.23	195
	Internal Toggle Switches†	4	4.3.6.2	169
	INTRODUCTION	1		1
J	JOG FEEDRATE SWITCH AND FEEDRATE OVERRIDE SWITCH	5	5.1.10	191
	JOG PUSHBUTTONS AND RAPID PUSHBUTTON	5	5.1.9	191
K	KEEPING OF NC TAPE	3	3.4.2	150

INDEX (Cont'd)

	Subject	Chapter	Par.	Page
L	LABEL SKIP FUNCTION	2	2.1.4	6
	LEAST INPUT INCREMENT AND LEAST OUTPUT INCREMENT	2	2.3.3	7
	Least Output Increment	2	2.3.3.2	8
	LINEAR INTERPOLATION (G01)†	2	2.8.3	27
	LIST OF ADDRESS CHARACTERS AND FUNCTION CHARACTERS	2	2.1.2	4
	LIST OF ALARM CODE	APPENDIX 5		A-35
	LIST OF DATA	APPENDIX 6		A-54
	LIST OF G CODES	2	2.8.1	23
	LIST OF PARAMETER NUMBERS	APPENDIX 2		A-7
	LIST OF SETTING NUMBERS	APPENDIX 1		A-1
	LIST OF STANDARD INPUT/OUTPUT SIGNALS	APPENDIX 4		A-26
	LIST OF TAPE CODE	3	3.1.1	148
	LOADING PART PROGRAM TAPE INTO MEMORY	4	4.4.1	175
	LOADING PART PROGRAMS AND NC DATA INTO MEMORY (IN)	4	4.4	174
	LOADING PART PROGRAMS BY MDI	4	4.4.3	177
M	MACHINE CONTROL STATION	5		187
	Macro Program Call Commands	2	2.8.23.1	66
	MACRO PROGRAMS (G65 AND G67)	2	2.8.23	66
	Maintenance History Display (MAINTENANCE)	4	4.3.9.5	174
	MAKING ADDITION TO A PART PROGRAM	4	4.4.2	176
	MANUAL ABSOLUTE SWITCH	5	5.1.21	194
	MANUAL INTERRUPTION POINT RETURN SWITCH	5	5.1.25	195
	MANUAL INTERRUPTION POINT RETURN†	5	5.2.5	209
	MANUAL OPERATION†	6	6.3	213
	MANUAL OPERATION INTERRUPTING AUTOMATIC OPERATION	6	6.7	216
	MANUAL PULSE MULTIPLY SELECT SWITCH†	5	5.1.7	189
	MANUAL REFERENCE POINT RETURN SWITCH†	5	5.1.14	192
	MANUAL RETURN TO REFERENCE POINT	5	5.2.1	198
	MAXIMUM PROGRAMMABLE DIMENSIONS	2	2.3.4	8
	MAXIMUM SPINDLE-SPEED SETTING (G50)†	2	2.8.21	62
	M CODES FOR INTERNAL PROCESSING (G99 TO M109)	2	2.7.2	19
	M CODES FOR STOP (M00, M01, M02, M30)	2	2.7.1	19
	MDI OPERATION INTERRUPTING AUTOMATIC OPERATION	6	6.9	216
	MEASURED WORKPIECE VALUE DIRECT INPUT†	5	5.2.3	199
	MEM DATA (MEMORY DATA) KEYS	4	4.1.11	154
	Message Display (ALARM)†	4	4.3.9.2	172
	M-FUNCTION LOCK SWITCH (AUXILIARY FUNCTION LOCK)	5	5.1.20	194
	MISCELLANEOUS FUNCTIONS (M-FUNCTION)	2	2.7	19
	MODE SELECT SWITCH	5	5.1.1	187
	MODIFYING PART PROGRAM BLOCKS	4	4.6.3	180
	M3-DIGIT BCD OUTPUT†	2	2.7.7	21
	Multi-block Writing and Operation in MDI Mode	4	4.3.3.1	159
	MULTIPLE CORNERING (G111, G112)†	2	2.8.30	134
	MULTIPLE REPETITIVE CYCLES (G70 TO G76)†	2	2.8.25	94
	MULTI-START THREAD CUTTING (G32)†	2	2.8.16	41
N	NC TAPE	3	3.3	150
	NC TAPE CHECK	3	3.3.3	150
	NC TAPE HANDLING	3	3.4	150
	NC TAPE PUNCH	3	3.3.2	150
	NC TAPE PUNCHING	3		148
	NEW COORDINATE SYSTEM SETTING FUNCTION	2	2.6.5	17
	NEXT KEY	4	4.1.6	153
	New Tool Setter Function	5	5.2.3.9	206
	No. of Servo Lag Pulses Display (ERROR PULSE)	4	4.3.4.8	165
	Notes	2	2.7.8.3	23
	NOTES WHEN USING THE CONVENTIONAL G50T★★★★FUNCTION	2	2.6.6	19
O	Offset Calculation of Automatic Nose R Compensation Approach and Relief	2	2.8.19.3	57
	Offset Screen Display	5	5.2.3.8	205
	Operation	5	5.1.8.3	190
	Operation Commands	2	2.8.23.5	79
	Operation Expression for Coordinate System Setting	2	2.6.5.3	18

INDEX (Cont'd)

	Subject	Chapter	Par.	Page
O	OPERATION IN TAPE AND MEMORY MODE	6	6.6	215
	Operation Procedure	5	5.1.31.4	196
	OPERATION PROCEDURE	5	5.2	198
	OPERATION PROCEDURE	6		213
	Operation Time Display	4	4.3.9.4	174
	OPTIONAL BLOCK SKIP (/1-/9)†	2	2.2.3	7
	OPTIONAL BLOCK SKIP SWITCH	5	5.1.17	193
	ORG (ORIGIN) KEYS	4	4.1.9	154
	OTHER M CODES	2	2.7.6	21
	OUTPUTTING PART PROGRAM TO PAPER TAPE	4	4.7.1	183
	OUTPUTTING SETTING/PARAMETER DATA TO PAPER TAPE	4	4.7.3	184
	OUTPUTTING TOOL OFFSETS TO PAPER TAPE	4	4.7.2	184
	Overview	2	2.6.5.1	17
	Overview of Macro Program Body	2	2.8.23.3	69
P	PAGE KEYS	4	4.1.7	153
	PAPER TAPE	3	3.3.1	150
	Parameter of Bit Display Format	4	4.3.7.1	171
	Parameter of Decimal Display Format	4	4.3.7.2	171
	Parameters	2	2.7.8.2	22
	Parameters	5	5.2.3.2	203
	PART PROGRAM AND NC DATA OUTPUT OPERATIONS	4	4.7	183
	Pattern Repeating (G73)	2	2.8.25.4	104
	Peck Drilling in Z-axis (G74)	2	2.8.25.6	107
	Position	4	4.3.4.4	163
	Position [ABSOLUTE]	4	4.3.4.2	163
	Position [EXTERNAL]	4	4.3.4.1	162
	Position (EXTERNAL) "0" Setting	5	5.2.3.6	205
	Position [INCREMENT]	4	4.3.4.3	163
	POSITION (G00, G06)	2	2.8.2	24
	Positioning (G00)	2	2.8.2.1	24
	POSITION STORED PUSHBUTTON†	5	5.1.30	196
	POWER ON/OFF OPERATION	4	4.2	155
	POWER ON/OFF PUSHBUTTONS	4	4.1.1	152
	Precautions in Programming G70 through G76	2	2.8.25.9	112
	PREPARATION FOR STORED LEADSCREW ERROR COMPENSATION AND STORED STROKE LIMIT†	6	6.4	214
	PREPARATION FOR TURNING OFF POWER	6	6.10	217
	PREPARATIONS FOR AUTOMATIC OPERATION	6	6.5	214
	PREPARATORY FUNCTIONS (G-FUNCTION)	2	2.8	23
	PROCESS SHEET	3	3.2.1	148
	PROGRAMMING	2		1
	PROGRAMMING	3	3.2	148
	PROGRAMMING OF ABSOLUTE ZERO POINT (G50)	2	2.8.20	61
	PROGRAM MIRROR IMAGE (G68, G69)†	2	2.8.24	92
	PROGRAM NUMBER	2	2.2.1	6
	PROGRAM NUMBER AND SEQUENCE NUMBER	2	2.2	6
	PROGRAM RESTART SWITCH†	5	5.1.26	195
	PROGRAM RESTART†	5	5.2.6	209
	Program Return	4	4.3.4.5	164
	PUSHBUTTONS, KEYS, AND LAMPS	4	4.1	151
R	RADIUS PROGRAMMING FOR CIRCULAR INTERPOLATION (G22, G23)†	2	2.8.9	33
	RAPID TRAVERSE RATE	2	2.4	10
	RAPID TRAVERSE RATE	2	2.4.1	10
	RAPID TRAVERSE RATE OVERRIDE SWITCH	5	5.1.11	192
	REFERENCE POINT CHECK (G27)	2	2.8.10	34
	REFERENCE POINT LAMPS	5	5.1.15	193
	Registration of Macro Programs	2	2.8.23.7	83
	RESET KEY	4	4.1.12	155
	RETURN FROM REFERENCE POINT (G29)	2	2.8.12	36

INDEX (Cont'd)

	Subject	Chapter	Par.	Page
S	2ND REFERENCE POINT RETURN (G30)†	2	2.8.13	37
	SEQUENCE NUMBER	2	2.2.2	7
	Setting Data of Bit Display Format	4	4.3.6.1	168
	Setting Data of Decimal Display Format	4	4.3.6.3	169
	S4-DIGIT PROGRAMMING A†	2	2.5.2	13
	S4-DIGIT PROGRAMMING B†	2	2.5.3	14
	SIMULTANEOUS CONTROLLABLE AXES	2	2.3.2	7
	SINGLE BLOCK SWITCH	5	5.1.16	193
	SKIP FUNCTION (G31)†	2	2.8.14	37
	SUBROUTINE PROGRAM (M98, M99)	2	2.7.5	20
	Spindle Counter	4	4.3.4.7	165
	SPINDLE SPEED OVERRIDE SWITCH†	5	5.1.13	192
	SPINDLE-SPEED FUNCTION (S-FUNCTION)	2	2.5	13
	SPLICING NC TAPES	3	3.4.1	150
	S2-DIGIT PROGRAMMING (SPECIAL SPECIFICATIONS)	2	2.5.1	13
	STANDARD NC OPERATOR'S STATION WITH CRT CHARACTER DISPLAY	4		151
	Stock Removal in Facing (G72)	2	2.8.25.3	101
	Stock Removal in Turning (G71)	2	2.8.25.2	95
	STORED LEADSCREW ERROR COMPENSATION	APPENDIX 3		A-25
	STORED STROKE LIMIT (G36-G39)	2	2.8.18	44
	SWITCHING UNITS ON THE CONTROL STATION	5	5.1	187
T	TAPE FORMAT	2	2.1	1
	TAPE FORMAT	2	2.1.1	1
	The G40 G01 X_Z_I_K_ ; Command Cancel Function is Also Available in the Automatic Nose R Function	2	2.8.19.4	61
	TOOL FUNCTION (T-FUNCTION)	2	2.6	14
	T 4-DIGIT PROGRAMMING	2	2.6.1	14
	TOOL OFFSET MEMORY†	2	2.6.2	14
	TOOL POSITION OFFSETS	2	2.6.3	15
W	WORK COORDINATE SYSTEM SHIFT†	2	2.6.4	17
X	X-AXIS DIAMETER/RADIUS SWITCHING	2	2.3.6	10

1. INTRODUCTION

YASNAC LX3, "Ultraspeed dual processor CNC" is a combination of two high-performance 16-bit microprocessors running in parallel. Incorporating our modern system technique, it is designed to provide the highest lathe performance.

- The dual processor CNC system drastically reduces the data processing time to meet high-speed cutting. Block-to-block stop time decreased by the use of high-speed buffer function and buffering function.
- Enhanced cutting capability includes a maximum of 24 meters/min feed command, precise feed E command, 500-milimeter lead thread cutting, continuous thread cutting, multiple thread cutting, and variable pitch thread cutting.
- To meet FMS trends, program interrupt function, tool life control, user macro, tool set error correction, stored stroke limit per tool, and other functions can be installed.
- Part program memory can be extended to a maximum of 320 meters. Its data input/output interface is available with FACIT, RS232C and, in addition, RS422 serial interface capable of high-speed long distance transmission.
- Programming is further facilitated by improved tool radius compensation function, G50-work coordinate system setting, angle-specified linear interpolation, and combined beveling/rounding function.
- The servo function uses a drastically miniaturized and low-noise, newly transistorized PWM control unit and a high-performance DC servo motor.

  The position feedback is available with the standard pulse generator (PG) system and, the inductosyn-applied complete closed loop system.

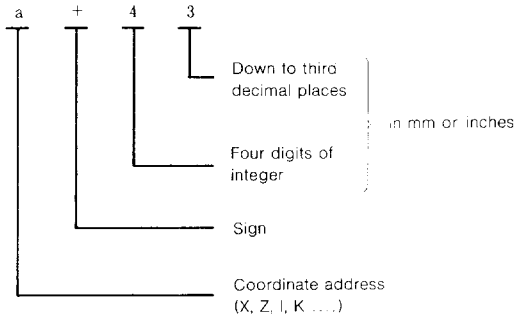
2. PROGRAMMING

2.1 TAPE FORMAT

2.1.1 TAPE FORMAT

A variable block format conforming to JIS# B 6313 is used for YASNAC LX3.
Table 2.1 shows the tape format. Numerals following the address characters in Table 2.1 indicate the programmable number of digits.

EXAMPLE



Japanese Industrial Standard

Note: The decimal point may be omitted in actual programming. For making a program including decimal points, refer to 2.1.3 Decimal Point Programming.

The leading zeros can be suppressed for all address codes. Plus signs need not be programmed, but all minus signs must be programmed. In the manual, EOB code in a program example is represented by a semicolon (;). In actual programming, CR (EIA code) or LF/NL (ISO code) should be used instead of the semicolon (;).

Table 2.1 Tape Format

No.	Address	Metric Output		Inch Output		B: Basic O: Option
		Metric Input	Inch Input	Metric Input	Inch Input	
1	Program No.	O4		O4		B
2	Sequence No.	N4		N4		B
3	G-Function	G3		G3		B
4	Coordinate Word a: X, Z, I, K, U, W, R	(a – 53)	(a – 44)	a – 53	a – 44	B
5	Feed/min	F50	F32	F50	F42	B
6	Feed/rev and Thread Lead	F32	F24	F42	F24	B
		E34	E26	E44	E26	B
7	S-Function	S2		S2		B
		S4		S4		O
8	T-Function	T (2 + 1)		T (2 + 1)		B
		T (2 + 2)		T (2 + 2)		O
9	M-Function	M3		M3		B
10	Dwell	U (P) 53		U (P) 53		B
11	Program No. Designation	P4		P4		B
12	Sequence No. Designation	Q (P) 4		Q (P) 4		B, O
13	No. of Repetitions	L8		L8		B
14	Angle Designation for Straight Line	A (B) 33		A (B) 33		O
15	Angle Designation for Multiple Thread	B3		B3		O

Notes:

1. Inch/Metric output is set by setting parameter #6007 D_3.
2. Inch/Metric input is set by setting (#6001 D_0).
3. F codes for feed/min or feed/rev can be switched by G98, G99.

Table 2.2 List of Program Commands

Address		Metric Output		Inch Output	
		Metric Input	Inch Input	Metric Input	Inch Input
Program No. O		1-9999		1-9999	
Sequence No. N		1-9999		1-9999	
G function G		0-199		0-199	
Coordinate Address[1] X, Z, I, K, U, W, R		±99999.999 mm	±3937.0078 in.	±99999.999 mm.	9999.9999 in.
Feed/min	F	1-24000 mm/min	0.01-944.88 in/min	1-60960 mm/min	0.01-2400.00 in/min
Feed/rev and Thread Lead	F	0.01-500.00 mm/rev	0.0001-19.6850 in/rev	0.01-1270.00 mm/rev	0.0001-50.0000 in/rev
	E	0.0001-500.0000 mm/rev	0.000004-19.685000 in/rev	0.0003-1270.0000 mm/rev	0.000010-50.000000 in/rev
S-function	S2	0-99		0-99	
	S4	0-9999		0-9999	
T-function	T3	0-999		0-999	
	T4	0-9999		0-9999	
M-function		0-999		0-999	
Dwell U, P		0.001-99999.999 sec		0.001-99999.999 sec	
Program No. Designation		1-9999		1-9999	
Sequence No. Designation		1-9999		1-9999	
No. of Repetitions		1-99999999		1-99999999	
Angle Designation for Straight Line[2]		0-±360.000°		0-±360.000°	
Angle Designation for Multiple Thread		0-360°		0-360°	

Note: For angle designation of included angle for G76, see 2.8.26.8 Automatic Threading Cycle (G76).

2.1.2 LIST OF ADDRESS CHARACTERS AND FUNCTION CHARACTERS

Table 2.3 Address Characters

Address	Meaning	B: Basic O: Optional
A	Angle designation for G01 and G111, included angle for G76	O
B	Spindle shift angle 01 multiple thread, angle designation for multiple cornering	O
C	User macro character	O
D	Depth of cut and number of cutting cycles for G71 to G76	O
E	Specifications for precise feed and precise lead for cutting	B
F	Specifications for normal feed and normal lead for cutting	B
G	Preparatory function (G-function)	B
H	User macro character	O
I	X-component of arc center, canned cycle parameter, beveling value (radius value)	B, O
J	User macro character	O
K	Z-component of arc center, canned cycle parameter, beveling value	B, O
	Incremental value of variable lead thread	O
L	Number of subprogram repetition, G13 to G16 angle and coordinate	B, O
M	Miscellaneous function (M-function)	B
N	Sequence number	B
O	Program number	B
P	Dwell, canned cycle starting sequence number, program number, user macro number	B, O
Q	Subprogram starting sequence number, canned cycle ending sequence number	B, O
R	Radius of arc, rounding value, tool radius value	B, O
S	Spindle function (S-function), maximum spindle revolution	B
T	Tool function (T-function), tool coordinate memory number	B, O
U	X-axis incremental command value, dwell, canned cycle parameter	B, O
V	User macro character	O
W	Z-axis incremental command value, canned cycle parameter	B, O
X	X-axis coordinate value	B
Y	User macro character	O
Z	Z-axis coordinate value	B

Table 2. 4 Function Characters

EIA Code	ISO Code	Function	Remarks
Blank	NuL	Error in significant data area in EIA Disregarded in ISO	
BS	BS	Disregarded	
Tab	HT	Disregarded	
CR	LF/NL	End of Block (EOB)	
	CR	Disregarded	
SP	SP	Space	
ER	%	Rewind stop	
UC		Upper shift	
LC		Lower shift	
2-4-5 bits	(	Control out (comment start)	EIA: Special code
2-4-7 bits	)	Control in (comment end)	
+	+	Disregarded, User macro operator	
–	–	Minus sign, User macro operator	
0 to 9	0 to 9	Numerals	
a to z	A to Z	Address characters	
/	/	Optional block skip	
Del	DEL	Disregarded (Including All Mark)	
.	.	Decimal point	
Parameter starting	#	Sharp (Variable designation)	EIA: Special code
*	*	Asterisk (Multiplication operator)	
=	=	Equal mark	
[	[	Left bracket	
]	]	Right bracket	
$	$	User macro operator	
@	@	User macro operator	
?	?	User macro operator	

Notes :

1. Characters other than the above cause error in significant data area.
2. Information between Control Out and Control In is ignored as insignificant data.
3. Tape code (EIA or ISO) is automatically recognized.

2.1.3 DECIMAL POINT PROGRAMMING

Numerals containing a decimal point may be used as the dimensional data of addresses related to coordinates (distance), angle, time and speed. They can be input from punched tape or MDI.

Decimal points can be used in the following address words.

Coodinate words; X, Z, U, W, I, K, R

Angle words: A, B

Feedrate word: F, E

Time words: U, P

EXAMPLE

	[mm]	[inch]
X15. ———	X15.000 mm	or X15.0000 in.
Z20.5 ———	Z20.500 mm	or Z20.5000 in.
(G99)F.2† —	F0.20 mm/rev (for F32)	or F0.2000 in./rev (for F24)
(G98)F25.6	F25 mm/min (for F50)	or F25.60 mm/min (for F32)
G04P1. ———	Dwell 1.000 sec	

When data without a decimal point is input, the control regards "1" as 0.001 mm (or 0.0001 inch).

2.1.4 LABEL SKIP FUNCTION

In the following cases the label skip function becomes effective, and LSK is displayed on the CRT.

- When the power supply is turned on.
- When the RESET operation is executed.

While the label skip function is effective, all data on the punched tape up to the first EOB code are neglected. When LSK is displayed on the CRT in the MEM (memory) or EDIT (editing) mode, it indicates the presence of a pointer at the leading end of the part program.

2.1.5 BUFFER REGISTER

During normal operation, one block of data is read in advance and compensation is computed for the follow-on operation.

In the tool radius compensation† mode, two blocks of data or up to 4 blocks of data are read in advance and compensation computing required for the next operation is executed. One block can contain up to 128 characters including EOB.

The blocks including the following M codes are not read in advance.

- M00, M01, M02, M30
- M codes (6 maximum) set by parameter commanding to stop advance-reading.

Notes :

1. This function is effective for G22 and G23 where the control is provided with Radius Programming for Circular Interpolation option.
2. Block-to-block stop time due to the time required to compute tool radius compensation is not eliminated or remains. To reduce this stopping time, use 2.7.3 Buffering Function (M93, M92) (optional). When operation of consecutive blocks up to 5 in M93 mode, inter-block stoppage time is reduced to zero.

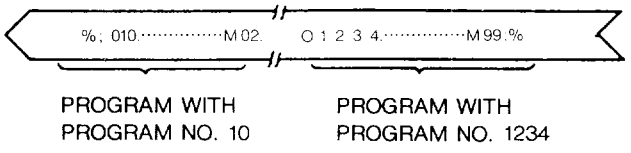
2.2 PROGRAM NUMBER AND SEQUENCE NUMBER

2.2.1 PROGRAM NUMBER

Program numbers may be prefixed to programs for the purpose of program identification.

Up to 4 digits may be written after an address character "O" as program numbers. Up to 99 program numbers can be registered in the control, and up to 199 or 999 can be registered employing an option.

One program begins with a program number, and ends with M02, M30 or M99. M02 and M30 are placed at the end of main programs, and M99 is placed at the end of subprograms.



ER (or % at ISO code) is punched on both end parts of the tape.

Notes :

1. The blocks for optional block skip such as /M02;, /M30;, /M99; are not regarded as end of programs.
2. It is possible with a parameter change (#6201D_0), to make the reading of M02, M30, and M99 ineffective as a program end, and to make the succeeding ER (EIA) or % (ISO) as a sign of program end.

2. 2. 2 SEQUENCE NUMBER

Integers consisting of up to 4 digits may be written following an address character N as sequence numbers.

Sequence numbers are reference numbers for blocks, and do not have any influence on the meaning and sequence of machining processes. Therefore, they may be sequential, non-sequential, and duplicated numbers, also not using any sequence number is possible. Generally, sequential numbers are convenient as sequence numbers.

When searching for sequence numbers, be sure to search or specify program numbers beforehand.

Notes :

1. Five or more digits must not be written as a sequence number.
2. When two or more blocks have the same sequence number, only one is retrieved and read, and no more searching is performed.
3. Blocks without sequence numbers can also be searched for with respect to the address data contained in the blocks.

2. 2. 3. OPTIONAL BLOCK SKIP (/1-/9)(†)

Those blocks in which "/n" (n = (1 - 9) is included are neglected between /n and the end of that block, when the external optional block skip switch for that number "n" is on.

EXAMPLE

```
/2 N1234 G01 X100 /3 Z200;
```

When the switch for /2 is on, the entire block is neglected, and when the switch for /3 is on, this block is read as if

```
N 1234 G01 X100;.
```

With "1," "1" may be omitted.

Notes :

1. The optional block skipping process is executed while the blocks are being read into the buffer resister. Once the blocks have been read, subsequent switching on is ineffective to skip the blocks.
2. While reading or punching out programs, this function is ineffective.
3. The block skip /2 - /9 is an option function, and /1 is a basic one.

2. 3 COORDINATE WORDS

Generally, commands for movements in axis directions and commands for setting coordinate systems are called coordinate words, and coordinate words consist of address characters for desired axes and numerals representing dimensions of directions.

2. 3. 1 COORDINATE WORDS

Address of Coordinate Words		Meaning
Main Axis	X, Z	Absolute coordinate position of target position
	U, W	Incremental distance (U: Direction in X-axis, W: Direction in Z-axis)
Radius Value for Circular Interpolation	I, K	Incremental distance between start point and center of circular arc. (I: X-axis component, K: Z-axis component)
	R†	Radius value of circular arc

Note: When G 90 and G 91 are used, addresses X and Z are not fixed as absolute value and follow according to G 90/G 91 designation. For details, refer to 2. 3. 5 Absolute and Incremental Inputs.

2. 3. 2 SIMULTANEOUS CONTROLLABLE AXES

The control provides two-axis control for X- and Z-axis. Number of simultaneously controllable axes, when commanded in the same block, is two axes, X and Z. For the axis without commands, movement will not occur.

2. 3. 3 LEAST INPUT INCREMENT AND LEAST OUTPUT INCREMENT

2.3.3.1 Input Unit and 10 Times Input Unit

The minimum input units that can be commanded by punched tape or MDI are shown below.

Least Input Increment

	1 x	10 x (10 times input unit)
Metric system	0.001 mm	0.01 mm
Inch system	0.0001 in.	0.001 in.

X-axis is specified for diameter.

2.3.3.1 Input Unit and 10 Times Input Unit (Cont'd)

Inch/MM input is selected by setting #6001D_0. Inch/MM input selection by G20/G21 is optional. Selection of multiplication factor x1/x10 is made by parameter #6006D_5.

Tool offset value must always be written in 0.001 mm (or 0.0001 inch), and offset is possible in these units.

In 0.01 mm increment system, the following operation must be made in the unit of 0.01 mm.

- Programming for operation in TAPE mode.
- Write operation in MDI mode.
- Programming for operation in MEMORY mode.
- Program editing operation in EDT mode.

Notes :

1. If NC tape programmed by 0.001 mm is fed into or stored in an equipment set by 0.01 mm increment, the machine will move ten times the intended dimensions.
2. If the increment system is switched when the contents of NC tape are stored in memory, the machine will move by ten times or one tenth of the commanded dimensions.
3. When the stored program is punched out on the tape†, the stored figures are punched out "as stored" regardless of switching of the increment system.
4. Multiplication factor 10X (10 times the input unit) is effective for distance command only. It does not function on the designation of time, angle, etc. When multiplication factor 10X is set as effective (#6006D_5 = 1), the same address word is multiplied by 10 or not depending on type of G command.

   EXAMPLE

   G04 U··· ;←—Not multiplied by 10 (Time)
   G00 U··· ;←—Multiplied by 10 (Distance)

2.3.3.2 Least Output Increment

Least output increment is the minimum unit of tool motion. Selection of metric system or inch system is made by parameter (#6007D_3).

Least Output Increment

	X-axis (Radius value)	Z-axis
Metric output	0.0005 mm	0.001 mm
Inch output	0.00005 in.	0.0001 in.

2.3.4 MAXIMUM PROGRAMMABLE DIMENSIONS

Maximum programmable values of move command are shown below.

Maximum Programmable Values

Metric Output	Metric input	±9999.999 mm.
	Inch input	±3937.0078 in.
Inch Output	Metric input	±99999.999 mm.
	Inch input	±99999.999 in.

In incremental programming, specified values must not exceed the maximum programmable values. In absolute programming, move amount of each axis must not exceed the maximum programmable value. THE MACHINE MAY NOT FUNCTION PROPERLY IF MOVE COMMAND OVER THE MAXIMUM PROGRAMMABLE VALUES IS GIVEN.

The above maximum programmable values also apply to distance command addresses I, K, R in addition to move command addresses X, U, W.

2.3.5 ABSOLUTE AND INCREMENTAL INPUTS

Both absolute input and incremental input can be used for the control.

- Absolute input is specified by the addresses X and Z.

  EXAMPLE: X··· Z··· ;

- Incremental input is specified by the addresses U and W.

  EXAMPLE: U··· W··· ;

- Absolute input and incremental input can be used in one block mixedly.

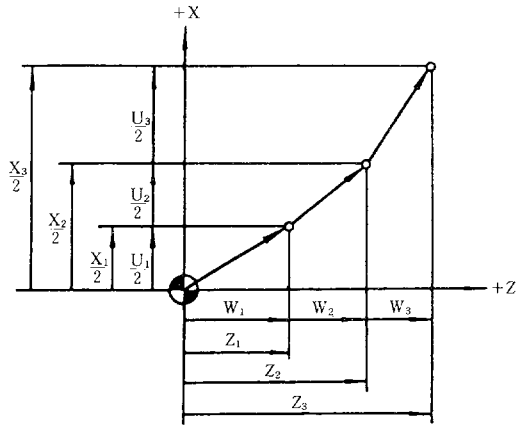
  EXAMPLE: X··· W··· ;
  U··· Z··· ;

Note : When addresses X and U or addresses Z and W are used in one block, the latter is effective.

The addresses I and K for designation of arc-center must be specified by the incremental dimension.

Table 2.5

Address	Increment System	Designation	Meaning
X	Absolute Input	Diameter	Position in X-axis direction *
Z		—	Position in Z-axis direction *
U	Incremental Input	Diameter	Move amount in X-axis direction
W		—	Move amount in Z-axis direction
I	Incremental Input	Radius	Distance in X-axis direction from starting point of arc to center
K		—	Distance in Z-axis direction from starting point of arc to center
R	Incremental Input	—	Direct programming of circular arc



X and Z: Absolute Input

U and W: Incremental Input

* Since X and U are designated by the values in diameter, the actual movement is the half of the values.

Fig. 2.1 Absolute Coordinate Values and Incremental Coordinate Values

Cases where G90 and G91 (absolute and incremental commands) are used.

· When special G code I (basic) or II (option) is selected, G90 and G91 codes can be used.

G code	Meaning
G90	Absolute command
G91	Incremental command

As shown below, G90 and G91 commands are effective only to addresses X and Z.

	Addresses	G90 Command	G91 Command
TAPE, MEM, MDI modes	X, Z	Absolute	Incremental
	U, W	Incremental	Incremental

EXAMPLE: G91 G00 X40. Z50.; ·····

Incremental move command

· Auxiliary data, I, K, R, etc., of circular interpolation are always incremental commands.

Note : G90 and G91 cannot be programmed together in the same block. If they are written in the same block, the one written later only is effective.

EXAMPLE: G01 G90 X80. G91 Z60.;

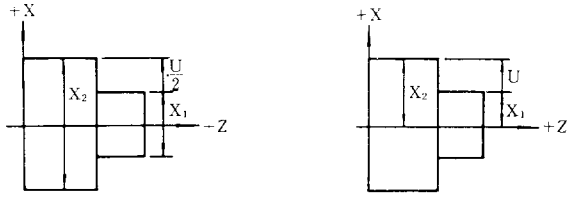
G91 is effective, and in this block, commands become incremental in both the X and Z axes.

2.3.6 X-AXIS DIAMETER/RADIUS SWITCHING

Addresses X and U for X-axis coordinate words are specified by diameter value. This is called diameter designation. The addresses X can be used for designation of both diameter and radius. The switching is made by the setting of parameter #6006D$_3$.

0: Diameter designation

1: Radius designation



(a) In the case of Diameter Designation (b) In the case of radius Designation

Fig. 2.2

Table 2.6

	Diameter Programming	Radius Programming
Address X command	Diameter value	Radius value
Address U command	Diameter incremental value	Radius value incremental value
X-axis position display	Diameter value	
Tool position offset value	Diameter value	
Tool coordinate data for work coordinate system	Diameter value	
Nose radius R	Radius value	
Feedrate F, E in X-axis direction	Radius value/rev Radius value/min	
Radius data I, R for circular interpolation	Radius value	
G90-G94, G70-G76, Parameters for cornering, and multiple cornering, D, I, K, P, Q, R	Radius value	

2.4 RAPID TRAVERSE RATE

2.4.1 RAPID TRAVERSE RATE

The rapid traverse motion is used for the motion for the Positioning (G00) and for the motion for the Manual Rapid Traverse (RAPID). The traverse rates differ among the axes since they are dependent on the machine specification and are determined by the machine tool builders. The rapid traverse rates determined by the machine are set by parameters in advance for individual axes.

When the tool is moved in rapid traverse in each axial direction simultaneously, motions in these axial directions are independent of each other, and the end points are reached at different times among these motions. Therefore, motion paths are normally not straight.

For override rapid traverse rates, Fo, 25% 50% and 100% of the basic rapid traverse rates, are available. Fo is a constant feedrate set by a parameter (#6231).

Range of Rapid Traverse Rate

(1) For each axis, rapid traverse rates can be set by parameters #6280, #6281 at some suitable multiple of 125 p/sec.
(p: Least output increment)

(2) The rapid traverse rate can be set to the upper limit shown below.

Metric Input	24,000 mm/min
Inch Input	2,400 in/min

The upper limit for X-axis speed is half the listed values. The optimum value of upper limit is set according to the machine. Refer to the machine tool builder's manual, for the definite value.

2.4.2 FEED FUNCTION (F- AND E-FUNCTION)

G code listed below must be designated before F, and E function is commanded.

G code	Function
G98	Designation of feedrate in mm/min.
G99	Designation of feedrate in mm/rev.

Note : For the details, refer to 2.8.28, "Feed Function Designation."

Since F, E codes are modal, these codes are effective until next F, E codes are given. However, when G98/G99 are switched, new F code must be designated.

In G98 mode, E code cannot be commanded. If commanded, PROG ERROR "030" will be activated.

2. 4. 2. 1 Feed Per Revolution (G 99 Mode)

(1) Tool feed per revolution of the spindle can be specified with F (normal feed) or E (fine feed).

(2) The feed ranges that can be specified by the F and E codes are as follows.

G 99 Mode, F and E Feed Ranges

		Format	Range of Feed/Revolution
Metric Output	Metric input	F 32	F 0.01 – F 500.00 mm/rev
		E 34	E 0.0001 – E 500.0000 mm/rev
	Inch input	F 24	F 0.0001 – F 19.6850 in./rev
		E 26	E 0.000004 – E 19.685000 in./rev
Inch Output	Metric input	F 32	F 0.01 – F 1270.00 mm/rev
		E 34	E 0.0003 – E 1270.0000 mm/rev
	Inch input	F 24	F 0.001 – F 50.0000 in./rev
		E 26	F 0.000010 – E 50.000000 in/rev

These feed ranges are subject to the following restrictions depending on the spindle speed S.

Metric output	F(E) × S ≦ 24,000 mm/min
Inch output	F(E) × S ≦ 2,400 in./min

Notes :

1. Program feed per revolution within such a range that the X-axis component remains below 12,000 mm/min or 1,200 in./min.
2. This upper limit may still be reduced by the performance limit of the machine.
   Refer to the machine tool builder's manual.

Notes :

1. A command "F0" causes data errors.
2. Any minus value should not be specified for F commands. If specified, the machine will not operate properly.

   EXAMPLE

   F-250 ; ····· Wrong

3. Feedrate commands in the direction of the X-axis must be given in radius.

EXAMPLE

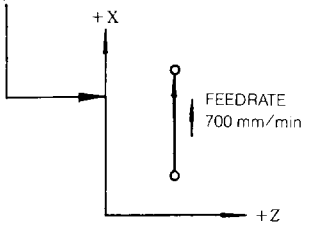
G99 S350 (r/min) ;

G01 U100. F200 ; ··· In case of F32.

In the above case, the feedrate is:

F x S = 2.0 mm/rev. x 350 r/min

= 700 mm/min



4. Values of F command at linear or circular interpolation represent the tangential feedrate when two axes are simultaneously controlled.

EXAMPLE 1

G99 S1000 (r/min) ;

G01 U60. W40. F50 ;

In the above case, the feedrate is

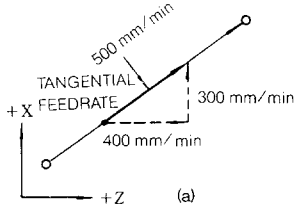
F x S = 0.5 mm/rev x 1000 r/min

= 500 mm/min

$= \sqrt{300^2 + 400^2}$

Z-axis feedrate component

X-axis feedrate component



EXAMPLE 2

G99 S1000 (r/min) ;

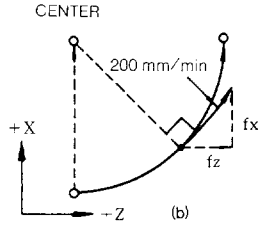
G03 U··· W··· I··· F20 ;

In the above case, the feedrate is:

F x S = 0.2 (mm/rev) x 1000 (r/min)

= 200 mm/min

$= \sqrt{fx^2 + fz^2}$



2. 4. 2. 2 Feed Per Minute (G 98 Mode)

(1) Tool feed can be specified in mm/min or in/min with F codes.

(2) The feed range that can be programmed with F codes is as follows.

G 98 Mode F Code Feed Range

		Format	Range of Feed per Minute
Metric Output	Metric Input	F 50	F 1. – F 24000. mm/min
	Inch Input	F 32	F 0.01 – F 944.88 in./min
Inch Output	Metric Input	F 50	F 1. – F 60960. mm/min
	Inch Input	F 42	F 0.01 – F 24000.00 in./min

Notes :

1. Program feed-per-minute values so that the X-axis speed component will not exceed half the above upper limit feedrates.
   EXAMPLE
   G 98 G 01 U 300. F 1200;
   (Metric output, metric input)
2. The upper limit value is further subject to the limitation imposed by the machine performance. Refer to the machine tool builder's manual. This upper limit value is to be set in parameter #6228.

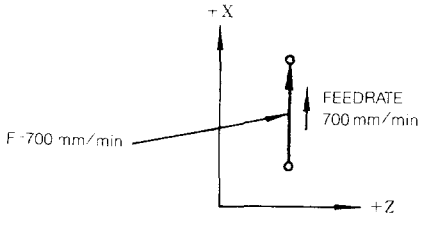
Notes :

1. Do not write F command in F0 or negative values.
2. Commands in the X-axis direction indicate speeds in radius.

   Example

   G98;

   G01 X200. F700 ;



Values of F command at linear or circular interpolation represent the tangential feedrate when two axes are simultaneously controlled.

EXAMPLE 1

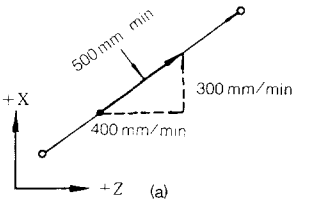
G98 ;

G01 U60. W40. F500 ;

In this case,

$$F = 500 \text{ (mm/min)} = \sqrt{300^2 + 400^2}$$

300 : X-axis component
400 : Z-axis component



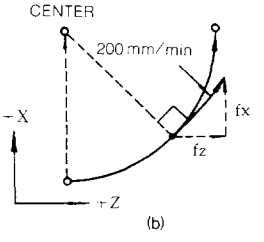
EXAMPLE 2

G98 ;

G03 X··· Z··· I··· F200 ;

In this case,

$$F = 200 \text{ (mm/min)} = \sqrt{fx^2 + fz^2}$$



2. 4. 3 AUTOMATIC ACCELERATION AND DECELERATION

Acceleration and deceleration for rapid traverse and for cutting feed are automatically performed without programming.

2. 4. 3. 1 Acceleration and Deceleration of Rapid Traverse and Manual Feed

In the following operation, the pattern of automatic acceleration and deceleration is linear. (See Fig. 2.3.)

- Positioning (G00)
- Manual rapid traverse (RAPID)
- Manual continuous feeding (JOG)
- Manual HANDLE feeding (HANDLE)

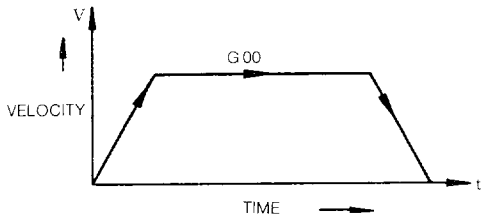


Fig. 2.3

Rapid traverse rate and the acceleration/deceleration constant of rapid traverse rate can be set by parameter. (#6280 to #6287)

As shown in the following operation, the two-step linear acceleration and deceleration can be specified. (independent of each axis) (See Fig. 2.4.)

· Cutting feed (G01 to G03)

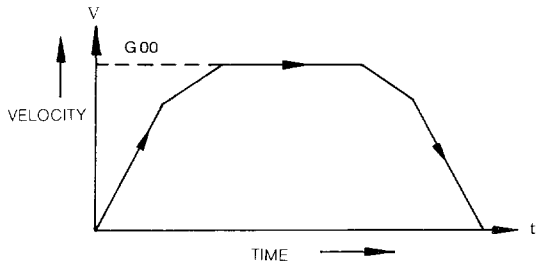


Fig. 2.4

Feedrate time constants are set at 2 msec intervals and feedrate bias is set at 2kpps intervals by parameters. (#6092, #6093)

Note : The automatic acceleration/deceleration parameters are set to the optimum values for the respective machines. Do not change the setting unless it is required for special application.

2.5 SPINDLE-SPEED FUNCTION (S-FUNCTION)

2.5.1 S 2-DIGIT PROGRAMMING (SPECIAL SPECIFICATIONS)

The spindle speed is specified by two digits following the address S (S00 to S99).

For each S code and its corresponding spindle speed (r/min), refer to the machine tool builder's manual.

When a move command and an S code are issued in a block, execution will depend on the machine tool design and construction (Whether the S command is executed together with the move command or after the completion of tool movement). Refer to the machine tool builder's manual.

Once specified, S-code is modal and effective until the next S-code. When the spindle stops at M05 (spindle stop), S-command is stored in memory of the unit.

EXAMPLE

```
G00  S11  M03  ;
           ··· S command
               Spindle CW
     X···  Z···  ;
G01  Z···  F···  ;
     .
     .
     .
G00  X···  Z···  M05  ;
           ··· Spindle stop
     ··· M03  ;
     X···  Z···  ;
G01  Z···  F···  ;
     S22  ;
     X···  Z···  F···  ;
     .
     .
```

S11: Effective

S11: Effective

S22: Effective

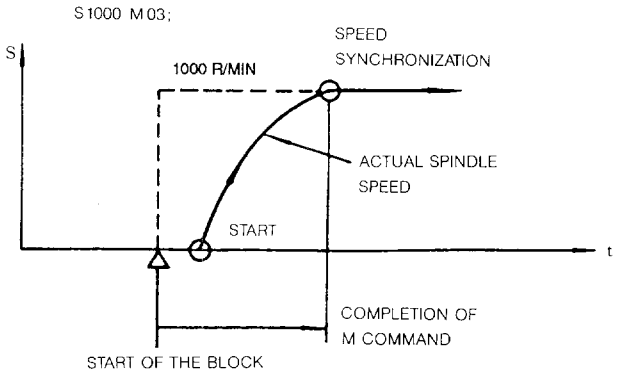
Note : The two-digit BCD output is sent to the machine when S and two-digit command is issued.

2.5.2 S4-DIGIT PROGRAMMING A†

(1) Four digits following S (S□□□□) are used to specify the spindle speed in r/min.

(2) When S command is given in a block together with M03 (spindle forward running) or the M04 (reverse running), the control proceeds to the next block after the spindle speed reaches the speed given by the S code. For details, refer to the machine tool builder's manual.

EXAMPLE



2.5.2 S 4-DIGIT PROGRAMMING A† (Cont'd)

(3) S commands are modal. Although the spindle stops at the M05 command, the S command is retained. Therefore, when M03 (or M04) is given, the spindle runs according to the S command.

(4) When S command is changed after the spindle start by M03 or M04, S command should be given within the range of spindle speed selected by spindle gear.

Notes :

1. The lower limit of the spindle speed depends on the spindle drive. Refer to the machine tool builder's manual for the low-speed limit. Negative S commands must not be programmed.
2. When the control is provided with the S 4-digit command function, the "Spindle speed override" option can be built into it.
3. With machine tools with which the main spindle gear ratio changes can be specified by M codes, first write the applicable M code to preselect the desired gear ratio, and then, write the S command. Refer to the data of the machine tool builder for the number of gear ratios, the speeds at various gear ratios, and other details.
4. When the control is provided with this function, the spindle maximum speed commanding function with the instruction "G50 S··· ;" can be used.

2.5.3 S 4-DIGIT PROGRAMMING B†

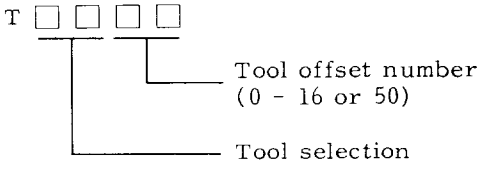
(1) This function is to modify the S4-digit command A output freely through the programmable machine interface.

(2) Basically, this function is used in the same way as the S 4-digit command A function, but it is normally used to set the manually controlled spindle speeds controlled by the rotary switch on the machine control station corresponding to S command speeds. For the details of S command speeds, refer to the machine tool builder's manual.

2.6 TOOL FUNCTION (T-FUNCTION)

2.6.1 T 4-DIGIT PROGRAMMING

(1) Four digits following the address T specifies the tool number.



(2) For applicable tool number to be specified, refer to the machine tool builder's manual.

Specifications	Tool Offset No.
a	0-16
b	0-50

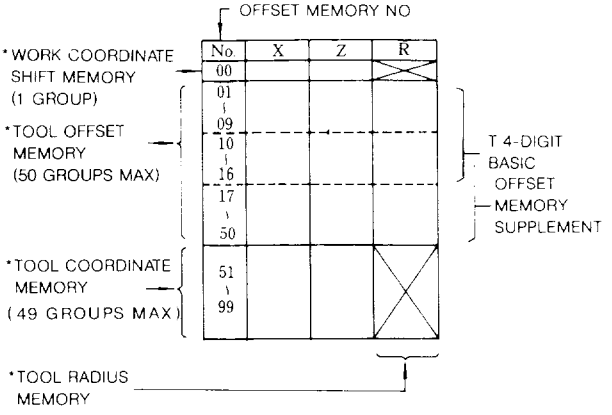
Notes :

1. When the tool number is changed by the T command, a turret lathe begins to index the tool instantaneously. Therefore, the turret should be removed, before the command, from the area where an accidental collision might occur.
2. Tool offset number 00 cancels the tool offset.

2.6.2 TOOL OFFSET MEMORY †

The area in which tool position offset values, tool radius compensation values, and other compensation data are stored is called Offset Memory.

(1) The entire memory areas of Offset Memory including the options are as shown below.



Note : For the actually usable range within the above Offset Memory, refer to the machine tool builder's manual.

(2) The "tool offset Nos." specified by the T function directly correspond to the "offset memory Nos.," and their contents are used for various compensations. However, the tool coordinate memory Nos. (for setting the work coordinate system) correspond to the tool selection Nos. in the T function. The work coordinate shift memory is an independent function, not related to the T function.)

(3) Write these data in the memory, before starting to operate the machine under automatic control. For the writing procedure, refer to 4.3.5, "Displaying and Writing Tool Offset Values." For writing into Tool Coordinate Memory, follow the procedure described in 6.2.3, "Work Measurement Value Direct Input†."

2.6.3 TOOL POSITION OFFSETS

When the tool offset number is specified, the offset value corresponding to the tool offset number is added algebraically to the command value in the program and the tool is moved to the offset position. Therefore, the difference between the coordinate values of the programmed tool tip and the actual tool tip must be stored into tool offset memory in advance as the offset value.

When the coordinate value of the actual tool tip has changed due to tool wear or some other reasons, the tool position offset values should be set again. Thus, the programmed machining is attained without correcting the program.

(1) Range of tool position offset value

The programmable range of tool offset value is shown below.

Output	Input	Setting Range
Metric Output	Metric input	0 – ±8388.607 mm
	Inch input	0 – ±330.2601 in.
Inch Output	Metric input	0 – ±9999.999 mm
	Inch input	0 – ±838.8607 in.

(2) Sign of tool position offset values

Store the tool position offset values in the Offset Memory. The offset value is the deviation from the tool tip position of the reference tool which is determined as zero.

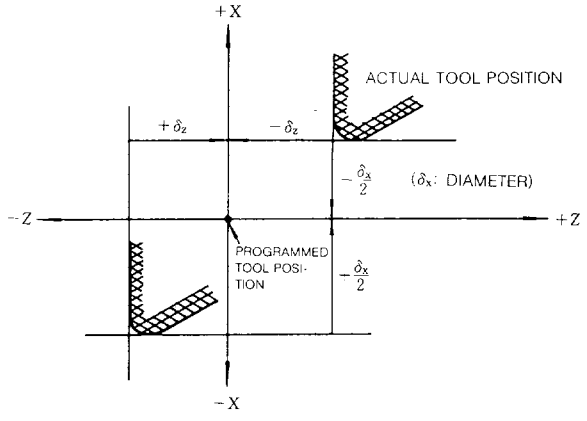


Fig. 2.5

(3) Description of tool position offset motion

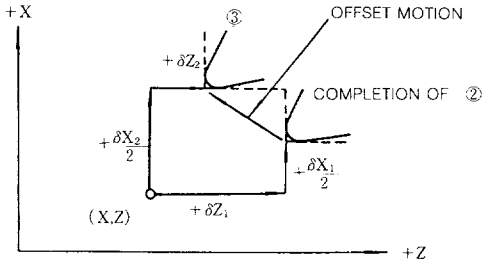
As mentioned above, when the tool specified by the address T and 4 digits is moved, the offset value corresponding to the tool offset number is added to the command value in the program algebraically and the tool tip is moved to the offset position.

When there is no move command in the block, the tool moves only by the offset value. Once, the tool offset number is designated, the tool moves always to the offset position until another number is designated. When the other offset number is designated or the offset value is changed, the offset value is compensated for by the amount of the difference between the old and new offset values.

OFFSET VALUE

T101 ↓ $(+\delta X_1, +\delta Z_1)$

T115 ↓ $(+\delta X_2, +\delta Z_2)$



EXAMPLE

```
T101 ; ..................... ①
  .
  .
  .
G01 X··· Z··· F(E)··· ; ····· ②
T115 ; ..................... ③
                    (Block of the
                    offset motion)
```

(4) Move speed with tool offset

The move speed of tool offset is determined by the feedrate command that is effective in the block. Therefore, the feedrate command (G00 or G01 F···) should be issued before or in the block containing the tool offset number.

EXAMPLE

```
G50  X···  Z···  ;
G00  S···  M03  T0108  ;
     X···  Z···  ;
     .
     .
     .
```

Offset motion is made at the rapid traverse rate.

(5) Instructions for commanding tool position offset

Tool position offset is executed by designating the tool offset number corresponding to the actual tool must be designated.

a. Tool offset starts at the block in which the T-code is commanded. When T-code is read, the tool selection signal (BCD) is fed and the tool starts to move by the offset value corresponding to the tool offset number. Since T code is modal, it is retained until the other T code is designated.

EXAMPLE

```
G00  T0202  ; ···
```

The tool number N02 is selected. Tool offset motion is made according to the contents of the tool offset number 02.

b. When the tool offset value must be changed, the T-code whose tool offset number is rewritten should be commanded again.

EXAMPLE

```
G00  T0202  ;
G01  X···  Z···  F···  ;
     .
     .
     .
G01  T0216  ;
```

Tool offset number 02 is replaced with 16. Tool offset motion is made at the cutting feedrate.

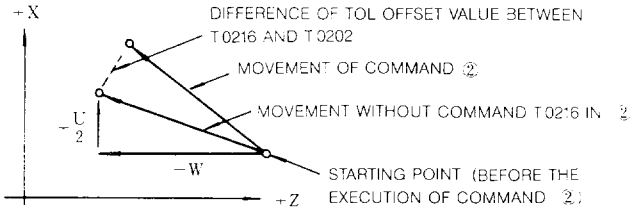
Note that if the tool number is changed in this case, the tool indexing motion starts.

c. The angle of taper cutting can be changed by the following procedure.

T code for change of tool offset number should be commanded in the block together with cutting feed command.

EXAMPLE

```
① G00  T0202  ;
  G01  X···  Z···  F···  ;
       .
       .
       .
② G01  U+···  W-···  F···  T0216  ;
```



When the T command and the move command are issued in the same block, the tool nose moves to the offset position. Therefore, in the above case, the taper angle is corrected by the difference of the offset value between T0202 and T0216.

d. When the tool position offset is required to cancel, the T code with the tool offset number 0 or 00 (T □□ 00) must be commanded. The tool position offset is instantaneously cancelled.

EXAMPLE

```
G00  T0202
G01  X···  Z···  F···  ;
     .
     .
     .
G01  U+···  W-···  F···  T0216  ;
     .
     .
     .
③ G00  X···  Z···  T0200  ; ····
```

The offset motion is cancelled. Tool moves according to the position specified by X and Z.

The block ③ of EXAMPLE can be divided into two blocks.

```
G00  X···  Z···  ;
T0200  ; ·····
```

Only cancel motion is made at rapid traverse rate.

Notes :

1. Tool position offset is cancelled by RESET operation.
2. The tool offset must be cancelled before M02 or M30 is commanded.
3. The tool offset should be cancelled also before Automatic Zero Return (G28) is commanded.
4. When the control is reset by M02 or M30 command or by executing RESET operation, the tool offset number becomes 0 (or 00).
5. When the Zero Return (auto or manual) is executed, the tool offset is cancelled automatically.
6. The tool offset must be also cancelled before Zero Return Check (G27) is commanded. If the G27 is commanded at the state where the tool offset is effective, the control will be the state of Zero Return check error, because the tool offset value is added to the programmed position.

2.6.4 WORK COORDINATE SYSTEM SHIFT †

With this function, coordinate systems set by G50, the Work Coordinate System Setting function, etc. can be shifted through desired distances.

(1) Shift values in the X and Z axes can be written into the Work Coordinate System Shift Memory (one group) with which the offset memory No. is "00," by the same procedure as for writing tool offset values.

(2) The written shift values become effective from the moment described below.

a. When G50 coordinate system is set

b. When G50GT work coordinate system is set

c. When automatic coordinate system is set

d. Position Absolute display is reset by ORG key

That is, when these coordinate systems listed above are set, the shift values are simply added. Tools are not shifted.

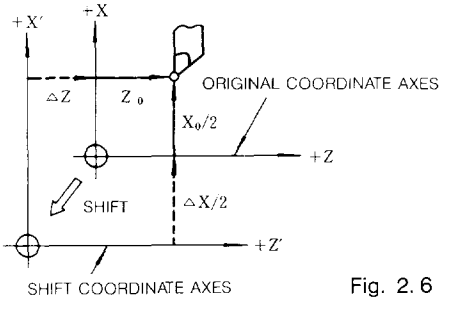


Fig. 2.6

For positive shift values ΔX and ΔZ, the coordinate axes are shifted in the direction shown above. Xo and Zo are original coordinate system setting values.

(3) This shift function is executed at each time any of the conditions described in a, b, c, and d is met.

(4) When the contents of Work Coordinate System Shift Memory are rewritten, the new shift values become effective from the moment the operation a, b, c, or d above is subsequently executed.

(5) The procedure of 5.2.3, " MEASURED WORKPIECE VALUE DIRECT INPUT" is effective for the Work Coordinate Shift Memory with an offset memory No. "00."

Notes :

1. The shift command by the Work Coordinate Shift function can not be cancelled unless the setting value is changed to "0." No reset operation is effective in cancelling it.
2. T □□ 00 ; ····· Tool position offset cancel
   G50 T□□00 ; ··· Work coordinate system setting

   The tool offset No 00 in these instructions has nothing to do with the contents of Work Coordinate Shift Memory.
3. When G50 coordinate system is set or when position absolute display is reset by ORG key 1, parameter #6018 D_7 determines whether work coordinate system shift amount is effective or not.

2.6.5 NEW COORDINATE SYSTEM SETTING FUNCTION

2.6.5.1 Overview

A completely new approach to coordinate system setting is employed in this function. The following are the features.

(a) A coordinate system is a machine coordinate system.

(b) The tool nose point can always be displayed on the current value display (absolute).

(c) There is no need to set the coordinate system in the program.

2.6.5.2 Functions

What kind of coordinate system setting is performed and at what frequency ?

2.6.5.3 Operation Expression for Coordinate System Setting

The following is the operation expression for coordinate system setting at various time frequencies ?

X-axis coordinate value = Machine position + tool coordinate memory + work coordinate system shift amount

Z-axis coordinate value = Machine position + tool coordinate memory + work coordinate system shift amount

(1) The machine position is called the position machine.

(2) The tool coordinate memory value number is of two types; the number when the timing for the next coordinate system setting is manual and when it is automatic.

(3) The work coordinate system shift amount is called the offset T00, X, or Z data.

2.6.5.4 Timing of Coordinate System Setting under the Manual Mode

Under the manual mode, coordinate system setting is made with the following times (a) to (c). The tool coordinate memory number is created from the tool number binary value set in I/O input #13174 (TP1) to #13178 (TP8), to be used for operation.

The coordinate system can also be set up inside the NC, or by a request from the sequencer. When set by a request of the sequencer, coordinate system setting is executed to turn on output #12194 (end of coordinate system setting output) when input #13127 (coordinate system setting request input) turns on.

(a) Upon zero point return. (NC internal setting at label skip, or sequencer setting for other cases.)

(b) On the tool setter, when the tool nose contacts the sensor upon measurement. (NC internal setting)

(c) When turret indexing is performed by manual operation. (Sequencer setting)

Note: Coordinate system cannot be set by manual operation, when parameter #6011 D0 = 1 [when the tool coordinate memory number follows the last two digits of T**$$].

2.6.5.5 Timing of Coordinate System Setting under the Automatic Mode

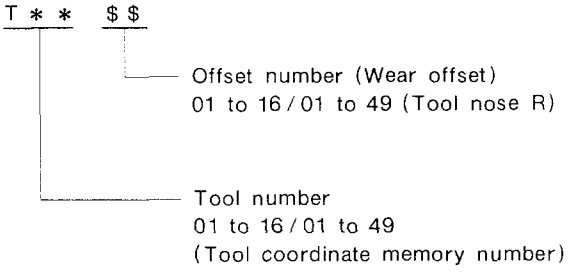
Under the automatic mode, coordinate system setting is performed when the turret is called up by the T code. The tool coordinate memory number uses the turret number commanded on the command screen or the offset number for operation.

Unlike the conventional offset method, the T code command in the coordinate system setting specification is given as follows.

The setting of parameter #6011 D0 decides whether to set the tool coordinate memory number at the front two digits of T4-digit, or at the last two digits of the T4-digit.

The execution of the coordinate system setting differs as follows, by the parameter setting.

(1) When set at the front two digits (#6011 D0 = 0)



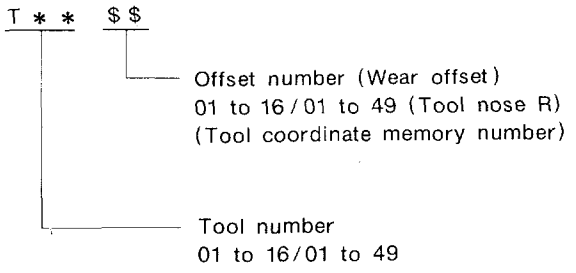
Note: The tool coordinate memory number is: (tool number + 50). Thus, the tool coordinate memory value selects the contents of (tool number + 50).

(a) By executing the T**$$ command, the turret is called up wherever the tool post is located, and is moved for the offset amount of the offset number, to execute the coordinate system setting corresponding to the selected tool coordinate memory number.

(b) By executing the T**00 command, the turret is called up wherever the tool post is located, and the offset amount cancel movement is executed, to execute the coordinate system setting corresponding to the selected tool coordinate memory number.

(c) The tool nose coordinate system is always set by the coordinate system setting.

(2) When set at the back two digits (#6011 D0 = 1)



Note : The tool coordinate memory number is: (offset number + 50). Thus, the tool coordinate memory value selects the contents of (offset number + 50).

(a) By executing the T**$$ command, the turret is called wherever the tool post is located, and is moved for the offset amount of the offset number, to execute the coordinate system setting corresponding to the selected tool coordinate memory number.

(b) By executing the T**00 command, the turret is called wherever the tool post is located, and the offset amount cancel movement is executed. No coordinate system setting is performed.

(c) The tool nose coordinate system is always set by the coordinate system setting.

2.6.6 NOTES WHEN USING THE CONVENTIONAL G50T ★★★★ FUNCTION

When using the coordinate system setting specification, do not give the G50 T**** command. Error will occur if commanded.

2.7 MISCELLANEOUS FUNCTIONS (M-FUNCTION)

The miscellaneous function is specified with the address M and a maximum 3 digits. The function of each M code (M00 to M99) is determined by the machine, except for several M codes. Refer to the machine tool builder's manual for the function of M codes except for the following M codes concerned with the control.

2.7.1 M CODES FOR STOP (M00, M01, M02, M30)

To stop the NC control and machine, the following codes are provided.

M00: Program stop

M01: Optional stop

M02: End of program

M30: End of tape

These commands stop the advance reading of the control. For these M codes, M 2-digit BCD code and their respective decoded signals are output.

2.7.2 M CODES FOR INTERNAL PROCESSING (M90 TO M109)

M90 through M109 are for internal processing. Even when they are programmed, no external output signal (BCD and decoded output) is sent.

M96 †: Tool radius compensation: circular path mode

M97 †: Tool radius compensation: intersection computing mode

M98: Subroutine program call

M99: Subroutine program end

M100 to 109: Not used (for special application)

2.7.3 BUFFERING FUNCTION (M93, M92) †

(1) The following M codes are issued for buffering function.

M code	Meaning
M92 ◥	1-block buffering
M93	4-block buffering

Note: When power is applied, the current M code is changed to the M code marked with ◥. However, it is not changed by RESET operation.

(2) 4-block buffering (M93)

When M93 ; command is given, the control enters the 4-block buffering mode, which remains until M92 is commanded subsequently. In this mode, up to 4 blocks of data are read in advance for subsequent operation. With programs in which the operation time for the 4 blocks read in advance is longer than the reading and processing time of the subsequent 4 blocks, interruption between blocks can be eliminated. This function is effective in avoiding a shiny streak on the workpiece caused by feed stop between blocks.

(3) 1-block buffering (M92)

When M92 command is given, the 4-block buffering mode is cancelled, and the 1 block buffering mode is restored.

Note : While the tool radius is being compensated for with the M93 function, up to two blocks not containing move commands are permitted, and as the result, up to 6 blocks may be read in advance.

2.7.3 BUFFERING FUNCTION (M93, M92)† (Cont'd)

EXAMPLE

```
N51 M93 ; ——— Start of 4-block advance reading.
N52 G01 U··· F··· ;  ┐
N53 X··· Z··· ;      │ Stop between blocks for tool radius compensation or other calculation can be avoided.
N54                  │
    .                │
    .                │
    .                ┘
M58 M92 ; ——— Cancelling 4-block advance reading.
```

2.7.4 CIRCULAR PATH MODE ON/OFF ON TOOL RADIUS COMPENSATION (M97, M96) †

These M codes are effective when the control is provided with the tool nose radius compensation option.

(1) The following M codes are used.

M code	Meaning
M96 ◥	Tool radius compensation circular path on
M97	Tool radius compensation circular path off (Execution of intersection point)

Note: When power is applied, the current M code is changed to the M code marked with ◥.
However, it is not changed by RESET operation.

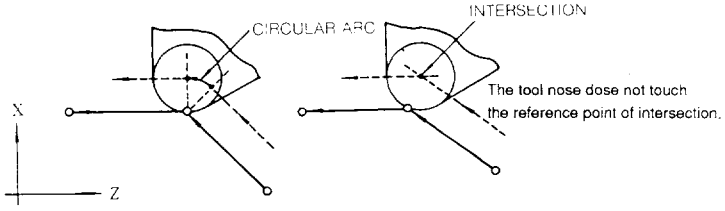
(2) With the tool radius compensation mode by G41 to G44, the locus of the tool (center of tool radius) for commanded workpiece contour lines with the angle between tangents larger than 180° is in the following two categories.

a. M96 mode

The center of the tool nose radius describes a circular arc around the perimeter in the contour line.

b. M97 mode

The center of the tool nose radius moves along the locus that is formed by straight lines shifted from the contour line by the distance equal to the tool radius.



M96 mode (circular arc) M97 mode (calculation of intersection)

Fig. 2.8

(3) Commands of M96 and M97 become effective from the edge in the following command blocks.

```
a. G01 X··· Z··· F··· ;  ┐
   (G01) X··· Z··· M96   │ From the move around the edge in this block.
         (or M97) ;      ┘

b. G01 X··· Z··· F··· ;  ┐
   M96 (or M97) ;        │ From the move around the edge in this block.
   (G01) X··· Z··· ;     ┘
```

2.7.5 SOUBROUTINE PROGRAM (M98, M99)

With this function, subroutine programs which have been numbered and stored in advance are called and executed as many times as desired.

(1) The following M codes are used for this function.

M code	Meaning
M98	Call of subroutine program
M99	End of subroutine program

(2) Call of subroutine program (M98)

M98 P··· Q··· L··· ;

With this command, the subroutine program starting with a sequence No. following Q in the part program with the program No. specified by P is called and is executed L times.

However, when

- P is omitted:
  subroutine program following the sequence No. Q in the main program is called.
- Q is omitted:
  subroutine program starting at the leading end of the program No. specified by P is called.
- L is omitted: execution is only once.

Subroutine programs can be nested up to 4 times.

(3) End of subroutine program (G99)

M99 ; is written at the end of subroutine program to end it.
When this code is written, the operation returns to the block immediately following the main block in which the subroutine program was called after the execution of the subroutine program.

M99 P··· ;

When this is written at the end of a subroutine program, the operation returns to the sequence No. specified by P in the main program.

(4) Simple jump command

M99 P ••• ;

When this command is used in the main program, the operation simply jumps to the sequence No. specified by Q in the main program. If Q is omitted, the program simply jumps to the leading end of the main program.

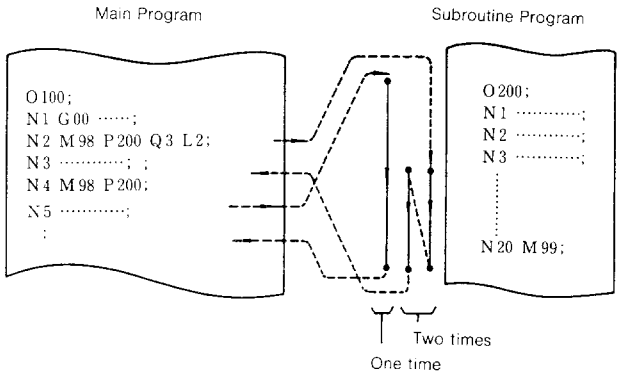
```
N1 G50 X0 Z0 ;
N2 G00 ... ;
     .
     .
     .
     .
N9 M99 ;
```

Writing multi blocks (10 lines maximum) of this program and executing cycle start make endless operation.

Notes :

1. When the program No. specified by address P and the sequence No. specified by Q are not found, alarm code 041 is displayed.
2. While command L for the number of repetitions is under execution, the remaining number of repititions can be displayed. For details refer to 4.3.2.2.
3. This function can be used when subroutine programs are stored in the part program memory. Main programs can be commanded through NC tapes or the part program memory.
4. When subroutine programs are nested more than 4 times, alarm code "042" is displayed.

EXAMPLE



2.7.6 OTHER M CODES

(1) How to use the other M codes other than the above depends upon the machine. Refer to the machine tool builder's manual.

Tadle 2.7 Typical Example of M Codes for Machine

M code	Meaning	Remarks
M03	Spindle forward running	Direct switching from M03 to M04 cannot be done. M05 must be inserted between them.
M04	Spindle reverse running	
M05	Spindle stop	
M08	Coolant on	–
M09	Coolant off	

(2) When these M codes are commanded in the same block with move command, execution will depend on the machine tool design and construction. (Whether the M commands are executed simultaneously with or after completion of move command.)

(3) For these M-code commands, the control outputs M 2-digit BCD codes.

2.7.7 M 3-DIGIT BCD OUTPUT †

When the control is provided with the M 3-digit BCD output option, it can command M 3-digit codes between M00 and M999.

(1) M codes between M00 and M89, and between M110 and M999 are output in 3-digit BCD codes.

(2) M90 through M109 are internal processing M codes, and no BCD code for them is output. See 2.7.2 M CODES FOR INTERNAL PROCESSING.

2.7.7 M 3-DIGIT BCD OUTPUT† (Cont'd)

(3) With M00, M01, and M30, decode signals are output in addition to the BCD output. See 2.7.1, " M CODES FOR STOP."

(4) The specific usages of the M 3-digit codes depends on machine tool design. Refer to machine tool builder's manual.

2.7.8 HIGH-SPEED M FUNCTION

This function is used to execute the M function at high-speed without the need of the ending response.

The M code is not output when the M code is commanded, but the M decode output is set/reset. Thus, there is no need for the M code decode processing and FIN processing in the programmable controller.

The M code that perform the high-speed M function processing is preset in the parameter. (There are both a setting parameter and a resetting parameter.)

When resetting by the parameter, it can be set to hold or reset the decoding output.

2.7.8.1 I/O Channel

(1) For decode output

±1224	MD7	MD6	MD5	MD4	MD3	MD2	MD1	MD0

2.7.8.2 Parameters

(1) High-speed M function

#6007 D5 ..... 0: The function is disabled
1: The function is enabled

(2) M code setting parameter for setting the decode output

The M code is set in the parameter corresponding to the decode output bits.

Up to four M codes can be set in a single parameter.

(a) #6644 ... For setting the M code corresponding to the decode output 'MD0 to MD3'

±1224						MD3	MD2	MD1	MD0
±6644						××	××	××	××

(Commanded by 2 digits)

(b) #6645 ... For setting the M code corresponding to the decode output 'MD4 to MD7'

±1224	MD7	MD6	MD5	MD4			
±6645	××	××	××	××			

(Commanded by 2 digits)

(Example of setting)

#6644 ... 63 62 61 60
#6645 ... 67 66 65 64

(3) M code setting parameter for resetting the decode output

Sets the following parameters, the same as in the setting parameter of (2).

(a) #6646 ... For setting the M code corresponding to the decode output 'MD0 to MD3'

±1224						MD3	MD2	MD1	MD0
±6646						××	××	××	××

(Commanded by 2 digits)

(b) #6647 ... For setting the M code corresponding to the decode output 'MD4 to MD7'

±1224	MD7	MD6	MD5	MD4			
±6647	××	××	××	××			

(Commanded by 2 digits)

(Example of setting)

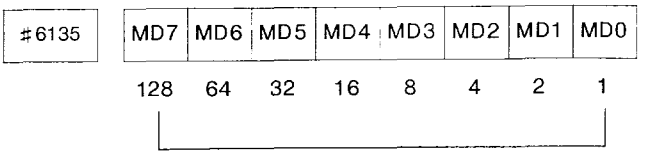
#6644 ... 73 72 71 70
#6645 ... 77 76 75 74

(4) Decode output hold/reset setting parameter (upon reset)

(a) Sets whether to hold or reset the decode output upon reset.

#6135 D0 to D7

(b) When the decode output is to be held, the numerals corresponding to each bit are added to the total. The result is output to #1224.



Value of each bit when the decode output is to be held

(Example of setting)

M61 ... Set MD1 output
M71 ... Reset MD1 output

M64 ... Set MD4 output
M74 ... Reset MD4 output

M67 ... Set MD7 output
M77 ... Reset MD7 output

When the decode output is to be kept upon reset, set the parameters as follows.

#6644 00006100
#6645 67000064 M code setting for setting

#6646 00007100
#6647 77000074 M code setting for resetting

The value of each bit corresponding to MD4 and MD7 are added to #6135.

16 + 128 = 144

144 is set in #6135.

#1224 is 1001000 upon reset.

2.7.8.3 Notes

(1) Do not set the following M codes when setting the original M code.

- M00, M01, M02, M30
  When these M codes are set, the original M code functions are lost; the high-speed M code has the priority.
- M90 to M99
  When these M codes are set, the M code functions have the priority.

Check that the above M codes are not found when setting the M code.

(2) If an other process is waiting to be completed in the same block, it waits for the first process to be completed.

(3) Do not command two or more M codes in the same block.

* If executed, alarm occurs.

2.7.8.4 Example of High-speed M Function Processing

The following are simple examples of the high-speed M functions.

(1) Sets the parameter (for setting, resetting).

#6644 63 62 61 60 M code setting for
#6645 67 66 65 64 setting

#6646 73 72 71 70 M code setting for
#6647 77 76 75 74 resetting

(2) 01;

M60; The left program is executed.
M61; The bits corresponding to #1224 D0 to
M62; D7 are set.
M63; M60 to M67 do not wait for FIN.
M64; If an M code is to be held, set the total
M65; of each bit corresponding to #6135.
M67; The value to be held is set in #1224 at
M30; the end of the program.

(3) 02;

M70; The left program is executed.
M71; The bits corresponding to #1224 D0 to
M72; D7 are reset.
M73; The M codes M70 to M71 do not wait for
M74; FIN.
M75;
M76;
M77;
M30;

2.8 PREPARATORY FUNCTIONS (G-FUNCTION)

2.8.1 LIST OF G CODES

Address G, plus up to 3 digits specify the meaning of the block. Table 2.8.1 gives G codes and their groups.

(1) G codes are broadly classified into the following two types.

	Meaning
Modal G-code	G-code effective until the other G-code of the same group is commanded.
Non-modal G-code	G-code effective only in the commanded block.

(2) G codes in groups from 01 through 11 are modal. When the control is energized with the power switch, the G codes marked with ◤ in Table 2.8.1 are automatically selected.

2.8.1 LIST OF G CODES (Cont'd)

(3) G codes of * group in the Table 2.8.1 are non-modal. They should not be commanded together with the other G codes in one block.

(4) The modal G codes can be commanded mixedly in a block.

(5) G codes in Class B are basic, and those in Class O are options. The use of optional G codes is determined by the machine tool design. See the machine tool builder's manual.

(6) Standard G codes can be coverted to special G codes I by parameters. This is a basic feature, and, when parameter #6005D7 is set to 1, standard G codes are converted to special G code I.

(7) When the special G code II option is incorporated in the control, the setting of parameter #6005D7 to 1 will convert G codes to special G codes II. Setting the parameter to 0 will reconvert the G codes to the standard G codes.

2.8.2 POSITIONING (G00, G06)

2.8.2.1 Positioning (G00)

(1) G00 X(U)··· Z(W)··· ;

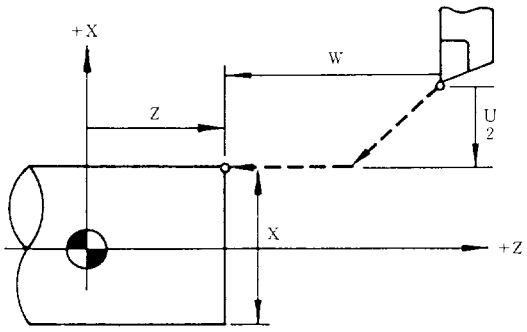
This command moves a tool at rapid traverse rate to the point (X, Z) in the coordinate system set by the G50 command or moves it away by (U, W) from the present point for each axis independently.

(2) For the rapid traverse rate, as it depends upon the machine, refer to the machine tool builder's manual.

EXAMPLE

X-Axis: 12 m/min

Z-Axis: 6 m/min



(3) Along the axes specified by G00, the machine slide moves in rapid traverse rates, independently of each other. The resultant tool locus may not be a straight line, and when working out the program, care must be taken to avoid fouling between the tool and the workpiece.

(4) G00 is a modal G code in the 01 group. When it is commanded, it remains effective until other G codes in the 01 group are commanded.

(5) For the positioning with G00, the pulse distribution is started only after the ERROR DETECT state is turned on, and the program advances to the next block only upon the activation of the ERROR DETECT state after the completion of the pulse distribution. When this G code is used, therefore, the workpiece edges are machined true, and rounding is avoided.

(6) Notes:

a. The ERROR DETECT ON state means the decrease of the servo lag pulses to the permissible level after the pulse distribution for move command.

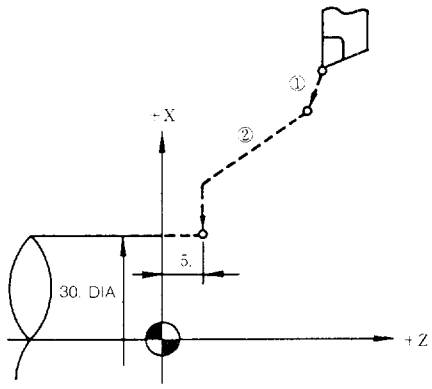
b. When T code is commanded, G00 should be put in the T-code block. G00 is required for designation of tool traverse rate for tool offset motion using T code.

EXAMPLE

G50 X150. Z100. ;

① G00 T0101 S1000 M03 ; ···
··· G00 for designation of traverse rate for tool offset motion

② (G00) X30. Z5. ; ··· G00 can be omitted in positioning.



B: Basic
O: Optional

Table 2.8 List of G Codes

G Code	Special G Code I	Special G Code II	Group	Function		Section
◤ G00	◤ G00	◤ G00	01	Positioning (rapid traverse feed)		B
◤ G01	◤ G01	◤ G01		Linear interpolation, angle programming for linear interpolation		B, O
G02	G02	G02		Circular interpolation CW, (radius R designation)		B, O
G03	G03	G03		Circular interpolation CCW, (radius R designation)		B, O
G04	G04	G04	*	Dwell		B
G06	G06	G06		ERROR DETECT OFF positioning		B
G10	G10	G10		Tool offset value setup		O
G11	G11	G11	01	Beveling	Cornering	O
G12	G12	G12		Rounding		O
G20	G20	G70	05	Inch input specification		O
G21	G21	G71		Metric input specification		O
G22	G22	G22	01	Radius programming for circular interpolation CW		O
G23	G23	G23		Radius programming for circular interpolation CCW		O
G27	G27	G27	*	Reference point return check		B
G28	G28	G28		Automatic return to reference point		B
G29	G29	G29		Return from reference point		B
G30	G30	G30		Retunrn to 2nd reference point		O
G31	G31	G31		Skip function		O
G32	G33	G33	01	Threadcutting, continuous threadcutting, multi-start threadcutting		B, O
G34	G34	G34		Variable lead threadcutting		O
G35	G35	G35	*	Tool set error compensation		O
G36	G36	G36	07	Stored stroke limit 2nd area ON		O
G37	G37	G37		Stored stroke limit 2nd area OFF		O
G38	G38	G38	08	Stored stroke limit 3rd area ON		O
G39	G39	G39		Stored stroke limit 3rd area OFF		O
◤ G40	◤ G40	◤ G40	06	Tool radius compensation cancel		O
G41	G41	G41		Tool radius compensation No. 1		O
G42	G42	G42		Tool radius compensation No. 2		O
G43	G43	G43		Tool radius compensation No. 3		O
G44	G44	G44		Tool radius compensation No. 4		O

◤ shows the G codes selected when the control is powered or reset.

Notes :

1. The following G codes for initial state when power is applied can be set by parameters.

Group	G code	Parameter
01	G00 or G01	#6005 D_6
04	G98 or G99	#6005 D_1
03	G90 or G91	#6005 D_0

2. When the control is reset, whether G code of 01 group should be G00 or kept as the current one can be set by parameter #6005D_6.
3. Radius programming for circular interpolation can be made by G02 and G03 instead of G22 G23, respectively.
4. Cornering can be programmed by G01 instead of G11 and G12. Refer to 2.8.7 Cornering.
5. Initial states of G codes of 05, 07, 08 groups when power is applied are determined by their respective setting data (#6001D_0, D_1, D_2).

2. 8 PREPARATORY FUNCTIONS (G-FUNCTION) (Cont'd)

Table 2. 8 List of G Codes (Cont'd)

B: Basic
O: Optional

G Code	Special G Code I	Special G Code II	Group	Function		Section
G 50	G 92	G 92	*	Coordinate system setup		B
				Maximum spindle revolution setup, work coordinate system setup		O
G 51	G 51	G 51	*	Return of current display value to origin		O
G 65	G 65	G 65		User macro simple call		O
G 66	G 66	G 66	09	User macro modal call		O
◤ G 67	◤ G 67	◤ G 67		User macro modal call cancel		O
G 68	G 68	G 68	10	Mirror image by programming ON		O
◤ G 69	◤ G 69	◤ G 69		Mirror image by programming OFF		O
G 70	G 70	G 72	*	Finishing cycle	Multiple repetitive cycles	O
G 71	G 71	G 73		Stock removal in turning		O
G 72	G 72	G 74		Stock removal in facing		O
G 73	G 73	G 75		Pattern repeating		O
G 74	G 74	G 76		Peck drilling in Z-axis		O
G 75	G 75	G 77		Grooving in X-axis		O
G 76	G 76	G 78		Automatic threadcutting cycle		O
G 90	G 77	G 20	01	Turning cycle A		B
G 92	G 78	G 21		Threading cycle		B
G 94	G 79	G 24		Facing cycle B		B
G 96	G 96	G 96	02	Constant surface speed control		O
◤ G 97	◤ G 97	◤ G 97		Constant surface speed control cancel		O
◤ G 98	◤ G 94	◤ G 94	04	Feed per minute (mm/min)		B
◤ G 99	◤ G 95	◤ G 95		Feed per revolution (mm/rev)		B
	G 90	◤ G 90	03	Absolute command		B
	G 91	G 91		Incremental command		B
G 122	G 122	G 122	11	Tool registration start	Tool life control	O
◤ G 123	◤ G 123	◤ G 123		Tool registration end		O
G 111	G 111	G 111	*	Taper multiple beveling/rounding		O
G 112	G 112	G 112		Arc multiple beveling/rounding		O

◤ shows the G codes selected when the control is powered or reset.

Notes :

1. The following G codes for initial state when power is applied can be set by parameters.

Group	G code	Parameter
01	G 00 or G 01	#6005 D_6
04	G 98 or G 99	#6005 D_1
03	G 90 or G 91	#6005 D_0

2. When the control is reset, whether G code of 01 group should be G00 or kept as the current one can be set by parameter #6005D_6.
3. Radius programming for circular interpolation can be made by G02 and G03 instead of G22 G23, respectively.
4. Cornering can be programmed by G01 instead of G11 and G12. Refer to 2.8.7 Cornering.
5. Initial states of G codes of 05, 07, 08 groups when power is applied are determined by their respective setting data (#6001D_0, D_1, D_2).

2.8.2.2 Error Detect off Positioning (G06)

(1) G06 X(U)··· Z(W)··· ;

With this command the positioning process is identical to that of G00 except for the following aspects.

a. G06 is a non-modal G code in the * group. It is effective only in the programmed block.

EXAMPLE

G00 X··· Z··· ;
G06 X··· Z··· ; —— Move by G06
X··· Z··· ; —— Move by G00

b. **With the positioning of G06, the positioning pulse distribution is immediately started on the completion of the pulse distribution for the preceding block, after making an ERROR DETECT check, and the program advances to the next block after the completion of the pulse distribution process. For this reason, wrokpiece edges are rounded to the extent of servo lag pulses.**

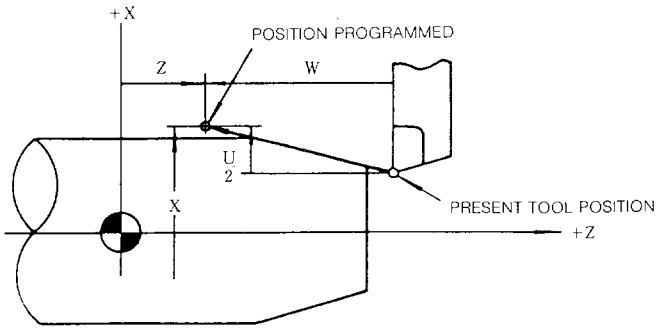
Note: The ERROR DETECT ON/OFF signals (SMZ) are effective only for cutting feeds, and have no influence on the motion under G00 and G06.

2.8.3 LINEAR INTERPOLATION (G01) †

· G01 X(U)··· Z(W)··· F(E)··· ;

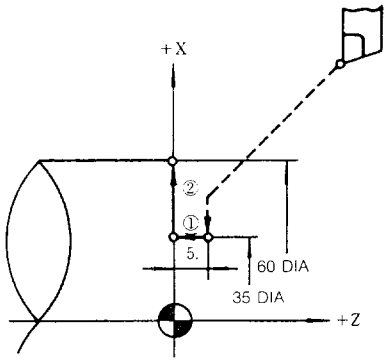
A tool is moved to the point (X, Z) on a straight line at the traverse rate designated by the F or E code in the coordinate system set by G50 moved away by (U, W) from the present point.

· F or E code must be specified in the block containing the G01 or in the previous block. If not, it causes a format error. Feedrate designated by the F or E code is the tangential feedrate.



EXAMPLE

G50 X100. Z60. ;
G00 T0202 S600 M03 ;
X35. Z5. ;
① G01 Z0 F1. ;
② X60. F0.2 ;
---- Executed by linear interpolation G01.



· Angle programming for linear interpolation †

With the control equipped with this option, linear interpolation can be commanded at specified angles.

G01 X(U)··· A··· F(E)··· ; (a)

or

G01 Z(W)··· A··· F(E)··· ; (b)

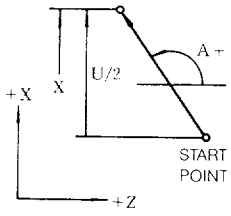
With these commands, a linear interpolation will be executed by specifying angle A in the + direction of the Z-axis and distance either in X- or Z-axis direction. The feedrate in the tangential direction is specified by the F or E code. The range of angle specifiable with address A is as follows.

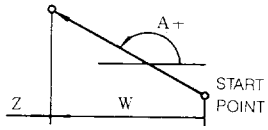
	Programmable Range of Angle A
Metric Input	0 – ±360.000°
Inch Input	

2. 8. 3 LINEAR INTERPOLATION (G 01) † (Cont'd)

Table 2. 9

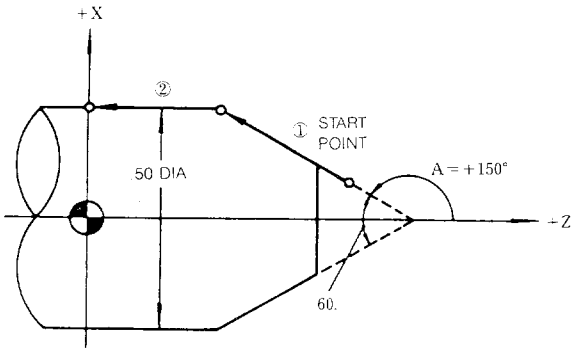
Sign	Meaning
A+	Angle counterclockwise from +Z-axis.
A−	Angle clockwise from +Z-axis.





EXAMPLE

```
.
.
① G01 X50. A+150. F0.3 ;
② G01 Z0. A-180. ;
```



2. 8. 4 CIRCULAR INTERPOLATION (G 02, G 03)

· G02(G03) X(U)··· Z(W)··· I··· K··· F(E)··· ;

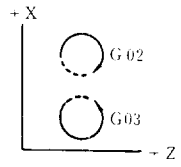
A tool is moved on the circular arc whose center is away from the present position by (I, K). The end point of the arc is (X, Z) in the coordinate system set by G50 or away from the present position by (U, W)

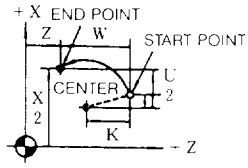
A tool moves along a circular arc at the traverse rate specified by the F or E code.

· The meanings of G02, G03 and each address are shown below.

Table 2.10

	Meaning
G 02	Circular interpolation, clockwise
G 03	Circular interpolation, counterclockwise
X (U)	End point of arc on X-axis (Diameter value)
Z (W)	End point of arc on Z-axis
I	Distance from start point of arc to arc center on X-axis (Radius value)
K	Distance from start point of arc to arc-center on Z-axis





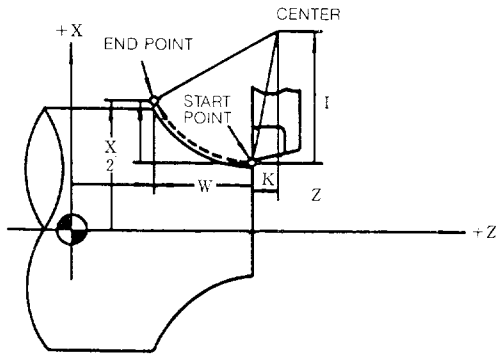


Fig. 2. 9

Circular interpolation of an arc on multiquadrant can be programmed in a single block.

EXAMPLE

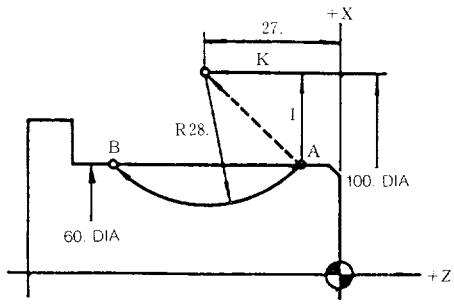


Table 2.11

Arc Center Coordinate	(10000, −2700)
I	$\frac{100-60}{2} = 20$ mm
K	$-\sqrt{28^2-20^2} = \sqrt{-384}$ $= -19.596 \rightarrow -19.60$ mm

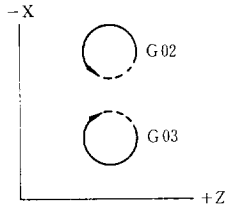
The above case can be programmed as follows.

```
.
.
G01 Z-·· F··· ;
G02 X60. Z-46.6 I20. K-19.6 F··· ;
.
.
```

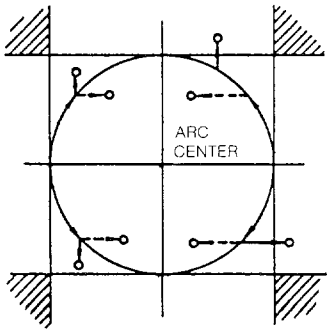
· The feedrate commanded by the F code is a tangential feedrate.

Notes :

1. The direction of the arc of G02 for Clockwise is defined as follows.
"When viewing the X, Z plane in -Y direction in the right-hand coordinate system, the tool moves clockwise from the beginning point of the arc."
Therefore, the direction of rotation in the plate (-X,Z plane) in the figure below is presented inversely.



2. When the end point of arc is not designated on the circumference specified by the radius, the alarm is not displayed and the tool path is as follows. The mark o indicates the end point of arc.



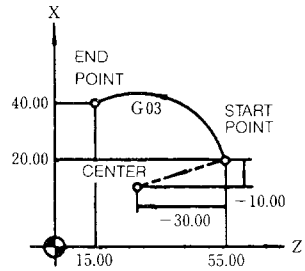
Note that if the end point is designated in the shaded area, the alarm is not displayed and the tool will continue to move endlessly.

The end point coordinate should be precisely commanded when the circular interpolation is applied to the tool nose radius compensation, or the tool may not move properly. Generally, it is recommendable to calculate up to the next digit of least input increment and count fractions over 1/2 as one and disregard the rest.

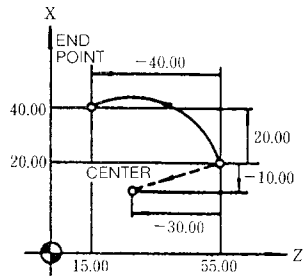
3. When the control is provided with Radius Programming for Circular Interpolation, radius value can be commanded by G02, G03 instead of G22, G23.

EXAMPLE

a. G03 X80.0 Z15.0 I-10.0 K-30.0 F150 ;



b. G03 U40.0 W-40.0 I-10.0 K-30.0 F150 ;



2.8.5 DWELL (G04)

(1) G04 U(P)··· ;

This command interrupts feed for the length of time designated by the address U or P.

(2) Dwell is programmed as an independent block.

(3) The maximum length of time which can be designated with address U or P is as follows.

Dwell time:	0.001 to 9999.99 seconds

Dwell time is not influenced by input/output increment.

EXAMPLE:

G04 U3.5 ··· 3.5-second dwell

G04 P3500 ··· 3.5-second dwell

Notes:

1. G04 is a non-modal G code.
2. The counting of dwell time is started from the instant the control enters the ERROR DETECT ON state upon completion of the move command block before G04. Therefore, with G04 U0 ;, the control advances to the next block immediately after detecting the ERROR DETECT ON state.

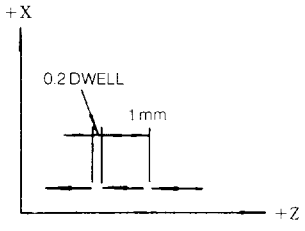
EXAMPLE

G01 W-1. F25 ; — 1 mm feed

G04 U0.2 ; — 0.2 second dwell time

G01 W-1. ; — 1 mm feed

With the above program, chip cutting feed is obtained.



2. 8. 6 TOOL OFFSET VALUE (G 10) †

With G10 command, tool offset values can be set and corrected.

(1) G10 P··· X(U)··· Z(W)··· R··· ;

With this command, tool offset values are set or corrected in part programs.

Table 2. 12

	Meaning
P	For specifying tool offset No.
X Z	For changing the tool offset value to the specified value.
U W	For adding the specified value to the original tool offset values.
R	For changing the tool radius to the specified value.

The offset values for which no address is programmed are not changed.

EXAMPLE

G10 P16 X32.5 W0.05 ;

- W0.05: 0.05 mm is added to Z-axis value
- X32.5: X-axis value is set to 32.5 m
- P16: Contents of tool offset No. 16 is changed

(2) The above format is used to make offset value tapes, and to store the values in Offset Memory at once.
The tape format is as follows.

```
Label
% ;
G10 P ... X ... Z ... R ... ;
G10 P ... X ... Z ... R ... ;

G10 P ... X ... Z ... R ... ;
%
```

2.8.7 CORNERING (G11, G12) †

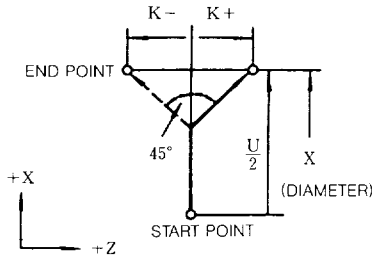
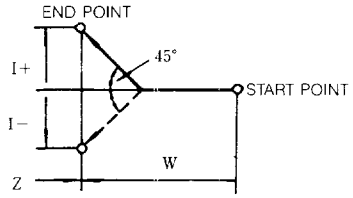
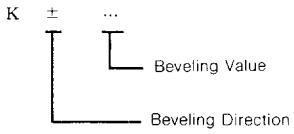
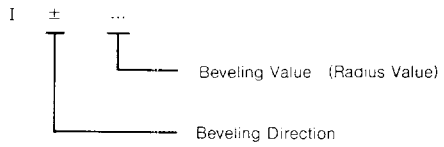
(1) Beveling (G11)

$$G11 \left\{ \begin{matrix} X(U)\cdots & K\cdots \\ Z(W)\cdots & I\cdots \end{matrix} \right\} F(E)\cdots \ ;$$

This command removes the sharp corner of workpiece. Addresses X and Z cannot be specified simultaneously in a block.

Meaning of each address is shown below.

Table 2.13

Beveling for X-axis	Beveling for Z-axis
G11 X(U)... K... F(E)... ;	G11 Z(W)... I... F(E)... ;
	
	

Beveling values (K and I) are limited within the following values.

$$|K| < |U/2| \quad , \quad |I| < |W|$$

The command exceeding the above value causes format error.

```
G00 X30. Z0 ;
① G11 Z-20. I8. F30 ;
② (G11) X80. K-7. ;
```

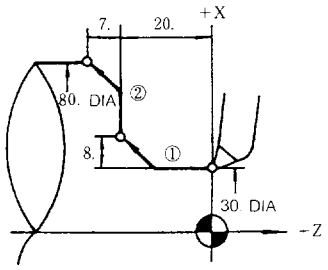


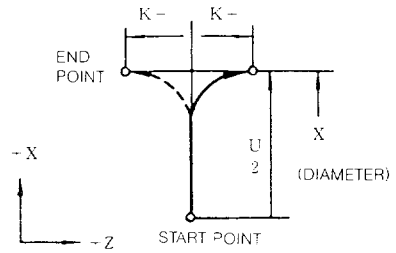
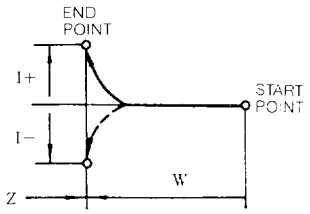
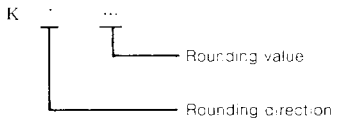
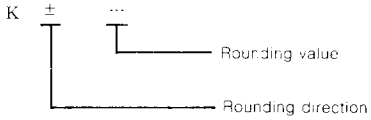
Fig. 2.10

(2) Rounding (G12)

$$G12 \left\{ \begin{matrix} X(U)\cdots & K\cdots \\ Z(W)\cdots & I\cdots \end{matrix} \right\} F(E)\cdots \ ;$$

This command performs the rounding of the corner. Addresses X and Z cannot be specified simultaneously in a block. The corner is formed as a quarter-round. Meaning of each address is shown below.

Table 2.14

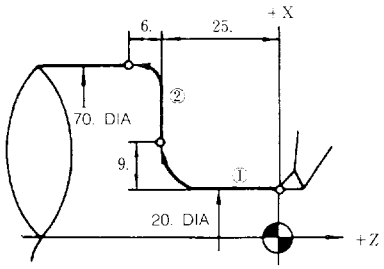
Rounding for X axis	Rounding for Z axis
G12 X(U)... K... F(E)... ;	G12 Z(W)... K... F(E)... ;
	
	

Rounding values (K and I) are limited within the following values.

$$|K| < |U/2| \quad , \quad |I| < |W|$$

The command exceeding the above value causes format error.

```
G00 X20. Z0 ;
① G12 Z-25. I9. F30 ;
② (G12) X70. K-6. F20 ;
```



Notes :

1. G11 and G12 are modal G codes in the A group. They remain effective until other G codes in the group A are commanded.
2. G11 and G12 are for one axis only. If they are commanded for both axes in the same block, they constitute a format error.

   EXAMPLE

   G12 X··· W··· K··· ; Error "050"
3. In the G10 or G12 modes, no block without I and K nor block in which I and K are 0 can be commanded. If such a block is commanded, correct tool movement can not be assured.
4. Tool radius compensation function† is effective to the blocks containing G11 or G12.
5. In the finish form commands G70 through G73 of the special canned cycle †, blocks containing G11 or G12 can be commanded.
6. G01 code can be used instead of G11 to specify identical beveling.

   G01 { X(U)··· K··· / Z(W)··· I··· } F(E)··· ;
7. G01 code can be used instead of G12 to specify rounding. However, in this case, R must be used instead of I and K.

   G01 { X(U)··· R··· / Z(W)··· R··· } F(E)··· ;

2.8.8 INCH/METRIC DESIGNATION BY G CODE (G20, G21)†

Unit of measurement (metric or inch) of input data is selectively specified by the following G codes.

G code	Input Unit
G20	Inch input
G21	Metric input

These G codes are programmed at the leading end of a block of its own. If these G codes are commanded, the units of all the following motions are changed.

a. Subsequent part programs

b. Tooloffset values

c. Part of settings and parameters

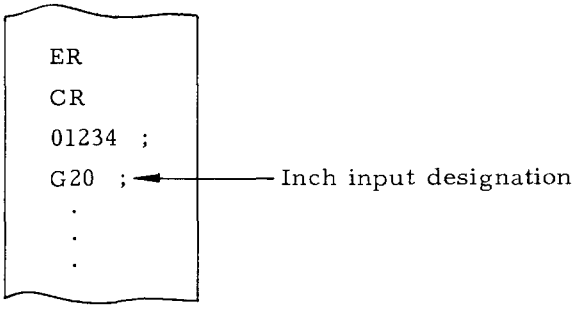
d. Part of manual movements

e. Displays

Notes :

1. When G20 or G21 is commanded, the setting of inch/metric selection is changed. Therefore, the state of G20/G21 at the time of power application depends on the setting by parameter #6001D0.

   EXAMPLE



2. When G20/G21 selection is commanded in the program, take the following procedure beforehand.

   a. Cancel work coordinate system (G50T), if used.

   b. Cancel tool position offset, and tool radius compensation (G41 - G44).

3. Take the following procedure after the command of G20/G21 selection.

   a. Program absolute zero point for all axes before move command (G50).

   b. In principle, make the display reset operation when current position display (external) is used.

4. The tool offset values are processed differently in the G20 mode and the G21 mode. G20/G21 must be commanded after modifying the tool offset values.

Stored Offset Values	Processing in G20 (Inch)	Processing in G21 (Metric)
15000	1.5000 in.	15.000 mm

2.8.9 RADIUS PROGRAMMING FOR CIRCULAR INTERPOLATION (G22, G23)†

In programming circular interpolation (G02, G03), the control requires the data of the arc-center coordinates. Normally, they are given by using the addresses I and K.

(1) In programming of G22 or G23, the control automatically calculates the arc center coordinates (I, K) from the radius value designated by the address R and performs circular interpolation.

$$\left.\begin{matrix} G22 \\ (G23) \end{matrix}\right\} X(U)\cdots\ Z(W)\cdots\ R\cdots\ F(E)\cdots\ ;$$

A tool moves along the circular arc whose center is radius R away from the present position. The end point of arc is at coordinates (X, Z) set by G50 or is away from the present position by (U, W). Tool moves along the circular arc at feedrate designated by F code.

(2) The meanings of G22, G23 and each address are shown below.

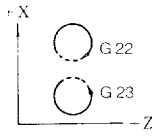
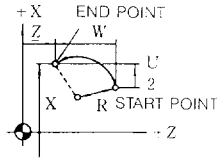
(3) Designation of radius value R

Radius value R is commanded by incremental value with a sign of radius programming.

In this case,
When radius value R > 0, an arc, describing less than 180°, and when R < 0, an arc describing more than 180° are specified.

2.8.9 RADIUS PROGRAMMING FOR CIRCULAR INTERPOLATION (G22, G23) † (Cont'd))

Table 2.15

	Meaning	
G22	Circular interpolation by radius for CW	
G23	Circular interpolation by radius for CCW	
X(U)	The X-coordinate of the end of the arc (Diameter value)	
Z(W)	The Z-coordinate of the end of the arc	
R	Distance from the start point of arc to arc center (Incremental value with sign)	

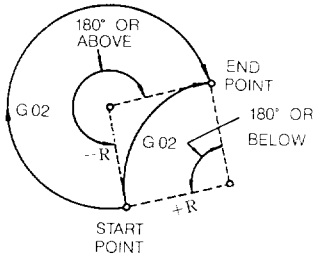


Fig. 2.11

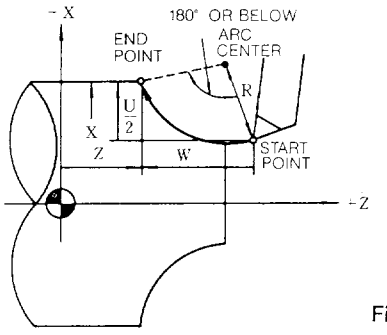
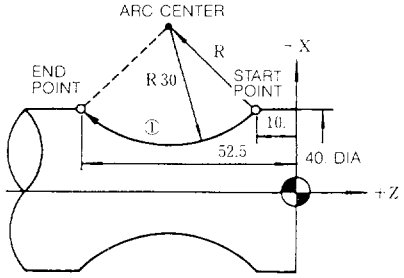


Fig. 2.12

EXAMPLE

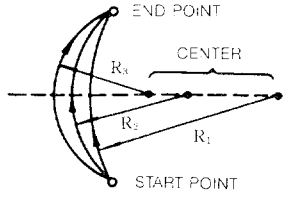
G01 X40. Z-10. F20 ;

① G02 (X40.) Z-52.5 R30. (F20) ;



Notes :

1. G22 and G23 codes are modal. They are kept until other G code of 01 group is commanded.
2. In the G22 or G23 mode, the block in which R is not contained or R is designated as zero should not be commanded. Radius cannot be designated by I and K.
3. When R is varied with both start and end points fixed, the tool will move along the following circular arc.



Therefore, in the following case, the arc center does not exist which causes data error (alarms "031," "034").

$$R < \frac{\text{(Distance between start point and end point)}}{2}$$

4. Tool nose radius compensation is effective for the block containing G22 or G23.
5. The block containing G22 or 23 can be designated in finishing shape commands of special canned cycles (G70 to G73).
6. When the control is provided with radius programming option, circular interpolation by radius (R) programming can be made by G02, G03 instead of G22, G23.

2.8.10 REFERENCE POINT CHECK (G27)

(1) G27 X(U)··· Z(W)··· ;

With this command, the tool is positioned to the absolute coordinate point (X, Z) or incremental coordinate point (U, W) by moving along the two axes simultaneously, and then, the position is checked for conformance with the reference point. For the axis for which no command is given, positioning and checking are not executed.

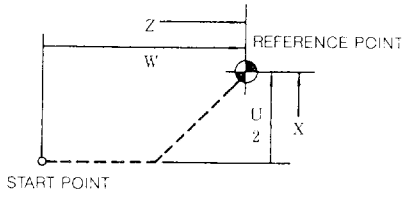


Fig. 2.13

(2) If the position is the reference point, the return-to-reference lamp lights. The position is the reference point in all the axial directions specified, the automatic operation is continued further. If the position is not the reference point even along one axis, this constitutes the return-position-error, and the automatic operation is interrupted. (Cycle start lamp goes off.)

Notes :

1. The reference point is an fixed point on the machine tool to which the tool can return by the motion under the control of the automatic reference point return or G28 automatic reference point return function. See 6.2.1, "Automatic Reference Point Return."
2. If G27 is commanded in the tool position offset mode, the tool returns to the position displaced from the reference point by the tool offset value. Positioning cannot be made at the reference point. Before commanding G27, cancel the tool offset mode.

   EXAMPLE

   a. Cancelling tool offset in the block preceding G27.

      T □ □ 00 ;

      G27 U··· W··· ;

   b. Cancelling in the block containing G27.

      G27 U··· W··· T □ □ 00 ;

3. The mirror image function is effective with the motion commanded by G27. To avoid the return position error, command G27 in the G69 mode (Opposite tool post mirror image off).

2.8.11 AUTOMATIC RETURN TO REFERENCE POINT (G28)

(1) G28 X(U)··· Z(W)··· ;

With this command, the tool can be brought back to the reference point automatically after passing through an interim point. In other words, the tool positions to the commanded absolute coordinate position (X, Z) or incremental position (U, W) by moving simultaneously along the two axes, and then automatically returns to the reference point by the reference point return function. The specified point (X, Z) or (U, W) is known as "INTERIM POSITIONING POINT," or "INTERIM POINT."

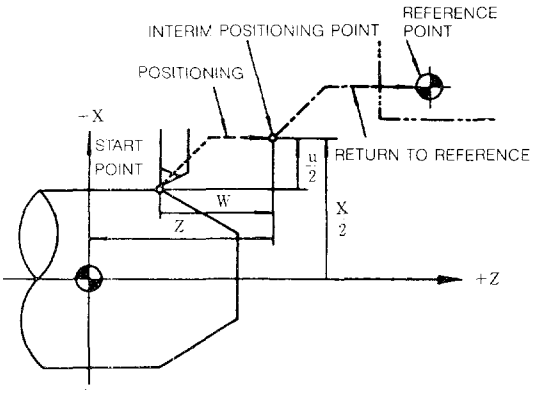


Fig. 2.14

The tool does not move along the axis for which instruction is omitted.

(2) When the return to reference motion is completed, Reference Point Return lamp for the returned axis lights. When the tool returns to the reference point in both axes, the automatic operation is resumed.

(3) The series of RETURN TO REFERENCE motions are as follows. With initial power application, the return motions to the reference point are as shown below in the low-speed mode as shown below.

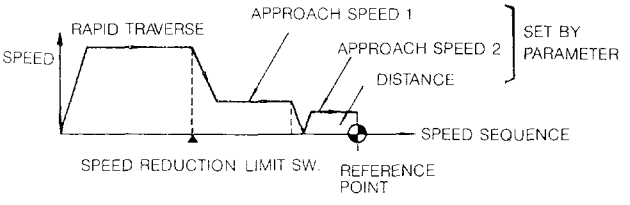


Fig. 2.15

Thereafter, the RETURN TO REFERENCE POINT motion is in rapid traverse as under the command of G00.

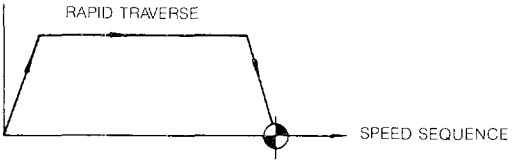


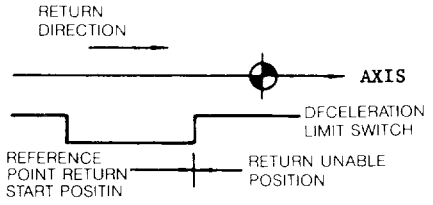
Fig. 2.16

(4) However, when low traverse speed is specified by parameter #6010D5 set to 1, the same low motion speed as in the 1st time is obtained.

2.8.11 AUTOMATIC RETURN TO REFERENCE POINT (G28) (Cont'd)

Notes :

1. For parameter setting and other details of the low traverse speed return motion shown in Fig. 2.8.11.2, refer to 5.2.1 Manual Return to Reference Point.
2. The starting point for RETURN TO REFERENCE POINT motion must be in the area shown in Fig. 2.8.11.4. Fig. 2.8.11.3 can be started from any position.



3. Before writing G28 in the program, cancel the tool position offset as shown below.

   a. Cancelling in the preceding block

   T □ □ 00 ;

   G28 X··· Z··· ;

   b. Cancelling in the block containg G28

   G28 X··· Z··· T □ □ 00 ;

4. When G28 is written with the tool position offset or tool radius compensation on, the offset or compensation is automatically cancelled.

2.8.12 RETURN FROM REFERENCE POINT (G29)

(1) With this function, the tool is positioned to a specified point via the interim point, after it has been once returned to the reference zero point by the AUTOMATIC RETURN TO REFERENCE ZERO COMMAND (G28).

G28 X··· Z··· ; Point A → B → C (reference zero point)

Point B

G29 X··· Z··· ; Point C → B → D

Point D

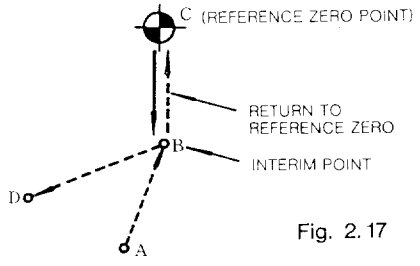


Fig. 2.17

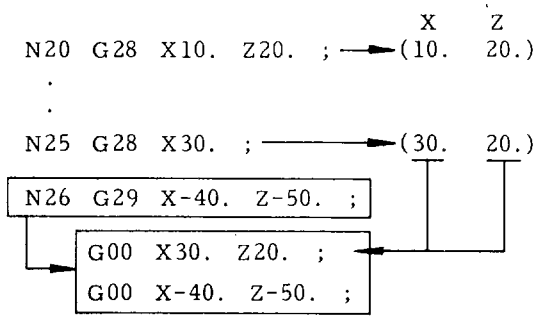
(2) When G29 is used, consideration on the distance between points B and C is unnecessary in programming. Especially when incremental instructions are used, this function is useful to return the tool to the original coordinate system after returning to the reference zero.

(3) Motions C → B and B → D are made simultaneously along the two axes in rapid traverse. However, the tool will not move in the direction for which instruction is omitted.

(4) Where G28 is programmed several times, the point B created by the latest G28 instruction is effective for the motion by G29.

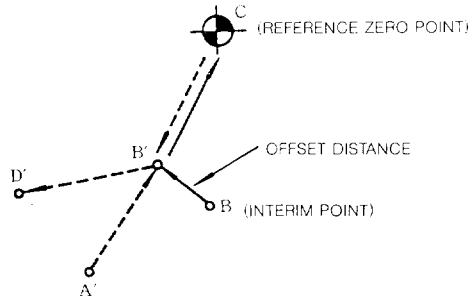
EXAMPLE (absolute input)

Coordinates of interim point is equivalent to these two blocks.



Notes :

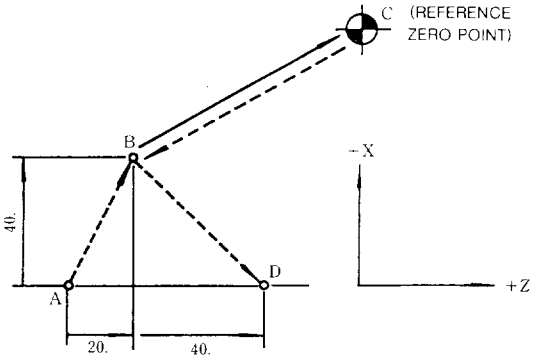
1. Commanding G29 without the execution of G28 after turning on the control constitutes an error "059."
2. In principle, cancel tool offset before programming G28 or G29. If they are programmed while offset is effective, the interim point B will be offset, and the tool will pass point B'.



3. Commanding G29 in the TOOL RADIUS COMPENSATION mode (G40 - G44) or in CANNED CYCLES (G70 - G76, G90, G92 and G94) constitutes an error.

EXAMPLE

```
N50 T0300 ;
N51 G28 U80. W20. ;
N52 T0400 ;
N53 G29 U-80. W40. ;
```



2.8.13 2ND REFERENCE POINT RETURN (G30)†

(1) G30 X(U)··· Z(W)··· ;

With this command, the tool first moves to an interim positioning point (X, Z) or (U, W) in two axial directions simultaneously, and then, moves to the 2nd reference point. The tool does not move along the axis for which no coordinate position is specified.

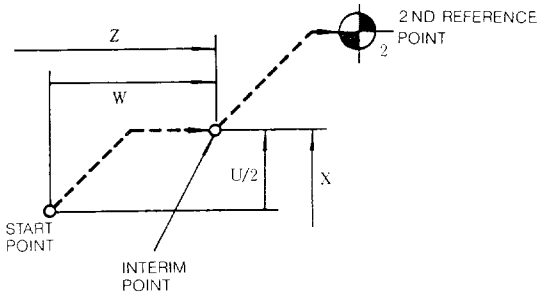
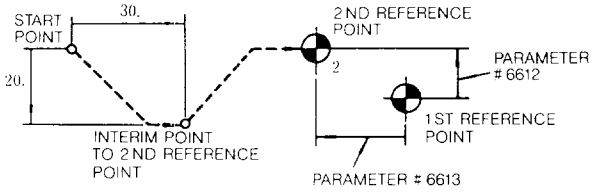


Fig. 2.18

(2) The 2nd reference point is specified in advance in terms of the distance from the 1st reference point commanded by G28, by parameters #6612 and #6613.

EXAMPLE

G30 U-40. W30 ;



Notes :

1. Before commanding G30, after the energization of the control, G28 or MANUAL RETURN TO REFERENCE POINT must be executed.
2. For the 2nd REFERENCE POINT RETURN motion, there is no area from where returning is impossible, and the tool can be return from any position.
3. The same notes 3. and 4. of (5) for 2.8.11 Automatic Return to Reference Point apply to G30 command.
4. When G29 is commanded after G30, the tool moves via the interim point specified by G30 to the position specified by G29. However, the interim point is renewed only in the axis specified by G30.

2.8.14 SKIP FUNCTION (G31)†

(1) G31 X(U)··· Z(W)··· (F(E)···) ;

With this command, a special linear interpolation is executed. During the interpolation movement under this instruction, the tool interrupts the interpolation motion immediately, and proceeds to the next block, when a skip signal is received.

(2) The motion after the receipt of a skip signal varies with the instruction of the next block.

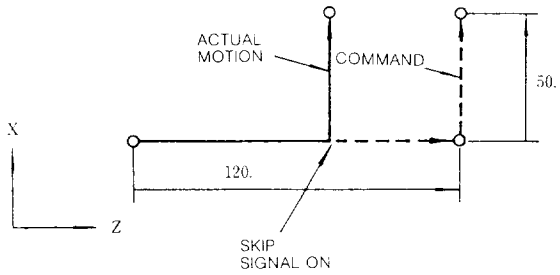
a. When the next block is programmed in incremental values:

The tool moves incrementally in accordance with the next block from the point where the interpolation is interrupted.

2.8.14 SKIP FUNCTION (G31) † (Cont'd)

EXAMPLE

```
G31 W120. ;
G01 U100. ;
```

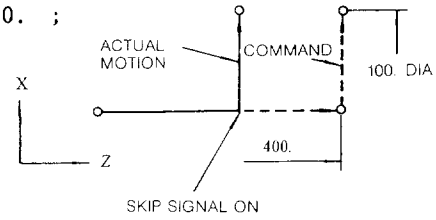


b. When the next block is programmed in absolute values only for one axis:

The tool moves to the specified coordinate position in the specified axis. It remains at the position where the skip signal is received, if axis is not specified.

EXAMPLE

```
G31 Z400. ;
G01 X100. ;
```

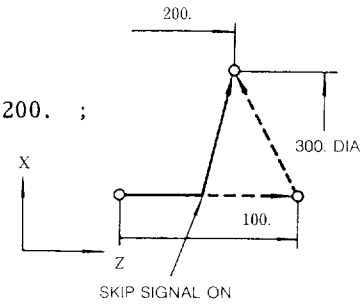


c. When the next block is programmed in absolute values along two axes:

The tool moves to the commanded position from the point at which a skip signal is received.

EXAMPLE

```
G31 W100. ;
G01 X300. Z200. ;
```



(3) G31 is a non-modal G code.

When no skip signal is received during the execution of the block containing G31, the tool stops at the end of the block, and alarm "087" is displayed.

(4) The feedrate for blocks containing G31 are set in the following two methods, selectively specified by parameter #6019D4.

a. Designation by F in the same way as with normal programs.

b. Presetting of feedrates by parameter #6232.

(5) When a skip signal is received, the coordinate values at that moment are automatically stored as parameter data.

#6568 for storing X coordinate value

#6569 for storing Z coordinate value

These data can be used as system variables in user macros.

Notes :

1. When parameter #6004D0 is set to 1, the program is advanced to the next block automatically even when no skip signal is received during the execution of the block of G31.
2. Before programming G31, be sure to program G40 for cancelling TOOL RADIUS COMPENSATION. Failure to do this initiates alarm "024."
3. Commanding G31 with skip signal ON advances the program to the next block without executing the current block.

2.8.15 THREAD CUTTING, CONTINUOUS THREAD CUTTING (G32)

This function is for cutting straight threads, taper threads, scrolls and for continuous threading.

(1) G32 X(U)··· Z(W)··· F(E)··· ;

With this command, the tool cuts threads up to the point (X, Z) specified in absolute coordinates or (U, W) specified in incremental coordinate values, at a lead designated by F or E code.

(2) The range of leads to be specified by F and E codes is as follows.

Table 2. 16

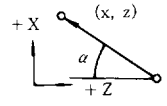
		Format	Programmable Range for Threadcutting
Metric Output	Metric Input	F32	F0.01–F500.00 mm
		E34	E0.0001–E500.0000 mm
	Inch Input	F24	F0.0001–F19.6850 in.
		E26	E0.000004–E19.685000 in.
Inch Output	Metric Input	F32	F0.01–F1270.00 mm
		E34	E0.0003–E1270.0000 mm
	Inch Input	F24	F0.001–F50.0000 in.
		E26	E0.000010–F50.000000 in.

F code is for normal thread cutting.

E code is for precise thread cutting.

(3) The direction of lead specified by F and E codes is shown below.

Direction of Lead

Limitation of Taper Angle		Direction of Lead
	$\alpha \leqq 45°$	Lead in the direction of Z-axis
	$\alpha > 45°$	Lead in the direction of X-axis

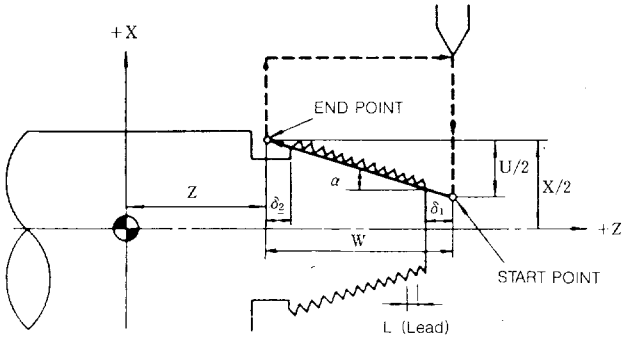


Fig. 2. 19

Feedrates are limited by spindle-speed S as follows.

Metric Output	F(E)×S ≦ 24,000 mm/min
Inch Output	F(E)×S ≦ 2,400 in./min

The upper limit of X-axis speed component is half the above.

(4) Command format of threadcutting is shown below.

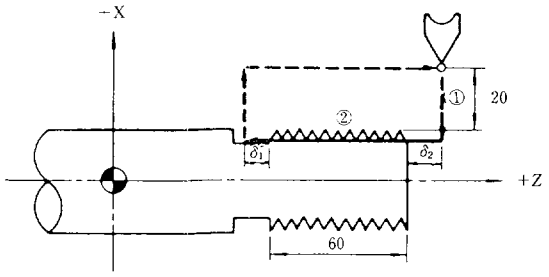
Table 2. 17

Type		Command format
Straight Thread	Normal	G32 Z(W)... F... ;
	Precise	G32 Z(W)... E... ;
Taper Thread	Normal	G32 X(U)... Z(W)... F... ;
	Precise	G32 X(U)... Z(W)... F... ;
Scroll Thread	Normal	G32 X(U)... F... ;
	Precise	G32 X(U)... E... ;

EXAMPLE: Straight Thread

Thread lead L = 5.0 mm
δ_1 = 5.0 mm
δ_2 = 3.0 mm
Cutting depth = 1.0 mm

```
① G00 U-42. ;
② G32 W-68. F5.0 ;
   G00 U42. ;
       W 68. ;
       U-44. ;
   G32 W-68. ;
   G00 U44. ;
```

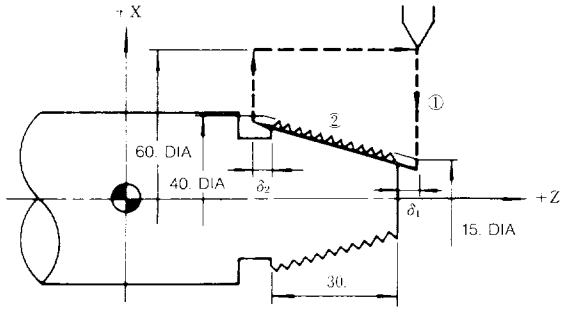


2.8.15 THREAD CUTTING, COUTINUOUS THREAD CUTTING (G32) (Cont'd)

EXAMPLE: Taper Thread

Thread lead L = 4.0 mm
δ_1 = 3.0 mm
δ_2 = 2.0 mm
Cutting depth = 1.0 mm

```
G00  X13.  ;
G32  X38.  W-35.  F4.0  ;
G00  X60.  ;
     W35.  ;
     X-11.  ;
G32  X36.  W-35.  ;
G00  X60.  ;
     .
     .
     .
```



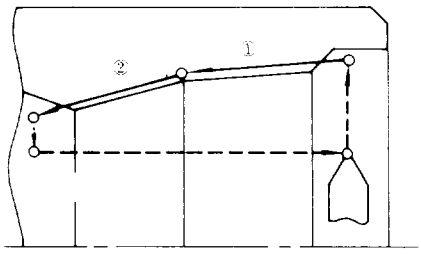
(5) Continuous thread cutting

① G32 X(U)··· Z(W)··· F(E)··· ;

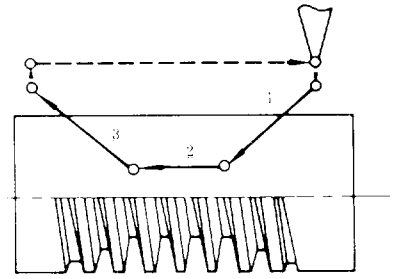
② (G32) X(U)··· Z(W)··· ;

③ (G32) X(U)··· Z(W)··· ;

This command executes thread cutting.



(a) Pipe Joint

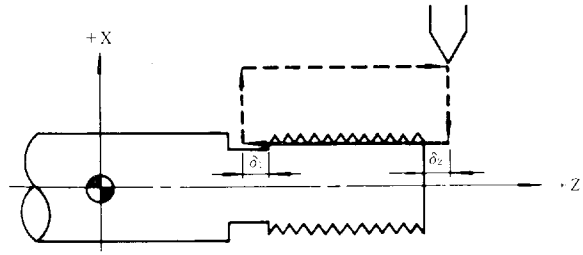


(b) Worm Screw

Since the stop time between thread cutting blocks is approximately zero, smooth, continuous thread cutting is possible. If thread lead specification is changed midway, the thread becomes irregular near the boundary of blocks.

Notes :

1. Allowances δ_1 and δ_2 are required for thread cutting because lead error occurs near the starting and end points.



2. If spindle speed is not constant during thread cutting, the leads become incorrect due to the servo lag.
3. Threading up for thread is not effective at G32. If necessary, G92 (or G76†) should be commanded.
4. The following operation is disregarded during thread cutting including G32

   · Feedrate Override ··· Regarded as 100%
   · Feed Hold Operation
   · Single Block Operation

5. The G32 command should not be commanded in G98 mode.
6. In Dry Run mode, the tool moves at Jog feedrate.
7. Do not command M code for continuous thread cutting. If commanded, the commanded block stops as feedhold and will not permit continuous thread cutting.

(6) Allowance for lead error(δ_1, δ_2)

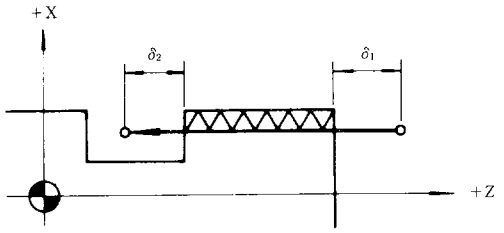


Fig. 2.20

δ_1 and δ_2 are obtained approximately from the following equation.

Table 2.18

	Equation	Meanings
δ_1	$\delta_1 > \frac{L.S}{60.K}(\ln\frac{1}{a} - 1)$	L(mm): Lead of thread S(r/min): Spindle speed K: Constant (Normal value: 30) a (−): Accuracy of thread $= \frac{\Delta L}{L}$ ...Lead error ln: Natural logarithm (log e)
δ_2	$\delta_2 > \frac{L.S}{60.K}$	

a	1/50	1/100	1/150	1/200	1/250	1/300
$(\ln\frac{1}{a} - 1)$	2.91	3.61	4.01	4.29	4.52	4.70

EXAMPLE

Lead of thread L = 3.0 mm
Spindle speed S = 500 r/min
Thread cutting a = 1/100

$$\delta_1 > \frac{L.S}{60.K}(\ln\frac{1}{a} - 1)$$

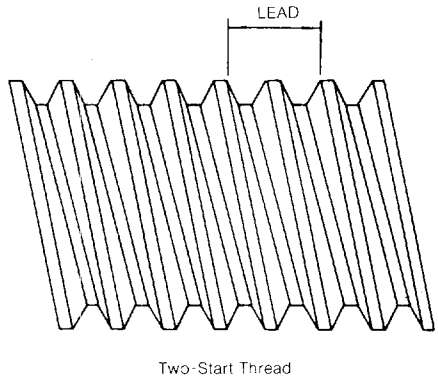
$$= \frac{3.0 \times 500}{60 \cdot K} \times 3.61 = 3.0 \text{ mm}$$

$$\delta_2 > \frac{L \cdot S}{60 \cdot K} = \frac{3.0 \times 500}{60 \cdot K} = 0.83 \text{ mm}$$

2.8.16 MULTI-START THREAD CUTTING (G32)†

With this function, multi-start threads containing two or more threads per lead can be machined without shifting the starting point. In thread cutting, the tool feed is started in phase with a start point pulse (1 pulse/revolution) generated by a pulse generator installed on the spindle to control the starting point of thread always at the same position around the workpiece circumference.

With the multi-start thread cutting function, after cutting a thread by controlling the starting point by the starting point pulse, another thread is cut by starting the cutting feed at an angular position of the spindle which is displaced from the starting pulse position by a preset angle.



Two-Start Thread

Fig. 2.21

(1) G32 X(U)··· Z(W)··· F(E)··· B··· ;

With this command, the tool cuts a thread starting at an angular position which is displaced from the position corresponding to the starting pulse by an angle specified by B, to X(U) or Z(W) point, at a lead specified by F or E code.

(2) The data specified by address B in the multi-start thread cutting function is as follows.

Least input increment: 0.001 deg.

Programmable range: $0 \leqq B \leqq 360.000$

When decimal point input is used, B1.=1deg. The B code is non-modal, and is effective only in the programmed block.

2. 8. 16 MULTI-START THREAD CUTTING (G 32) † (Cont'd)

(3) Number of starts and B code

In principle, the thread starting points on the workpiece circumference should divide the circumference into equal portions.

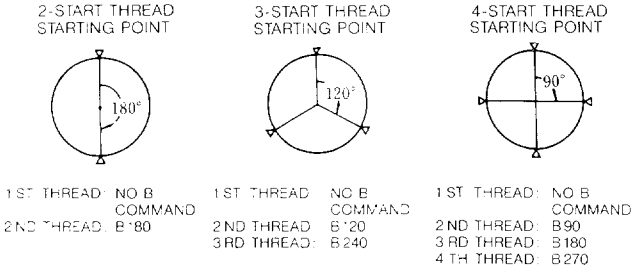


Fig. 2. 22

Notes :

1. Since the angular position detection pulses (4096 pulses/rev.) generated from the spindle pulse generator is used to define the angular position of the spindle with respect to the starting point as controlled by the B command, the least detectable increment is 360°/4096 pulses ≒ 0.0879°/pulse. From the position commanded by B codes, an error up to ±1 pulse may occur.
2. The angular position from the starting pulse can be specified in both forward and reverse directions by B0 - B360.commands.
3. When B command is made to specify angles outside the permissible range (0 - 360.000), alarm "062" is displayed.
   If multi-start thread cutting B is specified, continuous thread cutting cannot be executed.

EXAMPLE

G32 W··· B90

G32 W··· Continuous thread cutting cannot be performed because of feedhold at this block.

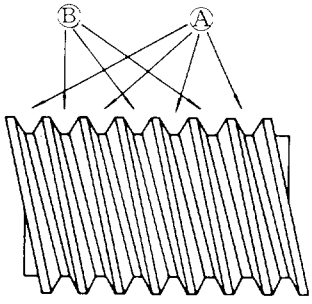
EXAMPLE: Two-start Thread

```
G00  U···  ;
G32  W···  F···  ;
G00  U···  ;
W···  ;
U···  ;
G32  W···  ;
  .
  .
  .
```

Threading of part Ⓐ

```
G00  U···  ;
G32  W···  B180.  ;
G00  U···  ;
W···  ;
U···  ;
G32  W···  B180.  ;
  .
  .
  .
```

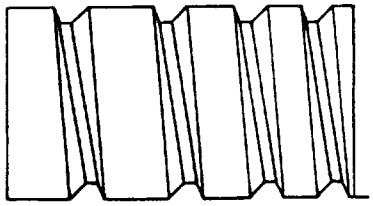
Ⓑ



2.8.17 VARIABLE LEAD THREAD CUTTING†

(1) G34 X(U)··· Z(W)··· K··· F(E)··· ;

With this command, variable lead screws are controlled with the increase or decrease of lead per revolution specified by address K.



VARIABLE LEAD SCREW

Fig. 2.23 Variable Lead Threadcutting

(2) The range of K programmable for variable lead screws is as follows.

Least input increment:

0.0001 mm/rev.	0.000001 in./rev.
(Metric Input)	(Inch Input)

Programmable range:

a. The highest feedrate is within the maximum programmable feedrate range (500 mm/rev (metric) or 50 in/rev. (inch))

b. The total displacement resulting from changes in lead is within the following.

4194.303 mm (metric output)

419.4303 in. (inch output)

c. Feedrate change corresponding to lead variation must not exceed 5,400 mm/min (metric output) or 540 in./mm (inch output).

d. Lead value should not be minus value.

Notes :

1. When variable lead threads are cut by continuous block programs, command pulses are interrupted at block junctions.
2. If K commands exceed the permissible range, alarm "060" will be displayed.
3. When G34 command is executed in the Dry Run mode, the tool moves only at the speed specified by the manual continuous feedrate command, if parameter SCRDRN (#6019 BIT5) is set to 1.
4. When parameter "1OIN" (#6006D5) is set to 1, least increment for K commands is 0.001 mm/rev or 0.0001 in./rev.
5. Commanding address B in G34 block causes alarm "060."

(3) Confirmation calculation for K command of variable lead thread cutting.

a. K command is restricted in the following conditions.

(i) Feedrate at end point must not exceed programmable range.
500 mm/rev (metric output) or 50 in rev (inch output)

(ii) Feedrate at end point must not be minus value.

(iii) Accumulated value of movement due to lead variation must not exceed 4194.303 mm (metric output) or 419.4303 in (inch output).

(iv) Feedrate change corresponding to lead variation must not exceed 5,400 mm/min (metric output) or 540 in/mm (inch output

b. The control checks the restriction described above using the following equation.

F: Fixed lead command (mm/rev or in./rev)

K: Variable lead command (mm/rev or in./rev)

W: Distance between start and end points on Z-axis (mm or inch).
For facing screw, distance is specified as U on X-axis.

S: Spindle speed (rev/min)

N: Spindle speed for movement between start and end points (rev)

$$N = \frac{-(F + \frac{K}{2}) + \sqrt{(F + \frac{K}{2})2 + 2 \cdot K \cdot W}}{K}$$

(i) Equation for limit in a. (i)

$F + \frac{K}{2} + KN \leq$ 500.000 mm/rev or 50.0000 in/rev

(ii) Equation for limit in a. (ii)

$(F + \frac{K}{2})^2 + 2\,KW > 0$

(iii) Equation for limit in a. (iii)

$\frac{1}{2}KN^2 \leq$ 4194.303 mm or 419,4303 in

(iv) Equation for limit in a. (iv)

$\frac{S}{60} \cdot K \cdot N \leq$ 5,400 mm/min or 540 in/min

2.8.18 STORED STROKE LIMIT (G36-G39)†

This function is for preventing the entry of the tool into the prohibited area, in both automatic operation mode and manual operation mode, to secure safer operation. Three types of prohibited areas can be set up as shown below.

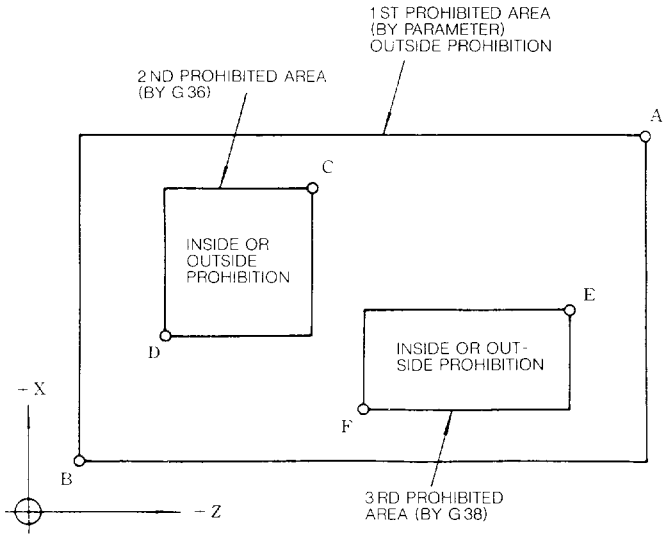


Fig. 2.25

(1) Setting 1st prohibited area

a. Set + side boundary A and - side boundary B by parameter.

b. The area outside the range between points A B is prohibited.

c. This area can not be specified by part programs.

(2) Setting 2nd prohibited area

a. Set + side boundary C and - side boundary D by the setting function.

b. Selectively designate the inside or the outside of the area between points C and E by parameter #6007D0.

# 6007 D_0	Meaning
0	Inside prohibition
1	Outside prohibition

c. In addition to the method by the setting function, the following instruction can be used to set a prohibited area, and to turn on the area check function also.

G36 U··· W··· I··· K··· ;

Point C coordinate — Point D coordinate

The area check function is turned off by a single code block G37 ;.

(3) Setting 3rd prohibited area

a. Set up + side boundary point E and - side boundary point F by the setting function.

b. Selectively designate the inside or the outside of the area between points E and F by parameter #6007D1.

#6007 D_1	Meaning
0	Inside prohibition
1	Outside prohibition

c. In addition to the method by the setting function, the following instruction can be used to set a prohibited area, and to turn on the area check function also.

G38 U··· W··· I··· K··· ;

Point E coordinate — Point F coordinate

The area check function is turned off by a single code block G39 ;.

Table 2.19 Parameters and Settings for Setting Prohibited Area

		X axis	Z axis	Division
1st Prohibited Area	Point A	#6600	#6601	Parameter
	Point B	#6606	#6607	
2nd Prohibited Area	Point C	#6500	#6501	Setting
	Point D	#6502	#6503	
3rd Prohibited Area	Point E	#6504	#6505	
	Point F	#6506	#6507	

(4) Coordinate system for area setting

The above Points A through G are all set in absolute values on the machine coordinate system (MACHINE). That is, the position is written as the distance from the (1st) reference point (1 = least output (move) increment). Therefore, this function will not become effective unless the manual or automatic RETURN TO REFERENCE ZERO is executed once after the energization of the control.

(5) Effective-ineffective selection of prohibited area

With the following setting, the 2nd and the 3rd prohibited areas are selectively made effective and ineffective.

Setting		Meaning
#6001 D_1	0	2nd prohibited area check OFF
	1	2nd prohibited area check ON
#6001 D_2	0	3rd prohibited area check OFF
	1	3rd prohibited area check ON

When G36 through G39 are commanded, these setting data are automatically rewritten. Therefore, the ON or OFF state ultimately specified by G code commands or setting function becomes effective. The 1st prohibited area is always in the CHECK ON mode.

(6) Starting area check

When the tool is returned to the reference point once manually or automatically after the energization of the control, the area check function is started immediately. Therefore, if the reference point is in the prohibited area, immediately, STROKE LIMIT ERROR will be caused. In this case, turn off the area check function, and change the data.

(7) Stored stroke limit error

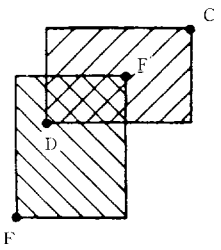
When the tool enters the prohibited area, it stops just inside the boundary line, and the control enters STORED STROKE LIMIT ERROR state. In this case, the tool can only be moved manually in the return direction.

(8) Displaying remaining distance

With this function, the distance between the current tool position and the boundary of the prohibited area in the X and Z directions are displayed on the CRT. Refer to 4.3.4.6. "Stored Stroke Limit ."

Notes :

1. The points on the boundary line in both axes are included in the prohibited area.
2. Two prohibited areas can be set with partial overlapping.
3. In the MACHINE LOCK ON mode, AREA CHECK function is not effective.



EXAMPLE

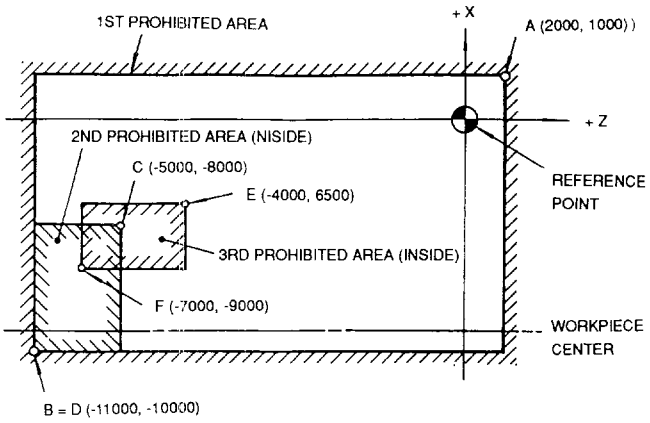


Table 2. 20

	Parameter/ Setting	Contents	
Inside/Outside	#6007 D_0	0	
	#6007 D_1	0	
Second Area	#6500	−5000	C
	#6501	−8000	
	#6502	−11000	D
	#6503	−10000	
Third Area	#6504	−6000	E
	#6506	−6500	
	#6506	−7000	F
	#6507	−9000	
First Area	#6600	2000	A
	#6601	1000	
	#6606	−11000	B
	#6607	−10000	

2.8.19 TOOL NOSE RADIUS COMPENSATION (G 40 THROUGH G 44)

Because of a nose radius of lathe tools, there is a deviation between the desired curve and the actual curve produced. Therefore, tool offsets are not enough for taper and circular cuttings. The tool nose radius compensation option resolves the problem of nose radius. See Fig. 2.26.

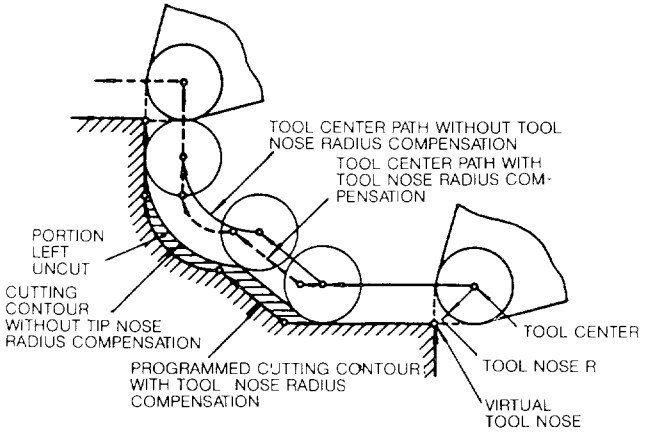


Fig. 2.26

(1) Tool nose radius values

a. Radius value storage

Tool nose radius value must be written in the storage before the tool nose radius compensation is commanded. Number of pairs that can be written in the storage depends upon the machine.

T 3-digit: 9 sets
T 4-digit: 16 or 50 sets

Refer to 2.6.2, " Tool Offset Memory†."

b. Range of tool nose radius values

Radius value can be set within the following range.

Metric	Inch
0 ±99.999	0 ±9.9999

c. Setting of tool nose radius values

Radius value of tool nose must be set without signs.

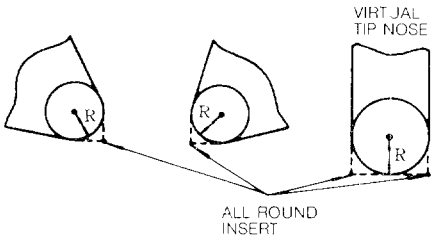


Fig. 2.27

For the writing of radius values for tool nose radius compensation, refer to Fig. 4.3.5 Displaying and Writing Tool Offset Data. The address character is R.

(2) T code designation

a. The T code for tool nose radius compensation must be programmed with sign (+ or -).

T ± □□ □□ (In the case of T 4-digit)

- Offset number for tool position offset and tool nose radius compensation
- Tool number
- Direction of tool nose radius compensation

"+" ··· Right side viewed in the direction of tool travel

"−" ··· Left side viewed in the direction of tool travel

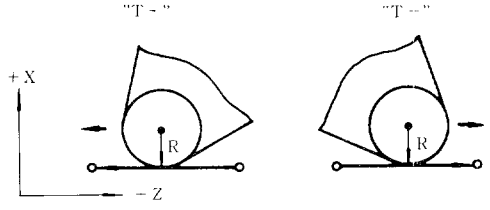


Fig. 2.28

b. When a tool is used for turning and for facing, as the direction of tool motion changes, the correct direction of compensation should be programmed with sign of T code.

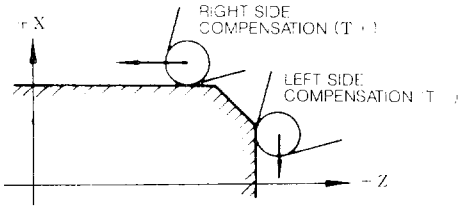


Fig. 2.29

The direction of compensation is changed from + to - or - to + during program execution. G40 or T□□ 00 command should not be necessarily programmed to cancel the tool nose radius compensation.

(3) G code designation (G40 to G44)

a. G code of tool nose radius compensation (G41 to G44)

One of G41, G42, G43 and G44 and T code should be programmed before the execution of tool nose radius compensation. These four G codes specify the relationship between the virtual tool nose and the tool center.

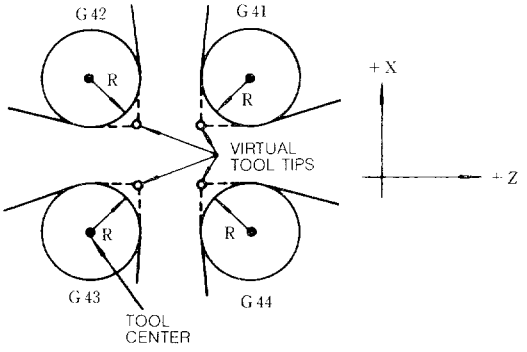


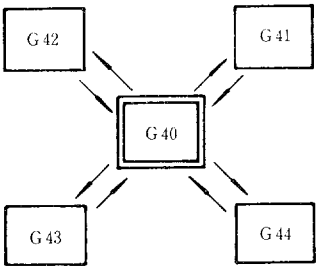
Fig. 2.30

Regardless of the mode of tool nose radius compensation, the current position of the virtual tool nose is displayed by depressing the POS pushbutton.

b. Issue G40 to cancel the tool nose radius compensation.

c. Cautions in programming G code

(i) Since G40 to G44 are modal G codes of 06 group, they are retained until the other G code is commanded. Before switching one of G41, G42, G43 and G44 to another, G40 must be intermediated to cancel the compensation.



(ii) When the power supply is turned on, G40 is in effect.

(iii) When the RESET button is depressed, G code of 06 group are cancelled and G40 becomes effective.

(4) Tool motion on the tool nose radius compensation

a. Fig. 2.31 shows the outline of the tool motion.

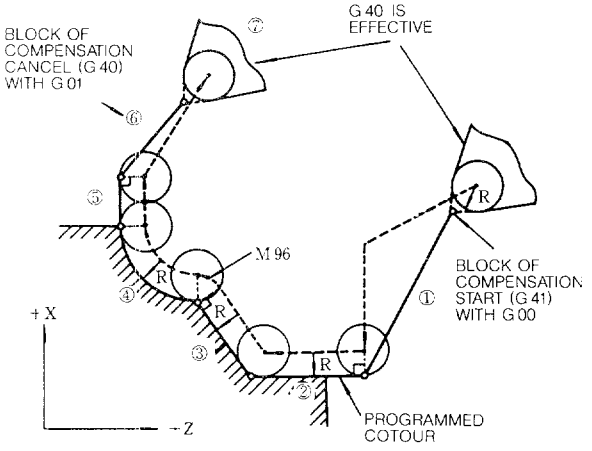


Fig. 2.31

(i) When the compensation is cancelled, the programmed contour meets with the path of virtual tool tip (⓪ and ⑦).

(ii) In compensation mode, the tool center path is deviated by radius from the programmed contour. Therefore, the path of virtual tool nose does not meet with programmed contour. But the current position displayed by depressing POS key is the position of virtual tool tip (② to ⑤).

(iii) The connection between two blocks in compensation mode is provided by the intersection of tool center paths (M97) and by the circular arc (M96). In the above diagram, blocks 3 and 4 are connected by a circular arc.

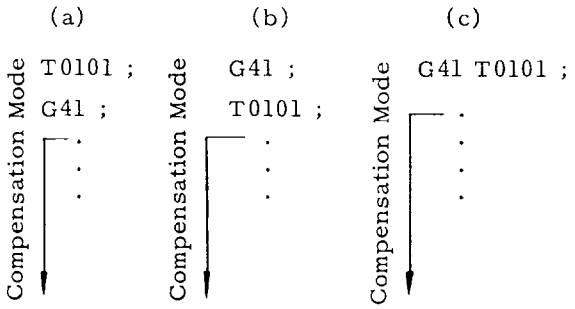
(iv) Block ① for compensation start and block ⑥ for compensation cancel perform the connection of compensation mode and compensation cancel mode. Program should be made carefully for these blocks.

b. Relationship between tool nose radius compensation and tool position offset

Tool nose radius compensation apply to the programmed contour which has been offset by the tool position offset function.

(5) How to enter compensation mode

a. Compensation mode is set when both the tool offset number by T code and G41 (or G42 through G44) are instructed. When this mode is set, tool nose radius compensation is started. More precisely, compensation mode is entered when the AND condition between T code and G code is established. Hence, the order in which these codes are specified does not affect the operation.



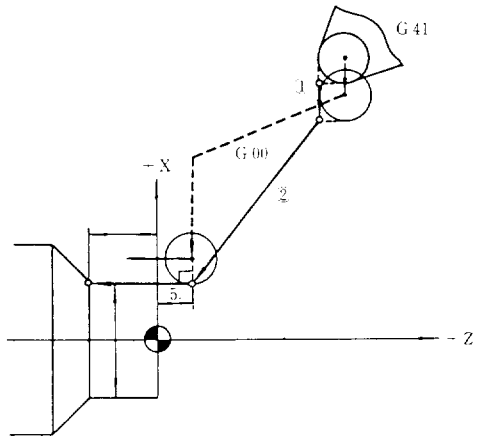
b. At the start of compensation, the tool center is offset onto the normal of the origin of the block G41 to G44 which entered the compensation mode first or of the block immediately after T code. The offset is made to the right of tool advancing direction when T+ is specified and to the left when T- is specified.

Sample Program (A):

```
        (a)
①  G00  T+0101  ;
②  G00  G41  X30.  Z5.  ;
③  G01  Z-18.  F0.25  ;
```

Movement at Start of Compensation (for G00)

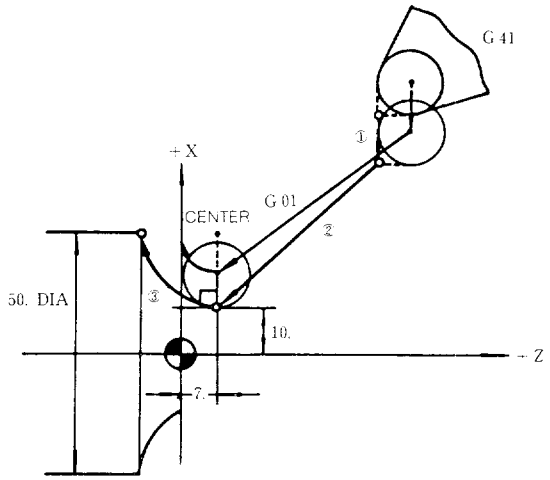
Compensation Mode




```
        (b)
①  G00  T+0202  ;
②  G01  G41  X20.  Z7.  F600  ;
③  G02  U30.  W-15.  I15  F0.2  ;
```

Movement at Start of Compensation (for G01)

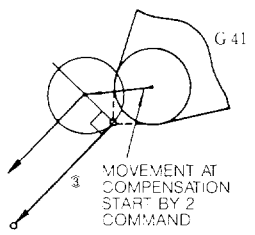
Compensation Mode



c. If the block of G41 (or G42 through G44) satisfying the compensation mode condition does not have the move command, the compensation starts and the tool center is moved on the normal.
Since G41 (or G42 through G44) involves such a movement, it is necessary to specify G00 or G01 in th last or current block for the G code of 01 group. Specification of a G code other than G00, G01, and G11 will result in alarm "026."

EXAMPLE (B): G41 (or G42 through G44) has no move command.

```
        (c)
①  G00  T+0303  ;
②  G01  G41  F···  ;
③  G01  X···  Z···  F···  ;
```

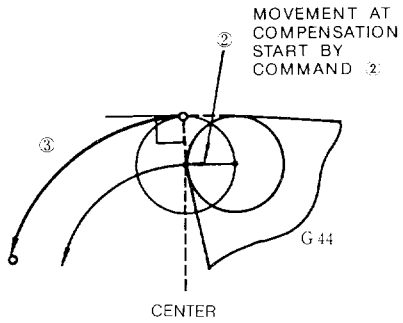


(d)

① G00 T-0404 ;

② G01 G44 F··· ;

③ G03 X··· Z··· I··· F··· ;



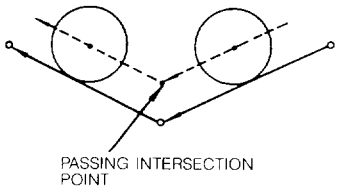
Note that the tool center is offset onto the normal to the start point of the block immediately after G41 (or G42 through G44) or T code, for each of above examples (a) through (b). If G41 (or G42 through G44) block or the block following T code has no move command, one block ahead is read and the compensation start operation is performed on that block. With no move command specified, up to two blocks may be programmed consecutively. However, if three or more blocks have no move command, an error is caused.

(6) Movement in compensation mode

When the tool nose radius compensation mode is entered by G41 (or G42 through G44) command, the tool center keeps moving along the path which has been offset by the tool nose radius by the program command, until the mode is cancelled by G40 or T□□00 command. The path is automatically calculated by the control. So, only the cut contour may be specified in the part program. However, the following should be considered for the inter-block movements and special contours:

a. Inter-block movement

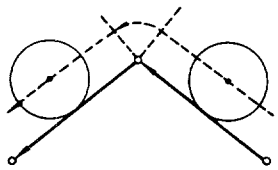
(i) For an inside corner (tangent-line angle is less than 180°), the intersection point is computed and is passed. (Intersection point computing formula.



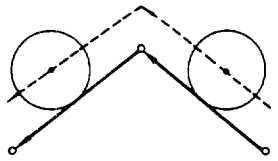
(ii) For an outside corner (tangent-line angle is more than 180°), the movement is controlled by the following M-code commands:

M96 ··· Tool radius compensation circular path ON

M97 ··· Tool radius compensation circular path OFF (execution of intersection calculation)



M96: Circular Path Mode



M97: Intersection Computing Mode

Movement of circular path is included in the previous block.
Normally, M96 is used for this operation However, when there is a possibility of an "overcut" in cutting special shapes with the M96, M97 should be used.

b. Movement in G00 mode

The instruction G00 positions tools independently along each axis toward the final offset position. Care should be taken on the cutter path so that tool does not contact the work.

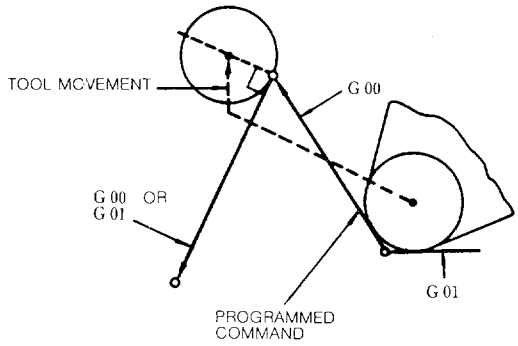


Fig. 2.32

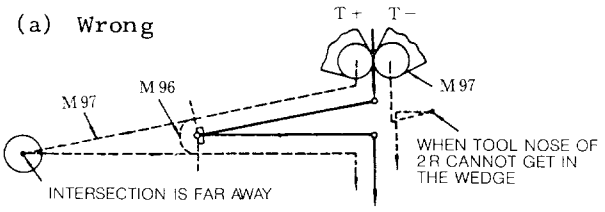
2.8.19 TOOL NOSE RADIUS COMPENSATION (G40 THROUGH G44)† (Cont'd)

(6) Movement in compensation mode (Cont'd)

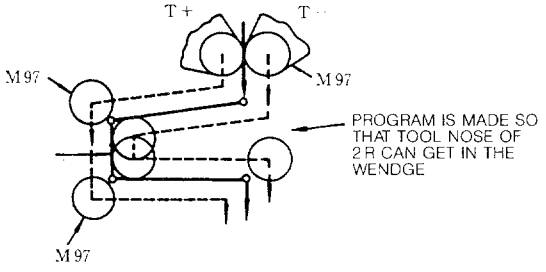
c. Programming consideration in compensation mode

(i) Be careful not to program a wedge-shaped cutting contour.

(a) Wrong

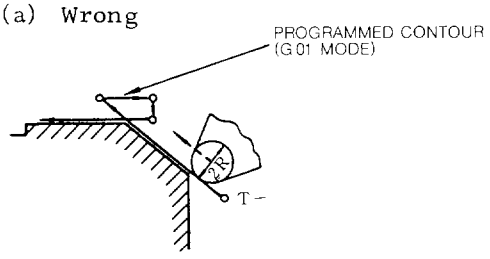


(b) Correct

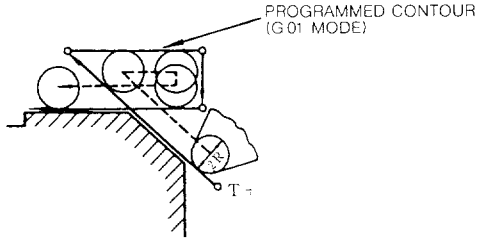


(ii) Program the tool movement so that the tool nose of 2R diameter can be in the contour.

(a) Wrong



(b) Correct



d. Command involving no movement in compensation mode

The control normally reads two blocks ahead during tool radius compensation mode and calculates the tool path. If either of these blocks gives no coordinate instructions such as G04 (dwell), the control reads a block further ahead and makes calculations. When coordinate instructions are missing in three or more blocks, tool radius compensation becomes impossible and accurate tool path cannot be obtained. Therefore, in a program where G41 to G44 are used ensure that, after them, two or more blocks without movement command in the compensation plane will not follow.

```
G01  G41  F···  ;
  .
  .
  .
G04  U···  ;  ┐
M···  ;       ┘  Compensation is normally made by the two or less blocks without move command.
  .
  .
  .
M40  ;
```

If no movement instruction is programmed in three consecutive blocks, tool center is offset on the normal line at the end point of the block immediate before them.

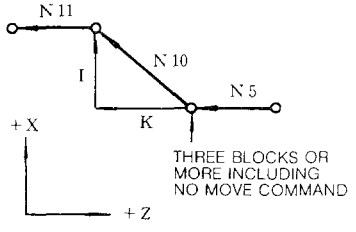
e. Use of dummy blocks

If it is impossible to specify a move command in three or more consecutive blocks and the offset on the normal line is not satisfactory, a dummy block may be inserted.
The dummy block does not cause an actual movement. This block is specifieid for the purpose of providing the data necessary for the tool nose radius computation. For the address of this dummy command, I and K are used.

I: X-axis dummy command (incremental).
K: Z-axis dummy command (incremental).

EXAMPLE:

```
N1 G01 G41 X··· Z··· F··· ;
N2 X··· Z··· ;
    .
    .
    .
N5 Z··· ;
N6 I··· K··· ;        ——Dummy block
N7 Mxx                ]
N8 G04 U··· ;         ] Three blocks or more
N9 MΔΔ                ]
N10 X··· Z··· ;
N11 Z··· ;
```

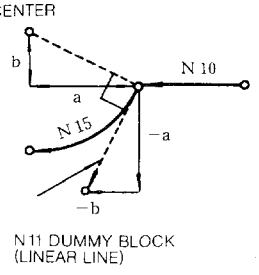


Namely, specify incremental commands I and K, which are equivalent to N10 block command, in N6 for the dummy block.

Remarks: If the purpose of the dummy block is a circular interpolation, generate the linear dummy block that specifies the direction of the tangent line at the start point of the circular command.

EXAMPLE:

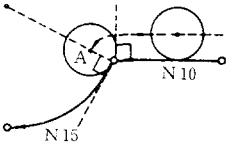
```
N10 G01 Z··· F··· ;
N11 G01 I(-a) K(-b) ;
N12 Mxx ;
N13 MOO ;          Dummy block
N14 MΔΔ ;
N15 G02 X··· Z··· I··· K··· ;
N16···
```



N11 DUMMY BLOCK
(LINEAR LINE)

I and K should be signed according to the type of the circular arc.

(M96: CIRCULAR PATH MODE)



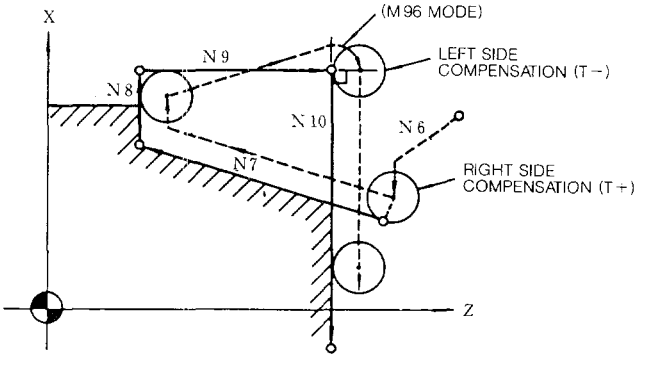
By dummy block N11, the linear command block of N10 stops at point A for the following circular movement.

f. Switching between T+ and T- in compensatior mode

This compensation provides the switching between T+ and T- without cancelling the compensation by G40 or T □□00 command.

EXAMPLE:

```
N5 G00 T+0101 ;      —— Designates right-hand side compensation facing the
N6 G41 X··· Z··· ;      proceeding direction
N7 G01 X··· Z··· F··· ;
N8 X··· ;
N9 T-0101 Z··· F600 ;  —— Changes left-hand side
N10 X··· F··· ;           compensation
```



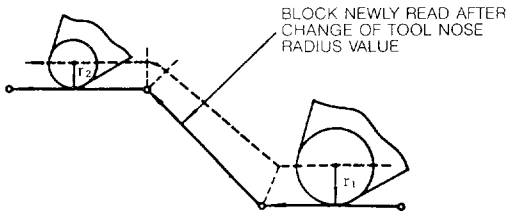
g. Modification of tool offset volume in compensation mode

It is invalid to newly specify a tool offset number by T code in compensation mode. The originally specified tool offset number remains valid until the compensation mode is cancelled by G40 or T□□00 command.

2.8.19 TOOL NOSE RADIUS COMPENSATION (G40 THROUGH G44) (Cont'd)

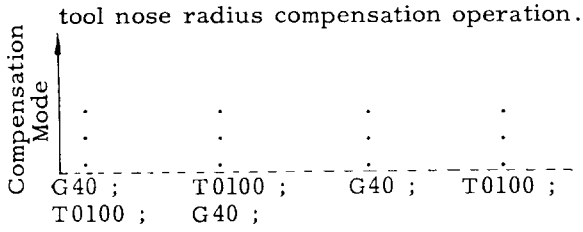
(6) Movement in compensation mode (Cont'd)

However, the tool nose radius value may be changed by varying the offset memory contents corresponding to the originally specified tool offset number, by means of MDI operation. After this modification, the new tool offset number is made valid beginning with the block newly stored in the prefetch buffer.



(7) How to cancel compensation mode

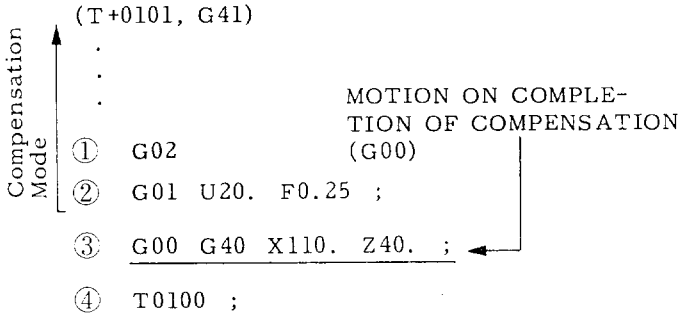
a. When G40 or T□□00 is specified, compensation mode is cancelled, terminating the tool nose radius compensation operation.

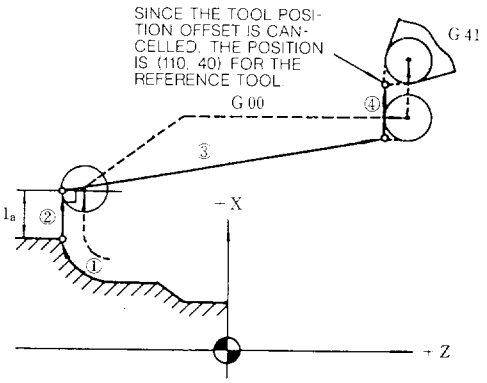


b. Upon termination of compensation, the tool center is offset onto the normal line to the end point of the final block in compensation mode, or the block immediately before that for which G40 or T□□00 has been specified. (Consequently, if a retracting which results in acute-angle contour is specified in G40 or T□□00 block, no uncut portion is produced.)
Then, the tool moves so that the virtual tool nose matches the end point of the move command specified in G40 or T□□00 block.

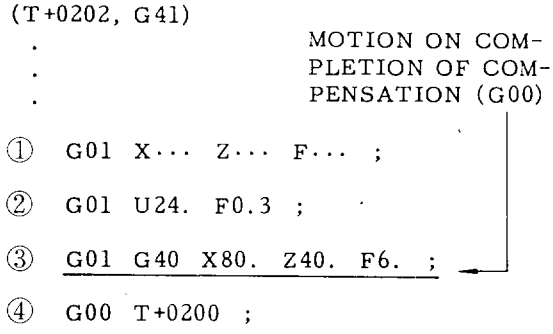
EXAMPLE A:

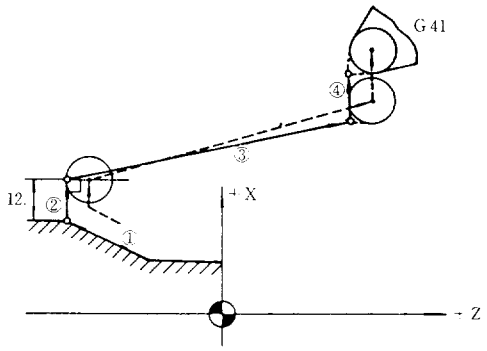
(a)





(b)





c. If the block of G40 (Nose radius compensation cancel) does not have the move command, the virtual tool nose moves to the specified end point.

G40 (and T□□00) commands contain movement. Specify G00 or G01 in the block previous to or the same as G40. If the G code of 01 group is other than G00, G01, or G11, alarm "027" will be activated.

EXAMPLE B: Move command is not included in G40 block for cancelling compensation.

(c)

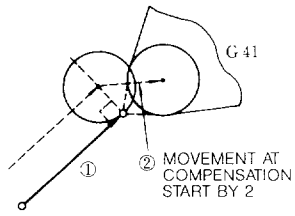
(T-0303, G41)

.
.
.

① G01 X··· Z··· F··· ;

② G01 G40 F··· ;

G00 T-0300 ;



(d)

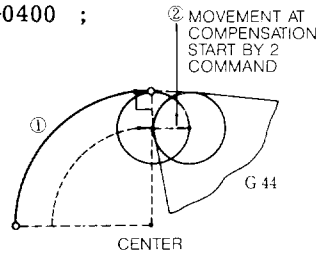
(T+0404, G44)

.
.
.

① G02 X··· Z··· K··· F··· ;

② G01 G40 F··· ;

G00 T+0400 ;



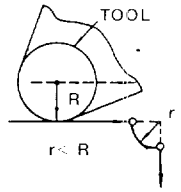
Note that, for each of above examples (a) through (b), the tool center is temporarily offset onto the normal line to the end point of the block immediately before G04 or T□□00 command.

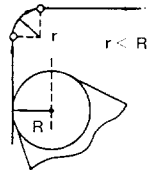
d. When tool nose radius compensation is cancelled by the use of T□□ 00 command, the tool position offset cancel operation is performed concurrently with the operation at tool nose radius compensation termination. The cancel operation provides the movement in which the virtual tool nose matches the last specified position for which tool position offset has been cancelled. If coexistence of these operations is not desired, cancel either of them by the use of G40 command.

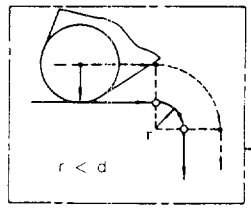
Notes:

a. Programmed shapes that produce input errors

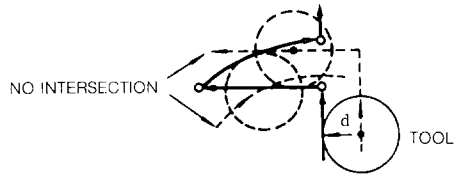
(i) When programming an inside arc with tool compensation, if programmed arc radius R ≦ tool radius d



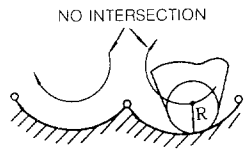




(ii) When no intersection point exists on the locus of the offset tool center.



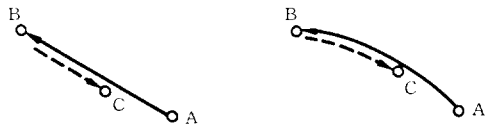
(a)



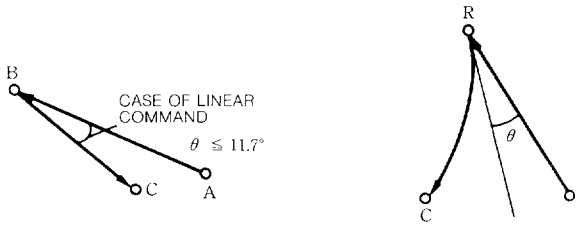
(b)

(iii) When reversing command or an angle close to reversing command is programmed in M97 (Outside Corner Circular Arc Path Off) mode.

(a) Reversing command



(b) Command close to reversing



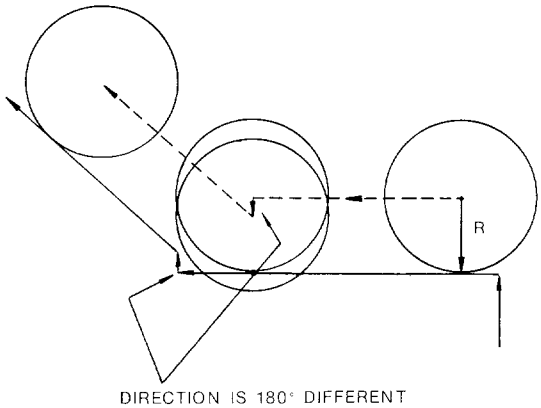
In M96 mode, all of the above shapes are correctly compensated.

b. Interference check

To prevent the tool from cutting deeper into the finished shape than was programmed in advance.

Interference check error is activated when the difference of moving direction is 90° to 270° between the programmed virtual tool nose and compensated tool center. When an error is detected, the block prior to the block in which overcutting may occur is immediately stopped and an alarm is shown.

This function does not always prevent overcutting. There might be occasions when overcutting is not detected as an error or normal cutting is detected as an error.



c. G codes usable in compensation mode

As a rule G codes other than shown below should not be used in the compensation mode.

Usable G codes	Remarks
G00, G01, G04, G06, G11 G96, G97 ...Constant surface speed control G98, G99 ...Feed function designation (G90, G91) ...Absolute/incremental designation	
G02, G03, G12, G22, G23 ...Command including circular arc G70, G71, G72, G73 ...Multiple repetitive cycle G111, G112 ...Multiple cornering (Beveling, rounding)	Inhibited in the block of compensation cancel or start

d. The subprogram (M98 or M99) may be specified in compensation mode.

e. When the tool offset number is T code command of "00," T□□00 command has the following two meanings:

(i) Tool position offset is cancelled.

(ii) Tool nose radius compensation is cancelled.

The following program can be specified:

```
N2  G41  ;
N3  G00  T+0101  ;
      .                    ]
      .                    ]
      .                    ]
      .                    ]   TOOL RADIUS
      .                    ]   COMPENSATION
      .                    ]   MODE WITH TOOL
      .                    ]   NO. "01"
      .                    ]
      .                    ]
N21 G00  T0100  ;          ]
      .                    ]
      .                    ]
      .                    ]   TOOL RADIUS
      .                    ]   COMPENSATION  } CAN-
      .                    ]                 } CEL
      .                    ]   TOOL POSITION
      .                    ]   OFFSET
      .                    ]
N25 G00  T-0202  ;         ]
      .                    ]
      .                    ]
      .                    ]   TOOL RADIUS
      .                    ]   COMPENSATION
      .                    ]   MODE WITH
      .                    ]   TOOL NO. "02"
      .                    ]
      .                    ]
N40 G00  T0200  ;
N41 G40  ;
```

f. Inhibition of MDI mode

(i) Operation in the MDI mode cannot be performed in the compensation mode. When RESET button is depressed, G00 (compensation cancel) becomes effective and the operation in the MDI mode becomes possible.

(ii) G40 through G44 cannot be written by the operation in MDI mode.

g. Command or operation for cancelling compensation

The following command or operation during tool radius compensation, cancels the compensation completely or temporarily. The command or operation should not be performed:

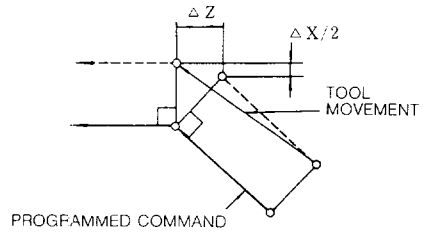
1. Three consecutive blocks without move command 2. M00 or M01 command 3. M-code set by parameter for stopping advance reading (6 Max)	Temporarily cancelled
4. G commands for stopping advance reading G36, G37, G38, G39 5. M02, M30 commands 6. Reset operation 7. Turning off power supply	Completely cancelled.

h. Commands causing error

The following commands must not be given, for they cause errors.

1. G28, G29, G30
2. G50, G51
3. G74, G75, G76
4. G90, G92, G94
5. G31 - G35
6. G68, G69
7. G122, G123

i. Even in M96 mode (tool nose radius compensation and circular arc path are on), if circular arc distances ΔX, ΔZ are smaller than the fixed values, the tool does not follow the corner circular arc path but moves directly to point B. The fixed values are those which are set by parameter #6230.



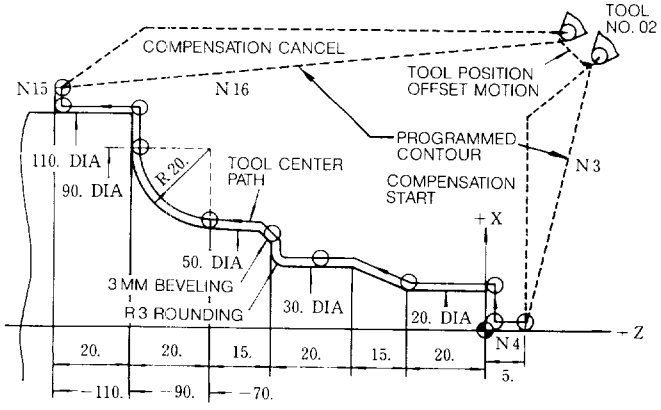
In case of $\Delta X/2 \leqq NEGNR$

$\Delta Z \leqq NEGNER$

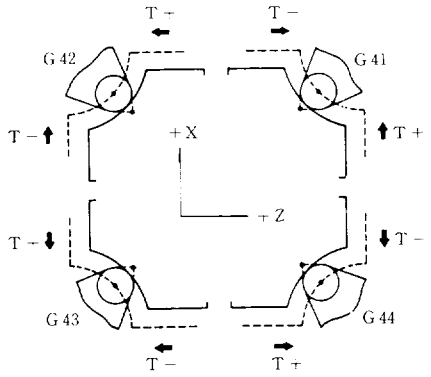
NEGNR: Constant value for parameter setting

EXAMPLE A:

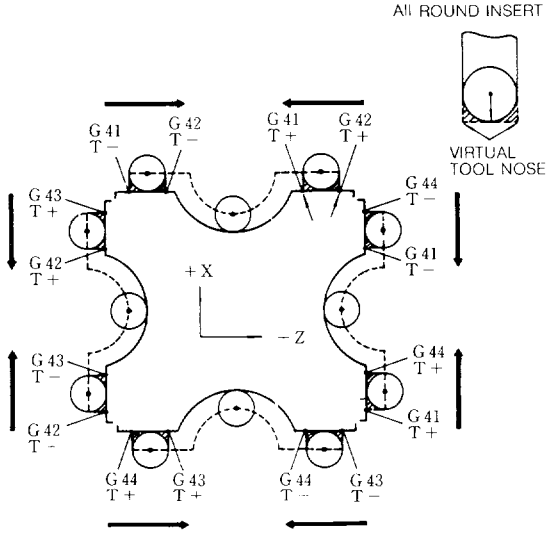
```
N1 G50 X140. Z20. ;
N2 G00 S1700 M03 T+0202 ;
N3 (G00) G41 X0 Z5. ;      ——TOOL NOSE COMPENSATION START BLOCK
N4 G01 Z0 F0.2 ;
N5 X20. ;
N6 Z-20. ;
N7 X30. W-15. S1100 ;
N8 G12 W-20. I3. ;         ————ROUNDING
N9 G11 X50. K-3. S700 ;    —BEVELING
N10 G01 X-70. ;                                          TOOL NOSE COMPENSATION
N11 G22 X90. Z-90. R20. S360 ; —— CIRCULAR ARC BY RADIUS
N12 G01 X110. S300 ;
N13 G04 U0 ;               ——DWELL
N14 (G01) Z-110. ;
N15 X120. ;
N16 G00 X140. Z30. T0200 ; —TOOL NOSE COMPENSATION COMPLETE BLOCK
N17 G40 ;
```



(7) How to cances compensation mode (Cont'd)



(a) Normal Insert



(b) All Round Insert (G code to be used is decided by setting side of virtual tool nose.)

Fig. 2.33 Relations between G code and Sign of T Code for Tool Nose Radius Compensation.

2.8.19.1 Automatic Nose R Function

Nose R compensation is performed by the nose R control point (0 to 9) command and G41, G42.

The offset side switching from the end face to the outer diameter becomes simple.

When the T command is executed, the tool is selected and the nose R and control point corresponding to the tool is also selected.

Then the tool coordinate system based on the machine coordinate system is set.

After the T command is given, the calculation for nose R compensation is performed by the control point and G41, G42.

2.8.19.2 Conditions for the Automatic Nose R Compensation Function to be Enabled

Automatic nose R compensation becomes effective by four conditions: the projected tool nose point designation set by the offset number, the radius of tool nose R, the offset side setting by G41 and G42, and the T code command.

The following are the details.

2.8.19.2.1 Designation of the projected tool nose point

(1) Direction of the projected tool nose point

The shift direction of automatic nose R is prescribed by determining the projected tool nose point position by specifying a numeral of 0 to 9.

The direction is as shown in Fig. 2.34 (X plus specification).

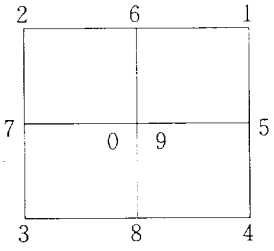


Fig. 2.34 Projected Tool Nose Direction

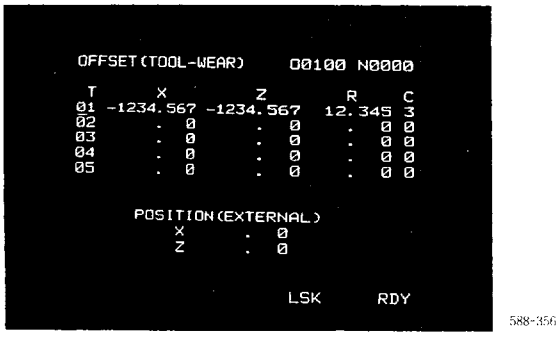
(2) How to set the projected tool nose point

The projected tool nose point is set through the offset screen (Fig. 2.35).

(a) Select the " OFS " function.

(b) Select the corresponding tool offset memory from T01 to T49.

(c) Enter " C "" 3 "" WR " via the MDI if the projected tool nose point is 3.



588-356

Fig. 2.35 Offset Screen

2.8.19.2.2 How to set the tool nose R data memory

The tool nose R data memory is set through the offset screen (Fig. 2.35).

(a) Select the " [OFS] " function.

(b) Select the corresponding tool offset memory from T01 to T49.

(c) Enter " [R] "" [8] "" [WR] " via the MDI if the tool nose R data is "0.8 mm".

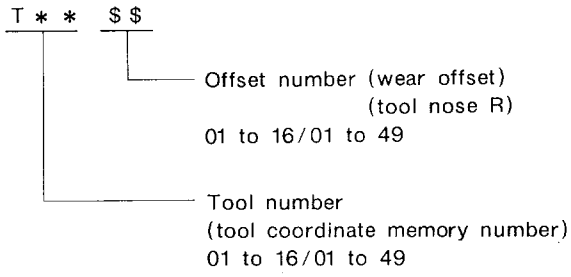
2.8.19.2.3 G41, G42 command

The offset direction of the current cutting is commanded by the G41, G42 commands (X plus specification).

G40: Nose R offset off
G41: Forward direction on the left is center of nose R
G42: Forward direction on the right is center of nose R

2.8.19.2.4 T code command

The T code number (4 digits) determines the coordinate system, wear offset, and tool nose R. For details, refer to par. 2.6.5 "New coordinate system setting function."



2.8.19.3 Offset Calculation of Automatic Nose R Compensation Approach and Relief

The automatic nose R compensation function has been improved from the nose R compensation function of the conventional YASNAC LX3.

2.8.19.3.1 The G00 → G00 block is not compensated. In short, positioning alone is performed.

2.8.19.3.2 Compensation is performed for approach G00 → G01 (G02, G03) by setting a projected vector (wall) on the tool to prevent excess infeed or shock, as follows.

Compensation is performed for relief G01 (G02, G03) → G00, by setting a projected vector (wall) to prevent insufficient cut, as follows.

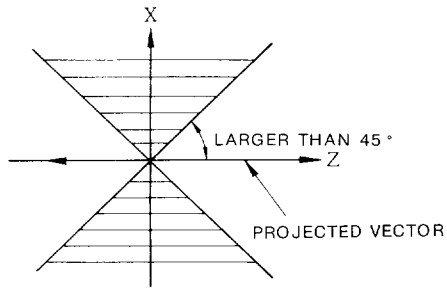
(1) Approach G00 → G01 or relief G01 → G00

(a) When the cutting command move axis direction is larger than 45° against the Z-axis ($|X/2| > |Z|$), a projected vector (wall) parallel to the Z-axis is assumed, to calculate the compensation with the cutting move axis.

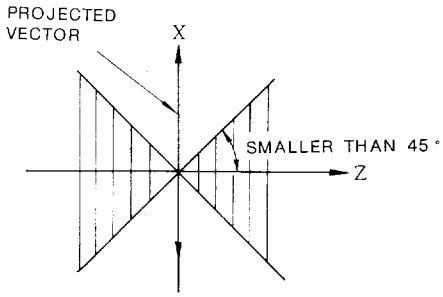
The machine moves toward this calculated position.

(b) When the cutting command move axis direction is smaller than 45° against the Z-axis ($|X/2| \leq |Z|$), a projected vector (wall) parallel to the Z- axis is assumed, to calculate the compensation with the cutting move axis.

The machine moves toward this calculated position.



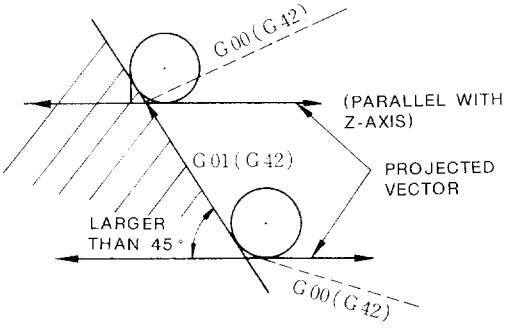
(a)

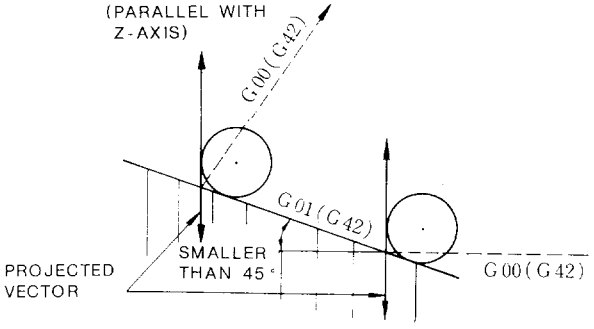


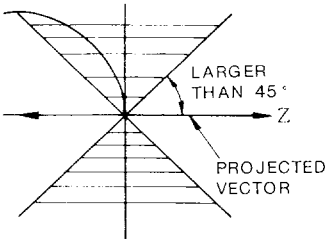
(b)

2. 8. 19. 3 Offset Calculation of Automatic Nose R Compensation Approach and Relief (Cont'd)

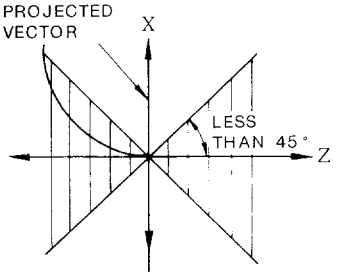
(Example) Tool nose point 3







(a)



(b)

(2) Approach G00 → G02 (or G03) or relief G02 (or G03) → G00

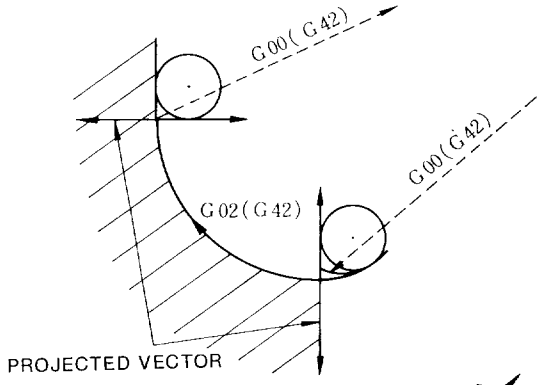
(a) When the circular cutting command move axis direction is larger than 45° against the Z-axis, a projected vector (wall) parallel to the Z-axis is assumed, to calculate the compensation with the cutting move axis.

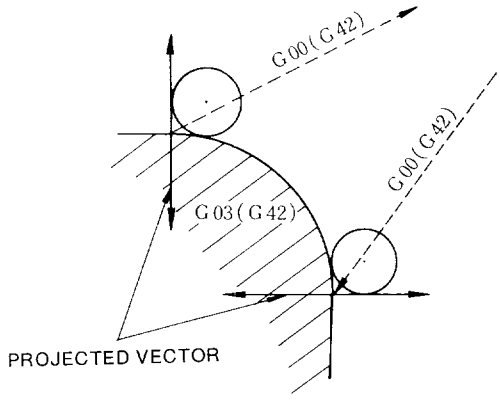
The machine moves toward this calculated position.

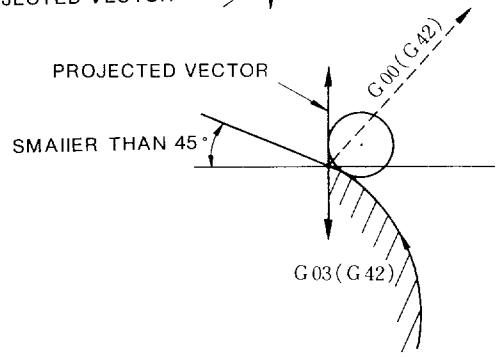
(b) When the circular cutting command move axis direction is smaller than 45° against the Z-axis, a vector (wall) parallel to the Z-axis is assumed, to calculate the compensation with the cutting move axis.

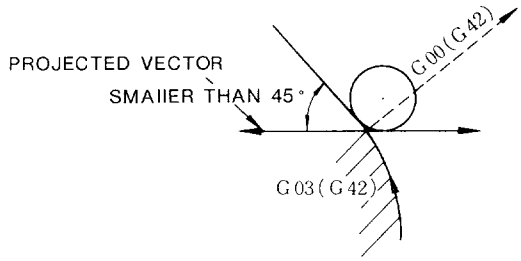
The machine moves toward this calculated position.

(Example) Tool nose point 3



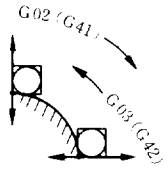
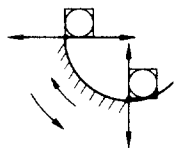
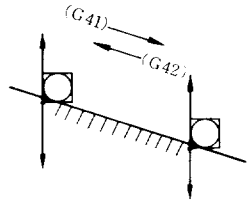
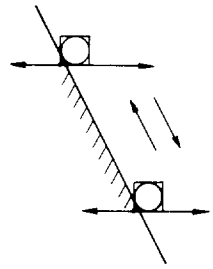
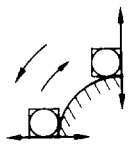
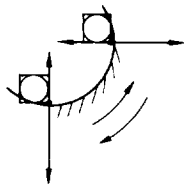
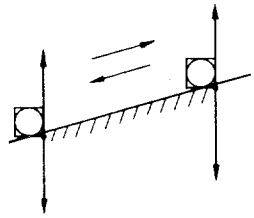
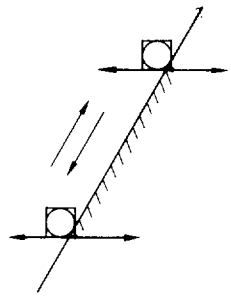
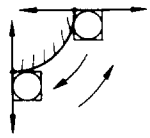
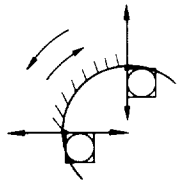
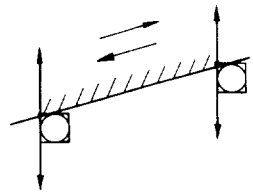
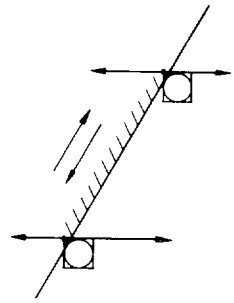
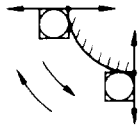
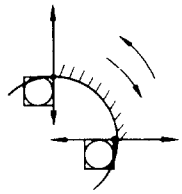
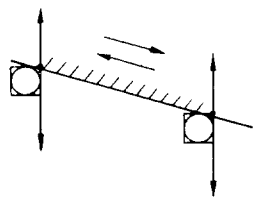
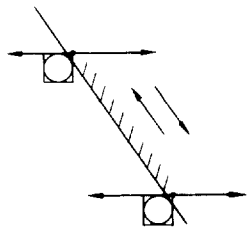






2. 8. 19. 3 Offset Calculation of Automatic Nose R Compensation Approach and Relief (Cont'd)

Table 2.20 A Automatic Nose R Approach and Relief Patterns

	CIRCULAR		LINEAR	
			45° OR LESS	MORE THAN 45°
① C=3				
② C=4				
③ C=2				
④ C=1				

2.8.19.4 The G40 G01 X__ Z__ I__ K__ ; Command Cancel Function is Also Available in the Automatic Nose R Function

The G40 G00 X__ Z__ I__ K__; command cancel function (mentioned later), as well as the same function by the G01 command are available.

This makes a projected wall by the I, K commands in the block before the G40 G01 command, to execute the nose R calculation.

Notes :

1. The nose R center is on the normal of the end point in the G40 G00 X__ Z__; block or G40 single-block.

2. The nose R complete cancel is performed in the block immediately before the G51 single-block or G28 block.

3. The cross point is calculated in the block immediately before G40 G00 X__ Z__ I__ K__; and the vector indicated by "I", "K."

4. When G01 and G41 (G42) are commanded at the same time, the nose R center will be on the normal of the start point of the next block.

   However, in canned cycle, start up can be performed by G01 G42 of the first block, but cannot be performed by repeating the command later.

5. G40 G01 X__ K__; and G40 G01 Z__ I__; are of the same command format as the chamfering command, but the cancel function has priority only when the G40 command is given.

2.8.20 PROGRAMMING OF ABSOLUTE ZERO POINT (G50)

Absolute coordinate system should be set before move command. After setting up the absolute coordinate system, all motions can be commanded on the absolute coordinate system.

(1) G50 X··· Z··· ;

This command makes the present position of tool tip the absolute coordinates (X, Z). The values with a sign following the addresses X and Z are the distances between tool tip and the absolute zero point (0, 0) to be set. Therefore, it can be said that "G50 command specifies the absolute zero point."

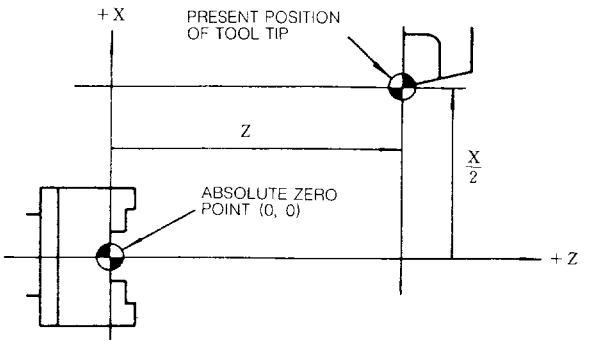


Fig. 2.34

(2) G50 U··· W··· ; (Incremental G50)

When the addresses U and W are specified instead of X and Z, the new absolute coordinate is set up by adding incremental values U (X-axis) and W (Z-axis) to the absolute coordinate previously set.

When the tools are very different in length, the incremental G50 (addresses U, W) is useful. The tools should be divided into two groups. Then, the difference between the length of the reference tool and that of the second group of tools can be set at the incremental G50 command and an absolute coordinate system can be stored.

G50 U100. W-100. ; ··· Setting of Position B

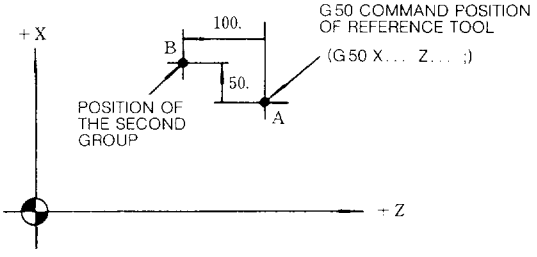


Fig. 2.35

2.8.20 PROGRAMMING OF ABSOLUTE ZERO POINT (G50) (Cont'd)

(3) Assume that the tool No. 01 is reference tool, and perform the setting of the following coordinate system for this tool:

```
G50 X80. Z62. ;
```

then, select the tool No. 02 which has the tool position compensation value shown in the figure below and peroform the compensation operation, and the tool No. 02 moves to point A.

```
N3 G50 X80. Z62. ;
N4 G00 T0101 ;
    .
    .
    .
N10 G00 T0202 ;
    .
    .
    .
```

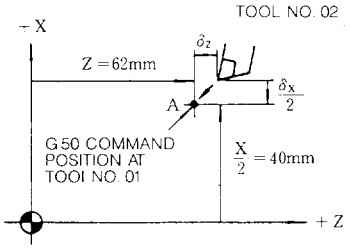


Fig. 2.36

If the coordinate system setting is performed with the reference tool and tool position compensation is applied to the other tool as shown above, the tool movement may be programmed on a single coordinate system for all tool noses.

Notes :

1. When T, S and M commands are programmed in the block following that containing G50, G00 should be programmed in the block. This designates the traverse rate for tool offset motion.

```
G50 X··· Z··· ;
G00 S500 M03 T0101 ;
```

2. G50 is a nonmodal G code which is valid only in the specified block. Generally, the other G codes, and M, S, and T codes cannot be specified in the same block. Note that G50 S··· ; or G50 T··· ; command is a separate feature and is not for coordinate system setting.

3. G50 should be commanded after the tool offset and tool radius compensation are cancelled.

4. When the power supply is turned on, the present position of tool is set to the coordinate (0, 0). Therefore, the absolute coordinate system should be set up before operation.

5. The current position of the tool in G50 coordinate system is shown in "POSITION ABSOLUTE" of current position display.

6. The coordinate system which was set is not affected by reset operation. The coordinate system is reset by one of the following operations:

   (i) The reset operation by ORG key (see 4.3.4.2, POSITION ABSOLUTE) is performed.

   (ii) G50 X0 Z0 ; command is written in MDI mode and is executed.

   (iii) The power is turned on again.

7. When setting work coordinate system by G50, parameter #6018 D_7 can select whether Work Coordinate System Shift in 2.6.5 will be effective or not.

2.8.21 MAXIMUM SPINDLE-SPEED SETTING (G50)†

This function is used for the control provided with S 4-digit designation option.

(1) G50 S··· ;

Four digits following the address S specifies the upper limit of spindle speed in rpm. If an S command exceeding the limit is issued in subsequent blocks, the spindle speed is governed at the upper limit.

(2) In G96 (Constant Surface Speed Control) mode, when spindle speed rises up too fast as the current X-coordinate of the tool is too small, the spindle speed is clipped the limit.

EXAMPLE

```
G50 S2000 ;
```

The maximum spindle speed is clipped at 2000 r/min.

Notes :

1. Maximum spindle speed specified by G50 can be displayed on the CRT display. Refer to 4.3.2.1, "Command Data Display."

2. The specified maximum spindle speed is not cleared by reset operation.

3. In case of S 4-digit designation B, unit of address S is not shown by rpm. Refer to machine tool builder's manual. For S 2-digit designation, this function cannot be used.

2.8.22 WORK COORDINATE MULTI-SHIFT (G50T, G51)†

This feasure is used in combination with "5.2.3, Measured Workpiece Value Direct Input" option. Hence it is necessary for the programmer to be familiar with paragraph 5.2.3.

The purpose of this feature is to retain a "work coordinate system" with a certain point on the work used as absolute zero point by performing G50T coordinate system setting at the replacement position of each tool. In other words, programming may be performed with a single coordinate system throughout the entire machining.

(1) Tool coordinate value memory (number)

a. It is necessary, before specifying G50T, to write the coordinate data for each tool to the tool coordinate memory. For the writing procedure, see 5.2.3, "Measured Workpiece Value Direct Input."

b. The number of available tool coordinate memory units corresponds to the number of tool offset memory combinations as shown below.

	Number of Tool Offset Memory Combinations	Available Tool Coorrdinate Memory Number
1	0 to 9	51 to 59 (9)
2	0 to 50	51 to 99 (49)

c. For ease of use, write the coordinate data for tool No. 01 in tool coordinate memory No. 51, etc., as shown below:

Tool Coordinate Memory	Tool No.
51 ⟷	⟷ 01
52 ⟷	⟷ 02
⋮	⋮
80 ⟷	⟷ 30

d. It is assumed that the tool coordinate memory contains the following coordinate data Xtn and Ztn for each tool Tn:

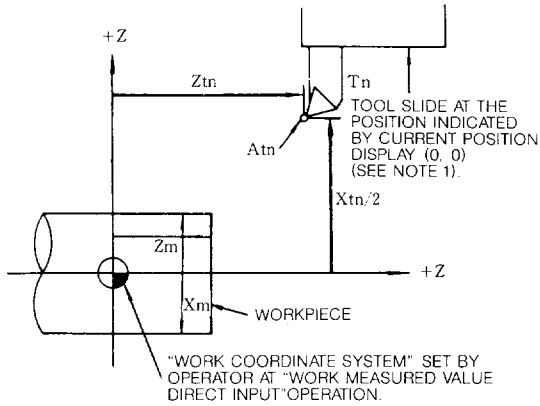


Fig. 2.37

(2) Work coordinate system setting (G50T)

a. G50 T□□△△

- Specifies tool offset number (00 to 50)
- Specifies tool coordinate memory number (51 to 99)

Using this command, set the coordinate system for each of X-axis and Z-axis with the following work coordinate system setting value:

$$\text{Work coordinate system setting value} = \left[\begin{array}{l}\text{Unit current}\\ \text{position value}\\ \text{(Note 1)}\end{array}\right] + \left[\begin{array}{l}\text{Content of}\\ \text{programmed tool}\\ \text{coordinate memory}\end{array}\right] + \left[\begin{array}{l}\text{Content of}\\ \text{programmed tool}\\ \text{offset memory}\end{array}\right]$$

Note: "Unit current position value" is the one which is shown in "POSITION [EXTERNAL] screen of the current position display (POS) on CRT display.

b. Usually, specify "00" in the tool offset number specification field ΔΔ.

EXAMPLE:

G50 T5100 ;

- "00" specifies the work coordinate system setting with the contents of tool offset memory being zero.

When the above program is specified with the tool slide at a given position (for example, -x, -z in the unit current position display), the work coordinate system defined by the operator is set correctly as shown below:

2.8.22 WORK COORDINATE MULTI-SHIFT (G50T, G51)† (Cont'd)

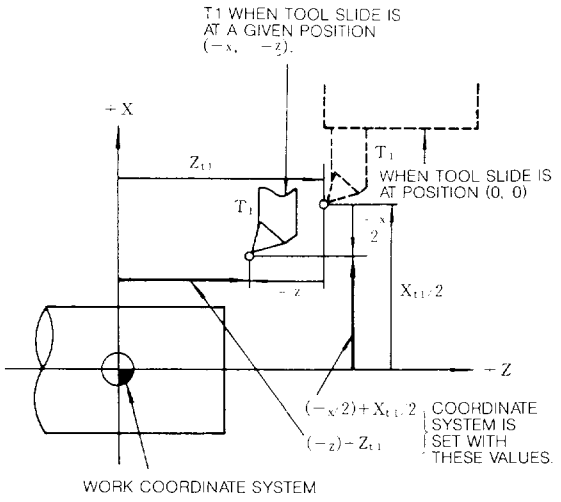


Fig. 2.38

c. G50 T0000 ;

By this command, the coordinate system is set with the unit current position value. This means that the canceling of the work coordinate system setting is performed with the content of tool coordinate memory = "0" and the content of tool offset memory = "0" by the specification of T0000.

(3) Return to current position origin (G51)

a. By G51 ; command, tool is returned to the point at rapid traverse rate where the unit current position value is (0, 0), on both X-axis and Z-axis.

b. With a part program that uses work coordinate system setting, the machining start point is the current position value (0, 0) in principle. Hence, the use of G51 command facilitates the return to the start point (0, 0) after completion of machining.

c. G51 command should always be specified on a single block basis.

Notes :

1. G50 T and G51 are nonmodal G codes which are valid only for the specified blocks.
2. When this function is used, set parameter #6005D5 to 0 (G50 preset of POS-EXTERNAL display is off).
3. G51 ; command is equivalent to the following two block commands.

```
G50 T0000 ;
G00 X0 Z0 ;
```

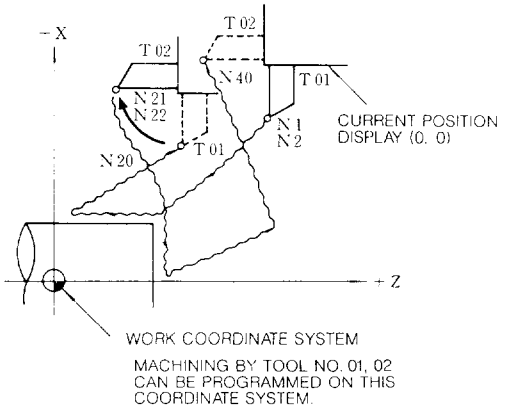
Consequently, after the execution of this command, the tool offset number is cancelled together with the work coordinate system, setting the tool offset number to "00."

4. When the G50 T work coordinate system setting is performed, 2.6.4. " Work Coordinate System Shift" is made valid.
5. The current position of the tool in the set work coordinate system is shown in the current position value "POSITION ABSOLUTE." It is not shown in POSITION EXTERNAL.
6. The work coordinate system set by G50 T cannot be cancelled by a reset operation.

EXAMPLE A:
(The start point is current position display (0, 0))

```
N1 G50 T5100 ; ——— Work coordinate system
                   setting for tool No. 01.
N2 G00 T0101 M03 S100 ;
   .               Selection of tool No. 01
   .               (Note 1).
(Machining by tool No. 01)
   .
   .
   .
N20 G00 X··· Z··· ; —— Positioning to a given
                       point.
N21 G50 T5200 ; ——— Work coordinate system
                    setting for tool
                    No. 02.
N22 G00 T0202 ; ——— Selection of tool
   .                No. 02 (Note 1).
   .
   .
(Machinig by tool No. 02)
   .
   .
   .
N40 G51 ; ——— Return to current position
              display (0, 0).
```

Note 1: The tool position offset in T0101 and T0202 commands may be used for the compensation for tool wear. When specified during machining, the tool position offset may also be used for the compensation for taper machining.

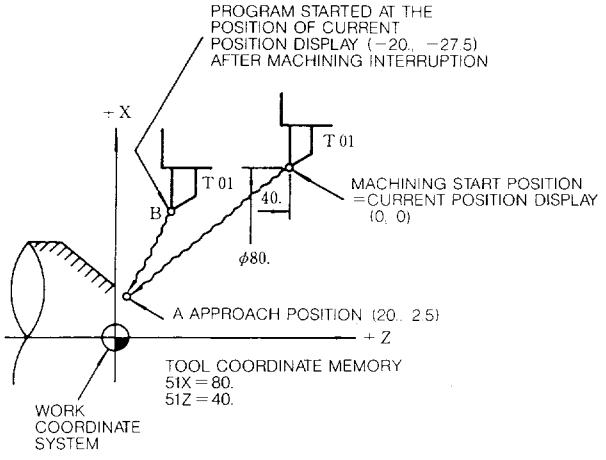


Notes :

EXAMPLE B:

If the machining which was started by the following program is interrupted and the program is restarted without returning the tool to the machining start point, the tool correctly moves to the first approach position.

```
N1  G50  T5100  ;
N2  G0101  ;
N3  G96  S150  M03  ;
Ⓐ N4  G00  X20.  Z2.5  ;
      .
      .
      .
```



This is because N1 G50 T5100 ; command at point B performs coordinate system setting with the following values to retain the work coordinate system, thus keeping approach position A unchanged:

X = (-20.) + (80.) = 60.

Z = (-27.5) + (40.) = 12.5

EXAMPLE C:

This example shows a program for which the replacement position of each tool is different from each other, and the values for work coordinate system setting.

Tool Coordinate Memory

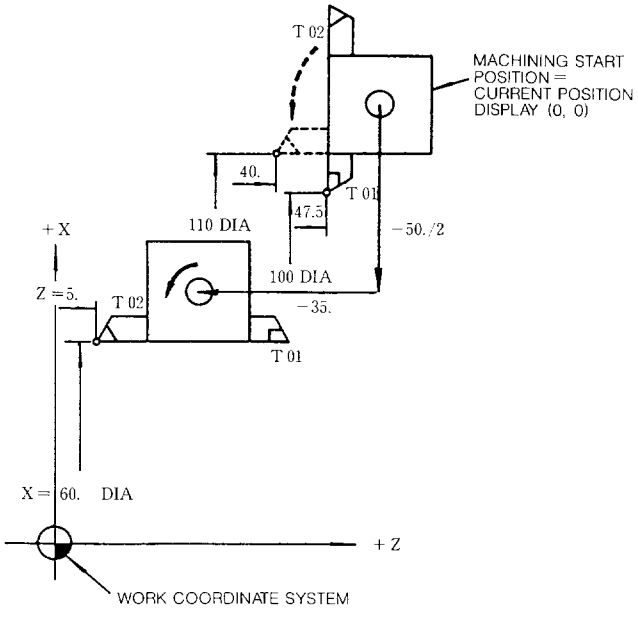
No.	X	Z
51	100.	47.5
52	110.	40.

```
N1  G50  T5100  ;
N2  G00 T0101  M03  S1000  ;
      .
      .
  (Machined by T01)
      .
      .
      .
N25  G50  T0000  ;
N26  G00  X-50.  Z-35.  ;
                ··· Tool replacement position
                    of T02 is current position
                    display (-50, -35).
N27  G50  T5200  ;
N28  G00  T02020  M03  S800  ;
      .
      .
   (Machining by T02)
      .
      .
      .
N48  G51  ;
```

The coordinate system setting values by this command are as follows:

X = (-50.) + 110. = 60.

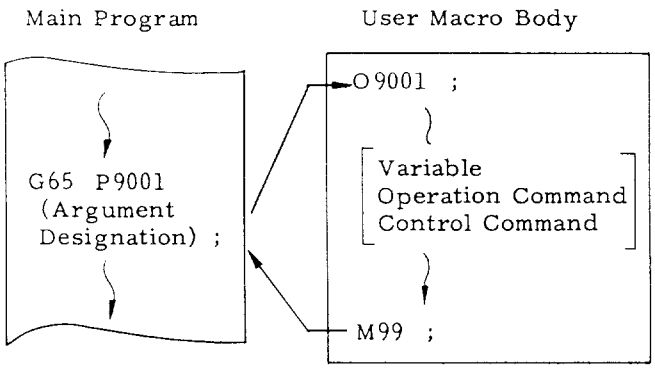
Z = (-35.) + 40. = 5.

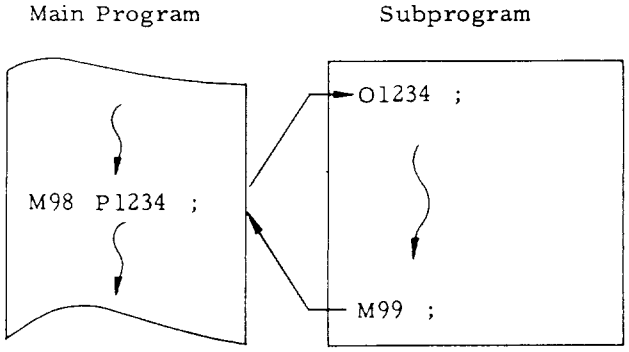


2.8.23 MACRO PROGRAMS (G65 AND G67)

Special programs written by the machine builder or user by the use of a group of instructions are registered in the part program memory. These programs can be called by the use of G65 or G66 command to execute them.

These special programs are referred to as the macro program body, which can be written and stored in the same format as a subprogram.





However, unlike a subprogram, a macro program allows:

(1) Use of variables.

(2) Computation between variables or between constants.

(3) Use of control commands such as a conditional branch.

These features enable the user macro body to provide a generalized program that requires complicated computations and decisions.

The "argument designation: in calling a macro program body from the main program makes it possible to assign the real numbers to the variables in the body. This enables this macro program to run as a series of specific programs that provide tool movements.
In this manual, the macro program body is sometimes referred to as simply, "macro."

2.8.23.1 Macro Program Call Commands

A macro program body may be called in the following five manners:

No.	Type of Call	Code	Remarks
1	Simple call	G65	
2	Modal call	G66	G67: For cancel.
3	Call by arbitrary G code	Gxx	
4	Call by M code	Mxx	
5	Call by T code	Txxxx	4 digits max.

(1) Simple Call (G65)

G65 P··· L··· (argument designation) ;

The macro program, whose program number was specified by P, is called and is executed L times. The default value of L is 1. When the designation of an argument to the macro program is desired, specify it in (argument designation). "Argument designation" is the assignment of real numbers to the "local variables" used in the macro program. For details, see 2.8.23.2, "ARGUMENT DESIGNATION."

(2) Modal Call (G66 and G67)

G66 P··· L··· (argument designation) ;

This command calls the macro subroutine specified by program number P. Each time a move command is executed, the specified macro is run L times.

G67 ;

This command cancels the modal call mode.

(3) Macro Call by Arbitary G Code

Gxx (argument designation) ;

This provides the command with is equivalent to G65 P... (argument designation) ;. For Gxx, ten G codes of G01 through G199 excluding those designated by NC maker can be set by parameter. The macro program numbers which correspond to these G codes are as follows:

#6120 ··· Sets G code which calls the macro of program number O9010.

#6121 ··· Sets G code which calls the macro of program number O9011.

)

#6129 ··· Sets G code which calls the macro of program number O9019.

Note : Macro call by arbitrary G code permits only single nesting. Namely, the macro which was called for by using arbitrary G code, M code, or T code does not permit another macro call by arbitrary code.

(4) Macro Call by M Code

G··· X··· Z··· Mxx ;

This command may call macros. In this case, the macro is executed after the move command is completed in that block. MF and M codes are not transmitted. For Mxx, four M codes may be designated by parameter excluding M00, M01, M02, M30, M90 through M99.

#6130 ··· Sets M code which calls the macro of program number O9001.

#6131 ··· Sets M code which calls the macro of program number O9002.

#6132 ··· Sets M code which calls the macro of program number O9003.

#6133 ··· Sets M code which calls the macro of program number O9004.

NOTE THAT THE MACRO CALL BY M CODE DOES NOT PERMIT ARGUMENT DESIGNATION.

When a macro M code is programmed in a macro subroutine that has been called by an arbitrary G code or by a macro M or T code, it will be processed like a normal M code.

(5) Macro Call by T Code

All the T code commands provide a macro call command.

G··· X··· Z··· Txxxx ;

With this command, the macro of program number O9000 is executed after the move command in the same block is completed.

Transmission of T code and TF signal is not performed. Whether a T code is to be used as a macro call command may be specified by the following parameter:

Parameter No.

#6134

0 ··· T code designation is handled as a T code.

1 ··· T code designation is handled as a macro call command to call the macro of program number O9000.

When a T code is specified as a macro call command, the value designated by T "xxxx" (up to decimal 4 digits) becomes the argument of common variable #149. NOTE THAT THE ARGUMENT DESIGNATION OTHER THAN THIS IS NOT PERMITTED. When a T code is programmed in a macro subroutine that has been called by an arbitrary G code or by a macro M or T code, it will be processed like a normal T code.

(6) Multiple Call

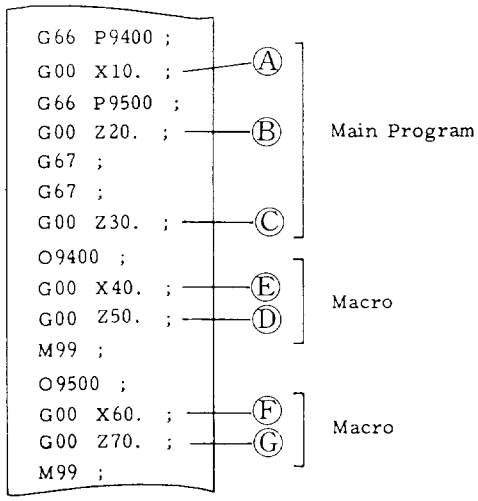
a. G65 Simple Call And G66 Modal Call

As a subprogram is called from another subprogram, a macro program may be called from another macro program. Quadruple nesting is permitted for simple call and modal call combined. Multiple call is disabled for the macro call by arbitrary G code, or M code or T code.

b. Multiple Call by G66 Modal Call

In modal call, each time a move command is executed, the designated macro is run. This is also valid for the move command in the macro called by multiple call. The macros are sequentially executed from the one designated latest.

Sample Program



The above sample program is executed in the following order:

Ⓐ → Ⓔ → Ⓓ → Ⓑ → Ⓕ → Ⓔ → Ⓓ → Ⓖ → Ⓔ → Ⓓ → Ⓒ

2.8.23.2 Argument Designation

Argument is the real value to be assigned to a variable used in the macro program body. Argument designation, therefore, is the act of assigning real values to variables. Argument designation is of type I and type II, which can be selected as required.

(1) Argument Designation I

Argument may be designated in any address except for G, L, N, O, and P. The relationships between the argument designation addresses and the variables are as shown below.

Address of Argument Designation I	Variable in Macro Program Body
A	#1
B	#2
C	#3
D	#7
E	#8
F	#9
H	#11
I	#4
J	#5
K	#6
M	#13
Q	#17
R	#18
S	#19
T	#20
U	#21
V	#22
W	#23
X	#24
Y	#25
Z	#26

For the address in which no argument need be designated, the command may be omitted.

Sample Program

```
                #1   #2    #24     #26
                 ↑    ↑     ↑       ↑
G65 P9100 L3 A30  B60  X34.8  Z50.2 ;
                                 |
                          Argument Designation
                          Part
```

(2) Argument Designation II

A, B, and C arguments and 10 sets of I, J, and K arguments may be designated. I, J, and K must be designated in this order. The relationships between the argument designation addresses and the variables are as shown below.

Address of Argument Designation II	Variables in User Macro Body
A	#1
B	#2
C	#3
I_1	#4
J_1	#5
K_1	#6
I_2	#7
J_2	#8
K_2	#9
I_3	#10
J_3	#11
K_3	#12
I_4	#13
J_4	#14
K_4	#15
I_5	#16
J_5	#17
K_5	#18
I_6	#19
J_6	#20
K_6	#21
I_7	#22
J_7	#23
K_7	#24
I_8	#25
J_8	#26
K_8	#27
I_9	#28
J_9	#29
K_9	#30
I_{10}	#31
J_{10}	#32
K_{10}	#33

The suffixes 1 through 10 to I, J, and K are determined by the order of the designated I, J, and K combinations.

For the address in which no argument need be designated, the command may be omitted.

Sample Program

```
                    #4   #5   #6   #7   #9
                     ↑    ↑    ↑    ↑    ↑
G65 P9005 A··· B··· C··· I··· J··· K··· I··· K··· ;
          |_____________________________|
                        |
              Argument Designation Part
```

(3) Position of Decimal Point in Argument

An argument may generally be designated with a sign and decimal point. For the designation without decimal point, the position of decimal point is as shown on the next page.

Address in Argument Designation	Metric Input	Inch Input
A, B	3	3
D, H	0	0
E	4	6
F (In G99 mode)	2	4
F (In G98 mode)	0	2
I, J, K, C	3 (2)	4 (3)
M, S, T	0	0
Q	0	0
R	3 (2)	4 (3)
U, V, W	3 (2)	4 (3)
X, Y, Z	3 (2)	4 (3)

The value shows the position of decimal point as counted from the least significant digit. The value in parentheses indicates the number of digits that follows decimal point at the time of parameter #6006-D5 = 1.

(4) Considerations in Argument Designation

a. Argument designation types I and II may be used concurrently. If the same variable has been designated twice, the last one is validated.

b. For both types I and II, addresses I, J, and K should be designated in this order. The other addresses may be designated in any order.

c. In the argument designation part, negative sign and decimal point may be used regardless of the address.

d. In G65 and G66 blocks, G65 and G66 should always be specified before each argument designation. This holds true with the macro call by G code.

2.8.23.3 Overview of Macro Program Body

A macro program body is programmed using the combination of the following commands.

(1) Variables

a. Local variable (#1 through #33)

b. Common variable (#100 through #549)

c. System variable (#1000 through #5104)

(2) Operation Commands

a. Arithmetical operations (+, -, *, /, ···)

b. Functional operations (SIN, COS, ROUND,···)

(3) Control Commands

a. Branch command (IF ⟨qualification⟩ GO TO n)

b. Repeat command (WHILE ⟨qualification⟩ DO m)

Using these commands, a program which requires complicated operations and conditional judgements may be written in the general format. Hence, the feature of user macro is to enable the programming of the wide range of NC functions from a simple machining cycle which is rather a subprogram to a special, complicated canned cycle, and the storing of these cycles in the machine. Described below are details of the commands mentioned above.

2.8.23.4 Variables

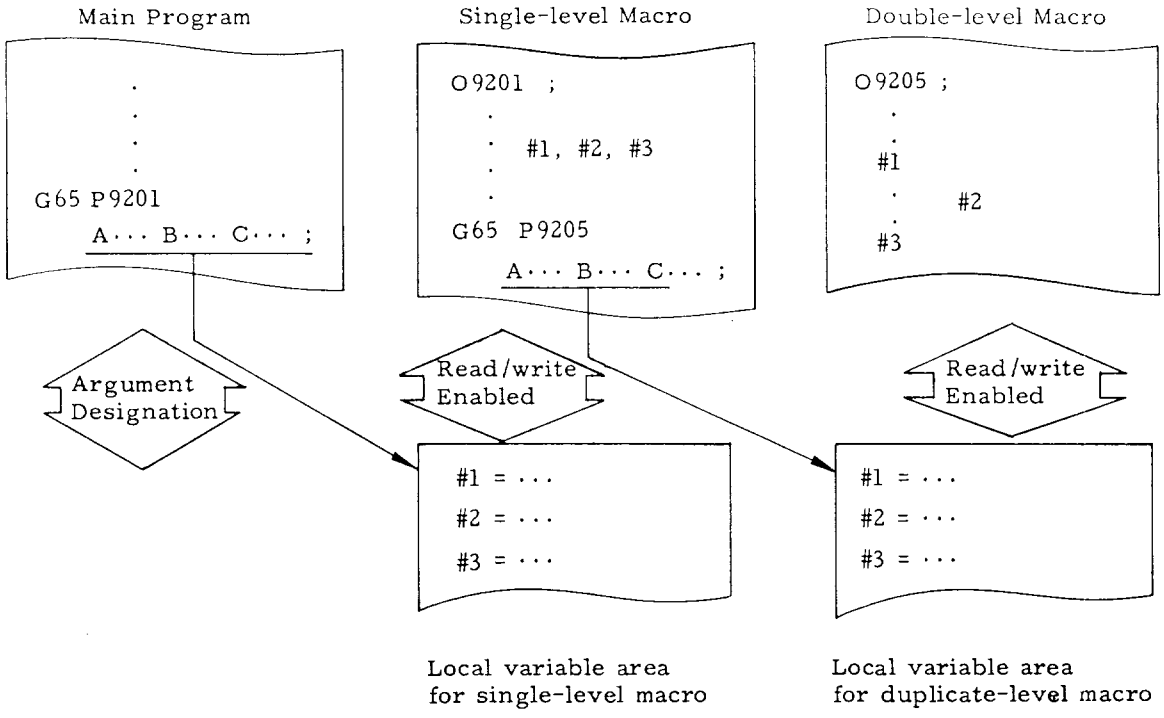
Instead of directly assigning a value to an address in a macro program body, the address may be designated by a variable. When this variable is called during execution, the corresponding value is fetched from the variable area to provide the address value.

There are three types of variables: local variable, common variable, and system variable. Each is identified by a variable number.

To the local variables, real numbers can be assigned using the argument designation part of macro call command by G65 or G66.

(1) Local Variables (#1 through #33)

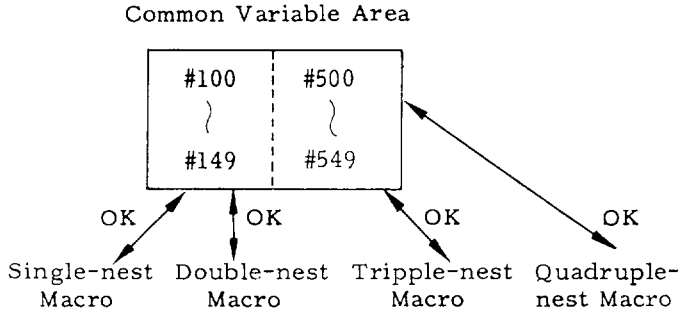
A local variable is the one that is used for each macro locally, That is, when the local variable is used, the variable area (#1 through #33) is independently allocated for each macro call, Certain values are stored by argument designation, and the results of operations in macro are retained.



Hence, the variables #1, #2, #3, ··· of the same macro assume different values each time it is called. Each local variable is reset for each macro call and is registered by argument designation. The variable not designated becomes "blank." Each local variable is set to "blank" at the time of power-on and reset operations.

(2) Common Variables (#100 through #149, #500 throuth #549)

A common variable may be shared by all macros and through all macros of all nesting levels. That is, the common variable enables a macro to refer to the results obtained by another macro.

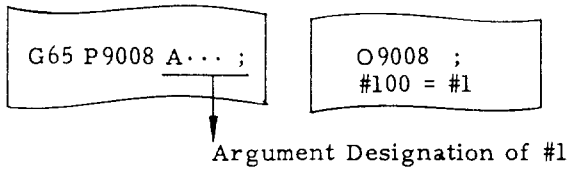


Common variables are divided into the following two types depending on clear conditions:

a. #100 through #149: These common variables are cleared at the time of power-on and reset operations and are set to "blank." In some controls, they are not cleared by reset operation if parameter #6008D1 is set at 1.

b. #500 through #549: These common variables are not cleared at the time of powe-on and reset operations.

The common variables are available to the user without restrictions. They cannot be designated by arguments. Indirectly, however, they can be designated as follows:



(3) System Variables

A system variable is the one whose use is unique to the system. There are following types of system variables:

a. Interface input signals ... #1000 through #1015, #1032†

b. Interface output signals ... #1100 through #1115, #1132†

c. Tool offset amount, tool coordinate data. and tool wear amount ... #2001 through #2050, #2051 through #2080, #2081 through #2099, #2101 through #2150, #2151 through #2180, #2181 through #2199, #2201 through #2250

d. Alarm message display ... #3000

e. Clock ... #3001, #3002

f. Single-block stop and auxiliary-function completion wait control ... #3003

g. Feed-hold feedrate-override, and exact-stop control ... #3004

h. RS232C data output ... #3100 (print out feature).

i. Modal information ... #4001 through #4120

j. Positional information ... #5001 through #5102

Note: The interface input and output signals of a. and b. may not be installed. Follow the specifications of the machine tool builder.

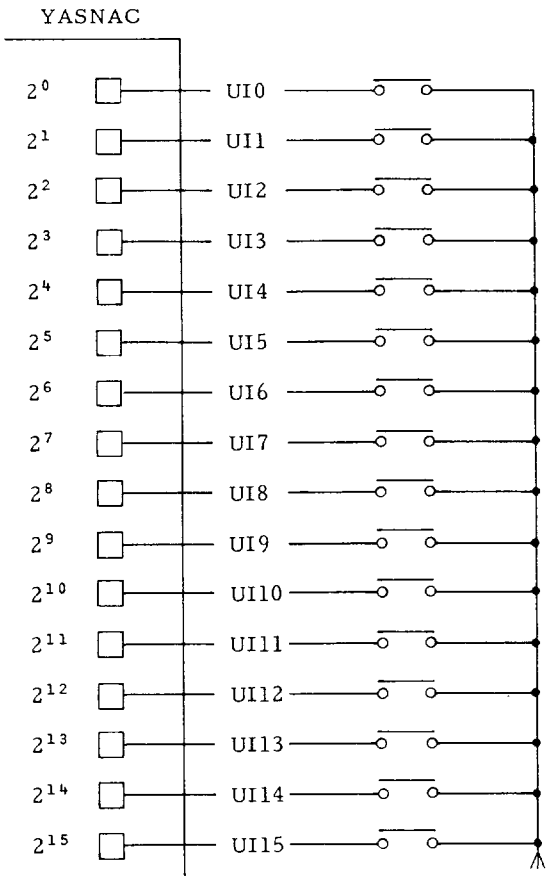
The following paragraphs describe the details of the variables mentioned above.

a. Intervace Input Signals (#1000 Through #1015, #1032)†

i. When one of the system variables, #1000 through #1015, is specified to the right-hand side of an operational expression, the on/off state of each of user-macro-dedicated 16-point input signals is read. The relationships between the input signals and the system variables are as shown below.

#1007	#1006	#1005	#1004	#1003	#1002	#1001	#1000
UI7 2^7	UI6 2^6	UI5 2^5	UI4 2^4	UI3 2^3	UI2 2^2	UI1 2^1	UI0 2^0
#1015	#1014	#1013	#1012	#1011	#1010	#1009	#1008
UI15 2^{15}	UI14 2^{14}	UI13 2^{13}	UI12 2^{12}	UI11 2^{11}	UI10 2^{10}	UI19 2^9	UI18 2^8

Variable Value	Input Signal
1	Contact Closed
0	Contact Open



Each read variable is 1. 0 or 0.0 when the associated contact is "closed" or "open" respectively, regardless of the unit system of the machine.

ii. When system variable #1032 is designated, the input signals (UI0 through UI15) that consist of 16 points (16 bits) are collectively read as a decimal positive value.

$$\#1032 = \sum_{i=0}^{15} \#[1000 + i] * 2^i$$

Sample Program

```
IF [ #1015 EQ  O ] GO TO 100;
```

Bit 2^{15} (UI15) is read and, if it is "0," a branch is made to sequence number N100.

```
#130 = #1032 AND 255
```

Bits 2^0 through 2^7 (UI0 through UI7) are collectively read to be stored in common variable #130 as a decimal positive value.

Note: System variables #1000 through #1032 cannot be placed to the left-hand side of operational expressions.

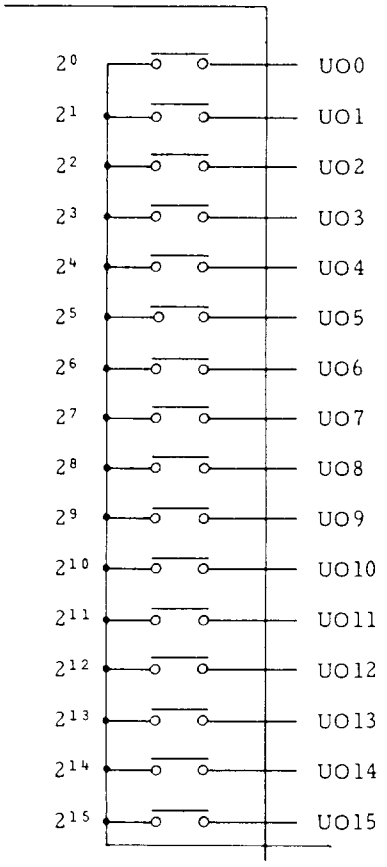
b. Interface Output Signals (#1100 Through #1115, #1132)

i. When one of system variables #1100 through #1115 is specified to the left-hand side of an operational expression, an on or off signal can be sent to each of the user-macro-dedicated 16-point output signals. The relationships between the output signals and the system variables are as shown below:

#1107	#1106	#1105	#1104	#1103	#1102	#1101	#1100
UO7 2^7	UO6 2^6	UO5 2^5	UO4 2^4	UO3 2^3	UO2 2^2	UO1 2^1	UO0 2^0
#1115	#1114	#1113	#1112	#1111	#1110	#1109	#1108
UO15 2^{15}	UO14 2^{14}	UO13 2^{13}	UO12 2^{12}	UO11 2^{11}	UO10 2^{10}	UO9 2^9	UO8 2^8

Variable Value	Output Signal
1	Contact Closed
0	Contact Open

YASNAC



When 1.0 or 0.0 are substituted in any of #1100 through #1115, the associated output contact is output in the "closed" or "open"state.

ii. When system variable #1132 is specified, the output signals (UO0 through UO15) that consist of 16 points (16 bits) are collectively output. At this time, the decimal positive value substituted in #1132 is output in the form of binary 16-bit value.

$$\#1132 = \sum_{i=0}^{15} \#\left[1100 + i\right] * 2i$$

iii. With system variables #1100 through #1132, the value sent last is retained. Hence, when one of them is written to the right-hand side of an operational expression, its value is read.

iv. Considerations

When any values other than 1.0 or 0.0 are substituted into one of the system variables, #1100 through #1115, the values are handled as follows:

"Blank" and any values less than 0.5 are assumed to be "0." Any values of 0.5 and over and other than "blank" are assumed to be "1."

Sample Program

```
#1107 = #10 ; (#10 = 1.5)
```

The output signal of bit 2^7 (UO7) is outputted in the contact (closed) state.

```
#1132 = (#1132 AND 240) OR (#8 AND 15)
```

The output signal of bits 2^4 through 2^7 (UO4 through UO7) are outputted without change and the contents of local variable #8 are outputted to the output signals of bits 2^0 through 2^3 (UO0 through UO3).
(Decimal 240 = 11110000 Binary 15 = 00001111)

c. Tool Offset Amount And Tool Coordinate Data, Tool Wear Amount

```
#2001 - #2050, #2051 - #2099, #2151 - #2199,
#2101 - #2150, #2201 - #2250
```

i. When one of the system variable #2001 through #2250 is specified to the right-hand side of an operational expression, the tool offset amount, tool coordinate data, and tool wear amount can be read.

ii. The relationships between the tool offset numbers and the system variables are as shown below:

	System Variable	Tool Offset Memory No.
X-axis	#2001 to #2050	01 to 50
Z-axis	#2101 to #2150	01 to 50
Tool Nose Radius	#2201 to #2250	01 to 50

	System Variable	Tool Coordinate Memory No.
X-axis	#2051 to #2099	51 to 80
Z-axis	#2151 to #2180	51 to 80

iii. When one of the above system variables is specified to the left-hand side of an operational expression, its value can be changed.

Sample Programs

```
#116 = #2016 ;
```

The contents of tool offset number 16 for X-axis are substituted for common variable #116.

```
#2081 = #24 ;
```

The tool wear amount (memory No. 81) of X-axis is erased and the contents of local variable #24 are set.

d. Alarm Message Display (#3000)

When a condition to be alarmed occurs in a user macro program, system variable #3000 may be specified to put the machine in the alarm state.

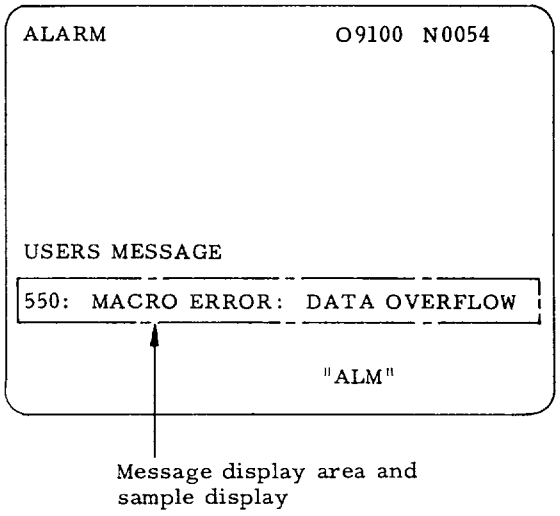
i. #3000 = n (<alarm message>);

Using this command, specify the alarm message (less than 32 characters) preceded by a 3-digit alarm number n and enclosed with control-in and control-out symbols. The alarm number should be three digits and not be one used by the machine.

ii. When this #3000 command is executed, "ALM" or "A/B" is displayed on the bottom of CRT screen regardless of the mode and function. Its message can be seen by the following operation:

Press ALM function key.

The alarm number and message and displayed on the bottom of CRT screen.



Message display area and sample display

When RESET key is pressed after removal of the cause of alarm, the message display and the alarm state can be cleared.

Sample Program

```
#3000 = 550 (MACRO ERROR: DATA OVER-
            FLOW)
```

e. Clock (#3001, #3002)

i. When system variable #3001 or #3002 for clock is specified, the clock can be read.

System Variable	Type	Unit	At Power-On	Count Condition
#3001	Clock 1	1 ms	Reset to "0"	Always
#3002	Clock 2	1 s	Same as power-off time	When STL signal is on

ii. To preset the clock, substitute the value with this system variable put at the left-hand side of the expression.

Sample Program

```
#3001 = 0 ; ··· The clock is preset to value "0."
```

iii. Restrictions

The accuracy of clock 1 is 8 ms. When 4294968000 msec has been reached, and overflow occrus, setting the clock to "0."

The accuracy of clock 2 is 8 ms. When 429496800 sec has been reached, an overflow occurs, setting the clock to "0."

Sample Program

Main Program

```
G65 P9351 A5.;
```

Argument designation of variable #1

··· Macro to wait for 5 seconds

Macro Program

```
O9351 ;
#3002 = 0 ;
WHILE [ #3002 LE #1 ] D01 ;
END 1 ;
M99 ;
```

f. Single Block Stop And Auxiliary Function Completion Wait Control (#3003)

When the value listed in the following table is substituted in system variable #3003, the single block switch can be disabled or the next block may be entered without waiting for the checking of the finish signal (FIN) of the auxiliary function (MST).

When the finish signal is not waited for, the distribution end signal (DEN) is not transmitted. In this case, the FIN is waited for in the block with the check skip cleared. Hence, when the FIN is not waited for, be careful not to specify the next auxiliary function.

#3003	Single Block Switch	FIN Signal
0	Valid	Waited
1	Invalid	Waited
2	Valid	Not waited
3	Invalid	Not waited

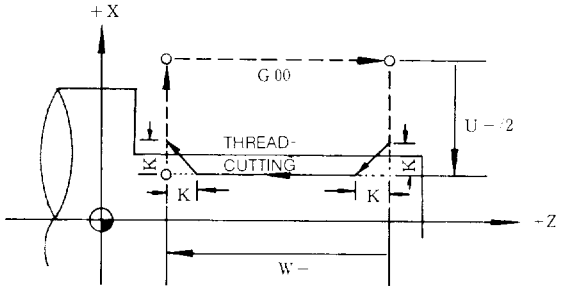
g. Feed-Hold, Feedrate-Override, And Positioning Completion Control (#3004)

When the value listed in the following table is substituted in system variable #3004, feed hold, feedrate override, and positioning completion can be made valid or invalid.

#3004	Feed Hold	Feedrate Override	Positioning Completion
0	Valid	Valid	Valid
1	Invalid	Valid	Valid
2	Valid	Invalid	Valid
3	Invalid	Invalid	Valid
4	Valid	Valid	Invalid
5	Invalid	Valid	Invalid
6	Valid	Invalid	Invalid
7	Invalid	Invalid	Invalid

Sample Program

Special Threadcutting Cycle (Incremental Command)



Macro Call

```
G65 P9093 U-··· W-··· K··· F··· ;
```

#9: Lead

#6: Without sign, radius value

#23: Negative value

#21: Negative value, diameter value

Macro Program

```
O9093 ;
M93 ;                          ——— Buffering 4 blocks
#10 = ROUND [#6] *2 ;
#11 = ROUND [#21] +#10 ;
#12 = ROUND [#23] +#10 ;
#3003 = 1 ;                    ——— Single block switch invalidated
G00 U#11 ;
#3004 = 7 ;                    ——— Feed hold invalidated
G22 U-#10 W-#6 F#9 ;                Feedrate override invalidated
G32 W#12 ;                          Positioning completed
G32 U#10 W-#6 ;
#3004 = 0 ;                    ———
G00 U-#11 ;
G00 W-#23 ;
#3003 = 0 ;                    ———
M92 ;
M99 ;
```

h. RS232C Data Output (#3100)

When system variable #3100 is specified, messages and NC internal data can be output to external equipment via RS232C data input/output interface. If the external equipment is a printer, the above information is printed.

i. Output of Messages

#3100 = (<Message>)

When this command is specified, the message enclosed by control-in and control-out is output, via RS232C interface.

Each output message is followed by CR/LF (Carriage Return/Line Feed). Hence, when #3100 = ( ) is specified, only CR/LF is outputted, which is useful in tabulating the punched data.

Note: When the message is output, it should be enclosed by control-in and control-out.

ii. Output of Data

#3100 = [< variable >]

When this command is specified, the value of the local variable, common variable, or system variable at the right-hand side is output via RS232C interface as plus or minus decimal 9 digits (4 digits after the decimal point, 5 digits before the decimal point) data.

Notes:
1. The value is rounded out to the fifth digit after the decimal point.
2. When the value is of 6 digits or more before the decimal point, the asterisk is output.

iii. The above output is performed when system variable #3100 is executed in the macro program. It is required, therefore, to previously attach the external equipment such as a printer via RS232C interface and preset the parameters that use the interface.

Sample Program

```
#3100 = (  ) ; ··· Carriage return/line feed
#3100 = (TOOL OFFSET 01);
#3100 = (␣␣ X␣␣␣␣␣␣␣␣␣␣
        Z ␣␣␣␣␣␣␣␣␣␣ R);
#3100 = [#2001]; ··· = 10.000 mm
#3100 = [#2101]; ··· = -10.000 mm
#3100 = [#2201]; ··· = 0.800 mm
#3100 = (  ) ;
```

Printout Data

```
TOOL OFFSET 01
     X          Z          R

␣␣␣␣10.0000␣␣␣-10.0000 ␣␣␣␣␣0.80000
```

A maximum of 6 digits (data plus signs) before the decimal point can be output.

i. Modal Information(#4001 Through #4120)

i. When one of system variables #4001 through #4120 is specified, the modal commands that are specified up to the immediately preceding block can be known. These modal commands are sometimes called the current values of modal information commands.

System Variable	Modal Information
#4001	G code (group 01)
⋮	⋮
#4021	G code (group 21)
#4108	E code
#4109	F code
#4114	Sequence number
#4115	Program number
#4119	S code
#4120	T code

Notes:
1. M code can not be read out because of non-modal information.
2. The value of either E (#4108) or F (#4109) is specified just before being stored, giving #4108 and #4109 the same value.

ii #4001 through #4120 cannot be placed to the left-hand side of the operation expression.

Sample Program

Main Program

```
G65 P9602 <Argument Designation> ;
```

Macro Program

```
O9602 ;
#1 = #4001 ;          ← G codes (G00 through G03) of 01 group are retained.
G00 X··· Y··· ;
G01 Z··· F··· ;
G03 X··· Z··· R··· ;
G00 Z··· ;
G#1 ;                 ← G codes of 01 group are restored.
M99 ;
```

j. Positional Information (#5001 Through #5102)

When system variables #5001 through #5102 are specified, various positional information can be obtained.
The unit of the information is millimeters or inches.

	Unit
Metric input	0.001 millimeter
Inch input	0.0001 inch

In the macro program body, the "input unit x 10" feature is invalid.

System Variable	Positional Information	Read During Move
#5001	X-axis block end position (ABSIO)	Enabled
#5002	Z-axis block end position (ABSIO)	
#5021	X-axis current position (ABSMT)	(Note) Enabled
#5022	Z-axis current position (ABSMT)	
#5041	X-axis current position (ABSOT)	(Note) Enabled
#5042	Z-axis current position (ABSOT)	
#5061	X-axis skip signal position (ABSKP)	(Note) Enabled
#5062	Z-axis skip signal positon (ABSKP)	
#5081	X-axis tool offset amount	Enabled
#5082	Z-axis tool offset amount	
#5101	X-axis servo position deflection amount	(Note) Enabled
#5102	Z-axis servo position deflection amount	

Note: Reading of #5021, #5022, #5041, #5042, #5101, and #5102, when commanded during movement, will be performed after completion of the movement. Accordingly, nose radius compensation is not performed.

Mnemonic	ABSIO	ABSMT	ABSOT	ABSKP
Meaning	End position of block immediately before.	Command current position (same as POS. MACHINE display).	Command current position (same as POS. ABSOLUTE display).	Position at which skip signal did not go on in G31 block.
Coordinate System	Work coordinate system	Machine coordinate system	Work coordinate system	Work coordinate system
Tool Position Offset	Not included		Included	Included
Tool Radius Compensation Amount	Not included		Included	Included

Notes:
1. When the skip signal is not turned on in G31 block, the skip signal position is at the end of G31 block.
2. The "input unit x 10" feature is valid up to the macro call block (the argument designation part by G65 or G66) but is in valid in the user macro body.
3. System variables #5001 through #5102 may not be placed to the left-hand side of operational expression.

Sample Program

The tool is positioned to the specified location (X, Z) on machine coordinate system, performs the specified M feature, and returns to the start point.

Main Program

```
G65 P9603 X... Z ... M... ;
```

#13
#26
#24

Macro Program

```
O9603 ;
#1 = #5001 ;
#2 = #5002 ;

G91 ;
G00 X [#24-#5021] ;
G00 Z [#26-#5022] ;
M#13 ;
G00 Z#2 ;
G00 X#1 ;
M99 ;
```

List of Variables

Variable No.	Meaning
# 1 to #33	Local variables.
#100 to #149	Common variables (reset to "blank" at power-off).
#500 to #549	Common variables (retained at power-off).
#1000 to #1015	Interface input signals (each signal for each bit).
#1032	Interface input signal $\left(\sum_{i=0}^{15} \#[1000+i] * 2^i\right)$.
#1100 to #1115	Interface output signals (each signal for each bit).
#1132	Interface output signal $\left(\sum_{i=0}^{15} \#[1100+i] * 2^i\right)$.
#2001 to #2050 #2101 to #2150 #2201 to #2250	Tool offset amount (X-axis, Z-axis, nose radius)
#2051 to #2080 #2151 to #2180	Tool coordinate data (X-axis, Z-axis)
#2081 to #2099 #2181 to #2199	Tool wear amount (X-axis, Z-axis)
#3000	Alarm message display.
#3001	Clock 1 (In units of 1 ms).
#3002	Clock 2 (in units of 1 s).
#3003	Single block stop, auxiliary function complete wait control.
#3004	Feed-hold, feedrate-override, and exact-stop control.
#3100	RS 232 C data output (print out feature).
#4001 to #4120	Current value of modal information command.
#5001 to #5002	End position of the immediate preceding block (for each axis).
#5021 to #5022	Current position of machine coordinate system (for each axis).
#5041 to #5042	Current position of POS. UNIVERSAL (for each axis).
#5061 to #5062	Position at which G 31 skip signal is turned on (for each axis).
#5081 to #5082	Effective tool position offset amount (X-axis, Z-axis).
#5101 to #5102	Servo position deflection amount (for each axis).

(4) Variable Representation

Each variable is represented in a variable number that follows #.

a. How to designate a number directly:

#i (i = 1, 2, 3, 4, ...)

Sample
#10
#130
#2000

b. How to designate an expression as a variable number:

[<expression>]

Sample
[#100]
[#500 + 1]
[#20/2]

In the following description, variable #i may be replaced with variable #[< expression >].

(5) Variable Reference

a. The value that follows an address may be replaced by a variable.

When < address> #i or <address> -#i is specified, the value of the variable or its negative value (complement, more exactly) is made the specified value of the address.

Sample
```
#30 = 1. 0 ;
#101 = 100. ;
#103 = 300. ;
#140 = 0. 3 ;
G#30 X#101 Z-#102 F#140 ;
```

The above specification is equivalent to the specification below.

```
G01 X100. Z-300. F0.3 ;
```

Notes:

1. Address /, O, and N may not refer to variables.

Sample
/#8, N#100 ... Error.

2. A variable number may not be replaced with a variable.

Sample
##20 ... Error.
#[#20] ... Correct.

3. When a variable is used as address data, the values below the least significant digit are rounded.

Sample

(i) When #1 = 45.2346
X#1 ... = X45.235 mm (for metric input)

(ii) When #2 = 0.255
F#2 ... = F0.26 (mm/rev)

(iii) When #3 = 5.37672
G04 P#3 ... = G04 P5.377 (sec)

(iv) When #4 = 2.7236
M#4 ... M03
G#4 ... G03

4. Value for each address should not exceed the maximum programmable value.

5. The value that follows an address may be replaced with <expression>.

6. The constant without decimal point enclosed in brackets [ ] is assumed to have a decimal point at its end.

(6) Undefined Variable

The value of an undefined variable is assumed to be "blank." An undefined variable occurs in the following situations:

a. The local variable for which argument designation was not performed in macro call command.

b. Common variables #100 through #149 at the time of power-on and reset operations.

c. The local variables and common variables for which the values were not written from MDI panel.

Designation and function of <blank> is classified in the following two versions A and B. The control is set for either version. Switching from versions A to B and from B to A cannot be interchanged.

		Version A	Version B
1	Concept of #0.	•No conception of #0. •Commanding #0 causes alarm.	•#0 defined as variables of <blank>. •Commanding #0 at the lefthand side of the equation.
2	Variable <blank> is commanded in the replacement equation.	•Where #2 is <blank>, command #3 = #2. means #3 = 0.	•Where #2 is <blank>, command #3 = #2; means #3 = <blank>.
3	Variable <blank> is commanded in the part program.	•Where #2 is <blank>, command 600 × #2; is equivalent to command G00; G00 × 0;	•Where #2 is <blank>, command 600 × #2; is equivalent to command G00; (Address is ignored.)
4	Variable <blank> is commanded in the condition of EQ and NE.	•Where #2 is <blank>, #3 is 0. ①Condition "IF #3 EQ #2" is established. ②Condition "IF #3 NE #2" is not established.	•Where #2 is <blank>, #3 is 0. ①Condition "IF #3 EQ #2" is established. ②Condition "IF #3 EQ #2" is eatablished.
5	Others	#3 = # [#0 + #0] — In these command, #3 = <Blank> #3 = #2 * #0; #3 = #0 + #0; #3 = #0/#0; #3 = 5 * #0; — In these commands, #3 = 0. #3 = 2 − #0; means #3 = 2 #3 = 5/# 0; causes alarm. <Blank> in the replacement described above is treated as "0." •Condition IF #3 GE #2 is established when #2 and #3 are <blank>, or #2 is 0 and #3 is <blank>. •Condition IF #3 LT #2 is not established when #2 and #3 are <blank>, or #2 is <blank>, and #3 = 0.	

2.8.23.5 Operation Commands

Various operations can be performed between variables and between variables and constants. The operation expression is represented in the form of #i = <expression>, in which <expression> is a general arithmetic operational expression produced by combining variables and constants with operators and functions. The available opearations and functions are as follows. Instead of #j and #k, constants may be used.

(1) Variable Difinition and Replacement

#i = #j ··· definition, replacement.

(2) Add-Type Operations

#i = #j + #k ··· Sum.
#i = #j − #k ··· Difference.
#i = #j OR #k ··· Logical sum (for each of 32 bits).
#i = #j XOR #k ··· Exclusive logical sum (for each of 32 bits).

(3) Multiply-Type Operations

#i = #j * #k ··· Product.
#i = #j / #k ··· Quotient.
#i = #j AND #k ··· Logical product (for each of 32 bits).

Note: In OR, XOR, or AND operation, the variable value (or constant) is converted into the binary 32-bit equivalent and the operation is performed on each bit.

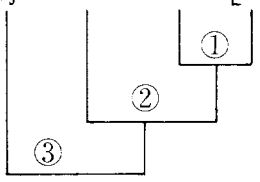
(4) Functions

#i = SIN [#j] ··· Sine (in degrees).
#i = COS [#j] ··· Cosine (in degrees).
#i = TAN [#j] ··· Tangent (in degrees).
#i = ATAN [#j/#k] ··· Arctangent.
#i = SQRT [#j] ··· Square root.
#i = ABS [#j] ··· Absolute value.
#i = BIN [#j] ··· Convert from BCD.
#i = BCD [#j] ··· Convert into BCD.
#i = ROUND [#j] ··· Produce integer by rounding.
#i = FIX [#j] ··· Truncate the fractions.
#i = FUP [#j] ··· Raise the fractions to a unit.

(5) Combinations of Operations

The above operations and functions may be used in combinations. A functional operation is performed first. Then, a multiply-type operation is performed. An add-type operation is performed last.

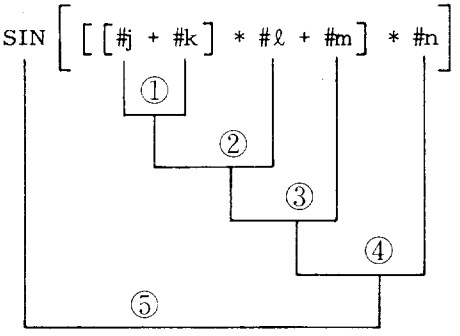
Sample #i = #j + #k * SIN [#ℓ]



(6) Change of Operational Order by [ ]

Priority may be given to an operation by enclosing it in brackets [ ]. Up to quintuple (five-hold) nesting of brackets is permitted including those of functional operations.

Sample #i = SIN [[[#j + #k] * #ℓ + #m] * #n]



(7) Considerations for Operational Commands

a. The constant without decimal point used in <expression> is assumed to have a decimal point at its end.

b. When used in conditional Expression IF or WHILE, function ROUND truncates the fractions.

c. When used in address data, function ROUND rounds off the part below the least significant digit.

Sample (a)

#10 = 12.3758

When the least significant digit of address X is 0.001 mm, the following command

G00 X [ROUND [#10]];

means

G00 X12.376 ;

because 8 of 12.3758 is rounded.

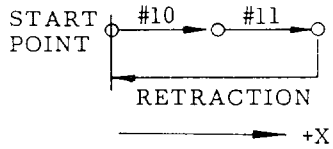
This command is also equivalent to

G00 X#10 ;

Usually, ROUND is not used as mentioned above; it is used as shown below:

Sample (b)

When ROUND is used as follows, the program returns to the start point correctly:



① #10 = 12.3758 ;
② #11 = 13.1236 ;
③ G00 U#10 ;
④ G01 U#11 F··· ;
⑤ G00 U- [ROUND [#10] + ROUND [#11]] ;

This is because the data of #10 and #11 in ③ and ④ blocks are substantially rounded before being executed.

If ⑤ block is

⑤ G00 U- [#10 + #11] ;

then, the movement is made by the following amount:

U- [#10 + #11] = U - [12.3758 + 13.1236]
= U - [25.4994]
= U - [25.499]

On the other hand, block movement of ③+④is

U#10 + U#11 = U12.376 + U13.124
= U25.500

Hence, the program of ⑤ is not correct.

(8) Operational Errors

The data format and the operational errors in the macro programs are as follows:

a. Data Format

The numeric data handled in macro programs are of the floating point format.

M * 2E

where, M is sign + data 52-bit binary,
E is sign + data 10-bit binary.

b. Operational Errors

Each time an operation is performed, the following error is caused and is accumulated. The number of significant digits is 15 to 16, which compensates the error sufficiently.

2.8.23.6 Control Commands

The commands which control the flow of the micro-program are of the following two types:

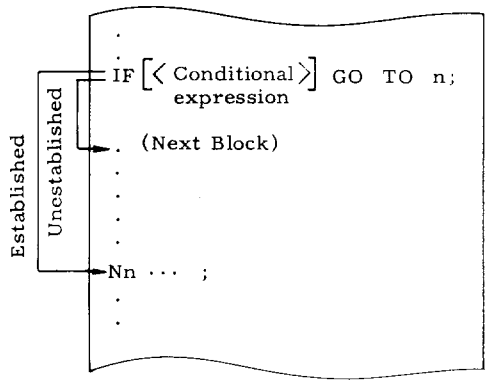
a. Branch command ... IF [< conditional expression>] GO TO n ;

b. Repeat Command ... WHILE [< conditional expression >] DO m ,

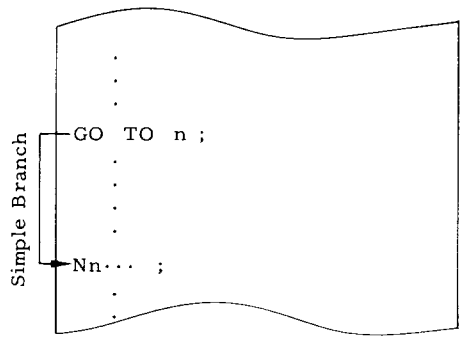
(1) Branch Command

a. IF [< conditional expression >] GO TO n ;

If < conditional expression> of this command is established, a branch is made to the block of sequence number n within the same program. When a variable or an expression is used for n, the branch destination may be changed. If the condition is not satisfied, the program proceeds to the next block.



IF [< conditional expression >] may be omitted to provide a simple branch command as shown below:



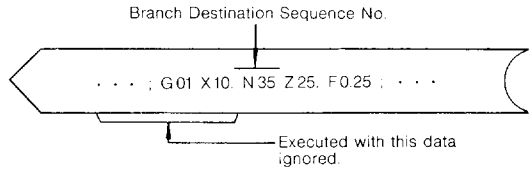
Conditional expressions are EQ, NE, GT, LT, GE, and LE. They are represented as follows:

Conditional Expression	Meaning
#i EQ #j	(#i = #j)
#i NE #j	(#i ≠ #j)
#i GT #j	(#i > #j)
#i LT #j	(#i < #j)
#i GE #j	(#i ≧ #j)
#i LE #j	(#i ≦ #j)

A constant and < expression> may be used for #i and #j. A variable and <expression > may be used for n.

Notes:

1. The sequence number must be located at the head of the block when it is called for by a branch command. Otherwise, the data prior to the sequence number is ignored as shown below:



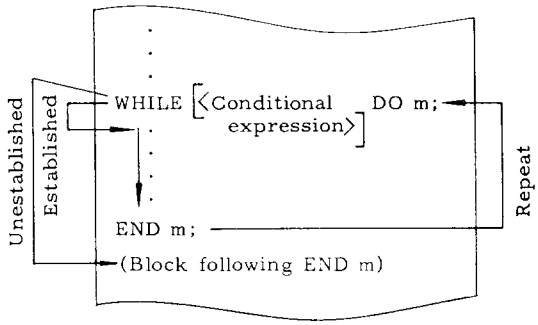
2. The reverse branch on the program takes longer execution time than the forward branch.

(2) Repeat Command

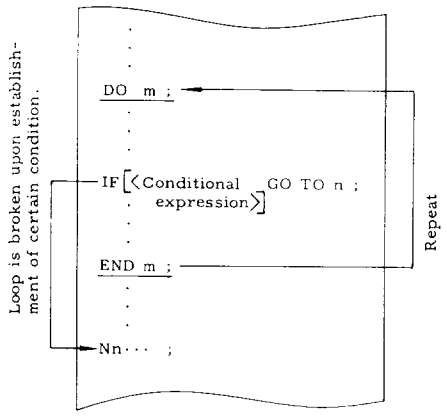
a. WHILE [< conditional expression >] DO m ; (m = 1, 2 and 3)

END m ;

While <conditional expression> is satisfied, the blocks between DO m and END m are repeated. When it is unsatisfied, the processing branches to the block following END m.



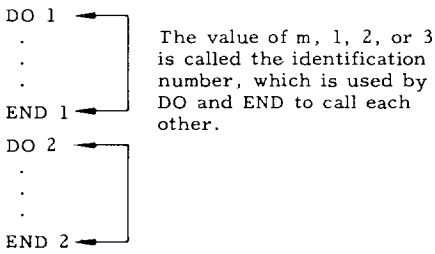
When the specification is made omitting WHILE [<conditional expression>], the blocks between DO m and END m are repeated infinitely. Generally, this is used in the format shown below.



Notes:

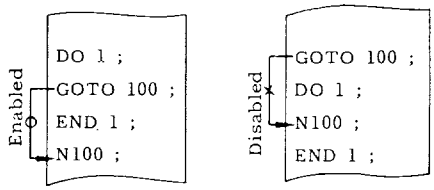
1. DO m should be specified before END m.

2. m of DO m and END m should have the same value. However, only 1, 2, or 3 may be specified in m.

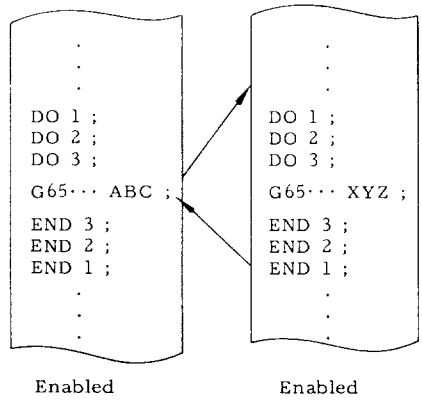


3. The same identification number may be used repeatedly except where repeat ranges overlap.

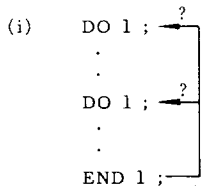
4. To get out of a DO loop, a GO TO n can be used. However, a GO TO n does not enable entrance to a DO loop as shown below:

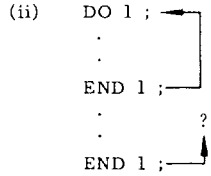


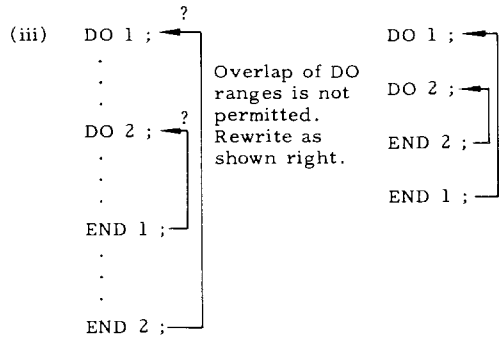
5. Triple DO-loop nesting is permitted for each micro program.



6. The codings shown below cause an error:







(iv)

```
DO 1 ;
.
.
DO 2 ;
.
.
DO 3 ;
.
.
DO 1 ;  <-- Quadruple nesting. Max. nesting permitted is triple.
.
.
.
END 1 ;
.
.
END 3 ;
.
.
END 2 ;
.
.
END 1 ;
```

(v)

```
DO 1 ;
.
.
.
N7000··· ;  <-- DO loop may not be entered from outside.
.
.
.
END 1 ;
.
.
.
IF···GO TO 7000 ;
```

2.8.23.7 Registration of Macro Programs

(1) How to Make Registration of Macro Programs

The registration and edit of macro program bodies are performed in the same manner as usual part programs and subprograms. Hence, there is no program size restriction that applies to the user macro body. Part programs, subprograms, and macro programs may be stored together in the part program memory to its full capacity.

(2) Classification of Program Numbers

The program numbers are classified into the following:

Program No.	Classification
O8000 to O8999	When D4 of #6004 is set to 1, the registration, erase, and edit of programs are disabled.
O9000 to O9999	When D7 of #6021 is set to 1, the registration, erase, and edit of programs are disabled.

2.8.23.8 Display and Write of Local and Common Variables

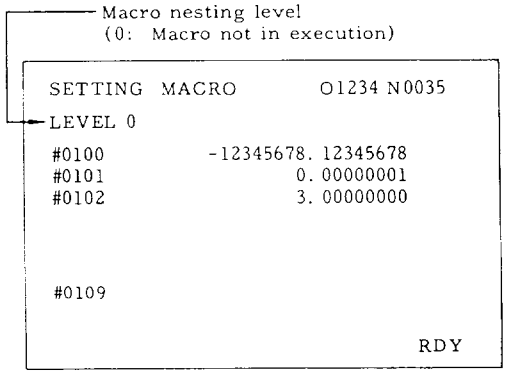
Local variables (#1 through #33) and common variables (#100 through #149, #500 through #549) can be displayed and written by the following operations:

(1) Display Operations

Display of Variables

a. Press SET function key. Mode select position may be provided anywhere.

b. Key-in the variable number and press CURSOR ↓ key or CURSOR ↑ key. However, # need not be keyed in. Ten sets of variable numbers including the specified variable number and their data are displayed. The data are displayed in the signed 8-digit integer part and the 8-digit fraction part.



Sample Display of Common Variables

2.8.23.8 Display And Write Of Local And Common Variables (Cont'd)

c. Press [PAGE ↓] key or [↑ PAGE] key, and the display may be scrolled up or down.

Remarks

a. Common variables may always be displayed for review.

b. For local variables, those of the macro currently executed are displayed. Consequently, when a macro of a nesting level is in execution, the local variables belonging to macros of the other nesting levels cannot be seen. The local variables after completion of execution are all rest to "blank."

(2) Write Operations

Writing of Values to Variables

a. Press SET function key. Mode select position may be provided anywhere.

b. Key-in the variable number to be written press [CURSOR ↓] key or [↑ CURSOR] key. However, # need not be keyed-in. The keyed-in variable number is specified and the cursor is positioned to it.

c. Key-in the value to be written. Press WR key. The keyed-in value is stored as the data of the variable number with the cursor positioned.

d. Press [CURSOR ↓] key or [↑ CURSOR] key or [↑ PAGE] key or [PAGE ↓] key to move the cursor.

e. Repeat operations in c. and d. to write the values to the desired variables.

Remarks

a. Common variables can always be changed.

b. Local variables may not be written at any time other than when a macro is in execution. Any attempt to do so is invalidated. However, rewriting of local variables during macro execution may cause an unexpected failure. Before attempting the rewriting, stop the machine operation by single stop function and check to see if it is safe to rewrite variables.

c. The written local variables and common variables #100 through #149 are reset to "blank" by the reset operation or the power-on operation.

2.8.23.9 Considerations and Remarks for Macro Programs

(1) Summary of Restrictions

a. Available Variables

#1 through #33 ··· Local variables.

#100 through #149 ┐
#500 through #549 ┘ Common variables.

System variables

b. Available Variable Values

Maximum value ··· $\pm 10^{+308}$

Minimum value ··· $\pm 10^{-308}$

c. Constant Values Usable in < Expression >

±(8 digits above decimal point). (7 digits below decimal points).

Sample Maximum value ±99999999.9999999

Minimum value ±0.0000001

d. Operational Accuracy

Decimal 15 digits significant.

e. Macro Call Maximum Nesting Level

Quadruple (four-hold).

f. Maximum Nesting Level of Repeat Command

Triple (three-hold) for each macro.

g. Repeat Command (DO) Identifier m

m = 1, 2, and 3.

h. Maximum Nesting Level of Brackets

Quintuple (five-hold).

(2) Difference between Macro Program and Subprogram.

a. User macros G65 and G66 allow argument designation but the subprogram (M98) does not.

b. The macro program directly branches to the macro program body without executing any command that was specified in G65 or G66 block and has no relationship with the macro. With the subprogram, however, a branch is performed after the execution of the command (if any) other than P and L in M98 block.

c. The maximum nesting level of macro program is quadruple including G65 and G66 calls. That of subprograms is also quadruple but separately.

d. If macro programs are specified via MDI during automatic operation, the maximum nesting level is restricted to quardruple. With subprograms, up to four levels of nesting are permitted in tape mode or memory mode, or separately in MDI mode.

(3) Relationship with MDI Operation

a. MDI writing permits the macro call and the execution of the called macro.

b. MDI writing does not permit or execute macro body commands such as operational commands and control commands.

c. When a macro program being executed is stopped by the single block stop function, any MDI writing command not related to the macro may be specified and executed.

(4) Relationship with Address Search

The address search function is not permitted to search for the sequence numbers in the macro body.

(5) Relationship with Single Block Switch

a. The operational command and control command blocks do not single-block stop if the single block switch is turned on. This switch is enabled for the other macro program blocks.

b. However, when setting number #6004D_1 = 1, the single block switch is enabled for the operational command and control command.

c. System variable #3003 (for the control of single block stop see 2.8.23.5) and setting #6004 D_1 mentioned above operate as shown below:

Setting #6004	System Variable #3003	When Single Block Switch is on
D1=0	=1 or 3	None of the operational commands, control commands, and general commands stop.
D1=0	=0 or 2	Operational commands and control command do not stop. General commands stop.
D1=1	=1 or 3	None of the operational command, control commands, and general commands stop.
D1=1	=0 or 2	All of the operational commands, control commands, and general commands stop.

(6) Relationship with Optional Block Skip

The slash "/" character used in the right-hand side of an operational expression or in brackets is assumed to be the operator for quotient. It does not mean the optional skip.

(7) Parameter Setting of Program Number Classification

(1) Disabling of Program Registration, Erase, And Edit

The following setting is permitted to protect the registered user macros and subprograms from inadvertent destruction:

Setting
Number

#6004

D2 = 1 ··· The programs of program numbers #8000 through #8999 are disabled for registration, erase, and edit.

D2 = 0 ··· Registration, erase, and edit are enabled.

Parameter
Number

#6021

D7 = 1 ··· The programs of program numbers #9000 through #9999 are disabled for registration, erase, and edit.

D7 = 0 ··· Registration, erase, and edit are enabled.

(8) Effects of Reset Operation

a. A reset operation resets all local variables (#1 through #33) and part of common variables (#100 through #149) to "blank."

b. A reset operation resets the user-macro multiple call state and the multiple DO loop state, making the program pointer return to the program head.

2.8.23.9 Considerations and Remarks for Macro Programs (Cont'd)

(9) Special Codes Usable in Macro Program Body

(1) The special codes listed below may be used in the user macro body:

	Code	Use	EIA Code 8	7	6	5	4	◦	3	2	1	ISO Code 8	7	6	5	4	◦	3	2	1
	SP	For comment				○		◦				○		○			◦			
*	(	For alarm message comment				○	○	◦		○				○		○	◦			
*	)			○			○	◦		○		○		○		○	◦			○
	+	Add		○	○	○		◦						○		○	◦		○	○
	−	Subtract		○				◦						○		○	◦	○		○
	:	For comment		○				◦	○	○				○	○	○	◦		○	
	/	Divide			○	○		◦			○			○		○	◦	○	○	○
	#	Variable	Parameter designation											○			◦		○	○
*	*	Multiply					○	◦	○	○		○		○		○	◦		○	
*	=	Equal			○	○	○	◦		○	○	○		○	○	○	◦	○		○
*	[	Bracket (open)				○	○	◦	○			○	○		○	○	◦		○	○
*	]	Bracket (close)			○		○	◦	○			○	○		○	○	◦	○		○
	$	For comment	○			○		◦	○					○			◦	○		
	@		○				○	◦	○	○	○	○	○				◦			
	?		○		○	○		◦	○	○				○	○	○	◦	○	○	○
	.	Decimal point		○	○		○	◦		○	○			○		○	◦	○	○	

Notes:

1. For the hole pattern of EIA code, when the character is attached with an asterisk, the pattern shown above is standard. However, other patterns may be specified by using the following parameters:

#6110 ··· [
#6111 ···]
#6112 ··· *
#6113 ··· =
#6114 ··· (
#6115 ···)

Read the desired hole pattern in the binary value, convert it into the decimal equivalent, and set it to the parameter. For example, the hole pattern shown below is set as "152":

8	7	6	5	4	0	3	2	1
○			○	○	◦			

When the value of the parameter is "0," the hole pattern listed in the above table is provided.

2. When the codes shown below are output from the NC unit for punch-out or other purposes, the upper code (UC) or lower code (LC) is output immediately before.

a. Codes preceded by UC ··· #, +, $, ?.

b. Code preceded by LC ··· @.

c. Codes preceded by UC only at parameter designation ... (,), *, =.

2.8.23.9.1 Outline of User Macro External Output Command

The following macro commands are available in addition to the standard user macro commands.

(1) Open command (POPEN)
(2) Data output command (BPRNT or DPRNT)
(3) Close command (PCLOS)

The above commands are used to output variables and characters via external devices with the RS-232C interface.

2.8.23.9.2 Details of Each Command

(1) Open command (POPEN)

(Format) POPEN [a] ;

[a] → RS-232C port number

(Function) The DC2 control code is output from the NC side. Give this command before the sequential data output commands. The RS-232C number can be specified as 1 or 2.
The default is the 1st RS-232C.

(Example) POPEN; ..... Opens the 1st RS-232C.
POPEN [2] ; .... Opens the 2nd RS-232C.

(2) Data output command (BPRNT or DPRNT)

(a) BPRNT

(Format)

BPRNT [a #b [c] ....] ;

[c] → Effective digits below decimal point
#b → Variable
[a → Character

(Functions)

1 The commanded characters are directly output in ISO code.
- Alphabets (A to Z)
- Numerals
- Special characters (*, /, +, -)

Can be output. However, '*' is output by the space code.

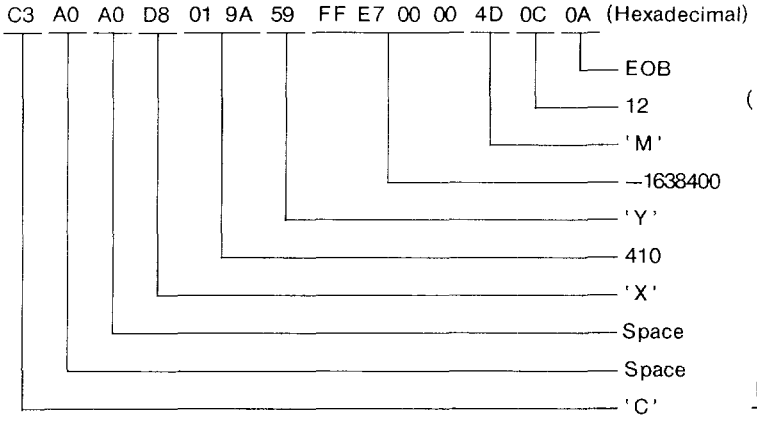
2 The value of the variables is regarded as 2-word data (32 bits) considering the digits below the decimal point, and is output from the higher order bytes, directly as binary data.

The effective digits after the decimal point are commanded in parentheses after the variable command.

3 After outputting the command data, the EOB code is output in ISO code.

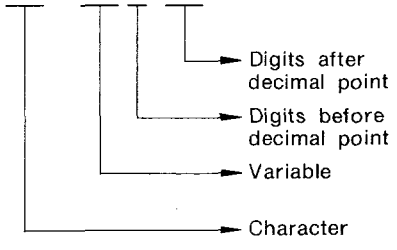
(Example) BPRNT [C ** X #100 [3] Y #101 [3] M #10 [0];

When the variables are;
#100 = 0.40956
#101 = -1638.4
#10 = 12.34
it is output as;



(b) DPRNT

(Format) DPRWT [a # b [c d]·····] ;



(Functions)

1 The commanded characters are directly output in ISO code, as in the (i) BPRNT command.

2 The variable values are output from the higher order digit, digit-by-digit, for as many digits as commanded. The decimal point is also output in ISO code.

The variable value is output by commanding the variable number after "#". Then the digits before the decimal point, and the digits after the decimal point are each commanded in parentheses.
Assume that the variables have a maximum of 8 digits. When the higher order digit is 0, the output follows parameter #6016D7.

[Parameter #6016D7 = 0: Output space code.
#6016D7 = 1: No output.]

When the digits after the decimal point of the commanded variable is other than 0, as many digits after the decimal point as commanded, are output. When the commanded digits after the decimal point is 0, no decimal point is output.

When the sign is positive, the output of the space code follows parameter #6016D7.

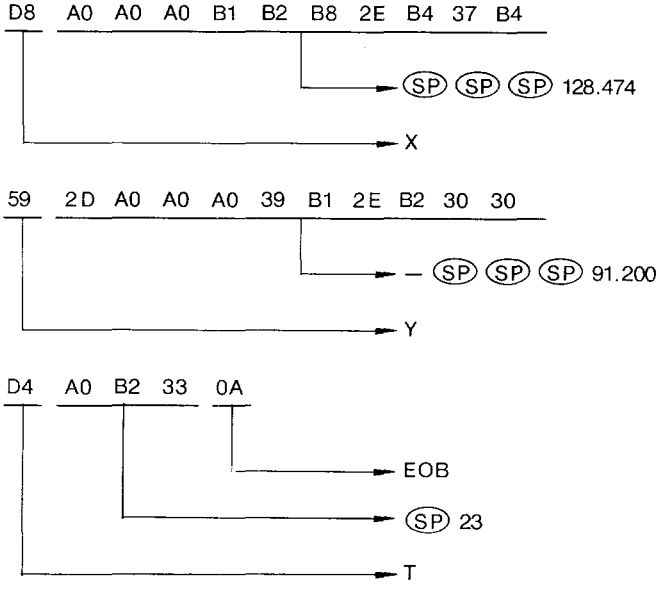
[Parameter #6016D7 = 0: Output space code.
#6016D7 = 1: No output.]

After outputting the command data, the EOB code is output in ISO code.

(Example) DPRNT [X#2[53]Y#5[53]T#30[20]];
When;
Variable value #2 = 128.47398
#5 = 91.2
#30 = 123.456

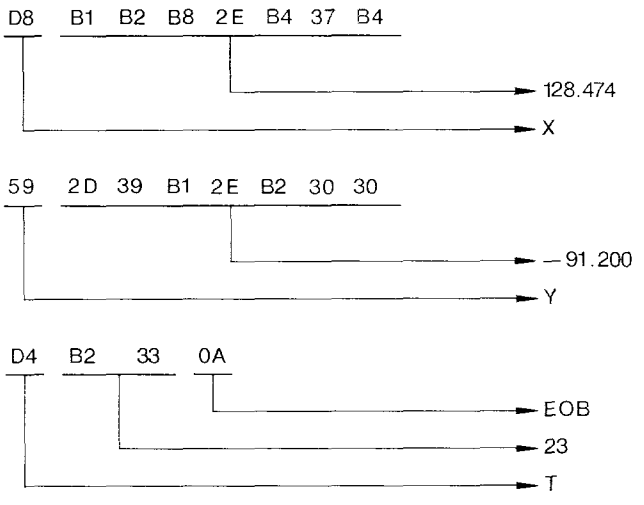
And;

1 When parameter #6016D7 = 0



2.8.23.9 Considerations and Remarks for Macro Programs (Cont'd)

2 When parameter #6016D7 = 1



(3) Close command (PCLOS)

(Format) PCLOS [a];

RS-232C port number

(Function) The DC4 control code is output from the NC side.
This is commanded after all the data output command is given.
The RS-232C number is given in the same manner as the (1) POPEN command.

(Example) PCLOS; .... Closes the 1st RS-232C.

2.8.23.9.3 Setting Necessary when Using This Function

(1) When outputting data by the BPRNT, DPRNT command, set #6022D3 = 0 (no parity bit upon ISO tape output).

If #6022D3 = 1, the data cannot be output correctly.

(2) When outputting data by the DPRNT command, set whether the leading zero is to be output by spaces.

#6016D7 = 0: Output space as leading zero when outputting data by the DPRNT command.
= 1: No leading zero is output.

2.8.23.9.4 Notes

(1) The open command (POPEN) and close command (PCLOS) do not need to be commanded in sequence.
Once the open command is given, there is no need to give another open command until the close command is given.

(2) The output data output command can be stopped by resetting the command. All later data will also be erased.

Therefore, if reset is commanded by M30, etc. At the end of the output program, it is necessary to give the close command at the end of the program, wait until all the data are output, and then command M30, etc.

(3) User macro and I/O interface option is required to use this function.

2.8.23.10 Alarm Number Of Macro Programs

Shown below are the user-macro-associated alarms and their causes.

105 MACRO ERROR (CONSTANT)

The number of constants is in excess of the specified range.

106 MACRO ERROR

There are too many G67 cancel codes.

107 MACRO ERROR (FORMAT)

A format other than expression has an error.

108 MACRO ERROR (UNDEFIN #NO)

The value not defined as a variable number is designated.

109 MACRO ERROR (#NO NOT LEFT)

The variable of assignment statement is the one that is disabled for assignment.

110 MACRO ERROR ([ ] 5 LIMIT)

The bracket nesting level is in excess of the upper limit (5).

111 MACRO ERROR (MOVE G66 - M99)

A move command is specified in the macro end command M99 called by G66.

112 MACRO ERROR (5)

The macro call nesting level is in excess of the upper limit (4).

113 ———

114 MACRO ERROR (DO FORMAT)

DO and END are not paired.

115 MACRO ERROR ([ ]UNMATCH)

The format of <expression> has an error.

116 MACRO ERROR (DO - END NO.)

DO m is not in the range of $1 \leq m \leq 3$.

117 ———

118 MACRO ERROR (GO TO N)

GO TO n is not in the range of $0 \leq n \leq 9999$.

2.8.23.11 Exercises of Macro Programs

(1) Canned Cycle by G92

T (Teacher): We have discussed many complicated rules you have to understand to write user macros. Now, let's create some user macros as exercises. Let's take straight thread-cutting cycle by G92, because it is a simple operation.

S (Student): Where shall we start?

T: An example of usual G92 command takes the following format:

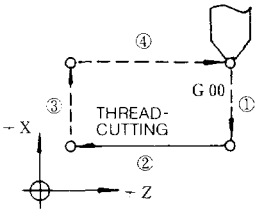
(P1)

```
G92  U-50.  W-60.  F6.0  :
```

This command is divided into the following and executed within the NC unit. It is assumed that Rapid Pull Out of Threading is not included in this command.

(P2)

```
① G00 U-50. ;
② G32 W-60. F6.0 ;
③ G00 U50. ;
④ G00 W60. ;
```



First, these moving distances and lead threads can all be converted into variables.

S: They are local variable #1 through #33, aren't they? But which type of local variable?

T: Type I for small number of variables. This type allows the use of U, W and F and therefore makes the argument designation easier to understand.

S: OK. When type I is used, we have the following variables:

(P3)

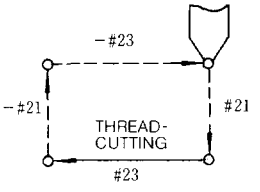
```
U-50.  W-60.  F6.0
 |      |      |
#21    #23    #9
```

T: Right. Using these variables, rewrite the former program (P2).

S: OK.

(P4)

```
① G00 U#21 ;
② G32 W#23 F#9 ;
③ G00 U-#21 ;
④ G00 W-#23 ;
```



Is this all right?

T: Yes. Just add this and we have a complete user macro body.

```
⑤ M99 ;
```

S: That's easy.

T: Then, using G65, create this macro call and the macro program body in the complete formats.

S: Let me try it.

Supposing the program No. of macro program body is O9093; the macro call command will be:

(P5)

```
G65  P9093  U-50.  W-60.  F6.0 ;
```

The macro program body is:

(P6)

```
O9093 ;
G00 U#21 ;
G32 W#23 F#9 ;
G00 U-#21
G00 W-#23 ;
M99 ;
```

T: That looks OK.

****

S: I think something is wrong. With this program, I have to specify points W and F every time!

T: That's true. With a usual canned cycle, when points W and F have been specified once, their values are retained. Thereafter, only U is specified.

S: Do you have any trick to overcome this inconvenience?

T: I do. In such a case, common variables (#100 - #549) help. Using common variables, write the macro to designate the position of points W and F.

S: I've got it! Now, I divide the macro body into two parts as follows:

(P8)

```
O9000 ;
#100 = #23 ;
#101 = #9 ;
M99 ;
```

(P9)

```
O9093 ;
G00 U#21 ;
G32 W#100 F#101 ;
G00 U-#21 ;
G00 W-#100 ;
M99 ;
```

and I write the macro call as follows:

(P10)

```
G65 P9000 W-60. F6.0 ;
G65 P9093 U-50. ;
G65 P9093 U-51.4 ;
G65 P9093 U-52.6 ;
G65 P9093 U... ;
  .
  .
  .
```

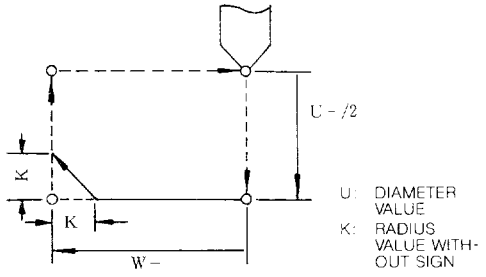
T: Very good.

****

S: I'd like to try to program Rapid threading pull-out.

T: OK. How about designating the width of rapid threading pull-out using address K?

S: All right. Let's see .....



Macro call is as follows:

```
G65 P9000 W-60. K4.8 F6.0 ;
G65 P9093 U-50. ;
G65 P9093 U-51.4 ;
G65 P9093 U-... ;
  .
  .
  .
```

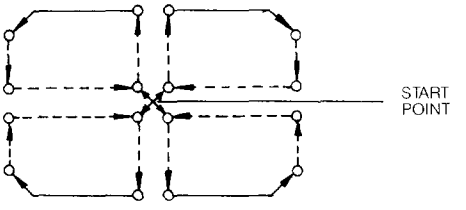
Macro body is as follows:

```
O9000 ;
#100 = #23 ;
#101 = #9 ;
#102 = ABS [ #6];
M99 ;
```

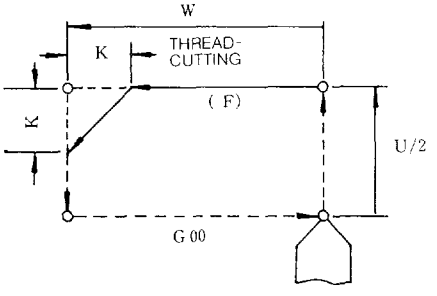
```
O9093 ;
#10 = ROUND [ #102] *2 ;
#11 = ROUND [ #21] + #10 ;
#12 = ROUND [ #100] + ROUND [ #102] ;
G000 U#21 ;
G32 W#12 F#101 ;
G32 U#10 W-#102 ;  <-- RAPID THREADING PULL-OUT
G00 U-#11 ;
G00 W-#100 ;
M99
```

Is this OK?

T: Yes. Your reasoning is right. Practically, you had better prevent a malfunction by programming #3003: invalid control of single block or #3004: invalid control of feedhold. This threadcutting can be performed in U- and W-directions only. Now we'd like to expand this function in four directions.



S: I see. Let me think.
--- How about the next program?



U, K: Designation with a sign
K: Designation without a sign

U = #21 (DIAMETER VALUE)
W = #23
K = #6 (RADIUS VALUE)
F = #9

The macro call command is as follows:

```
G65 P9000 W-45. K4.0 F5.0 ;
G65 P9093 U40. ;
G65 P9093 U41.4 ;
G65 P9093 U... ;
```

The macro program body is as follows:

```
O9000 ;
#100 = #23 ;<------ W
#101 = #9 ; <------ F
#102 = ABS [#6] ;<---- |K|
M99 ;
```

```
O9093 ;
#3003 = 1 ;<------ Single block invalid
M93 ; <---------- 4-block buffering
#10 = ROUND [#102] *2 ;
IF [ABS][#21] LT #10] GO TO 4 ;
IF [ #21 GT 0 ] GO TO 1 ;
IF [ #21 EQ 0 ] GO TO 4 ;
#11 = ROUND [#21] + #10 ;<--U: Negative
#12 = #10 ;
GO TO 2 ;
N1 #11 = ROUND [#21] -#11 ;<--U: Positive
#12 = -#10 ;
N2 #13 = ROUND [#102] ;
IF [ABS [#100] LT #13] GO TO 4 ;
IF [ #100 GT 0 ]G0T0 3 ;
IF [ #100 EQ 0 ]G0T0 4;
#14 = ROUND [#100] + #13 ;<--W: Negative
#15 = -#13 ;
GO TO 5 ;
N3 #14 = ROUND [#100] -#13 ;<--W: Positive
#15 = #13
```

```
GO TO 5 ;
N4 #3000 = 499 (MACRO INPUT ERR.) :
                    |<---Error display
N5 G00 U#21 ;
#3004 = 7 ;<---[ Feedhold
               | Feedrate override  ] Invalid
               | Positioning
               [ completion
G32 W#14 F#101 ;
G32 U#12 W#15 ;<-[ Rapid threading
                 [ pull-out
#3004 = 0 ;
G00 U-#11 ;
G00 W-#100 ;
M92 ;
#3003 = 0 ;
M99 ;
```

T: Well. If U or W = 0, and |U/2| or |W| < K, error will be displayed in your programming. That's good.

2.8.24 PROGRAM MIRROR IMAGE (G68, G69)†

Program mirror image is the feature to reverse the NC program operation in all directions around the work center line (Z-axis) by the use of G command.

(1) G68 ; ··· Program mirror image on.
The program mirror image on state is held until G69 is specified.

G69 ; ··· Program mirror image off.
The program mirror image off state is held until G68 is specified.

When program mirror image is on, the X-axis operation by the NC program is inverted with Z-axis being the center line. The manual operations (manual continuous feed and handle/step feed) are not affected by this feature.

(2) Details of program mirror image

When the PROGRAM mirror image feature is on, the movement by the NC program is inverted with Z-axis being the center line. The following inversion is processed in the NC unit:

a. X command for X-axis coordinate value is inverted.

b. U command for X-axis incremental coordinate value is inverted.

c. I command for X-axis coordinate value of arc center is inverted.

d. Circular motion direction inverted.

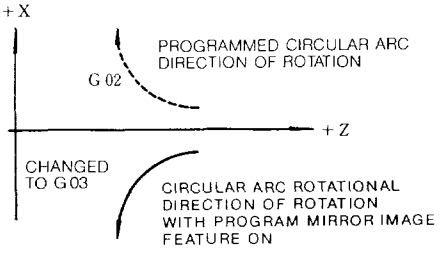


Fig. 2.39

e. I command for X-axis beveling/rounding volume and direction is inverted.

f. I command for canned cycle taper X-axis distance is inverted.

g. U and I commands for special canned cycles finishing allowance, etc. are inverted.

h. The operational direction after X-axis tool position compensation is inverted.

i. T command for tool nose radius compensation tool nose center is inverted in the sign.

j. G command for tool nose radius compensation virtual tool nose position is inverted.

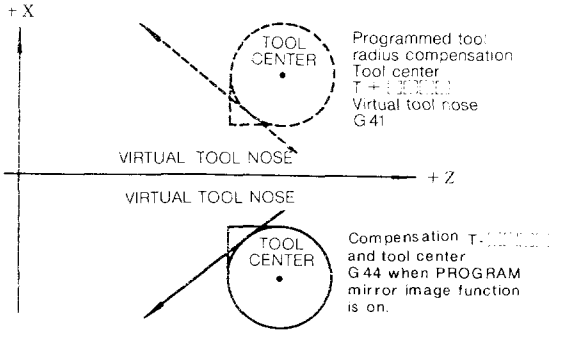


Fig. 2.40

(3) Cautions for G68 and G69 commands

a. G68 and G69 are modal G commands which belong to "10" group. They must be specified on a single block basis, in principle.

b. G69 (program mirror image off) is used at the time of power-on, reset operation, and program reset.

c. These commands must be specified in the tool nose radius compensation cancelled state.

d. These commands may not be specified in the finishing shape program of the special canned cycle.

e. If automatic origin return "G28 X··· Z··· ;" is specified when the X-axis mirror image feature is on, the positioning of the intermediate point specified in X and Z is affected by the mirror image, but the machine origin, which is an absolute position, is not affected by this feature.

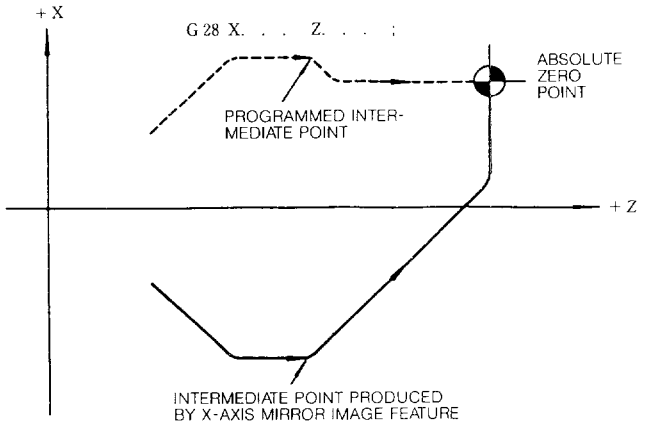


Fig. 2.41

EXAMPLE:

Described below is a sample program which uses G68 and G69 for the opposed tool rest shown in the diagram on the following page, and the X-axis movements.

Power on

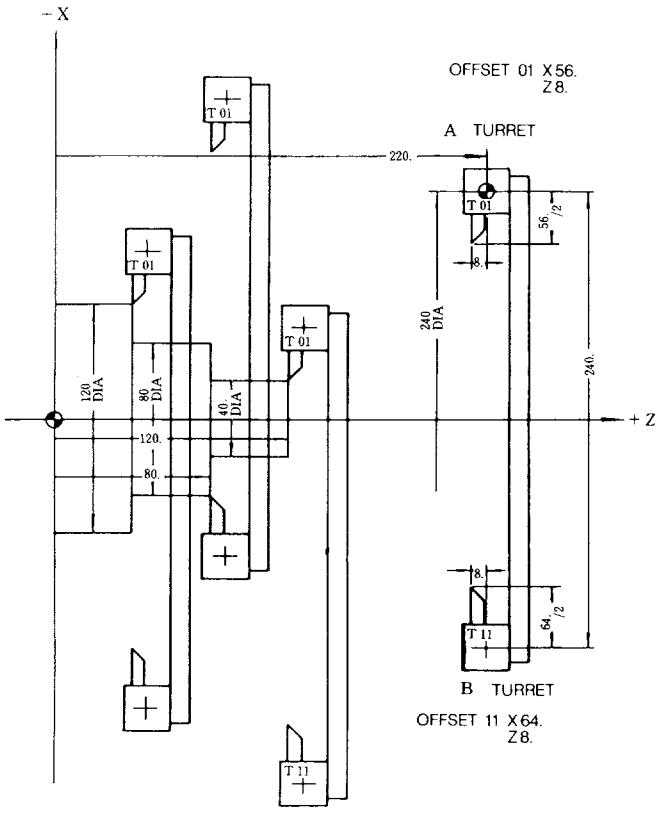
↓

Manual reference point return

↓

Program execution

	X-axis command value display	X-axis movement
N0001 G50 X240. Z220. ;	240.000	
N0002 X40. Z120. T0101 ;	96.000	U-144.000
N0003 G68 ;	-96.000	
N0004 G50 U480. ;	+384.000	
N0005 X80. Z80. T1111 ;	+144.000	U240.000
N0006 G69 ;	-144.000	
N0007 G50 U480. ;	336.000	
N0008 X120. Z40. T0101 ;	176.000	U-160.000



2.8.25 MULTIPLE REPETITIVE CYCLES (G70 TO G76)†

2.8.25.1 General

This option makes program simple and short. For instance, both stock removal and finishing are performed only by commanding the finishing work shape.

Table 2.21 Multiple Repetitive Cycles

G code	Name	Remarks	
G70	Finishing cycle		
G71	Stock removal in turning	Finishing by G70 possible	Tip nose radius compensation possible
G72	Stock removal in facing		
G73	Pattern repeating		
G74	Peck drilling in Z axis	Tip nose radius compensation impossible.	
G75	Grooving in X axis		
G76	Automatic threadcutting		

(1) G70 through G76 are in * group and non-modal.

(2) The program of finishing shape specified by G71, G72 and G73 are stored in memory. The memory capacity for the finishing shape is 45 blocks.

Program of finishing shape ≦ 45 blocks

Note: When cornering (G11, G12) and multiple cornering (G111, G112) are used, each block containing them must be counted as the value listed below.

	No. of blocks
One block including G11 or G12	Two blocks
One block including G111	Four blocks
One block including G112	Five blocks

(3) The internal memory for storing the finishing shape program:

To shorten the stock removal cycle computation time, the finishing shape program is binary-converted and then is stored in the memory for storing finished shape program (one pair) in the unit. This memory is called the internal memory for finishing shape program, which differs from the part program memory.

(4) In the block after the cycle of G70 through G76, the G codes of 01 group should be specified again. This is because the 01-group G codes specified before the cycle may have been changed to other G codes by the execution of this cycle.

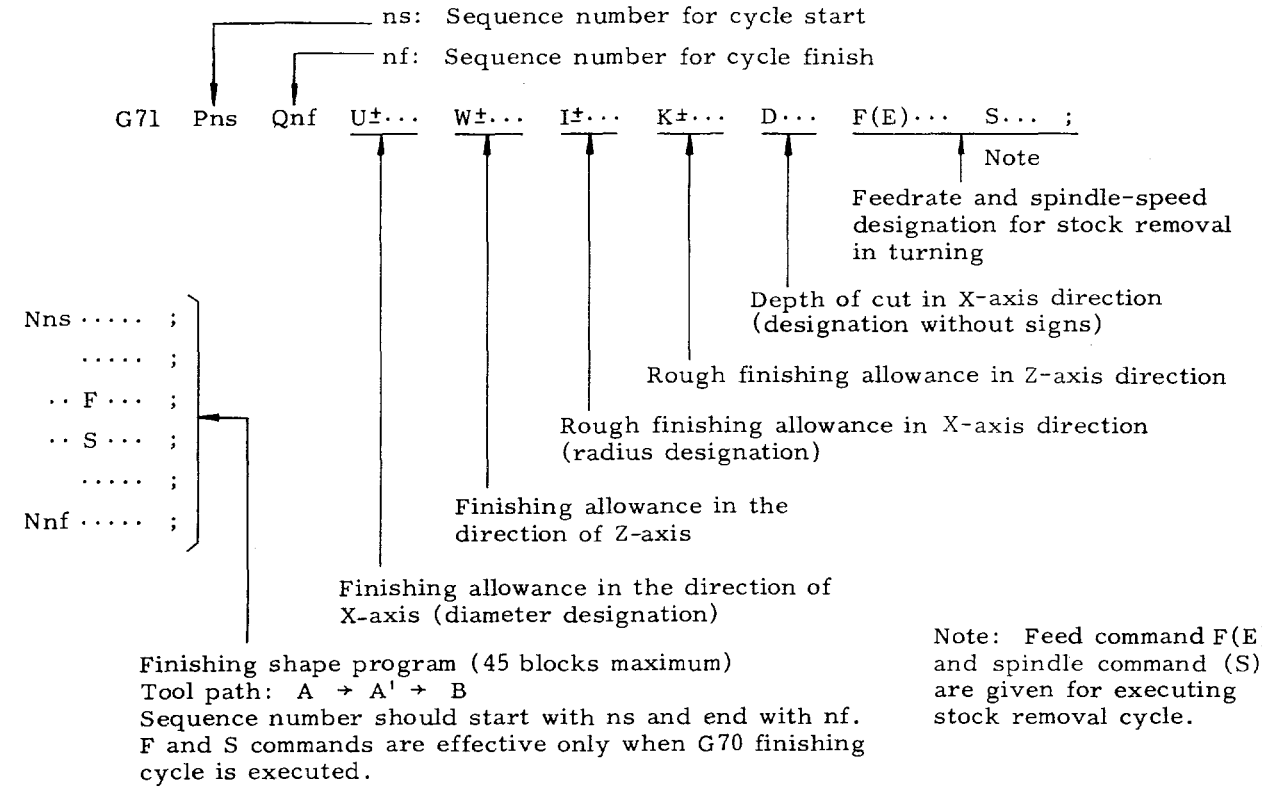
(5) It is possible to perform tool nose radius compensation on the cycle of G70 through G73.

(6) Tool nose radius compensation cannot be performed on the cycle of G74 through G76. Any attempt to do so will result in an error.

2.8.25.2 Stock Removal in Turning (G71)

Stock removal in turning with the finishing allowance remained uncut can be commanded by G71. Commands for finished contour are different between monotonous increase/decrease and concaved shaped path.

(1) Monotonous increase/decrease finishing shape



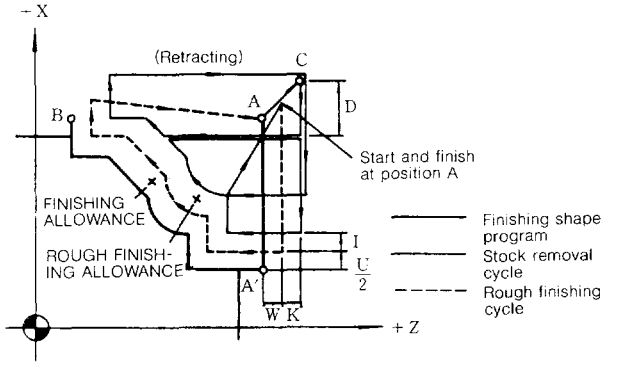


Fig. 2.42

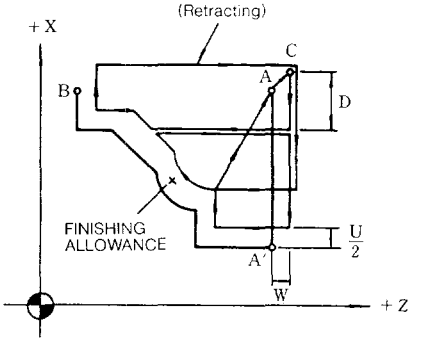


Fig. 2.43

G71 starts at point A, executes rough finish cycle (——) and the rough finishing cycle (---), and returns to point A to be terminated.

In the case of I = 0 and K = 0 (or no designation), the rough finishing cycle is omitted.

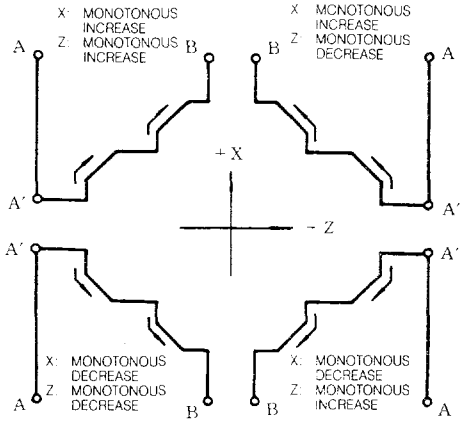
Retracting is performed by rapid traverse G00. Thrust motion depends on the speed (G00 or G01) specified by the program of AA'.

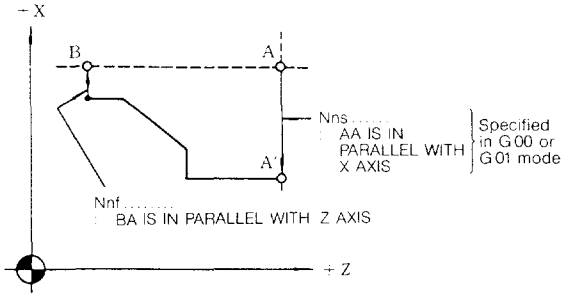
Each depth of cut D along X-axis can be overrided by 10% step within the range of 0 to 200%, by G71/G72 cut depth override selection or setting. See item (3), g.

Notes :

1. The tool path of finishing shape should be programmed to be monotonous increase or decrease in X and Z coordinates.

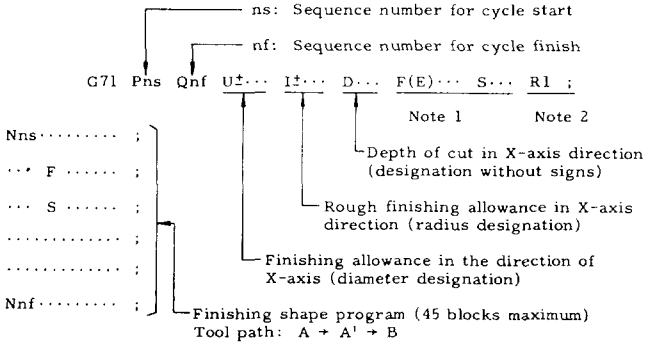
2. The following should be taken into consideration in programming the start block (Nns) and the end block (Nnf) of a finishing shape program.





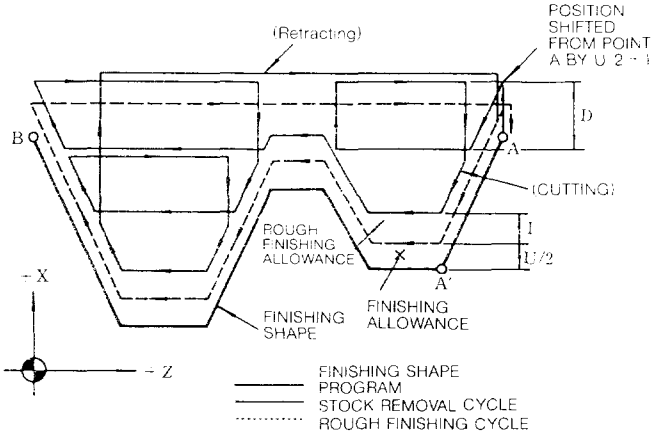
(2) Concaved finishing contour

a. Command format



Note 1: Specifies the feed command (F(E)) and spindle command (S) for the execution of stock removal cycle.

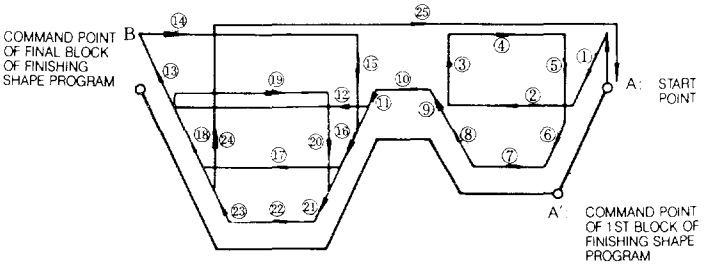
Note 2: Computes the cutting path for the concaved finishing shape program if R1 is specified.



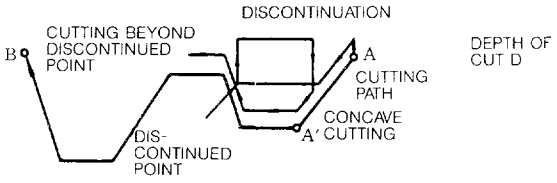
G71 starts at point A, executes the rough cutting cycle (——) and the rough finishing cycle (---), and returns to point A to be terminated. If I is not specified, the rough finishing cycle is skipped.
Retracting is performed by rapid traverse G00. Thrust motion depends on the speed (G00 or G01) specified by the program of AA'. Each depth of cut D along X-axis can be overrided by 10% step within the range of 0 to 200%, by G71/G72 cut depth override selection or setting. See item (3), g.

b. Cautions for concaved finishing shape program

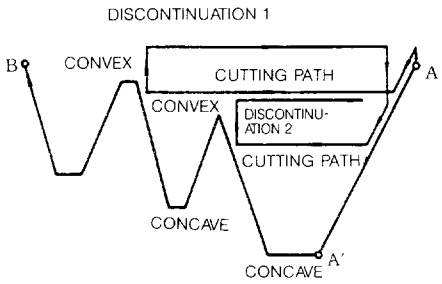
(i) Rough cutting cycle by G71 starts from the closest concave to the start point.



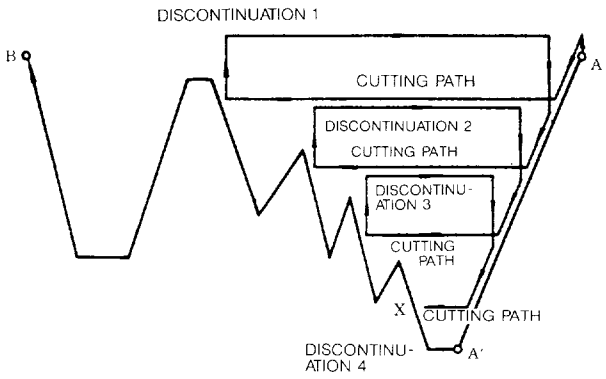
Since cutting starts with the concave nearest the start point, the cutting path is interrupted if it hits the convex beyond the concave. The concave is cut to its bottom. Then, the cutting cycle returns to the interrupted point to perform cutting beyond it.



(ii) For a simple concave, only one interrupted point is provided. However, for a complex concave containing a smaller concave and a convex as shown below, the cutting path is interrupted first at the larger convex then at the smaller convex.



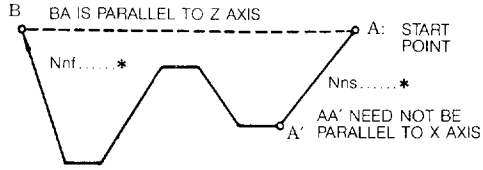
The maximum number of interrupted points that allows cutting is three, beyond which cutting is disabled, causing "097" error. Within this limitation, any number of concaves is allowed.



Sample Contour Not Allowing Cutting :ing

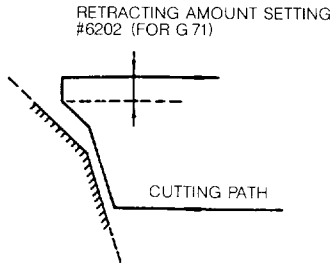
(iii) Any contour having an overhang does not allow cutting. Hence, the Z-axis specification value of the finishing shape program should be a monotonous variation.

(iv) The termination block for the finishing shape program has the following limitations:



For the G command of the termination block (Nnf··· ;), specify G01 or G00, in principal.

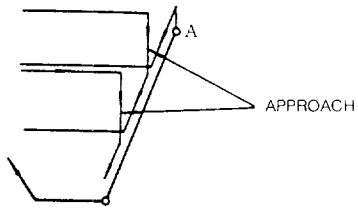
(v) The retracting amount after each approach cycle may be set by the setting.



(vi) Each block of the finishing shape program should be of monotonous increase or monotonous decrease. A circular arc which extends over two or more quadrants must be divided into two blocks before being programmed.

(vii) Generally, Z-axis finishing allowances W and K are not specified. Otherwise, a bite is caused into the wall of the corresponding side. If "R1" is not specified, the conventional monotonous increase/monotonous decrease stock removal cycle is provided.

(viii) Approach is performed at the feedrate. It is not affected by the G code of the finishing shape program. Hence, depending on the finishing shape program, positioning may be performed by rapid traverse after the approach at feedrate.



(3) Rules in programming G71

a. Addresses U, W, I and K must be programmed with signs. If a wrong sign is programmed, the workpiece may be gouged. An address D for depth of cut must be programmed without signs.

b. Finishing shape program must be programmed immediately after the block containing G71. Even a block between them is ignored.

c. When F and S codes are not specified in the block containing G71, F and S codes specified in the preceding block are effective for G71 mode.
F and S codes specified in the program of finishing shape become effective only for G70 mode and are disregarded in G71 mode.

d. The following should be taken into consideration in programming the start block (Nns) and the end block (Nnf) of a finishing shape program.

Usable G code	Remarks
G00, G01, G03, G06, G22, G23	–
G11, G12	A block containing these codes must be counted as two blocks.
G111	A block containing these codes must be counted as four blocks.
G112	A block containing these codes must be counted as five blocks.

e. When a program has entered the tool nose radius compensation mode before the G71 is commanded, the compensation is effective for the G71 cycle.

However, the compensation is executed not in the stock removal cycle but in the rough finishing cycle.
Thus, the compensation is ineffective for the program in which the rough finishing cycle is omitted. (I = 0, K = 0)

f. The above rules and cautions in programming G71 also apply to G72 cycle. In other words, the G72 cycle is the same as G71 except that cutting is made in parallel with X-axis.

g. Cut depth override of G71 and G72

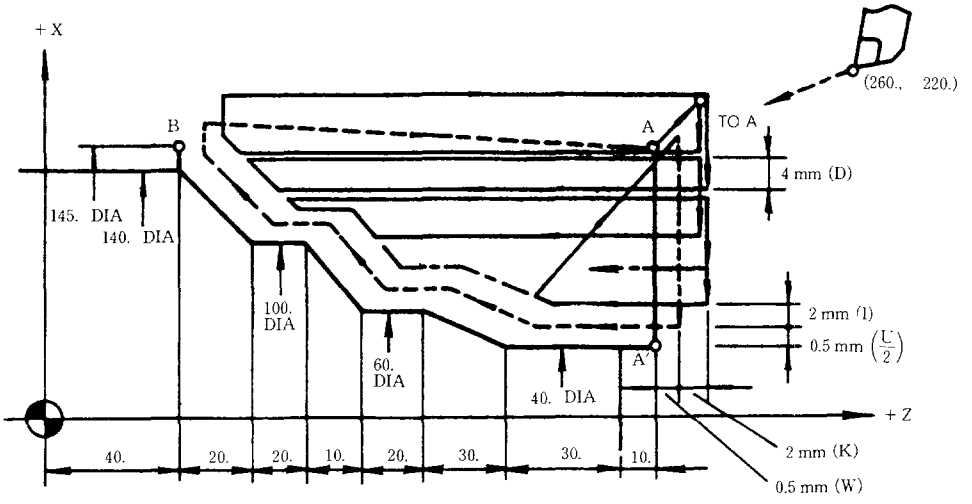
Ten percent step override may be applied, within the range of 0 to 200%, to the depth of cut D of each time in the following two manners:

(i) By setting #6004D3 through D7. (Set with 5-bit code.)

(ii) By G71/G72 cut depth override switching. Either of the above methods shown in (i) or (ii) may be selected by parameter #6023D2.

Parameter	Function
#6023 $D_2 = 0$	Setting in (i)
#6023 $D_2 = 0$	Setting in (ii)

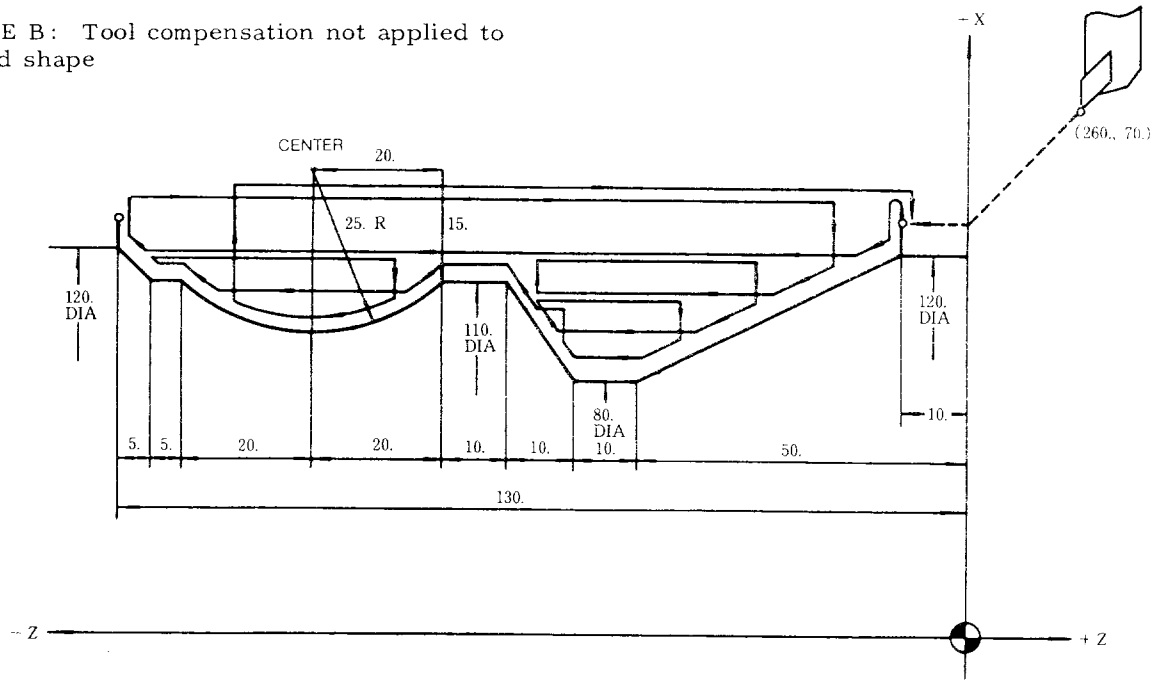
EXAMPLE A: Tool nose compensation applied to finishing shape without concaves




```
N1 G50 X260. Z220. ;
N2 G00 S1000 M03 T0101 ;
N3 G41 ;
N4 X145. Z180. ;                    ← Stock removal in turning
N5 G71 P6 Q13 U1. W0.5 I2. K2. D4. F0.3 S800 ;
N6 G00 X40. S800 ; ··· Rapid traverse
                       cutting
N7 G01 W-40. F0.15 ;
N8 X60. W-30. S600 ;
N9 G12 W-20. I5. ; ··· Equivalent to
                       2 blocks
N10 G01 X100. W-10. S300 ;
N11 W-20. ;
N12 X140. W-20. S200 ;
N13 X145. ;
N14 G40
N15 G00 X260. Z220. T0100 ;
```

Finishing shape = 9 blocks

EXAMPLE B: Tool compensation not applied to concaved shape




```
N01  G50 X260. Z70. ;
N02  G00 S500 M03 T0101 ;
N03  X124. Z-10. ;                          ← Stock removal in facing
N04  G71 P5 Q14 U2. D6. F0.2 S250 R1 ;
N05  G01 X120. ;
N06  X80. Z-50. F0.1 S500 ;
N07  W-10. ;
N08  X 110. W-10. ;                         } Finishing shape
N09  W-10. ;
N10  G02 X90. W-20. I15. K-20. ;
N11  X110. W-20. I25. ;
N12  G01 W-5. ;
N13  X120. W-5. ;
N14  X124. ;
N15  G00 X260. Z70. T0100 ;
N16  T0202 ;
N17  G50 X255. Z70. ;
N18  X124. Z-10. ;
N19  G70 P5 Q14 ;                           ← Executes finishing cycle of G71 shown above.
```

2.8.25.3 Stock Removal in Facing (G72)

This cycle provides stock removal and rough finishing in facing with the finishing allowance remaining

G71 is for cutting in parallel with Z axis and G72 is for cutting in parallel with X axis.

(1) Monotonous increase/decrease finishing shape

a. Command format

G72 Pns Qnf U±... W±... I±... K±... D... F(E)... S... ;

Nns ..... ;
..... ;
F ..... ;
S ..... ;
..... ;
Nnf ..... ;

Depth of cut in Z axis direction (without signs)

Significance of each address is the same as that of G71 command.

Finishing shape program (45 blocks maximum)
Tool path: A → A' → B

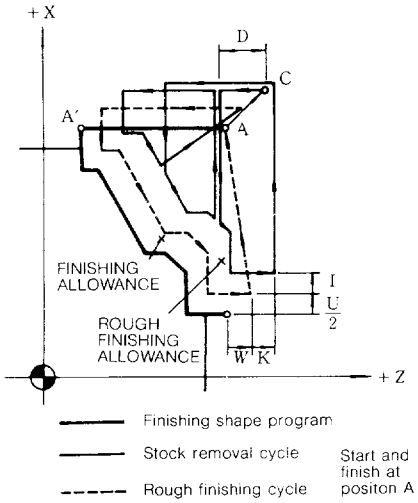


Fig. 2.44

Cycle starts at point A executes stock removal cycle and rough finishing cycle, returns to A when completed.

In case of I = 0 and K = 0 (or no designation), the rough finishing cycle is omitted.

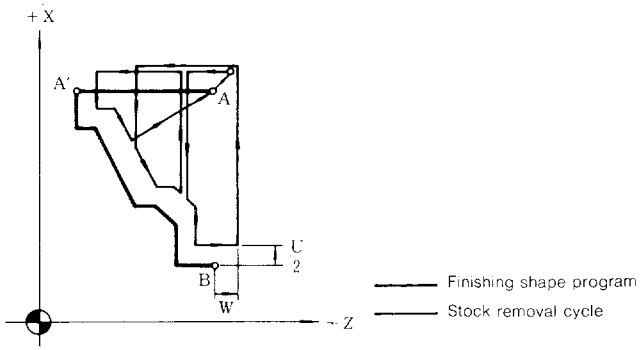


Fig. 2.45

Retracting is performed by rapid traverse G00. Thrust motion depends on the speed (G00 or G01) specified by the program of AA'. Each depth of cut D along Z-axis can be overrided by 10% step within the range of 0 to 200%, by G71/G72 cut depth override selection or setting.

b. Rules and cautions in programming finish shape

G72 is the same as G71 except that the tool cuts into the workpiece in parallel with Z axis. Refer to 2.8.25.2 Stock Removal in Turning.

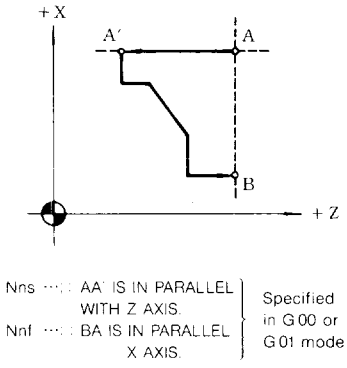
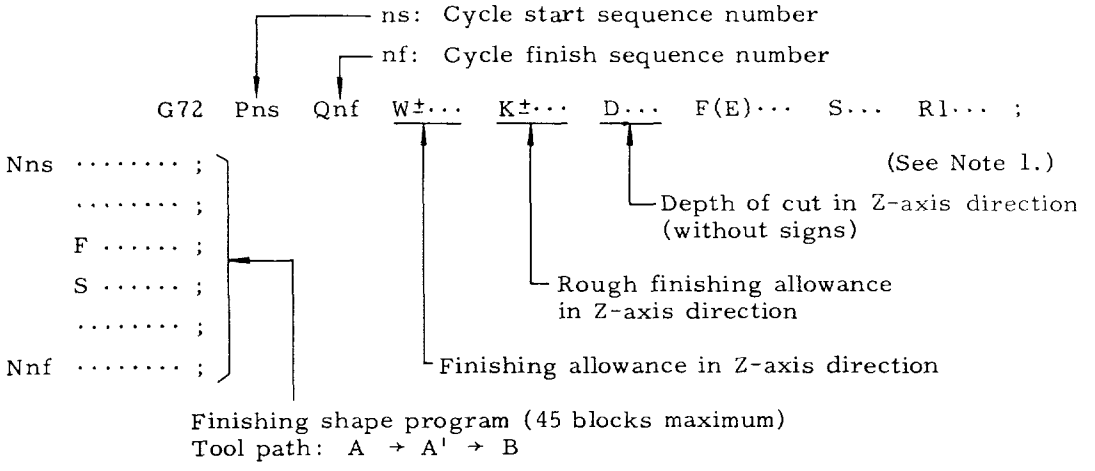
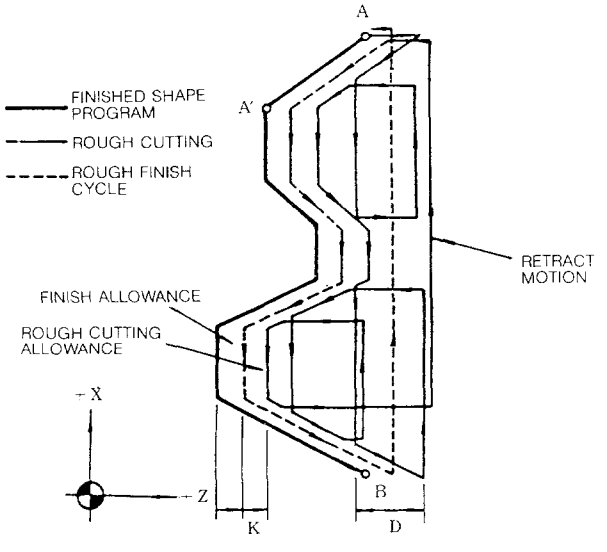


Fig. 2.46

(2) Concaved finishing shape



Note 1: If "R1" is specified, cutting path is computed for the concaved finishing shape program.



G72 starts at point A, performs stock removal cycle (——) and rough finishing cycle (---), and returns to point A when completed. If K is not specified, rough finishing cycle is skipped. Retracting is performed by rapid traverse G00. Cutting rate depends on the speed (G00 or G01) specified by the program of AA'.
Each depth of cut D along X-axis can be overrided by 10% step within the range of 0 to 200%, by G71/G72 cut depth override selection or setting.

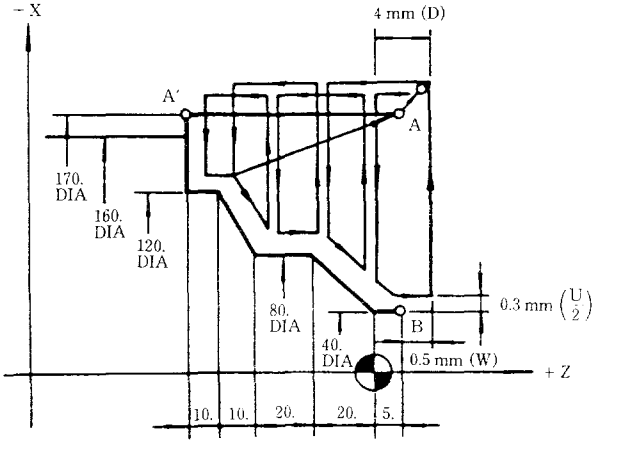
b. Cautions of concaved finishing shape program

G72 only performs the operations which are parallel to Z axis. The same cautions as with G71 are applied to G72 except that the retracting amount of G72 may be set by setting #6203.

(3) Cautions for G72 command

The same cautions as with G71 are applied to G72.

EXAMPLE: In case of I = K = 0, tool nose radius compensation is not applied.



```
N1  G50 X260. Z60. ;
N2  G00 S1000 M03 T0202 ;
N3  X170. Z5. ;                    ┌── Stock removal in facing
N4  G72 P5 O11 U0.6 W0.5 I0 K0 D4.0 F0.3 S200 ;
N5  G01 Z-60. F0.15 ; ····· Cutting at feed
N6  X120. S250 ;
N7  Z-50. ;
N8  X80. Z-40. S400 ;
N9  Z-20. ;
N10 X40. Z0 S800 ;
N11 Z5. ;
N12 G00 X260. Z60.
N13 T0303 ;
N14 X170. Z5. ;
N15 G70 P5 O11 ; ····· Executes finishing cycle
```

Finishing shape program

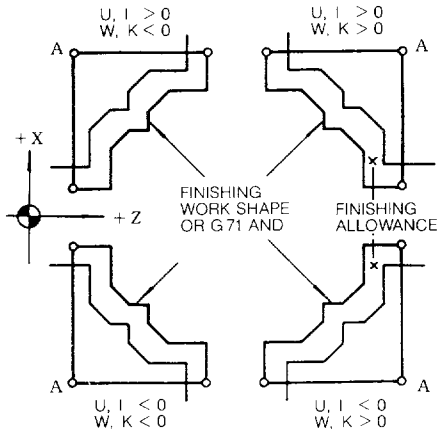
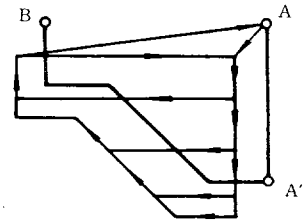


Fig. 2.47 Relation between Finishing Shape Program and Signs of Addresses U, W, I and K

A wrong sign will cause a grouging of the workpiece as shown below.

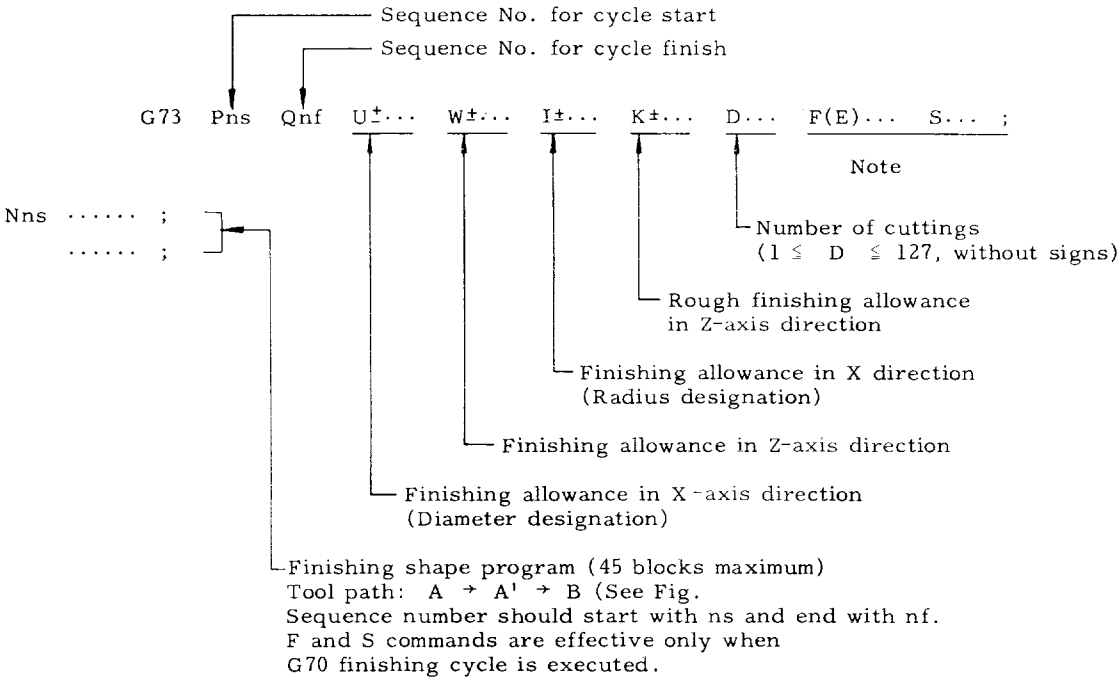


In the case that U, W, I and K < 0 are erroneously programmed.

2.8.25.4 Pattern Repeating (G73)

This cycle is useful for cutting the workpiece such as moldings and forgings whose cutting shapes are roughly made beforehand.

(1) Command format



Note: Feed command (F(E)) and spindle speed command(S) are given for executing closed loop cutting cycle.

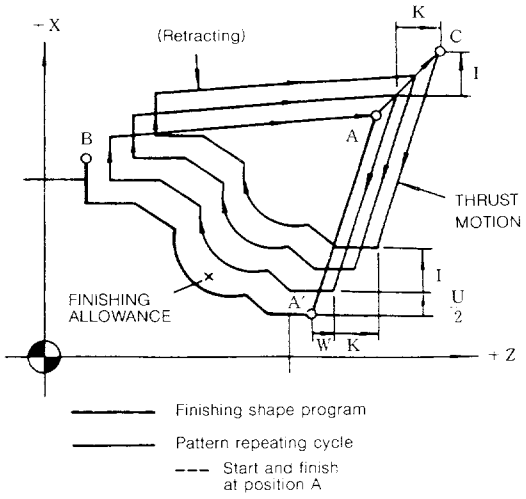


Fig. 2.48

(2) Rules in programming G73

a. Address U, W, I and K must be programmed with signs.

b. Address D for number of cuttings must be programmed without signs, obeying the following restriction.

$$1 \leq D \leq 127$$

D command out of the above range causes data error. (Alarm code "096")
When D is 1, the cutting of I and K values is completed in a single cycle remaining finishing allowance.

c. Finishing shape should be programmed immediately after the block containing G73.

d. The start (Nns) and end (Nnf) block of a finishing shape cycle must be programmed with G00 or G01.
But these 2 blocks need not be parallel with X or Z axis.

e. Finishing shape program does not need to be repetitive monotonous increase or decrease in X or Z coordinate.

Notes :

1. When F(E) and S functions are not specified in the block containing G73, the F (E) and S functions specified in the preceding blocks are effective in the pattern repeating cycle. F(E) and S functions specified in the program of finishing work shape are effective in finishing cycle G70 and ignored in pattern repeating cycle.
2. Table below shows the G codes which can be specified in the program excluding the blocks of Nns and Nnf.

Usable G codes	Remarks
G01, G06, G02, G03, G22, G23	–
G11, G12	Counted as 2 blocks
G111	Counted as 4 blocks
G112	Counted as 5 blocks

3. When I and K are 0 or not designated, it causes input error. (Alarm code "096" is displayed.)
4. Δ I and Δ K (rough cutting allowance per cycle) are calculated as follows.

$$\Delta I = \frac{I}{D-1}\ ,\ \Delta K = \frac{K}{D-1} \qquad \text{where} \quad D \geqq 2$$

Note that the control ignores the value below 0.001 millimeter. As a rule, the program should be made so that Δ I and Δ K are not smaller than 0.001 millimeter.

Processing of Δ I and Δ K

EXAMPLE 1

In case of I = 0.005 mm, K = 0.005 mm, D = 7

$$\Delta I = \frac{0.005}{6} = 0 \qquad \Delta I = \frac{0.005}{6} = 0$$

} Input error occurs.

EXAMPLE 2

In case of I = 0.01 mm, K = 0.01 mm, D = 7

$$\Delta I = \frac{0.01}{6} = 0.001 \text{ mm}$$

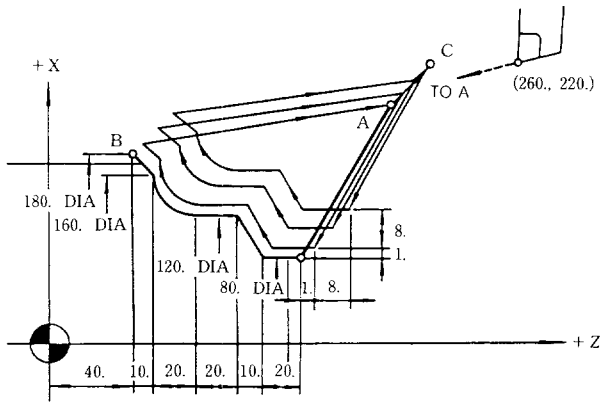
$$\Delta K = \frac{0.01}{6} = 0.001 \text{ mm}$$

Therefore, the cutting allowance of each cycle is as follows.

1st to 6th cycle ··· Δ I = Δ K = 0.001 mm
7th cycle ····· Δ I = Δ K = 0.004 mm

(6) When the program has entered the tool nose radius compensation mode before G73 is commanded, the compensation is effective for all cycles of G73.

EXAMPLE




```
N10 G50 X260. Z220. ;
N11 G00 S300 M03 T0303 ;
N12 X220. Z160. ;                    ┌ Pattern repeating
N13 G73 P14 O19 U2. W1. I8. K8. D3 F0.3 S200 ;
N14 G00 X80. W-40. S400 ;
N15 G01 W-20. F0.15
N16 X120. W-10. S300 ;                Finishing shape program
N17 W-20. ;
N18 G22 X160. W-20. R20. S200 ;
N19 G01 X180. W-10. ;
N20 G00 X260. Z220. ;
```

2.8.25.5 Finishing Cycle (G70)

After rough cutting of G71, G72 and G73, the finishing cutting can be made by the commands following G70.

(1) G70 Pns Qnf ;

nf: Sequence number for cycle finish

ns: Sequence number for cycle start

This command permits the execution of the finishing shape program in G71, G72 or G73 which is commanded previously.

F(E) and S functions specified in the finishing shape program are effective in the finishing cycle.

F(E) and S functions for rough cutting specified in the block containing G71, G72 or G73 are ignored in the finishing cycle.

Note: The internal memory for storing the finishing shape program:
To shorten the stock removal cycle time, the finishing shape program is binary-converted and then is stored in the memory (one) pair) in the unit. This memory is called the internal memory for finishing shape program.

(2) G70 does not need to be commanded immediately after the block of G71, G72 or G73. Necessary information such as tool change from a rough cutting cutter to a finishing cutter can be inserted between them.

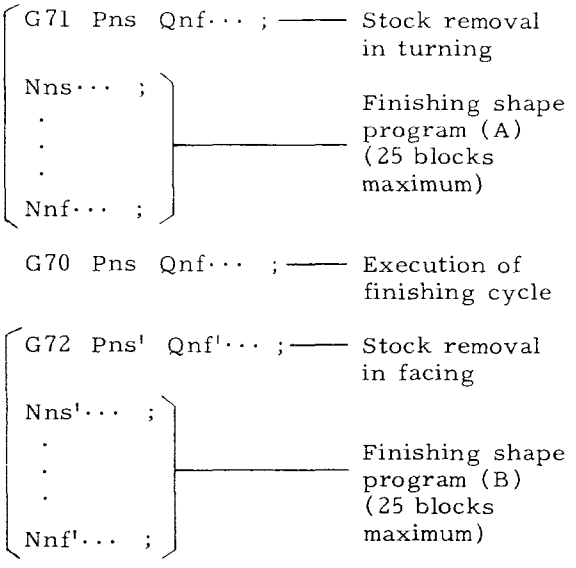
However, the following command and operation should not be programmed between them.

Inhibited Command and Operation	Result
M02 and M30 commands with internal reset	Finishing shape program in the memory are eliminated.
Reset operation	

(3) Storage and search of the finishing shape program

The processing of the finishing shape program is different in tape operation mode and memory operation mode.

a. Operation in Tape Mode



After executing the above program the program (A) is eliminated and the program (B) is retained in the finishing shape memory. Therefore, the finishing command with G70 in the trailing program is effective for the finishing shape program (B).
If the sequence number specified by G70 does not match the sequence number in the memory for finishing shape program, an error ("091") is caused.

b. Operation in memory (MEM) mode

When the sequence number specified by G70 matches the sequence number in the memory of finishing shape program, the finishing cycle is normally executed. If they do not match, a search for the specified finishing shape program from the part programs is performed. Then, the program is stored in the memory and is executed. This is called the finishing shape program search feature. This feature enables, only in memory mode, the programming in which the stock removal cycle (or closed loop cutting cycle) is performed two or more times as shown below and then the finishing cycle is performed for each stock removal cycle:

Stock removal cycle (A)
↓
Stock removal cycle (B)
↓
Finishing cycle (A)
↓
Finishing cycle (B)

Supplement:

(i) The use of the finishing shape program feature increases the cycle time as follows:

· Example of finishing shape program search time:

When the machining tape is 5 m (2000 charcters) long and the finishing shape program is stored in the middle of the tape, the cycle time is about 100 msec.

· Example of the execution time for storing the searched finishing shape program in internal memory:

When the size of the finishing shape program is 30 blocks, the execution time is about 1 sec. However, the storing of the searched program in internal memory is performed in the block which precedes the block of G70 command. Hence, the cycle time is increased by the amount obtained by subtracting the execution time of the preceding block from the above time (about 1 sec).

(ii) This search feature is executed only in the part program having the program number for which G70 is specified.

Notes :

1. For the sequence number ns for cycle start and nf for cycle finish, the following case causes input error.

When the sequence numbers ns and nf of G70 are not commanded in the finishing shape program (Tape operation)
When the sequence number ns of G70 is commanded in advance of sequence number ns, or ns = nf.

2. When the program has entered the tool nose radius compensation mode before G70 is commanded, the compensation is effective for G70.

2.8.25.6 Peck Drilling in Z-axis (G74)

This command permits the operation of peck drilling with pecking motion in parallel with Z axis.

(1) Command format

G74 X(U)±··· Z(W)±··· I··· K··· D··· F(E)··· (R1) ;

Note 1

Feedrate

Retracting amount of the tool at the cutting bottom (without signs)

Depth of cut in Z-axis direction (without signs)

Move amount in X-axis direction (without signs)

Z coordinate of position C

X coordinate of position B

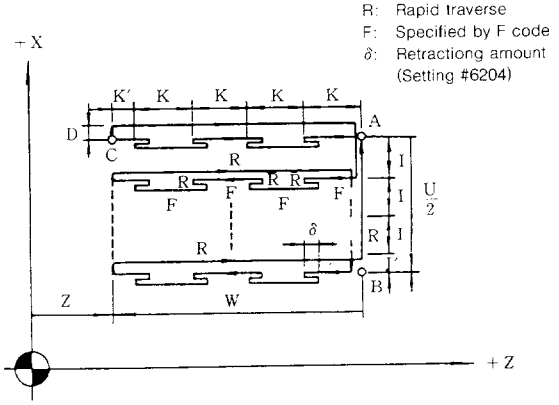


Fig. 2.49

The cycle starts and finishes at position A. The fixed amount of pecking is set by the parameter No. 74.

Note 1: The above diagram is of the cause where "R1" is not specified. When "R1" is specified, the retracting amount (δ) for each approach is ignored and the tool returns to the approach start point, namely point A in Z-axis.
This cycle starts at point A and ends at it.

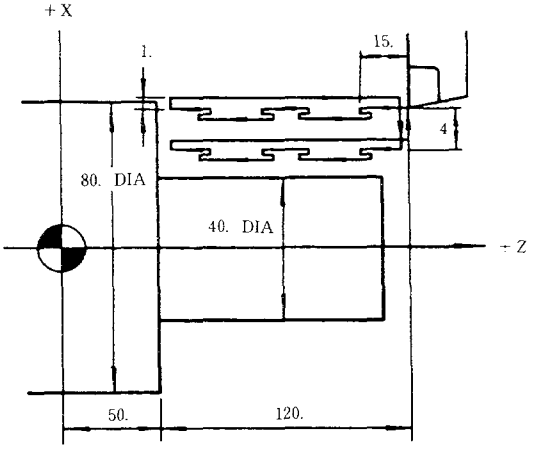
2. 8. 25. 6 Peck Drilling in Z-axis (G74) (Cont'd)

Notes :

1. Addresses I, K and D must be programmed without signs.
2. When the command of I $>|U/2|$ is issued, the cycle finishes after the finish of the pecking motion from position B following the pecking motion from position A.
3. When the command of K $>|W|$ is issued, the cutting is made at once to the cutting bottom without the pecking motion.
4. When D is programmed as 0 or D is not programmed, the retracting motion is not made at the cutting bottom.
5. The final cutting amount in the Z direction K' and the final move amount in the X direction I' are automatically calculated.
6. If X(U) , I or D is omitted, only one-cycle operation is made in the direction of Z-axis, which is used for drilling.
7. When the contents of setting #6204 are set to 0, the cutting is made at once to the cutting bottom without pecking motion.
8. The tool nose radius compensation is ineffective for G74 and G75.

EXAMPLE

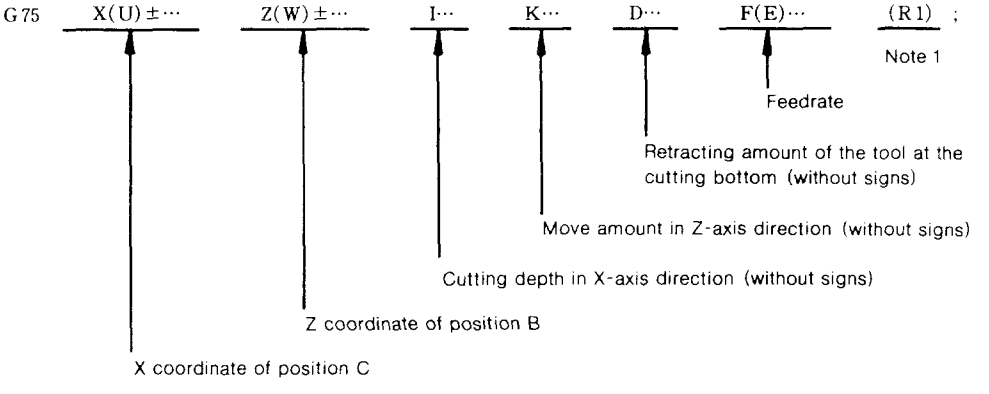
G74 X40. Z50. I4. K15. D1. F0. 25 ;



2. 8. 25. 7 Grooving in X-Axis (G75)

(1) This command permits the operation of peck drilling with pecking motion in parallel with X-axis.

a. Command format



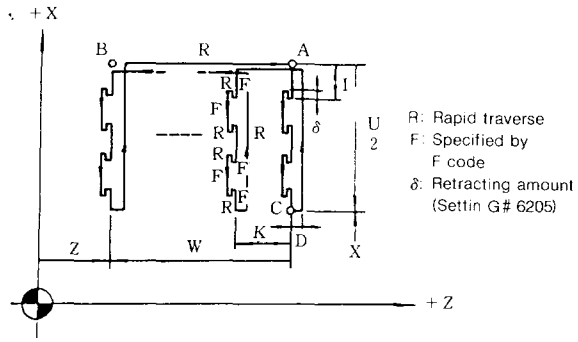


Fig. 2.50

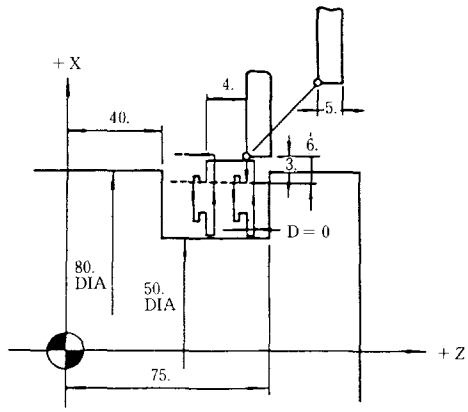
Note 1: The above diagram is of the case where "R1" is not specified. When "R1" is specified, the retractingamount (δ) for each approach is ignored and the tool returns to the approach start point, namely point A in X-axis.
This cycle starts at point A and ends at it.

Notes: G74 permits the cutting in the direction of Z-axis, and G75 in the direction of X-axis. Therefore, the cautions in programming G75 is the same as those of G74. Refer to 2.8. 25.6 Peck Drilling in Z-axis (G74).

EXAMPLE:

N1 G00 X86. Z70. ;

N2 G75 X50. Z40. I6. K4. (D0) F0.2 ;



2.8.25.8 Automatic Threading Cycle (G76)

This cycle provides automatic cutting of straight and taper threadings along the angle of thread.

(1) Command format

G76 X(U)±··· Z(W)±··· I±··· K··· D··· F(E)··· A··· ;

- A···: Angle of thread (deg)
- F(E)···: Lead of thread
- D···: First depth of cut (without signs)
- K···: Height of thread (without signs)
- I±···: Distance of tapered section (radius designation)
- Z(W)±···: Z coordinate of position C
- X(U)±···: X coordinate of position C

The sign of figure following the address I is decided by the direction of position B' viewed from position C.

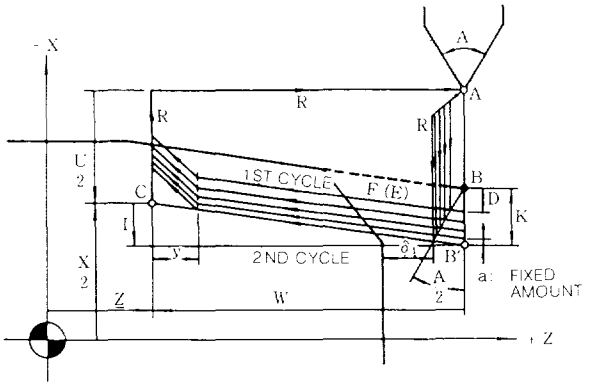


Fig. 2. 51

The following shows the cutting position around point B (In case of taper thread). The G76 cycle starts and ends at point A.

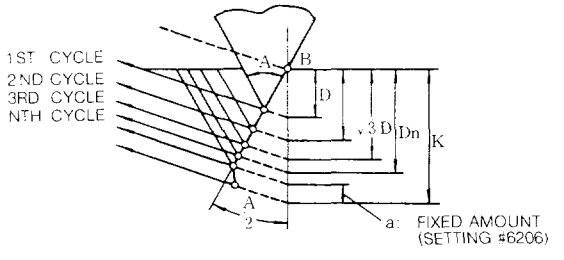


Fig. 2. 52

The depth of cut in Nth cycle is:

$$Dn = \sqrt{n}\,D$$

The following six angles can be used as the command of thread angle.

$$A = 0°,\ 29°,\ 30°,\ 55°,\ 60°,\ 80°$$

Cutting in final cycle is made with the depth of fixed amount a, which is set by the setting #6206.

(2) Straight thread

When the address I is 0 or not designated, a straight thread is cut as shown below.

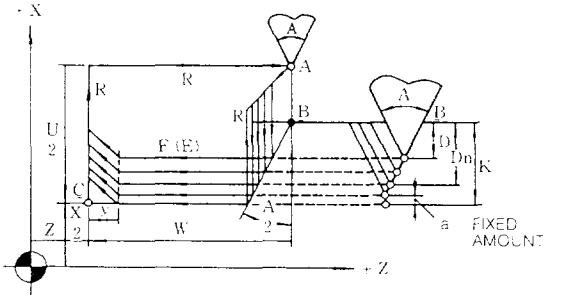


Fig. 2. 53

(3) Rapid threading pull-out

If "Rapid threading pull-out input (CDZ)" is on when G76 is specified, Rapid threading pull-out is performed. (γ) may be set to parameter #6080, in the range of 0 to 25.5L and in increments of 0.1L, where L is the lead of the specified thread.

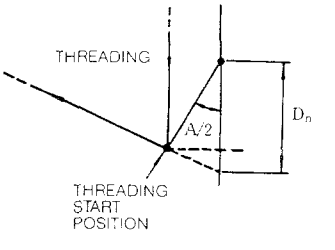
Notes :

1. The depth of cut D in the first cycle is restricted by height of thread K as follows.

$$\frac{1}{6}K \leq D < K$$

Addresses K and D must be programmed without signs.

2. When taper threading is commanded with an effective angle, except 0, X coordinate of threading start position does not extend to the depth of cut.



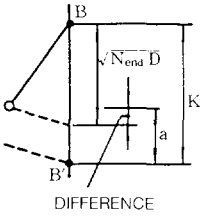
3. If the thread angle other than the above listed (0°, 29°, 30°, 55°, 60°, 80°) is arbitrarily commanded, the next larger angle is selected.

EXAMPLE

(Command) A15 ⟶ (Execution) A29

When A > 80°, A80 is executed.

4. When the depth of cut in the final cycle along the thread angle ($\sqrt{Nend}$ D) is not met with (K - a), the difference between K and a is deducted from D. The depth of cut in first cycle never becomes larger than D.



EXAMPLE

In case of D = 5.0 mm, K = 9.8 mm, a (Fixed amount) = 0.2 mm

$\sqrt{Nend}$ D = $\sqrt{4}$ x 5.000 = 10.000 mm

Difference = $\sqrt{Nend}$ D - (K - a)
= 10.000 - (9.800 - 0.200)
= 0.400 > 0

Thus, the depth of cut in each cycle is as follows.

First cycle ····· 5.000 - 0.400
= 4.600 mm

Second cycle ··· $\sqrt{2}$ x 5.000 - 0.400
= 6.671 mm

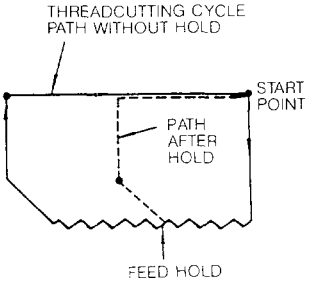
Third cycle ··· $\sqrt{3}$ x 5.000 - 0.400
= 8.260 mm

Fourth cycle ··· $\sqrt{4}$ x 5.000 - 0.400
= 9.600 mm

Fifth cycle ····· 9.600 + 0.200 (a)
= 9.800 mm

5. Thread cutting hold (option)

When this option is used, depressing the FEED HOLD during thread cutting causes the tool to perform threading up immediately and then return to start point A. When parameter #6019D7 = 1, the tool stops at position B where Rapid threading pull-out is terminated. Then, the tool returns to point A upon depression of CYCLE START.



If the thread cutting hold option is not selected, depressing of the FEED HOLD during thread cutting does not hold the operation, which stops at point C where the retracting operation is completed.

(4) Tool nose radius compensation† is not available for G76.

(5) G code of 01 group must be newly programmed in the next block of G76 cycle.

EXAMPLE

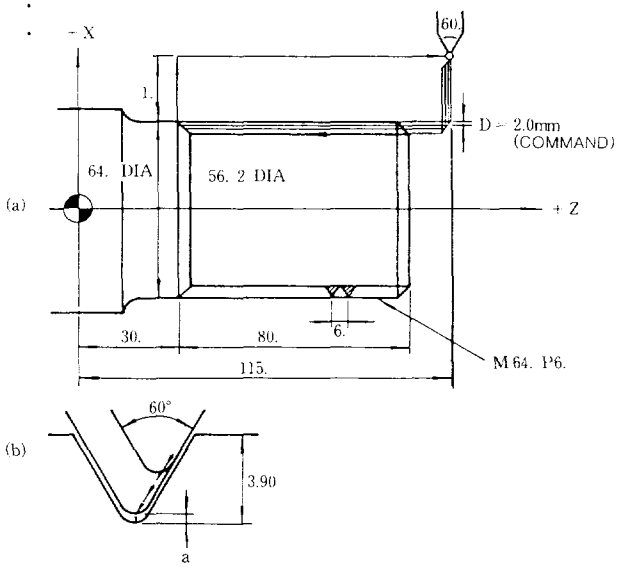
G76··· ;

G00 M30 ;

2.8.25.8 Automatic Threading Cycle (G76) (Cont'd)

EXAMPLE:

```
      .
      .
G00  X66.  Z115.  ;
G76  X56.2  Z30.  K3.9  D2.  F6.  A60  ;
G00  ...
```



Depth of cut for each cycle when a (fixed amount) is 0.2 mm

1st cycle	——	1.700 mm
2nd cycle	——	2.528 mm
3rd cycle	——	3.164 mm
4th cycle	——	3.700 mm
5th cycle	——	3.900 mm

Though D200 (2.0 mm) is programmed, the actual depth of cut becomes 1.7 mm by the calculation of $\sqrt{Nend}$ D because of the difference between $\sqrt{Nend}$ D and (K - a).

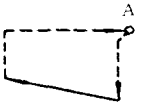
2.8.25.9 Precautions in Programming G70 through G76

Prohibition of MDI mode

- Operation in MDI mode cannot be made while multiple repetitive cycles (G70 through G76) are executed.
- Multiple repetitive cycles (G70 through G76) cannot be written-in through the operation in MDI mode.

Single block operation

- Executing G70 through G76 at SINGLE BLOCK switch ON brings the following results.

G70, G71, G72, G73, G74, G75	Program Stops at Every Block.
G76	 Program stops at position A after each cycle is finished.

Symmetrical pattern

- The symmetrical four patterns can be commanded by each of G71 to G76.
- Signs of U, W and I should be properly specified in the finishing shape program for G71 to G73. (See Fig. 2.8.25.33)
- Command position of (X, Z) or (U, W) with respect to position A should be properly specified for G74 to G76.

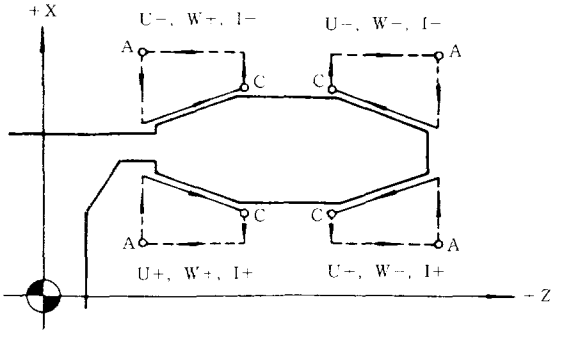


Fig. 2.54 Four Patterns of G76

Table 2. 22 Multiple Repetitive Cycles

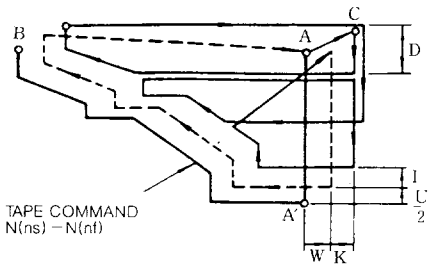
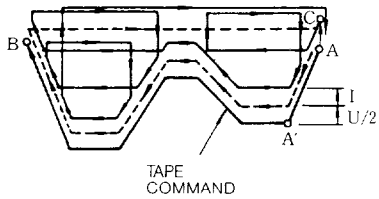
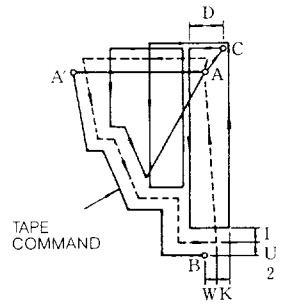
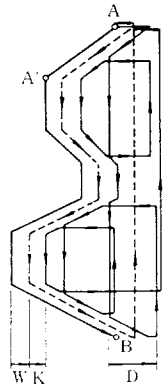
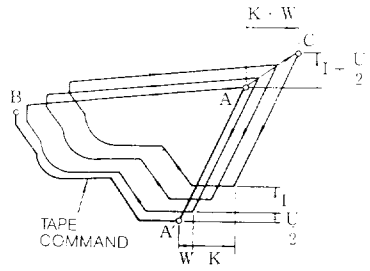
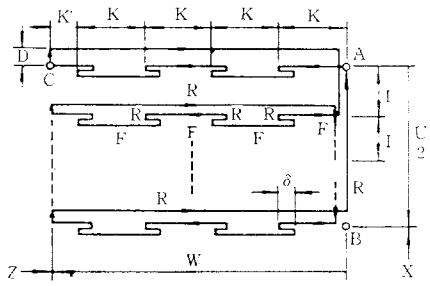
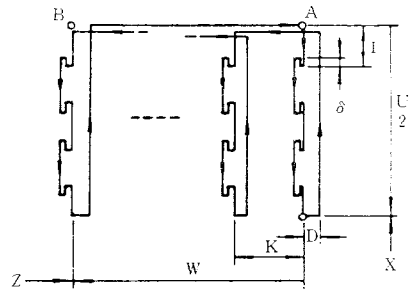
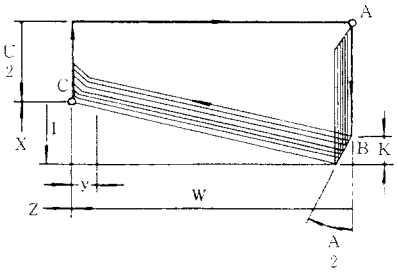
	Cutting Cycle	Command Format
G 71 Stock Removal in Turning	 	(1) Monotonous increase/decrease path G 71 Pns Qnf U... W... I... K... D... F(E)... S... ; Nns... ; . . . Nnf... ; } Finishing shape program (2) Concaved shape G 71 Pns Qnf U... I... D... F(E)... S... R1; Nns... ; . . . Nnf... ; } Finishing shape program (U, W, I, and K must be programmed with signs.)
G 72 Stock Removal in Facing	 	(1) Monotonous increase/decrease path G 72 Pns Qnf U... W... I... K... D... F(E)... S... ; Nns... ; . . . Nnf... ; } Finishing shape program (2) Concaved shape G 72 Pns Qnf W... K... D... F(E)... S... R1 ; Nns... ; . . . Nnf... ; } Finishing shape program (U, W, I, and K must be programmed with signs.)

Table 2. 22 Multiple Repetitive Cycles (Cont'd)

	Cutting Cycle	Command Format
G73 Pattern Repeating		G73 Pns Quf U... W... I... K... D... F(E)... S... ; Nns... ; . . . Nnf... ; (U, W, I, and K must be programmed with signs.) Finishing shape program
		G70 Pns Qnf ;
G73 Peck Drilling in Z-axis		G74 / G75 } X(U)... Z(W)... I... K... D... F(E)... R1 ; (1) "R1" not specified: The operation shown left is performed. (2) "R1" specified: Retracting amount (δ) for each approach is ignored and the tool returns to the level of point A every time. (I, D, K must be programmed without signs.)
G75 Grooving in X-axis		
G76 Automatic Thread-cutting		G76 X(U)... Z(W)... I... K... D... F(E)... A... ; A: Angle of thread (0°, 29°, 30°, 55°, 60°, 80°) K and D must be programmed without signs. $\frac{1}{6}K \leqq D \leqq K$

2.8.25.10 Improved Multiple Repetitive Cycle Function

2.8.25.10.1 Nose R compensation can also be executed in the finishing form program against the G70 to G73 cycles.

This adds G41 and G42 to the G codes that can be specified in the blocks of the finishing form program (except for the Nsf. Nnf blocks). Nose R compensation is enabled by the G41 or G42 command blocks in the rough finishing cycle and finishing cycle.

When commanding G41/G42 in the starting block of the finishing form program, also command the G00 or G01 move command.
Commanding in a single block is prohibited.

- Upgrade of "Par. 2.8.25.2 (notes upon commanding the G71 command) (4), (5)"
- Upgrade of "Par. 2.8.25.4 (notes and remarks) (5)"
- Upgrade of "Par. 2.8.25.5 (notes and remarks) (2)"

2.8.25.10.2 The T code command can also be executed against the G70 to G73 cycles in the finishing form program.

This enables the F, S, T codes commanded in the finishing form program (G70) and disables them during outer form rough cutting.

- Upgrade of "Par. 2.8.25.2 (notes upon giving the G71 command) (3)"
- Upgrade of "Par. 2.8.25.4 (notes and remarks) (1)"

2.8.25.10.3 A finishing form program in which start point and end point differ can also be commanded against the G70 to G72 cycles.

- Upgrade of "Par. 2.8.25.2 (notes on the finishing form program) (1)". (BA does not need to be parallel with the Z-axis.)
- Upgrade of "Par. 2.8.25.3 (notes on the finishing form) (2)". (BA does not need to be parallel with the X-axis.)

2.8.25.10.4 When both rough finishing margins I and K are omitted against the G70 to G73 cycles, finishing margin U and W can be executed as the rough finishing margin.

Upgrading is enabled by related parameter #6009 D1 = 1.

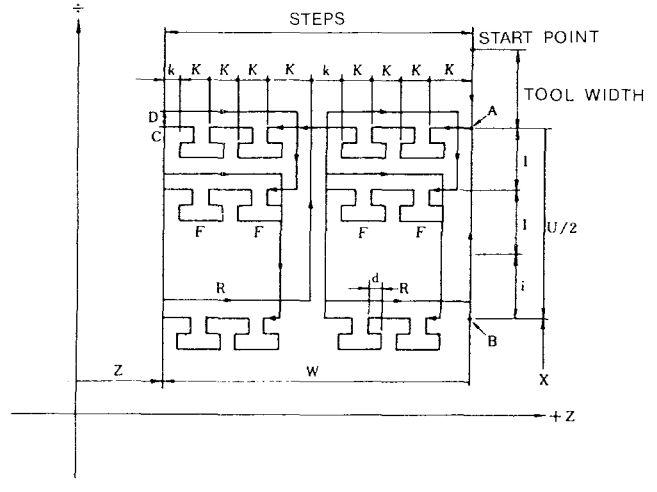
2.8.25.10.5 The A command (infeed steps) and the B command (cutting tool width) are added to the G74 (end face cutting cycle) and the G75 (outer diameter cutting cycle) functions to allow execution of the groove machining canned cycle.

(1) G74 executes a cycle as shown in Fig. 1.118, by the following command.

(Upgrade of "Par. 2.8.25.6 (1)".)

G74X(U)_ Z(W)_ I_ K_ D_ A_ B_ F(E)_ (R1);

- F(E): Feed command
- B: Tool width (without sign)
- A: Infeed steps (without sign)
- D: Clearance at the bottom (without sign)
- K: Z-axis direction infeed (without sign)
- I: X-axis direction move distance (without sign)
- Z(W): C point Z-axis coordinate (with sign)
- X(U): B point X-axis coordinate (with sign)



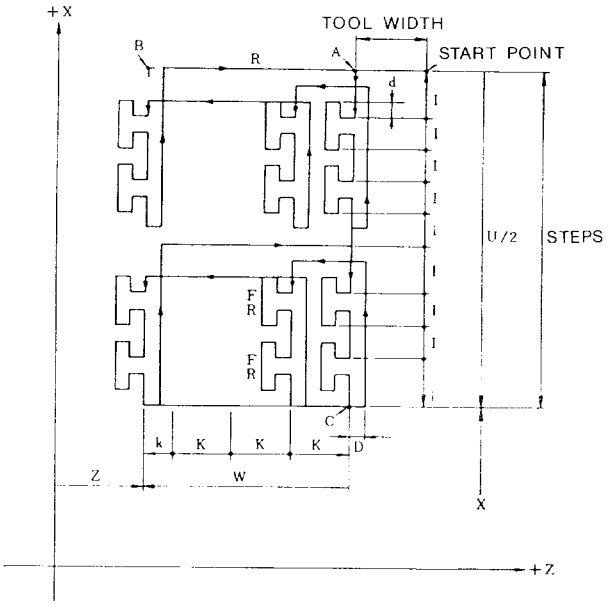
(2) G75 executes a cycle as shown in Fig. 1.119, by the following command.

(Upgrade of Par. 2.8.25.7 (1)".)

G75X(U)_ Z(W)_ I_ K_ D_ A_ B_ F(E)_ (R1);

- F(E): Feed command
- B: Tool width (without sign)
- A: Infeed steps (without sign)
- D: Clearance at the bottom (without sign)
- K: Z-axis direction infeed (without sign)
- I: X-axis direction move distance (without sign)
- Z(W): B point Z-axis coordinate (with sign)
- X(U): C point X-axis coordinate (with sign)

2. 8. 25. 10 Improved Multiple Repetitive Cycle Function (Cont'd)



2.8.25.10.6 Notes of the improvement of the multiple repetitive cycle functions

(1) The following conditions are added to Pars. 2.8.25.10.1, 2.8.25.10.2, and 2.8.25.10.3.

(a) The upgrading effective by related parameter #6011 D1 = 1.

(b) This improvement expands the maximum storage of the finishing form program to 39 blocks.

(Improvement of Par. 2.8.25.1 (general) (2).)

(2) Notes on improvement made to Par. 2.5 G74/G75

(a) Without both commands A and B, the conventional G74/G75 movement is executed.

(b) When B alone is commanded, tool width shift is performed at the beginning and end of G74/G75. Details are as follows.

(i) The first movement is the shift for the tool width, from the positioning location of the block immediately before G74/G75 to the X-axis command direction (G74) or Z-axis command direction (G75).

(ii) The final movement is to positioning from the positioning location of the block immediately before G74/G75 to the tool-width shift position and then to return to the position of the block immediately before G74/G75.

(c) When A alone is commanded, only groove machining is performed without executing tool width shift.

(d) When A is commanded, the return amount is the #6208 setting amount (G74) or #6209 setting amount (G75). If the contents are "0", no pecking is performed. (Addition to Par. 2.8.25.6 Notes and remarks).)

(e) Alarm 96 occurs when the command is groove width < B (tool width).

2. 8. 26 CANNED CYCLES (G90, G92, G94)

A series of basic lathe operations specified usually in four blocks, can be commanded in one block.

There are the following three canned cycles.

2. 8. 26. 1 Turning Cycle A (G90)

(1) Straight turning cycle

G90 X(U)··· Z(W)··· F(E)··· ;

The cycle ① to ④ shown below is executed by this command.

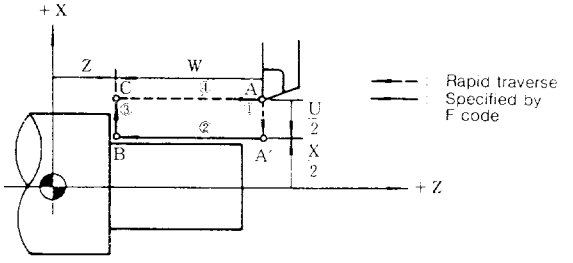


Fig. 2.55

Since G90 is modal, the cycle operation will be continued by specifying the depth of cut in the X-axis direction in the following blocks as follows.

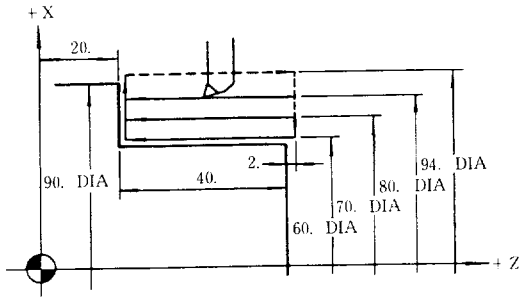
X(U)··· ;
X(U)··· ;

EXAMPLE:

```
N10 G00 X94. Z62. ;
N11 G90 X80. W-42. F0.3 ;
N12 X70. ;
N13 X60. ;
N14 G00··· ;
```

N11 G90 X80. W-42. F0.3 ; — Cycle start

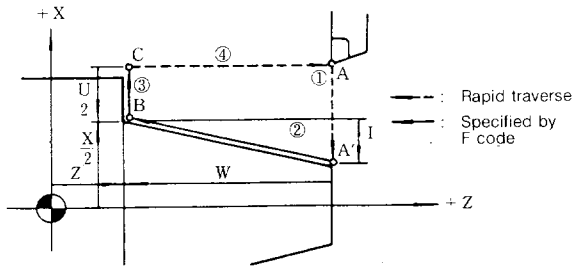
N12 X70. ; / N13 X60. ; — G90 is executed with depth cut changed.



(2) Taper cutting cycle

G90 X(U)··· Z(W)··· I··· F(E)··· ;

The cycle ① to ④ shown below is executed by this command.

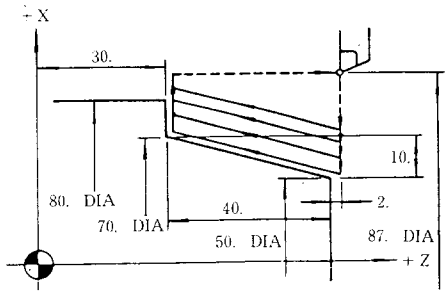


The sign of figure following the address I is decided by the direction of position A' viewed from position B.

Fig. 2.56

EXAMPLE:

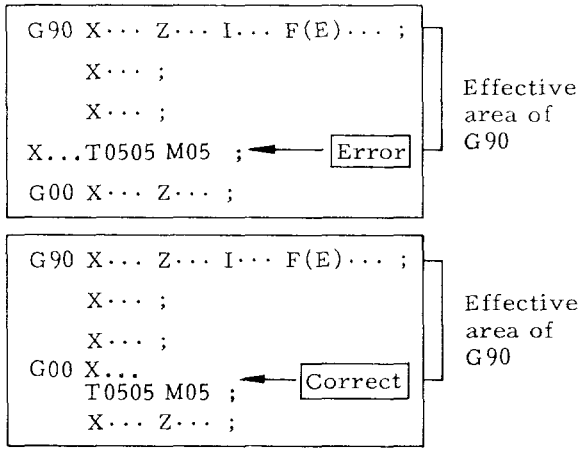
```
N20 G00 X87. Z72. ;
N21 G90 X85. W-42. I-10.5 F0.25 ;
N22 X80. ;
N23 X75. ;
N24 X70. ;
N25 G00··· ;
```



2.8.26.1 Turning Cycle A (G90) (Cont'd)

(3) Cautions in programming G90

a. T, S and M functions for G90 cycle must be specified as a rule beforehand in the preceding blocks. T, S and M blocks without move commands are effective in the effective area of G90.



The effective area of G90 is from the block containing G90 to the one before the block in which the other G code of 01 group is specified. This rule also applies to the G92 and G94 described later.

b. In Single Block mode, the execution of G90 cycle stops after the completion of the cycle ① to ④ .

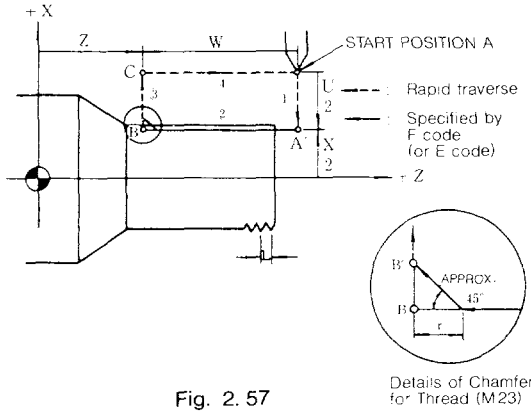
2.8.26.2 Threading Cycle (G92)

(1) Straight threading cycle

G92 X(U)··· Z(W)··· F(E)··· ;

Lead designation (L)

The cycle ① to ④ shown below is executed by this command.



Details of Chamfering for Thread (M23)

Fig. 2.57

Rapid threading pull-out:

If "rapid threading pull-out input (CDZ)" is on when G92 is specified, rapid threading pull-out is performed. Rapid threading pull-out value (γ) may be set to parameter #6080, in the range of 0 to 25.5L and in increments of 0.1L, where L is the lead of the specified thread. Generally, it is convenient to create and use the sequence in which "rapid threading pull-out (CDZ)" is turned on/off through the use of a given M code.

Since G92 is modal, the cycle operation will be continued by specifying depth of cut in the direction of X-axis.

X(U)··· ;
X(U)··· ;

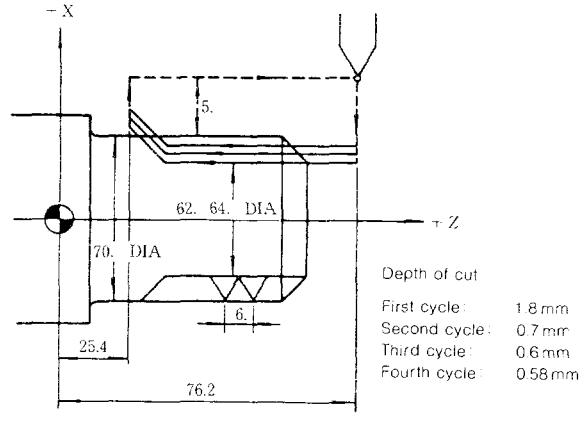
EXAMPLE:

N30 G00 X80. Z76.2 M○○ ;--M○○: Rapid threading pull-out ON

```
N31 G92 X66.4 Z25.4 F6. ;
N32 X65. ;
N33 X63.8 ;
N34 X62.64 ;
```

Threading cycle cutting four times

N35 G00 X100.Z100. M△△ ;—M△△: Rapid threading pull-out OFF



(2) Straight thread cutting cycle with angle

G92 X(U)··· Z(W)··· K··· F(E)··· ;

This command permits the thread cutting along the angle of thread. The cycle ① to ④ shown below is executed.

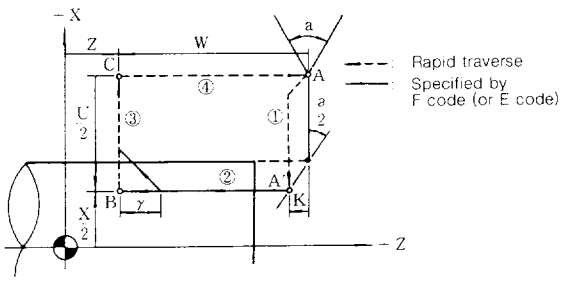


Fig. 2.58

The sign of figure following address K is decided by the direction of position A' viewed from position A.

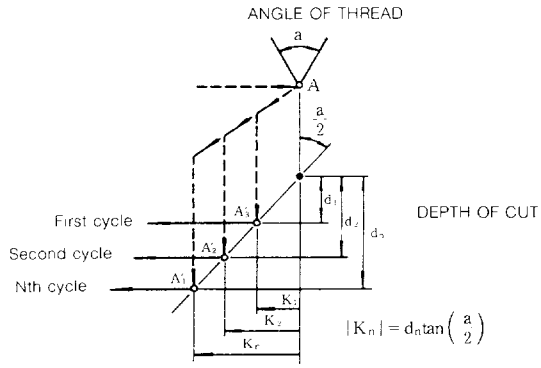
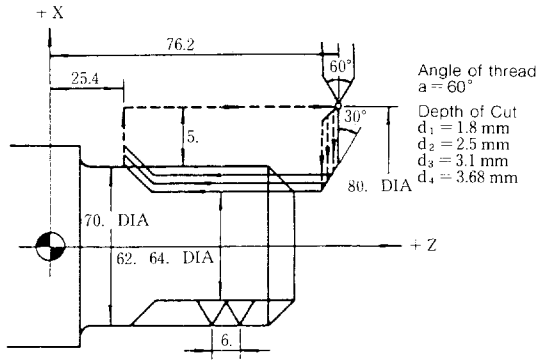


Fig. 2.59

For the threading along the angle of thread, the K for each cycle obtained from the above formula should be programmed.

a	tan (a/2)
29°	0.258618
30°	0.267949
55°	0.520567
60°	0.577350
80°	0.839100

EXAMPLE



Calculation of K··· K = d·tan(60°/2)

K_1 = -1.8 x 0.57735 = -0.866 mm

K_2 = -2.5 x 0.57735 = -1.443 mm

K_3 = -3.1 x 0.57735 = -1.790 mm

K_4 = -3.68 x 0.57735 = -2.125 mm

```
N40 G00 X80. Z76.2 M00 ;
N41 G92 X66.4 Z25.4 K-0.87 F6. ;
N42 X65. K-1.44 ;
N43 X63.8 K-1.79 ;
N44 X62.64 K-2.13 ;
N45 G00 X100. Z100. MΔΔ ;
```

(3) Taper threading cycle

G92 X(U)··· Z(W)··· I··· F(E)··· ;

The cycle ① to ④ shown below is executed by this command.

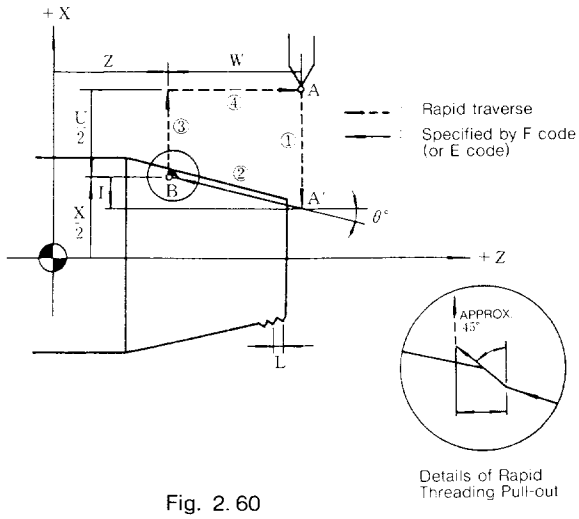


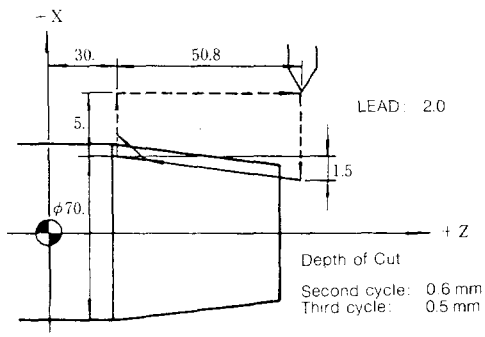
Fig. 2.60

The sign of figure following address I is decided by the direction of position A' viewed from position B. Since G92 is modal, the cycle operation is continued by specifying threading depth in the X-axis direction in the trailing blocks as follows.

```
X(U)··· ;
X(U)··· ;
```

EXAMPLE:

```
N50 G00 X80. Z80.8 M00 ;
N51 G92 X70. W-50.8 I-1.5 F2. ;
N52 X68.8 ;
N53 X67.8 ;
N54 G00 X100. Z100. MΔΔ ;
```



(4) Taper threading cycle with angle (Cutting along the angle of thread)

G92 X(U)··· Z(W)··· I··· K··· F(E)···;

This command permits the threading along the angle of thread in taper threading. The figure following the address K must be programmed with a sign.

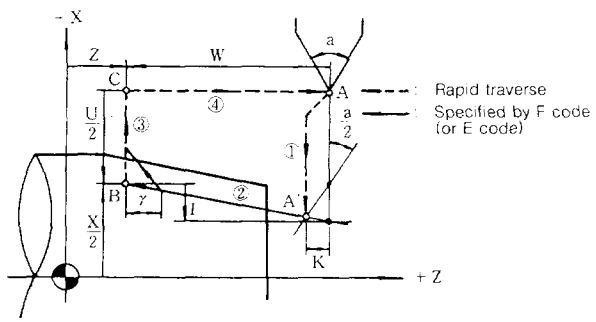


Fig. 2.61

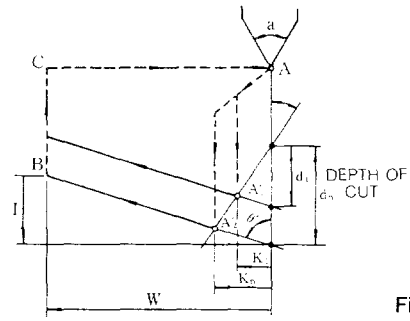


Fig. 2.62

For the threading along the angle of thread, the K for each cycle obtained from the following formula should be programmed.

$$|Kn| = \frac{dn.\tan(a/2)}{1 \pm \left|\frac{I}{W}\right|.\tan(a/2)}$$

The sign in the denominator depends on θ':

θ' < 90° → "+"
θ' > 90° → "−"

Note: For the control equipped with Multiple Repetitive Cycle†, the above troublesome calculation can be omitted by using G76 (Automatic Thread Cutting Cycle). The control performs the above calculation automatically by the G76 command.

(5) Cautions in programming G92

a. T, S and M functions for G92 cycle must be specified as a rule beforehand in the preceding blocks. T, S and M blocks without move commands are effective in the effective area of G92.

b. In Single Block mode, the execution of G92 cycle is stopped after the completion of the cycle ① to ④.

c. Thread cutting hold (option)

When this option is used, depressing the FEED HOLD during thread cutting causes the tool to perform threading up immediately and then return to start point A.
When parameter #6019D_7 = 1, the tool stops at position B where threading up terminated. Depressing the CYCLE START at start point A or B restarts the current processing from the beginning.

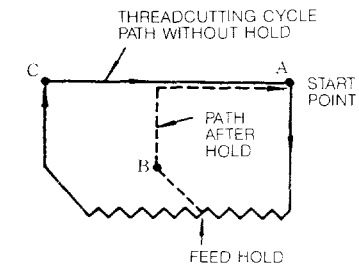


Fig. 2.63

If the threadcutting hold option is not selected, depressing of the FEED HOLD during thread cutting does not hold the operation, which stops at point C where the retracting operation is completed.

d. Six angles of thread can be used in Multiple Repetitive cycle G76. In the G92 command, arbitrary angles of thread can be performed for threading.

When G92 (threading cycle) is used with chamfering ON, thread pull-up value of 0 may activate alarm "064."

2.8.26.3 Facing Cycle B (G94)

(1) Straight facing cycle

G94 X(U)··· Z(W)··· F(E)··· ;

The straight facing cycle ① to ④ is executed by this command.

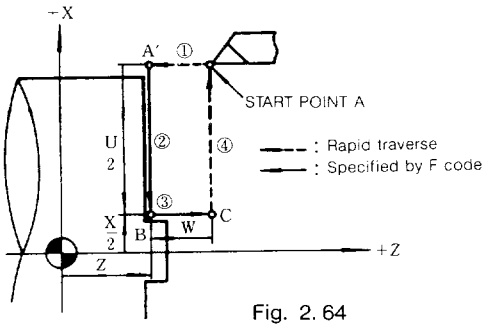


Fig. 2.64

Since G94 is modal, the cycle operation will be continued by specifying depth of cut in the Z-axis direction in the following blocks as follows.

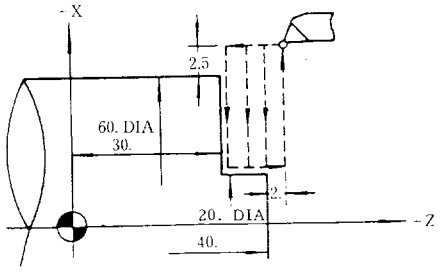
Z(W)··· ;

Z(W)··· ;

EXAMPLE:

```
N60 G00 X65. Z42. ;
N61 G94 X20. Z38. F0.35 ;
N62 Z34. ;
N63 Z30. ;
N64 G00··· ;
```

Three cycles by G94 command.



(2) Taper facing cycle

G94 X(U)··· Z(W)··· K··· F(E)··· ;

The taper facing cycle ① to ④ is executed by this command.

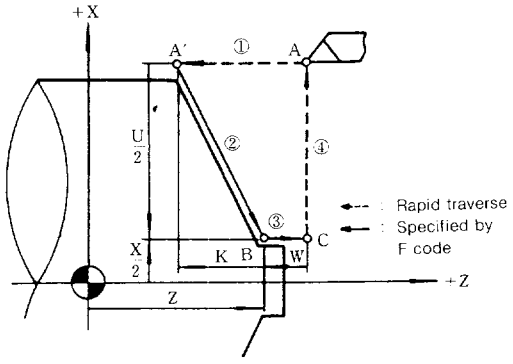


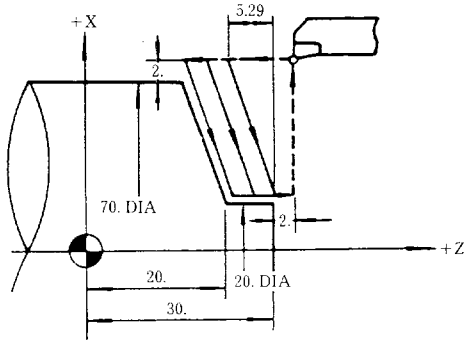
Fig. 2.65

The sign of figure following the address K is decided by the direction of position A' viewed from position B.

EXAMPLE:

```
N70 G00 X74. Z32. ;
N71 G94 X20. Z30. K-5.29 F0.3 ;
N72 Z25. ;
N73 Z20. ;
N74 G00··· ;
```

Three cycles by G94 command

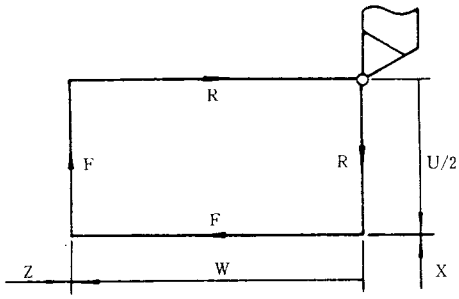
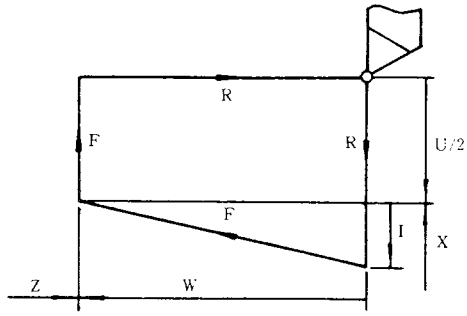
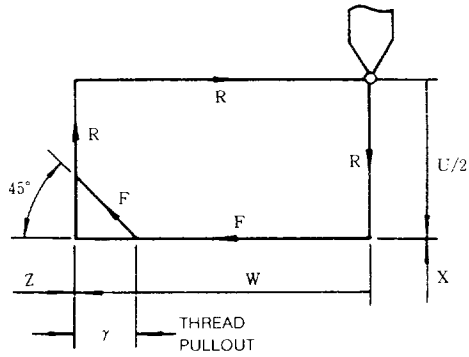
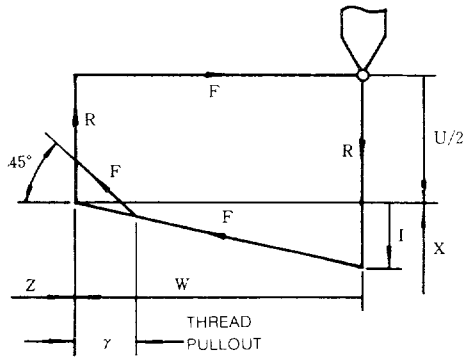
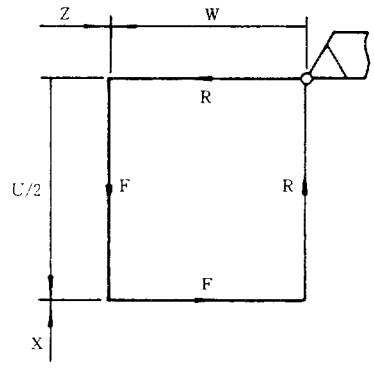
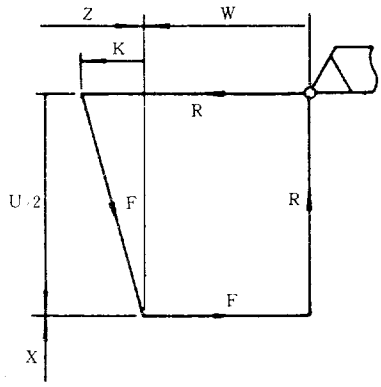


(3) Cautions in programming G94

a. T, S and M functions for G94 cycle must be specified as a rule beforehand in the preceding blocks. T, S and M blocks without move commands are effective in the effective area of G94.

b. In Single Block mode, the execution of G94 cycle is stopped after the completion of ① to ④.

Table 2. 23 Canned Cycles

Code	Straight Cycle	Taper Cycle
G 90 Turning Cycle	G 90 X (U)... Z (W)... F (E)... ;	G 90 X (U)... Z (W)... I... F (E)... ;
		
G 92 Threading Cycle	G 92 X (U)... Z (W)... F (E) .. ;	G 92 X (U)... Z(W)... I... F (E)... ;
		
G 94 Facing Cycle	G 94 X (U)... Z (W)... F (E)... ;	G 94 X (U)... Z (W)... K... F (E)... ;
		

2.8.27 CONSTANT SURFACE SPEED CONTROL (G96, G97)†

This feature may be installed when the S4 digit specification option is selected.

The following G codes are used:

G96	Specifies the constant surface speed control.
G97	Cancels the constant surface speed control.

At the time of power-on, G97 (cancel) is provided.

These are modal G codes which belong to 02 group.

(1) Constant surface speed control (G96)

a. G96 S··· (M03) ;

Using this command, specify the surface speed of the work in a numeral of up to 4 digits after address S. The unit of the surface speed is as follows:

	Unit
Metric input	m/min
Inch input	ft/min

When the surface speed is specified, the equipment assumes the X-axis current value (Note 1) to be the diameter of the work and computes the spindle RPM every 100 msec so that the specified surface speed is obtained. The computation result is sent as the analog voltage output or the binary 12-bit signal output. In the subsequent blocks, the surface speed may be varied by S specification.

b. For a machine tool that allows spindle gear change, instruct the M code for gear change specification before the block of G96 command. For details, refer to the instruction manual for that machine tool.

EXAMPLE:

```
N8  MΔΔ  ; ———— M code for spindle
                 gear change (gear ratio
                 No. 4 is specified).
N9  G96  S100  M03  ;
    .
    .
    .
```

c. Before the block of G96 command, the spindle maximum RPM should be specified by G50. This specification prevents the spindle RPM from getting abnormally high by the computation of surface speed when the X-axis current value is smaller.

EXAMPLE:

```
N10  G50  S2000  ; ——— Specified the upper
                       limit of spindle RPM.
N11  MΔΔ  ;
N12  G96  S150  M03  ;
```

(2) Constant surface speed control cancel (G97)

G97 S··· (M03) ;

Using this command, directly specify the spindle RPM in a numeral of up to 4 digits after address S. The constant surface speed control is cancelled and the usual spindle feature of S 4-digit specification is resumed.

(3) Cautions for constant surface speed control

a. How to set coordinate system (See Note)

When performing the constant surface speed control, make programming by setting G50 coordinate system or G58 work coordinate system so that X-axis coordinate value of the spindle rotational center become "0." In other words, the coordinate system should be set such that its X-axis coordinate value correctly represents the diameter of the work at its machining point.

b. How to handle tool position offset amount

(i) When performing the constant surface speed control, parameter #6020D1 is generally set to "0." This enables the control to perform the computation for constant surface speed control without adding the tool position offset amount to the specified coordinate value. If a large value is used for the offset amount, the tool position offset is normally executed and, at the same time, the constant surface speed control is correctly performed.

(ii) When #6020D1 is set to "1," the value (Specified coordinate value + tool position offset amount) is assumed to be the work diameter to compute the constant surface speed. Hence, care should be taken not to use a large value for the offset amount by performing coordinate system setting for each tool and using the tool position offset for tool wear compensation.

2.8.27 CONSTANT SURFACE SPEED CONTROL (G96, G97)† (Cont'd)

Note: #6020D1 and #6020D7 are independend from each other.

#6020D7 = "0" ··· The position obtained by adding the tool position offset amount and the tool nose radius compensation amount is shown in the current position display "POSITION ABSOLUTE."

#6020D7 = "1" ··· The position obtained by not adding the tool position offset amount is shown in the current position display "POSITION ABSOLUTE." Thus, the switching of the computation for constant surface speed control and the switching of the computation for current position display on CRT are performed separately.

c. The spindle gear may be changed in the maximum of 4 steps. The parameters for the gear change are as shown below. For details, see the Parameter List.

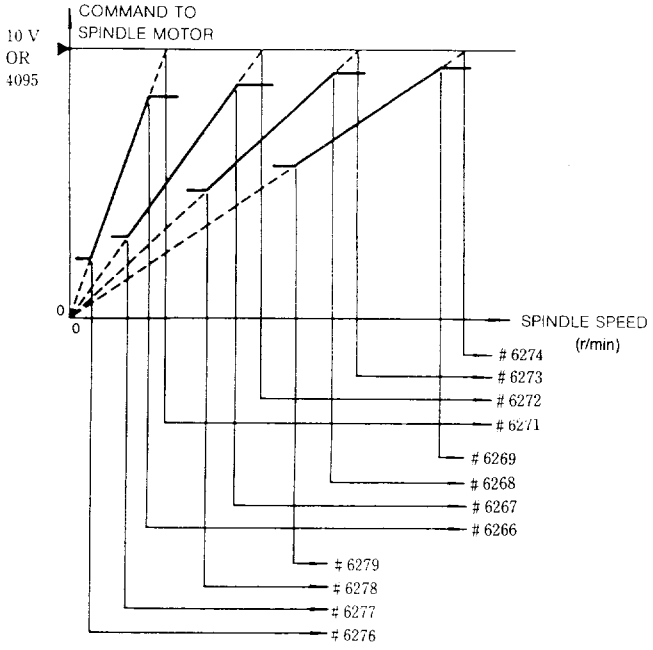


Fig. 2.66

d. When parameter #6020D0 = "1," the constant surface speed control is performed on all positioning blocks (G00, G06) also. For positioning, however, the constant surface speed is computed for the end coordinate of the positioning block. The constant surface speed is at every moment computed for only the cutting block.
When #D6020D0 = "0," the constant surface speed control is performed only on the cutting block and the positioning block immediately before it. For the positioning block, the constant surface speed is computed for its end coordinate.

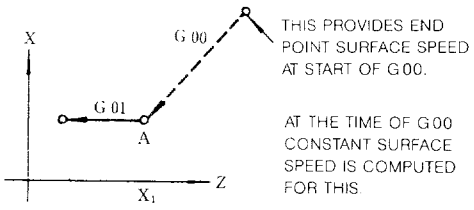
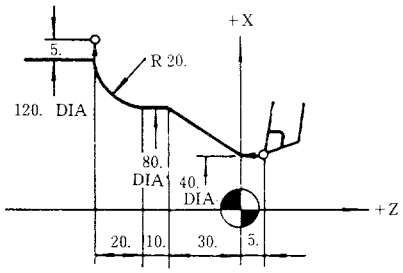


Fig. 2.67

EXAMPLE: #6020 = 0

```
N4 G50 S1500 ;      ┌ Upper limit of spindle
                    └ speed clamp (in r/min)
N5 MΔΔ ;            ─ M code for gear change
N6 G96 S150 M03 ;   ┌ Constant surface
                    └ speed 150 m/min
N7 G00 X40. Z5. ;   ┐
N8 G01 Z0 F0.15 ;   │
N9 X80. Z-30. ;     │ Constant surface speed control
N10 W-10. ;         │
N11 G22 X120. W-20. R20. ;  │
N12 G01 U10. ;      ┘
N13 G97 S500 ;      ┌ Cancel of constant surface speed control
N14 G50 S2000 ;     ┘
```



2.8.28 FEED/MINUTE AND FEED/REVOLUTION SWITCHOVER

These G codes are used to switch between the designation of feed per minute and the designation of feed per revolution, before specifying F(E) code (feed) command.

(1) G98 ;

By this command, the F(E) code specified after is executed on a feed per minute basis.

G98	Meaning
Metric input	mm/min designation
Inch input	in/min designation

(2) G99 ;

By this command, the F(E) code specified after is executed on a feed per revolution basis.

G99	Meaning
Metric input	mm/rev designation
Inch input	inch/rev designation

Notes :

1. G98 and G99 are modal G codes, which remain valid until the designation is changed.
2. When switching between G98 and G99 is performed, F(E) code designated before is cancelled. So, it must be designated again. Otherwise, an error is caused at the next cutting block.
3. The initial G code at power-on may be switched between G98 and G99 by the use of the following parameter:

Parameter	Initial G Code
#6005 $D_1 = 0$	G98
#6005 $D_1 = 1$	G99

2.8.29 TOOL LIFE CONTROL (G122, G123)†

This feature allows a long, unattended operation by replacing the tools each time the designated number of pieces or the designated use hour is reached.

(1) The tool life control consists of the following three steps:

a. Registration of tool information

(i) Registration of tool group number. The tool numbers of the tools of the same type are registered as one group.

(ii) Registration of life of each tool group. The number of pieces that can be machined by a tool to be registered in each tool group and the usable hours of it are registered.

(iii) Registration of use offset number of each tool.

These must be registered from the part program in the memory of the equipment before starting machining.

b. T designation for tool life control

For the T designation in the part program, the special T designation for tool life control must be used.

c. Input/output signals for tool life control.

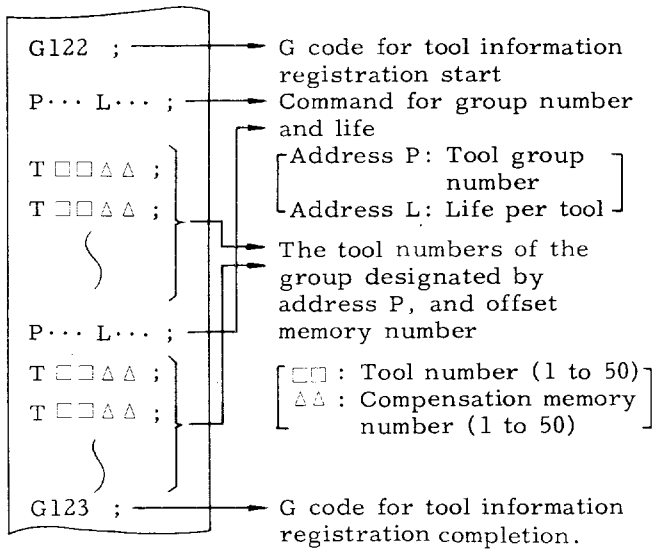
- Tool life control I/O signal.
- Tool skip input.
- Tool replacement completion input.
- Tool replacement completion group number input.

The above inputs are provided to implement the highly efficient tool life control feature.

The following describe these three steps in detail:

(2) Registration of tool information

a. The part program for tool information registration



Group No.		19 groups, 1 to 19, may be used.
Life per Tool	Number of pieces	For groups 1 to 9, life is designated by the number of works (1 to 9999 works).
	Use hours	For groups 10 to 19, life is designated by use hours (1 to 9999 minutes).

b. When the part program for tool information registration is executed, the following screen is shown in the COM function display on CRT. The screen enables the confirmation of the registered information. (Depress PAGE key to display the following screen.)

EXAMPLE:

The display of the information registered by executing the following program is shown.

```
O··· ;
G122 ;
P1 L7890 ;
T0101 ;
T0303 ;
T0606 ;
T0909 ;
T1111 ;
T1313 ;
P12 L7890 ;
T0404 ;
T0808 ;
T1010 ;
T1212 ;
T1414 ;
G123 ;
M30 ;
```

```
TOOL LIFE CONTROL     01234  N1234

  TOOL GROUP  1  (LIFE 7890 COUNT)
      COUNTER        0

T0101 ;
T0303 ;
T0606 ;
T0909 ;
T1111 ;
T1313 ;

                                RDY
```

Fig. 2.68 Tool Life Data On Tool Life Control by No. of Pieces to be Machined

```
TOOL LIFE CONTROL     01234  N1234

  TOOL GROUP 12 (LIFE 7890 MIN.)
  TIMER                0 MIN.

  T0404 ;
  T0808 ;
  T1010 ;
  T1212 ;
  T1414 ;

                                RDY
```

Fig. 2.69 Tool Life Data on Tool Life Control by Operating Time

(3) T designation for tool life control

Using the following special T commands, create the part program for machining:

a. Use the following T commands:

T□□90	The tool of the group specified in "□□" is given as T command. However, tool position offset is cancelled.
T□□91	The tool of the group specified in "□□" is given as T command and, at the same time, the content of the registered offset number is applied as offset.

Note: When two or more compensation memories are used for a single tool, refer to the description in subparagraph (5), a.

b. When a group number registered by the number of pieces machinable per tool of group number 1 to 9 is specified, give "T□□99" (□□ is the group number) for the command for counting the number of pieces, at the completion of machining.
When the group numbers 10 to 19 which designate the maximum operating time of a single tool, the control automatically counts the feed time as operating time. T-command need not be programmed for time counting on part program.

c. Display for confirmation of the contents of "T□□91" command.

EXAMPLE:

When the following part program has been executed after the tool information registration described in subparagraph (2):

```
O···
G00 T0191 ;  ——①
  ⌇
T0190 ;      ——①′
  ⌇
T00 T1291 ;  ——②
  ⌇
T1290 ;      ——②′
  ⌇
T0199 ;
M99
```

At the execution of the part program specified with tool life control, the tool number in the current group and the offset number being used by "T□□91" command may be shown in COM display on CRT screen.

In the above program, ① shows that "T0191" specifies the machining by "T0606" in the registered tool information.
(①′ shows that "T0190" becomes "T0600.")

```
TOOL LIFE CONTROL      O1234 N1234

 TOOL GROUP 1   (LIFE 7890  COUNT)
     COUNTER     1234

 T0101 ; END
 T0303 ; END
 T0606 ; CUTTING
 T0909 ;
 T1111 ;
 T1313 ;

                                RDY
```

Fig. 2.70 Example A of Display of Tool No. and Offset No. of Current Execution

```
TOOL LIFE CONTROL      O1234 N1234

 TOOL GROUP 12  (LIFE 7890 MIN.)
     TIMER       1234 MIN.

 T0404 ; END
 T0808 ; CUTTING
 T1010 ;
 T1212 ;
 T1414 ;

                                RDY
```

② shows that "T1291" command specifies the machining by "T0808" in the registered tool information.
(②′ shows that "T1290" becomes "T0800" command.)

Fig. 2.71 Example B of Display of Tool No. and Offset No. of Current Execution

(4) Input/output signals for tool life control

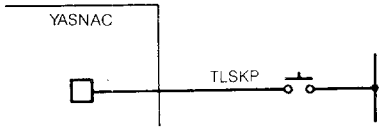
The following input/output signals are provided for the control:

Table 2.24

Signal Name	Description
TLSKP	Tool skp input.
TLCH	Tool replacement request output.
TLRST	Tool replacement completion input.
TLA 11 TLA 12 TLA 14 TLA 18 TLA 21	Tool replacement completion group number input. (BCD code).

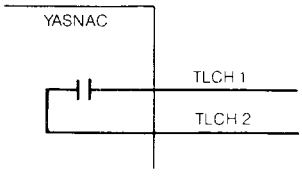
The operation for each input/output depends on the operation panel and external sequences provided by the machine builder. For details, refer to the instruction manual offered by the machine builder. The following describes the control function in response to the input/output signals to facilitate the understanding of the tool life control feature.

a. TLSKP: Tool skip input



This is the input signal for replacing the tool before the registered life of it expires. When TLSKP input is "closed" during the time from the output of the tool registered by group number "1 to 19" as T command to the output of another T command, the control outputs the T command of the next tool registered in the group at the time of the next T command output.

b. TLCH: Tool replacement request output



All tools in each group number are given when the registered life (the number of pieces or use hours) expires. TLCH output turns ON when:

- Tool life expires in executing T□□ 99 commands when number of workpieces is specified.
- Tool life expires and T command of its group number is given when tool life is determined by use hours.

(iii) T command for life completion group number was executed.

When the tool replacement request output signal goes on, alarm number 157 is displayed on the CRT screen.

c. TLCH1, TLCH2: Tool replacement request output

After all machining life of registered tool with either tool group No. expires, TLCH1 and 2 are closed by performing program end or reset operation. When main input is closed, check for the tool No. of machining life expired and replace it.

Note: When tool replacement request inputs TLCH1 and 2 are closed; automatic start is prohibited in automatic operation mode.

When the tool replacement request output signal goes on, look at the CRT screen, namely, the tool life control display screen of COM function, make confirmation of the number of the group in which all tools have come to the expiration of useful life, and replace them.

EXAMPLE:

Shown below is a display example in which the life of the tools belonging to group number "1" has been completed.

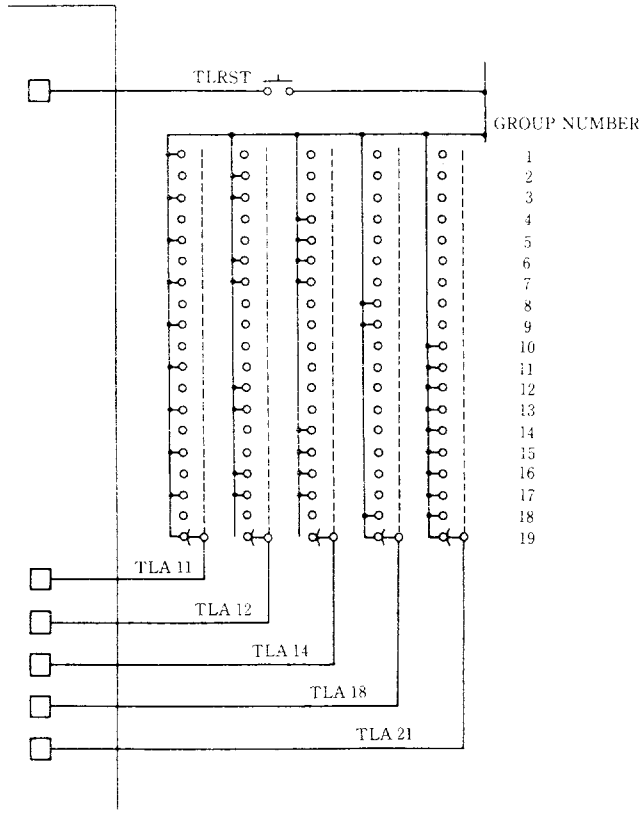
```
TOOL LIFE CONTROL     01234   N1234

  TOOL GROUP  1 (LIFE 7890 COUNT)
      COUNTER   7890

  T0101 ;  END
  T0303 ;  END
  T0606 ;  END
  T0909 ;  END
  T1111 ;  END
  T1313 ;  END
```

d. Tool replacement completion input (TLRST) and tool replacement completion group number inputs (TLA11 through TLA21). These signals are used to indicate the replacement completion to the control after the tool replacement request output (TLCH1, TLCH2) goes on and the tools whose life has expired are replaced with new ones.

When the tool replacement operation is completed, set the group number to "tool replacement completion group number input (TLA11, TLA12, TLA14, TLA18, TLA21) and put the tool replacement completion input (TLRST) in the "closed" state.



When the tool replacement operation has been completed with the tool replacement completion input "closed," look at the tool life control display screen of COM function on CRT and check to see if the control internal memory is accepting the completion of the replacement operation. When this operation is found accepted, perform a reset operation (depress RESET key on MDI & CRT panel or "close external reset input ERS), and the tool replacement request output goes off and the displayed alarm code disappears.

EXAMPLE:

Shown below is a display example in which the life of the group "12" tools has expired.

```
TOOL LIFE CONTROL     01234   N1234

  TOOL GROUP  12 (LIFE 7890 MIN.)
      TIMER        7890  MIN.

  T0404 ;  END
  T0808 ;  END
  T1010 ;  END
  T1212 ;  END
  T1414 ;  END

                                RDY
```

Shown below is a display example in which "12" is set as the tool replacement completion group number input and the tool replacement completion input is "closed" after replacement of all tools.

```
TOOL LIFE CONTROL     01234   N1234

  TOOL GROUP  12 (LIFE 7890 MIN.)
      TIMER           0 MIN.)

  T0404 ;
  T0808 ;
  T1010 ;
  T1212 ;
  T1414 ;

                                RDY
```

(5) Supplementary explanation for tool life control

a. The tool life control in which multiple tool offset memories are used by a single tool is performed as follows:

(i) Make specification as follows by the use of the part program for tool information registration:

```
G122 ;
P··· L··· ;
T □□ △△ ;
T □□ △△ ;
T □□ △△ ;
T □□ △△ ;
T □□ △△ ;
T □□ △△ ;
T □□ △△ ;
T □□ △△ ;
T □□ △△ ;
T □□ △△ ;
P··· L··· ;
T···
  ⟩
G123 ;
```

□□ : Tool number

△△ : Tool offset memory number

As shown above, consecutively specify the tool numbers to be used and the offset memory numbers in the group to be specified by address P. Up to 5 offset memories may be used for one tool.

EXAMPLE:

```
G122 ;
P1 L7890 ;
T0101 ;
T0111 ;
T0121 ;
T0131 ;
T0141 ;
T0202 ;
T0212 ;
T0222 ;
T0232 ;
T0242 ;
P··· L··· ;
  ⟩
G123 ;
```

This is the display of tool life control group No. 1 of COM function on CRT screen at registration of tool information by the execution of the above program.

```
TOOL LIFE CONTROL    O1234 N1234

 TOOL GROUP 1 (LIFE 7890 COUNT)
     COUNTER       0

 T0101 ; END        T0232
 T0111              T0242
 T0121
 T0131
 T0141
 T0202 ; CUTTING
 T0212
 T0222

                                RDY
```

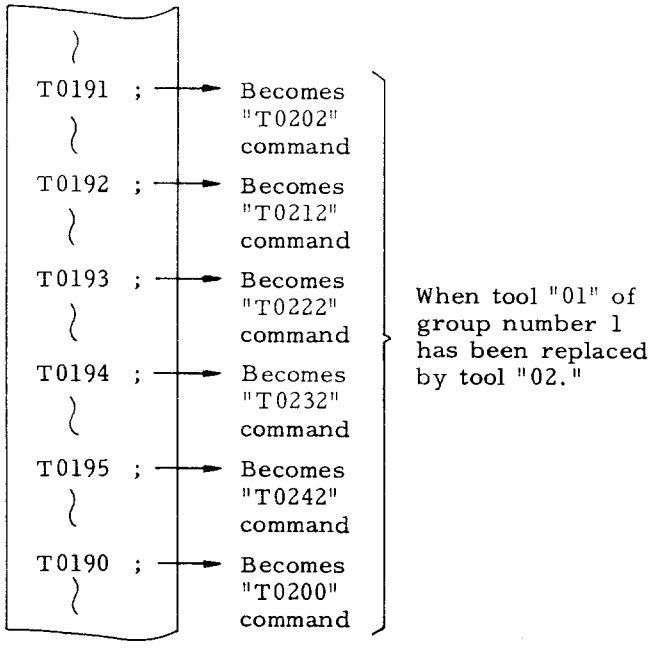
Fig. 2.72 Display of Tool Offset Memories Used for One Tool

ii) The part program
Use the following T commands:

Table 2.25

T□□90	The tool of the group specified in "□□" is given as T command.	Compensation is "00."
T□□91		Compensation is the compensation memory number registered first.
T□□92		Compensation is the compensation memory number registered second.
T□□93		Compensation is the compensation memory number registered third.
T□□94		Compensation is the compensation memory number registered fourth.
T□□95		Compensation is the compensation memory number registered fifth.

The T commands in the execution of the following program after the registration of tool information shown in EXAMPLE are as follows:



The display of tool life control group No. 1 of COM function on CRT screen at execution of the above program is as follows:

```
TOOL LIFE CONTROL    01234 N1234

  TOOL GROUP 1 (LIFE 7890 COUNT)
      COUNTER       0

  T0101 ;          T0232
  T0111 ;          T0242
  T0121 ;
  T0131 ;
  T0141 ;
  T0202 ;
  T0212 ;
  T0222 ;

                              RDY
```

Fig. 2.73

b. Tool life control in which work coordinate system setting (G50 T···) is used.

(i) Use of work coordinate system setting does not require to modify the program for tool information registration.

(ii) The work coordinate system setting command to be used in the part program should be as follows:

G50 T□□90 ;
└ Group number (1 to 19)

There are following restrictions:

- The tool number used is 01 to 30.
- The tool coordinate memory number to be used for each tool is as follows:

Table 2.26

Tool No.	Tool Coordinate Memory No.
01	51
02	52
03	53
⟩	⟩
30	80

(iii) When the command "G50 T □□ 90" is executed in the following programs, the control operates as shown in the table below.

Program for Tool Information Registration

```
O···
G122 ;
P1 L100 ;
T0101 ;
T0303 ;
T0606 ;
P··· L···
  ⟩
  ⟨
G123 ;
M30 ;
```

Part Program

```
O···
  ⟩
  ⟨
T0190 ;
G50 T0190 ;
  ⟩
  ⟨
```

Tool no. of Group no. 01	Operation by "G50 T0190"
01	"G50 T5100"
03	"G50 T5300"
06	"G50 T5600"

c. Tool life control and setting

The registration of tool information and the monitoring of the currently used tools are performed using the setting area. Shown below is the list of settings used for tool life control.

Note: "Registration of tool information not from the part program but by the writing of settings" and "the presetting of the number of tools by settings" are provided for special operations. However, these operations should not be performed in principle.

(6) List of Settings for Tool Life Control

a. Registration of group number for each tool

Table 2.27

Setting No.	Contents	Remarks
#8601	The group number of tool to be selected by "T01 * *" command.	Tool groups are "1 to 19." The tool whose life has expired has a minus value.
#8602	The group number of tool to be selected by "T02 * *" command.	
⟩	⟩	
#8650	The group number of tool to be selected by "T50 * *" command.	

* * indicates compensation number.

b. Registration of life of each tool group

Table 2.28

Setting No.	Contents	Remarks
#6161	The life of group number "1." (The number of machinable pieces)	The setting range of the number of machinable pieces is 1 to 9999 (units).
⟩	⟩	
#6169	The life of group number "9." (The number of machinable pieces)	
#6170	The life of group number "10." (Machinable hours)	The setting range of the number of machinable hours is 1 to 9999 (minutes).
⟩	⟩	
#6179	The life of group number "19." (Machinable hours)	

c. Registration of tool offset memory and tool number

Table 2.29

Setting	Contents	Remarks
TOFN01 (#8651)	The tool number which uses offset memory 01 (T * * 01).	Tool number is "1 to 50."
TOFN02 (#8652)	The tool number which uses offset memory 02 (T * * 02).	
⟩	⟩	
TOFN50 (#8700)	The tool number which uses offset memory 50 (T * * 50).	

d. Registration when multiple tool offset memories are used by a single tool

Table 2.30

Setting	Contents	Remarks
TOFO01 (#8701)	The registration of the offset designation number in the tool life control program of offset memory 01 (T * * 01).	The number not used in tool life control is "0." The number useb in tool life control program is 91 to 95 91 designation is is "1." 92 designation is "2." 93 designation is "3." 94 designation is "4." 95 designation is "5."
TOFO02 (#8702)	The registration of the offset designation number in the tool life control program of offset memory 02 (T * * 02).	
⟩	⟩	
TOFO50 (#8750)	The registraion of the offset designation number in the tool life control program of offset memory 50 (T * * 50).	

When the part program shown in sub-paragraph (5), a, (i) is registered, any one of 1 to 5, and 0 is stored here.

e. Monitoring of the currently used tool in each group

Table 2.31

Setting	Contents	Remarks
TG1CNT (#6181)	The number of pieces being machined by the currentlyused tool of group number "1."	The number of machinable pieces is set in #6161 to #6179.
⟩	⟩	
TG9CNT (#6189)	The number of pieces being machined by the currently used tool of group number "9."	
TG10CNT (#6190)	The use time of the currently used tool of group number "10."	The machinable hour is set in #6180 to #6189.
⟩	⟩	
TG19CNT (#6199)	The use time of the currently used tool of group number "19."	

Notes :

1. To use the tool life control feature, select the "T 4-digit designation" and the "offset memory addition" options.
2. When "90" through "95," or "99" is specified in the low-order 2 digits of the T command (4-digit designation) in the part program, the control executes the tool life control feature. A T-command other than above does not cause the execution of this feature and is processed as an ordinary T command. Hence, it is possible to specify the T command for tool life control and an ordinary T command in a single part program.
3. The maximum number of "tool number + offset memory number" pairs which can be registered in a single tool group is 16.
4. G122 and G123 commands should always be specified on a single block basis.
5. Between G122 command and G123 command, only the following should be specified:

   P: Group number.

   L: Life for each tool.

   T: Tool number and compensation memory number.

   Any other addresses cannot be specified.

6. The tool replacement completion input is accepted also when the tool replacement request output is not on (the usable tool is still left) if the automatic operation is in the hold state. Care should be used in the handling of this feature.

7. Tool life control is applied only to the T command in the part program. The tool operation by the manual intervention during run is not affected by this control.

8. The count time of the tool life control by use hour is held after the power-off. Precisely, however, the maximum of one minute of count time may be discarded between power-off and power-on. This is because the count time is held in units of a minute.

9. At the start of the tool information registration by G122 command, the control cancels all the registered contents before starting new registration. Hence, the registration of partial tool information is not allowed. Always register the entire tool information.

10. "G50 T□□91" to "G50 T□□95" may be specified for the work coordinate system setting command to set the work coordinate system added with the contents of offset memory. In this case, however, the deviation of machining start position or the like may be caused. This specification should not be performed, in principle.

11. For tool life control during use (group No. 10 to 19), do not use multiple tool offset memories for a single tool. (Refer to (5) Supplementary Explanation in 9.)

    In this case, if T- command is given for changing tool offset values after tool life expectancy has been reached, the tool will be exchanged for a new one causing an unexpected impact on the machine.

(7) List of alarms to be given by tool life control

Table 2. 32

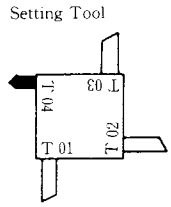
Alarm No.	Cause
150	G121 or G122 command is not specified on a single block basis.
151	The designation of group number P is not provided. Or a value other than 1 ≦ P ≦ 19 is designated.
151	The designation of life per tool L is not provided. Or a value other than 1 ≦ L ≦ 9999 is designated.
152	A value other than 1 ≦ tool number ≦ 50 is designated for the tool number.

Table 2.32 (Cont'd)

Alarm No.	Cause
152	A value other than 1 ≦ compensation memory number ≦ 50 is designated for the compensation memory number.
153	The tool information of the same group number is registered twice.
154	It was attempted to register more than 16 pairs of "tool number + cnmpensation memory number" in a single group number.
150	An address other than P, L, and T is designated in the tool information registration program.
159	More than 6 pairs of compenastion memory numbers are registered for a single tool.
155	The tool of the specified group number is not registered.
155	T□□92, T□□93, T□□94, or T□□95 is specified but the corresponding compensation memory number is not registered.
152	Zero or a value greater than 20 is designated in (group number) of T 90 through T 95, or T 99.
158	Some registered tool in the tool group designated in □□ of work coordinate system setting (G50 T□□90) has a tool number greater than "31."
156	Tool life control is designated in the control having no "T 4-digit designation and" offset memory addition" options.
157	The tool replacement request output is on.

EXAMPLE

Setting Tool



2.8.29 TOOL LIFE CONTROL (G122, G123)† (Cont'd)

Tool No.	Offset Memory	Type of Cutting
T01	01	Rough cutting
T02	02	
T03	03	
T04	04	Finish cutting

Programs for Tool Information Registration

```
O··· ;
G122 ;
P01 L10 ;
T0101 ;
T0202 ;
T0303 ;
G123 ;
M30 ;
```

When the above program is executed, T0101, T0202, and T0303 are registered as group number 1 and with the number of machinable pieces per tool being 10.

Part Program

```
O··· :
N0001 G50 X··· Z··· ;
G00 T0191 ;  ——①
     ( ROUGH CUTTING PROGRAM )
G00 X··· Z··· T0190 ;  ——②
N002 G50 X··· Z···
G00 T0404 ;
     ( FINISH CUTTING PROGRAM )
G00 T0400 ;
T0199 ;
M99 ;
```

When the above program is executed, blocks of ① and ② provide the following commands:

No. of Executions	Block shown by ①	Block shown by ②
1-10 times	G00 T0101	G00 T0100
11-20 times	G00 T0202	G00 T0200
21-30 times	G00 T0303	G00 T0300
31 ST time	Tool change request	

The processing after the completion of tool life control:

When the machining of T01, G02, and T03 has been completed and the tool replacement request output is turned on, input the value of compensation memory for the new tool then perform the following operations:

(i) Set "1" as the tool replacement completion number input (see (4), d.)

(ii) Turn on the tool replacement completion input (see (4), d.)

(iii) Perform the RESET operation.

2.8.30 MULTIPLE CORNERING (G111, G112)†

These commands are used to perform beveling and rounding on the taper and circular arc portions of a work.

G code	Meaning
G111	Multiple cornering on taper portion.
G112	Multiple cornering on circular arc portion.

These commands enable the control to perform beveling and rounding on taper and circular arc portions without making complex computation.

2.8.30.1 Taper Multiple Cornering (G111)

The following four operations may be specified in a single block:

Taper → Beveling / Rounding → Taper → Beveling / Rounding

The typical contours for which taper multiple cornering is specified are shown in Figs. 2.8.30.1 to 2.8.30.5 on pages 102 and 103.

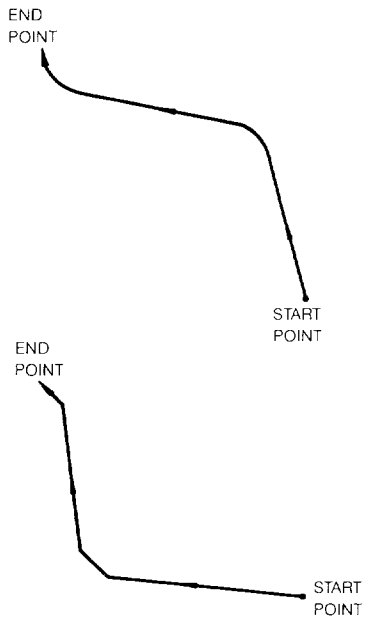
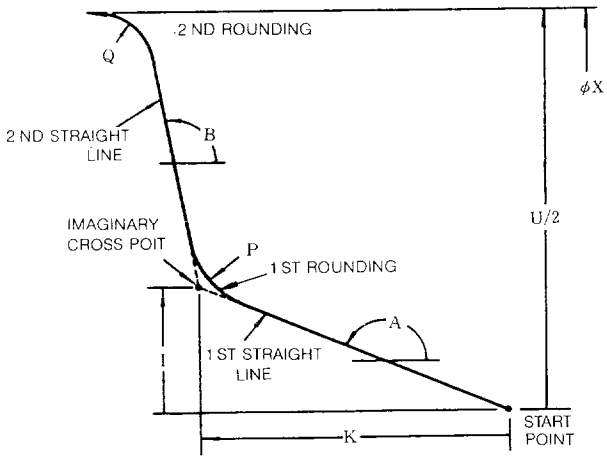


Fig. 2.74

(1) Command format for configurations for multiple cornering

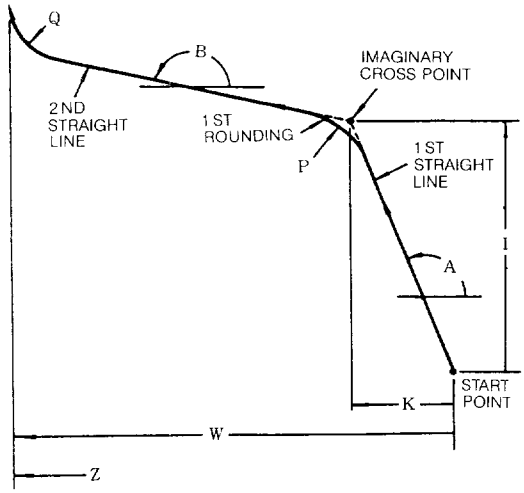
a.



G111 X(U)··· I··· A··· B··· P··· Q··· ;
G111 X(U)··· K··· A··· B··· P··· Q··· ;
G111 X(U)··· I··· K··· B··· P··· Q··· ;

Fig. 2.75

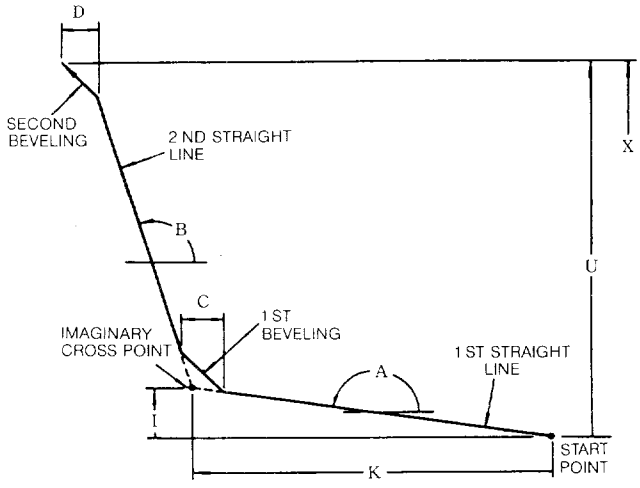
b.



G111 Z(W)··· I··· A··· B··· P··· Q··· ;
G111 Z(W)··· K··· A··· B··· P··· Q··· ;
G111 Z(W)··· I··· K··· B··· P··· Q··· ;

Fig. 2.76

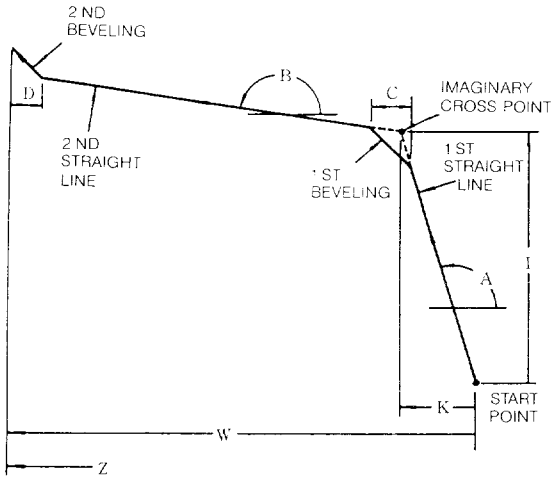
c.



G111 X(U)··· I··· A··· B··· C··· D··· ;
G111 X(U)··· K··· A··· B··· C··· D··· ;
G111 X(U)··· I··· A··· B··· C··· D··· ;

Fig. 2.77

d.




```
G111 Z(W)··· I··· A··· B··· C··· D··· ;
G111 Z(W)··· K··· A··· B··· C··· D··· ;
G111 Z(W)··· I··· A··· B··· C··· D··· ;
```

Fig. 2. 78

(2) Meaning of addresses

The following address words may be specified for the taper multiple cornering command. Simply specifying the contour determining address words permits the required operation.

Table 2. 33

Address Word	Contents	Unit
X (U)	X-axis end point coordinate (U: Increment from start point)	1 = 0.001 m or 1 = 0.0001 in.
Z (W)	Z-axis end point coordinate (W: Increment from start point)	
A	Move angle of the first straight line	1 = 0.001 deg
B	Move angle of the second straight line	
I	Virtual intersection between first and second straight lines. X-axis distance from start point (radius value).	1 = 0.001 mm or 1 = 0.0001 in.

Table 2. 33 (Cont'd)

Address Word	Contents	Unit
K	Virtual intersection between first and second straight lines. Z-axis distance from start point.	1 = 0.001 mm or 1–0.0001 in.
P	The first rounding radius (without sign).	
Q	The second rounding radius (without sign).	
C	The first beveling amount (without sign).	
D	The second beveling amount (without sign).	

(3) Designation of contours

a. Designate the contour as shown below:

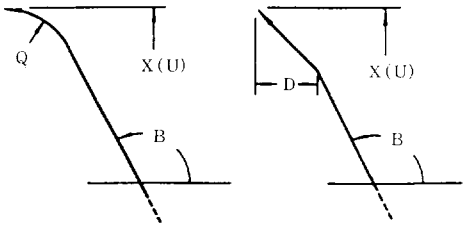
Table 2. 34

First Straight Line	A: First straight line move angle I: Virtual intersection X-axis distance from start point K: Virtual intersection Z-axis distance from start point	Specify two.
First Beveling or Rounding	C: First beveling amount D: First rounding radius	Specify either.
Second Straight Line	B: Second straighetline move angle. X(U): X-axis end point coordinate [(U): Increment from X-axis start or end point] Z(W): Z-axis end point coordinate [(W): Increment from Z-axis start or end point]	Specify two. However, the following combinations are not permitted: X and U Z and W
Second Beveling or Rounding	D: Second beveling value Q: Randius for second rounding Either D or Q should be commanded.	

b. The first rounding touches the first and second straight lines.

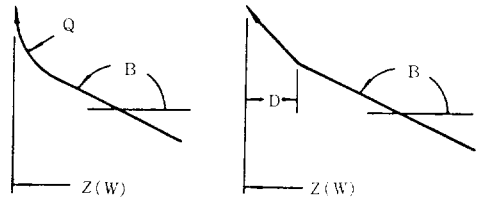
c. The second beveling and rounding depend on the designation of second straight line as shown below:

(i) The second straight line is specified with B and X(U).



ROUNDING TOUCHING THE STRAIGHT LINE PARALLEL TO Z-AXIS

(ii) The second straight line is specified with B and Z(W).



ROUNDING TOUCHING THE STRAIGHT LINE PARALLEL TO X-AXIS

d. The second beveling and rounding are preformed in the direction in which the second straight line advances. For details, refer to Table 2.35.

Table 2.35 Directions of Second Beveling

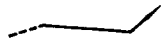
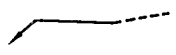
B command value for second straight line move angle	Beveling direction (+X, +Z)	Other conditions
B = 0 = −360.000 = 360.000	Beveling in X, Z positive direction 	First straight line moves in X positive direction.
	Beveling in X negative direction, Z positive direction	First straight line moves in X negative direction.
0 < B, B < 90.000 −360.000 < B, B < −270.000	Beveling in X, Z positive birection	
B = 90.000 = −270.000	Beveling in X, Z positive direction	First straight line moves in Z positive direction.
	Beveling in X positive, Z negative direction	First straight line moves in Z negative direction.
90.000 < B, B < 180.000 −270.000 < B, B < −180.000	Beveling in X positive, Z negative direction	
B = 180.000, = −180.000	Beveling in X positive, Z negative direction	First straight line moves in X positive direction.
	Beveling in X negative, Z negative direction 	First straight line moves in X negative direction.

Table 2. 35 Directions of Second Beveling (Cont'd)

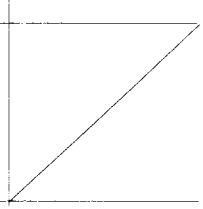
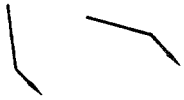
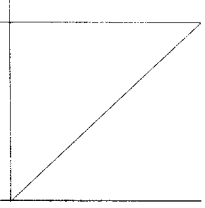
B command value for second straight line move angle	Beveling direction	Other conditions
180.000 < B, B < 270.000 −180.000 < B, B < −90.000	Beveling in X, Z positive direction	
B = 270.000 = −90.000	Bevelig in X, Z negative direction	First straight line moves in Z negative direction.
	Beveling in X negative, Z positive direction	First straight line moves in Z positive direction
270.000 < B, B < 360.000 −90.000 < B, B < 0	Beveling in X negative, Z positive direction 	

Table 2. 36 Directions of Second Rounding

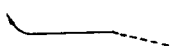
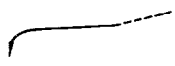
B command value for second straight line move angle	Rounding direction	Other conditions	
B = 0 = 360.000 = −360.000	Rounding in X negative, Z positive direction 	First straight line moves in X-axis negative direction	X (U) command may not be used.
	Rounding in X, Z positive, direction 	First straight line moves in X-axis positive direction	

Table 2. 36 Direction of Second Runding (Cont'd)

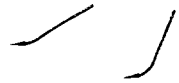
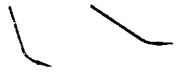
B command value for second straight line move angle	Rounding direction	Other conditions	
0 < B, B < 90.000 −360.000 < B, B < −270.000	Rounding in X, Z positive direction Rounding touching the straight line parallel to X-axis	Second straight line is specified with B, Z (W).	
	 Rounding touching the straight line parallel to Z-axis	Second straight line is specified with B, X (U).	
B = 90.000 = −270.000	Rounding in X, Z positive direction	First straight line moves in Z-axis positive direction.	Z (W) command may not be used.
	Rounding in X positive, Z negative direction	First straight line moves in Z-axis negative direction.	
90.000 < B, B < 180.000 −270.000 < B, B < −180.000	Rounding in X positive, Z negative direction  Rounding touching the straight line parallel to Z axis	Second straight line is specified with B, X (U).	
	Rounding touching the straight line parallel to X axis	Second straight line is specified with B, Z (W).	

Table 2.36 Direction of Second Rounding (Cont'd)

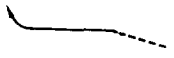
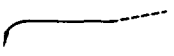
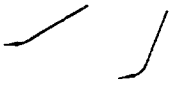
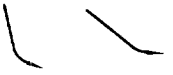
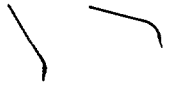
B command value for second straight line move angle	Rounding direction (+X, −Z)		Other conditions
B = 180.000 = −180.000	Rounding in X positive, Z negative direction 	First straight line moves in X-axis positive direction	X (U) command may not be used.
	Rounding in X, Z negative direction 	First straight line moves in X-axis negative direction	
180.000 < B, B < 270.000 −180.000 < B, B < −90.000	Rounding in X, Z negative direction Rounding touching the straight line parallel to X axis		Second straight line is specified with B, Z (W).
	 Rounding touching the straight line parallel to Z axis		Second straight line is specified with B, X (U).
B = 270.000 = −90.000	Rounding in X, Z negative direction	First straight line moves in Z-axis negative direction	Z (W) command may not be used.
	Rounding X negative, Z positive direction	First straight line moves in Z-axis positive direction	

Table 2.36 Direction of Second Rounding (Cont'd)

B command value for second straight line move angle	Rounding direction (−X, +Z)	Other conditions
270.000 < B, B < 360.000 −90.000 < B, B < 0	Rounding in X negative, Z positive direction  Rounding touching the straight line parallel to Z axis	Second straight line is specified with B, X (U).
	 Rounding touching the straight line parallel to X axis	Second straight line is specified with B, Z (W).

e. Supplementary description

(i) When all B, X(U), and Z(W) of the second straight line are specified, the first straight line may provide one of the A, I, and K commands.

(ii) The taper multiple cornering command specifies the first and second and straight lines by selecting addresses X, Z, I, K, A, and B. Hence, unlike the other G commands, X, Z, I, K, and B do not allow the omission of "0" specification. The specification of "0" have different meaning for X, Z, I, K, and B. "0" should always be specified.

Omission of address "0" specification of taper multiple cornering command.

Table 2.37

Address	"0" specification may be omitted/not omitted
X Z I K A B	May not be omitted.
P Q C D	May be omitted. (Cornering amount is "0")

(iii) When the second straight line is designated by X(U) and Z(W) commands, the second beveling and rounding are disabled. If this designation is attempted, an error is caused.

(iv) The combination of the first beveling and the second rounding or the first rounding and the second beveling is also available.

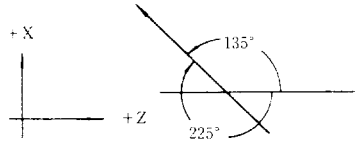
(v) When the first straight line designation addresses A, I, and K are all specified, A is ignored. The first straight line is created by I and K commands alone.

(vi) When the second straight line designation addresses B, X(U), and Z(W) are all specified and two of the first straight line designation addresses A, I, and K are further specified, B is ignored. The second straight line is created by X(U) and Z(W) commands.

(vii) How to specify straight line move angles A and B:

Specify a positive value for the counterclockwise rotational angle from Z-axis positive direction, and a negative value for the clockwise rotational angle from Z-axis positive direction.
(Specification range: $-360.000 \leqq A, B \leqq 360.000$)

EXAMPLE:

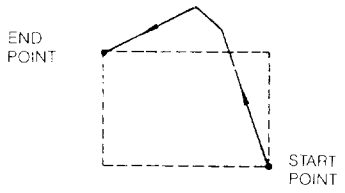


A 135 or A 225

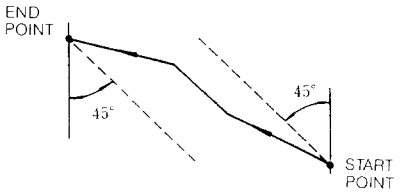
Notes:

1. G111 is nonmodal G code and is valid only in the specified block.
2. Addresses M, S, and T cannot be specified in the block specified with G111.
3. If the first beveling portion to be specified by address C of the block specified with G111 has the contour shown below, the specification is disabled:

(i) Outside the rectangle enclosed by the start and end points.



(ii) Between the straight line going from start point to end point at angle of 45° and the straight line going from end point to start point at angle of 45°.



The end point in the above diagram is the end point of the second straight line with no second beveling and rounding.

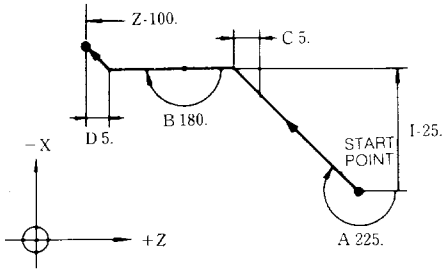
4. When G111 block is executed on a single block basis, the movement up to the end point is performed assuming the maximum of four blocks.
5. Writing G111 block in the buffer, the control unit performs all computations for the first and second straight lines and the first and second beveling and rounding. For some contours, the computation time requires more than 500 msec. If the move time in the preceding block is shorter than the computation time, the movement stops, sometimes causing undesirable effects on the cutting surface. To prevent the stop of movement due to the computation time, it is recommended to provide the buffering state (M93 command) before the four blocks before specifying G111 block command.

6. Some commands which assume that the 45-degree straight line, going from start point to end point, is the first line or second line may cause an error.

Example 1:

```
G111 Z-100. I-25. A225.
     B180.  C5.  D5.;
```

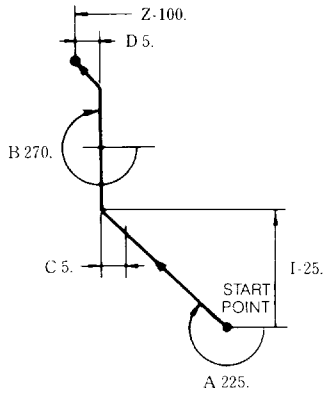
This command will result in error.



Example 2

```
G111 Z-100. I-25. A225.
     B270.  C5.  D5.;
```

This command will not cause an error.



7. List of alarms caused by incorrect G111 command

Table 2. 39

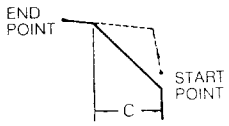
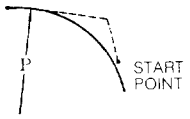
Alarm Code	Cause
140	Commanding one address of addresses B, X (U), Z (W) specifying second straight line.
140	Commanding two addresses of addresses B, X (U), Z (W) specifying second straight line. In addition to this, one or no address commanded among addresses A, I, K specifying first straight line.
140	Address C specifying first beveling and address P specifying first rounding commanded.
140	Address D specifying second beveling and address Q specifying second rounding commanded.
141	Angle for angle programming A, B by G111 out of range $-360 \leqq A, B \leqq 360$.
142	Ist beveling part outside the rectangle composed by start and end points.
142	Ist beveling portion between 45° straight lines of start to end points and end to start points.
143	No intersection between Ist and 2nd straight lines.
143	Ist straight line and second straight line on the same line.
144	M, S, T commanded in G111 block.
143	Command values for addresses A, I, K specifying first straight line are determined as follows, and programmed shape cannot be formed.

Command value for A	
−360.000, −180.000, 0, 180.000, 360.000	Address I commanded for specifying first straight line.
−27.000, −90.000, 90.000, 270.000	Address K commanded for specifying first straight line.

143	Command values for addresses B, X (U), Z (W) specifying first straight line are determined as follows, and programmed shape cannot be formed.

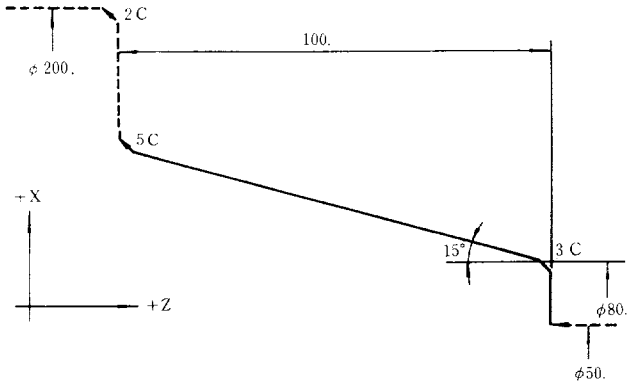
Command value for B	
−360.000, −180.000, 0, 360.000	Address X (U) commanded for second straight line.
−270.000, −90.000, 90.000, 270.000	Address Z (W) commanded for specifying second straight line.

Table 2. 39 (Cont'd)

Alarm Code	Cause
143	Command values for addresses C and D for beveling too large for the programmed shape. Operation cannot be made according to the command. 
143	Command values for addresses P and Q specifying radius for rounding too large for the programmed shape. Operation cannot be made according to the command. 
140	Commanding addresses X and Z specifying second straight line and Q and D specifying second beveling and rounding.

EXAMPLE

a. Taper combined beveling



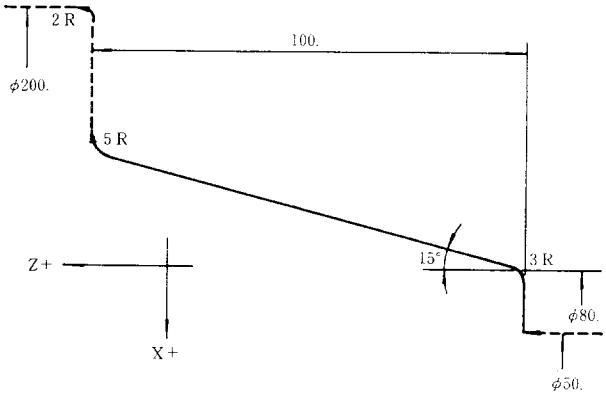
(G01 W··· ;)——φ50 command shown by broken line

G111 W-100. I15. A90. B165. C3. D5. ;

or G111 W-100. I15. K0 B165. C3. D5. ;

Command shown by solid line

b. Taper combined rounding



(G01 W··· ;)——φ50 command shown by broken line

G111 W-100. I15. A90. B165. P3. Q5. ;

or G111 W-100. I15. K0 B165. P3. Q5. ;

Command shown by solid line

(G12) X200. K-2. ;)—Command shown by broken line after the command shown by solid line

2.8.30.2 Circular Arc Multiple Cornering (G112)

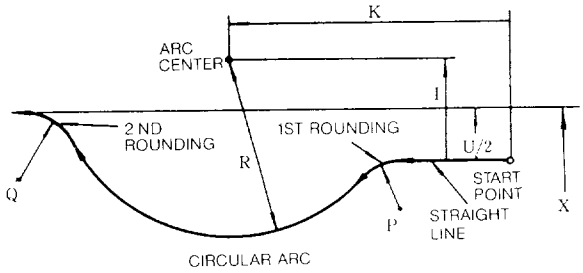
G112 be able to specify the following four operations in a single block:

Straight line → Beveling Rounding → Circular arc → Beveling Rounding

Depending on the direction of arc combined beveling/rounding or taper in turning combined beveling/rounding in facing may be executed.

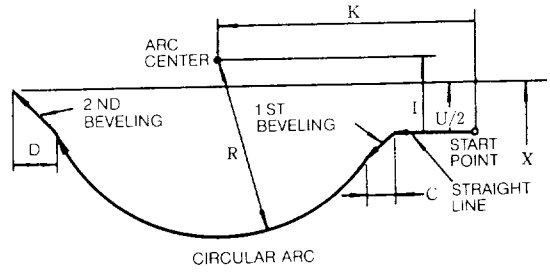
(1) Cutting configurations and command format

a. Arc combined rounding in turning



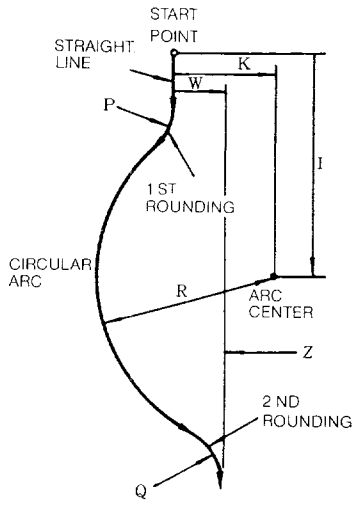
G112 X(U)···I ··· K··· P··· Q··· R··· ;

b. Arc combined beveling in turning



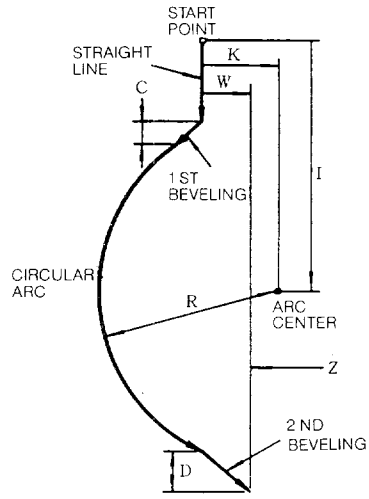
G112 X(U)··· I··· K··· C··· D··· R··· ;

c. Arc combined rounding in facing



G112 Z(W)···I ··· K··· P··· Q··· R··· ;

d. Arc combined beveling in facing



G112 Z(W)··· I··· K··· C··· D··· R··· ;

2. 8. 30. 2 Circular Arc Multiple Cornering (G 112) (Cont'd)

(2) Meaning of addresses

The word addresses for circular arc multiple cornering command are as shown below:

Table 2. 40

Address Word	Contents	Unit
X (U)	X-axis end point coordinates for arc combined rounding in turning (U: Increment from start point)	1 = 0.001 mm or 1 = 0.0001 in. (decimal point may be entered)
Z (W)	Z-axis end point coordinates for arc combined rounding in facing (W: Increment from start point)	
I	X-axis distance from arc center start point	
K	Z-axis distance from arc center start point	
R	Circular arc radius	
P	First rounding radius (without sign)	
Q	Second rounding radius (without sign)	
C	First beveling amount (without sign)	
D	Second beveling amount (without sign)	

(3) Designation of contours

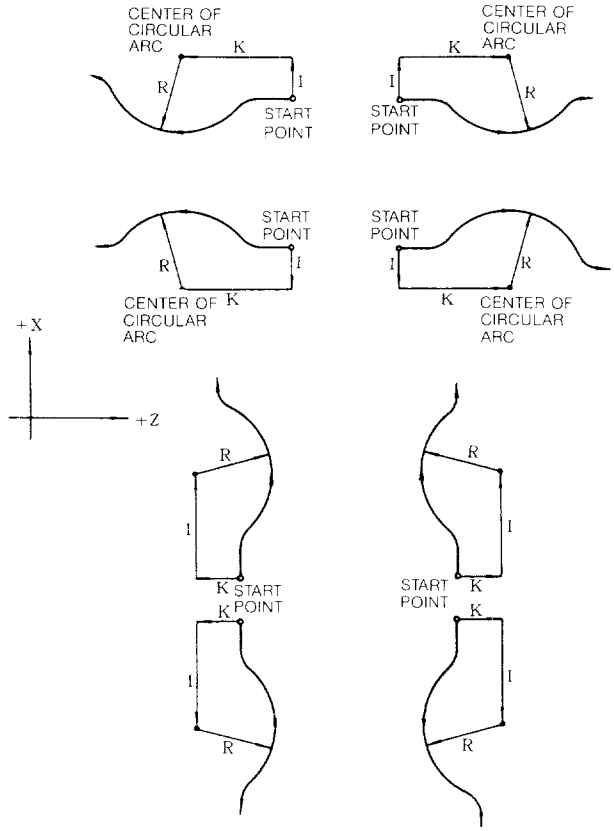
a. The contours of the portions to be subjected to circular arc multiple cornering are as shown below:

Table 2. 41

Straight Line	The straight line which is parallel to Z-axis (arc in turning) or X-axis (arc in facing) from the start point.
Circular Arc	The point from which the arc is circulated is designated by I and K commands from the start point.
First Beveling	The beveling which is performed in the dimensions designated by C command at the intersection of straight line and circular arc.

Table 2. 41 (Cont'd)

First Rounding	The rounding which is performed in the radius designated by P command in contact with straight line and circular arc.
Second Beveling	The beveling which is performed in the dimensions designated by D command at the intersection between the circular arc and the straight line parallel to Z-axis designated by X (U) command (circular arc in turning) or the straight line parallel to X-axis designated by Z (W) command (arc in facing).
Second Rounding	The rounding which is performed in the radius designated by Q command in contact with the circular arc and the straight line parallel to Z-axis designated by X (U) command (circular arc in turning) or the straight line parallel to X-axis designated by Z (W) command (arc in facing).

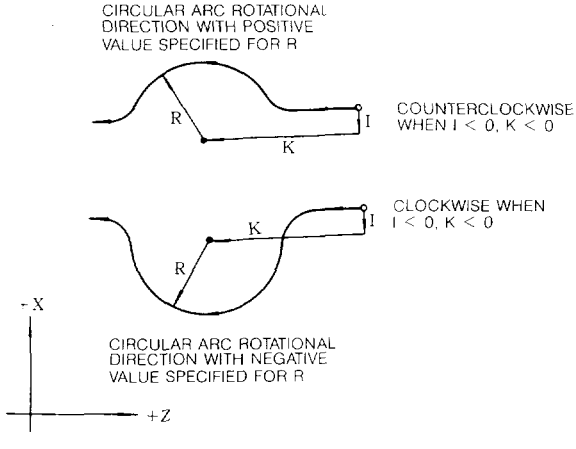


c. To supplement the above description, the following discriminants for determination of the circular arc cutting directions in the control unit are provided:

Table 2. 42

Values of I and K Commands	Circular Arc Rotational Direction	
	Circular arc in turning	Circular arc in facing
$I \geq 0$, $K \geq 0$	Counterclock wise (CCW) (Equivalent to G03)	Clockwise (CW) (Equivalent to G02)
$I \geq 0$, $K < 0$	Clockwise (CW) (Equivalent to G02)	Counterclock wise (CCW) (Equivalent to G03)
$I < 0$, $K \geq 0$		
$I < 0$, $K < 0$	Counterclock wise (CCW) (Equivalent to G03)	Clockwise (CW) (Equivalent to G02)

When a negative value is specified for the circular arc R, the circular arc cutting direction mentioned above may be inverted as follows:



d. Address words X(U) and Z(W) are used to make discrimination between radius measuring circular arc and front circular arc. They cannot be omitted if the end point and the start point are on the same coordinates ("U0" or "W0" should be specified). When the address words other than X(U) and Z(W) are omitted, the following results are obtained:

Table 2. 43

Address Word	Result
I	"I0" command is provided.
K	"K0" command is provided.
R	"R0" command and alarm are provided.
P Q C D	"0" command is provided. Beveling and rounding are not performed.

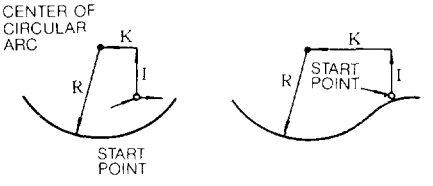
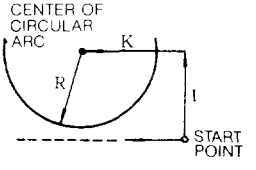
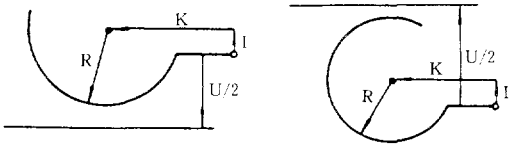
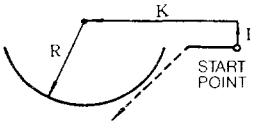
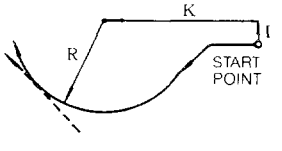
Notes :

1. G112 is a nonmodal G code and is valid only for the specified block.
2. The block specified with G112 does not allow the specification of M, S, and T.
3. When G112 block is executed on a single block basis, the movement up to the end point is performed assuming the maximum of four blocks.
4. When G112 is used for the finishing shape blocks G71 (stock removal cycle in turning), G72 (Stock removal cycle in facing), and G73 (pattern repeating) of multiple repetitive cycles the block specified with G112 is equivalent to five blocks.
5. Do not specify other codes to the G112 block; otherwise, an error will be caused.
6. After wiring G112 block to the buffer, the control unit performs all computations for the straight line, the circular arc, and the first and second beveling and rounding. For some contours, the computation time becomes more than 500 msec. If the move time in the preceding block is shorter than the computation undesirable effects on the cutting surface. To prevent the stop of movement due to the computation time, provide the buffering state (M93 command) before the four blocks before specifying B112 block command.
7. List of alarms to be given by G112 command error

Alarm Code	Cause
145	X(U) or Z(W) not specified.
145	Both X(U) and Z(W) specified.
145	R not specified. Or "0" specified.
145	I and K not specified. Or "0" specified for both.

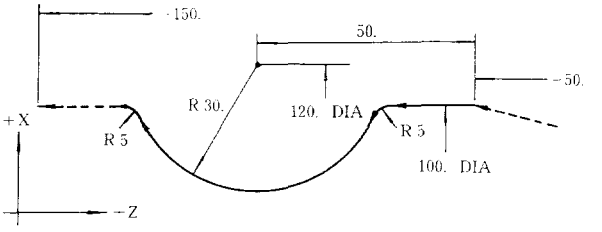
2. 8. 30. 2 Circular Arc Multiple Cornering (G112) (Cont'd)

Table 2. 44

Alarm Code	Cause
145	Both P and C specified.
145	Both Q and D specified.
144	M, S, or T specified.
146	Tool moves in the direction reverse to the center of circular arc from the start point. 
146	There is no intersection between circular arc and straight line. 
146	There is no intersection between circular arc and end point command. 
146	Beveling specified by C command cannot be performed. 
146	Beveling specified by D command cannot be performed. 

EXAMPLE:

a. Arc combined rounding

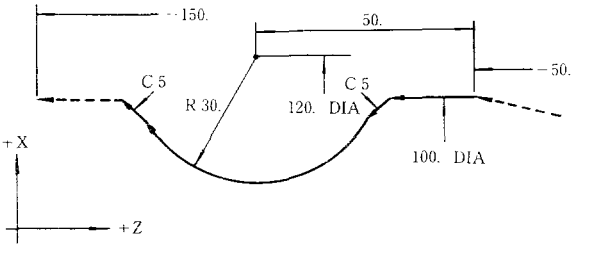


(G01 X100. Z-50. ;) ··· Shown by broken line before circular arc

G112 U0I10. K-50. P5. Q5. R30. ;

(G01 X-150. ;)··· Shown by broken line after circular arc

b. Arc combined beveling



(G01 X100. Z-50. ;)··· Shown by broken line before circular arc

G112 U0I10. K-50. C5. D5. R30. ;

(G01 Z-150. ;) ··· Shown by broken line after circular arc

2. 8. 31 ABSOLUTE/INCREMENTAL PROGRAMMING (G90, G91)

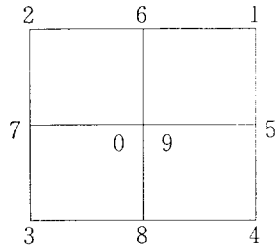
G code	Meaning
G90	Absolute designation
G91	Incremental designation

For the details of the G codes, see 2.3.5, " Absolute and Incremental Inputs."

2.8.32 GROOVE WIDTH COMPENSATION (G150, G151)*

Groove width compensation is to shift the coordinate for the groove tool width and to change the tool nose point by the G151 command, according to the tool width and control point set via the offset screen. This function allows groove machining without considering the tool width in the program.

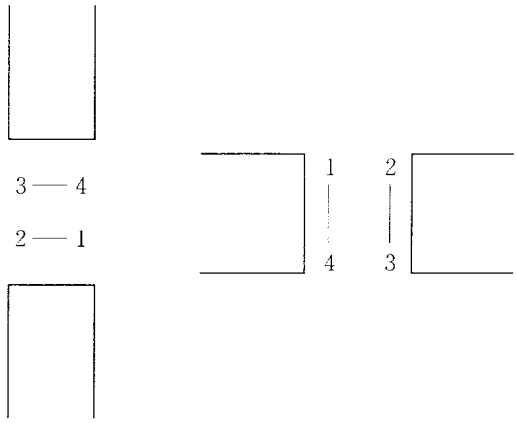
G150 ... Groove width compensation OFF
G151 ... Groove width compensation ON



Contrl point diagram

The compensation command shifts the tool nose point as follows. The compensation is canceled by the G150 command or by the T command.

Inner, outer diameter groove End face groove



Set the tool width in the offset numbers from OFS31, as follows.

(Example)

	Offset	Tool width
Tool No. 4	OFS04 X, Z, R, C	OFS34 X, Z
Tool No. 5	OFS05 X, Z, R, C	OFS35 X, Z
Tool No. 6	OFS06 X, Z, R, C	OFS36 X, Z

A control point is the tool nose point used in the automatic nose R function, which is indicated by "C" on the right corner of the offset screen.

When it is a diameter designation, the X-value end face groove is shifted by two times the tool width.

Commanding shift on both X and Z axes can be set in the parameter to trigger an alarm.

PM6018 D0 1: Alarm occurs upon X/Z axes shift in groove compensation.
0: Alarm does not occur upon X/Z axes shift in groove compensation.

Notes :

1. The program cannot be started under the G151 status. Always cancel G151 once before starting the program.
2. Commanding G151 without commanding the control point will cause an alarm, as the direction of the shift cannot be determined.
3. G150, G151 are members of group 13.
4. This G code cannot be used unless the new optional coordinate system setting function is provided.

(Example of groove width compensation program)

```
G00 X0;
G151 ; ..... Enable compensation
Z0;
X-20;        Shift the coordinate for the tool
             width.
G01 Z10;     The shift amount is set as offset
             number.
G00 X0;
G150; ..... Cancel compensation
Z0;
```

3. NC TAPE PUNCHING

3.1 TAPE CODE

3.1.1 LIST OF TAPE CODE

Both EIA code and ISO code are available for punching a paper tape.

EIA code (EIA RS-244-A).
ISO code (ISO R840).

Punching patterns according to these codings are shown in Table 3.1.

Before programming, select the code to be used. EIA and ISO codes cannot be punched mixedly through a tape.

3.1.2 EIA/ISO AUTO RECOGNITION

The control performs automatic recognition of EIA/ISO code. It recognizes the code punched on tape by reading the first EOB code in Label Skip state, and automatically adjusts the follow-on data to read by the recognized code. RESET operation[1] activates Label Skip state and cancels this function.

Note : The setting #6002D$_7$ can specify the code when NC Internal data is output (punched out), and does not affect the tape reading operation.

#6002 D$_7$	Meaning
= "0"	Code output
= "1"	Code output

3.2 PROGRAMMING

3.2.1 PROCESS SHEET

The programming is performed with the process sheet. It is recommended that the process sheet to match final specifications should be made by users, considering the readily perceived form and convenience for rewriting. Fig. 3.1 shows an example of the process sheet.

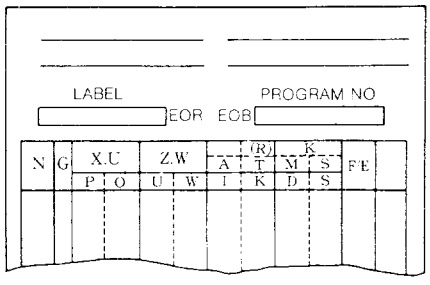
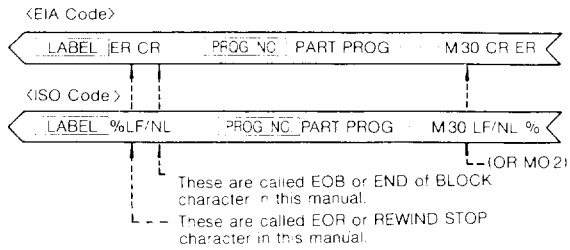


Fig. 3.1 An Example of the Process Sheet

3.2.2 GENERAL PROGRAM FORM

(1) A part program will be generally made in the following form.



(2) Any LABEL can be written at the beginning of tape to classify easily the tapes. In label skip function the control ignores the data from LABEL to the first EOB code. Therefore, the undesignated address or function characters can be used as LABEL. In addition, the modified code which disregards parity is also available.

(3) EOR code next to LABEL means the stop point of tape rewinding.

(4) Where storing NC tape data into memory, with the label skipped, the memory stores the data between the first EOB code and the next EOR code. Therefore, EOR code at the end of tape must not be omitted.

PRECAUTION IN PROGRAMMING

(1) A block ends with EOB (End-of-Block) character. EOB character is represented by CR in EIA code and LF/NL in ISO code. In this manual, mark ; is substituted for them to read easily this manual.

(2) A part program ends with the block including M02 (End-of-Program), M30 (End-of-Tape) or M99 (End-of-Program).

(3) When M02 or M30 is commanded, automatic operation[2] is stopped. In most cases, the control is reset, or rewinds the tape (or memory) automatically. As the details are determined by the machine, refer to the machine tool builder's manual.

[1] RESET operation means resetting the control by depressing the RESET key on the operator's control station or remotely.
[2] "Automatic operation" means operation in TAPE, MDI, or MEM mode.

Table 3.1 Tape Code

	EIA CODE									CHARACTERS		ISO CODE								
	8	7	6	5	4	○	3	2	1			8	7	6	5	4	○	3	2	1
			○			○				0				○	○		○			
						○			○	1		○		○	○		○			○
						○		○		2		○		○	○		○		○	
				○		○		○	○	3				○	○		○		○	○
						○	○			4		○		○	○		○	○		
				○		○	○		○	5				○	○		○	○		○
				○		○	○	○		6				○	○		○	○	○	
						○	○	○	○	7		○		○	○		○	○	○	○
					○	○				8		○		○	○	○	○			
				○	○	○			○	9				○	○	○	○			○
		○	○			○			○	a	A			○			○			○
		○	○			○		○		b	B			○			○		○	
		○	○	○		○		○	○	c	C	○		○			○		○	○
		○	○			○	○			d	D			○			○	○		
		○	○	○		○	○		○	e	E	○		○			○	○		○
		○	○	○		○	○	○		f	F	○		○			○	○	○	
		○	○			○	○	○	○	g	G			○			○	○	○	○
		○	○		○	○				h	H			○		○	○			
		○	○	○	○	○			○	i	I	○		○		○	○			○
		○		○		○			○	j	J	○		○		○	○		○	
		○		○		○		○		k	K			○		○	○		○	○
		○				○		○	○	l	L	○		○		○	○	○		
		○		○		○	○			m	M			○		○	○	○		○
		○				○	○		○	n	N			○		○	○	○	○	
		○				○	○	○		o	O	○		○		○	○	○	○	○
		○		○		○	○	○	○	p	P			○		○	○			
		○		○	○	○				q	Q	○		○		○	○			○
		○			○	○			○	r	R	○		○		○	○		○	
			○	○		○		○		s	S			○		○	○		○	○
			○			○		○	○	t	T	○		○		○	○	○		
			○	○		○	○			u	U			○		○	○	○		○
			○			○	○		○	v	V			○		○	○	○	○	
			○			○	○	○		w	W	○		○		○	○	○	○	○
			○	○		○	○	○	○	x	X	○		○		○	○			
			○	○	○	○				y	Y			○		○	○			○
			○		○	○			○	z	Z			○		○	○		○	
						○				Blank	NUL						○			
			○		○	○			○	BS		○					○			
			○	○	○	○	○	○		Tab	HT						○			○
	○					○				CR	LF/NL						○		○	
										–	CR	○					○	○		○
				○		○				SP		○		○			○			
					○	○		○	○	ER	%	○		○			○	○		○
		○	○	○	○	○	○			UC	–									
		○	○	○	○	○		○		LC	–									
*										–	(			○		○	○			
*										–	)	○		○		○	○			○
		○	○	○		○				+				○		○	○		○	○
		○				○				–				○		○	○	○		○
		○				○	○	○		○	:			○	○	○	○		○	
			○	○		○			○	/		○		○		○	○	○	○	○
		○	○	○	○	○	○	○	○	Del	DEL	○	○	○	○	○	○	○	○	○
	○	○	○	○	○	○	○	○	○	All Mark		○	○	○	○	○	○	○	○	○
	See Note 2.									#		○		○			○		○	○
*	○			○	○	○				*		○		○		○	○		○	
*	○				○	○	○			=		○		○	○	○	○	○		○
*	○		○	○		○				[		○	○		○	○	○		○	○
*	○		○			○		○		]		○	○		○	○	○	○		○
	○			○		○	○			$				○			○	○		
	○				○	○	○	○	○	@		○	○				○			
	○			○	○	○	○	○		?				○	○	○	○	○	○	○
		○	○		○	○		○	○	.				○		○	○	○	○	

Notes:

1. For the hole pattern of EIA code of the characters with an asterisk, the pattern shown in the table is standard. However, other patterns may be specified by parameters.
2. EIA code of character # can be designated by the parameter #6017.

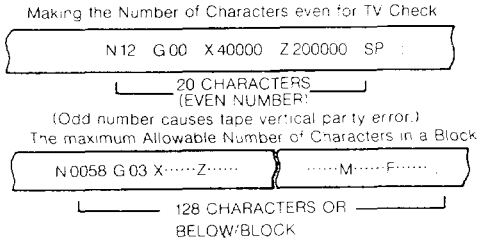
3. 2. 2 GENERAL PROGRAM FORM (Cont'd)

(4) The character specified on 2.1.2 Address and Function Characters should be used for programming, but others should not.

(5) Where the tape vertical parity check (TV check) is made, number of characters in a block must be even. If odd, it should be made even by using "SP" character.

(6) The disregarded characters such as "BS, Tab, SP, UC, LC and Del" should be avoided from the significant data area, if unnecessary.

The maximum allowable number of characters in a block is 128. The disregarded characters such as "Del, BS and Tab" are not included in them.



(7) SP (Space) character

SP character is usually disregarded when tape data is read in. However, in the following cases, SP is read and its function for providing space on the CRT screen is effective.

"SP" used in parentheses.

3. 3 NC TAPE

3. 3. 1 PAPER TAPE

Eight-channel paper tape for computers complying with JIS[1]-6243 is used as standard. The dimensions are 25.4 ±0.08 mm (1 inch) width and 0.108 mm (0.0042 inch) thickness.

It is recommended that the color of the tape is black or gray, but not that of high transparency. If the tape with high transparency is used, the tape reader may misread it.

3. 3. 2 NC TAPE PUNCH

NC tape must be punched out with the tape puncher for EIA code or ISO code according to contents of process sheet.

When punching the tape, at the beginning and the end of the tape, provide the feed holes part needed for the tape feeding. Where the punched tape is wound on the reel† of tape reader, the feed holes part will be 70 cm in length.

3. 3. 3 NC TAPE CHECK

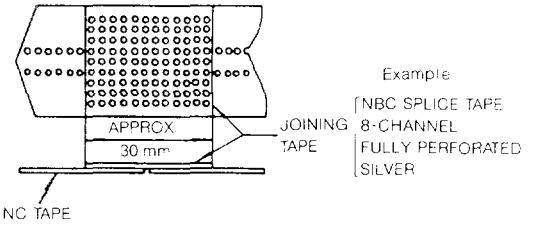
NC tape can be checked by using the following function.

- Machine lock
- M function lock†
- Dry run
- Single block operation

3. 4 NC TAPE HANDLING

3. 4. 1 SPLICING NC TAPES

To splice NC tapes, stick a joining tape (0.08 mm thickness) with sprocket holes, or fully perforated joining tape on the one side of the spliced NC tape. Before using the spliced NC tape, make sure that the sprocket holes are in position. The joining part of tapes should not be extremely thick, and do not use the rigid adhesive agent without flexibility.



Splicing of NC Tape

3. 4. 2 KEEPING OF NC TAPE

For life expectancy of NC tape, the following handling is recommended.

- When keeping NC tape, avoid moisture and oil.
- Do not handle the tape with oil-stained gloves.

Properly kept tapes will permit 300 times of reading and rewinding.

1 Japanese Industrial Standard

4. STANDARD NC OPERATOR'S STATION WITH CRT CHARACTER DISPLAY

4.1 PUSHBUTTONS, KEYS, AND LAMPS

Fig. 4.1 shows an overall view of NC operator's station with 9" CRT display with keyboard on right side of CRT. The names and functions of operator devices are as follows.

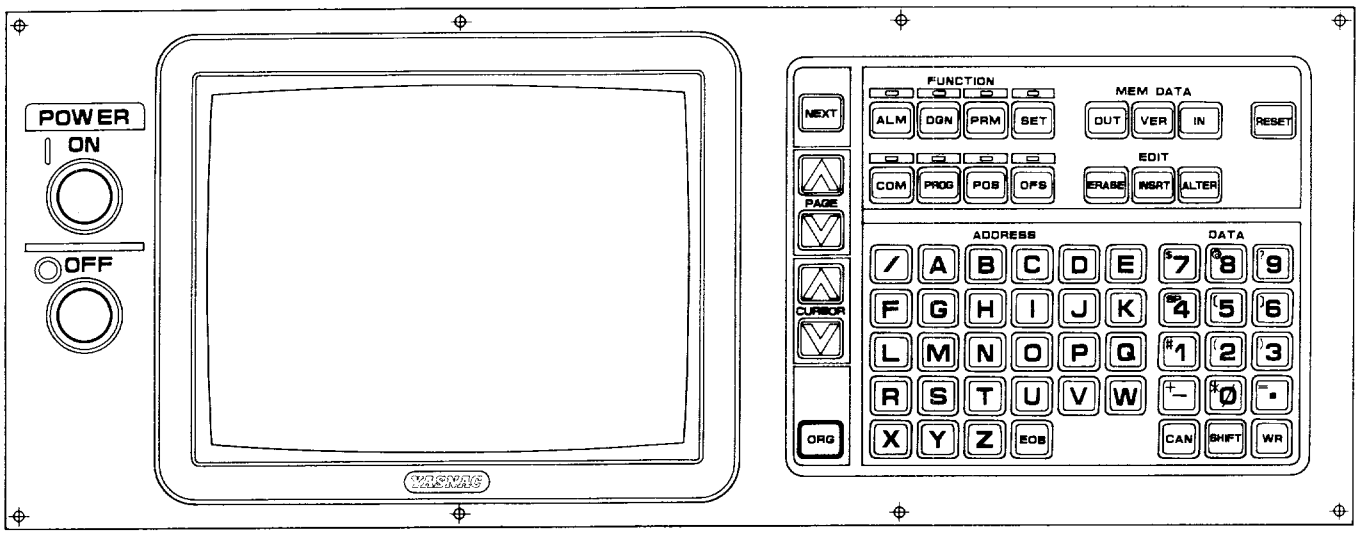


Fig. 4.1 Standard NC Operator's Station with 9" CRT Character Display (Keyboard on Right Side of CRT)

NC operator's station with 9" CRT display (keyboard below CRT) is also available. Operation is the same with both types. Instructions given in this manual are for operator's panel with keyboard on right side of CRT.

4.1.1 POWER ON/OFF PUSHBUTTONS

· POWER ON pushbutton

To turn on the power for the control: Depress the pushbutton first to turn on the control power and depress it again to turn on the servo power. Push this button to recover the servo power after an emergency stop.

· POWER OFF pushbutton

To turn off the power for the control: Depress it to turn off both the servo and control powers.

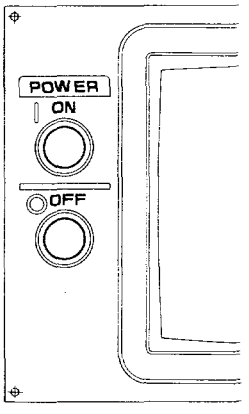


Fig. 4.2

4.1.2 CRT CHARACTER DISPLAY

According to each operation, this display indicates the alpha-numerical data in a regular size, double-size and quadruple-size of the regular size.

Braun tube size: 9 inches

Maximum number of characters:
32 characters x 16 lines =
512 characters (at regular size)

Indicating characters:

Numerals - 0 through 9, -, ., +

Alphabetic characters - A through Z

Special code - ; (EOB), / (slash), etc.

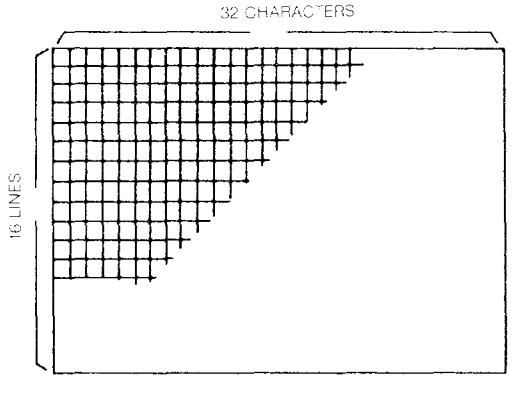


Fig. 4.3 Braun Tube

4.1.3 FUNCTION KEYS

The key selects one of eight functions for the operation of the display and MDI. Pushing a key makes it effective and light up.

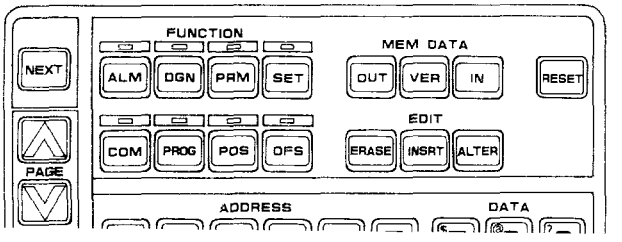


Fig. 4.4

· ALM (Alarm) key:

Select this key for display of alarm and status codes. The function becomes effective when the power is turned on or an alarm occurs.

· DGN (Diagnosis) key:

Select this key for display of input/output signal status.

· PRM (Parameter) key:

Select this key for display or writing-in of parameters.

· SET (Setting) key:

Select this key for display or writing-in of setting data.

· COM (Command) key:

Select this key for display or writing-in (MDI) of the command data for automatic operation.

· PROG (Program) key:

Select this key for display or writing-in of a part program.

· POS (Position) key:

Select this key for display of various current positions.

· OFS (Offset) key:

Select this key for display or writing-in of tool offset values.

4.1.4 ADDRESS KEYS

These keys are to designate an address character when writing in various data.

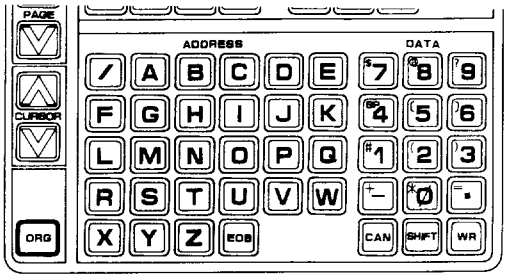


Fig. 4.5

Note: Special characters

/ (Slash) key: For an optional block skip command.

EOB (EOB) key: For the block end command. On the CRT display, ";" is displayed instead of "EOB."

4.1.5 DATA KEYS

These keys consist of 15 keys in total, such as 0 through 9, - (minus) . CAN SHIFT WR , and can be used for writing-in of such all numeral values as tool offset value setting data, parameter data, and so on, in addition to command value.

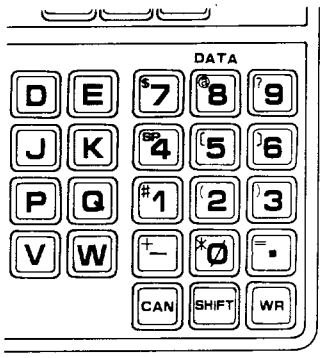


Fig. 4.6

Notes:

*0 to ?9 key
- (minus) key : For input of numerical data

. (decimal point) key: For input of decimal point

CAN (Cancellation) key:
For cancellation of the numeric value or address data erroneously keyed.

WR (write) key:
For storing address data by address keys and data keys into buffer storage.

SHIFT (shift) key:
Depressing SHIFT key after depressing *0 to ?9 , - or . key makes the display turn into * to ? , + , = which are written on the upper left corner of the keys. These special characters are used in user macro.

4.1.6 NEXT KEY

The NEXT key is used for special purpose and expanding function in display or writing data.

- Writing of additional tape in EDIT mode.
- Display of specified number in DGN function.
- For other special purpose and expanding function.

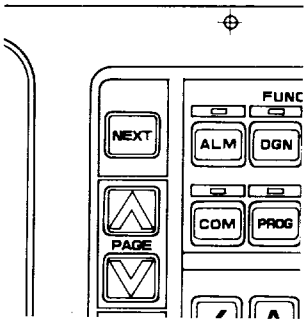


Fig. 4.7

4.1.7 PAGE KEYS

The PAGE key is used to display the next page or the previous page when CRT display is regarded as page.
For example, when a bundle of tool offset values are displayed by OFS key, this key is pushed to display the next bundle of tool offset values, which just looks like opening the pages of a book.

(1) Depressing PAGE ↓ key displays the next page.

(2) Depressing ↑ PAGE key displays the previous page.

(3) Keeping the PAGE key depressed makes the page step automatically forward or backward.

4.1.8 CURSOR KEYS

The CURSOR control key is used to move the cursor. For example, when a page of parameter data are displayed by PRM key.

(1) Depressing [CURSOR ↓] key moves the cursor backward.

(2) Depressing [↑ CURSOR] key moves the cursor forward.

(3) Keeping the cursor control key depressed makes the cursor move automatically forward or backward.

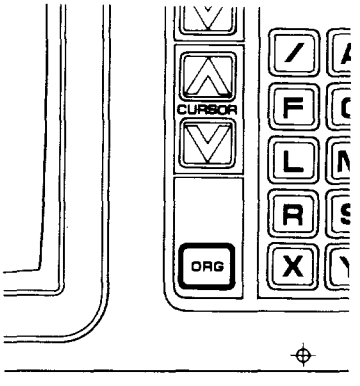


Fig. 4.8

4.1.9 ORG (ORIGIN) KEYS

The ORG key is used to set the current position of the machine tool as the zero point of coordinate system.
The origin setting can be made for each axis.
The reference coordinate system means the coordinate system which is set by G92 command or the automatic coordinate system setting.

ORG key is used for the following operation.

- Reset of current position (Position External/ Absolute)
- Reset of operation time

4.1.10 EDIT KEYS

These keys are for editing a stored part program.

ERASE key: Used for erasure of data in storage.

INSRT key: Used for insertion of data in memory.

ALTER key: Used for alteration of data in memory.

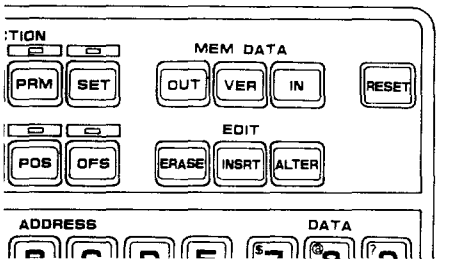


Fig. 4.9

4.1.11 MEM DATA (MEMORY DATA) KEYS

TAPE KEYs are to start the tape operation except in the automatic operation mode. They are effective only in the EDT mode.

(1) OUT key

This key is to start outputting various data in memory through data I/O interface.

(2) IN key

This key is to start storing various data into memory through tape reader or data I/O interface.

(3) VER key

This key is to start verifying between memory data and punched tape data.

4.1.12 RESET KEY

This key resets the control.

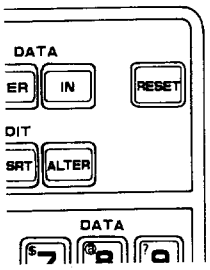


Fig. 4.10

Operations to be executed by this RESET key are:

- Move command cancel
- Buffer register clear
- Alarm code release if the cause is eliminated
- Tool offset cancel
- Auxiliary function cancel
- Label skip function ON
- Memory pointer rewind
- Sequence number reset
- RST signal transmission
- Resetting G codes

Refer to 2.8.1, " List of G Codes and Groups."

The following will not be affected by operating the RESET key.

- Current position values of each axis.
- F commands
- S, T and B commands
- Tool offset values, setting data, parameter data

Note: Depressing the RESET key or the remote reset pushbutton is defined as "Reset operation" in this manual.

4.2 POWER ON/OFF OPERATION

4.2.1 TURNING ON POWER

Check the machine before turning on power, referring to the machine tool builder's manual for details. Operations after completion of inspections are as follows.

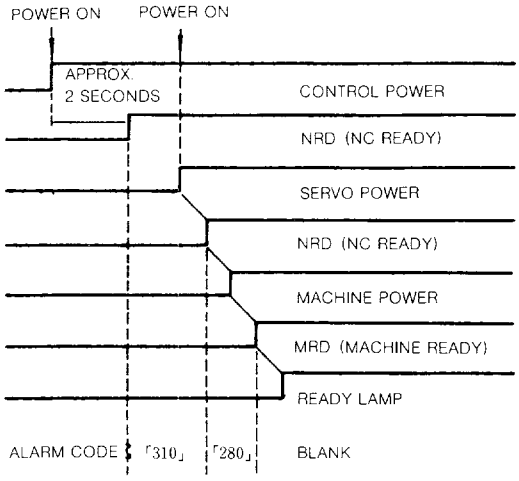


Fig. 4.12 Sequence of Turning on Operation

(1) Depressing the POWER ON pushbutton to turn on the control power. The internal timer will be read in about two seconds. Then the servo power is ready for turning on, which is shown by alarm code "310."

(2) Depress the POWER ON pushbutton again to turn on the servo power. The NRD (NC READY) signal is sent out when the NC power is normally supplied.

(3) When the NRD signal turns on the machine power, and the MRD (MACHINE READY) signal returns back to the control, the READY lamp will be lit.

4. 2. 2 TURNING OFF POWER

Depressing the POWER OFF pushbutton causes both the servo and control powers to be turned off simultaneously. However, for more stable operation, use the following procedure.

(1) First depress the EMERGENCY STOP pushbutton to cut off the servo power. The NRD (NC READY) signal is interrupted, which usually results in turning the machine power, too.

(2) Depress the POWER OFF pushbutton to cut off the control power.

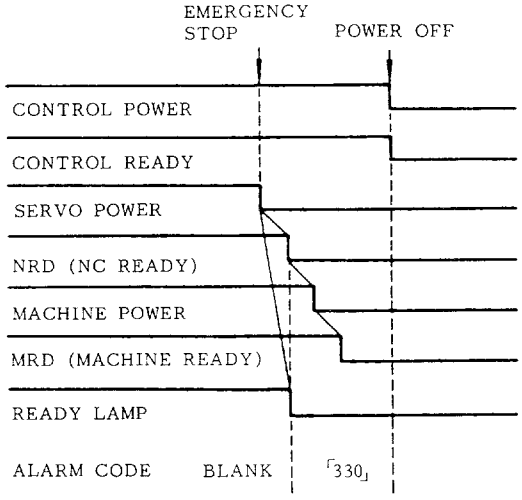


Fig. 4. 13 Sequence of Turning off Operation

4. 3 DISPLAY AND WRITING OPERATION

4. 3. 1 CONSTANT DISPLAY

The following display is made on both the top and bottom on the displayed picture of CRT, irrespective of the FUNCTION key currently selected.

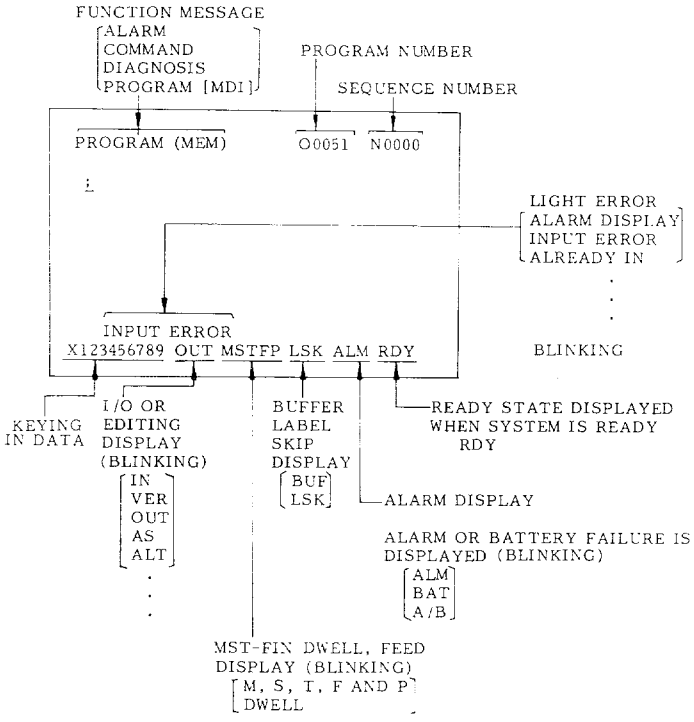


Fig. 4. 15

(1) Function message

Any one of the following eight function messages corresponding to the function key is displayed at the top of CRT display.

ALARM	COMMAND
DIAGNOSIS	PROGRAM
PARAMETER	POSITION
SETTING	OFFSET

(2) Program No.

O and 4 digits of program No. under execution is constantly displayed at the top of CRT display irrespectively of function key.

(3) Sequence No.

N and 4 digits of program No. under execution is constantly displayed at the top of CRT display irrespectively of function key.

(4) Display of keying data

Up to 32 characters of keyed in data can be displayed at one time. The data is processed by using ERASE key, INSRT key, ALTER key, etc.

(5) Display of I/O and editing (blinking)

The following messages are flickerlingly displayed during loading of punched tape, address search or edition.

"IN" ··· loading tape

"VER" ··· verifying tape

"OUT" ··· punching tape out

"AS" ··· searching address

"ALT" ··· altering data in EDIT mode

"INS" ··· inserting data in EDIT mode

"ERS" ··· erasing data in EDIT mode

(6) Display of MST-FIN signal waiting, dwelling and feeding

"M" ··· waiting for FIN signal of M command

"S" ··· waiting for FIN signal of S command

"T" ··· waiting for FIN signal of T command

"F" ··· feeding
"R" is displayed at rapid traverse

"P" ··· loading tape

"DWELL"
··· dwelling

M, S, T, F and P are displayed independently each other.

(7) Display of the state of buffer full and label skip

"BUF" ··· displayed at completion of advance reading

"LSK" ··· displayed at label skip on

(8) Display of alarm (blinking)

Alarm continues to be displayed flickeringly until the cause is removed and RESET operation is made.

"ALM" ··· indicates alarm state occuring

"BAT" ··· indicates battery alarm occuring

"A/B" ··· indicates both of alarm and battery alarm occuring

(9) Display of ready state

"RDY" ··· indicates the system is normal and the control is operable.

(10) Display of light errors (flickering)

The messages shown below indicate light errors which occur in keying or searching operation. Differing from the alarm codes, these error messages are cleared by depressing some key. (Generally CAN key)

"INPUT ERROR!" ···Format error of keyed-in data

"ALREADY IN!" ···The same number of part program has been stored already.

"EDIT LOCK ON!" ···Editing operation is made with Edit Lock on

"MEMORY OVER!" ···Part program to be stored is beyond memory capacity

"PROGRAM OVER!"
···Registered number of part program is beyond 99 (basic) or 199 (option)

"NOT FOUND!" ···Desired data has not been located.

4.3.2 COMMAND DATA DISPLAY

(1) Depress COM key.

Anyone of the following three digits appears.

a. Command data (COMMAND)

b. Repetition number of subprogram (SUB PROG. NESTING)

c. State of tool life control (TOOL LIFE CONTROL)

(2) The above display steps forward or backward by depressing PAGE ↓ or PAGE ↑ one by one.

4. 3. 2. 1 Command Data Display

The display shows the block data under execution or just prior to execution in which compensation calculations have been completed. The conditions of the data to be displayed is as follows.

(1) The data shows the contents of the active register during an automatic operation or a feed hold.

(2) While the control is stopped at a block end, the contents of the buffer regiter are displayed. If the buffer register blank (BUF is not displayed), the contents of the just executed block are displayed.

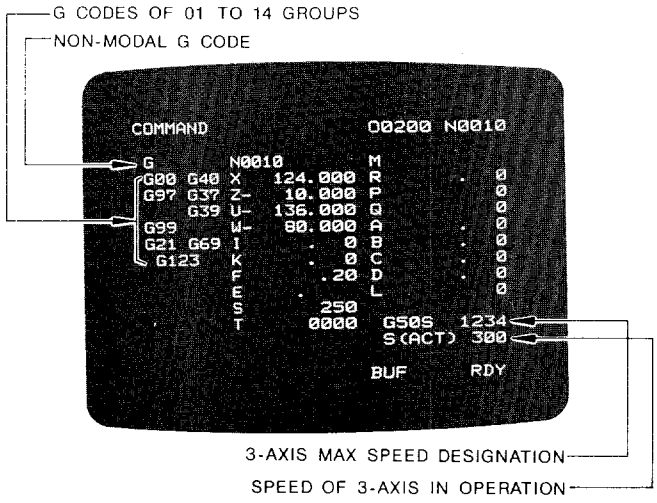


Fig. 4. 16 Command Display

4. 3. 2. 2 Display of Subprogram Run Status (SUB PROG. NESTING)

When the program being executed is in the subprogram called by M98 (subprogram call command), the following information is displayed:

CALL: The program number (O to NO) and sequence number (N to NO) specified with M98 (subprogram call command)

START: The program number (O to NO) and number (N to NO) of the subprogram called by M98

LOOP: The remaining number of repetitions of the subprogram by L □□□□ (subprogram repeat command, □□□□ indicates the number of repetitions)

NEST: The order in which subprogram multiple call commands are called.

Example of Subprogram Run Status Display

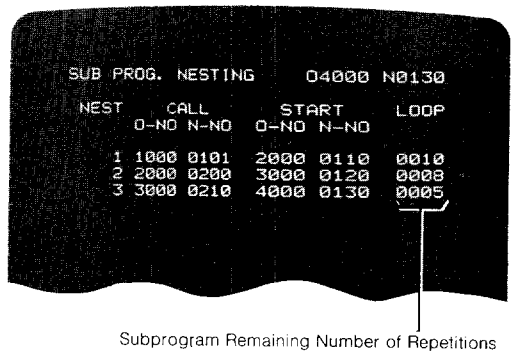
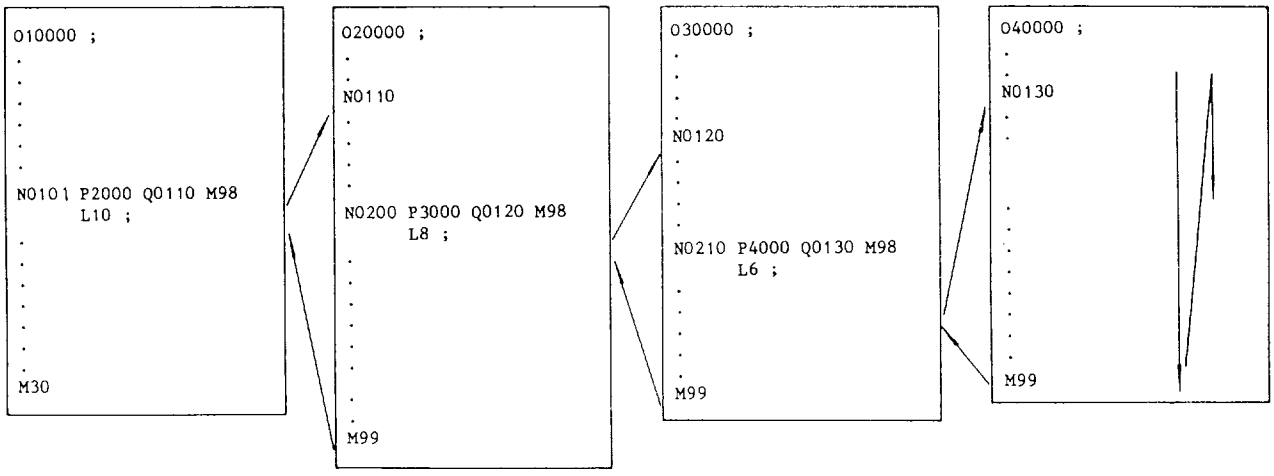


Fig. 4. 17



CRT screen displays that the subprogram has executed the 3rd level one time and entered into the execution of 3rd time of the 3rd level

4. 3. 2. 3 Display of Tool Life Control Use Status (TOOL LIFE CONTROL)

The following information is displayed for the status (e.g. condition, use) of each tool in each tool group under the tool life control feature:

LIFE: The life of the tool group displayed on the screen

Groups 1 to 9: The number of machining operations.

Groups 10 to 19: Machining time.

COUNTER/TIMER: The number of machining operations/machine time of the currently used tool.

T codes corresponding to the tools registered in the group displayed on screen are all shown.

T□□**; END
T□□**; END
} The tool whose life has expired.

T□□**; CUTTING → The tool currently used

T□□**;
.
.
.
.
T□□**;
} The tool to be used.

```
TOOL LIFE CONTROL     01234   N1234

  TOOL GROUP  1 (LIFE 7890 COUNT)
      COUNTER   1234

  T01**;  END          T21**
  T03**;  END          T26**
  T06**;  CUTTING      T29**
  T09**;               T31**
  T11**;
  T13**;
  T16**;
  T19**;
```

Fig. 4. 18 Example A of Tool Life Control Use Status

```
TOOL LIFE CONTROL     01234   U1234

  TOOL GROUP  12 (LIFE 7890 MIN.)
      TIMER        1234 MIN.

  T04**;  END          T24**;  END
  T08**;  END          T28**;  CUTTING
  T10**;  END          T30**;
  T12**;  END          T32**;
  T14**;  END          T36**;
  T18**;  END          T38**;
  T20**;  END          T40**;
  T22**;  END
```

Fig. 4. 19 Example B of Tool Life Control Status

4. 3. 3 WRITING IN BLOCKS AND DISPLAYING CONTENTS BY MDI

4. 3. 3. 1 Multi-block Writing and Operation in MDI Mode

(1) Multi-block writing in MDI mode

By the following operations, a maximum of 10 lines of data may be written in MDI mode.

a. Select MDI mode.

b. Depress the PROG function key.
··· PROGRAM (MDI) is displayed on the screen.

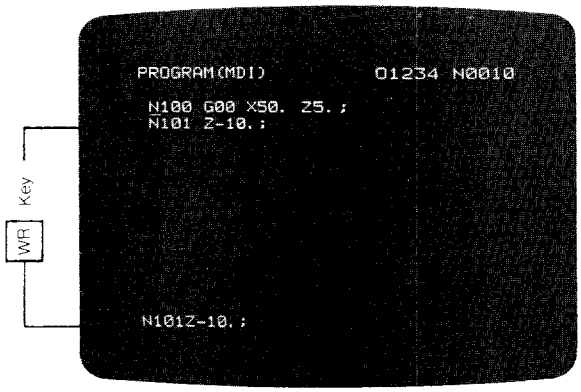
c. Depress the RESET key.
The buffer for MDI is emptied.

d. The part program is written by the use of the address key and data key. As shown below, the keyed data is written to the bottom line on CRT screen from left to right. The maximum number of characters that can be written at a time is 32. If the data is comprised of 32 characters or less, it may be keyed in over multiple words or blocks. However, when the 10th character is keyed in, the normal display at the right of this line is blanked.

e. Depress the WR key.
The keyed data is stored in the MDI buffer. The blanked display is restored to normal.

f. Up to 10 lines of the part program for MDI operation may be written by repeating the operations in d. and e. above.

4. 3. 3. 1 Multi-block Writing and Operation in MDI Mode (Cont'd)



Referred to as "the data which has, just been entered" (32 characters maximum).
Enter [N], [1], [0], [1], [Z], [−], [1], [0], [.], [EOB] in this order.

Note: The depression of the EOB key displays ";."

Fig. 4. 20

(2) Editing MDI data

The [CURSOR ↓], [CURSOR ↑], ERASE, INSRT, and ALTER keys permit editing multi-block data written in. Address (word) pointed to by the cursor will be edited. The [CURSOR ↓] and [CURSOR ↑] keys move the cursor forward and backward.

a. ERASE key: When this key has been depressed, the whole word designated is erased.

b. INSRT key: This key inserts the data which has just been entered next to the word the cursor points to.

c. ALTER key: This key replaces the word which the cursor points to with the data which has just been entered.

d. WR key: This key appends the data which has just been entered at the end of the program displayed.

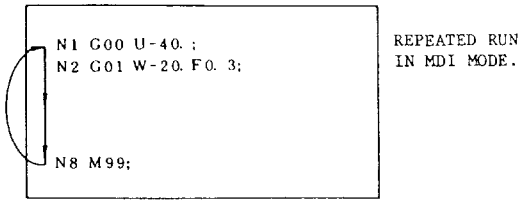
In MDI mode, only one screen currently displayed may be edited. Unlike EDIT mode and MEM mode, the display and edit of multiple screens cannot be performed. When the RESET key is depressed, the stored programs are all erased.

(3) Operation in MDI mode

a. Depressing the CYCLE START button in MDI mode can automatically execute the part programs stored in the MDI buffer. When the PROG function is active, the cursor is displayed at the head of the block currently executed. When the execution of all part programs is completed, the part programs and the CRT displays are erased.

b. If "M99" is written at the end of a part program, this program is executed repeatedly. The repetition may be stopped by depressing the FEED HOLD then RESET.

c. While a program is being run, the PROG function need not be active. Depressing POS can display the current values on the CRT.



4. 3. 3. 2 Display in Memory Run Mode (PROGRAM [MEM])

The part program being executed in memory run mode (MEM mode) may be displayed by the following operations:

(1) Select the MEM mode.

(2) Depress the PROG function key. On the CRT screen, the cursor is positioned at the head of the block currently executed. The cursor moves to the next block when its execution is started.

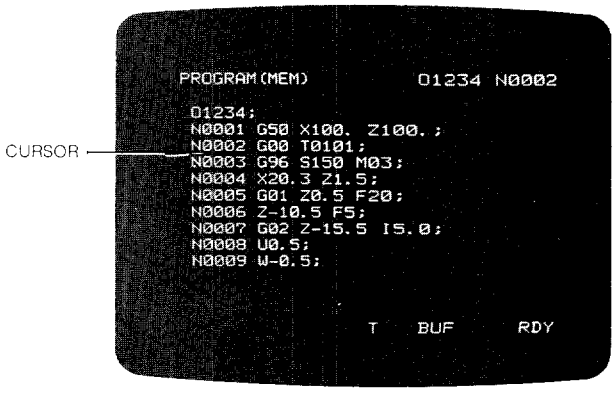


Fig. 4. 21 Display of Part Program in Memory Operation

Up to 10 lines may be displayed at a time. When execution of the ninth has been completed, the next page appears with the tenth line of the last screen appearing at the top.

4.3.3.3 Display in EDIT Mode

See 4.6 EDIT.

4.3.3.4 Address Search

Search continues until a data (character string) held on tape or in the memory which coincides with the data (character string) entered through the NC operator's station. The contents of tape will be searched in TAPE mode and those of the part program memory in MEM or EDIT mode.

(1) Operation (MEM, EDIT mode)

a. Select MEM, or EDIT mode.

b. Depress the PROG function key.

c. Depress the RESET key. "LSK" appears and the pointer returns to the top of the program number in MEM mode.

d. Search is performed in one of the following three methods:

(i) Key-in the word (an address data) to be searched. If the leading zero of the data is omitted, the search operation is still possible.

(ii) Key-in only a single character without data. This permits searching the character read first.

(iii) Depress the NEXT key then key in any data (less than 32 characters) to be searched. In this case, the search operation is performed exactly according to the keyed data (character string or numeral string), thus disabling the omission of leading zero.

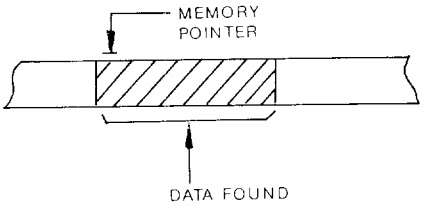
e. Depress the CURSOR ↓ key. Search starts. "AS" blinks during search.

f. When the NEXT key is depressed, depress it again to cancel the pattern search function.

(2) Completion of search

a. When the search is completed, "AS" will disappear.

(i) In MEM or EDIT mode, the pointer of the part program memory points to the data of block found (indicated by the cursor). In all cases, only search will be performed but neither BUF display nor advance reading will be performed.



b. "AS" disappears and "NOT FOUND!" appears on the CRT if the desired data is not found. This message will disappear when you depress a key (CAN normally) of the control station.

(3) Remarks

a. Commands encountered during search will be ignored even if they are modal commands.

b. On Cycle Start after search, the data of a block which the pointer points to will be read in and executed.

4.3.3.4 Address Search (Cont'd)

(4) Search of program number

The address search function also permits to search a part program out of those stored in the memory.

a. Select MEM or EDIT mode.

b. Depress the PROG function key.

c. Depress the RESET key.

d. Enter the program number "O □□□□." Leading zero can be omitted.

e. Depress the [CURSOR ↓] key.

The designated program number will be searched. The result of search is as described in (2). In MEM mode, you may depress the CYCLE START button immediately after completion of search to start automatic operation from the beginning of the program.

4.3.4 CURRENT POSITION DISPLAY

The current position of X-, or Z-axis can be displayed at any time in all modes. Operating procedure is as follows.

(1) Depress POS key.

One of the following will be displayed on the CRT screen.

a. POSITION EXTERNAL
b. POSITION ABSOLUTE
c. POSITION INCREMENT
d. POSITION
e. PROGRAM RETURN
f. DISTANCE TO LIMIT
g. PULSE COUNTER
h. ERROR PULSE
i. COMMAND PULSE

(2) Depress [PAGE ↓], [PAGE ↑] keys to select the page including any of the above.

Note: Page including error pulse or command pulse will be displayed when SYSTEM No. switch is set at 4.

4.3.4.1 Position [EXTERNAL]

(1) The current value to be displayed in EXTERNAL is the accumulated value of the tool movement from the position reset to "0" by the ORG key.

(2) How to reset POSITION [EXTERNAL]

Display the POSITION [EXTERNAL] screen on the CRT, select the axis by the address key, and depress the ORG key. The display of the selected axis becomes "0." This display reset operation is always valid even during time when the movement is being made by the automatic run of part program.

Notes:

1. When parameter POSEXT (#6005D5) is set to "1," the value displayed in POSITION EXTERNAL becomes the same value as displayed in POSITION ABSOLUTE .

2. Regardless of the state of parameter POSEXT, the value displayed in the external current value display (option) is the same as displayed in POSITION [EXTERNAL]. Hence, resetting the external current value display automatically causes the reset of POSITION EXTERNAL .

3. The "value in the current value display of the equipment" described in 2.8.22, Work Coordinate Multi-Shift G50T and 5.2.3, Measured Workpiece Value Direct Input refers to this value in POSITION EXTERNAL.

4. The display lock feature is valid for POSITION [EXTERNAL].

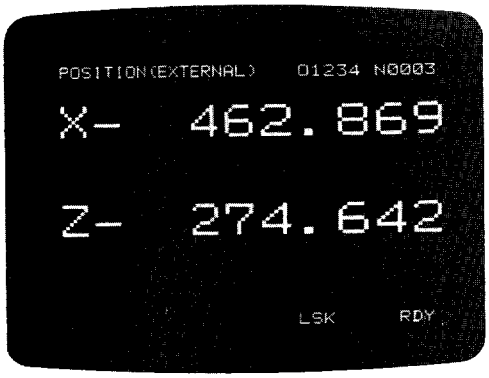


Fig. 4.22 Display of POSITION [EXTERNAL]

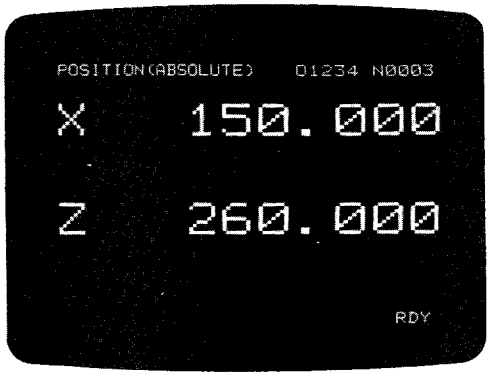


Fig. 4.23 Display of POSITION [ABSOLUTE]

4.3.4.2 Position [ABSOLUTE]

(1) The current value displayed in POSITION ABSOLUTE indicates the tool position on the coordinate system provided by coordinate system settings. Coordinate system setting is performed in the following cases:

a. The execution of G50 coordinate system setting.

b. The operation of automatic coordinate system setting (option).

c. The current value reset operation by the ORG key (see (2) below).

d. The execution of G50T work coordinate system setting.

(2) How to reset POSITION [ABSOLUTE]

The reset operation by the ORG key described above is as follows:

Display the POSITION ABSOLUTE screen on CRT, select the axis by the address key, and depress the ORG key. The display of the selected axis becomes "0." However, this display reset operation is valid only in the manual operation modes (RAPID, JOG, and STEP (or HANDLE)). Depressing of the ORG key is invalid during operation or in the buffer full state.

Note: If the display lock is on, the display of POSITION [ABSOLUTE] is not locked.

4.3.4.3 Position [INCREMENT]

Displayed in this mode are:

- In automatic mode, distance to the end point of the block at every moment.
- In manual mode, distance to the position where manual operation is to start.
- The increment display in manual mode will be cancelled in automatic operation mode.

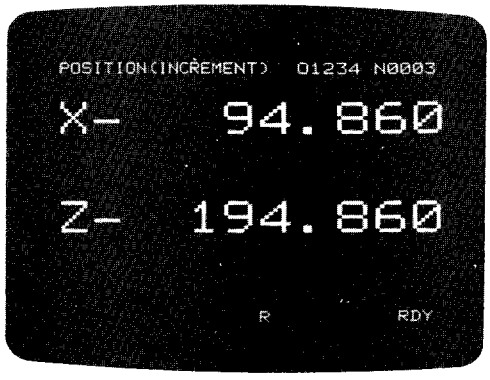


Fig. 4.24 Display of POSITION INCREMENT

4.3.4.4 Position

(1) In POSITION, all positions are collectively displayed.

(2) POSITION MACHINE displays the tool current position on the coordinate system with the reference point returned by the reference point return feature being "0." The data on the following features are defined on this coordinate system:

4. 3. 4. 4 Position (Cont'd)

a. Stored stroke limit.

b. Stored stroke limit as arranged by tool.

c. Leadscrew error compensation.

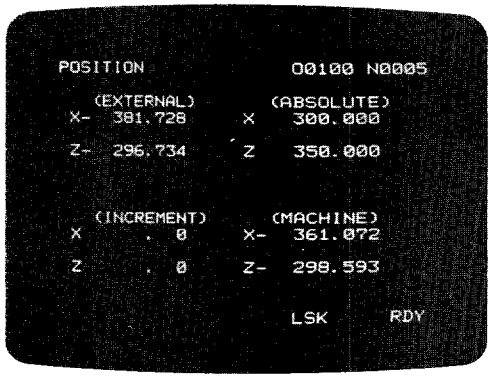


Fig. 4. 25 Display of POSITION

(3) Clear position

The [X] [Z] " [ORG] " key can be operated in the position screen, to display "0" as the external current positions. The external current position can be displayed as "0" by the following operation.

1 [X] " [ORG] " .... Change the X-axis value of the external current value display to "0."

2 [Z] " [ORG] " .... Change the Z-axis value of the external current position display to "0."

3 [X] [Z] " [ORG] " .... Change the X and Z axes values of the external current position display to "0."

The value other than the external current positions does not change.

The same operation can be performed on the external current position screen.

4. 3. 4. 5 Program Return

PROGRAM RETURN displays the information necessary for program restart. For details, see 6.2.6, " Program Restart."

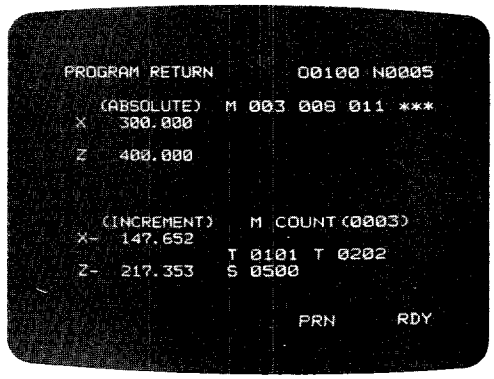


Fig. 4. 26 Display of PROGRAM RETURN

4. 3. 4. 6 Stored Stroke Limit †

(1) STORED STROKE LIMIT displays the remaining number of pulses in the four directions of X-axis plus/minus and Z-axis plus/minus from the tool current position to each boundary of the first, second, and third limit areas.

The setting of STORED STROKE LIMIT and the display of remaining number of pulses:

a. STORED STROKE LIMIT
X-axis plus direction boundary value

b. STORED STROKE LIMIT
X-axis minus direction boundary value

c. STORED STROKE LIMIT
Z-axis plus direction boundary value

d. STORED STROKE LIMIT
Z-axis minus direction boundary value

} Set by parameter

① The value of X-axis plus remaining number of pulses

② The value of X-axis minus remaining number of pulses

③ The value of Z-axis plus remaining number of pulses

④ The value of Z-axis minus remaining number of pulses

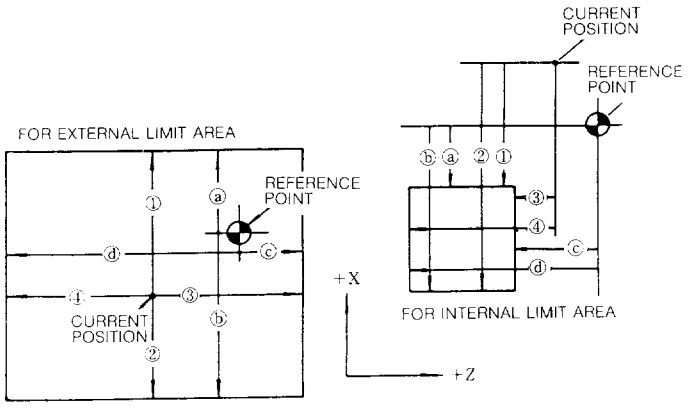


Fig. 4. 27

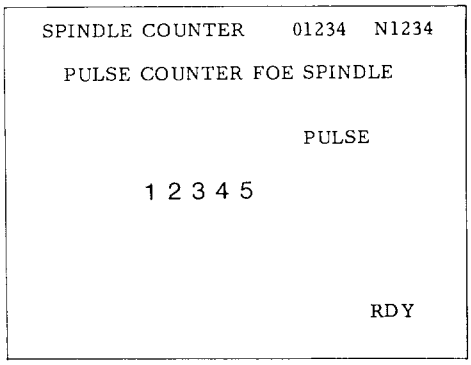


Fig. 4. 29 Display of No. of Spindle PG Pulses

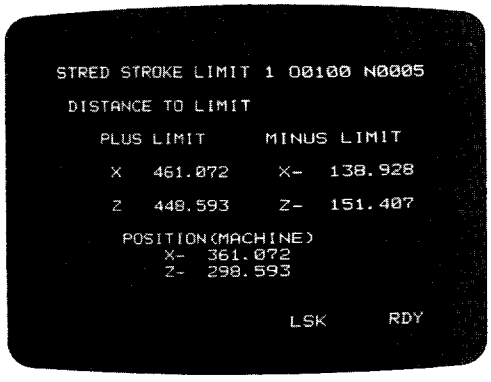


Fig. 4. 28 Display of Remaining No. of Pulses in First Stored Stroke Limit

(2) The display shown above will correspon to 1st, 2nd, and 3rd prohibited area.

4. 3. 4. 7 Spindle Counter

SPINDLE COUNTER displays the number of spindle PG pulses from the "spindle indexing origin" during the execution of the spindle indexing feature. The display is performed on a "1" = 1 pulse basis.

4. 3. 4. 8 No. of Servo Lag Pulses Display (ERROR PULSE)

ERROR PULSE displays the difference between the momentarily changing command position and the tool current position. The display is performed on a "1" = 1 pulse basis (the minimum move unit).

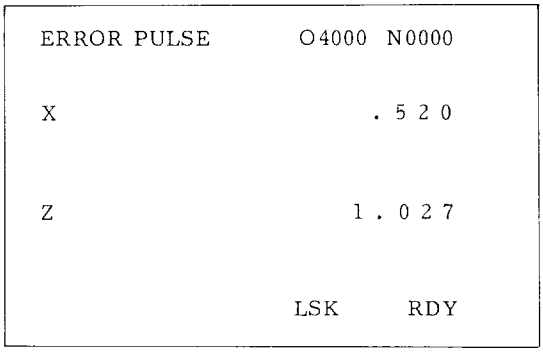


Fig. 4. 30 Display of No. of Servo Lag Pulses

4. 3. 4. 9 Command Pulse Accumulation Register Display (COMMAND PULSE)

COMMAND PULSE displays the contents of the command pulse integration register (SMC register) in the control unit. This screen is displayed only when the system number switch is "4."

The SMC register is set to "0" when the power is turned on and keeps adding command pulses until the power is turned off.

Note : When a value is set to parameter XSMCB (#6658: for X-axis) or ZSMCB (#6659: for Z-axis), the value obtained by subtracting the value set above from the content of the command pulse integrating register is displayed.

4. 3. 4. 9 Command Pulse Accumulation Register Display (COMMAND PULSE) (Cont'd)

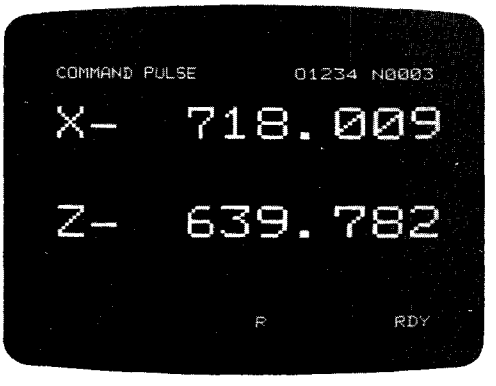


Fig. 4. 31

4. 3. 5 DISPLAYING AND WRITING TOOL OFFSET DATA†

The tool offset amount is stored in the offset memory in the unit. Regardless of modes, the tool offset amount may be displayed and written any time including the automatic operation time.

(1) Display of tool offset amount

The display of tool offset amount and other offset memory contents is performed by the following operations:

a. Select the OFS function key.

b. Key-in a two-digit tool offset number like 0, 1, and then depress [CURSOR ↓] or [CURSOR ↑].
The tool offset amount and the tool nose radius are displayed in five pairs including the tool offset number of the keyed value. The cursor is positioned to the designated tool offset number.

c. The preceding tool offset number may be designated by depressing [CURSOR ↓] or [CURSOR ↑]. If the operation is performed outside the range of the tool offset numbers displayed on one screen, the following or preceding five pairs of tool offset amounts are displayed.

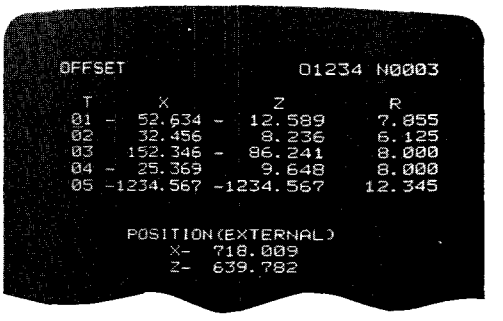


Fig. 4. 32 Display of Tool Offset Amount

d. The screen of the following or preceding five pairs of tool offset amounts may be displayed by depressing [PAGE ↓] or [PAGE ↑] key. In this case, the cursor is positioned to the first of the displayed tool offset numbers.

e. When the options shown below are selected, designating the corresponding offset memory number by depressing [PAGE ↓] or [PAGE ↑] key or by two-digit value key-in operation can display each data.

(i) Work coordinate system shift option:
Work coordinate system shift amount.

(ii) Work measuring value direct input option:
Tool coordinate data.

(iii) Tool wear compensation option:
Tool wear amount.

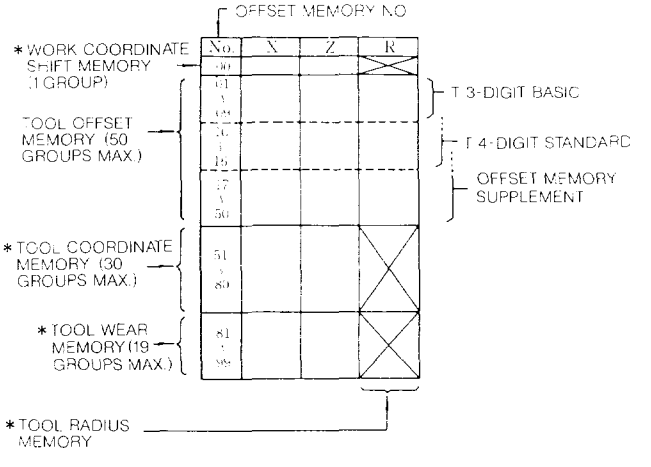


Fig. 4. 33

(2) Writing of tool offset amount

The writing of the offset amount and other offset memory contents is performed by the following operations:

a. Depress the OFS function key.

b. By the operation of PAGE key and CURSOR key or keying in two digits designate the tool offset number to be written.

c. Using the address key and the data key, key in the address data to be written. In this case, the meaning depends on the address as shown below:

Table 4.1

Address	Axis Designation	Meaning
X	Write to X axis	Writing of an absolute value. Namely, the value is written to memory without change.
Z	Write to Z axis	
U	Write to X axis	Writing of an increment. Namely, the current value is added to the preceding value and the result is written to memory.
W	Write to Z axis	
R	Tool nose radius	Writing of an absolute value.

d. Depress the WR key.

The address data keyed in is written to the tool offset memory according to the meaning of the data.

e. Repeat the operations of c. and d. and b. through d. to write all necessary tool offset amounts.

f. When the options shown below are selected, if the corresponding offset memory number is designated at the operation of d., the data may be written to memory. However, address R has no significance and therefore should not be designated.

(i) Work coordinate system shift option: Work coordinate system shift amount.

(ii) Work measuring value direct input option: Tool coordinate data.

(iii) Tool wear compensation option: Tool wear amount.

Notes:

1. The contents of offset memory are retained after the power is turned off.
2. Regardless of modes, the writing by the above operations is always possible including automatic run time.
3. The tool offset amount rewritten in automatic run is made valid from reading of the command of the next block. For the tool offset amount of the currently executed block or the block stored in the prefetch buffer, the value before change is used.
4. When the data shown below is rewritten, the timing in which the data value is made is as follows:

   (i) Work coordinate system shift amount: The time when G50, G50T, or other coordinate system setting is performed next.

   (ii) Tool coordinate data: The time when G50T is specified next.

   (iii) Tool wear compensation amount: The time when T99ΔΔ is specified next.

5. The writing operation of the above tool coordinate data is performed when the PST INPUT button is not depressed. If this button is depressed, the data to be written is completely different from the above data. For details, see 6.2.3 Work Measuring Value Direct Input.

6. The contents of offset memory are all erased by the following operations:

   (i) Depress the OFS function key.

   (ii) Key-in 0 - 9 9 9 9 , and depress the ORG key. The contents in offset memory are all erased.

4. 3. 6 DISPLAYING AND WRITING SETTING DATA (SETTING)

With this unit, the setting data is stored in the internal memory. According to the contents of this memory, a particular function is turned on/off. These contents are also used for the control constants of functions. For details, refer to Appendix 1, List of Setting Numbers. The display and write of setting data are always enabled including the time when automatic run is being performed.

The setting data is of the following two types:

(1) Setting data of bit display format.

(2) Setting data of decimal display format.

4. 3. 6. 1 Setting Data of Bit Display Format

Setting numbers "#6000 through #6004" have the setting data of bit display format. Each number has 8-bit information of "D7 through D0," each bit displaying on/off of the corresponding function.

(1) Display of setting data (bit display format)

Setting data is displayed by the following operations:

a. Select the SET function key.

b. Key-in 4-digit numeric setting number and depress [CURSOR ↓] or [↑ CURSOR] key. "#" need not be keyed in. Up to four pairs of setting data including the keyed setting number are displayed. To the right of bit display, the decimal value indicating the sum of the data on that line is shown. The cursor is positioned to the designated setting number.

c. The setting number designation may be updated by [CURSOR ↓] or [↑ CURSOR] key, and the screen may be updated by [PAGE ↓] or [↑ PAGE] key.

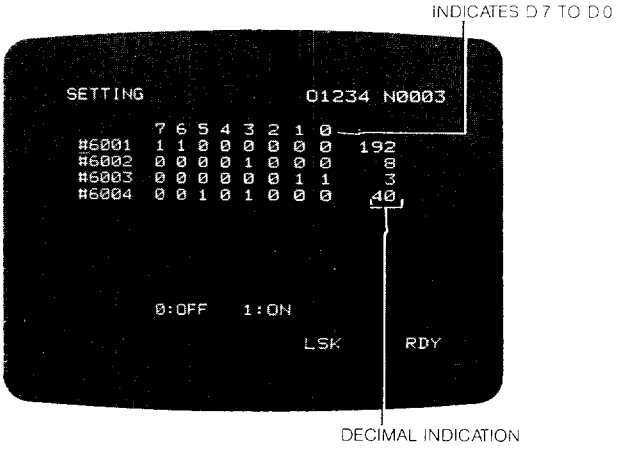


Fig. 4. 34 Display of Setting Data Shown in Bit Display

Note: When #6000 is designated, the screen dedicated to the display of on/off state of the "internal toggle switch" is provided. For details, see 4.3.6.2.

(2) Writing of setting data (bit display format)

The writing of the setting data of bit display format is performed by the following operations:

a. Depress the SET function key.

b. By the operation of PAGE key and CURSOR key or keying in 4 digits, designate the setting number to be written.

c. Depress the INSRT key. The cursor moves to the bit data from a setting number. Designate the data of D7.

d. Depress the [CURSOR →] key. Each time the key is depressed, the cursor moves by one bit toward D0. Locate the cursor at a desired bit position.

e. Depress the WR key. The designated bit data reverses (0 to 1 or 1 to 0). If you depress the WR key again, the bit data will reverse again. Normally, "1" designates on state and "0" off state,

f. To write data in decimal mode, locate the cursor at the right most column (decimal data).

EXAMPLE: Writing in decimal mode

Entered data	7	6	5	4	3	2	1	0	
0 WR ⟶	0	0	0	0	0	0	0	0	0
2 5 5 WR ⟶	1	1	1	1	1	1	1	1	255

g. Repeat steps b, through f. to write required data. If you keep the [CURSOR ↓] or [↑ CURSOR] key depressed, the cursor will move column by column in the screen automatically.

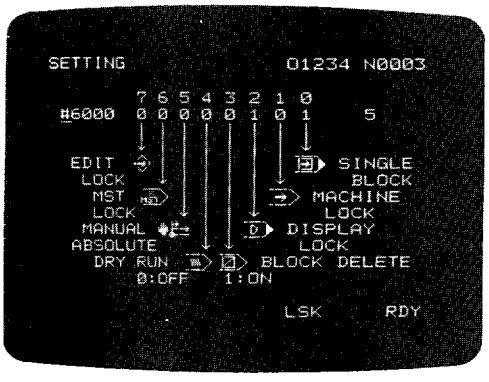
h. When data has been written, depress the INSRT key. Cursor returns to the position of setting number normally, this sequence of operations begins and ends both with the depression of the INSRT key.

4. 3. 6. 2 Internal Toggle Switches†

(1) When the eight basic function switches shown below are omitted from the machine operator's station, each function may easily be turned on/off by the setting operation from the NC operator panel.

(2) The setting numbers are #6000D7 through D0 of bit display format. By the operation of the writing of bit display format setting described above, turn on/off each function.

When "1" is set, the function is turned on. When "0" is set, the function is turned off.



Display shows that Single Block and Display Lock are on.

Fig. 4. 35 Display of Internal Toggle Switch Status

Notes :

1. The internal toggle switch is an optional function. Hence, which internal toggle switch is available depends on the controlled machine. The display for unused toggle switches is blank. For details, refer to the instruction manual of the machine in question.

2. When the machine control station is provided with the switches that turn on and off the above functions, the state of the switch on the machine control station is ORed with that of the operator's panel to determine the final ON/OFF state.

Setting Data	Switches on Machine Control Station	Resultant ON/OFF
"0" = OFF	OFF	OFF
"0" = OFF	ON	ON
"1" = ON	OFF	ON
"1" = ON	ON	ON

4. 3. 6. 3 Setting Data of Decimal Display Format

The following setting numbers have the setting data of decimal number display format:

#6160 through #6219

#6500 through #6579

#8600 through #8750

These setting numbers are used for the control constants of tool life control and multiple repetitive cycles.

(1) Display of setting data (decimal display format)

The setting data is displayed by the following operations:

a. Depress the SET function key.

b. Key in the 4-digit setting number and depress [CURSOR ↓] or [↑ CURSOR] key. The maximum of 10 lines of setting data including the keyed setting number are displayed. The cursor is positioned to the designated setting number.

c. The setting number designation may be updated by [CURSOR ↓] or [↑ CURSOR] key, and the screen may be updated by [PAGE ↓] or [↑ PAGE] key.

4. 3. 6. 3 Setting Data of Decimal Display Format (Cont'd)

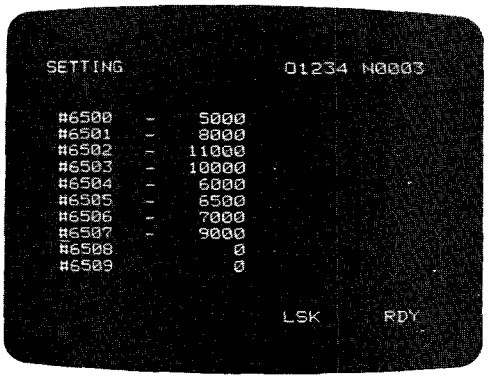


Fig. 4. 36 Display of Setting Data in Decimal Mode

(2) Writing of setting data (decimal display format)

The writing of the setting data of decimal display format is performed by the following operations:

a. Depress the SET function key.

b. By the operation of PAGE key and CURSOR key or keying in 4-digit value, designate the setting number to be written.

c. Key-in the value by the data key and depress the WR key. The keyed value is written as the data of the setting number designated by the cursor.

d. Repeat operations of b. and c. to write the necessary setting data.

< REMARKS >

· System No. setting (#6219)

Set system No. at "1" to write parameter number. System number can be set by setting the value of #6219 through the operator's panel.

Setting of #6219

"0": SYSTEM

For normal operation. Writing parameters is prevented.

"1": PARAMETER

To write parameters. At this position, Cycle start is prevented.

"4": TEST (0)

Normal operation is similar to the case of "0" SYSTEM. Self-diagnostics of the memory contents and checking of reference zero return point are omitted.

Notes :

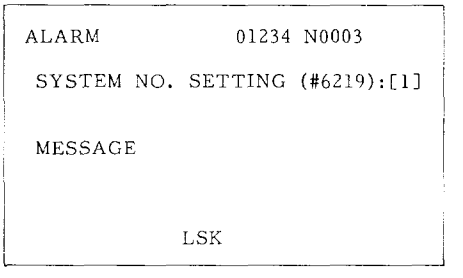
1. Setting values other than those described above will prevent correct operation.

2. Setting can be effective only when the system number switch provided on the upper part of the unit is set at "0."

Alarm Display of System No. Setting (#6219)

When setting #6219 is set at "1" or "4," Cycle Start is prohibited or self-diagnostics are over looked, causing unexpected errors.

To prevent this, alarm status is displayed if #6219 is set at a value other than "0."



System number #6219 should be set at "0" except for writing parameters.

· System number can also be set by the rotary switch on the main printed circuit board. For details, refer to Maintenance Manual.

4. 3. 7 DISPLAYING AND WRITING PARAMETERS

In this system, varying parameters are stored in the memory and they determine operating conditions such as tape code and feedrate. For details, see Appendix 2, "LIST OF PARAMETER NUMBERS." The parameters may be displayed at any time even during automatic operation.

The parameters are of the following two types:

(1) The parameters of bit display format.

(2) The parameters of decimal display format.

4. 3. 7. 1 Parameters of Bit Display Format

Parameter numbers #6005 through #6049 indicate the parameters of bit display format. Each number has 8-bit information of D7 through D0.

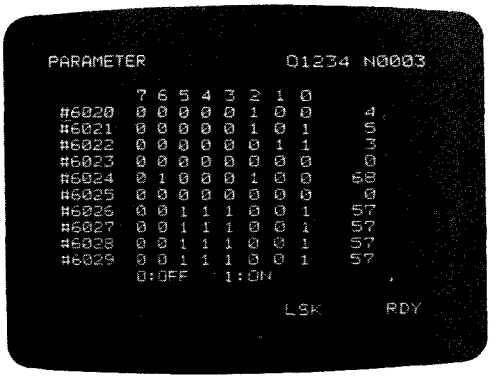


Fig. 4. 37 Display of Parameters in Binary Mode

4. 3. 7. 2 Parameters of Decimal Display Format

The following parameter numbers indicate the parameters of decimal display format:

#6050 through #6149

#6160 through #6349

#6500 through #6659

#8000 through #8225

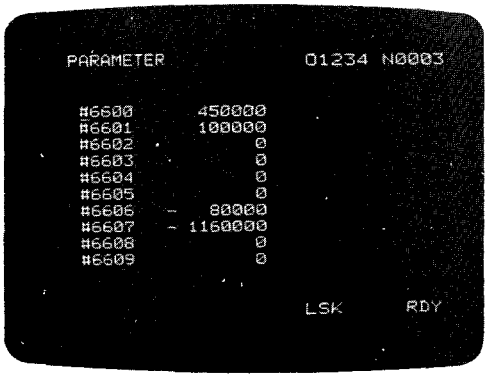


Fig. 4. 38 Display of Parameters in Decimal Mode

4. 3. 7. 3 Displaying and Writing Parameters

(1) Operations for parameter display

The operations for parameter display are the same as those for setting display except that the PRM function key is depressed instead of the SET function key. For details, see 4. 3. 6.

(2) Operations for parameter writing

a. The parameter values are preset according to the performance of the machine and applications. Therefore, you should consult the machine tool builder if you want to change parameter settings.

b. The parameters are protected with a system No. switch provided on the tape reader so that they should not be destroyed by wrong operation. Normally system No. 0 is selected and, at this time, the parameters cannot be rewritten by any operation.

c. The operations for parameter writing are the same as those for setting display except for the following:

(i) First, set the system No. switch to "1."

(ii) Depress the PRM function key instead of the SET function key. Then, the parameter data may be written by the same operations as those for the writing of setting data of bit display format (4. 3. 6. 1, (2)) or for the writing of setting data of decimal display format (4. 3. 6. 3, (2)).

(iii) After the completion of the writing operation, set the system No. switch to "0."

d. When parameters have been changed, be sure to reset the control by depressing the RESET key.

Some of the parameters cannot be effective until power is turned off and then turned on again. For setting, refer to APPENDIX 2, "LIST OF PARAMETERS."

4. 3. 8 DISPLAYING STATUS INPUT/OUTPUT SIGNALS

Depress the DGN function key, and the state of every input/output signal will be displayed on the CRT. This is possible at any time even during automatic operation.

For more detail of this operation, see "YASNAC LX3 Maintenance Manual."

The state of the input/output signal is also given in the hexadecimal notation at the rightmost column for the ease of maintenance work.

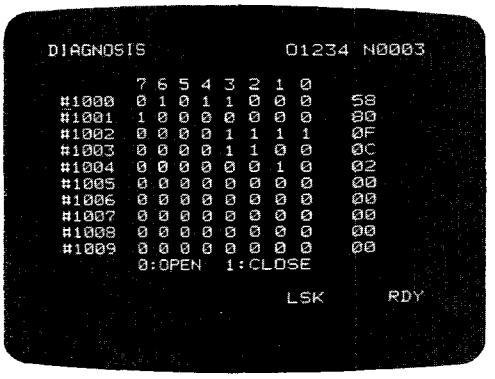


Fig. 4. 39

4. 3. 9 ALARM CODE (ALM) DISPLAY

Alarm codes and other data are displayed by the following operations:

(1) Depress the ALM function key. One of the following screens is displayed:

a. Display of the alarm number and the message. (ALARM)

b. Display of operating time. (TIMER)

c. Display of registered program number. (PROGRAM NO. TABLE)

d. Display of maintenance history. (MAINTENANCE)

(2) The above screens may be selected by depressing PAGE ↓ or ↑ PAGE key.

4. 3. 9. 1 Alarm Code Display

If an alarm status has happened, "ALM" or "A/B" (on battery alarm) blinks on the bottom line of the screen regardless of working mode and function. If this happens, the detailed information of the alarm status may be displayed by the following operation.

(1) Depress the ALM key.

Then up to four pairs of alarm code and message will be displayed, with more serious one on a higher line.

Note: The alarm screen will appear during an alarm state and, therefore, it is not needed to operate the PAGE ↓ key.

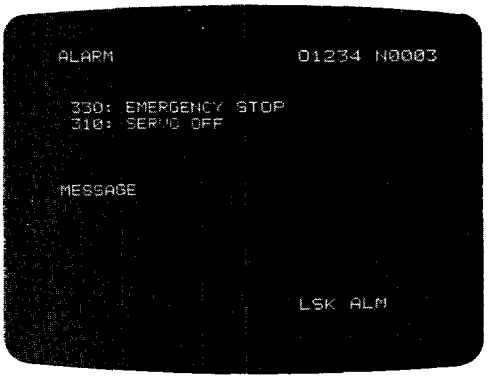


Fig. 4. 40 Display of Alarms

(2) To reset the alarm status and screen, remove the cause of alarm then depress the RESET key.

For the detail of alarm code, refer to Appendix 5 List of Alarm Codes.

4. 3. 9. 2 Message Display (ALARM)†

This feature can display messages on the CRT screen by the instruction of PC when the machine sequence control option (PC system) is built in the unit. Normally, this feature is used to display the cause of the alarm detected by the PC.

When the message display instruction (macro instruction) is executed in the PC, "ALM" or "A/B" flashes at the bottom of the CRT screen regardless of mode and function.

In this case, the message may be displayed by the following operation:

(1) Depress the ALM function key. The message is displayed at the bottom of the screen along with the sequence error code. The message to be displayed depends on the machine. For details, refer to the instruction manual of the machine in question.

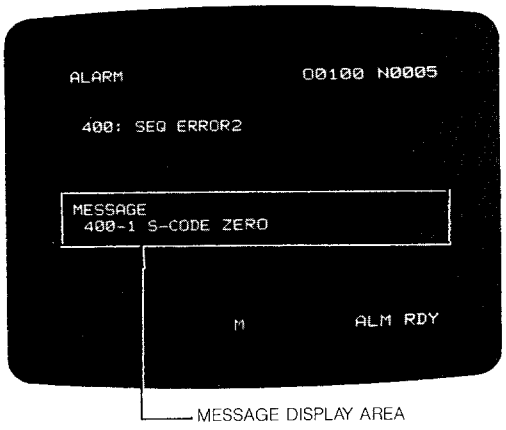


Fig. 4.41 Example of Message Display

(2) When the RESET key is depressed after removing the cause of the alarm which displayed the message, the message display and alarm state may be cleared.

Note: In some cases, only the message display is provided without displaying "400: SEQ ERROR."

4.3.9.3 Display and Deleting of Registered Program Number (PROGRAM NO. TABLE ☐☐)†

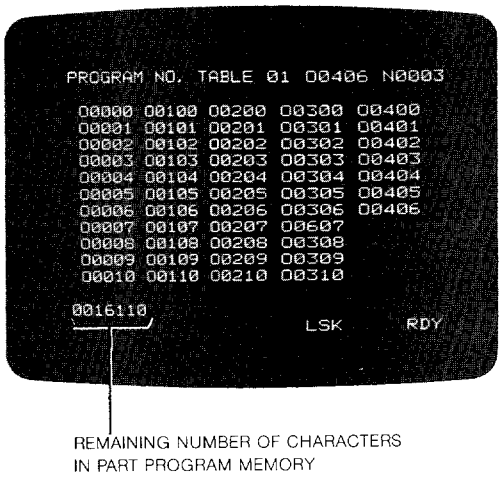
This screen displays all registered program numbers and the number of remaining characters in the part program memory.

(1) The number of program numbers that can be registered depends on options.

No.	Max Number of Programs	Type	Program No. Table
1	99	Basic	01 to 02
2	199	Option 1	01 to 04
3	999	Option 2	01 to 19

(2) All program numbers already registered are displayed. By depressing [PAGE ↓] or [↑ PAGE] key, the page shown below may be obtained.

(3) Depress the PROG key and then depress ERASE key to delete part program.



The remaining number of characters in part program memory is displayed in the lower left corner of the screen.

Fig. 4.42

Note: This screen displays only the registered program numbers. A program number is registered by depressing the PROG function key in EDIT mode.

The program directory screen used to be displayed in the alarm functions, but is now improved to be displayed in the program functions.

Therefore, the program directory screen is no longer displayed in the alarm function.

In the program function, the program directly screen and program screen are displayed alternately regardless of the mode, each time the " [PROG] " screen is depressed. When it is under the edit mode, program I/O verification, and deletion can be made on the program directly screen.

4. 3. 9. 4 Operation Time Display

The system counts the duration of automatic operation and it may be displayed. This function permits the display of the time it has taken for a single piece of work or the total operational time of the system.

(1) Three kinds of operation time will be displayed in hours, minutes, and seconds.

a. POWER ON: Total operating time after POWER ON

b. CYCLE START: Total operating time of CYCLE START

c. FEED: Total operating time of FEED

d. EXTERNAL INPUT†: Total operating time while external input signal is ON (optional)

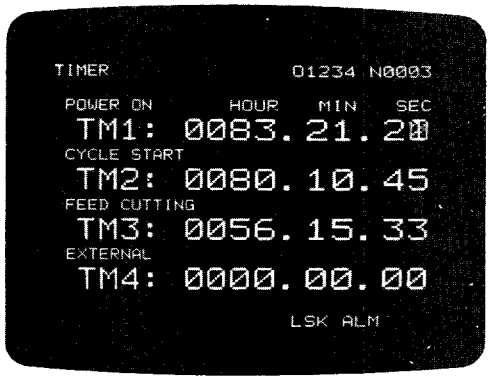


Fig. 4. 43 Operation Time Display

(2) Reset of operating time display

The above operating time displays may be reset separately by the following operations. In the state where the operating time is displayed;

a. Depress "1" then ORG keys.
The time of "POWER ON" is reset.

b. Depress "2" then ORG keys.
The time of "CYCLE START" is reset.

c. Depress "3" then ORG keys.
The time of "FEED CUTTING" is reset.

d. Depress "4" then ORG keys.
The time of "EXTERNAL INPUT" is reset.

Unless this display reset operation is performed, then operating time display is retained when the power is turned off.

e. When "1," "2," "3," "4," "ORG" keys are depressed, all the times shown above are reset at a time.

4. 3. 9. 5 Maintenance History Display (MAINTENANCE)

(1) This screen displays the maintenance information on the control unit. This display is independent of the control unit functions.

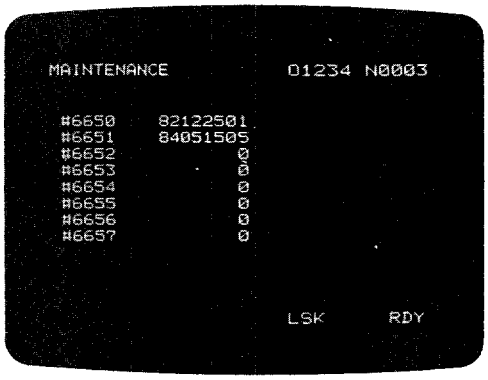


Fig. 4. 44

4. 4 LOADING PART PROGRAMS AND NC DATA INTO MEMORY (IN)

This paragraph describes the operations for storing the following data into the corresponding internal memory of the NC:

- Part program
- Tool offset amount
- Setting data and parameters

If these data are to be stored in the form of a punched tape, enter them through the tape reader or the data input/output interface (option). For the methods of setting the input/output equipment (setting #6003) and the baud rate (parameters #6026 and #6027), refer to "YASNAC-LX3 Maintenance Manual" Date Input/Output Interface. The following description is made assuming that this option is installed.

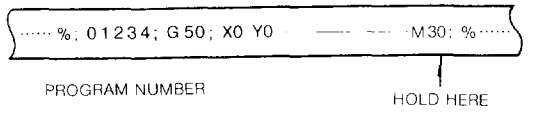
4.4.1 LOADING PART PROGRAM TAPE INTO MEMORY

(1) Loading a part program which has a program number.

a. Select EDIT mode.

b. Depress the PROG key.

c. Load the NC tape to the tape reader or an equivalent external device.



d. Depress the RESET key.

e. Depress the IN key.

Then the system starts to read the tape and enlists the program number punched on the tape as the first record. The system checks for duplicator of program number as in 1. Operation ends with error if the designated program number is not found on the tape.

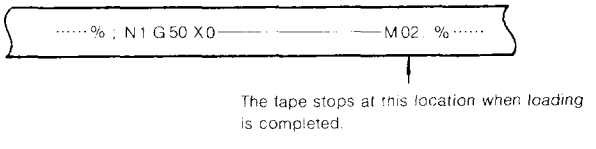
When the tape reader has read "M02 ;," "M30;," or "M99;," it stops and "IN" disappears from the CRT. Now the part program has been stored in memory.

(2) Loading a part program which has no program number.

a. Select EDIT mode.

b. Depress the PROG keys.

c. Load the NC tape to the tape reader or an equivalent external device.



d. Depress the RESET key.

e. Depress the address O key then enter the program number.

f. Depress the IN key.
The system starts to read the tape. If the keyed-in program number coincides with the registered program number, "ALREADY IN" blinks on the CRT screen. If this happens, delete the program number, then repeat steps a. through f. while the tape is being read, "IN" blinks on the CRT.

g. When the tape reader has read "M02 ;," "M30 ;," or "M99 ;," it stops and "IN" disappears from the CRT. Now the part program has been stored in memory.

Notes :

1. Program number "O0000" is always in the registered state, so it cannot be erased. This program number should not be used in general.

2. The tape which has no program number may be stored as described before. However, write a program number to the head of the tape, in principle. The operation of "Oxxxx IN" described before causes only program number registration. It does not cause the storing of information of "O0000" into the part program memory. Only the program number on tape is stored into the memory. Assume that a tape having no program number is stored and then all part programs are punched out by depressing "0," "-," "9," "9," "9," "9," and "OUT" keys. Since this tape contains programs with no program number, the correct restoring of all part programs may not be performed by depressing "0," "-," "9," "9," "9," "9," and "IN" keys.

3. Consequently, when a tape having no program number has been stored, write the program number to the head of part program by the EDIT operation.

4. The stop position of tape can be changed by parameter #6021D_0, with parameter setting "1", program completes by M02, M30, or M99. However, with parameter setting "0", program does not stop till %.

EXAMPLE:

N1 G50 X0 Z0 ;

When this is in the first block, position the cursor to N and key in as follows (in EDIT and PROG modes):

Oxxxx ; N1 ALTER

(3) Storing a program with program numbers changed I

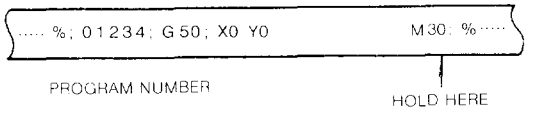
To register a program with a program number different from the one punched on tape, perform the following operations:

a. Select the EDIT mode.

b. Depress the PROG key.

4.4.1 LOADING PART PROGRAM TAPE INTO MEMORY (Cont'd)

c. Set the NC tape to the tape reader or the external equivalent equipment.



d. Depress the RESET key.

e. Key in "O" and PROGRAM NUMBER.

f. Depress the IN key.

The program number entered from the key is registered in preference to the program number punched on the tape. At this time, the program number on the tape is written to the part program memory simply as a label.
M02 ;, M30 ; or M99 ; is read and the storing operation is completed.

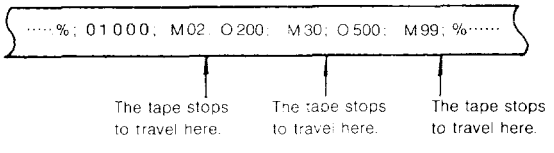
Note: If a program is stored with a changed program number as described above, the program number punched on the tape is stored in the part program memory without change. Consequently, to avoid the confusion in the later handling, replace the program number in the part program memory with changed program number by the EDIT operation.

(4) Loading part programs from a tape

a. Select EDIT mode.

b. Depress the PROG key.

c. Load the NC tape to the tape reader or an equivalent external device.



d. Depress the RESET key.

e. Depress the IN key.

Then the system starts to read the tape and enlists the program number punched on the tape as the first record. The system checks for duplication of program number as described in (2).

The tape reader stops each time it has read "M02 ;," "M30 ;," or "M99 ;."

f. Depress the IN key again.

The tape reader resumes to read the tape. Repeat this operation until all programs are loaded.

(5) Storing a program with program numbers changed II

When "O" key is depressed and program number is keyed in before depressing IN key as described in (4) above, the keyed in program number is registered in preference to the program number punched on the NC tape

(6) Loading programs continuously

Programs existing on a tape as shown in (4) may be loaded continuously without interruption. For this purpose, depress "O," "-," "9," "9," "9," and "9" before the first depression of the IN key. The tape reader stops at the position of "%."

4.4.2 MAKING ADDITION TO A PART PROGRAM

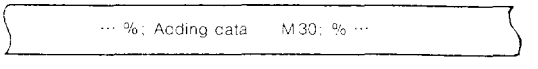
Perform the following operation to add data to a part program which is already loaded.

a. Select EDIT mode.

b. Depress the PROG key

c. Depress the O key then enter the part program number and depress the CURSOR key.
The system searches the designated program

d. Load the tape of adding data to the tape reader.



e. Depress the RESET key.

f. Depress the NEXT and IN keys in this order.

The data will be read from the tape into the memory.

Notes:

1. You cannot add data to the middle of a program. If necessary, delete the last part of the program by editing operation and add the data.

2. Continuous storing operation of NEXT and IN needs reset key operation every time.

4.4.3 LOADING PART PROGRAMS BY MDI

Part programs may be loaded not through the tape reader but by MDI operation. Perform the following.

a. Select EDIT mode.

b. Depress the PROG key

c. Depress the RESET key.

d. Depress the O key then enter the part program number and depress the WR key.

   The designated program number will be registered. If this number already exists, "ALREADY IN" blinks and, in this case, it is required to delete the registered program number.

e. Write the part program by operating the address key and the data key. As shown in the figure below, the keyed in data is displayed on the bottom line from left to right sequentially. The maximum number of characters that can be written at a time is 32. Within this limit, data may be keyed in over multiple words or blocks. However, when the 10th character is keyed in, the normal display shown to the right of the line is blanked.

f. Depress the INSRT key.

   The keyed in data is stored in the part program memory.

g. Repeat the operations of e. and f. above to write the part program. The program edit operation is enabled by the use of ERASE, INSRT, and ALTER keys during this program storing operation.

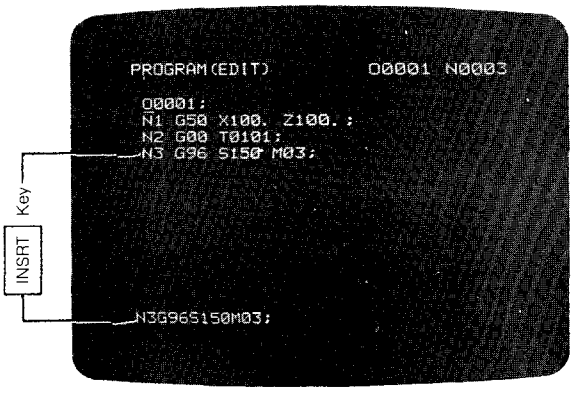


Fig. 4.45

h. Key in M02 ;, M30 ;, or M99 ; and depress INSRT key. This completes the storing of the part program.

4.4.4 INPUTTING TOOL OFFSET DATA INTO MEMORY

Normally, the tool offset data is written to the tool offset memory by MDI operation. This data may also be entered in the form of a punched tape.

(1) The format of the tool offset data tape is as shown below:

```
LABEL % ;
T01 X....Z....R.... ;
T02 X....Z....R.... ;
T03 X....Z....R.... ;
 T99 X....Z....;
 %
```

(2) The storing of the data by the above tape is performed by the following operations:

a. Select EDIT mode.

b. Depress OFS key.

c. Set the tool offset data tape onto the tape reader.

d. Depress the RESET key.

e. Depress the IN key.

   The tape reader starts to read the tape. "IN" blinks on the CRT while the data are read.

f. The tape reader stops when it has read "%" (or "ER"). "IN" disappears from the CRT. Now the tool offset data has been read into memory.

(3) Remarks

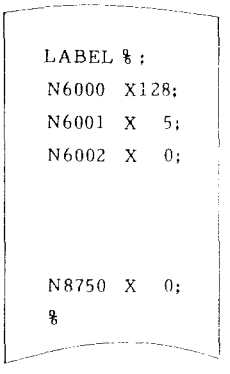
G10:

In the case of the tool offset data tape of the format by tool offset data designation, performing the cycle start in TAPE mode causes to store the data into the tool offset memory.

4.4.5 INPUTTING SETTING DATA AND PARAMETER DATA

Through setting data and parameter data are input in their memories by MDI operation normally, they may also be entered by means of paper tape. Setting data and parameter data may be input from a single tape.

(1) The tape format is as follows.



(2) The input operation is as follows.

a. Set the SYSTEM No. switch at 1.

b. Select EDIT mode.

c. Depress the PRM key.

d. Depress the RESET key.

e. Depress IN key.

The tape reader starts to read the tape. "IN" blinks on the CRT while the data are being read.

The tape reader stops when it has read "%" (or "ER"). "IN" disappears from the CRT. Now the setting/parameter data have been read into memory.

f. Return the system No. switch to 0.

g. Turn the power off then on.

The NC operation by the newly stored setting data and parameter may be performed.

Note: At the end of operation e., the operations on the NC operator's panel are all invalidated. Hence, the power must be turned on again.

4.5 TAPE VERIFYING

The punched tape of the data shown below may be compared to the contents of the NC internal memory to check if they match.

- Part program
- Tool offset data
- Setting data and parameter

The punched tape is entered through the NC tape reader or the data input/output interface (option). For the methods of setting the input/output equipment (setting #6003) and the baud rate (parameters #6026 through #6029), refer to "YASNAC-LX3 Maintenance Manual." The following description is made assuming that this option is installed.

4.5.1 VERIFYING PART PROGRAM TAPE

(1) Verifying a part program tape having program number

a. Select the EDIT mode.

b. Depress the PROG function key.

c. Set the part program tape to the tape reader.

d. Depress the RESET key.

e. Depress the VER key.

The tape is started to compare the contents of the part program memory to the contents of the part program tape. During this operation, "VER" keeps flashing. If a mismatch is found, "INPUT ERROR" is displayed flashing.
When a match is found and this operation is completed, the tape reader stops, upon which "VER" display is erased.

Note: By the operation of "RESET, VER," the verifying feature verifies the data from the tape head to % code.

(2) Verifying a part program tape having no program number

a. Select the EDIT mode.

b. Depress the PROG function key.

c. Set the part program to the tape reader.

d. Depress the RESET key.

e. Depress the "O" key and key in program number.

f. Depress VER key.

The tape starts to compare the contents of the part program memory to the contents of the part program tape. During this operation, "VER" keeps flashing. If a mismatch is found, "INPUT EROR" is displayed flashing. When a match is found and this operation is completed, the tape reader stops, upon which "VER" display is erased. If the keyed-in program number is not found in the memory, "NOT FOUND!" is displayed flashing. In this case, depress the CAN key and start with the operation of d.

Notes :

1. The operations for verification with a program number different from the program number punched on the tape are the same as those of (2) above. The keyed-in program number is processed in preference to the punched program number.
2. Verification by the operation of "Oxxxx VER" regards the punched information as the information on the keyed-in program number. Hence, when verifying a tape containing program numbers, no program number should be keyed in.

(3) Verifying a tape containing multiple part programs

Multiple part programs punched in a single tape are continuously verified by the following operations:

a. Select the EDIT mode.

b. Depress the PROG function key.

c. Set the part program tape to the tape reader.

d. Depress the RESET key.

e. Depress the VER key.

When M02, M30, or M99 is read, the tape reader does not stop but all the part programs are continuously verified up to % code. When the verification is completed, the tape reader stops at the position of % code.

4.5.2 VERIFYING TOOL OFFSET VALUE TAPE

The contents of the tool offset value tape are compared to the contents of the offset memory by the following operations:

(1) Select the EDIT mode.

(2) Depress the OFS function key.

(3) Set the tool offset value tape to the tape reader.

(4) Depress the RESET key.

(5) Depress the VER key.

The tape starts to compare the contents of the tape to the contents of the tool offset memory. During this operation, "VER" keeps flashing. If a mismatch is found, "INPUT ERROR" is displayed flashing. When a match is found and this operation is completed, the tape reader stops, upon which "VER" display is erased.

4.5.3 VERIFYING SETTING AND PARAMETER TAPES

The setting data tape or the parameter tape is compared to the respective contents of the memory. It is possible to punch the setting data and parameter on a single tape and store them at a time by the following operations:

(1) Select the EDIT mode.

(2) Depress the PRM function key.

For the tape punched only with setting data, depressing the SET key causes the same effect.

(3) Set the setting data and/or parameter tape to the tape reader.

(4) Depress the RESET key.

(5) Depress the VER key.

The tape starts to compare the contents of the tape of the contents of setting or parameter. During this operation, "VER" keeps flashing. If a mismatch is found, "INPUT ERROR" is displayed flashing. When a match is found and this operations is completed, the tape reader stops, upon which "VER" display is erased.

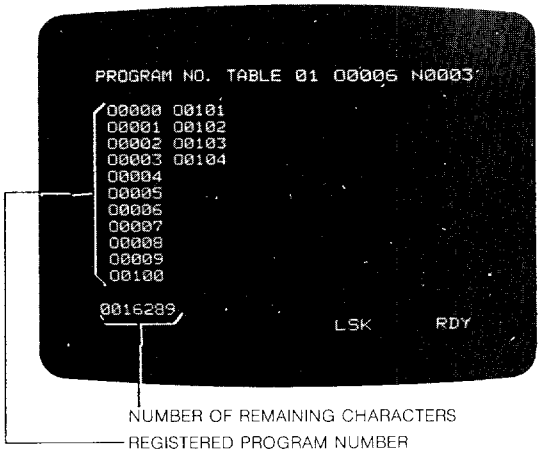
4. 6 EDIT

4. 6. 1 CHECKING REGISTERED PART PROGRAM NUMBER

Before editing part programs, make confirmation of the registered program numbers and the remaining number of characters in the part program memory by the following operations:

(1) Depress the PROG function key.

(2) Depress the CURSOR ↑ or CURSOR ↓ key to select the screen (PROGRAM NO. TABLE□□) of registered program number display.



This screen only displays information but also deletes program No. and therefore cannot be used to register program numbers.

Fig. 4. 46

4. 6. 2 DISPLAYING AND CHECKING STORED PART PROGRAMS

Stored part programs may be displayed on the CRT screen to check their contents by the following operations:

(1) Select the EDIT mode.

(2) Depress the PROG function key.

(3) Depress the RESET key.

(4) Key-in O PROGRAM NUMBER.

(5) Depress the CURSOR ↓ key.

The designated program number is searched. One screen of data (for 10 lines) from the head of the searched program is displayed on the CRT. If the program number has not been found, "NOT FOUND" is displayed flashing. This display may be reset by depressing CAN key in general.

(6) The preceding or following screen may be displayed on the CRT by depressing the ↑ PAGE or PAGE ↓ key.

(7) When the CURSOR ↑ or CURSOR ↓ key is depressed, the cursor is moved to the preceding or following word on a word basis.

The above operations make the word (address and data) designated by the cursor ready for such edit operations as modification, insertion, and deletion.

Note : The search for a program number may be performed also in the MEM mode. But the cursor movement by the PAGE ↓ or ↑ PAGE key is disabled.

4. 6. 3 MODIFYING PART PROGRAM BLOCKS

Modification of part programs is all performed in the EDIT mode and the PROG function by the operations which follow the operations described in the preceding paragraph 4.6.2 Displaying And Checking Stored Part Programs.

(1) Depress the page key and the cursor key to designate the word to be modified.

(2) Depress the address key and the data key to enter the word to be modified. As shown in the figure below, the keyed-in data is displayed in the bottom line on the CRT screen from left to right sequentially. The maximum number of characters that may be written at a time is 32. Within this limit, data may be keyed-in over multiple words or blocks.

(3) Depress the ALTER key.

The word designated by the cursor is deleted and the newly keyed-in data is displayed in that place. After modification, the altered word is in the designated state.

(4) Repeat the operations of (1) through (3) to modify any number of words.

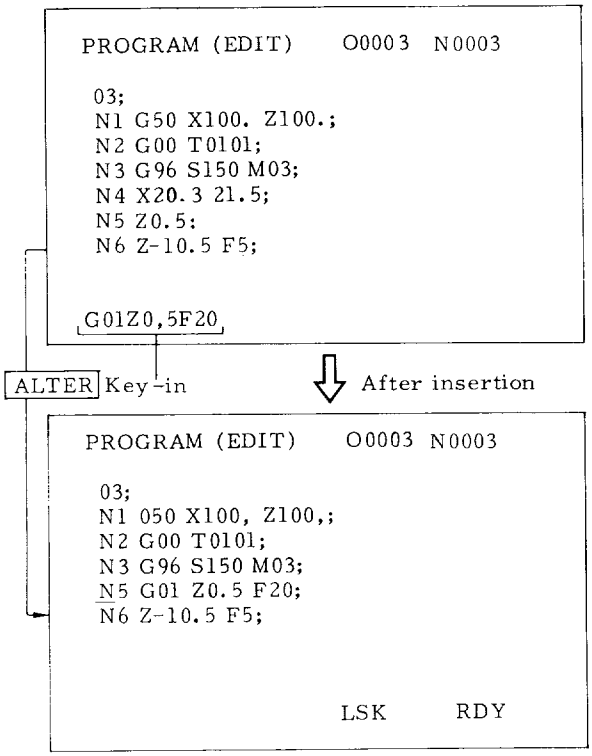


Fig. 4.47

4.6.4 ADDING PART PROGRAMS

A part program is added in the EDIT mode and the PROG function by the operations shown below which follow the operations of 4.6.2 Displaying And Checking Stored Part Programs.

(1) Depress the page key and the cursor key to designate the word immediately before the portion to be added.

(2) Depress the address key and data key to key in the word to be added. Within the limit of 32 characters, the data may extend over multiple words or blocks.

(3) Depress the INSRT key.

The keyed-in data is added to the portion immediately after the word designated by the cursor. After the addition, the added word is in the designated state.

(4) Repeat the operations of (1) through (3) to add any number of words.

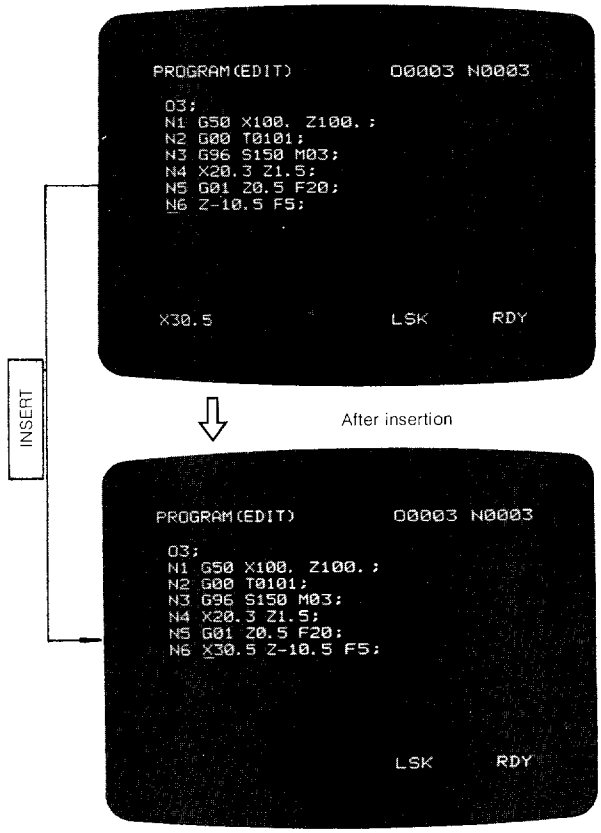


Fig. 4.48

4.6.5 DELETING PART PROGRAM BLOCKS

Part program blocks are deleted in the EDIT mode and the PROG function.

(1) Deleting words

Perform the operations below which follow the operations of 4.6.2 Displaying And Checking Stored Part Programs:

a. Depress the page key and the cursor key to designate the word to be deleted.

b. Depress the ERASE key.

The word designated by the cursor is deleted. After the deletion, the word subsequent to the deleted word is in the designated state.

c. Repeat the operations of a. and b. to delete any number of words.

(2) Deleting blocks

Data of one block may be deleted by the following operations:

a. Depress the page key and the cursor key to position the cursor to the head of the block to be deleted.

b. Depress EOB then ERASE.
All data of the block designated by the cursor are deleted.

Note: If the cursor is positioned to the middle of the block to be deleted and EOB key and ERASE key are depressed, the data from the cursor position to ";" code are deleted. Since ";" is also deleted, the words left undeleted are included in the next block.

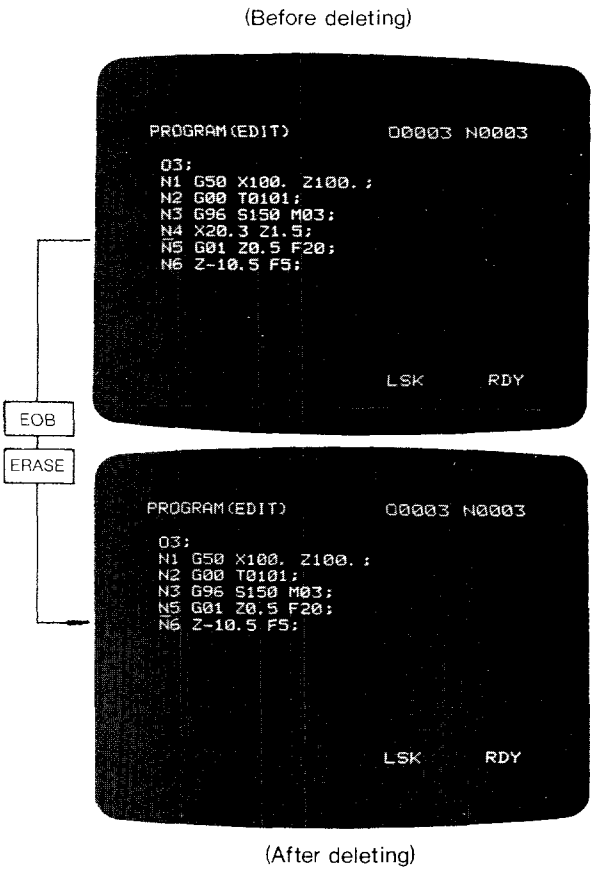


Fig. 4.49

(3) The data from the word where the cursor is located, to the block pointed by the sequence number specified by N**** can be erased as follows.

a. Enter address N followed by the destination sequence number, in the program screen.

(N #1 2 3 4)

b. Then, depress the ERASE key, and the block to N**** are erased. The cursor will be displayed at the head of the block next to the erased block.

When N**** is at the end of the program, the cursor will be displayed at the last address of the block immediately before the erased block.

If N**** cannot be found, a "NOT FOUND" alarm will appear on the original screen.

(4) Deleting program number

A program number is deleted by the following operations:

a. Select the EDIT mode.

b. Display the registered program No. of PROG function or ALM function.

c. Depress the PROG function key.

d. Depress "O" key and key-in the program number.

e. Depress the ERASE key.

The keyed-in program number and the corresponding part program are deleted.

(5) Deleting all program numbers

All program numbers are deleted by the following operations:

a. Select the EDIT mode.

b. Depress the PROG function key or ALM function key.

c. Key in 0, - 9 9 9 9.

d. Depress the ERASE key.

All registered program numbers and part programs are deleted. However, only program number "0" is newly registered with EOB not deleted.

4.7 PART PROGRAM AND NC DATA OUTPUT OPERATIONS

The following data, which are stored in the NC internal memory, may be sent to the external equipment through the data input/output iterface (option):

- Part program
- Tool offset value
- Setting data and parameter

When the external equipment has the tape punch feature, the data may be punched out; when it has the print feature, the data may be printed out. For the methods of setting the input/output equipment (setting #6003) and the baud rate (parameters #6026 through #6029) through this data input/output interface, refer to "YASNAC-LX3 Maintenance Manual."

4.7.1 OUTPUTTING PART PROGRAM TO PAPER TAPE

(1) The part program of the designated program number is punched out by the following operations:

a. Connect the external equipment such as the tape puncher to the NC via the data input/output interface.

b. Make the external equipment relay for operation.

c. Power on the NC.

d. Select the EDIT mode.

e. Depress the PROG function key.

f. Check to see if the external equipment is ready.

g. Depress the RESET key.

h. Depress "O" key and key in program number.

i. Depress the OUT key.

The part program of the keyed-in program number is output to the external equipment. If it is a tape puncher, tape punch is performed. When the output of the part program is completed, the tape puncher stops automatically. During the output of data, "OUT" is flashing.

j. To discontinue the punch out operation, depress the RESET key. However, the discontinued operation cannot be resumed. Go back to f. and repeat the operations all over again.

Note: When RESET, OUT are operated without keying in the program number, the part program of the currently displayed program number is outputted.

(2) Punch out of all part programs

All registered part programs may be output to the external equipment by the above operations except that 0 - 9 9 9 9 must be keyed in the operation of h.
All part programs stored in the memory are output (punched out, etc.) consecutively.

4.7.1 OUTPUTTING A PART PROGRAM TO PAPER TAPE (Cont'd)

Note: The contents of program number O0000 are outputted only when #6231D3 = 1.

4.7.2 OUTPUTTING TOOL OFFSETS TO PAPER TAPE

Tool offsets may be output to paper tape by the following operations:

(1) Connect the external equipment such as the tape puncher to the NC via the data input/output interface.

(2) Make the external equipment ready for operation.

(3) Power on the NC.

(4) Select the EDIT mode.

(5) Depress the OFS function key.

(6) Check to see if the external equipment is ready.

(7) Depress the RESET key.

(8) Depress the OUT key.

All contents of the offset memory such as tool offsets and tool coordinate data are outputted to the external equipment. If the external equipment is a tape puncher, tape punch out is performed. When the output of the data is completed, the external equipment stops automatically.

(9) To discontinue the punch out operation, depress the RESET key. However, the discontinued operation cannot be resued. Go back to (6) and repeat the operations all over again. The format of the punched out tape is the same as that of the tape described in 4.4.4, (1).

4.7.3 OUTPUTTING SETTING/PARAMETER DATA TO PAPER TAPE

Setting/parameter data may be punched out by the following operations.

(1) Connect the external equipment such as the tape puncher to the NC via the data input/output interface.

(2) Make the external equipment ready for operation.

(3) Power on the NC.

(4) Select the EDIT mode.

(5) Depress the PRM function key if the output of both setting data and parameter is desired. If the output of only setting data is desired, depress the SET function key.

(6) Check to see if the external equipment is ready.

(7) Depress the RESET key.

(8) Depress the OUT key.

All setting data and parameters are output to the external equipment. If it is a tape puncher, tape punch out operation is performed. When the output of the data is completed, the external equipment stops automatically.

(9) To discontinue the punch out operation, depress the RESET key. However, the discontinued operation cannot be resumed. Go back to (6) and repeat the operations all over again. The format of the punched out tape is the same as that of the tape described in 4.4.5, (1).

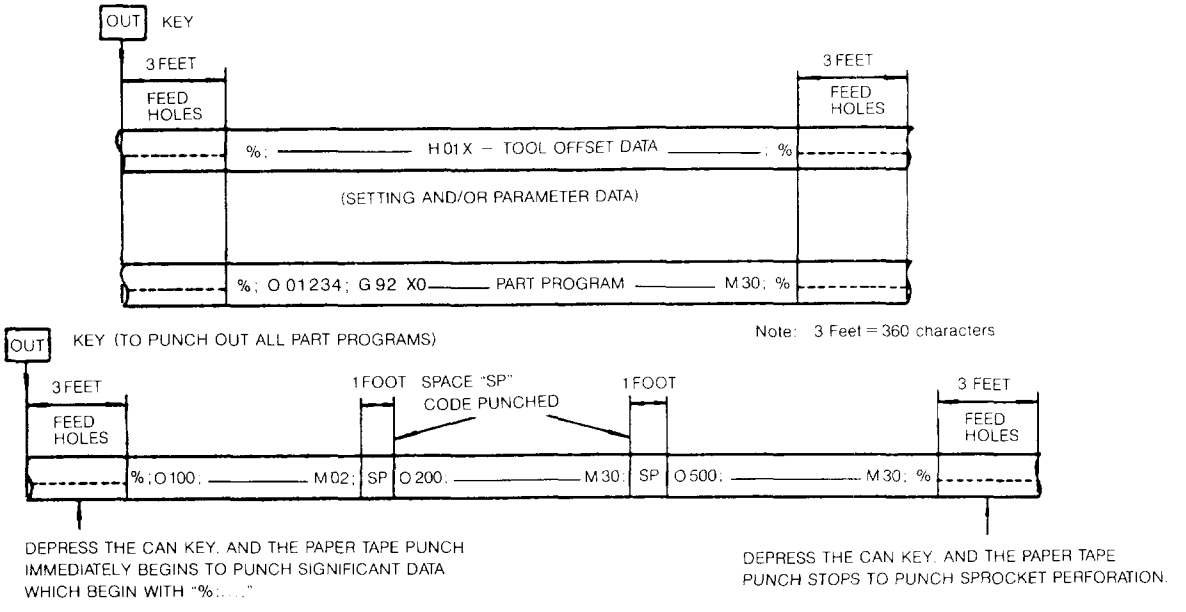


Fig. 4.50 Data and Program Formats on Paper Tape

4. 8 SUMMARY OF STORING AND EDITING OPERATIONS

	Operation			Edit Lock	System No. Switch	Mode	Function	Procedure
Parameter	Storing from NC operator's panel keyboard				1		PRM	Parameter number → CURSOR Data → WR
	Storing from tape (*1) (*2)				1	EDIT		RESET → IN
	Punch out (*3)							RESET → OUT
	Matching with tape (*1)							RESET → VER
Setting	Storing from NC operator's panel keyboard						SET	Setting number → CURSOR Data → WR
	Storing from tape				1	EDIT		RESET → IN
	Punch out							RESET → OUT
	Matching with tape							RESET → VER
Offset	Storing from NC operator's panel keyboard						OFS	Offset number → CURSOR Data → WR
	Storing from tape					EDIT		RESET → IN
	Punch out							RESET → OUT
	Matching with tape							RESET → VER
	Clear of all offsets							0 → −9999 → ORG
Part program	Storing from NC operator's panel keyboard			OFF		EDIT	PROG	0 → Program number → WR Repeat of edit opeation "addition of address data"
	Storing from tape	One part program	Tape with number 0	OFF				RESET → IN
			Tape without number 0 (*4)	OFF				RESET → 0 → Program number → IN
		All part programs on tape		OFF				RESET → 0 → −9999 → IN
		Addition to registered part program		OFF				RESET → NEXT → IN
	Punch Out	Designated part program						RESET → 0 → Program number → OUT
		All part programs						RESET → 0 → −9999 → OUT
	Matching with tape	One part program	Tape with number 0					RESET → VER
			Tape without number 0 (*4)					RESET → 0 → Program number → VER
		All part programs on tape						RESET → VER
	Edit	Modify of address data (*5)		OFF				CURSOR (Set to address data to be modified) → Address data → ALTER
		Add of address data (*5)		OFF				CURSOR (Set to address data just before addition) → Address data → INSRT
		Delete of one address data		OFF				CURSOR (Set to address data to be deleted) → ERASE
		Delete of one block (*5)		OFF				CURSOR (Set to address data at head of block to be deleted) → EOB → ERASE

	Operation		Edit Lock	System No. Switch	Mode	Function	Procedure
Part program	Address search				MEM EDIT		Data of the searched address → CURSOR
	Cleal	Specified part program	OFF		EDIT	PROG or ALM	0 → program number to be cleared → ERASE
		All part programs	OFF				0 → −9999 → ERASE
	Address search				MEM EDIT	PROG ALM	Address data to be searched → CURSOR
	Cleal	Designated part program	OFF		EDIT		0 → program number to be searched → ERASE
		All part programs on tape	OFF				0 → −9999 → ERASE

Notes:

*1 If the tape contains setting information, it is also stored and matched at the same time.

*2 When data has been stored from a parameter tape, turn the power on and off.

*3 Setting is punched out at the same time.

*4 Storing of a part program having a program number different from program number 0 on tape is performed by the same operation as for "tape without program number 0."

*5 Within the limit of 32 characters, addition of multiple address data and the change to one address data are permitted.

*6 When the cursor to the address data in the middle of a block and EOB and ERASE keys are depressed, the data following the cursor position is deleted.

5. MACHINE CONTROL STATION

5.1 SWITCHING UNITS ON THE CONTROL STATION

Fig. 5.1 shows a typical layout of switching units on the machine control station. For details, refer to the machine tool builder's manual.

5.1.1 MODE SELECT SWITCH

This switch selects operation mode of the NC system and consists of 6 positions (JOG, HANDLE, TAPE, MDI, MEM, EDT).

(1) JOG: To feed the tool continuously by manual opearation. Feedrate is set by FEEDRATE OVERRIDE switch.

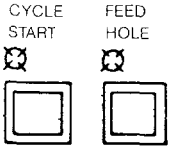
(2) HANDLE/STEP: To feed the tool by operating the manual pulse generator†. Where the control is not provided with a manual pulse generator, the tool is fed by step manually operating the JOG PUSHBUTTON(S).

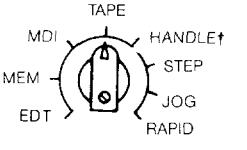
(3) TAPE: To automatically control the NC system with NC tape.

(4) MDI: To enter the block of data through the DATA keyboard and control the system automatically with the data.

(5) MEM: To automatically control the system with the stored part program.

(6) EDT: To store the part program into memory and edit the part program.





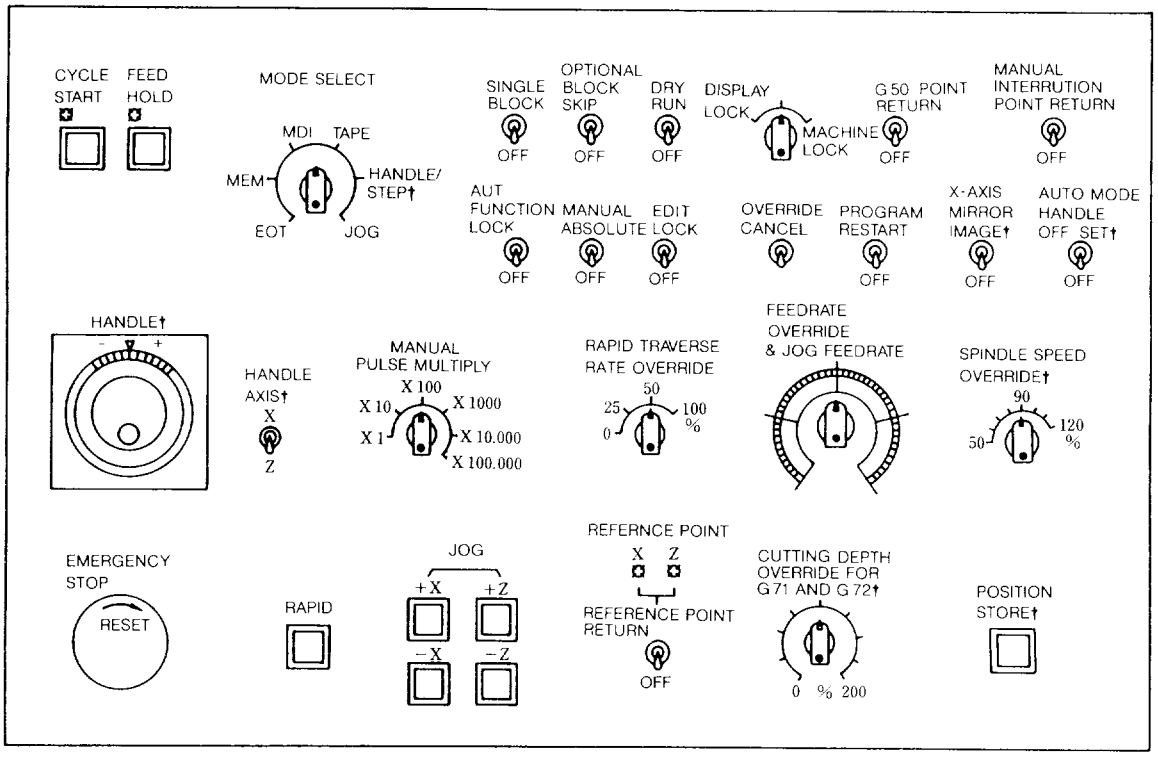


Fig. 5.1

5.1.2 CYCLE START PUSHBUTTON AND LAMP

Depress this pushbutton to start the system in the automatic operation mode (TAPE, MDI and MEM). The CYCLE START indicating lamp lights when automatic operation starts. Depress it again to start the operation after temporary stop by operating FEED HOLD pushbutton or MODE SELECT switch.

5.1.3 FEED HOLD PUSHBUTTON AND LAMP

Depress this pushbutton to temporarily stop automatic operation. The CYCLE START lamp goes off and the FEED HOLD lamp remains illuminated during temporary stop.

When the FEED HOLD pushbutton is depressed during feed operation, the feedrate is decreased immediately and the motion is stopped. Feedhold is not active during threadcutting by G32, G92, or G76 or dwell by G04. Rapid thread pull-up and retracting motion at G92 and G76 commands cannot be interrupted.
However, when the NC is provided with a "THREADING FEED HOLD" (option), the tool feed can be stopped temporarily even during thread cutting by the G92 or G76 command.

If it is depressed while M-, S-, or T-function without move command is being executed, the FEED HOLD lamp will light, but these functions will be executed continuously. On completion of the function, the lamp goes off and machine operation is stopped.

Depress the CYCLE START pushbutton to restart the operation after temporary stop by operating FEED HOLD pushbutton.

5.1.4 EMERGENCY STOP PUSHBUTTON

Depress this pushbutton to emergency-stop the machine. The servo power is turned off and the machine is stopped immediatly by dynamic brake. The RDY indication will disappear NC ALARM lamp lights and alarm code "330" is displayed.

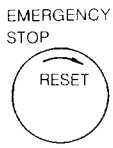
To restart the system after emergency stop, take the following procedure.

(1) Turn the EMERGENCY STOP pushbutton clockwise to release the locking.

(2) Depress the RESET key. Alarm code "310" replaces "330."

(3) Turn on the servo power again by depressing POWER ON pushbutton. NC ALARM LAMP is extinguished and READY lamp lights up.

The operation is effective in the reverse order of steps (2) and (3). Use this switch also for turning off the system.



5.1.5 HANDLE DIAL† (MANUAL PULSE GENERATOR)

The dial used as a manual pulse generator to feed the tool manually with the MODE SELECT switch set to the HANDLE. HANDLE operation is effective for an axis. Procedure of HANDLE operation is as follows:

(1) Set the MODE SELECT switch to the HANDLE.

(2) Select the axis to be operated with HANDLE AXIS select switch.

(3) Set the move amount per graduation of the dial by setting MANUAL PULSE MULTIPLY switch. (See Table 6.1.)

(4) Rotate the dial to move the selected axis.

Turning it clockwise causes the axis to move in the plus direction. The axis moves in the minus direction by turning it counterclockwise.

5.1.6 HANDLE AXIS SELECT SWITCH †

This switch is used to select an axis to be operated. For operation of the switch, see 5.1.5, "Handle Dial†."

5.1.7 MANUAL PULSE MULTIPLY SELECT SWITCH †

This switch is used to:

(1) Select the value from Table 6.1.7.1 corresponding to a single graduation of the HANDLE dial in the HANDLE mode.

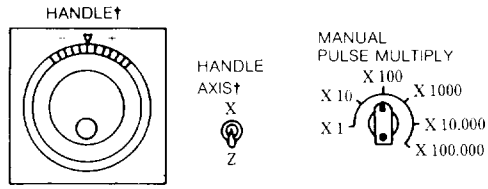
(2) Select the move amount (1 step) from Table 6.2 corresponding to each depression of JOG pushbutton in the STEP mode.

5.1.8 HANDLE DIALS FOR SIMULTANEOUS CONTROL OF UP TO TWO AXES †

When a manual pulse generator is connected for each axis, the tool can be manually moved along selected two axes simultaneously.

(1) The tool move distance per graduation of the HANDLE dial for the manual pulse generator is determined by the MANUAL PULSE MULTIPLY switch (Table 6.1). This switch is effective on all the three axes.

(2) Set the mode select switch to HANDLE, and turn the HANDLE dials for the desired axes in the positive or negative direction.



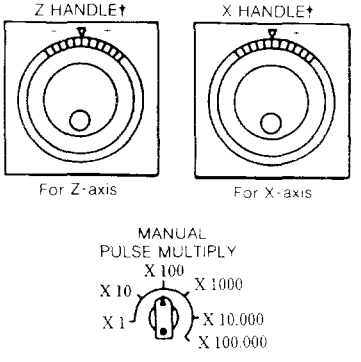
5.1.8 HANDLE DIALS FOR SIMULTANEOUS CONTROL OF UP TO TWO AXES † (Cont'd)

Table 5.1 Selection of Move Amount in the HANDLE Mode†

	Metric	Inch
× 1	0.001 mm/graduation	0.0001 in./graduation
× 10	0.01 mm/graduation	0.001 in./graduation
× 100 × 1000 × 10000 × 100000	0.1 mm/graduation	0.01 in./graduation

Table 5.2 Selection of Move Amount in the STEP Mode

	Metric	Inch
× 1	0.001 mm/step	0.0001 in./step
× 10	0.01 mm/step	0.001 in./step
× 100	0.1 mm/step	0.01 in./step
× 1000	1.0 mm/step	0.1 in./step
× 10000	10.0 mm/step	1.0 in./step
× 100000	100.0 mm/step	10.0 in./step



Parameter #6009 D_4 can permit setting any multiplication factor when MANUAL PULSE MULTIPLY switch is set at X100. Multiplication factor is set by parameter #6223.

Example: When 50 is set for #6223, X1, X10, X50, can be set.

5.1.8.1 Handle Interpolation Function

5.1.8.2 General

When the handle interpolation mode is commanded while executing the program under memory mode, tape mode, or MDI mode, the handle can be turned to the plus direction to control the cutting speed of the G01, G02, or G03 commands.

5.1.8.3 Operation

(1) Select the handle interpolation mode.

The mode can be set by the setting parameter or I/O input.

Setting #6001 D4: 1 = Enable handle interpolation mode
0 = Disable handle interpolation mode

I/O signal #13066: 1 = Enable handle interpolation mode
0 = Disable handle interpolation mode

(2) Depress the cycle start button.

(3) "F" blinks to stop the machine before executing the G01 (or G02, G03) program.

(4) Turn the handle to the plus side, to perform cutting interpolation for as long as the handle is turned. The F command is overridden in proportion to the handle rotation speed.

The relation between the handle rotation speed and the override is set in parameter #6349.

(Example)

When the relation between the rotation speed of the handle and the override is "1 rotation/second = 100%", the rate becomes as follows.

Parameter #6349 = 100 (pulses/second)

1 rotation/second or more	100%
0.5 rotations/second	50%
0.01 rotations/second	1%
Less than 0.001 rotations/second	Stop

(5) The move stops when the handle is stopped or when turned to the minus side.

(6) Turn off the handle interpolation mode when the cutting ends.

Notes:

1. The move has no disconnection if the handle is turned faster than 0.2 rotation/second, but the move is disconnected if the handle is turned slower than 0.2 rotation/second.

2. When the handle interpolation mode is turned off during handle interpolation, the remaining move distance is executed by the F command.

3. When there are two handles (2-handle specification), the 1st handle alone is enabled.

4. Handle interpolation is disabled during thread cutting.

5. There is no need to select the handle axis for handle interpolation.

6. The handle interpolation mode is disregarded during automatic mode handle offset.

7. The override is normally commanded regardless of the setting on the feed override selection switch.

8. During dry run, the override changes by the dry run speed. Thus, the override varies according to the setting on the feed override selection switch and the setting of the rapid feedrate.

5.1.9 JOG PUSHBUTTONS AND RAPID PUSHBUTTON

This pushbutton is used to feed the tool manually.

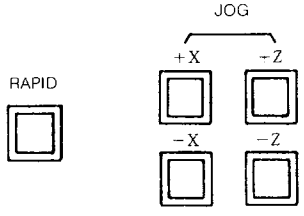
(1) With any of pushbuttons +X, -X, +Z, or -Z with RAPID button depressed, the axis can be moved rapidly until the button is released.

(2) These pushbuttons move the tool at the speed set by JOG FEEDRATE switch in the JOG mode.

(3) Each time the pushbutton is depressed in the STEP mode, the tool is moved by the value per step set by MANUAL PULSE MULTIPLY select switch. Maximum feedrate per step is determined by parameter # "6222."

Notes :

1. JOG pushbuttons work on two axes simultaneously.

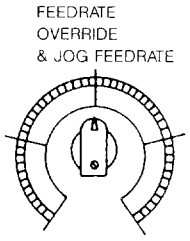


2. Parameter #6009 D3 can provide JOG command instead of RAPID command until reference point return for each axis is completed after the power is turned on. In the reference point return mode, usual RAPID operation is executed in both negative and positive directions.

5.1.10 JOG FEEDRATE SWITCH AND FEEDRATE OVERRIDE SWITCH

In the automatic operation mode (TAPE, MEM, MDI), this switch is used to adjust the feedrate by 10% from 0 to 200% of the programmed feedrate specified with an F function at whatever position the switch may be set. Feed during tapping by G32, G92, and G76 follows F command. Where OVERRIDE CANCEL switch is set on, the tool will be moved at the programmed feedrate by F code regardless of switch setting.

The JOG FEEDRATE switch is used to select the jog feedrate in the JOG mode. Up to 32 steps of feedrate can be specified. Jog feedrate depends on the machine tool. For definite values, refer to the machine tool builder's manual. See Table 5.1.10. The JOG feedrate can be preset by parameters #6233 to 6264.



5.1.10 JOG FEEDRATE SWITCH AND FEEDRATE OVERRIDE SWITCH (Cont'd)

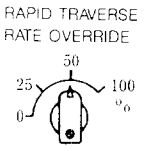
Table 5.3 Jog Speed and Feedrate Override

Step	Feedrate Override	Jog Feedrate	
		Parameter No.	mm/min
0	0%	#6233	0
1	10%	#6234	1
2	20%	#6235	2
3	30%	#6236	4
4	40%	#6237	6
5	50%	#6238	8
6	60%	#6239	10
7	70%	#6240	12
8	80%	#6241	15
9	90%	#6242	20
10	100%	#6243	25
11	110%	#6244	30
12	120%	#6245	40
13	130%	#6246	50
14	140%	#6247	60
15	150%	#6248	80
16	160%	#6249	100
17	170%	#6250	120
18	180%	#6251	150
19	190%	#6252	200
20	200%	#6253	250
21		#6254	300
22		#6255	400
33		#6256	500
24		#6257	600
25		#6258	800
26	0%	#6259	1000
27		#6260	1200
28		#6261	1500
29		#6262	2000
30		#6263	2500
31		#6264	3000

Note: Jog feedrate depends on the machine tool. For definite values, refer to the machine tool builder's manual.

5.1.11 RAPID TRAVERSE RATE OVERRIDE SWITCH

This switch is used to adjust the traverse rate by F0, 25, 50 and 100%. 100% Rate is the rapid traverse rate set by parameter #6280 and #6281. The switch is effective both in automatic operation including G00 command and in manual operation (RAPID mode). F0 is set by parameter #6231.



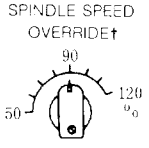
Parameter #6018 D_2 can provide six steps of traverse rates of "100%, 50%, 25%, 10%, 5%, and F_0."

5.1.12 FEEDRATE OVERRIDE CANCEL SWITCH

Turning on the FEEDRATE OVERRIDE CANCEL switch prevents the function of FEEDRATE OVERRIDE switch.
Feedrate is fixed at 100%.

5.1.13 SPINDLE SPEED OVERRIDE SWITCH †

With this switch, the current spindle speed can be changed to an override speed which is set at 10% intervals between 50 and 120% of the current spindle speed.



Parameter #6018 D_1 can provide override function in increments of 10% between 10% and 200%.

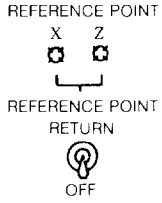
5.1.14 MANUAL REFERENCE POINT RETURN SWITCH †

This switch is for bringing the tool back to the reference point manually.

For its operation method, refer to 6.2.1 Manual Return To Reference Point.

5.1.15 REFERENCE POINT LAMPS

These lamps indicate that the tool is positioned on the reference point. They light when the tool is brought to the reference point through the manual or automatic return to reference point (G28), or by the reference point return check (G27), and goes out as the tool moves away from the reference point by a subsequent operation.



5.1.16 SINGLE BLOCK SWITCH

With this switch turned on, individual block-by-block operation is obtained. A block of data is executed each time the CYCLE START pushbutton is activated. In the automatic operation mode, the machine stops by turning on this switch after finishing the current block.

5.1.17 OPTIONAL BLOCK SKIP SWITCH

This switch selects whether the data in blocks including a "/" is disregarded or not.

(1) While the switch is on, all the commands in a block programmed after a "/" are neglected. However, block data appearing before the "/" remains effective.

(2) While this switch is off, blocks including a "/" are executed along with other blocks. This switch is ineffective on the block under execution and blocks stored in the advance-reading buffer. When this switch is turned on during an automatic operation cycle, it works on the block read after the switching on has occured.

Notes :

1. The two commands "/" and "/1" are equivalent.
2. With the control provided with the optional block skip B function†, 8 independent blocks can be skipped with the switching of the switches corresponding to "/2" through "/9."

5.1.18 DRY RUN SWITCH

Turning on the DRY RUN switch in the TAPE, MDI or MEM mode causes the tool to move at the speed selected by the JOG FEEDRATE switch, ignoring all programmed F-functions. F commands can be displayed as they are programmed. This switch may be used to check the program.

Rapid traverse (G00) rate for dry run operation can be set by setting parameter # "6006D_2."

Parameter #"6006 D_2"	Rapid Traverse at Dry Run Operation
"0"	Rapid traverse rate
"1"	Jog feedrate

Notes :

1. Switching the DRY RUN switch during automatic operation becomes effective on the current block. Switching it in mm/rev mode becomes effective on the next block.
2. Rapid traverse rate override is kept effective during dry run operation.

5.1.19 DISPLAY LOCK/MACHINE LOCK SWITCH

This switch functions to stop updating the position EXTERNAL display, or to stop move command pulses to the servos. Stop the machine to operate the switch.

"OFF"

Usual operation is made at "OFF" position in both manual and automatic operation. The machine and position EXTERNAL display operate according to the command by CYCLE START operation or manual operation.

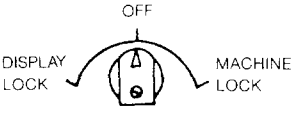
"DISPLAY LOCK"

This position is used to exclude the tool movement value from the display. Universal display is not updated, though the machine moves.

5.1.19 DISPLAY LOCK/MACHINE LOCK SWITCH (Cont'd)

"MACHINE LOCK"

With the switch at MACHINE LOCK, axis movement including Zero Return is inhibited. The position display is updated. M-, S-, and T-functions are executed. This position is selected to preset the display or to check the tape data.



5.1.20 M-FUNCTION LOCK SWITCH (AUXILIARY FUNCTION LOCK)

(1) When the M-FUNCTION LOCK switch is on, it ignores the M, S, and T commands. To check the tape data, the operation by the switch is used in combination with MACHINE LOCK function.

(2) The following M codes are executed even if the switch is set on.

a. M00, M01, M02, M30

Both its decoded signals and its BCD codes are sent out to the machine.

b. M90 to M109 (Internal processing M code)

BCD code is not sent out.

(3) Turning on the M-FUNCTION LOCK switch during automatic operation becomes effective on the block after the next block of the current block.

(4) This switch does not affect S 4-digit programming (option).

5.1.21 MANUAL ABSOLUTE SWITCH

(1) When MANUAL ABSOLUTE switch is on.

When automatic operation is restarted after interrupted by manual operation, the tool performs the rest of the command in the interrupted block from the end point of manual operation. The tool moves in parallel with the path specified by the program.

When the command of the next block is G00 or G01, the tool moves automatically to the coordinate specified by the program.

Then the operation is performed according to block of data.

When the command of the next block is G02 or G03 (circular interpolation), the interpolation is performed in parallel with program command. The tool automatically returns to the target coordinate when G00 or G01 is commanded after the interpolation.

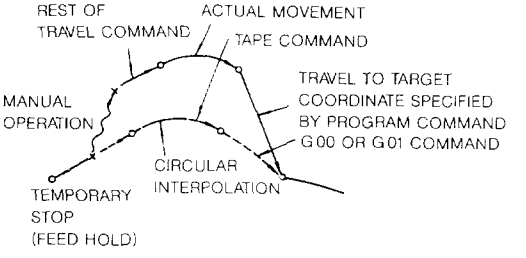


Fig. 5.2 Tool Movement with MANUAL ABSOLUTE Switch On

(2) When MANUAL ABSOLUTE switch is off

After the automatic operation is interrupted by manual operation, the coordinate system is shifted. Therefore the tool performs the reset of the travel command and continues operation in parallel with program command.

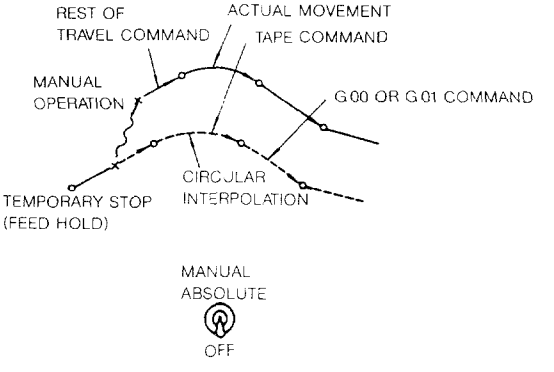


Fig. 5.3 Tool Movement with MANUAL ABSOLUTE Switch Off

5.1.22 EDIT LOCK SWITCH

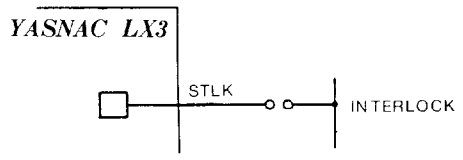
When EDIT LOCK switch is ON, the following operations cannot work. If operation is made, "EDIT LOCK" blinks.

(1) ERASE, INSERT, AND ALTER keys

(2) Storing of NC tape

5.1.23 INTERLOCK INPUT (INTERLOCK)

INTERLOCK INPUT is used for stopping the axis movement during automatic operation. Turning on the interlock (closed) during axis movement in automatic operation mode stops the axis with automatic operation activation lamp (STL) lighting. Turning off the interlock resumes the axis motion. The interlock input does not affect the movement by manual operation.



5.1.24 G50 POINT RETURN SWITCH †

This switch is for returning the tool to the coordinate system setup point (where G50 has been programmed) manually. For its usage, refer to 5.2.4, "G50 Point Return."

5.1.25 MANUAL INTERRUPTION POINT RETURN SWITCH

This switch is for manually returning the tool to where the NC was switched over from the AUTO mode to the MANUAL mode in order to make intervention with manual control. For the usage, refer to 5.2.5, "Manual Interruption Point Return."

5.1.26 PROGRAM RESTART SWITCH †

This switch is for restarting the part program from any desired sequence No. For the usage, refer to 5.2.6, "Program Restart."

5.1.27 X-AXIS MIRROR IMAGE SWITCH †

This switch is for turning on the MIRROR IMAGE function with respect to the Z axis. When this switch is turned on, the sign of all the X-coordinate command values is reversed. The mirror image function is effective on all the X-coordinate values including those by G50. This switch is effective only while the offset function is cancelled on X-coordinate values.

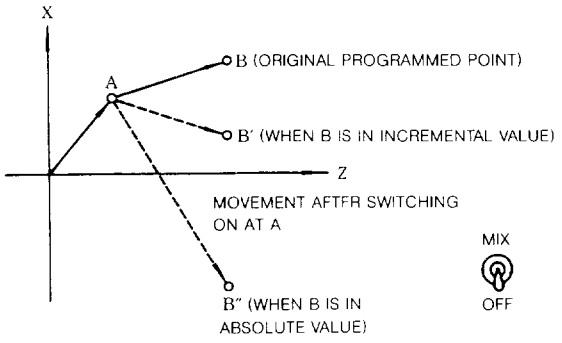


Fig. 5.4 Motion with the Switch Turned on at Point A

5.1.28 AUTO MODE HANDLE OFFSET SWITCH †

This is a switch for enabling tool motion through the use of manual pulse generator. For the usage, refer to Paragraph, 5.2.7, "Auto Mode Handle Offset."

5.1.29 CUTTING DEPTH OVERRIDE SWITCH † FOR G71 AND G72

This switch is for effective cutting depth override on the specified depth of cut command D in outside rough turning cycle (G71) and face rough turning cycle (G72) between 10 and 200% at 10% increments. However, to make this switch effective, Parameter #6023D2 should be set to 1 in advance.

5.1.30 POSITION STORE PUSHBUTTON *

This button switch is for directly inputting measured workpiece values.

When this button is pushed, the current tool values (position or external display values) are stored temporarily in the register,the MEASURED WORKPIECE VALUE DIRECT INPUT mode is turned on, and the LED of the OFS function key flickers. At the same time, the CRT displays the tool offset value diagram.

For detailed procedure, refer to 5.2.3, "Measured Workpiece Value Direct Input†."

5.1.31 FS AUTOMATIC EDITING FUNCTION

5.1.31.1 General

The effective values of the feedrate and spindle speed during automatic operation are stored and feed back to the NC part program to create the optimized program. The feedrate (F code) and spindle speed (S code) will hereafter be referred to as FS.

5.1.31.2 Input/Output

The following I/O signals are required for the FS editing.

5.1.31.2.1 Input

(1) FSCM (#13134) ... FS editing mode

Close this signal for FS editing, to prepare to store the FS and change the FS memory.

(2) FSMEM (#13135) ... FS storage

Close this signal during automatic operation, to call up the operating feedrate or spindle speed data into the internal memory.

(3) FSCH (#13136) ... FS memory change

Close this signal while holding automatic operation, to reflect the data called up during automatic operation in the NC part program.

(4) FSCLR (#13126) ... FS data clear

Close this signal to clear the stored FS data.

5.1.31.2.2 Output

(1) FSMD (#12183) ... During FS editing mode

This signal indicates that FSCM is closed, and it is ready for FS storage and FS memory editing.

When this signal is switched from "closed" to "open", the feedrate and spindle speed data are canceled.

(2) FSCE (#12184) ... End of FS memory alteration

This signal closes when the FSCH closes and the data are reflected in the NC part program, and opens again when the FSCH opens.

(3) FSCLRE (#12233) ... End of FS data clear

This signal closes when the FSCLR closes and the stored FS data are cleared, and opens again when the FSCLR opens.

5.1.31.3 Functions

Rough FS data can be programmed in the NC part program, to allow optimization of the feed override and spindle override after actual cutting.

The optimization data and position in the actual cutting program can be stored in the internal memory of the NC, and reflected in the part program by closing the FS memory alteration input.

5.1.31.4 Operation Procedure

(1) Be ready for operation.

(2) Search the operation program under the memory mode.

(3) Close FSCM.

(4) Check that FSMD is output, and start the memory operation.

(5) Change the feedrate and spindle speed for optimization.

(6) Close FSMEM when the optimization is completed.

(7) Repeat (5), (6) in sequence.

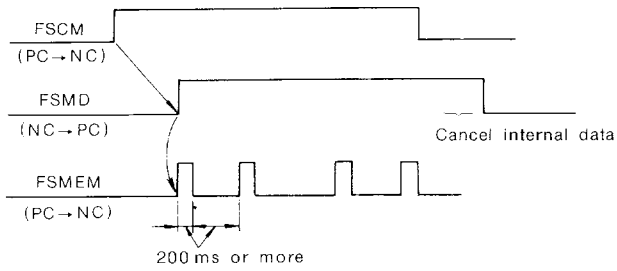
(8) Hold the operation (label skip).

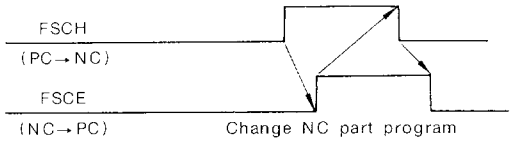
(9) Close FSCH. ... The execution value stored by FSMEM is stored in the part program.

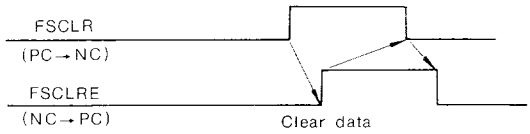
(10) Open FSCH. ... FSCE opens.

(11) Open FSCM. ... The data stored in (5), (6) is canceled.

5.1.31.5 Timing Chart







5.1.31.6 Example of FS Editing

The following two editing patterns are available, by switching parameter #6008 D6.

(1) Edit F and S each time (#6008 D6 = 0)

(a) (Before editing)

```
O1000;
N1 G28 U0 W0;
N2 G50 TS100;
N3 T0103;
N4 G40 G97 G99 S1500 M08;
N5 G00 X43.236 Z10. M03;
N6 Z1.;
N7 G01 X39.216 Z-6.5 F.15;
  -F override 80%, S override
  110%  FSMEM closed
N8 X38.864;
N9 G03 X18. Z-20.142 R19.85;
  -F override 60%, S override
  100%  FSMEM closed
N10 G01 Z-20.5;
N11 X12. Z-22.232;
  -F override 50%, S override
  80%  FSMEM closed
N12 G40 X10.F1.;
:
:
```

(b) (After editing)

```
O1000;
N1 G28 U0 W0;
N2 G50 TS100;
N3 T0103;
N4 G40 G97 G99 S1500 M08;
N5 G00 X43.236 Z10. M03;
N6 Z1.;
N7 G01 X39.216 Z-6.5 F.12 S1650;
  -F override 80%,
  S override 110%
N8 X38.864;
N9 G03 X18. Z-20.142 R19.85 F.09 S1500;
  -F override 60%,
  S override 100%
N10 G01 Z-20.5;
N11 X12. Z-22.232 F.08 S1200;
  -F override 50%,
  S override 80%
N12 G40 X10. F1.;
:
:
```

(2) Edit F and S backward (#6008 D6 = 1)

The F and S are edited back to the block where the FSMEM closed the last time.

(a) (Before editing)

```
O1000;
N1 G28 U0 W0;
N2 G50 TS100;
N3 T0103;
N4 G40 G97 G99 S1500 M08;
N5 G00 X43.236 Z10. M03;
N6 Z1.;
N7 G01 X39.216 Z-6.5 F.15;
  -F override 80%, S override
  110%  FSMEM closed
N8 X38.864;
N9 G03 X18. Z-20.142 R19.85;
  -F override 60%, S override
  100%  FSMEM closed
N10 G01 Z-20.5;
N11 X12. Z-22.232;
  -F override 50%, S override
  80%  FSMEM closed
N12 G40 X10. F1.;
:
:
```

(b) (After editing)

```
O1000;
N1 G28 U0 W0;
N2 G50 TS100;
N3 T0103;
N4 G40 G97 G99 S1200 M08;
  -S override 80% is reflected
N5 G00 X43.236 Z10. M03;
N6 Z1.;
N7 G01 X39.216 Z-6.5 F.08;
  -F override 50% is reflected
N8 X38.864;
N9 G03 X18. Z-20.142 R19.85;
N10 G01 Z-20.5;
N11 X12. Z-22.232;
N12 G40 X10. F1.;
:
:
```

5.1.31.7 Conditions of the Specifications to Perform FS Editing

(1) Conditions for the FS data to be stored in the internal memory

(a) FSMD is being output

(b) It is during automatic operation (STL is closed).

(c) The execution values of the feedrate and spindle speed are stored at the rise of the FSMEM.

(d) All the data stored in (3) are canceled at the fall of the FSCM.

(e) The maximum count of storage (count of the FSMEM) is 128 times.

(2) Conditions for the FS data to be reflected in the part program

(a) FSMD is being output.

(b) Automatic operation is being held. (Label skip.)

When the FSCH turns open → close under the above conditions, the FS data are inserted in each block, according to the stored data.

Notes :

1. When the FSMEM is closed two or more times in a single block, the newer feedrate and spindle speed become effective. The FSMEM count is not counted up.

2. When the FSMEM is closed in the subprogram, the FS data are inserted in the subprogram itself. This needs special attention, as the subprogram may also be called from other programs.

3. The FS editing of the multiple repetitive cycle can be performed in the outer form rough cutting cycle alone.

4. When the FSMEM is closed during constant surface speed control, the F of the "closed" block is changed but not the S.

5. The cycle start button is disabled while the FSCH is closed.

6. The F and S data are stored if the FSMEM is closed during cutting.

7. The S data are stored if the FSMEM is closed during rapid feed.

8. Warning is given if the FSMEM is closed for 128 times or more. No more data are stored by closing the FSMEM.

9. Closing the FSCM during automatic operation does not close the FSMD.

10. Cycle start cannot be enabled while the FSMEM is closed.

11. Warning occurs if editing is performed with the FSMD closed, after the FS editing is completed.

5.2 OPERATION PROCEDURE

5.2.1 MANUAL RETURN TO REFERENCE POINT

With this function, the tool is returned to the reference point manually. The procedure is as follows.

(1) Set the mode select switch to RAPID or JOG.

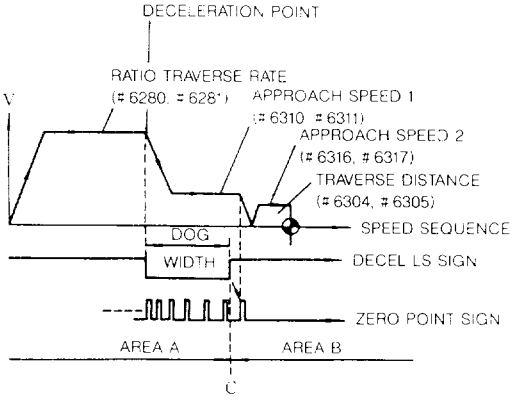
(2) Manually move the tool to a position some distance away from the reference point. When the tool is within the range A shown below, it can be brought back to the reference point in the normal way, as described below.

(3) Turn on the REFERENCE POINT RETURN switch.

(4) Keep the JOG button for the return direction depressed. The tool starts to move as in the normal manual control, but the speed is decelerated at the deceleration point, and the motion stops automatically at the reference point.

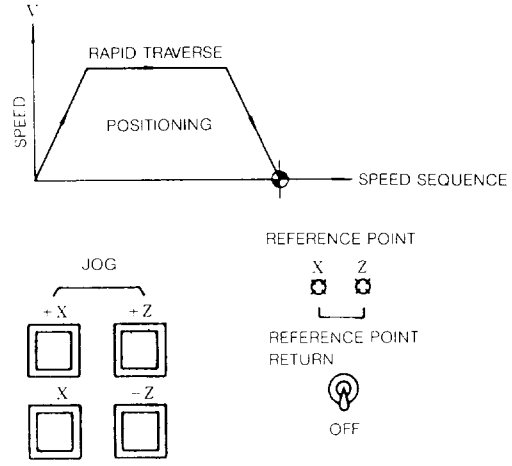
(5) Then, the REFERENCE POINT lamp for the relevant axis lights.

(6) After the second reference point return, the operation will be the same as positioning. Parameter #6010 (MZRNHS) permits the same operation as that when the power was first turned on.



Notes :

1. As long as the power is turned on, either the manual or the automatic return to reference point can be initiated, regardless of the tool position, but the tool will not return to the reference point accurately if the tool is started from a point in the area B. Be sure to bring the tool into the area A before initiating a manual or automatic return motion.
2. Once the tool is returned to the reference point, the point C is stored, and if the reference return motion is initiated from a point in the area B, this is regarded as an error. Start the reference return motion from a position in the area A.
3. Once the tool is returned to the reference point, it can not be further moved in the same direction unless the REFERENCE POINT RETURN switch is turned off.
4. While the MACHINE LOCK switch is on, the reference point return function is ineffective.
5. Do not return the tool to the reference point by the manual reference point return function, while the buffer is loaded with blocks read in advance of execution, because the stored motion data will be erased by the reference point return motion.



5.2.2 AUTOMATIC COORDINATE SYSTEM SETTING †

With this function, a new coordinate system is set up automatically upon the return of the tool to the reference point by the manual reference point return function. The coordinates of the new origin are preset with the following parameters. The coordinate system set up by this function is equivalent to the ones set up by G50.

(1) Parameters for metric system

Parameter	Meaning
#6636	X-axis coordinate
#6637	Z-axis coordinate

(2) Parameters for inch system

Parameter	Meaning
#6630	X-axis coordinate
#6631	Z-axis coordinate

(3) Axis can be selected by parameter #6015 for both metric and inch systems.

5.2.3 MEASURED WORKPIECE VALUE DIRECT INPUT †

(1) Writing into tool coordinate memory

With this function, coordinate data for the respective tools required for setting G50T work coordinate systems can be written into the TOOL COORDINATE MEMORY by simple processes. The required processes are as follows.

a. Bring the tool slide to the START point.

b. Set the current coordinate displayed values as (0, 0).
The "current coordinate displayed values" as meant here are values displayed as POSITION EXTERNAL on the CRT. To reset these values, push the address key for the desired axes, and then, push the ORG key, while the data are being displayed on the CRT. With this keying, the coordinate values become (0, 0).

5.2.3 MEASURED WORKPIECE VALUE DIRECT INPUT (Cont'd)

c. Push the OFS function key.

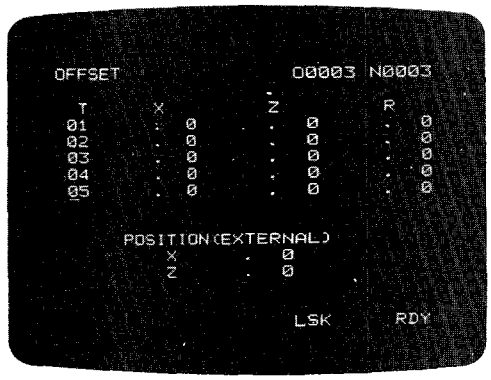


Fig. 5.5

The tool offset values will be displayed as shown above. However, the POSITION EXTERNAL values in the lower part of the CRT will have been changed to 0 for both axes, by the keying under the operation shown in b.

d. Select the HANDLE/STEP JOG mode.

e. Select a tool for which tool coordinate data are to be written in.

f. Start the spindle, and test-turn the workpiece circumference (surface A) with that tool in the MANUAL mode.

g. Push the CURRENT VALUE STORE button (PST input) on the machine control station first, and then, retract the tool and stop the spindle.
--- When the CURRENT VALUE STORE button is pushed, the displayed POSITION EXTERNAL values are temporarily stored in the register, and at the same time, the LED for the OFS function key starts to flicker, and the DIRECT INPUT mode is turned on.

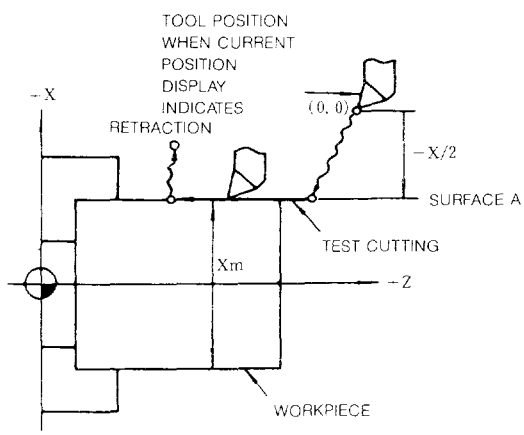


Fig. 5.6

In the example shown above, the -x value is temporarily stored in the register.

h. Measure the outer diamter of the turned workpiece and read the value "xm."

i. Select the desired tool coordinate memory No. (one among offset Nos. 51 through 80). For example, if the tool coordinate memory No. 51 is desired, key-in 5 , 1 and depress the CURSOR ↓ key.

j. Then, proceed to key-in x , measure value (xm) and WR.
--- The NC makes the calculation to obtain the tool coordinate data expressed by the following equation, and stores them into the specified tool coordinate memory X-axis. The writing of the required X-axis data has been completed.

$$\left\{\begin{matrix}\text{Tool coordinate} \\ \text{data}\end{matrix}\right\} = \left\{\begin{matrix}\text{Keyed-in} \\ \text{measured value}\end{matrix}\right\} - \left\{\begin{matrix}\text{Current values temporarily} \\ \text{stored in register}\end{matrix}\right\}$$

--- Equation A

k. Repeat the same processes on the Z-axis by test turning the end face (surface B) of the workpiece with the tool.

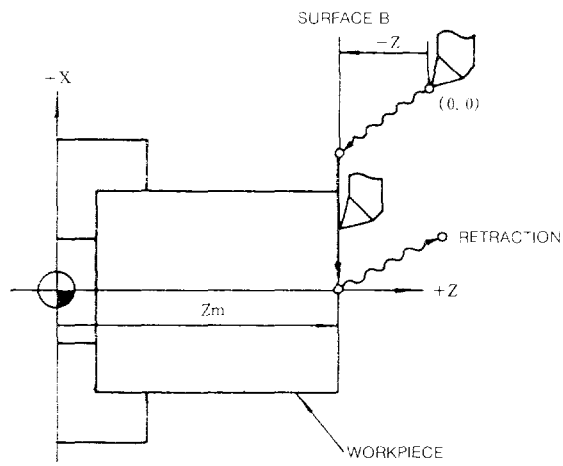


Fig. 5.7

l. Push the CURRENT VALUE STORE button again, and then, retract the tool and stop the spindle.
--- The current values at the time of the button pushing are stored again in the register temporarily.

m. Measure the distance "Zm" between the desired work coordinate origin to the test cut surface (surface B).

n. Key-in Z , measured value (Zm), and WR in this sequence.
--- The NC performs the same calculation as before with respect to Z-axis, and stores the results in the Z-axis of the tool coordinate memory.

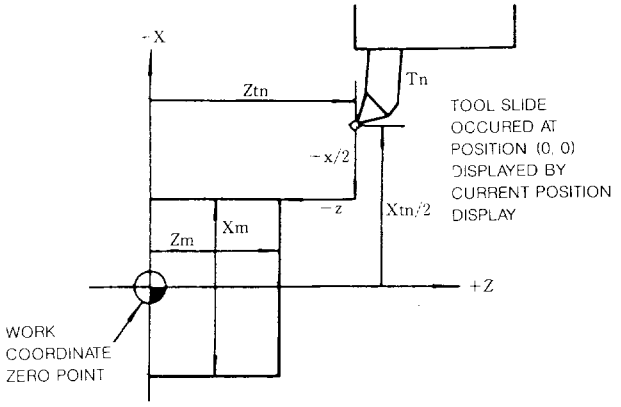


Fig. 5. 8

With the above processes, all the data for one tool has been written into the tool coordinate memory.

o. Repeat the processes e. through o. for all the tools to write their data into the tool coordinate memory.

p. After writing all the tool data, push the RESET key to cancel the DIRECT INPUT mode.

With the above processes, Xtn and Ztn values for all the tools as shown below are stored in the tool coordinate memory.

The reason for this is that the following calculations have been performed.

$$Xtn = Xm - (-x) = Xm + x$$

$$Ztn = Zm - (-z) - Zm + z$$

Notes :

1. The above example is based on test cutting. The method using a gauge or a workpiece of known dimensions is also convenient.

2. When tool positions for test cutting of contact are on the minus coordinate area, key-in the measured values in negative.

3. When Parameter #6020D$_2$ is "0," tool offset memory (offset Nos. 01 through 50) can be written in the same way as above. When parameter #6020D is "1," the following calculation equation is used only for tool offset memory (offset Nos. 01 through 50).

$$\begin{bmatrix}\text{Stored}\\ \text{data}\end{bmatrix} = \begin{bmatrix}\text{Current value temporarily}\\ \text{stored in the register}\end{bmatrix} - \begin{bmatrix}\text{Keyed-in}\\ \text{value}\end{bmatrix}$$

--- Equation B

As can be seen, in Equation B, sign is reversed from that in Equation A given before.

4. When parameters are not used, and the CURRENT VALUE STORE button is pushed, the data are always calculated by Equation A for writing into the TOOL COORDINATE MEMORY (51 - 81) and the WORK COORDINATE SYSTEM SHIFT MEMORY (00).

5. When the CURRENT VALUE STORE button is pushed, the OFS function is automatically turned on, its LED blinks, and the tool offset value is displayed.

(2) Automatic writing into tool offset memory

When Parameter #6020D$_2$ is set to "1," tool offset values are automatically written into the tool offset memory by the following processes. The tool offset values written in this case are distance between the position of the reference tool and the tool under consideration.

a. Bring the reference tool to any desired reference point by manual control.

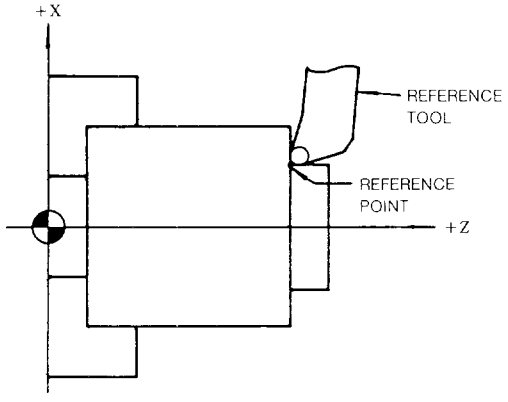


Fig. 5. 9

b. Reset the current coordinate values (the displayed POSITION EXTERNAL values) to (0, 0).

5.2.3 MEASURED WORKPIECE VALUE DIRECT INPUT † (Cont'd)

c. Push the OFS function key.

--- The displayed POSITION EXTERNAL values in the lower area of the CRT have been changed to (0, 0).

d. Retract the reference tool, and bring the tool by manual control to where it can be replaced conveniently.

e. Select a tool with which offset value should be written in.

f. Bring the selected tool manually to the reference point, push the CURRENT VALUE STORE button, and retract the tool.

--- The current values at the time of button pushing are temporarily stored in the register.

g. Select a "tool offset No." for writing (one among 01 through 50).
For example, to write into tool offset No. "02," key-in 0 , 2 , and CURSOR [↓] .

h. Then, go on keying as follows.

X , 0 , and WR ··· (for X axis calculation and storing)
Z , 0 , and WR ··· (for Z axis calculation and storing)

--- Now, the control performs the following calculation separately for X and Z axes, and stores the results to the specified tool offset memory.

$$\begin{bmatrix}\text{Stored}\\ \text{data}\end{bmatrix} = \begin{bmatrix}\text{Current values temporarily}\\ \text{stored in register}\end{bmatrix} - \begin{bmatrix}\text{Keyed-in}\\ \text{value}\end{bmatrix}$$

However, since no value is keyed in the above process,the "current values temporarily stored in register" only are stored as tool offset values.

i. By repeating the processes i. through g. with all the tools, their tool offset values can be automatically written. After completing the processes, push the RESET key. The values written by the above processes are all differential distances between the reference tool and the actual tools in consideration.

Notes : For tools with which tool nose directions are different from that of the reference tool, prepare a 2nd reference point as shown below, and bring the tool to that point manually.

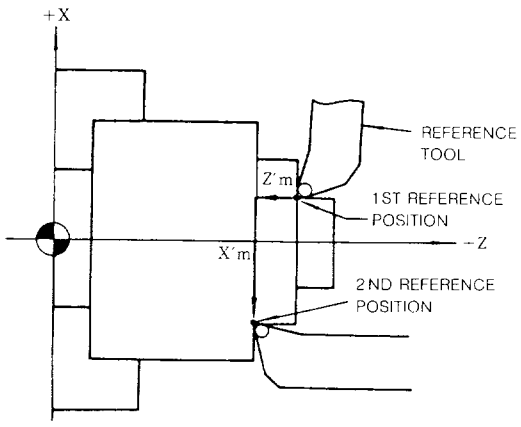


Fig. 5. 10

Push the CURRENT VALUE STORE button at that position, and then, retract the tool.

Then, make the following keying, in place of the ones given in g. above.

[X] , [measured value (Xm')] , [WR]
[Z] , [measured value (Zm')] , [WR]

where Xm' and Zm' are distances with signs from the 1st reference point to the 2nd reference point. In the example shown above, these values have minus signs. With the above processes, the distances between the reference tool and a tool having different point direction are stored in the specified tool offset memory.

5.2.3.1 Tool Setter Function

Automatic writing into the tool coordinate memory is enabled simply by equipping the tactile sensor and letting the tool nose touch the tactile sensor by manual operation.

This function simplifies the preparatory work when changing the tool.

Writing into the work coordinate system shift memory is also done automatically.

This function can be used to simplify operation with various machining methods and workpieces.

5.2.3.2 Parameters

When using the tool setter function, parameters must be set in the NC for automatic operation.

Set the distance of each contact of the tactile sensor from the projected zero point, with sign.

(1) Setting parameters

#6618 ... Position where contact point "XP +" turns on. (diameter value)
#6619 ... Position where contact point "XP -" turns on (diameter value)
#6620 ... Position where contact point "ZP +" turns on.
Set the distance from "ZP-" with sign.
#6621 ... Position where contact point "ZP-" turns on.
Set "0" so that this point becomes the temporary Z "0" point of the tactile sensor.

"1" = 0.001 mm (metric input)
0.0001 in. (inch input)

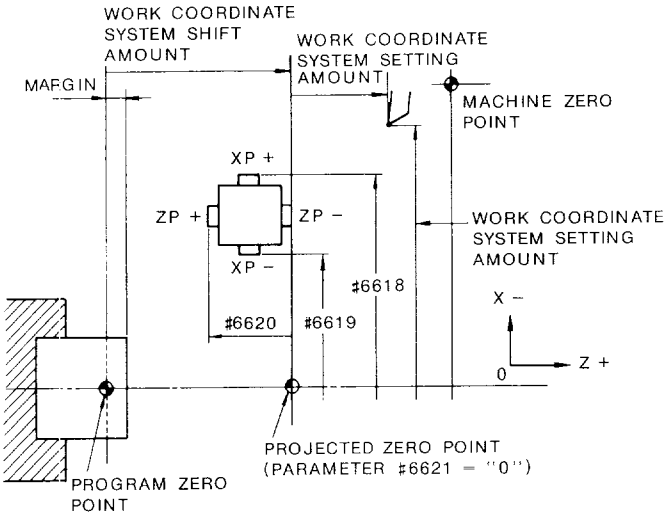


Fig. 5.10 A Tool setter

5.2.3.3 Automatic Writing into the Tool Coordinate Memory

The setting amount of the work coordinate system is automatically written into the tool coordinate memory, by the following operation.

(1) Perform manual zero point return.
POSITION (ABSOLUTE), POSITION (MACHINE) becomes "0."

(2) Turn on the measurement mode switch.
It automatically switches to the OFS function, to cause the OFS lamp, to blink.

(3) Move the presetter down.
The tool coordinate memory number corresponding to the currently selected tool number is automatically selected and displayed.
Note that automatic selection cannot be made if parameter #6020 D3 is "0."

(4) Select the tool to be measured.
If the tool coordinate memory number is not automatically selected, move the cursor to the desired tool coordinate memory number using the cursor key and page key.

Note: When parameter #6020 D3 is "1" and the tool coordinate memory number is automatically selected, manual memory number selection using the cursor key and page key is prohibited.

(5) Contact the tool nose on "XP+" or "XP-" by manual operation.
The feed stops and the buzzer sounds when the tool nose touches the sensor, to automatically write the value operated by the following expression into the tool coordinate memory "X" where the cursor is set.
[Tool coordinate memory value] = [Parameter #6618 or #6619] - [POSITION (MACHINE) value when the tactile sensor turned ON]
The value of the offset memory "X" of coordinate memory number -50 is then "0." This occurs only when both D3 and D6 of parameter #6020 are "1."

Note: When the sensor touches the tool nose, axis interlock is applied on the corresponding direction, to prohibit the move.

(6) Contact the tool nose on "ZP+" or "ZP-" by manual operation.
As in (4), the value operated by the following expression is automatically written into the tool coordinate memory "Z" where the cursor is set.
[Tool coordinate memory value] = [Parameter #6620 or #6621] - [POSITION (MACHINE) value when the tactile sensor turned ON]
The value of the offset memory "Z" of coordinate memory number -50 is then "0." This occurs only when both D3 and D6 of parameter #6020 are "1."

(7) Perform (4), (5), and (6) on all tools used.

(8) Return the in-machine presetter to the zero point position.

(9) Turn off the measurement mode switch.
The blinking of the OFS lamp stops.

5.2.3.4 Automatic Writing into the Work Coordinate System Shift Memory

The Z-axis work coordinate system shift amount can automatically be written by the following operation.

This operation is to be performed after writing into the tool coordinate memory is completed.

(1) Turn on the measurement mode switch.

The OFS lamp blinks.

When doing this immediately after writing into the tool coordinate memory, perform the following process, after Par. 4.2.3.3 (3) to (8).

(2) Check that the offset screen displays the OFFSET (WORK - SHIFT) screen.

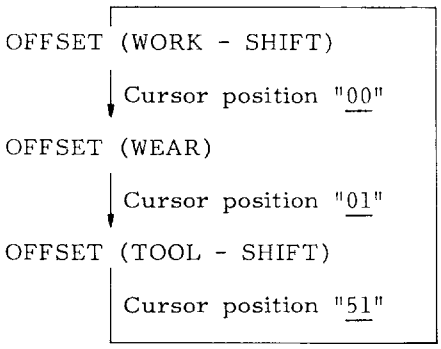
(3) Select the tool that can perform end face cutting on the workpiece.

When manual cursor selection (parameter #6020 D3 = "0") is selected, move the cursor to the tool coordinate memory number corresponding to the selected tool.

After setting the cursor, depress the OFS function key once more. The offset screen will display the OFFSET (WORK - SHIFT) screen.

Note :

The offset screen changes in the following cycle, each time the OFS function key is depressed.



The cursor position is as shown above.

This function cannot be used when the tool setter writing mode is selected in the automatic cursor selection specification.

(4) Chuck the workpiece to be machined.

(5) Cut the end face of the workpiece a little by manual operation.

(6) Depress the "PST" switch.

The buzzer sounds, and the value operated by the following expression is automatically written in the work coordinate system shift memory "Z."

[Work coordinate system shift amount "Z"] = -[Value of tool coordinate memory "Z" selected by the cursor] - [Value of POSITION (MACHINE) when "PST" is depressed]

(7) Release the tool.

(8) Turn off the measurement mode switch.

The OFS lamp stops blinking.

Performing work coordinate system setting B by the above operation sets coordinate system where the end face of the workpiece is the zero point.

Shifting the program zero point from the end face of the workpiece can be performed as follows.

Rewrite the value of "Z" in the work coordinate system shift memory, using "W."

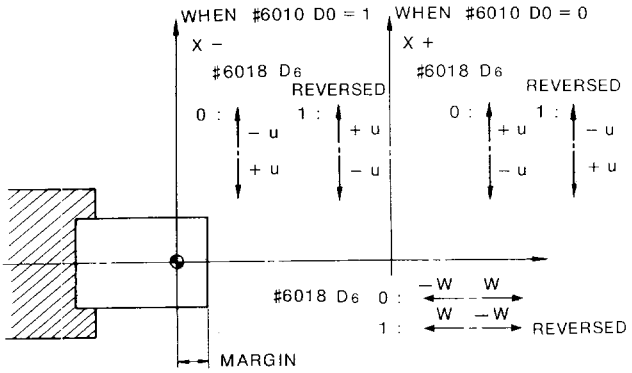
Enter the distance from the program zero point to be shifted to the end face of the workpiece (margin) with sign, following W.

The entered numeral is added to the work coordinate system shift memory. The sign is determined by #6018 D6.

When #6018 D6 = 0
Workpiece shift + Margin (W with sign)
= 1
Workpiece shift - Margin (W with sign)

The value of the work coordinate system shift memory "X" is rewritten by "U" in the same manner.

When #6018 D6 = 0
Workpiece shift + Margin (U with sign)
= 1
Workpiece shift - Margin (U with sign)



#6010 D0 X-axis zero point return direction 1 : Negative
0 : Positive

Fig. 5.10B Example of automatic writing into the work coordinate system shift memory

5.2.3.5 Coordinate System Setting (G 50 X__ . Z__ .)

Parameter setting can be made to determine whether to add the work coordinate system shift when setting the coordinate system by G50.

When #6018 D7 = 0
The work coordinate system shift is added.
When #6018 D7 = 1
The work coordinate system shift is not added.

5.2.3.6 Position (EXTERNAL) "0" Setting

The POSITION (EXTERNAL) can be set to "0" upon the first manual zero point return after supplying the power.

Set "1" in parameter #6005 D4 to used this function. The "0" setting is also performed by commanding M53 for G51 return point setting in the program. Always command M53 alone.

5.2.3.7 Work Coordinate System Setting B

(1) Command

G50 T□□ xx;
- xx: Work coordinate system offset number
- □□: Work coordinate system setting number

The following value is set in the coordinate system by giving this command.

[Coordinate system setting value] =
[POSITION (MACHINE) value] +
[Tool coordinate memory value] +
[Work coordinate system offset value] +
[Work coordinate system shift amount]

Offset memory number .....

Normally, there is no need for the work coordinate system offset.

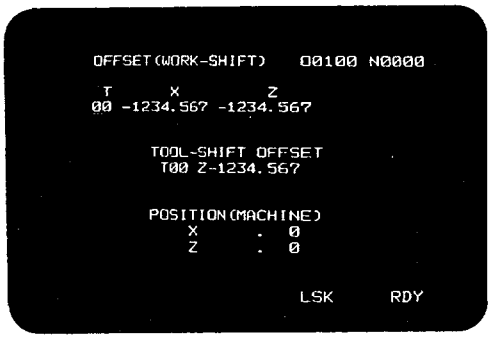
Command as follows, if the work coordinate system offset is not to be performed.

G50 T□□ 00;
- 00: Command zero

Offset amounts of Nos. 41 to 50 are used for the work coordinate system offset, but the offset can be made by Nos. 01 to 50.

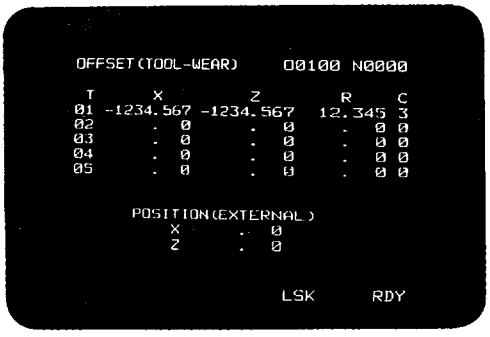
5.2.3.8 Offset Screen Display

The offset screens as shown in Figs. 5.10C, 5.10D and 5.10E are displayed by adding the tool setter function.



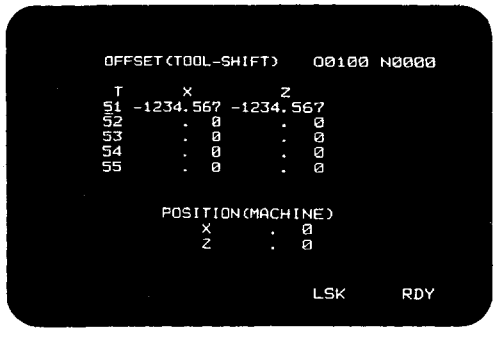
588-355

Fig. 5.10C 1st screen (work coordinate system shift memory display screen)



588-356

Fig. 5.10D 2nd screen (tool offset shift memory display screen)



588-357

Fig. 5.10E 3rd screen (tool coordinate system memory display screen)

5.2.3.9 New Tool Setter Function

5.2.3.10 General

The offsets and work coordinate system shifts can also be written automatically into the memory in machines without the tactile sensor, by manual operation.

This function allows easy preparatory operation upon tool exchange.

In this function, the contents of the tool coordinate memory and the work coordinate system shift differ from those of the conventional "tool setter function" with tactile sensor.

The contents of the conventional tool coordinate memory are changed from the differential distance from the tactile sensor to the selected tool nose, to the differential distance from the workpiece end face to the tool nose. The work coordinate system shift amount is changed from the shift from the projected zero point to the shift from the workpiece end face.

However, the theory of the coordinate system has not changed. This function is called the "new tool setter function."

5.2.3.11 Writing into the Tool Coordinate Memory

The amount in the tool coordinate memory can be written by the following manual operation, with or without the measuring tool for the in-machine setter.

The tool coordinate memory amount written on this occasion, is the differential distance from the workpiece end face to the tool nose.

5.2.3.12 Writing Procedure

(1) Perform manual zero point return.

(2) Mount the workpiece.

(3) Set "1" in the tool coordinate memory writing mode setting (6001 D6). This enables the writing mode. The screen then automatically switches to the " OFS " function OFFSET (TOOL - SHIFT) screen (Fig. 5.10F), to cause the " OFS " lamp to blink.

(4) Select the cutting tool.

The tool coordinate memory corresponding to the currently selected tool post number (tool post number entered in #1317) is automatically selected.

However, this function is effective only when parameter #6020 D3 is "1."

When the tool coordinate memory is automatically selected, memory number selection using the cursor or page key is prohibited.

When the tool coordinate memory is not automatically selected, move the cursor to the tool coordinate memory number, using the cursor key and page key.

(5) Cut the surface of the workpiece.

(6) To stop the spindle, do nothing with the X-axis, while releasing only the Z-axis.

(7) Measure the diameter of the workpiece.

(8) If the above setting is not on, select the [T] ** of the tool coordinate memory screen of the " OFS " function by manual operation.

(9) Then enter " [MX] " workpiece diameter " [WR] ", to automatically write the tool coordinate memory "X" by the following calculation.

(Tool coordinate memory "X") = (Workpiece diameter) - (Position machine "X" value)

(10) Cut the end face of the workpiece once.

(11) To stop the spindle, do nothing with the Z-axis, while releasing only the X-axis.

(12) Then enter " [MZ] " [Ø] " [WR] ", to automatically write the tool coordinate memory "Z" by the following calculation.

(Tool coordinate memory "Z") = - (Position machine "Z" value)

(13) Perform the above (4) through (12) on both the X and Z axes of all the tools.

(14) Set "0" in the tool coordinate memory writing mode setting (#6001 D6). The writing mode is canceled, to end the blinking of the " OFS " lamp.

Note: The setting of the tool coordinate memory writing mode is not effective if no zero point return is performed after the power is supplied.

5.2.3.13 Writing the Z-axis Work Coordinate System Shift Amount

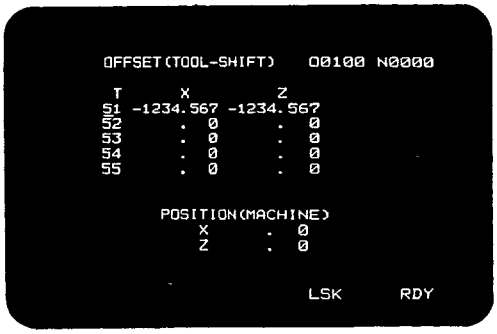
5.2.3.14 Writing Procedure

(1) Perform manual zero point return.

(2) Mount the workpiece.

(3) Set "1" in the workpiece coordinate system shift writing mode setting (#6001 D5). (Turn on the "measurement 2" key in compact conversational mode, to enable the writing mode).

It automatically switches to T00 of the "OFFSET (WORK - SHIFT)" screen of the " [OFS] " function, to cause the function key " [OFS] " lamp to blink.



588-355

Fig. 5.10 F Work coordinate system shift display screen

(4) Cut the end face of the recaps once.

(5) Do nothing with the Z-axis, while releasing only the X-axis.

(6) Measure the margin of the workpiece.

(7) Then enter " [P] "" [Z] " margin " [WR] ", to automatically write the tool coordinate memory "Z" by the following calculation.

THe tool offset memory corresponds to the tool post number entered in #1317.

(Work coordinate system shift memory "Z") = (Margin) - (Tool coordinate memory "Z") - (Position machine "Z" value)

(8) Changing the Z-axis work coordinate system shift amount

a. When the program zero point needs further shifting (for example, when there is clearance from the workpiece end face), the work coordinate system shift memory of both the X and Z axes can be rewritten by MDI operation. Calculate the shift amount,

depress the " [U] " or " [W] " key, and

input the numeral, to add the entered value to the currently displayed value.

The sign for the addition is determined by #6018 D6.

When #6018 D6 = 0
Workpiece shift + Margin (W with sign)
When #6018 D6 = 1
Workpiece shift - Margin (W with sign)

The value of the work coordinate system shift memory "X" is rewritten by " [U] " in the same manner.

when #6018 D6 = 0
Workpiece shift + Margin (U with sign)
When #6018 D6 = 1
Workpiece shift - Margin (U with sign)

b. The " [WR] " key and numeral keys are normally effective.
Fine adjustment of the margin can be made by adding or subtracting with the " [U] " or " [W] " keys, or entering the margin once more with " [P] "" [Z] " margin " [WR] ".

(9) Set "0" in the work coordinate system shift memory writing mode setting (#6001 D5) (turn off the "measurement 2" key in conversational mode), to cancel the writing mode and to end the blinking of the " [OFS] " lamp.

(10) Perform manual zero point return after changing the work coordinate system shift memory, to automatically set the work coordinate system.

Note that the work coordinate setting by manual zero point return is not performed when parameter #6011 D0 = 1. (When the tool coordinate memory number follows the lower order two digits of T**$$, refer to the <Instruction manual for coordinate system setting>.)

5.2.3.14 Writing Procedure (Cont'd)

Note: The work coordinate system shift memory value writing mode setting is invalid if a single zero point return is not performed after power supply.

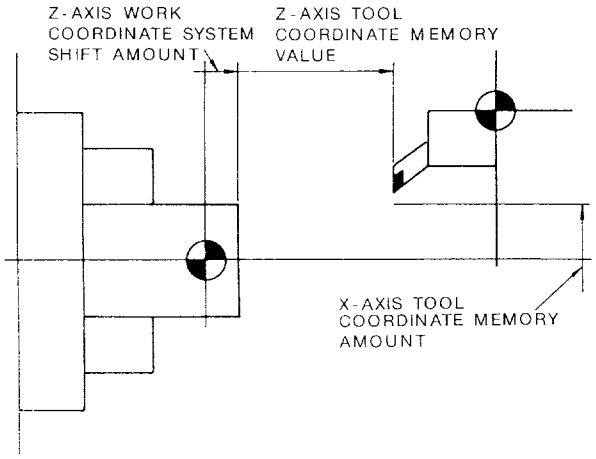


Fig. 5.10 G Coordinate system

5.2.4 G50 POINT RETURN *

With this function, the tool can be brought back to the start point of the program (where coordinate system was set up), from any position along the automatic operation tool locus. This is convenient when an automatic machining cycle is interrupted due to tool breakage, etc. and, after replacing the tool, etc., the automatic machining cycle is to be started from the beginning.

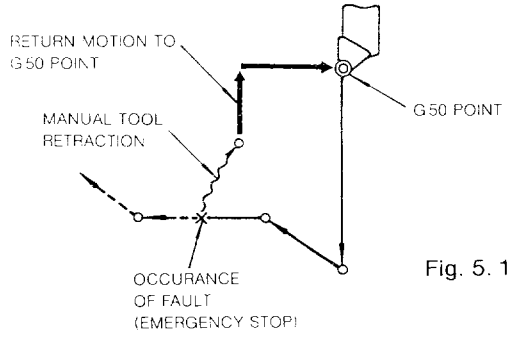


Fig. 5.11

The operation procedure for this return is as follows.

(1) Interrupt the automatic operation cycle by pushing the emergency stop button.

(2) Turn on the MANUAL mode.

(3) Retract the tool by the manual tool motion control.

(4) Eliminate the cause of the machining process interruption, and make the machine and workpiece ready for operation.

(5) Turn the G50 POINT RETURN switch on.

(6) Turn on the MANUAL JOG mode.

(7) Push one or two JOG keys that correspond to the required return motion direction among the four (+X, -X, +Z, -Z). The tool moves at a preset speed towards the G50 point, and stops at the G50) point.

(8) Turn off the G50 point RETURN switch.

(9) Turn on the AUTO mode, and push the RESET key on the MDI & CRT panel. (In the TAPE mode, reset the tape to the leading end.)

(10) Push the CYCLE START key to restart the automatic machining cycle.

Notes :

1. When the tool is at the G50 point, JOG motion control is not effective unless the G50 POINT RETURN switch is turned off.

2. The point where a coordinate system setup process has been executed latest is regarded as the G50 point. Therefore, in the following cases, the tool will not return to the start point of the machining cycle by this function.

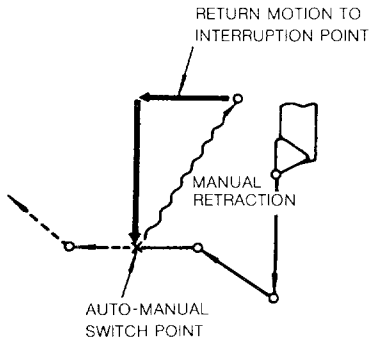
   a. The respective tools are set for different coordinate systems.

   b. When the ORG key is pushed after the interruption of the automatic cycles.

   c. When a coordinate system has been set up automatically.

5.2.5 MANUAL INTERRUPTION POINT RETURN †

With this function, automatic machining cycles can be interrupted and the tools can be retracted for workpiece measurement, chip removal, etc., and then, the tools can be brought back to the interruption point.



The procedure for this function is as follows.

(1) Interrupt the automatic machining cycle by any of the following methods.

(i) Turning on the SINGLE BLOCK switch

(ii) Pushing the FEED HOLD key

(2) Turn on the MANUAL mode.

(3) Retract the tool by manual control.

(4) Measure the workpiece, remove chips, or perform any other required operations.

Note: When the mode is switched from AUTO to MANUAL, the spindle conditions, etc. may change. For these details, refer to the operation manual of the machine tool prepared by the machine builder.

(5) Turn on the MANUAL INTERRUPTION POINT RETURN switch.

(6) Turn on the MANUAL JOG mode.

(7) Push the relevant one or two of the JOG keys (+X, -X, +Z, -Z). The tool moves towards the interruption point at the preset speed, and stops upon arrival at the interruption point.

(8) Turn off the MANUAL INTERRUPTION POINT RETURN switch.

(9) Turn the AUTO mode, and push the CYCLE START key to restart the automatic machining cycle.

Notes :

1. When the tool is at the interruption point, the JOG keys are ineffective, unless the MANUAL INTERRUPTION POINT RETURN switch is turned off.

2. When the RESET key on the MDI & CRT panel is pushed or an external reset input is received after the switching from the AUTO to MANUAL Mode, the MANUAL INTERRUPTION POINT RETURN function becomes ineffective thereafter.

3. When the tool has been manually retracted after a switching from the AUTO to MANUAL mode, then, the switch is returned to AUTO again, and thereafter, the switch is set again to MANUAL for manual tool motion, the point where the mode switch has been switched from AUTO to MANUAL last is regarded as the INTERRUPTION point.

5.2.6 PROGRAM RESTART †

With this function, when automatic machining cycles are interrupted due to tool breakage, chip entanglement, etc., by the EMERGENCY STOP button, the automatic machining cycles can be restarted, not from the beginning of the program, but from the beginning of the interrupted block.

(1) OPERATION PROCEDURE

a. Interrupt the automatic machining cycle by one of the following two functions.

(i) Emergency stop

(ii) Resetting

When the machine has been stopped by the emergency stop function, turn on the servo power supply and reset the ALARM code to make further machine motion possible.

b. Turn on the MANUAL mode and retract the tool.

c. Replace the broken tool, remove chips, etc. to prepare the machine for further operation.

When the tool is replaced, the offset values may have to be corrected.

d. Move the tool by manual control to return it to the start point of the part program. However, with a program in which G50T □□□□ is used, the tool need not be returned to the start point, and may be left at the position where chip removal, etc. has been made.

5.2.6 PROGRAM RESTART † (Cont'd)

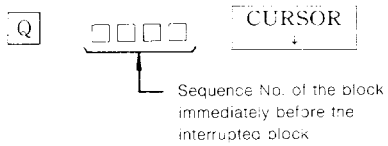
e. Turn on the AUTOMATIC mode.

f. Turn on the PROGRAM RESTART switch.

g. Push the PROG function key on the CRT & MDI panel.

h. Where the automatic operation is under the control of a tape, set the leading end of the tape to the tape reader.

i. Make the following keying on the CRT & MDI panel.



The NC starts the preparation from the starting end of the program to the block specified by Q. Upon the completion of the preparation, the CRT display changes to the "PROGRAM RETURN" page of the "POS" function to display the program restart data.

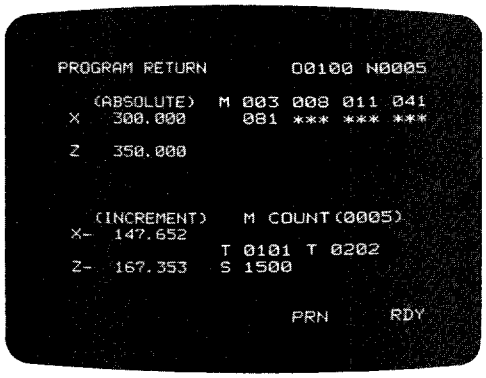


Fig. 5.12

Display of program restart data

(i) (ABSOLUTE): Display of current tool position

(ii) (INCREMENT): Display of the distance from the current tool position to the end point of the block specified by Q □□□□

(iii) Display of all the M codes and the number of M code commands programmed between the leading end of the restart program to the block specified by Q □□□□. However, if there are more than 28 M codes, those 28 M codes immediately next to the block specified by Q □□□□ in the program are displayed.

(iv) Display of the last programmed T command and the one before preceding the block specified by Q □□□□

(v) Display of the last programmed S command up to the block specified by Q □□□□

Note: M and T commands are displayed in the programmed sequence. Therefore, the last displayed one is the one programmed immediately before the block specified by Q □□□□

j. Turn off the PROGRAM RESTART switch.

k. Command required M, S, and T obtained them from the displayed program restart data as follows.

(i) Turn on the MDI mode.

(ii) Push the PROG function key on the CRT & MDI panel, and input the required M, S and T commands.

(iii) Push the CYCLE START key, and execute the M, S and T commands.

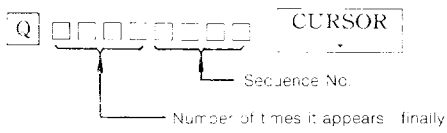
(iv) Push the POS function key on the CRT & MDI panel to revert the CRT to the program restart data display.
Check the conditions of the machine again.

l. Turn on the AUTOMATIC mode again.

m. Push the CYCLE START key. The tool moves to the PROGRAM RESTART position displayed in the (INCREMENT) column, by moving at JOG speed first along the X-axis, and then along the Z-axis, and then, restart the machining cycle from the block immediately following the block specified by Q □□□□.

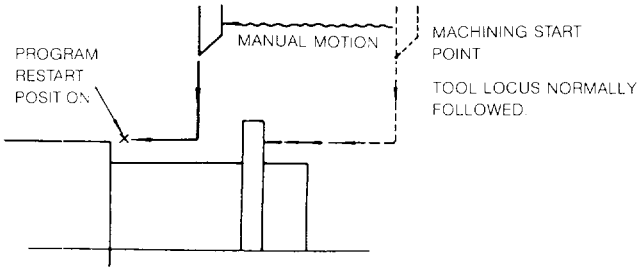
(2) Supplementary explanation

If the same sequence No. as for the block to be input by Q □□□□ is used several times in the program, input the command in the following form.



Notes :

1. Before letting the tool move to the PROGRAM RESTART position by motion along the X and Z axes in succession, be sure to check the tool for freedom from interference with the workpiece.
   If there is a possibility of a tool interference with the workpiece, first, move the tool in the MANUAL mode to a position from where it can move without interference, before bringing the tool to the PROGRAM RESTART position in the AUTOMATIC MODE.



   For the manual tool motion, turn on the MANUAL ABSOLUTE switch. If this switch is not turned on, the PROGRAM RESTART position will be shifted by the distance covered by the manual motion. For the manual motion required to avoid tool collision on the workpiece as described above, the following procedure is recommended.

   (i) Turn on the SINGLE BLOCK switch.

   (ii) Push the CYCLE START key to move the tool along the X-axis.

   (iii) Check the position.

   (iv) Push the CYCLE START key to move the tool along the Y-axis.

   (v) Check the tool position for the PROGRAM RESTART position.

   (vi) Turn off the SINGLE BLOCK switch, and push the CYCLE START key to restart the machining cycle.

2. If a reset process is executed after the display of the PROGRAM RESTART data by the keying of Q,□□□□, CURSOR [↓] the data is cancelled. When this happens, make the PROGRAM RESTART data display keying again.

3. While the NC is making the preparation for PROGRAM RESTART after the keying of Q,□□□□, CURSOR [↓], do not push the FEED HOLD key, turn the MODE switch, or make other manipulations, If these manipulations are made, make the PROGRAM RESTART data display keying again.

4. When the tool offset values are changed to compensate for the new tool, etc., the PROGRAM RESTART position is, needless to say, shifted accordingly.

5. If the PROGRAM RESTART switch is on, the CYCLE START key is ineffective.

6. In principle, do not move the machine with the MACHINE LOCK switch turned on, before and after pushing the PROGRAM RESTART switch.

7. If the block specified by Q,□□□□, CURSOR [↓] is not found, alarm code"120" is displayed.

8. Only those M and T commands which are output to the outside are displayed as PROGRAM RESTART data.
   Those M commands (M90 - M109) and T commands which are internally processed are not displayed.

9. Although not common with lathe operation, sometimes the power supply is turned off after interrupting the automatic cycle, and is turned on again before restarting the automatic cycle. In this case, be sure to return the tool to the reference point once, before starting the PROGRAM RESTART process.
   In this case, be sure to start the PROGRAM RESTART process even with those programs in which work coordinate system is used.

10. During the time after the display of the PROGRAM RESTART data till the start of the return motion to the PROGRAM RESTART position, the machine cannot be moved in the MDI mode. If the machine is returned to the reference position, alarm code "124" will be displayed.

11. Blocks in complex canned cycle programs for finish shapes cannot be specified by Q,□□□□, CURSOR [↓].

12. When the block in tool nose radius compensation is specified by Q, four digits, and CURSOR key, the machine cannot be moved in the MDI mode after the display of the PROGRAM START data.
    In this case, restart from the block before making tool radius compensation.

5.2.7 AUTO MODE HANDLE OFFSET

With this function, the handwheel for the manual pulse generator can be turned during an automatic operation cycle under the control of a tape, MDI or memory, to superimpose certain feed distances to the programmed feed distances. With this function, workpiece mounting errors, etc. can be compensated.

For this function, the required manual operations are as follows.

(1) Turn on the AUTO MODE HANDLE OFFSET switch.

(2) Select the axis along which motion is desired by the HANDLE AXIS SELECT switch. --- If the control is provided with the SIMULTANEOUS 2 AXES CONTROL MANUAL PULSE GENERATOR, the manual motion can be made along the two axes simultaneously.

(3) Select the movement distance per graduation of the handwheel with the MANUAL PULSE MULTIPLY switch. With this switch, the move distance per graduation can be selected among 1, 10 and 100 pulses.

(4) When the handwheel is turned, the tool motion along the axis selected in process (2) is superimposed on the programmed feed distance.

Turning CW: in plus direction
Turning CCW: in minus direction

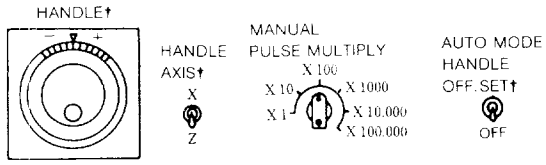


Fig. 5.13

Notes :

1. During the time the tool is moving in rapid traverse, the AUTO MODE HANDLE OFFSET motion is ineffective. It is effective only during interpolation motion.
2. In the alarm state, the AUTO MODE HANDLE OFFSET motion is ineffective.
3. While an interrupt input (STLK) is on, the AUTO MODE HANDLE OFFSET motion is ineffective.
4. The move distance by the AUTO MODE HANDLE OFFSET function is superimposed on the display of POSITION EXTERNAL and POSITION ABSOLUTE.
5. The AUTO MODE HANDLE OFFSET motion along the respective axes can be made ineffective by parameter settings.

Axis	Parameter		–
	No.	Setting	
X-axis	#6022 D_0 (HOFSX)	1	Effective
		0	Ineffectivc
Z-axis	#6022 D_1 (HOFSZ)	1	Effective
		0	Ineffective

6. When parameter #6022D_7 (HOFSMV) is set to 1, the motion by the AUTO MODE HANDLE OFFSET function is limited to the interpolation motion in automatic operation or temporary stop of interpolation.

6. OPERATION PROCEDURE

6. 1 INSPECTION BEFORE TURNING ON POWER

Make sure that the front and rear doors of the control are firmly closed. The control employs a totally-enclosed, dustproof enclosure to shut out surrounding air. If the door is open, lock it by turning two door locks. In addition, inspect the machine referring to the machine tool builder's manual.

6. 2 TURNING ON POWER

(1) Check to see that the main power is supplied for the control.

(2) Depress the POWER ON pushbutton on the operator's panel, and the control power is supplied and then the cooling fans will start running. Make sure that air blows out from the exhaust ports of the upper side of the control.

(3) Depress the POWER ON pushbutton again to turn on the servo power supply. When the machine is ready to operate, READY lamp lights.

(4) If READY lamp does not light, detect and eliminate the cause according to the alarm code displayed. Refer to 4.3.9 DISPLAYING ALARM CODE.

6. 3 MANUAL OPERATION[1]

When the MODE SELECT switch on the machine control station is set to RAPID, JOG, STEP or HANDLE position, the machine can be operated manually.

Operation in RAPID Mode

(1) Set MODE SELECT switch to RAPID.

(2) Select the speed using RAPID TRAVERSE RATE OVERRIDE switch.

Speed setting range: 100% - 50% - 25% - F0

(3) Push JOG button to select the axis and direction of movement. The machine moves at the specified speed while the JOG button kept pressed.

[1] Manual operation is defined as the operation in RAPID, JOG, STEP, or HANDLE.

Operations in JOG Mode

(1) Set MODE SELECT switch to JOG.

(2) Adjust the feedrate to the desired setting with JOG FEEDRATE switch (Up to 32 steps).

(3) Push JOG button to select the axis and direction of movement. The machine moves at the specified speed while the JOG button depressed.

Operation in STEP Mode

(1) Set MODE SELECT switch to STEP.

(2) Select the move amount per step using MANUAL PULSE MULTIPLY switch.

(Move amount setting range)

Metric: 0.001 - 0.01 - 0.1 - 1.0 - 10.0 - 100.0 mm/step

Inch: 0.0001 - 0.001 - 0.01 - 0.1 - 1.0 - 10.0 in/step

(3) Depress JOG button to select the axis and direction of movement. The machine moves by the move amount per step each time the button is depressed.

6.3 MANUAL OPERATION[1] (Cont'd)

Operation in HANDLE Mode †

The control with HANDLE dial† can permit the operation described below.

(1) Set MODE SELECT switch to HANDLE.

(2) Select the axis with HANDLE AXIS switch.

(3) Select the move amount of the machine corresponding to one scale of HANDLE dial using MANUAL PULSE MULTIPLY switch.

Metric: 0.001 - 0.01 - 0.1
(mm per graduation)

Inch: 0.0001 - 0.001 - 0.01
(inch per graduation)

Note: "X1000" or "X10000" is regarded as "X100."

(4) Rotate HANDLE dial

Turning the dial clockwise:
The machine moves in the positive direction.

Turning the dial counterclockwise:
The machine moves in the negative direction.

6.4 PREPARATION FOR STORED LEADSCREW ERROR COMPENSATION AND STORED STROKE LIMIT†

(1) Return to Reference Point

With an NC equipped with the stored leadscrew error compensation or the stored stroke limit functions, either of the following two reference point return motions must be performed after switching on the power supply and before starting automatic operation.

a. Manual return to reference point (See 5.2.1)

b. Execute G28 U0 W0 ; in the MDI mode.

This procedure is to teach the reference point to the control, since doing so is necessary because both leadscrew error compensation and stored stroke check are performed with reference to the reference point.

Checking Parameter #6006D_1, D_0

When the control is equipped with leadscrew error compensation function or the stored stroke limit function, set this parameter to "1." With the parameter #6006D_1, D_0 set to "1," a return to the reference point is required before starting cycles, alarm codes (001 - 002 "reference point return incomplete") are displayed, if the CYCLE START key is pushed without making a reference point return immediatly after turning on the power supply. Be sure to perform the manual operation for return to reference point.

6.5 PREPARATIONS FOR AUTOMATIC OPERATION

To start to operate the machine in the automatic mode, the machine must be brought to the start point, after the application of the power supply. The panel operation required for this varies with programs as shown below. For details, refer to the operation manual of the machine tool builder.

(1) When G28 (AUTOMATIC REFERENCE POINT RETURN) is used

Where G28 is written in the beginning of the program, move the tool manually to a point a short distance away from the reference point.
If the start point is on the side of the reference point from the traverse speed reducing point, the NC causes overtravel when the CYCLE START key is pushed.

EXAMPLE:

```
EOR ;
N001 G28 U0 W0 ;
N002 G50 X··· Z··· ;
```

(2) When MANUAL RETURN TO REFERENCE POINT function is used

Where G28 is not programmed, and the coordinate set up point is the reference point, bring the tool manually to the reference point before starting the automatic cycle operation. Refer to 5.2.1, "Manual Return to Reference Point."

EXAMPLE:

```
EOR ;
N001 G50 X··· Z··· ;
  .
  .
  .
  .
```

(3) When automatic and manual return to reference point functions are not used

To set up the programmed work coordinate system with workpiece as a basis without using the reference point, proceed as follows:

a. Select the reference tool and set the test workpiece.

b. Position the Z-axis at the workpiece face (reference surface) by manual operation.

c. Reset the current position display of Z-axis. Then Z-axis is determined as Z-axis coordinate point.

d. Position the X-axis at the outer surface of the workpiece by manual operation. Then the control executes cutting outer surface if necessary.

e. Turn on MACHINE LOCK and set the dimensions of the work outer surface as the current position display of the X-axis. Then the center is determined as X-axis coordinate point.

f. Turn off MACHINE LOCK.

g. Move the tool to the setup point for each axis, checking against the current position display.

When the operations mentioned above are proceeded correctly, tool position offset amount for the tool will be zero.

6.6 OPERATION IN TAPE AND MEMORY MODE

(1) Make sure that NC ALARM lamp is not illuminated. If illuminated, detect and eliminate the cause by the indication of alarm code. Refer to 4.3.9, "ALARM CODE DISPLAY."

(2) Check and correct the stored offset values, and then put the machine in the correct start point.

(3) Set the switches on the control station of machine to the proper positions.

- MODE SELECT switch
- SINGLE BLOCK toggle switch
- RAPID TRAVERSE RATE OVERRIDE switch
- MANUAL ABSOLUTE toggle switch
- OPTIONAL BLOCK SKIP toggle switch
- OPTIONAL STOP (M01) toggle switch
- DRY RUN toggle switch
- FEEDRATE OVERRIDE & JOG FEEDRATE switch

(4) Set the punched tape onto the tape reader. In MEM mode, this operation is not required.

(5) Depress RESET key on the control station. Then LSK will be illuminated and the memory will be rewound.

(6) Depress CYCLE START button to give a Cycle Start to the system.

(7) When the Feed Hold is required for the machine during the system operation, depress FEED HOLD button.

(8) If the unexpected event occurs in the system, immediately depress EMERGENCY STOP pushbutton.

6. 7 MANUAL OPERATION INTERRUPTING AUTOMATIC OPERATION

(1) Stop the automatic operation temporarily by depressing FEED HOLD pushbutton or by setting SINGLE BLOCK switch to ON position.

(2) Record the current positions of each axis on a paper using the current position display operation.

(3) Set MODE SELECT switch to manual operation mode (HANDLE, JOG or RAPID), and the machine can be manually operated.

(4) Return the machine manually to the recorded positions.

(5) Set MODE SELECT switch to the interrupted automatic-mode (TAPE, MDI or MEM).

(6) Depress CYCLE START pushbutton, and the machine will resume the automatic operation.

Notes :

1. Where MODE SELECT switch is changed without depressing FEED HOLD pushbutton.
   a. When the automatic-mode (TAPE, MDI or MEM) is changed to the manual-mode (HANDLE, JOG or RAPID), the machine rapidly slows down and stops.
   b. When the automatic-modes are changed the machine is stopped at the block end.
2. Where the machine is restarted by depressing CYCLE START button, the tool path shifted due to manual operation will be changed by ON-OFF operation of MANUAL ABSOLUTE switch. Refer to 6.1.21 MANUAL ABSOLUTE SWITCH. In manual operation mode, when the CYCLE START button is depressed after writing F, M, S, T or B† code by use of the same procedure as that of MDI operation, the command becomes effective and is executed as soon as written. This procedure is used to add new data to an active buffer. However, M00, M01, M02, M30 and M90 to M99 cannot be written.

6. 8 AUTOMATIC OPERATION IN MDI MODE

(1) Set MODE SELECT switch to MDI operation.

(2) Write up to 10 blocks of data by MDI operation, and execute by pressing CYCLE START. Refer to 4.3.3.1.

(3) Depress CYCLE START button, and automatic operation can be executed in MDI mode.

6. 9 MDI OPERATION INTERRUPTING AUTOMATIC OPERATION

To modify the block data after interrupting operation in TAPE or MEM mode, the following operation should be done after interrupting the operation.

(1) Turn on SINGLE BLOCK switch, and the operation is interrupted after the completion of the block being executed. At the same time, the next blocks of data may be read in advance.

(2) Display the data on CRT DISPLAY according to 4.3.3 WRITING IN BLOCKS AND DISPLAYING CONTENTS BY MDI and check it.

(3) Write the data referring to 4.3.3.1 Multi-block Writing and Operation in MDI Mode. Execute the data by depressing the CYCLE START button.

(4) Set back MODE SELECT switch to the interrupted automatic mode (TAPE or MEM).

(5) Return SINGLE BLOCK switch to OFF position.

(6) Depress CYCLE START button, and TAPE or MEM operation can be continued.

Notes :

1. Writing data by MDI cannot be executed in tool radius compensation modes (G41 - G44) because two-three blocks are read ahead.
2. Writing data by MDI cannot be performed in canned cycle modes (G70 - G76). The machine may not operate properly.
3. Excepting in tool radius compensation and canned cycle modes, MDI opeation is possible.

6. 10 PREPARATION FOR TURNING OFF POWER

(1) Make sure that the machine is at standby and CYCLE START lamp is extinguished.

(2) Check to see that NC ALARM is not indicated on CRT. If alarm is displayed, detect the causes of displayed alarm code and eliminate them. Refer to 4.3.9, "ALARM CODE DISPLAY."

(3) Inspect the machine referring to the machine tool builder's manual.

6. 11 TURNING OFF POWER

(1) Depress EMERGENCY STOP pushbutton to turn off the servo power supply.

(2) Depress POWER OFF pushbutton on the operator's panel to turn off the control power supply.

(3) Cut off the main power supply from the control.

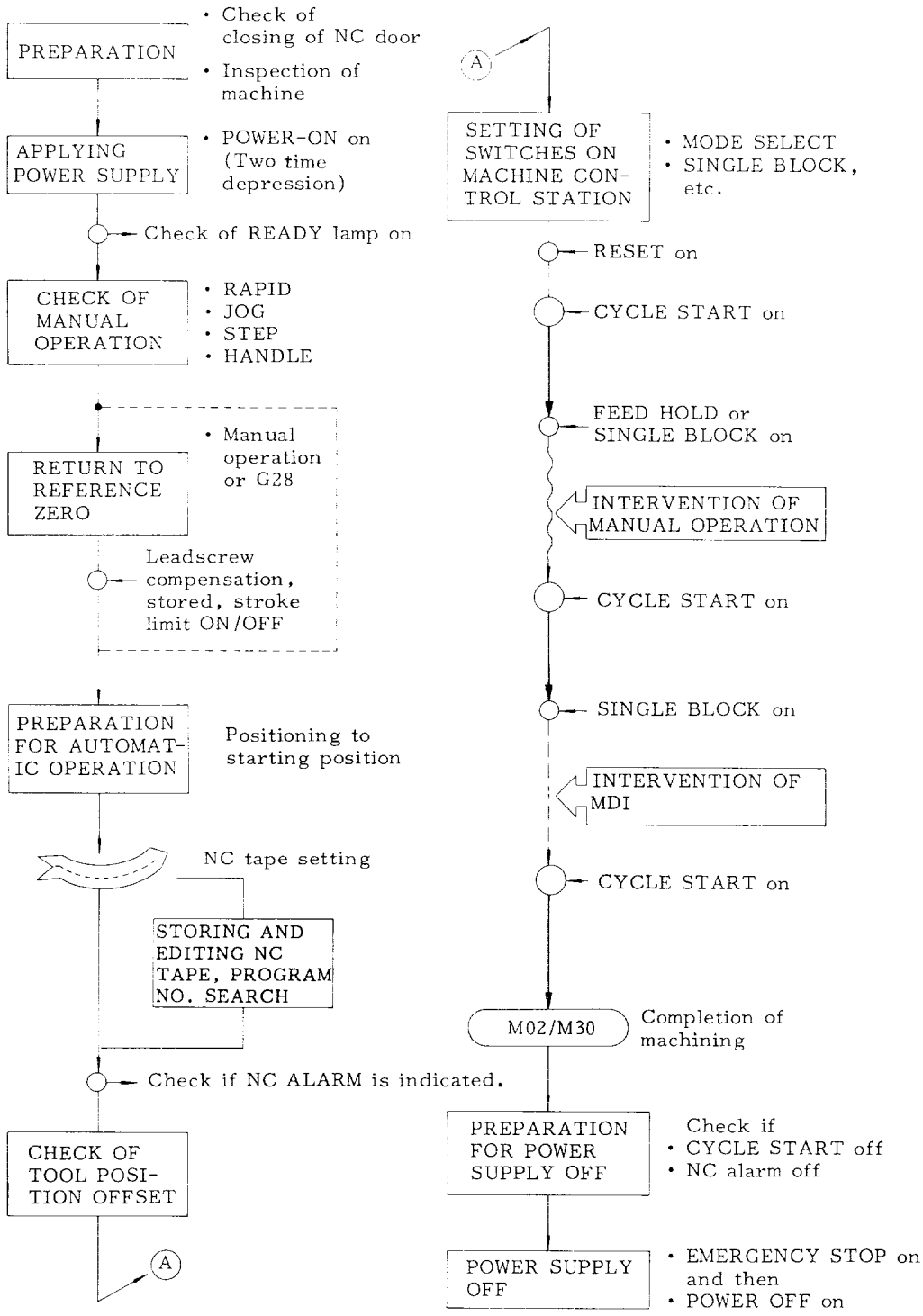


Fig. 6. 1 Operating Procedure

APPENDIX 1 LIST OF SETTING NUMBERS

Parameter numbers are classified in the following four groups.

Parameter numbers	Group
#6005 to #6029	Parameter (Setting by bit)
#6056 to #6134	Parameter (Setting by bite)
#6220 to #6344	Parameter (Setting by word)
#6600 to #6643	Parameter (Setting by double word)
#8000 to #8255	Respective value of Leadscrew Error Compensation (Setting by bite)

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
#6000								

INHEDTT D_7

1: Turns on Edit Lock function.
0: Turns off Edit Lock function.

AFLT D_6

1: Turns on Auxiliary Function Lock.
0: Turns off Auxiliary Function Lock.

ABST D_5

1: Turns on Manual Absolute function.
0: Turns off Manual Absolute function.

DRNT D_4

1: Turns on Dry Run function.
0: Turns off Dry Run function.

BDTT D_3

1: Turns on Block Delete function.
0: Turns off Block Delete function.

DLKT D_2

1: Turns on Display Lock function.
0: Turns off Display Lock function.

MLKT D_1

1: Turns on Machine Lock function.
0: Turns off Machine Lock function.

SBKT D_0

1: Turns on Single Block function.
0: Turns off Single Block function.

Notes:

1. These settings are for setting internal toggle switches.
2. When each switch is provided with machine control station, the logical sum of these settings and toggle switch setting determines function on/off state.

Internal toggle switch	OFF	ON	OFF	ON
Toggle switch on machine	OFF	OFF	ON	ON
Resultant ON/OFF state	OFF	ON	ON	ON

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
#6001								

BUZON D_7

1: Turns on touch buzzer (key switch on operator's panel).
0: Turns off touch buzzer.

PSTMD1 : D6

1: With PST1 mode is designation
0: No PST1 mode is designation

PSTMD2 : D5

1: With PST2 mode designation
0: No PST2 mode is designation

HINTPM: D4

1: Enable handle interpolation function.
0: Disable handle interpolation function.

CMCHK: D3

1: Enable CMOS spare area check function.
0: Disable CMOS spare area check function.

SLT 3: D_2

1: Effective on the third Stored Stroke Limit.
0: Ineffective on the third Stored Stroke Limit.

The value of limit automatically changes by G38 or G39 command in part program.

SLT 2: D_1

1: Effective on the second Stored Stroke Limit.
0: Ineffective on the second Stored Stroke Limit.

The value of limit automatically changes by G36 or G37 command in part program.

INCHMM D_0

1: Selects inch input increment.
0: Selects metric input increment.

#6002	D_7	D_6	D_5	D_4				

ISOEIA D_7

1: Punches out tape code with ISO code.
0: Punches out tape code with EIA code.

TVCHK D_6

1: Executes TV check.
0: Does not execute TV check.

UMO9000E D_5

1: Effective on the edit interlock in O9000's.
0: Ineffective on the edit interlock in O9000's.

UMO8000E D_4

1: Effective on the edit interlock in O8000's
0: Ineffective on the edit interlock in O8000's.

#6003			D_5	D_4			D_1	D_0

D_5, D_4
ODVCE 1 Selects RS232C interface for output port.
ODVCD 0 I/O interface.

D_1, D_0
IDVCE 1 Selects RS232C interface for input
IDVCE 0 port.

Setting Code		Input/Output Port	Parameter No. requiring Baud Rate Setting
0	1	RS232C No.1	#6026 #6028
1	0	RS232C No.1	#6027 #6029

#6004	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

COV161(D_7), COV81(D_6), COV41(D_5), COV21(D_4), COV11(D_3):

Sets the override of cut depth for Stock Removal in Turning (G71) and Stock Removal in Facing (G72) cycles.

COV161	COV81	COV41	COV21	COV11	Cut Depth Override
0	0	0	0	0	100%
0	0	0	0	1	10%
0	0	0	1	0	20%
0	0	0	1	1	30%
0	0	1	0	0	40%
0	0	1	0	1	50%
0	0	1	1	0	60%
0	0	1	1	1	70%
0	1	0	0	0	80%
0	1	0	0	1	90%
0	1	0	1	0	100%
0	1	0	1	1	110%
0	1	1	0	0	120%
0	1	1	0	1	130%
0	1	1	1	0	140%
0	1	1	1	1	150%
1	0	0	0	0	160%
1	0	0	0	1	170%
1	0	0	1	0	180%
1	0	0	1	1	190%
1	0	1	0	0	200%

Note: These settings are effective when parameter #6023 D_2 (COVP) is "0."

UMO8000 D_2

1: Inhibits editing and display and punchout operations of the part program of program No. 8000 to 8999.
0: Permits editing and display and punchout operations.

UMSBK D_1

1: Makes Single Block Stop effective for the programs in user macro when single block input is on.
0: Does not permit Single Block Stop for the user macro blocks commanding operation and control.

SKIPIN D_0

1: Executes the next block when the skip signal is not given before completion of movement of block including Skip Function (G31) or if the touch switch does not trip beyond the limit position in the direction of motion for tool set error compensation (G35).
0: Alarm "087" is displayed.

#6161	TG 1 LF
#6162	TG 2 LF
#6163	TG 3 LF
#6164	TG 4 LF
#6165	TG 5 LF
#6166	TG 6 LF
#6167	TG 7 LF
#6168	TG 8 LF
#6169	TG 9 LF

TG1LF to TG9LF:

Individual life expectancy for tools in groups 1 to 9 is set by part program.

Setting range: 0 - 9999 (Tool life control)
Setting: "1" = 1

#6170	TG 10 LF
#6171	TG 11 LF
#6172	TG 12 LF
#6173	TG 13 LF
#6174	TG 14 LF
#6175	TG 15 LF
#6176	TG 16 LF
#6177	TG 17 LF
#6178	TG 18 LF
#6179	TG 19 LF

TG10LF to TG19LF:

Individual life expectancy for tools in groups 10 to 19 is set by part program.

Setting range: 0 - 9999 (Tool life control)
Setting: "1" = 1 minute

#6181	TG 1 CNT
#6182	TG 2 CNT
⟆	⟆
#6198	TG 18 CNT
#6199	TG 19 CNT

TG1CNT to TG19CNT:

No. of times used and operating times are indicated individually for tools in groups 1 to 19.

Note: Writing is not permitted in this setting.

#6202	G 71 OFL

G71OFL:

Sets retraction value after completion of each cutting cycle in Stock Removal in Turning (G71).

Setting range: 0 - 65536
Setting: Least input increment

#6203	G 72 OFL

G72OFL:

Sets retraction value after completion of each cutting cycle in Stock Removal in Facing (G72).

Setting range: 0 - 65536
Setting: Least input increment

#6204	G 74 OFL

G74OFL:

Sets retraction value (δ) in Peck Drilling in Z-axis (G74).

Setting range: 0 - 65536
Setting: Least input increment

#6205	G 75 OFL

G75OFL:

Sets retraction value (δ) in Grooving in X-axis (G75).

Setting range: 0 - 65536
Setting: Least input increment

#6206	G76OFL

G76OFL:

Sets cut depth (in X-axis) "a" in Automatic Threadcutting (G76).

Setting range: 0 - 65536
Setting: Least input increment

#6207	TINON

When the tape without program no. is stored, program no. is set for the tape.

#6219	SNSW

Setting: 0, 1, or 4 of system No. switch

"0": SYSTEM

For usual operation. Writing parameters is prevented.

"1": PARAMETER

To write parameters. At this position, the Cycle Start is prevented.

"4": TEST (0)

The usual operation is similar to case of "0" SYSTEM. Self-diagnostics of the memory contents and checking of reference zero return point are omitted.

Notes :

1. Setting the values other than described above cannot permit the correct operation.
2. Setting can be effective only when the system number switch provided on the upper part of the unit is set at "0."

#6500	XSL2P
#6501	ZSL2P

XSL2P, ZSL2P:

Sets the boundary area in positive direction of Stored Stroke Limit second prohibit area on X-axis and Z-axis, respectively.

Setting range: 0 to ±99999999
Setting: Least output increment

#6502	XSL2M
#6503	ZSL2M

XSL2M, ZSL2M:
Sets the boundary area in minus direction of Stored Stroke Limit second prohibit area on X-axis and Z-axis, respectively.

Setting range: 0 to ±99999999
Setting: Least output increment

#6504	ZSL3P
#6505	ZSL3P

XSL3P, ZSL3P:

Sets the boundary area in positive direction of Stored Stroke Limit third prohibit area on X-axis and Z-axis, respectively.

Setting range: 0 to ±99999999
Setting: Least output increment

#6506	XSL3M
#6507	ZSL3M

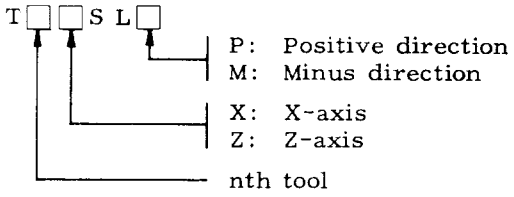
XSL3M, ZSL3M:

Sets the boundary area in minus direction of Stored Stroke Limit third prohibit area on X-axis and Z-axis, respectively.

Setting range: 0 to ±99999999
Setting: Least output increment

#6508	T1XSLP
#6509	T1ZSLP
#6510	T1XSLM
#6511	T1ZSLM
#6512	T2XSLP
#6513	T2ZSLP
#6514	T2XSLM
#6515	T2ZSLM
#6516	T3XSLP

#6517	T3 ZSLP
#6518	T3 XSLM
#6519	T3 ZSLM
#6520	T4 XSLP
#6521	T4 ZSLP
#6522	T4 XSLM
#6523	T4 ZSLM
#6524	T5 XSLP
#6525	T5 ZSLP
#6526	T5 XSLM
#6527	T5 ZSLM
#6528	T6 XSLP
#6259	T6 ZSLP
#6530	T6 XSLM
#6531	T6 ZSLM
#6532	T7 XSLP
#6533	T7 ZSLP
#6534	T7 XSLM
#6535	T7 ZSLM
#6536	T8 XSLP
#6537	T8 ZSLP
#6538	T8 XSLM
#6539	T8 ZSLM
#6540	T9 XSLP
#6541	T9 ZSLP
#6542	T9 XSLM
#6543	T9 ZSLM
#6544	T10 XSLP
#6545	T10 ZSLP
#6546	T10 XSLM
#6547	T10 ZSLM
#6548	T11XSLP
#6549	T11ZSLP
#6550	T11XSLM
#6551	T11ZSLM
#6552	T12 XSLP
#6553	T12 ZSLP
#6554	T12 XSLM
#6555	T12 ZSLM
#6556	T13 XSLP
#6557	T13 ZSLP
#6558	T13 XSLM
#6559	T13 ZSLM
#6560	T14 XSLP
#6561	T14 ZSLP
#6562	T14 XSLM
#6563	T14 ZSLM
#6564	T15 XSLP
#6565	T15 ZSLP
#6566	T15 XSLM
#6567	T15 ZSLM



Sets the distance of Stores Stroke Limit from reference point.

Setting range: 0 to ±99999999
Setting: Least output increment

#6568	XSKIP

Indicates X-axis coordinate value when the skip signal is detected.

#6569	ZSKIP

Indicates Z-axis coordinate value when the skip signal is detected.

#8601	TGPN01
#8602	TGPN02
⟩	⟩
#8649	TGPN49
#8650	TGPN50

TGPN01 to TGPN50:

Part program determines the number of groups including tools (number 01 to 50).

Setting range: 0 to 20
(Tool life control)

#8651	TOFN01
#8652	TOGN02
⟩	⟩
#8699	TOFN49
#8700	TOFN50

TOFN01 to TOFN50:

Part program sets tool number using offset value of offset memory numbers 01 to 50 orderly.

Setting range: 0 to 50
(Tool life control)

#8701	TOFO01
#8702	TOFO02
⟩	⟩
#8749	TOFO49
#8750	TOFO50

TOFO01 to TOFO50:

Part program sets the order of using offset values in offset memories "01" to "50," sequentially.

Setting range: 0 to 5
(Tool life control)

APPENDIX 2 LIST OF PARAMETER NUMBERS

Setting numbers are classified in the following four groups.

Setting numbers	Group
#6000 to #6004	Setting by bit
#6161 to #6219	Setting by word
#6500 to #6569	Setting by double words
#8601 to #8750	Setting of tool life control

#6005	D_7	D_6	D_5	D_4		D_1	D_0

GCDSP D_7

1: Uses special G code I as G code*.
0: Uses standard G code as G code.

RSTG01 D_6

1: Determines G code of 01 group as G01 when resetting.
0: Determines G code of 01 group as G00 when resetting.

POSEXT D_5

1: Presets position external display by setting coordinate system.
0: Does not preset position external display by setting coordinate system.

EXTSET D_4

1: Resets the value at POSITION EXTERNAL display to "0."
0: Does not reset the value at POSITION EXTERNAL display to "0."

PONG04 D_1

1: Sets the G code in the 05 group to G99 when power is applied.
0: Sets the G code in the 05 group to G98 when power is applied.

PONG03 D_0

1: Sets the G code in the 03 group to G91 when power is applied.
0: Sets the G code in the 03 group to G90 when power is applied.

* Where the control is provided with special G code II option, determination of setting is changed as follows.
  1: Uses special G code II.
  0: Uses standard G code.

#6006	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

SDASGN2, SDASGN1: D_7 , D_6

Setting of S4-digit (analog output) output.

SDASGN2	SDASGN1	At M03 Output	At M04 Output
0	0	Plus	Plus
0	1	Minus	Minus
1	0	Plus	Minus
1	1	Minus	Plus

lOIN D_5

1: Sets ten times the least input increment.
0: Sets the least input increment.

SAGRCH D_4

1: Checks to see if the spindle speed match signal (SAGR) is off upon transition from a rapid traverse block to a cutting feed block.
0: Provides no check on the spindle speed match signal (SAGR).

XRAD D_3

1: Radius designation.
0: Diameter designation.

RPDDRN D_2

1: Enables Dry Run in response to the rapid traverse command.
0: Disables dry Run in response to the rapid traverse commanc.

ZZRNILK D_1

1: Causes an alarm ("002") upon Cycle Start when Reference Point Return on Z-axis is not made manually after power is applied.
0: Causes no alarm.

XZRNILK D_0

1: Causes an alarm ("001") upon Cycle Start when Reference Point Return on X-axis is not made manually after power is applied.
0: Causes no alarm.

Note: Set "1" when Stored Lead Screw Error Compensation or Stored Stroke Limit is provided, set ZZRNILK at 1, XZRNILK at 1.

#6007	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

EDTSTLK D7

1: Does not cause an alarm upon Cycle Start without reset operation after part program edit operation.
0: Causes an alarm 005.

STUD D_6

1: Effective on Cycle Start when cycle start signal "1" changes to "0."
0: Effective on Cycle Start when cycle start signal "0" changes to "1."

HSPDM

1: Enable high-speed M function.
0: Disable high-speed M function.

RWDOUT D_4

1: Provides Rewinding Activate Signal when NC program is rewound by RESET & REWIND signal.
0: Provides no Rewinding Activate Signal when NC program is rewound by RESET & REWIND signal.

OUTPUT D_3

1: Sets the least output increment at 0.0001 inch.
0: Sets the least output increment at 0.001 mm.

SCRSOV D_2

1: Makes the Spindle Override 100% during tapping.
0: Does not make the Spindle Override 100% during tapping.

SLT3IO D_1

1: Establishes the prohibited area of the Stored Stroke Limit 3 outside the boundary.
0: Establishes the prohibited area of the Stored Stroke Limit 3 inside the boundary.

SLT2IO D_0

1: Establishes the prohibited area of the Stored Stroke Limit 2 outside the boundary.
0: Establishes the prohibited area of the Stored Stroke Limit 2 inside the boundary.

#6008	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

OFSOCLR: D7

1: Do not clear OFS00 X, Z by O-9999 ORG.
0: Clear OFS00 X, Z by O-9999 ORG.

FSMODCH: D6

1: Edit FS backward upon FS editing.
0: Edit FS each time upon FS editing.

PONM97 D_5

1: M97 command (calculation of intersection) is selected at power-on.
0: M96 command (circular arc) is selected at power-on.

FEEDCUT: D4

1: With feed when upon tape output from the NC.
0: Without feed when upon tape output from the NC.

M30RWD2 D_3

1: Rewinding: operates with M30 if the standard YASNAC tape reader is connected to the second RS232C.
0: No rewinding.

M30RWD1 D_2

1: Rewinding: operates with M30 if the standard YASNAC tape reader is connected to the first RS232C.
0: No rewinding.

CVSAVE D_1

1: Does not clear user macro command variable #100 thru #149 by reset.
0: Clears user macro common variables #100 thru #140 by reset operation.

ZRNOFS D_0

1: Cancels the commanded block when the second reference point by G30 is commanded during Tool Position Offset or Tool Nose Radius Compensation.
0: Cancels the blocks following the commanded block.

#6009	D_7	D_6	D_5	D_4	D_3	D_2	D_1

ZMOVILK D_7

1: After turning on power, if move command except by G28 is executed without returning Z-axis to reference point manually or automatically, alarm "001" will be caused.
0: Does not cause alarm in the same condition shown above.

XMOVILD D6

1: After turning on power, if move command except by G28 is executed without returning X-axis to reference point manually or automatically, alarm "001" will be caused.
0: Does not cause alarm.

OTALILK D_5

1: Does not cause an alarm at overtravel.
0: Causes an alarm at overtravel.

HPMUL D_4

1: Sets the value set by #6223 when MANUAL PULSE MULTIPLY switch is set at x100.
0: Regards multiplication factor as x100 when MANUAL PULSE MULTIPLY switch is set at x100.

ZRNRPD D_3

1: Provides JOG command instead of RAPID command until reference point return for each axis is completed after the power is turned on. In the reference point return mode, usual RAPID operation is executed in both negative and positive directions.
0: Provides usual RAPID TRAVERSE rate.

FKGHOS: D2

1: Enable multiple offset function.
0: Disable multiple offset function.

G71IKUW: D1

1: U, W are the rough cut margin when there is no K in G71 to G73 (multiple repetitive cycle) rough cutting.

0: U, W are not the rough cut margin when there is no K in G71 to G73 (multiple repetitive cycle) rough cutting.

#6010	D_7	D_6	D_5	D_4		D_1	D_0

TLCNT: D7

1: Control the number in all tool groups in tool life management.
0: Does not control the number in all tool groups in tool life management.

TLERR: D6

1: Alarm occurs when the life of a single tool has expired in tool life management.
0: Alarm does not occur when the life of a single tool has expired in tool life management.

AZRNHS D_5

1: Executes the first reference point return (deceleration limit switch) and the subsequent automatic reference point returns in the same way when power is applied.

0: Executes high-speed reference point return (position at reference point).

MZRNHS D_4

1: Executes the first reference point return and the subsequent automatic reference point returns in the same way when power is applied.

0: Executes high-speed reference point return.

ZRNDRZ, ZRNDRX D_1, D_0

Specify the start direction of Backlash Compensation on Z-, and X-axis, respectively.

1: Minus direction
0: Plus direction

#6011		D_6	D_5	D_4	D_3	D_2	D_1	D_0

S4SF: D6

1: Output the SF signal when S4-digit command is given.
0: Does not output the SF signal when S4-digit command is given.

NSRCHG: D5

1: Look-ahead is enabled in system variable #1000 to #3001.
0: Look-ahead is disabled in system variable #1000 to #3001.

SLT3S: D4

1: Enable 3rd stored stroke limit Z-axis barrier setting function.
0: Disable 3rd stored stroke limit Z-axis barrier setting function.

PSOPH: D3

1: Enable tool setter high-speed signal function.
0: Disable tool setter high-speed signal function.

NOTGCAN: D2

1: Enable automatic nose R function and cancel tool coordinate system setting function.
0: Disable automatic nose R function and tool coordinate system setting function cancellation

G71LVL: D1

1: Enable T code command and G41, G42 commands in the multiple repetitive cycle.
0: Disable T code command and G41, G42 commands in the multiple repetitive cycle.

TCOMEM: D0

1: The tool coordinate memory numbers are the lower 2 digits of the T**$$ command.
0: The tool coordinate memory numbers are the higher 2 digits of the T**$$ command.

#6014	D_7	D_6	D_5				

PCMEM: D_7

Stores PC unit malfunction. (Maintenance parameter)

SVMEM D_6

Stores Servo unit mulfunction (Maintenance parameter)

ACGCMEM: D5

1: Stores the operation error of ACGC120.

#6015						D_1	D_0

ATSUPZ, ATSUPX D_1, D_0

Specify whether or not the Automatic Coordinate System Setting is effective on the Z- and X-axis, respectively.

1: Effective
0: Ineffective

Note: The Automatic Coordinate System is established with the following parameters:

Inch system: #6631, #6630
Metric system: #6637, #6636

#6016	D_7				D_2	D_1	

DPRINT: D7

1: Output the leading 0 also when the higher order digits of the variable are 0 in the DPRINT function.

0: Output nothing when the higher order digits of the variable are 0 in the DPRINT function.

NOSEFF: D2

1: Disable interference check.
0: Enable interference check.

MRVDRN: D1

1: Enable switching to dry run during move under the MM/REV mode.
0: Disable switching to dry run during move under the MM/REV mode.

#6017	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

EIA#B7-B0 D_7 - D_0

Specify whether or not a hole is to be made on channels 8-1, respectively, in a code corresponding to symbol "#" (used with user macro) in the EIA code.

1: Hole
0: No hole

Example: EIA#B7-B0 = 01001001

The code with holes on channels 7, 4, and 1 is considered equivalent to symbol "#" in the EIA code. No code for use by the unit can be set.

Note: The specification of EIA#B7-B0 = 00000000 assumes that symbol "#" is not used in the EIA code.

#6018	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

G50WST

1: Does not add workpiece shift value at G50 coordinate system setting or resetting by ORG key.
0: Adds workpiece shift value at G50 coordinate system setting or resetting by ORG key.

WSSGN

1: Reverses U and W input code for work coordinate system shift value.
0: Does not reverse U and W input code for work coordinate system shift value.

OFSCYC

1: Displays cyclically by pressing offset function.
0: Does not display cyclically by pressing offset function.

G32ALM

1: Activates alarm due to too short cutting time of 1 block for continuous threadcutting.
0: Executes without waiting for cycle start if cutting time is too short for continuous threadcutting.

MAXUW

1: Warning if U or W input exceeds parameter #6626
0: Regards U or W input to offset data as usual input.

RPDOV

1: Provides six steps of rapid override (F0, 25, 50, 100%)
0: Provides four steps of rapid override (F0, 25, 50, 100%)

SPDOV

1: Sets spindle speed override range of 10 to 200% (10% increments)
0: Sets spindle speed override range of 50 to 120% (10% override increments).

G150XZ: D0

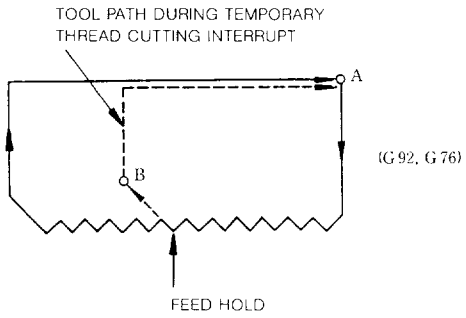
1: X, Z axes simultaneous shift causes alarm in G150, G151 groove width shift.
0: X, Z axes simultaneous shift does not cause alarm in G150, G151 groove width shift.

#6019	D_7		D_5	D_4	D_3	D_2		

G92FHP D7

Specifies the position of temporary stop of threadcutting.

1: Stops at the position B where Threading-up is completed.
0: Returns to start point A and stops after Threading-up is completed.



SCRDRN D5

1: Enables Dry Run at threadcutting.
0: Disables Dry Run at threadcutting.

SKPFED D_4

1: Employs the feedrate set in parameter #6232 (G31F) for the Skip Function command (G31).
0: Employs the F code command as the feedrate for the Skip Function command (G31).

ESPRST D_3

1: Does not turn on RST output with ESP input ON.
0: Turns on RST output with ESP input ON.

G31SKP D_2

1: Inputs the specified value to macro system variables #5001, 5002 during G31 execution.
0: Inputs the current value to macro system variables #5001, 5002 during G31 execution.

#6020	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

OFSDSP D7

1: Displays programmed position in current position display (POSITION ABSOLUTE).
0: Displays programmed position modified with tool position offset in current position display (POSITION ABSOLUTE).

OFSZRO: D6

1: Automatically clear the tool wear offset by external input.
0: Does not automatically clear the tool wear offset by external input.

FOVAB D_5

1: Effective with feedrate override signal "0."
0: Effective with feedrate override signal "1."

SSTPAB D4

1: Analog output zero with spindle S command zero input signal SSTP "0."
0: Analog output zero with spindle S command zero input signal SSTP "1."

AUTSET: D3

1: Automatically set the tool coordinate memory by external input.
0: Does not automatically set the tool coordinate memory by external input.

PSTSGN D_2

Shown in the calculation formula of storing data during MDI of measured work point into tool offset memories 01 to 50.

1: (Data of tool coordinate memory) = (Current value temporarily stored in the register) − (Written measurement value)

0: (Data of tool coordinate memory) = (Written measurement value) − (Current value temporarily stored in the register)

OFSG96 D_1

1: Specifies the surface speed calculated by the X-axis coordinate value modified by tool position offset value in Constant Surface Speed Control.
0: Specifies the surface speed calculated by the programmed X-axis coordinate value in Constant Surface Speed Control.

POSG96 D_0

1: Surface Speed Control functions on the block including Rapid Traverse (G00).
0: Surface Speed Control functions on the block including Rapid Traverse (G00), if programmed before the Cutting Feed block.

#6021	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

UMO9000 D_7

1: Inhibits editing and display and punchout operations of the part program of program No. 9000 to 9999.
0: Permits editing and display and punchout operations.

MERSIN D_6

1: Replaces the stored program with a new one when part program is already stored.
0: Displays ALREADY ALARM.

RSONOF D_5

1: Sets on and off RS (RS232C signal) by "%" character.
0: Keeps RS signal on until reading-in is finished.

CHKDR D_4

1: Recognizes DR.
0: Does not recognize DR.

O - 9999O D_3

1: Punches O0 when tape is punched with O, -, 9, 9, 9, 9 keyed in and OUT key depressed.
0: Does not punch O0 when tape is punched with O, -, 9, 9, 9, and 9 keyed in and OUT key depressed.

PONON D_2

1: Does not clear program No. on power application.
(Program number is stored at power supply shut off.)
0: Clears program No. on power application.

PRGNO D_1

1: Employs the value following address O or N as the program number (specifiable in one block).
0: Employs the value following address O as the program number.

M02M99 D_0

1: Considers M02, M30 and M99 as the program end when part program is stored into memory. At punchout with O-9999 depressed, add EOB in front of O number.
0: Does not consider M02, M30 and M99 as the program end when part program is stored into memory. At punchout with O-9999 depressed, do not add EOB in front of O number.

#6022	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

HOFSMV D7

Enables the movement of automatic mode handle offset during cutting feed by interpolation. Enables the movement of automatic mode handle offset except during execution of rapid traverse.

TLCC D_6

1: Effective on the next T code when offset amount is changed.
0: Effective on the next block when offset amount is changed.

TRDFH D_5

1: Executes the block next to the block specifying thread, and stops at single block operation or feedhold during thread cutting.
0: Stops on completion of the block specifying thread at single block operation or feedhold during thread cutting.

MABIN D_4

1: Ignores manual absolute function for incremental command by U and W.
0: Does not ignore manual absolute function for incremental command by U and W.

ISOPO D_3

1: Does not output parity bit (8th bit) when outputting ISO codes from NC by operating OUT key (in the EDIT mode).
0: Outputs parity bit.

ISOPI D_2

1: Ignores parity bit (8th bit) when outputting ISO codes by operating IN key (in the EDIT mode) and when reading-in ISO tape data in the TAPE mode.
0: Performs parity check.

HOFSZ, HOFSX D_1, D_0

Specifies whether automatic mode handle offset movement is effective or ineffective.

1: Effective automatic mode handle offset movement.
0: Ineffective automatic mode handle offset movement.

#6023	D_7	D_6	D_5		D_3	D_2	D_1	D_0

PERIAB D_7

1: Incremental setting of offset value for Stored Leadscrew Error Compensation.
0: Absolute setting of offset value for Stored Leadscrew Error Compensation.

PERST D_6

1: Regards "%" code as M30, if "%" is commanded before M02 or M30 in TAPE or MEM mode operaiton.
0: Ignores "%" code if commanded before M02 or M30 in TAPE or MEM mode operation.

MCHMST D_5

1: Lights feedhold lamp and stores M, S, and T commands when manual operation mode is selected during automatic operation.
0: Does not light feedhold lamp and M, S, and T commands are forced to reset when manual operation mode is selected during automatic operation.

ONOCHG D_3

1: Changes No. 0 by pressing 0, program number, and ALT keys.
0: Does not change No. 0 by pressing 0, program number, and ALT keys.

COVP D_2

1: Sets cut depth value override with cut depth override input in Stroke Removal in Turning (G71) and stock removal in facing (G72).
0: Sets cut depth value override with setting #6004.

CLNO D_1

1: Outputs "0" by inputting ":" and outputs ":" by inputting "0," in ISO code.
0: Does not perform the conversion above.

HSRWD D_0

1: Automatically starts at high speed in high-speed rewinding.
0: Does not start automatically at high speed in high-speed rewinding.

Input for RS232C No.1

#6026			D_5	D_4	D_3	D_2	D_1	D_0

SIF1CI D_5

Determines whether the input control code for RS232C interface is given or not.

1: Does not send control code.
0: Sends control code.

SIF1SI D_4

Determines the input stop bit for RS232C interface No. 1 as two bits or one bit.

1: Determines stop bit as two bits.
0: Determines stop bit as one bit.

SIF1BID - SIF1BIA D_3 - D_0

Sets input baud rate for RS232C interface No. 1

Baud Rate	SIF1BID	SIF1BIC	SIF1BIB	SIF1BIA
50	0	0	0	0
100	0	0	0	1
110	0	0	1	0
150	0	0	1	1
200	0	1	0	0
300	0	1	0	1
600	0	1	1	0
1200	0	1	1	1
2400	1	0	0	0
4800	1	0	0	1
9600	1	0	1	0

Output for RS232C No. 2

#6027			D_5	D_4	D_3	D_2	D_1	D_0

SIF2CI D_5

Determines whether output control code for RS232C No. 2 interface is sent or not.

1: Does not send control code.
0: Sends control code.

SIF1SO D4

Determines output stop bit for RS232C interface No. 2 as two bits or one bit.

1: Determines stop bit as two bits.
0: Determines stop bit as one bit.

SIF2BID-SIF2B1A D_3 - D_0

Sets output baud rate for RS232C interface No. 2.

Baud Rate	SIF 2 BID	SIF 2 BIC	SIF 2 BIB	SIF 2 BIA
50	0	0	0	0
100	0	0	0	1
110	0	0	1	0
150	0	0	1	1
200	0	1	0	0
300	0	1	0	1
600	0	1	1	0
1200	0	1	1	1
2400	1	0	0	0
4800	1	0	0	1
9600	1	0	1	0

Output for RS232C No. 1

#6028			D_5	D_4	D_3	D_2	D_1	D_0

SIF1CO D_5

Determines whether output control code for current loop and RS232C interface is sent or not.

1: Does not send control code.
0: Sends control code.

SIF1SO D_4

Determines output stop bit for current loop and RS232C interface as two bits or one bit.

1: Determines stop bit as two bits.
0: Determines stop bit as one bit.

SIF1BOD-SIF1BOA D_3 - D_0

Sets output baud rate for current loop and RS232C interface.

Baud Rate	SIF 1 BOD	SIF 1 BOC	SIF 1 BOB	SIF 1 BOA
50	0	0	0	0
100	0	0	0	1
110	0	0	1	0
150	0	0	1	1
200	0	1	0	0
300	0	1	0	1
600	0	1	1	0
1200	0	1	1	1
2400	1	0	0	0
4800	1	0	0	1
9600	1	0	1	0

Output for RS232C No. 2

#6029			D_5	D_4	D_3	D_2	D_1	D_0

SIF2CO D_5

Determines whether output control code for RS232C interface No. 2 is sent or not.

1: Does not send control code.
0: Sends control code.

SIF2SO D_4

Determines output stop bit for RS232C interface No. 2 as two bits or one bit.

1: Determines stop bit as two bits.
0: Determines stop bit as one bit.

SIF2BOD-SIF2BOA D_3 - D_0

Sets output baud rate for RS232C interface.

Baud Rate	SIF 2 BOD	SIF 2 BOC	SIF 2 BOB	SIF 2 BOA
50	0	0	0	0
100	0	0	0	1
110	0	0	1	0
150	0	0	1	1
200	0	1	0	0
300	0	1	0	1
600	0	1	1	0
1200	0	1	1	1
2400	1	0	0	0
4800	1	0	0	1
9600	1	0	1	0

#6056	XPSET
#6057	ZPSET
#6058	SPSET

XPSET, ZPSET, SPSET:

Sets position error range for X-, Z-axis and spindle (at indexing).

Setting range: 0 - 255
Setting: Least output increment

#6062	BSX
#6063	BSZ
#6064	BSS

BSX, BSZ, BSS:

Sets D/A conversion bit for X-, Z-axis, and spindle (at indexing).

#6062 #6063 #6064	Bit	Max No. of Error Pulses
4	12 bit	2047
5	13 bit	4095
6	14 bit	8191
7	15 bit	16383
8	16 bit	32765

$$\text{Max No. of Error Pulses} = \frac{\text{Max Feedrate (pulse/s)}}{\text{Position Loop Gain (1/s)}}$$

Setting range: 0 - 255
Setting: Least output increment

#6068	XPERML
#6069	ZPERML

XPERML, ZPERML:

Sets leadscrew error compensation multiplication factor for X- and Z-axis.
Outputs the result of the preset compensation value multiplied by the multiplication factor as the error compensation value.

Setting range: 0 - 255
(Setting 0 will not execute compensation.)

#6074	XSVER
#6075	ZSVER
#6076	SSVER

XSVER, ZSVER, SSVER:

Sets servo error limit for X-, Z-axis, and spindle. Position deviation exceeding the preset value activates an alarm "34Δ."

Setting range: 0 - 255
Standard setting: 16
Setting: 1/16 x (D/A saturation value)

#6080	CUPRD

Rapid threading pull-out width during thread-cutting

Setting range: 0 - 255
Setting: 0.1 lead

#6085	SIDSER

Spindle stop check timer at spindle index start.

Setting range: 0 - 225
Setting: 1 = 8 ms

#6108	UMEIA [
#6109	UMEIA]
#6110	UMEIA *
#6111	UMEIA =
#6112	UMEIA (
#6113	UMEIA)

UMEIAs:

Specify the punching pattern in EIA for special characters employed in user macro; [,] , *, =, (,), used in turn, beginning with #6108.

Setting range: 0 - 255
Setting: Sets the punching pattern using the decimal value converted from the binary value which defines the pattern.

Note: When "0" is set for each character, punching patternwill be as listed below.

Special Character	8	7	6	5	4	○	3	2	1
[				○	○	○	○		
]		○			○	○	○		
*					○	○	○	○	
=			○	○	○	○		○	○
(				○	○	○		○	
)		○			○	○		○	

#6114	NBUFM1
#6115	NBUFM2
#6116	NBUFM3
#6117	NBUFM4
#6118	NBUFM5
#6119	NBUFM6

NBUFM1, 2, 3, 4, 5, 6:

Sets up to 6 M codes for stopping advance reading function (buffering).

Setting range: 0 - 255

#6120	UMG1
#6121	UMG2
#6122	UMG3
#6123	UMG4
#6124	UMG5
#6125	UMG6
#6126	UMG7
#6127	UMG8
#6128	UMG9
#6129	UMG10

UMG1 - 10:

Sets G codes for calling user macro of program No. O9001 to O9004.

Setting range: 0 - 255

#6130	UMM1
#6131	UMM2
#6132	UMM3
#6133	UMM4

UMM1, UMM2, UMM3, UMM4:

Sets M codes for calling user macro of program No. O9001 to O9004.

Setting range: 0 - 255

#6134	UMT

UMT

1: Regards T-code command as macro call command calling the macro of program No. O9000.
0: Regards T-code command as basic T-code.

Note: This selection is effective only for the user macro option.

#6135	MD0 ~ MD7

MD0 to MD7 High-speed M function decode output holding is set.

Setting range: 0 to 255

#6208	G74OF1

G74OF1:

Set the G74 groove canned cycle retract amount.

Setting range: 0 to 65535

Unit: "1" = 1 input unit

#6209	G75OF1

G75OF1:

Set the G75 groove canned cycle retract amount.

Setting range: 0 to 65535

Unit: "1" = 1 input unit

#6210	G74CLR

G74CLR:

Set the G74 groove canned cycle clearance amount.

Setting range: 0 to 65535

Unit: "1" = 1 input unit

#6211	G75CLR

G75CLR:

Set the G75 groove canned cycle clearance amount.

Setting range: 0 to 65535

Unit: "1" = 1 input unit

#6212	G745FED

G745FED:

Groove canned cycle groove approach FEED override.

Setting range: 0 to 127

Unit: "1" = 100% (is also 100% when it is 0)

#6213	G71OVR

G71OVR:

G71/G72 Infeed reduction override.

Setting range: 0 to 99

Unit: "1" = 1%

#6214	G74DWL

G74DWL:

G74/G75 bottom of hole dwell time.

Setting range: 0 to 65535

Unit: "1" = 1 ms

#6220	MSTF

MSTF:

Sets the interval from the time M, S, and T codes are transmitted until the time MF, SF, and TF are transmitted.

Setting range: 0 - 65536 msec

#6222	HPMAX

HPMAX:

Specifies the maximum handle feedrate, which is commom to the all axes.
This parameter setting is in multiples of 8.

Setting: "1" = 125 pulses/sec

#6223	HPMUL

HPMUL:

Sets the value when MANUAL PULSE MULTIPLY switch is set at x100.

Setting: "1" = x 1 time

#6224	SAGRT

SAGRT:

Specifies the delay time for checking the spindle speed reaching signal (SAGR).

Setting range: 0 - 65536 msec

#6225	KPX
#6226	KPZ

KPX, KPZ:

Set the P-loop gain of command units for X- and Z-axis respectively.

Setting: Standard set: 1024

#6227	KPS

KPS:

Set the P-loop gain of spindle at indexing. The relations between KPS and BSS (#6064) as follows:

Max. Set of #6227	#6064	No. of Bit	No. of Max. offset pulse
4096	4	12	2047
2048	5	13	4095
1024	6	14	8191
512	7	15	16383
256	8	16	32765

#6228	G98MAX

G98MAX :

Specifies the maximum feedrate at G98 command (feed per minute) common to all axes.

Setting range:

Setting: "1" = 1000 pulses/min

#6230	NEGNR

NEGNR:

When a circular path is drawn in Tool Radius Compensation outside a corner approaching 180°, the movement follows describing a very small circular arc. This parameter is used to set the critical arc value, if this arc movement is considered to affect the workpiece surface machining.

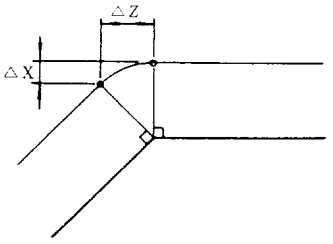
Setting range: 0 - 65536
Setting: Least input increment

The corner arc setting is ignored when:

$\Delta X \leqq$ NEGNR

$\Delta Y \leqq$ NEGNR

Standard setting = 5



#6231	ROVFO

ROVFO :

Specifies the FO speed for Rapid Traverse Override.

Setting range:
Setting: "1" = 125 pulses/sec

#6232	G31F

G31F:

Specifies the feedrate in the skip function (G31).

Setting range:

Setting: "1" = 1000 pulses/min

This setting is effective when parameter #6019D_4 (SKPFED) = 1.

#6233	JOG0
⋮	⋮
#6264	JOG31

JOG0~JOG31:

Specify the feedrates for the respective positions on the jog feedrate select switch.

Setting range:

Setting: "1" = 0.5 mm/min (metric output)
"1" = 0.05 in./min (inch output)

Switch Position	Feedrate Override %	Parameter		Continuous Manual Feedrate
		Number	Setting	mm/min
0	0	#6233	0	0
1	10	#6234	1	1
2	20	#6235	2	2
3	30	#6236	4	4
4	40	#6237	6	6
5	50	#6238	8	8
6	60	#6239	10	10
7	70	#6240	12	12
8	80	#6241	15	15
9	90	#6242	20	20
10	100	#6243	25	25
11	110	#6244	30	30
12	120	#6245	40	40
13	130	#6246	50	50
14	140	#6247	60	60
15	150	#6248	80	80
16	160	#6249	100	100
17	170	#6250	120	120
18	180	#6251	150	150
19	190	#6252	200	200
20	200	#6253	250	250
21	0	#6254	300	300
22	0	#6255	400	400
23	0	#6256	500	500
24	0	#6257	600	600
25	0	#6258	800	800
26	0	#6259	1000	1000
27	0	#6260	1200	1200
28	0	#6261	1500	1500
29	0	#6262	2000	2000
30	0	#6263	2500	2500
31	0	#6264	3000	3000

#6266	MACGR1
#6267	MACGR2
#6268	MACGR3
#6269	MACGR4

MACGR1-MACGR4:

Sets spindle speed upper limit for gear 1, 2, 3, and 4 orderly.

Setting range: 0 - 6000 (r/min)

When the setting is at 0, the speed is not clamped.

#6270	GRSREV

GRSREV:

Sets the speed command output to spindle motor when gear shift input (GRS) is given.

Setting value: $\frac{\text{Gear shift spindle motor speed}}{\text{Spindle motor max speed (Command = 10 V)}}$ x 2047 --- 12-bit output

$\frac{\text{Gear shift spindle motor speed}}{\text{Spindle motor max speed (Command = 10 V)}}$ x 32512 --- Analog output

Setting range: 0 - 6000

#6271	GR1REV
#6272	GR2REV
#6273	GR3REV
#6274	GR4REV

GR1REV-GR4REV:

Specify the maximum speed of the spindle, respectively, for gears 1, 2, 3 and 4 each selected by an input signal. Set the spindle speed applicable when the speed command voltage is 10 V.

Setting range: 0 - 6000 (r/min)

#6275	GSCREV

GSCREV:

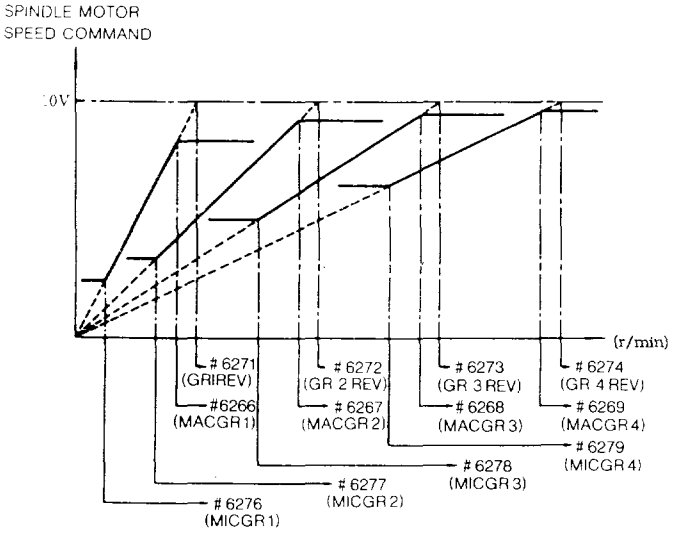
Specifies the spindle motor speed in effect when a spindle operation (GSC) input is entered.

Setting range: 0 - 6000 (r/min)

#6276	MICGR1
#6277	MICGR2
#6278	MICGR3
#6279	MICGR4

Specify the minimum speed of the spindle, respectively for gears 1, 2, 3 and 4 each selected by an input signal.

Setting range: 0 - 6000 (r/min)



#6280	RPDX
#6281	RPDZ

RPDX, RPDZ:

Specify the rapid traverse rate for X- and Z-axis, respectively.

Setting range: 0 - 3200
Setting: "1" = 125 pulses/sec

#6282	XBLP
#6283	ZBLP

XBLP, ZBLP:

Sets amount of backlash for X-, and Z-axis.

Setting range: -8192 - 8191 P

P: Least output increment

Note: When negative value is set, 65536 + "Setting value" is indicated.

#6286	ACCX1
#6287	ACCZ1

ACCX1, ACCZ1:

Set the time constant for Linear Accel/Decel for X- and Z-axis, respectively.

Setting range:

Setting: "1" = $125/8 \times 10^3$ P/sec^2
(P: least output increment)

#6304	XREFP
#6305	ZREFP

XREFP, ZREFP:

Sets the traverse distance for Reference Point Return, respectively, on the X- and Z-axis.

Setting range: 0 - 32767

Setting: "1" = 1 pulse

#6306	SCRXACC
#6307	SCRZACC

SCRXACG, SCRZACC:

Sets accel/decel time constant at threadcutting on X-, and Z-axis.

Setting: 131,070/time constant (ms)

Always read the notes for #6314, #6315.

#6308	SCRXBAS
#6309	SCRZBAS

SCRXBAS, SCRZBAS:

Sets accel/decel bias at threadcutting on X-, and Z- axis.

Setting: 1 = 500 pulses/s

Always read the notes for #6314, #6315.

#6310	XREFV1
#6311	ZREFV1

XREFV1, ZREFV1:

Specify the approach speed 1 for Reference Point Return, respectively, on the X- and Z-axes.

Setting range: 0 - 200

Setting: "1" = 125 pulses/sec

#6312	CUTXACC
#6313	CUTZACC

CUTXACC, CUTZACC:

Sets accel/decel time constant at normal threadcutting on X-, and Z-axis.

Setting: 131,070/time constant ms

Always read the notes for #6314, #6315.

#6314	CUTXBAS
#6315	CUTZBAS

CUTXBAS, CUTZBAS:

Sets accel/decel bias at usual threadcutting on X-, and Z-axis.

Setting: 1 = 500 p/s

Notes for Setting Accel/Decel Time Constant and Bias at Normal Threadcutting (#6306 - #6309, #6312 - #6315)

1. Set the same values for X- and Z-axis as a rule to ensure motion path accuracy.

2. When accel/decel time constant or bias is set under the different conditions between threadcutting and normal threadcutting, direct changing these two types cutting may influence speed factor at selected point. Be sure to program dwell or positioning code between these two types.

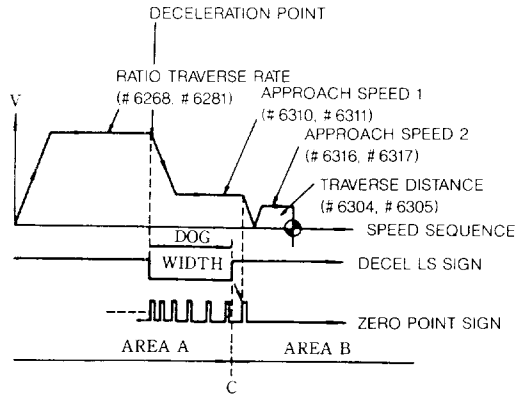
#6316	XREFV2
#6317	ZREFV2

XREFV2, ZREFV2:

Specify the approach speed 2 for Reference Point Return, respectively, on the X- and Z-axes.

Setting range: 0 - 200

Setting: "1" = 125 pulses/sec



· Reference point return direction: #6010 (ZRNDRX, ZRNDRZ)

#6318	RPDX2
#6319	RPDZ2

RPDX2, RPDZ2:

Set change speed for linear accel/decel of X- and Z-axis, respectively.

Setting: "1" = 125 pulses/s

#6320	ACCX2
#6321	ACCZ2

Set 2nd step constant for linear accel/decel of X- and Z-axis, respectively.

Setting: "1" = $125/8 \times 10^3$ p/s^2

Note: Parameter setting for linear 2nd step accel/decel must be set under the following conditions:

$$\begin{bmatrix}\text{2nd step accel/}\\ \text{decel change speed}\\ \text{\#6318/\#6319}\end{bmatrix} = \begin{bmatrix}\text{Rapid accel/}\\ \text{decel constant}\\ \text{\#6286/\#6287}\end{bmatrix} \times N$$

If the above conditions are not satisfied, the positioning may not perform correctly.

#6322	XPERED
#6323	ZPERED

XPERED, ZPERED:

Specify the number of the end point for Leadscrew Error Compensation, respectively, on the X- and Z-axes.

Setting range: 0 - 255

#6328	XPERST
#6329	ZPERST

XPERED, ZPERED:

Specify the number of the start point for Leadscrew Error Compensation, respectively, on X- and Z-axes.

Setting range: 0 - 255

#6334	XPEROR
#6335	ZPEROR

XPEROR, ZPEROR:

Specify the reference point for Leadscrew Error compensation, respectively, on the X- and Z-axes.

Setting range: 0 - 255

#6342	SIFREF

SIDREF:

Sets the reference point for spindle indexing.

Setting range: 0 - 4095

Setting: "1" = 1 pulse (= 360/4096 deg.)

#6343	SIDRV1

SIDRV1:

Sets the spindle speed for spindle indexing.

Setting range: 0 - 32767

Setting: 1 = 500 pulse per second

#6344	SIDCRP

SIDCRP:

Sets the spindle indexing creep speed.

Setting range: 0 - 32767

Setting: "1" = 500 pulse per second

#6347	BSBON X (X-AXIS)
#6348	BSBON Y (Y-AXIS)

Setting time from ESP to SVOFF in X-AXIS and Z-AXIS in order:
"1" = 8 ms (unit)
0 - 32767 (setting unit)

#6349	HINTPP

HINTPP:

Set the handle interpolation reference pulse count.

Setting range: 0 - 65535

Unit: "1" = 1 p/s

#6600	XSL1P
#6601	ZSL1P

XSL1P, ZSL1P:

Specify the plus direction boundary value for Stored Stroke Limit 1, respectively, on the X-, and Z-axes.

Setting range: 0 - 99999999

Setting: "1" = 1 pulse

#6606	XSL1M
#6607	ZSL1M

XSL1M, ZSL1M:

Specify the minus direction boundary value for Stored Stroke Limit 1, respectively, on the X-, Z-axes.

Setting range: 0 - 99999999

Setting: "1" = 1 pulse

#6612	XZP2L
#6613	ZZP2L

XZP2L, ZZP2L:

Specify the distance between the first and the second reference point, respectively, on the X-, Z-axes.

Setting range: -99999999 - 99999999

Setting: "1" = 1 pulse

#6616	WKSHTX

WKSHTX:

Set the X-axis workpiece shift offset.

Setting range: -99999999 - 99999999

Unit: "1" = Least command increment

#6617	WKSHTW

WKSHTW:

Set the workpiece shift amount measurement tool offset amount.

Setting range: -99999999 - 99999999

Unit: "1" = Least command increment

#6618	DFSX1

DFSX1:

Workpiece measurement value direct input parameter (1)
Set the distance from the reference tool nose position to the X1 contact surface.

Setting range: -99999999 - 99999999

Unit: "1" = Least command increment

#6619	DFSX2

DFSX2:

Workpiece measurement value direct input parameter (2)
Set the distance from the reference tool nose position to the X2 contact surface.

Setting range: -99999999 - 99999999

Unit: "1" = Least command increment

#6620	DFSZ1

DFSZ1:

Workpiece measurement value direct input parameter (3)
Set the distance from the reference tool nose position to the Z1 contact surface.

Setting range: -99999999 - 99999999

Unit: "1" = Least command increment

#6621	DFSZ2

DFSZ2:

Workpiece measurement value direct input parameter (4)
Set the distance from the reference tool nose position to the Z2 contact surface.

Setting range: -99999999 - 99999999

Unit: "1" = Least command increment

#6622	DFSXC

DFSXC:

Workpiece measurement value direct input parameter (5)
Set the X dimension of the gauge workpiece.

Setting range: -99999999 - 99999999

Unit: "1" = Least command increment

#6623	DFSZC

DFSZC:

Workpiece measurement value direct input parameter (6)
Set the Z dimension of the gauge workpiece.

Setting range: -99999999 - 99999999

Unit: "1" = Least command increment

#6626	MAXDUW

MAXDUW:

Sets upper limit of U and W for offset data. The input of the value exceeding the limit will cause an alarm.

Note: Parameter #6018 D_3 is set to "1," the parameter setting is effective.

#6630	XSETI
#6631	ZSETI

XSETI, ZSETI:

Specify the value for Automatic Coordinate System Setting at the time of inch input, respectively, on the X-, and Z-axes. A desired value should be set in inches for the distance between the first reference point and the reference point of the coordinate system to be established.

Setting range: -99999999 - 99999999

Setting: "1" = 0.0001 in.

#6636	XSETM
#6637	ZSETM

XSETM, ZSETM:

Specify the value for Automatic Coordinate System Setting at the time of metric input, respectively, on the X-, and Z-axes. A desired value should be set in millimeters for the distance between the first reference point and the reference point of the coordinate system to be established.

Setting range: -99999999 - 99999999

Setting: "1" = 0.001 mm

#6642	XPEINT
#6643	ZPEINT

XPEINT, ZPEINT:

Specify the compensation inverval in Leadscrew Error Compensation, respectively, on the X- and Z-axes.

Setting range: -99999999 - 99999999

Setting: "1" = 1 pulse

#6644	HSDM1

HSDM1:

Set the high-speed M function decode output set 1.

#6645	HSDM2

HSDM2:

Set the high-speed M function decode output set 2.

#6646	HSDM3

HSDM3:

Set the high-speed M function decode output reset 1.

#6647	HSDM4

HSDM4:

Set the high-speed M function decode output reset 2.

#6648	DIVERR

DIVERR:

Set the instruction point value code segment value for division error.

#8000	PEMN0
⋮	⋮
#8255	PEMN255

PEMN0-PEMN255:

Specify the respective values of Leadscrew Error Compensation.

Setting range: 0 - ±15 (Incremental designation)
0 - ±128 (Absolute designation)

Setting: "1" = Output increment

Incremental/absolute designation is selected by parameter #6023D_7 (PERIAB).

Axis for compensation is specified by parameters #6322, 6323, 6328, and 6329.

APPENDIX 3 STORED LEADSCREW ERROR COMPENSATION

This function automatically compensate for leadscrew error on each axis according to the compensation data set by parameter and is effective after completion of reference point return. The compensation data are made on the distances between the reference point on each axis and specified points.

Compensation axes: X, Z axes

No. of correction points: 256 Max.

Compensation base point: Reference point

Compensation interval: 6000 Pulses or more

Data setting system: Absolute/incremental (Set by Parameter #6023D$_7$ PERIAB)

Compensation value:

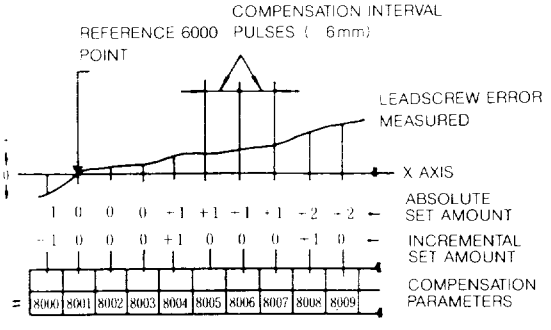
Minimum compensation unit: 1 pulse (least output increment)

Compensation multiplication factor: 3X max.

One-time-compensation value: 15 pulses max. (Compensation multiplication)

Notes:

1. Regardless of absolute/incremental setting, the difference between neighboring compensation values should be (15 pulses x compensation multiplication) and below.
2. Maximum set value in case of absolute setting is ±127 pulses. Compensation multiplication is taken on this value.
3. No. of correction points on each axis can be arbitrary as far as the total compensation points are within 256.



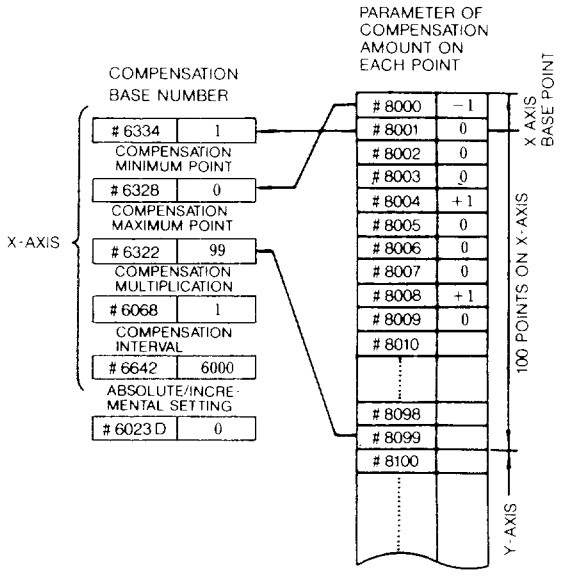


Table 3.1

	Axis	Parameter #	Functions
Compensation Interval	X	#6642 (XPEINT)	6000 OR MORE "1" = 1 Pulse
	Z	#6642 (ZPEINT)	
Absolute/Incremental Setting Switchable		#6023 D_7 (PERIAB)	"0" = Incremental setting "1" = Absolute setting
Compenastion Reference No.	X	#6334 (XPEROR)	Value of parameter # of compensation on each point minus 8000 will be written.
	Z	#6335 (ZPEROR)	
Compensation Max Point	X	#6322 (XPERED)	
	Z	#6323 (ZPERED)	
Compensation Min Point	X	#6328 (XPERST)	
	Z	#6329 (ZPERST)	
Compensation Value on Each Point	X	#8000 – #8255	0 to ±7 (Incremental setting) 0 to ±127 (Absolute setting) "1" = 1 pulse
	Z		
Compensation Multiplication Factor	X	#6068 (XPERML)	0 to 3 "1" = 1X
	Z	#6069 (ZPERML)	

APPENDIX 4 LIST OF STANDARD INPUT/OUTPUT SIGNALS

Table 4-1 shows the list of diagnostic numbers and signal names of standard input/output signals and monitor signals.

Diagnostic Number	Display
#1000–#1096	Input signals from machine
#1100–#1157	Output signals to machine
#1200–#1223	Output signals to machine interface (PC)
#1300–#1331	Input signals from machine interface (PC)
#1280–#1295	Monitor signals

Notes:

1. Monitor signals are used to check the internal condition of the control.
2. The functions of signals #1000 - #1096, #1100 - #1157

Refer to machine tool builder's manual.

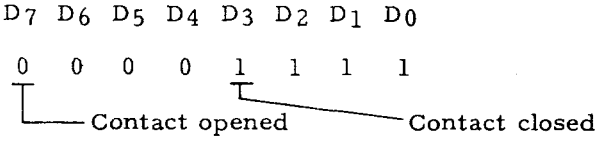


Fig. 4.1 Status Display of Input/Output Signals

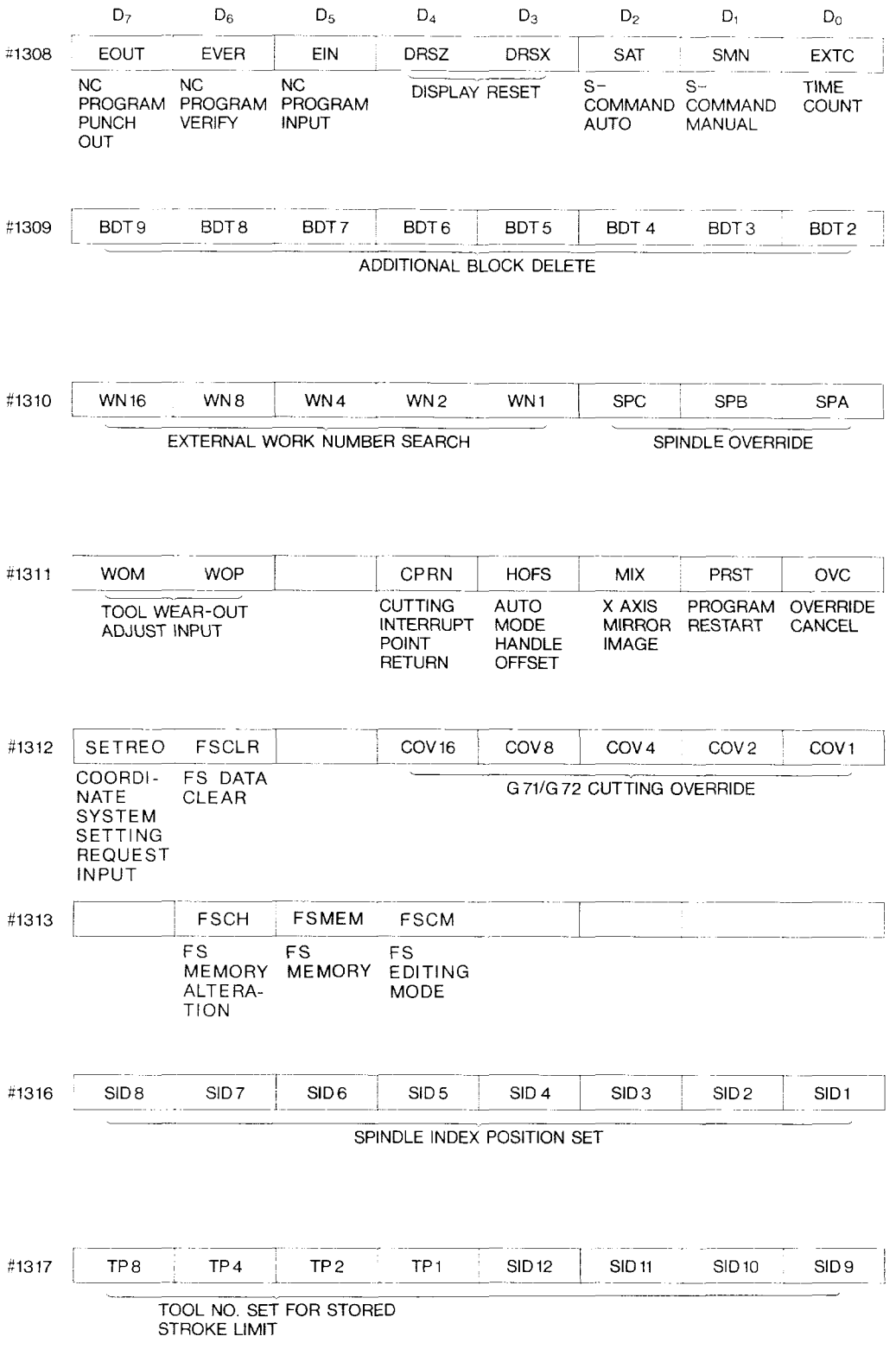
Table 4-1 List of Standard Input/Output Signals

Input Signals

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
#1300	EDT	MEM	D	T		H/S	J	RT
	EDIT	MEMORY	MDI	TAPE		HANDLE/ STEP	MANUAL JOG	MANUAL RAPID
#1301	MP1	ROV2	ROV1	FV16	FV8	FV4	FV2	FV1
	RAPID SPEED OVERRIDE			FEEDRATE OVERRIDE/MANUAL JOG SPEED				
#1302	HZ	HX	−Z	+Z	−X	+X	MP4	MP2
	MANUAL PG AXIS SELECT		MANUAL TRAVERSE AXIS DIRECTION SELECT				MANUAL PG MULTIPLY SELECT	
#1303	INHEDT	AFL	ABS	DRN	BDT	DLK	MLK	SBK
	INHIBIT EDIT	M. S. T LOCK	MANUAL ABS.	DRY RUN	BLOCK DELETE	DISPLAY LOCK	MACHINE LOCK	SINGLE BLOCK
#1304	ZRN	CDZ	SWZ	RWDH	SRN	PST	*SP	ST
	RETURN TO REFER-ENCE	THREAD CUT UP	ERROR DETECT	HIGH-SPEED REWIND	SET UP POINT RETURN	POSITION SET	FEED HOLD	CYCLE START
#1305	ERR1	ERR0	STLK	RWD	EOP	ERS	FIN	MRD
	EXTERNAL ERROR INPUT		INTER-RUPT	REWIND	END OF PROGRAM	EXTERNAL RESET	MST FIN	MACHINE READY
#1306	SAGR	HINTP	*DCZ	*DCX	*−LZ	*+LZ	*−LX	*+LX
	SPINDLE SPEED AGREE-MENT	HANDLE INTERPO-LATION MODE	DECREASE INPUT FOR REFERENCE POINT		OVERTRAVEL INPUT			
#1307	GRS	GSC	SSTP	SINV	GR4	GR3	GR2	GR1
	S-COMMAND CON-STANT	SPINDLE SPEED CONSTANT	S-COMMAND "0"	S-COMMAND INVERT	SPINDLE GEAR RANGE SELECT			

Table 4-1 List of Standard Input/Output Signals (Cont'd)

Input Signals



Input Signals

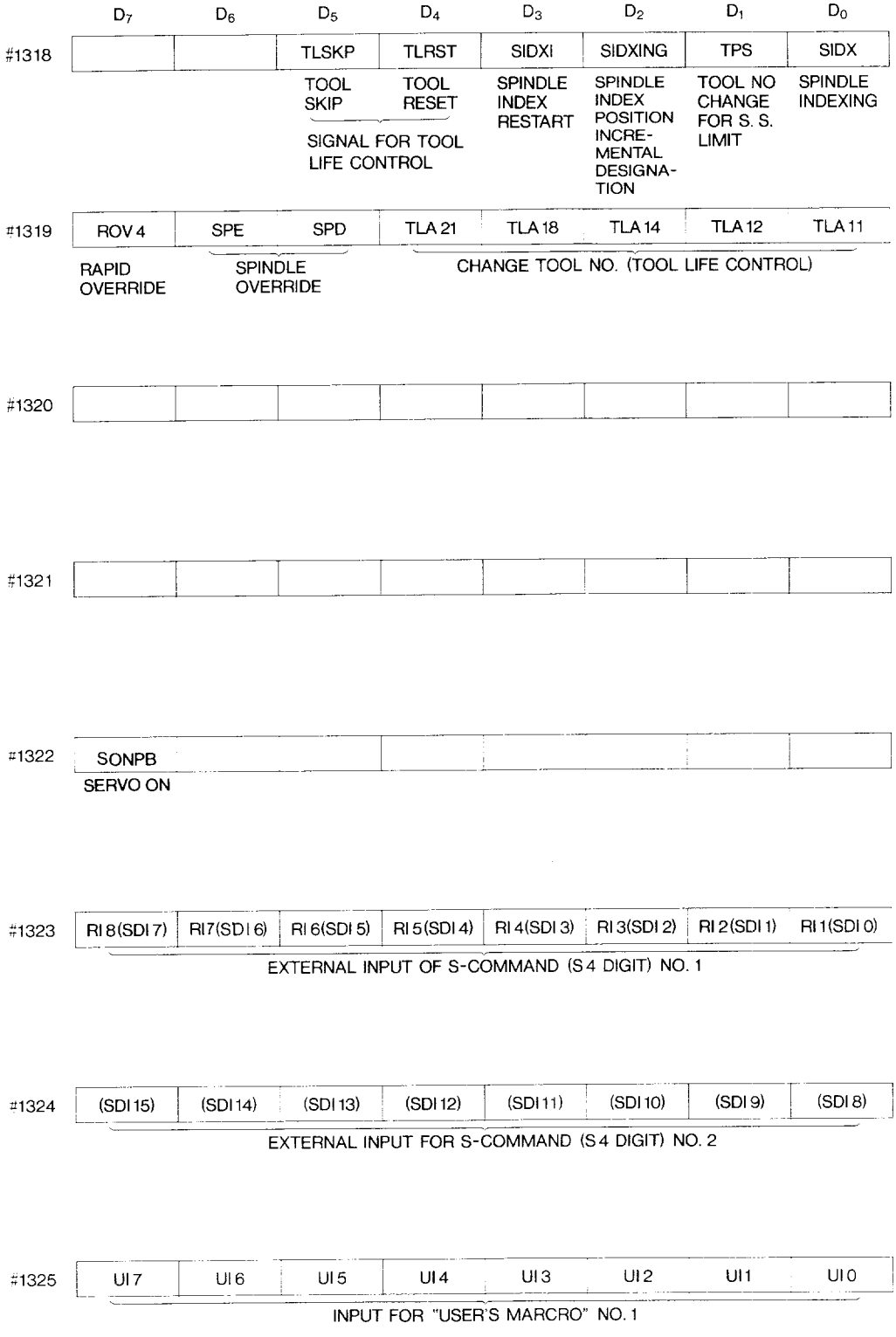
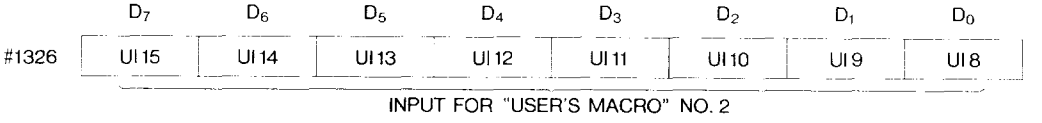
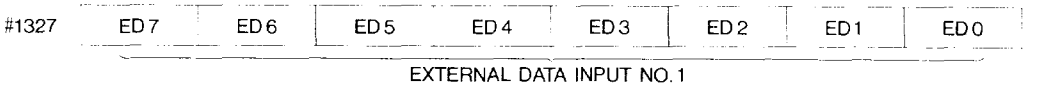
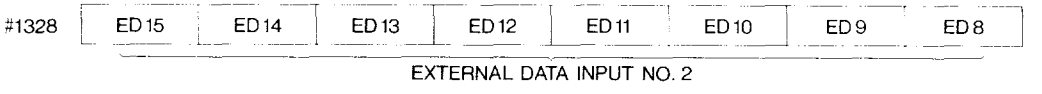


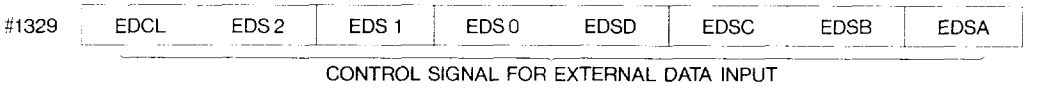
Table 4-1 List of Standard Input/Output Signals (Cont'd)

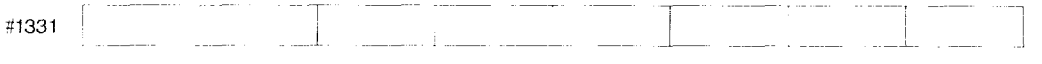
Input Signals

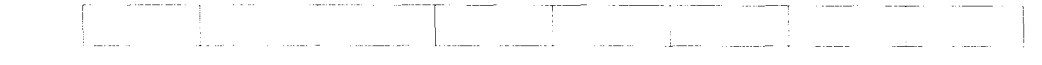












Output Signals

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
#1200	M28	M24	M22	M21	M18	M14	M12	M11

M FUNCTION BCD OUTPUT

#1201	M30R	M02R	M01R	M00R	M38	M34	M32	M31
	M30 DECODE OUTPUT	M02 DECODE OUTPUT	M01 DECODE OUTPUT	M00 DECODE OUTPUT				

#1202	TF	SF	MF	SINVA	IER	ESPS	RST	ALM
	T-FUNCTION SAMPLING OUTPUT	S-FUNCTION SAMPLING OUTPUT	M-FUNCTION SAMPLING OUTPUT	S4 DIGIT OUT INVERT STATUS	INPUT ERROR OUTPUT	EMERGENCY STOP OUTPUT	RESET OUTPUT	ALARM OUTPUT

#1203		EDTS	AUTO	MAN	THC	RWDS	OP	DEN
		EDIT OPERATING STATUS	AUTO MODE STATUS	MANUAL MODE STATUS	THREAD CUTTING STATUS	REWIND STATUS	FEEDING	POSITIONING END

#1204	S28	S24	S22	S21	S18	S14	S12	S11

S-FUNCTION BCD OUTPUT

#1205	T28	T24	T22	T21	T18	T14	T12	T11

T-FUNCTION BCD OUTPUT

#1206	2ZPZ	2ZPX	ZPZ	ZPX		G96S	SPL	STL
	Z AXIS	X AXIS	Z AXIS	X AXIS		CONSTANT SURFACE SPEED CONTROL	FEED HOLD LAMP	CYCLE START LAMP
	NO. 2 REFERENCE POSITION		REFERENCE POSITION					

#1207								

#1216	R08 (SDD7)	R07 (SDD6)	R06 (SDD5)	R05 (SDD4)	R04 (SDD3)	R03 (SDD2)	R02 (SDD1)	R01 (SDD0)

EXTERNAL OUTPUT FOR S-COMMAND (S4 DIGIT) NO. 1

Table 4-1 List of Standard Input/Output Signals (Cont'd)

Output Signals

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
#1217	(SDD15)	(SDD14)	(SDD13)	(SDD12)	RO12(SDD11)	RO11(SDD10)	RO10(SDD9)	RO9(SDD8)

EXTERNAL OUTPUT FOR S-COMMAND (S4 DIGIT) NO. 2

#1218				FSCE	FSMD		ASF	
				END OF FS MEMORY ALTERA-TION	UNDER FS EDITING MODE		S4 ANALOG OUT	
#1219	ESEND	EREND		SETEND	TLCH	SIDXO	TPSA	SIDXA
	EXTERNAL DATA SEARCH COMPLE-TION	EXTERNAL DATA INPUT COMPLE-TION		END OF COORDI-NATE SYSTEM SETTING OUTPUT	TOOL CHANGE COMMAND (TOOL LIFE CONTROL)	SPINDLE INDEX EXECUT-ING	S. 5. LIMIT AREA CHANGE END	SPINDLE INDEX END
#1220	UO7	UO6	UO5	UO4	UO3	UO2	UO1	UO0

OUTPUT FOR "USER'S MACRO" NO. 1

#1221	UO15	UO14	UO13	UO12	UO11	UO10	UO9	UO8

OUTPUT FOR "USER'S MACRO" NO. 2

#1222								

#1223					FSCLRE			
					END OF FS DATA CLEAR			

#1224	MD7	MD6	MD5	MD4	MD3	MD2	MD1	MD0

HIGH-SPEED M FUNCTIONS

Output Signals

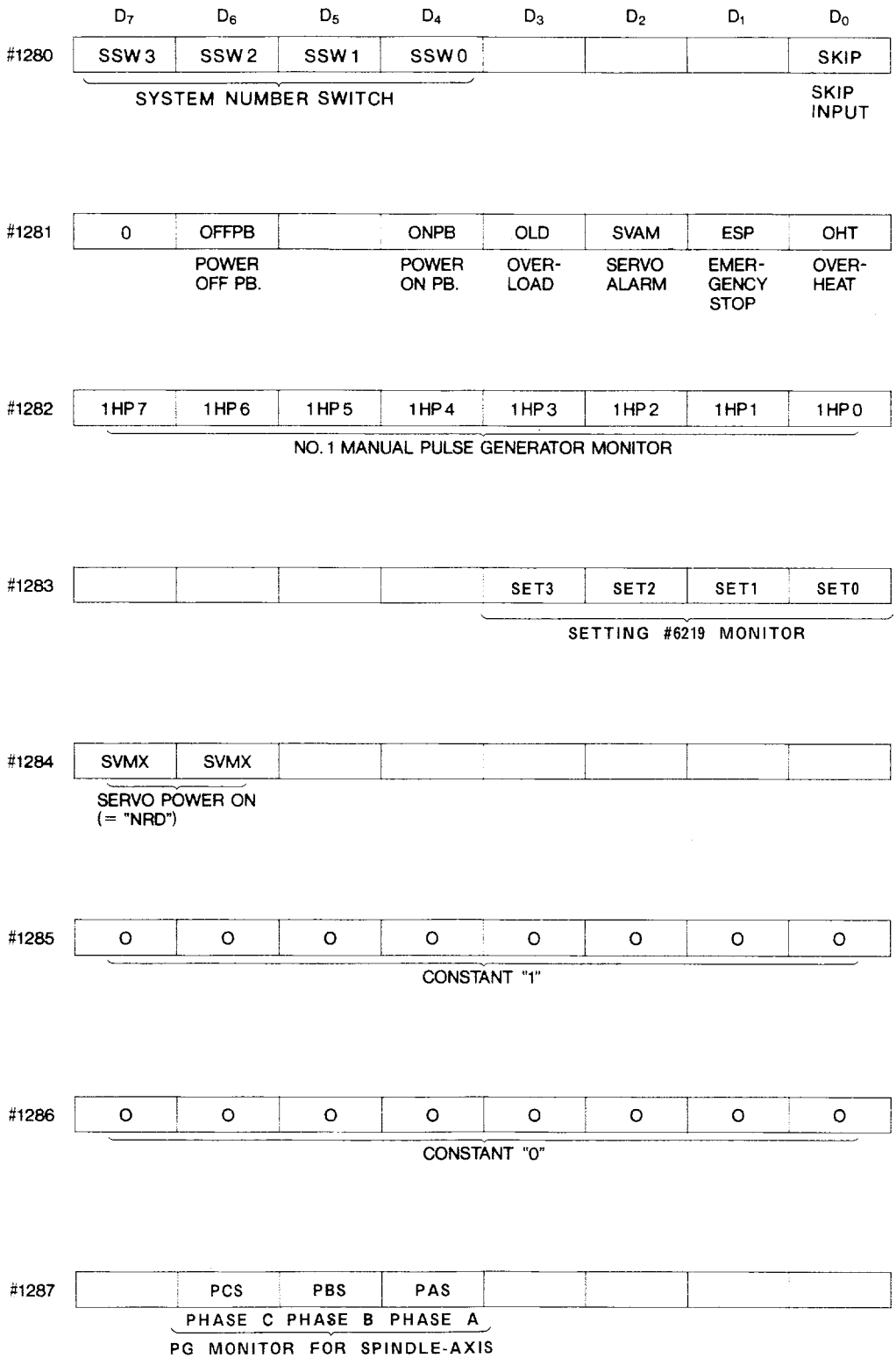
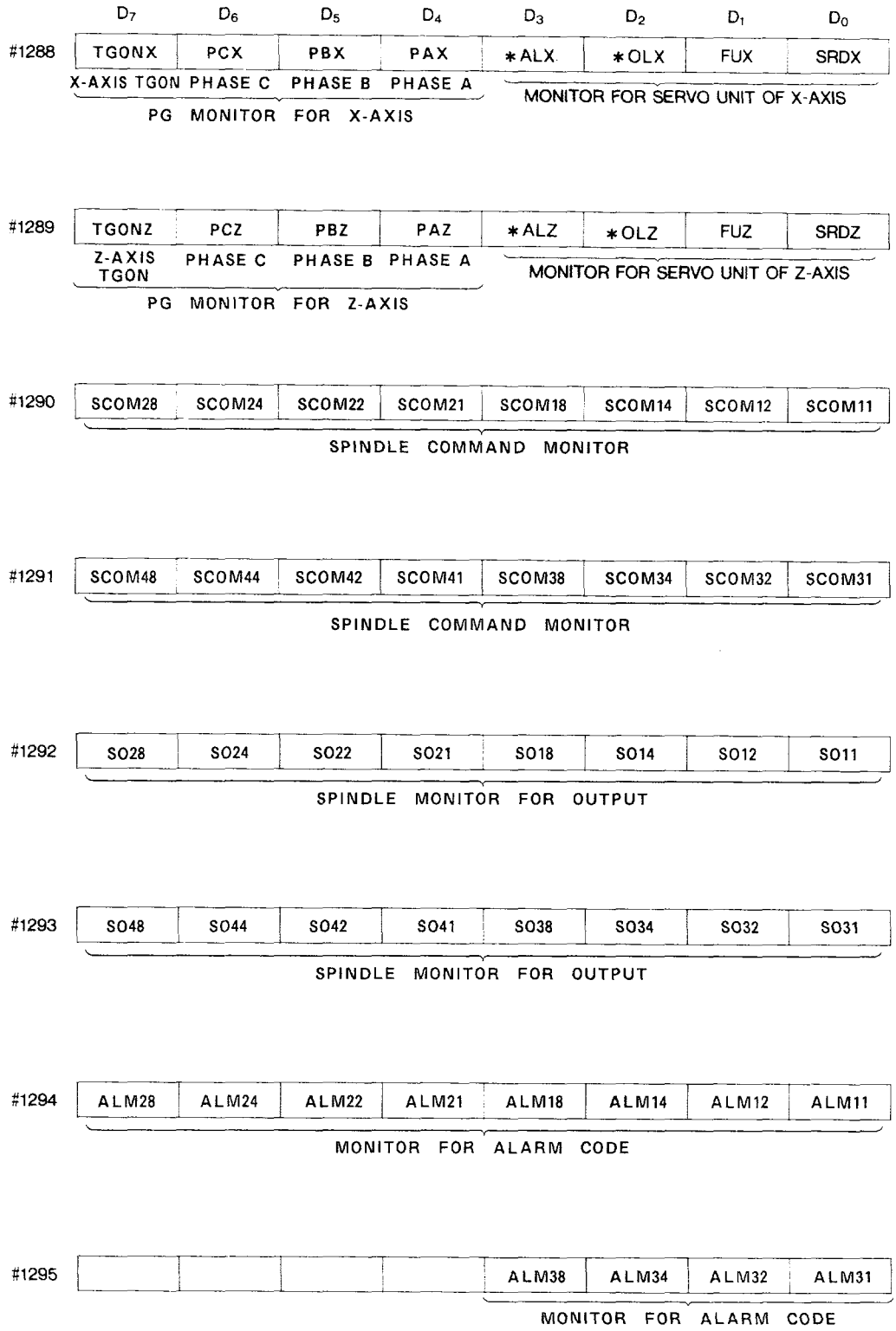


Table 4-1 List of Standard Input/Output Signals (Cont'd)

Output Signals



APPENDIX 5 LIST OF ALARM CODES

Code	Causes	Code	Causes
000		012	OVERFLOW (128 CH) BUFFER CAPACITY OVERFLOW IN A BLOCK (128 CHARACTERS).
001	ZR UNREADY (X) REFERENCE POINT RETURN NOT COMPLETED X.	013	PROG ERROR (NO ADDRESS) ADDRESS PLUS NO DATA AND NEXT ADDRESS COMMAND. OR NO ADDRESS PLUS DATA.
002	ZR UNREADY (Z) REFERENCE POINT RETURN NOT COMPLETED Z.	014	PROG ERROR ("−," ".") SIGN "−," SND "." NOT CORRECTLY USED.
003		015	PROG ERROR (UNUSABLE CH) UNUSABLE CHARACTER PROGRAMMED IN INSIGNFICANT DATA AREA.
004		016	
005	RESET UNREADY (AFTER EDITING) CYCLE START WITHOUT DEPRESSING RESET BUTTON AFTER EDITING.	017	PROG ERROR (8 DIGITS) INPUT DATA OVERFLOW (MORE THAN 8 CHARACTERS).
006		018	
007		019	
008		020	PROG ERROR (G) UNUSABLE G CODE OR G CODE NOT INCLUDED IN OPTIONS PROGRAMMED.
009		021	PROG ERROR (G) G CODES IN 1, AND * GROUPS PROGRAMMED SIMULTANEOUSLY IN A BLOCK.
010	TH ERROR TAPE HORIZONTAL PARITY ERROR.	022	
011	TV ERROR TAPE VERTICAL PARITY ERROR.	023	

APPENDIX 5 LIST OF ALARM CODES (Cont'd)

Code	Causes	Code	Causes
024	PROG ERROR (G, G41-44) UNUSABLE G CODE COMMANDED DURING NOSE RADIUS COMPENSATION.	036	PROG ERROR (P-G10) TOO LARGE P (NUMBER DESIGNATION) WHEN OFFSET IS PROGRAM-INPUT.
025		037	PROG ERROR (G10) TOO LARGE R WHEN WORK COORDINATE SYSTEM IS PROGRAM-INPUT.
026	PROG ERROR (G41-44) RISE ERROR IN NOSE RADIUS COMPENSATION START.	038	
027	PROG ERROR (G41-44) ERROR DURING NOSE RADIUS COMPENSATION CANCELLATION. (ERROR IN CIRCULAR INTERPOLATION MODE).	039	
028		040	PROG ERROR (M98, G65/66) P NOT PROGRAMMED IN G65/66 BLOCK. P OR Q NOT PROGRAMMED IN M98 BLOCK.
029		041	NO PROG PROGRAM NO. (SEQUENCE NO.) NOT FOUND WHEN PROGRAM IS CALLED BY M98, M99, G65, G66, G, M, AND T.
030	PROG ERROR (F/E) NO F OR E COMMAND IN FEED COMMAND. E COMMANDED IN G98 MODE.	042	PROG ERROR (M98, G65/66 NEST) SUBPROGRAM (M98) OR MACRO CALL (G65/66) FIVE-NESTED.
031	PROG ERROR (R = 0) CIRCLE WITH RADIUS 0 COMMANDED IN CIRCULAR ARC COMMAND	043	PROG ERROR (M91) P NOT SPECIFIED IN M91 BLOCK.
032		044	TOOL DATA ERROR (R AND C) THE NOSE R AMOUNT IS WRITTEN IN THE OFFSET NUMBER COMMANDED BY [ ] OF THE T ** [ ] COMMAND.
033		045	
034	PROG ERROR (G02/03) CIRCULAR ARC R DESIGNATION ERROR.	046	MACRO ERROR (NOSE R) DURING TOOL NOSE RADIUS COMPENSATION, VARIABLES LIKE #5041 AND #5042 WHICH CANNOT BE READ IN ADVANCE.
035	PROG ERROR (T OFS) TOO LARGE NO. OF T OFS CODE FOR TOOL RADIUS COMPENSATION AND TOOL LENGTH COMPENSATION.	047	PROG ERROR (G41-44) INTERFERENCE CHECK ERROR

Code	Causes	Code	Causes
048	PROG ERROR (G41-44) INTERSECTION POINT NOT OBTAINED BY INTERSECTION COMPUTATION.	060	PROG ERROR (G34) LEAD INCREASE/DECREASE VALUE EXCEEDING MAXIMUM PROGRAMMABLE VALUE DURING VARIABLE LEAD THREAD CUTTING. MINUS VALUE OF LEAD COMMANDED.
049	PROG ERROR (G41-44) REVERSE OR ALMOST REVERSE COMMANDED IN M97 MODE.	061	PROG ERROR (G11/G12 IN THREAD) ROUNDING, BEVELING COMMANDED IN THREAD CUTTING BLOCK.
050	PROG ERROR (G11/12) I, K, R NOT CORRECTLY COMMANDED FOR BEVELING AND ROUNDING. VALUES OF I, K, R TOO LARGE.	062	PROG ERROR (G32/G33) THREAD CUTTING COMMANDED IN G98 MODE. MULTI-START THREAD CUTTING COMMAND B OUT OF SPECIFIED RANGE.
051	PROG ERROR (G11/12) TAPERING COMMAND IN BLOCKS FOR BEVELING AND ROUNDING.	063	PROG ERROR (G92/G78/G21) RAPID THREAD PULL-UP VALUE IN X-AXIS DIRECTION IN THREAD CUTTING WITH BEVELING SMALLER THAN BEVELING VALUE SET BY PARAMETER.
052	PROG ERROR (G01) ANGLE PROGRAMMING NOT CORRECT DURING ANGLE PROGRAMMING LINEAR INTERPOLATION BY G01.	064	PROG ERROR (G92/G78/G21) RARID THREAD PULL-UP VALUE IN Z-AXIS DIRECTION IN THREAD CUTTING WITH BEVELING VALUE SET BY PARAMETER.
053	PROG ERROR (G50T/G92T) VALUES OF TOOL COORDINATE MEMORY OUT OF THE RANGE BETWEEN 51 TO 80 IN WORK COORDINATE SYSTEM SETTING BY G50T.	065	
054	G51 POS-EXT ERROR COMMAND VALUE (EXTERNAL VALUE) TO RETURN AT G51 EXECUTION TIME IS WRONG.	066	CANNOT CONTINUOUS THREAD TOO SHORT TIME FOR 1 BLOCK OF CONTINUOUS THREAD CUTTING.
055	PROG ERROR (M, S, T) M, S, T COMMANDS IN THE BLOCK IN WHICH M, S, T CODE CANNOT BE COMMANDED.	067	
		068	
056	PROG ERROR (AXIS) AXIS COMMAND IN G20, G21 BLOCKS. AXIS NOT CORRECTLY COMMANDED IN G04, G36-G38.	069	PROG ERROR (G151 DATA) ERROR IS FOUND IN THE G151 COMMAND DATA.
057		070	PROG ERROR (M02/M30/M99) MEMORY OPERATION COMPLETION COMMAND NOT GIVEN.
058	MIRROR IMAGE G28 IS COMMANDED DURING MIRROR IMAGE MODE.	071	G151 ERROR G151 COMMAND ERROR.
059	ZR UNREADY G28 NOT COMPLETED ON THE AXIS WHICH HAS G29 COMMAND OR REFERENCE POINT RETURN NOT COMPLETED ON THE AXIS WHICH HAS G30 COMMAND.	072	PROG ERROR (G50T/G50XZ) G50T OR G50XZ IS COMMANDED WHEN THE COORDINATE SYSTEM IS SET BY THE T CODE. G68 IS COMMANDED BY MIRROR IMAGE GRADE UP SPECIFICATIONS (#6019D1 = 1).

Code	Causes	Code	Causes
073		085	EXTERNAL CMP ERROR MULTIPLICATION FACTOR SET BY PARAMETER EXCEEDING 11 FOR EXTERNAL TOOL COMPENSATION.
074		086	EXTERNAL CMP ERROR ERROR INPUT TURNED ON DURING EXTERNAL TOOL COMPENSATION.
075		087	PROG ERROR (G31/G35) TOUCH SWITCH NOT ON WHEN MOTION REACHES END POINT BY SKIP OR TOOL SET ERROR COMPENSATION COMMANDS.
076		088	
077	RS 232C ERROR (OVER-RUN) 10 CHARACTERS MORE HAVE BEEN READ IN AFTER STOP CODE HAS BEEN TRANSMITTED THROUGH RS232C INTERFACE.	089	PROG ERROR (G90/G92/G94) UNUSABLE ADDRESSES SPECIFIED IN G90, G92, AND G94 BLOCKS.
078		090	PROG ERROR (G70-76/G72-78) P, Q NOT COMMANDED IN G70, 71, 72, 73 BLOCKS.
079		091	PROG ERROR (G70-76/G72-78) BLOCK OF SEQUENCE NO. SPECIFIED BY P, Q IN G70 NOT FOUND PROG NO. INCLUDING IN G70 BLOCK.
080	TOOL SET CMP ERROR T CODE COMMANDED BEFORE G35 BLOCK. G98 COMMANDED IN OR BEFORE G35 BLOCK.	092	PROG ERROR (G70-76/G72-78) NO. OF BLOCKS INCLUDING FINISHED SHAPE PROGRAM SPECIFIED BY P, Q IN G70, G71, G72, AND G73, OVER 46.
081	TOOL SET CMP ERROR ERROR OF PARAMETER SETTING FOR TOOL SET ERROR COMPENSATION (X).	093	PROG ERROR (G70-76/G72-78) UNABLE G-AND M-CODE IN FINISHED SHAPE PROGRAM SPECIFIED BY P, Q IN G70, G71, G72, AND G73.
082	TOOL SET CMP ERROR ERROR OF PARAMETER SETTING FOR TOOL SET ERROR COMPENSATION (Z).	094	PROG ERROR (G70-76/G72-78) BEVELING AND ROUNDING COMMANDS AS LAST MOVE COMMAND FOR FINISHED SHAPE PROGRAM SPECIFIED BY P, Q IN G70, G71, G72, AND G73.
083	TOOL WEAR CMP ERROR COMPENSATION NO. EXCEPT 01 TO 19 DESIGNATED AT TOOL WEAR COMPENSATION.	095	PROG ERROR (G70-76/G72-78) FAULTS IN FINISHED SHAPE PROGRAM SPECIFIED BY P, Q IN G71, G72. G29 COMMANDED IN FINISHING SHAPE PROGRAM SPECIFIED BY P AND Q IN G70, G71, G72, AND G73.
084	TOOL WEAR CMP ERROR TOOL WEAR COMPENSATION INPUTS WOM, WOP GIVEN SIMULTANEOUSLY.	096	PROG ERROR (G70-76/G72-78) D (CUTTING FREQUENCY) SPECIFIED BY G73 ZERO OR 128 OR MORE. I, K (ROUGH CUTTING) SPECIFIED BY G73 BOTH ZERO. D, K OF G76 EXCEEDING PROGRAMMABLE RANGE.

Code	Causes	Code	Causes
097	PROG ERROR (G70-76/G72-78) FOUR OR MORE PROCESSING INTERRUPTIONS BY FINISHED SHAPE PROGRAM IN STOCK REMOVAL CYCLE BY G71 R1, OR G72 R1.	109	MACRO ERROR (# NO. NOT LEFT) PROHIBITED VARIABLE DESIGNATED AS SUBSTITUTION.
098	PROG ERROR (G70-76/G72-78) DATA SPECIFIED BY G70 P, Q NOT REGISTERED IN INTERNAL KEEP MEMORY.	110	MACRO ERROR ([ ] 5 LIMIT) MULTIPLE LAYERS OF PARENTHESES EXCEEDDING THE UPPER LIMITS (5).
099	PROG ERROR (G70-76/G72-78) UNUSABLE ADDRESSES SPECIFIED IN G70 TO G76 BLOCKS. ADDRESS REQUIRED IN THE BLOCK NOT SPECIFIED.	111	MACRO ERROR (MOVE G66-M99) MOVE COMMAND IN M99 FINISHING COMMAND OF MACRO CALLED BY G66.
100	CAL ERROR (FIXED POINT) MAGNITUDE OF FIXED POINT DATA EXCEEDING UPPER LIMIT.	112	MACRO ERROR MULTIPLE LEVELS OF MACRO CALL EXCEEDING THE UPPER LIMIT 4.
101	CAL ERROR (FLOATING) EXPONENT OF FLOATING POINT DATA EXCEEDING ALLOWABLE RANGE.	113	
102	CAL ERROR (DIVISION) CALCULATION DIVISOR ZERO OR OVERFLOW ERROR.	114	MACRO ERROR (DO-FORMAT) "DO" NOT CORRESPONDING TO "END."
103	CAL ERROR (SQUARE ROOT) ROOT VALUE IS A NEGATIVE $\sqrt{(-)}$.	115	MACRO ERROR ([ ] UNMATCH) FORMAT ERROR IN <EQUATION> .
104	PROG ERROR (DOUBLE ADR) CHARACTER WHICH CANNOT BE REPEATED IN A BLOCK COMMAND IN REPETITION.	116	MACRO ERROR (DO END NO.) "m" IN DO m OUT OF RANGE $1 \leq m \leq 3$.
105	MACRO ERROR (CONSTANT) CONSTANT EXCEEDING THE LIMIT.	117	
106	MACRO ERROR TOO MANY CODES FOR CANCELLING G67.	118	MACRO ERROR (GO TO N) "n" in GO TO n 0 < n9999 APPROPRIATE SEQUENCE NO. NOT FOUND.
107	MACRO ERROR (FORMAT) ERROR IN THE FORMAT EXCEPT FOR EQUATION.	119	AS AFTER READ SUBPROG/MACRO ADDRESS SEARCHED DURING EXECUTION OF SUBPROGRAM OR MACRO PROGRAM.
108	MACRO ERROR (UNDEFIN #NO.) UNDEFINED VARIABLE NO. DESIGNATED.	120	PRTN ERROR (NOT FOUND) SEQUENCE NO. SEARCHED NOT FOUND IN PART PROGRAM.

APPENDIX 5 LIST OF ALARM CODES (Cont'd)

Code	Causes	Code	Causes
121	PRTN ERROR (G50/G92) G31 COMMANDED DURING PROGRAM RESTART.		
122		134	NO PROG (EXT) NOT FOUND PROGRAM NO. SPECIFIED BY EXTERNAL NO. SEARCH.
123		135	EXT DATA ERROR IN DATA GIVEN BY EXTERNAL DATA INPUT.
124	PRTN ERROR (MDI MOVE) AXIS OPERATED BY MDI AFTER PROGRAM RESTART PREPARATION.	136	
125		137	
126		138	
127		139	
128		140	PROG ERROR (G111/G112) ERROR IN ADDRESS WORD COMMANDING OF G111 BLOCK.
129		141	PROG ERROR (G111/G112) ANGLE FOR ANGLE PROGRAMMING A, B BY G111 OUT OF RANGE $-360 \leqq A, B \leqq 360$.
130	EXT DATA DATA ERROR IN A GROUP DATA.	142	PROG ERROR (G111/G112) 1ST BEVELING PORTION OUTSIDE RECTANGLE COMPOSED BY START AND END POINTS OR BETWEEN 45° STRAIGHT LINES OF START TO END POINTS AND END TO START POINTS.
131		143	PROG ERROR (G111/G112) ERROR IN G111 COMMAND BLOCK
132		144	PROG ERROR (G111/G112) M, S T COMMAND IN G111, G112 BLOCK.

Code	Causes
145	PROG ERROR (G111/G112) ERROR IN COMMANDING ADDRESS WORD FOR G112 BLOCK.
146	PROG ERROR (G111/G112) ERROR IN COMMANDING PROGRAMMED SHAPE FORMED BY G112 BLOCK.
147	
148	
149	
150	PROG ERROR (G122/G123) G 122 AND G 123 COMMAND BLOCKS ARE NOT INDEPENDENT COMMANDS. THE SAME TOOL IS REGISTERED WITH MORE THAN TWO GROUPS. THE SAME OFFSET MEMORY NO. IS USED WITH MORE THAN TWO TOOLS. ADDRESSES OTHER P, L AND T ARE INSTRUCTED TO THE TOLL INFORMATION REGISTRATION PROGRAM.
151	PROG ERROR (G122/G123) NO DESIGNATION OF GROUP NO. P IS GIVEN, OR A VALUE OTHER THAN 1 ≦ P ≦ 19 IS GIVEN. NO DESIGNATION OF L AS THE LIFE PER TOOL IS GIVEN, OR A VALUE OTHER THAN 1≦ L ≦ 9999 IS GIVEN.
152	PROG ERROR (G122/G123) A VALUE OTHER THAN 1≦ TOOL NO. ≦ 50 IS INSTRUCTED AS THE TOOL NO. A VALUE OTHER THAN 1 ≦ COMPENSATION MEMORY NO. ≦ 50 IS INSTRUCTED AS THE COMPENSATION MEMORY NO. VALUES OTHER 0 OR MORE THAN 20 ARE INSTRUCTED AS [ ][ ] (GROUP NO.) COMMANDS OF T[ ][ ] 90 TO T[ ][ ] 95 AND T[ ][ ] 99 COMMANDS.
153	PROG ERROR (G122/G123) THE TOOL INFORMATION OF THE SAME GROUP NO. IS REGISTERED TWICE.
154	PROG ERROR (G122/G123) MORE THAN 16 TYPES OF "TOOL NO. + COMPENSATION MEMORY NO." ARE ATTEMPTED TO BE REGISTERED WITH ONE GROUP NO.
155	PROG ERROR (G122/G123) TOOLS OF THE DESIGNATED GROUP NO. ARE NOT REGISTERED. T[ ][ ] 92, T[ ][ ] 93, T[ ][ ] 94 AND T[ ][ ] 95 ARE INSTRUCTED, BUT THE CORRESPONDING COMPENSATION MEMORY NOS. ARE NOT REGISTERED.
156	PROG ERROR (G122/G123) THE COMMAND FOR TOOL LIFE MANAGEMENT IS ISSUSED WITH EQUIPMENT THAT DID NOT HAVE THE ADDED FUNCTIONS OF "T4 DIGIT DESIGNATION" AND "OFFSET MEMORY ADDITION."
157	PROG ERROR (G122/G123) THE TOOL EXCHANGE REQUEST OUTPUT IS ON.
158	PROG ERROR (G122/G123) THERE ARE TOOL NOS. THAT ARE MORE THAN "31" WITH REGISTERED TOOLS WITHIN TOOL GROUP NOS. INSTRUCTED BY [ ][ ] OF WORKPIECE COORDINATE SYSTEM SETTING (G 50 T[ ][ ] 90).
159	PROG ERROR (G122/G123) MORE THAN 6 TYPES OF COMPENSATION MEMORY NOS. ARE REGISTERED WITH ONE TOOL.
170	MEM ERROR (OFS) TOOL COMPENSATION TOTAL CHECK ERROR.
171	
172	MEM ERROR (SET) SETTING AREA TOTAL CHECK ERROR.
173	MEM ERROR (PRM) PARAMETER AREA TOTAL CHECK ERROR.
174	MEM ERROR (KEEP) KEEP MEMORY TOTAL CHECK ERROR.
175	MEM ERROR (MACR) MACRO TOTAL CHECK ERROR.
176	
177	
178	TOOL NO. SET ERROR TOOL NO. INPUT SIGNALS #1317 (D4 TO D7) #1342 (D4 TO D6) ARE MORE THAN 51.

APPENDIX 5 LIST OF ALARM CODES (Cont'd)

Code	Causes	Code	Causes
179	OVER TEMP PANEL INSIDE TEMPERATURE TOO HIGH.	191	HDLC SOFT COMMAND ERROR NC SIDE RECEIVED DATA OTHER THAN TRANSMISSION PROTOCOL BETWEEN ACGC AND NC.
180	SEQ ERROR SEQUENCE ERROR (1)	192	HDLC 8530 SEND ERROR TRANSMISSION NC SEND-SIDE HARDWARE ERROR.
181		193	HDLC 8530 ERROR TRANSMISSION NC RECEIVE-SIDE HARDWARE ERROR.
182		194	HDLC NAK ERROR (10 TIMES) SAME DATA IS SENT 10 TIMES IN TRANSMISSION FROM NC TO ACGC WITHOUT SUCCESS.
183		195	HDLC CMOS FILE ERROR CMOS FILE READ OR WRITE REQUEST FROM ACGC TO NC IS NOT EXECUTED.
184		196	HDLC I/O DATA ERROR SYSTEM VARIABLE READING FROM ACGC TO NC IS NOT EXECUTED.
185		197	HDLC NC REQUEST ERROR OPERATION REQUEST COMMAND FROM ACGC TO NC IS WRONG.
186		198	HDLC OPERATION ERROR DATA TRANSFER FROM ACGC TO NC IS DELAYED MORE THAN 1 MINUTE DURING ACGC OPERATION.
187		199	AMGC COMMUNICATION SYSTEM ERROR COMMUNICATION ERROR BETWEEN AMGC AND NC.
188		200	ACGC2 COMMUNICATION SYSTEM ERROR COMMUNICATION ERROR BETWEEN ACGC120 AND NC.
189		201	OT (X) OVERTRAVEL X.
190	HDLC SYSTEM SOFT ERROR TRANSMISSION SYSTEM SOFTWARE ERROR.	202	OT (Z) OVERTRAVEL Z.

Code	Causes	Code	Causes
203		215	
204		216	
205		217	
206		218	
207		219	
208		220	S-OT2 (INSIDE) STORED STROKE LIMIT SECOND AREA (INSIDE INHIBIT).
209		221	S-OT2 (X) STORED STROKE LIMIT SECOND AREA (OUTSIDE INHIBIT) X.
210		222	S-OT2(Z) STORED STROKE LIMIT SECOND AREA (OUTSIDE INHIBIT) Z.
211	S-OT1 (X) STORED STROKE LIMIT FIRST AREA X.	223	S-OT3 (INSIDE) STORED STROKE LIMIT THIRD AREA (OUTSIDE INHIBIT).
212	S-OT1 (Z) STORED STROKE LIMIT FIRST AREA Z.	224	S-OT3 (X) STORED STROKE LIMIT THIRD AREA (OUTSIDE INHIBIT) X.
213		225	S-OT3 (Z) STORED STROKE LIMIT THIRD AREA (OUTSIDE INHIBIT) Z.
214		226	

APPENDIX 5 LIST OF ALARM CODES (Cont'd)

Code	Causes	Code	Causes
227		239	
228		240	
229		241	ZR ERROR-POS (X) REFERENCE POINT RETURN POSITION ERROR X.
230		242	ZR ERROR-POS (Z) REFERENCE POINT RETURN POSITION ERROR Z.
231	ZR ERROR-AREA (X) REFERENCE POINT RETURN AREA ERROR X.	243	
232	ZR ERROR-AREA (Z) REFERENCE POINT RETURN AREA ERROR Z.	244	
233		245	
234		246	
235		247	
236		248	
237		249	
238		270	

Code	Causes	Code	Causes
271	P-SET ERROR (X) P SET ERROR X.	283	
272	P-SET ERROR (Z) P-SET ERROR Z.	284	
273		285	
274		286	
275		287	
276		288	
277		289	
278		310	SERVO OFF SERVO POWER NOT SUPPLIED.
279		311	
280	MACH UNREADY MATCH RDY OFF.	312	
281		313	
282		314	

APPENDIX 5 LIST OF ALARM CODES (Cont'd)

Code	Causes	Code	Causes
315		327	UNFINISHED PROG GEN MEMORY GENERATION IS NOT PERFORMED. PROGRAM SPECIFIED BY 0, –, 9, 9, 9, 9 MUST BE ERASED.
316		328	UNSUITABLE MM21 MEMORY BOARD WHEN PROGRAM AREA GENERATION IS MADE DOES NOT MATCH THE CURRENT MEMORY BOARD.
317		329	PC CPU ERROR PC AND CPU ERROR. (SET #6014D7 TO 0 FOR RELEASE)
318		330	EMERGENCY STOP EMERGENCY STOP.
319		331	FUSE (X) FUSE BLOWN X.
320	NC UNREADY NC UNREADY P SET UNREADY.	332	FUSE(Z) FUSE BLOWN Z.
321		333	
322	CMOS VACANT AREA ERROR DATA OTHER THAN "0" ARE WRITTEN IN THE C-MOS MEMORY AREA OTHER THAN THE PARAMETER, OFFSET, KEEP MEMORY, COMMON VARIABLE, OR PROGRAM.	334	
323	ACGC2 SYNC ERROR SYNCHRONOUS ERROR OF ACGC 120 CPU AND NC SIDE CPU. (SET #6014D5 TO 0 FOR RELEASE)	335	
324	AMGC SYNC ERROR SYNCHRONOUS ERROR OF AMGC CPU AND NC SIDE CPU. (SET #6014D5 TO 0 FOR RELEASE)	336	
325	SERVO CPU ERROR SYNCHRONOUS ERROR OF SERVO CPU AND DATA CPU. (SET #6014D6 TO 0 FOR RELEASE)	337	
326	ACGC120/AMGC SERVO OFF REQ. THERE WAS A SERVO-OFF REQUEST FROM THE ACGC 120 OR AMGC.	338	

Code	Causes	Code	Causes
339		351	OL (X) OVERLOAD (1) X.
340		352	OL (Z) OVERLOAD (1) Z.
341	SERVO ERROR (X) SERVO ERROR X.	353	
342	SERVO ERROR (Z) SERVO ERROR Z.	354	
343		355	
344		356	
345		357	OL (OTHER) OVERLOAD (2).
346		358	
347		359	
348		360	
349		361	PG ERROR (X) PG ERROR X.
350		362	PG ERROR (Z) PG ERROR Z.

Code	Causes	Code	Causes
363		375	
364		376	
365		377	
366	PG ERROR (S) PG ERROR S.	378	
367		379	
368		380	
369		381	AXIS ERROR (1) AXIS ERROR (1).
370		382	AXIS ERROR (2) AXIS ERROR (2).
371	FG ERROR (1) FG ERROR 1.	383	AXIS ERROR (3) AXIS ERROR (3).
372	FG UNREADY FG UNREADY.	384	
373		385	
374		386	

Code	Causes	Code	Causes
387		399	
388		400	SEQ ERROR SEQUENCE ERROR (2).
389		401	
390	SERVO PACK ALARM (S) SERVO PACK ALARM S.	402	
391	SERVO PACK ALARM (X) SERVO PACK ALARM X.	403	
392	SERVO PACK ALARM (Z) SERVO PACK ALARM Z.	404	
393		405	
394		406	
395		407	
396		408	
397		409	
398		810	

APPENDIX 5 LIST OF ALARM CODES (Cont'd)

Code	Causes	Code	Causes
811		823	
812		824	
813		825	
814		826	
815		827	
816		828	
817		829	
818		830	CPU ERROR CPU ERROR (1).
819		831	
820	ROM ERROR ROM CHECK ERROR.	832	
821		833	
822		834	

Code	Causes
835	
836	
837	
838	
839	
840	CPU ERROR CPU ERROR (2).
841	
842	
843	
844	
845	
846	

Code	Causes
847	
848	
849	
910	TAPE-MEM ERROR MEMORY VERIFYING ERROR (OFF-LINE).
920	TAPE ERROR TAPE READING-IN ERROR (OFF-LINE).

ALARM "095"

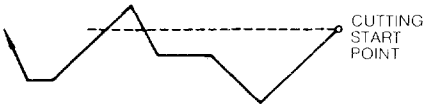
X-coordinates differnt between G71 command cutting start point and last block for finished shape program.

Z-coordinates different between G72 command cutting start point and last block for finished shape program.

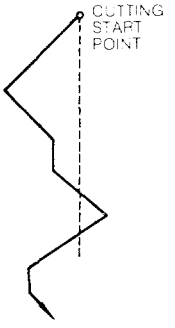
Z-coordinate for cutting start point by G71 command different from Z-coordinate for the first block of the finished shape program. (Command G71 ... R1 is excepted.)

X-coordinate for cutting start point by G72 command different from X-coordinate for the first block of the finished shape program. (Command G72 ... R1 is excepted.)

X-coordinate for finished shape program by G71 ... R1. Command exceeding cutting start point.



Z-coordinate for finished shape program by G72 ··· R1 command exceeding cutting start point.



ALARM "140"

- Commanding one or no address of addresses B, X(U), Z(W) specifying second straight line.
- Commanding two addresses of addresses B, X(U), Z(W) specifying second straight line. In addition to this, one or no address commanded among addresses A,I, K,specifying first straight line.
- Address C specifying first beveling and address P specifying first rounding commanded.
- Address D specifying second beveling and address Q specifying second rounding commanded.
- Commanding addresses X and Z specifying second straight line and Q and D specifying second beveling and rounding.

ALARM "143"

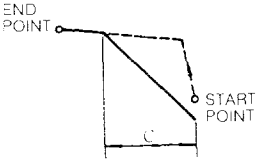
Command values for addresses A, I, K specifying first straight line are determined as follows, and programmed shape cannot be formed.

Command Value for A	—
−360.000, −180.000, C, 180.000, 360.000	Address I commanded for specifying first straight line.
−270.000, −90.000, 90.000, 270.000	Address K commanded for specifying first straight line.

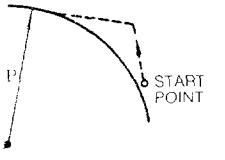
Command values for addresses B, X(U), Z(W) specifying second straight line are determined as follows, and programmed shape cannot be formed.

Command value for B	—
−360.000, −180.000 0, 180.000, 360.000	Address X (U) commanded for specifying second straight line.
−270.000, −90.000 90.000, 270.000	Address Z (W) commanded for specifying second straight line.

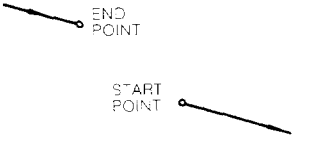
Command values for addresses C and D for beveling too large for the programmed shape. Operation cannot be made according to the command.



Command values for address P and Q specifying radius for rounding too large for the programmed shape. Operation cannot be made according to the command.



No intersecting point for first straight line and second straight line.



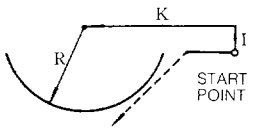
First straight line and second straight line on the same line.

ALARM "145"

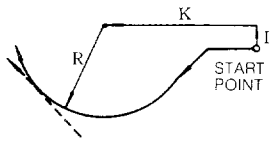
- X(U) or Z(W) not commanded.
- X(U) and Z(W) both commanded.
- R not commanded. Or "0" commanded for R.
- I and K not commanded.
  "0" commanded for I and K.
- P and C both commanded.
- Q and D both commanded.

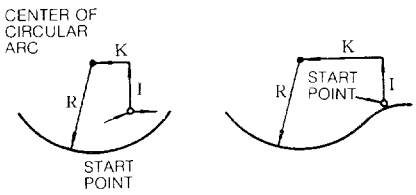
ALARM "146"

Beveling for command C cannot be made.

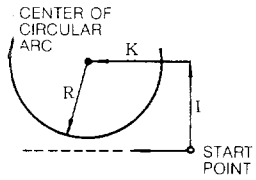


Beveling for command D cannot be made.

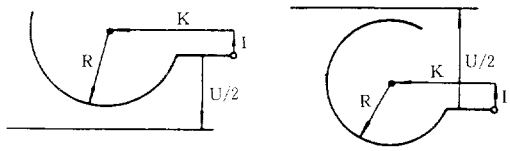




No intersecting point between circular arc and straight line.



No intersecing point between circular arc and end point.



APPENDIX 6 LIST OF DATA

Table 6.1 Address Characters

Address	Meaning	B: Basic O: Optional
A	Angle designation for G01 and G111, included angle for G76	O
B	Spindle shift angle 01 multiple thread, angle designation for multiple cornering	O
C	User macro character	O
D	Depth of cut and number of cutting cycles for G71 to G76	O
E	Specifications for precise feed and precise lead for cutting	B
F	Specifications for normal feed and normal lead for cutting	B
G	Preparatory function (G-function)	B
H	User macro character	O
I	X-component of arc center, canned cycle parameter, beveling value (radius value)	B, O
J	User macro character	O
K	Z-component of arc center, canned cycle parameter, beveling value	B, O
	Incremental value of variable lead thread	O
L	Number of subprogram repetition, G13 to G16 angle and coordinate	B, O
M	Miscellaneous function (M-function)	B
N	Sequence number	B
O	Program number	B
P	Dwell, canned cycle starting sequence number, program number, user macro number	B, O
Q	Subprogram starting sequence number, canned cycle ending sequence number	B, O
R	Radius of arc, rounding value, tool radius value	B, O
S	Spindle function (S-function), maximum spindle revolution	B
T	Tool function (T-function), tool coordinate memory number	B, O
U	X-axis incremental command value, dwell, canned cycle parameter	B, O
V	User macro character	O
W	Z-axis incremental command value, canned cycle parameter	B, O

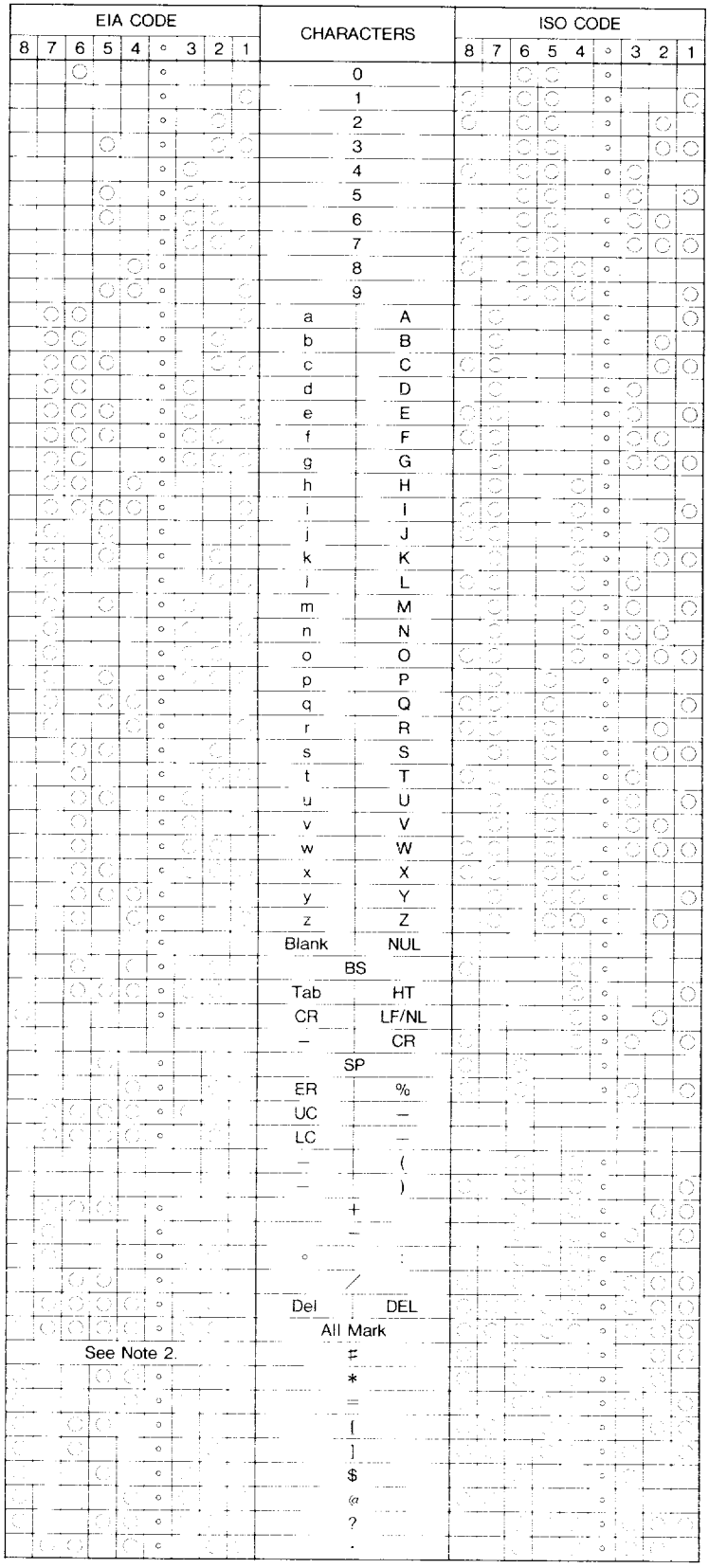
Table 6. 2 Function Characters

EIA Code	ISO Code	Function	Remarks
Blank	NuL	Error in significant data area in EIA Disregarded in ISO	
BS	BS	Disregarded	
Tab	TH	Disregarded	
CR	LF/NL	End of Block (EOB)	
	CR	Disregarded	
SP	SP	Space	
ER	%	Rewind stop	
UC		Upper shift	
LC		Lower shift	
2-4-5 bits	(	Control out (comment start)	EIA: Special code
2-4-7 bits	)	Control in (comment end)	
+	+	Disregarded, User macro operator	
−	−	Minus sign, User macro operator	
0 to 9	0 to 9	Numerals	
a to z	A to Z	Address characters	
0	:	User macro comment	
/	/	Optional block skip	
Del	DEL	Disregarded (Including All Mark)	
.	.	Decimal point	
Parameter starting	#	Sharp (Variable designation)	EIA: Special code
*	*	Asterisk (Multiplication operator)	
=	=	Equal mark	
[	[	Left bracket	
]	]	Right bracket	
$	$	User macro operator	
@	@	User macro operator	
?	?	User macro operator	

Notes :

1. Characters other than the above cause error in significant data area.
2. Information between Control Out and Control In is ignored as insignificant data.
3. Tape code (EIA or ISO) is automatically recognized.

Table 6. 3 Tape Code



Notes :

1. For the hole pattern of EIA code of the characters with an asterisk, the pattern shown in the table is standard. However, other patterns may be specified by parameters.
2. EIA code of character # can be designated by the parameter #6017.

Table 6. 4 Tape Format

No.	Address	Metric Output		Inch Output		B: Basic O: Option
		Metric Input	Inch Input	Metric Input	Inch Input	
1	Program No.	O 4		O 4		B
2	Sequence No.	N 4		N 4		B
3	G-Function	G 3		G 3		B
4	Coordinate Word a: X, Z, I, K, U, W, R	(a + 53)	(a + 44)	a + 53	a + 34	B
5	Feed/min	F 50	F 32	F 50	F 42	B
6	Feed/rev and Thread Lead	F 32	F 24	F 42	F 24	B
		E 34	E 26	E 44	E 26	B
7	S-Function	S 2		S 2		B
		S 4		S 4		O
8	T-Function	T (2 + 1)		T (2 + 1)		B
		T (2 + 2)		T (2 + 2)		O
9	M-Function	M 3		M 3		B
10	Dwell	U (P) 53		U (P) 53		B
11	Program No. Designation	P 4		P 4		B
12	Sequence No. Designation	Q (P) 4		Q (P) 4		B, O
13	No. of Repetitions	L 8		L 8		B
14	Angle Designation for Straight Line	A (B) 33		A (B) 33		O
15	Angle Designation for Multiple Thread	B 3		B 3		O

Notes :

1. Inch/Metric output is set by setting parameter #6007 D_3.
2. Inch/Metric input is set by setting (#6001 D_0).
3. F codes for feed/min or feed/rev can be switched by G 98, G 99.

Table 6.5 List of Program Commands

Address		Metric Output		Inch Output	
		Metric Input	Inch Input	Metric Input	Inch Input
Program No. O		1-9999		1-9999	
Sequence No. N		1-9999		1-9999	
G function G		0-199		0-199	
Coordinate Address[1] X, Z, I, K, U, W, R		±99999.999 mm.	±3937.0078 in.	±99999.999 mm.	9999.9999 in.
Feed/min	F	1-24000 mm/min	0.01-944.88 in./min	1-60960 mm/min	0.01-2400.00 in./min
Feed/rev and Thread Lead	F	0.01-500.00 mm/rev	0.0001-19.6850 in/rev	0.01-1270.00 mm/rev	0.0001-50.0000 in./rev
	E	0.0001-500.0000 mm/rev	0.000004-19.685000 in./rev	0.0003-1270.0000 mm/rev	0.000010-50.000000 in./rev
S-function	S2	0-99		0-99	
	S4	0-9999		0-9999	
T-function	T3	0-999		0-999	
	T4	0-9999		0-9999	
M-function		0-999		0-999	
Dwell U, P		0.001-99999.999 sec		0.001-99999.999 sec	
Program No. Designation		1-9999		1-9999	
Sequence No. Designation		1-9999		1-9999	
No. of Repetitions		1-99999999		1-99999999	
Angle Designation for Straight Line		0-±360.000°		0-±360.000*	
Angle Designation for Multiple Thread		0-360°		0-360°	

*For angle designation of included angle for G76, see 2.8.26.2 Automatic Threading Cycle (G76).

Table 6.6 Data Setting Range

Item		Metric output (screw)		Input output (screw)	
		Metric input	Inch input	Metric input	Inch inqut
Least input increment		0.001 or 0.01 mm	0.0001 or 0.001 in.	0.001 or 0.01 mm	0.0001 or 0.001 in.
Tool offset		0 – ±8388.607 mm	0 – ±330.2601 in.	0 – ±999.999 mm	0 – ±838.8607 in.
Tool radius		0 – ±99.999 mm	0 – ±9.9999 in.	0 – ±99.999 mm	0 – ±9.9999 in.
Minimum step/handle feed		0.001 mm	0.0001 in.	0.001 mm	0.0001 in.
Stored stroke limit area designation unit	Program designation	0.001 mm	0.0001 in.	0.001 mm	0.0001 in.
	Parameter & setting	0.001 mm		0.0001 in.	
Rapid traverse rate	Upper limit value	24 m/min		2400 inches/min	
Manual jog					
F0					
2nd reference point coordinate value		0 – ±99999.999 mm		0 – 9999.9999 in.	
Backlash compensation value		0 – 255 pulses (Note 1)		0 – 255 pulses	

Notes: 1. 1-pulse = least input increment
2. X-axis designated with diameter (except for pulse display)

Table 6.7 List of G Codes

B: Basic
O: Optional

G Code	Special G Code I	Special G Code II	Group	Function		Section
G00	G00	G00	01	Positioning (rapid traverse feed)		B
G01	G01	G01		Linear interpolation, angle programming for linear interpolation		B, O
G02	G02	G02		Circular interpolation CW, (radius R designation)		B, O
G03	G03	G03		Circular interpolation CCW, (radius R designation)		B, O
G04	G04	G04	*	Dwell		B
G06	G06	G06		ERROR DETECT OFF positioning		B
G10	G10	G10		Tool offset value setup		O
G11	G11	G11	01	Beveling	Cornering	O
G12	G12	G12		Rounding		O
G20	G20	G70	05	Inch input specification		O
G21	G21	G71		Metric input specification		O
G22	G22	G22	01	Radius programming for circular interpolation CW		O
G23	G23	G23		Radius programming for circular interpolation CCW		O
G27	G27	G27	*	Reference point return check		B
G28	G28	G28		Automatic return to reference point		B
G29	G29	G29		Return from reference point		B
G30	G30	G30		Retunrn to 2nd reference point		O
G31	G31	G31		Skip function		O
G32	G33	G33	01	Threadcutting, continuous threadcuting, multi-start threadcutting		B, O
G34	G34	G34		Variable lead threadcutting		O
G35	G35	G35	*	Tool set error compensation		O
G36	G36	G36	07	Stored stroke limit 2nd area ON		O
G37	G37	G37		Stored stroke limit 2nd area OFF		O
G38	G38	G38	08	Stored stroke limit 3rd area ON		O
G39	G39	G39		Stored stroke limit 3rd area OFF		O
G40	G40	G40	06	Tool radius compensation cancel		O
G41	G41	G41		Tool radius compensation No. 1		O
G42	G42	G42		Tool radius compensation No. 2		O
G43	G43	G43		Tool radius compensation No. 3		O
G44	G44	G44		Tool radius compensation No. 4		O

Notes :

1. G codes in groups from 01 through 11 are modal. When the control is energized with the power switch or reset, the G codes marked with ◤ are automatically selected.

   For G00/G01, G98/G99, and G90/G91, either one is selected as initial state by setting parameters.

2. G codes of * group are non-modal. They should not be commanded together with the other G codes in one block.

3. The modal G codes can be commanded mixedly in a block.

4. G codes in section B are basic.

5. Standard G codes can be converted to special G codes I by parameters. (basic feature)

6. Special G code II can be selected as optional function. When selected, the standard G codes and special G code II cannot be used.

7. The initial states of G codes of 05, 07, 08 group when the control is powered correspond to their respective setting data.

B: Basic
O: Optional

Table 6.7 List of G Codes (Cont'd)

G Code	Special G Code I	Special G Code II	Group	Function		Section
G50	G92	G92	*	Coordinate system setup		B
				Maximum spindle revolution setup, work coordinate system setup		O
G51	G51	G51	*	Return of current display value to origin		O
G65	G65	G65		User macro simple call		O
G66	G66	G66	09	User macro modal call		O
◤ G67	◤ G67	◤ G67		User macro modal call cancel		O
G68	G68	G68	10	Mirror image by programming ON		O
◤ G69	◤ G69	◤ G69		Mirror image by programming OFF		O
G70	G70	G72	*	Finishing cycle	Multiple repetitive cycles	O
G71	G71	G73		Stock removal in turning		O
G72	G72	G74		Stock removal in facing		O
G73	G73	G75		Pattern repeating		O
G74	G74	G76		Peck drilling in Z-axis		O
G75	G75	G77		Grooving in X-axis		O
G76	G76	G78		Automatic threadcutting cycle		O
G90	G77	G20	01	Turning cycle A		B
G92	G78	G21		Threading cycle		B
G94	G79	G24		Facing cycle B		B
G96	G96	G96	02	Constant surface speed control		O
◤ G97	◤ G97	◤ G97		Constant surface speed control cancel		O
◤ G98	◤ G94	◤ G94	04	Feed per minute (mm/min)		B
◤ G99	◤ G95	◤ G95		Feed per revolution (mm/rev)		B
	◤ G90	◤ G90	03	Absolute command		B
	◤ G91	◤ G91		Incremental command		B
G122	G122	G122	11	Tool registration start	Tool life control	O
◤ G123	◤ G123	◤ G123		Tool registration end		O
G111	G111	G111	*	Taper multiple beveling/rounding		O
G112	G112	G112		Arc multiple beveling/rounding		O
G150	G150	G150	13	Cancel groove width compensation		O
G151	G151	G151		Groove width compensation		O

Notes :

1. G codes in groups from 01 through 11 are modal. When the control is energized with the power switch or reset, the G codes marked with ◤ are automatically selected.

   For G00/G01, G98/G99, and G90/G91, either one is selected as initial state by setting parameters.

2. G codes of * group are non-modal. They should not be commanded together with the other G codes in one block.

3. The modal G codes can be commanded mixedly in a block.

4. G codes in section B are basic.

5. Standard G codes can be converted to special G codes I by parameters. (basic feature)

6. Special G code II can be selected as optional function. When selected, the standard G codes and special G code II cannot be used.

7. The initial states of G codes of 05, 07, 08 group when the control is powered correspond to their respective setting data.

YASNAC LX3

CNC SYSTEM FOR TURNING APPLICATIONS

TOKYO OFFICE Ohtemachi Bldg, 1-6-1 Ohtemachi, Chiyoda-ku, Tokyo, 100 Japan
Phone (03) 3284-9111, -9145 Telex YASKAWA J33530 Fax (03) 3284-9034
SEOUL OFFICE Seoul Center Bldg, 91-1, So Kong-Dong, Chung-ku, Seoul, Korea
Phone (02) 776-7844 Fax (02) 753-2639
TAIPEI OFFICE Union Commercial Bldg, 14F, 137, Nanking East Road, Sec 2, Taipei, Taiwan
Phone (02) 507-7065, -7732 Fax (02) 506-3837
YASKAWA ELECTRIC AMERICA, INC. : SUBSIDIARY
Chicago-Technical Center
3160 MacArthur Blvd. Northbrook, Illinois 60062-1917, U.S.A.
Phone (708) 291-0411 Fax (708) 291-1028
Los Angeles Office
7341 Lincoln Way, Garden Grove, California 92641, U.S.A.
Phone (714) 894-5911 Telex (230) 678396 YASKAWAUS TSTN Fax (714) 894-3258
New Jersey Office 30 Two Bridges Road, Fairfield, New Jersey 07006, U.S.A.
Phone (201) 575-5940 Fax (201) 575-5947
YASKAWA ELECTRIC EUROPE GmbH : SUBSIDIARY
Monschauerstrasse 1, W 4000 Dusseldorf 11, Germany
Phone (0211) 950030 Telex (41) 8588673 YASD D Fax (0211) 507737
YASKAWA ELECTRIC (SINGAPORE) PTE. LTD.
CPF Bldg, 79 Robinson Road No. 13-05, Singapore 0106
Phone 2217530 Telex (87) 24890 YASKAWA RS Fax (65) 224-5854

YASKAWA ELECTRIC CORPORATION

YASKAWA

TOE-C843-9.20D
© Printed in Japan January 1993 86-5 1TA
586-176