

GE Fanuc Automation

Computer Numerical Control Products

PMC Model PA1/PA3/SA1/SA2/SA3/SA5/SB/SB2/ SB3/SB4/SB5/SB6/SC/SC3/SC4/NB/NB2/NB6 Ladder Language

Programming Manual (Volume 2 of 2)

GFZ-61863E/14

July 2001

Warnings, Cautions, and Notes as Used in this Publication

Warning

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

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

Caution

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

Note

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

This document is based on information available at the time of its publication. While efforts have been made to be accurate, the information contained herein does not purport to cover all details or variations in hardware or software, nor to provide for every possible contingency in connection with installation, operation, or maintenance. Features may be described herein which are not present in all hardware and software systems. GE Fanuc Automation assumes no obligation of notice to holders of this document with respect to changes subsequently made.

GE Fanuc Automation makes no representation or warranty, expressed, implied, or statutory with respect to, and assumes no responsibility for the accuracy, completeness, sufficiency, or usefulness of the information contained herein. No warranties of merchantability or fitness for purpose shall apply.

©Copyright 2001 GE Fanuc Automation North America, Inc. All Rights Reserved.

DEFINITION OF WARNING, CAUTION, AND NOTE

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

WARNING

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

CAUTION

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

NOTE

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

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

PREFACE

This programming manual describes the method of generating ladder sequence programs for PMC.

It also describes the operation methods of CRT/MDI and SYSTEM P series for sequence programming.

This manual presents programming descriptions for the PMC models listed in the following table. Note that some models have been renamed; in the product name column, the old names are enclosed in parentheses, while the new names appear above the old names. However, the previous specifications are still applied to the renamed models. Thus, when using the renamed models, users should:

- Read the old names shown in this manual as the new names.
- Read the old names appearing on the units as the new names.

Applicable models

Renaming of PMC

Models

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

Product Name	Abbreviations	Applicable CNC
FANUC PMC-MODEL PA1	PMC-PA1	FANUC Power Mate–MODEL D FANUC Series 21–MODEL A
FANUC PMC-MODEL PA3	PMC-PA3	FANUC Power Mate–MODEL D/F/H FANUC Series 21–MODEL A
FANUC PMC–MODEL SA1 (Note 1) (Old Name : FANUC PMC–MODEL RA1)	PMC-SA1 (PMC-RA1)	FANUC Series 18–MODEL A/B FANUC Series 20–MODEL A FANUC Series 21–MODEL B FANUC Series 20 <i>i</i> –MODEL A FANUC Series 0 <i>i</i> –MODEL A FANUC Series 21 <i>i</i> –MODEL A/B Loader control function (Note 2)
FANUC PMC-MODEL SA2 (Note 1) (Old Name : FANUC PMC-MODEL RA2)	PMC-SA2 (PMC-RA2)	FANUC Series 18–MODEL A
FANUC PMC-MODEL SA3 (Note 1) (Old Name : FANUC PMC-MODEL RA3)	PMC-SA3 (PMC-RA3)	FANUC Series 18–MODEL A FANUC Series 20–MODEL A FANUC Series 21–MODEL B FANUC Series 0 <i>i</i> –MODEL A
FANUC PMC-MODEL SA5 (Note 1) (Old Name : FANUC PMC-MODEL RA5)	PMC–SA5 (PMC–RA5)	FANUC Series 21 <i>i</i> -MODEL A
FANUC PMC-MODEL SB (Note 1) (Old Name : FANUC PMC-MODEL RB) FANUC PMC-MODEL SB2 (Note 1) (Old Name : FANUC PMC-MODEL RB2)	PMC-SB (PMC-RB) PMC-SB2 (PMC-RB2)	FANUC Series 16–MODEL A
FANUC PMC-MODEL SB3 (Note 1) (Old Name : FANUC PMC-MODEL RB3)	PMC–SB3 (PMC–RB3)	FANUC Series 16–MODEL A/B FANUC Series 18–MODEL B

Product Name	Abbreviations	Applicable CNC
FANUC PMC–MODEL SB4 (Note 1)	PMC-SB4	FANUC Series 16–MODEL B
(Old Name : FANUC PMC-MODEL RB4)	(PMC-RB4)	FANUC Series 18-MODEL B
FANUC PMC-MODEL SB5 (Note 1)	PMC-SB5	FANUC Series 16–MODEL C
(Old Name : FANUC PMC–MODEL RB5)	(PMC-RB5)	FANUC Series 18-MODEL C
		FANUC Series 16 <i>i</i> –MODEL A
		FANUC Series 18 <i>i</i> –MODEL A
		FANUC Power Mate <i>i</i> -MODEL D/H
FANUC PMC–MODEL SB6 (Note 1)	PMC-SB6	FANUC Series 16-MODEL C
(Old Name : FANUC PMC-MODEL RB6)	(PMC-RB6)	FANUC Series 18-MODEL C
		FANUC Series 16 <i>i</i> –MODEL A
		FANUC Series 18 <i>i</i> –MODEL A
		FANUC Series 21 <i>i</i> -MODEL A
		FANUC Power Mate <i>i</i> -MODEL D/H
FANUC PMC-MODEL SB7	PMC-SB7	FANUC Series 16 <i>i</i> -MODEL B
		FANUC Series 18 <i>i</i> -MODEL B
		FANUC Series 21 <i>i</i> -MODEL B
FANUC PMC-MODEL SC (Note 1)	PMC-SC	FANUC Series 16–MODEL A
(Old Name : FANUC PMC-MODEL RC)	(PMC-RC)	
FANUC PMC-MODEL SC3 (Note 1)	PMC-SC3	FANUC Series 16–MODEL A/B/C
(Old Name : FANUC PMC-MODEL RC3)	(PMC-RC3)	FANUC Series 18–MODEL B/C
FANUC PMC-MODEL SC4 (Note 1)	PMC-SC4	FANUC Series 16–MODEL B/C
(Old Name : FANUC PMC-MODEL RC4)	(PMC-RC4)	FANUC Series 18–MODEL B/C
FANUC PMC-MODEL NB	PMC-NB	FANUC Series 15-MODEL B
FANUC PMC-MODEL NB2	PMC-NB2	
FANUC PMC-MODEL NB6	PMC-NB6	FANUC Series 15 <i>i</i> -MODEL A

NOTE

1 These models have been renamed; in the product name column, the old names are enclosed in parentheses, while the new names appear above the old names. However, the previous specifications are still applied to the renamed models.

Thus, when using the renamed models, users should :

- Read the old names shown in this manual as the new names.
- Read the old names appearing on the units as the new names.
- 2 PMC–SA1 is applied to the loader control side of a CNC having the loader control function.

The CNC models having the loader control function are as follows :

FANUC Series 16–MODEL A/B/C

FANUC Series 18–MODEL A/B/C

FANUC Series 21–MODEL B

FANUC Series 16*i*/18*i*/21*i*-MODEL A/B

Other manuals

However, it does not include all items required for sequence programming. For those required for sequence programming refer to the following manuals.

Name of manual	Reference items	Application
FANUC Power Mate-MODEL D/F CONNECTION MANUAL (B-62833EN)	Interface between PMC and CNC	PMC-PA1 PMC-PA3
FANUC Power Mate-MODEL H CONNECTION MANUAL (B–62683EN)	Interface between PMC and CNC	PMC-PA3
FANUC Power Mate <i>i</i> -MODEL D/H CONNECTION MANUAL (FUNCTION) (B–63733EN–1)	Interface between PMC and CNC	PMC-SB5 PMC-SB6
FANUC Series 16/18 CONNECTION MANUAL (B-61803E)	Interface between PMC and CNC	PMC-SA1 PMC-SA2 PMC-SA3 PMC-SB PMC-SB2 PMC-SB3 PMC-SC PMC-SC3
FANUC Series 16/18/160/180-MODEL B CONNECTION MANUAL (FUNCTION) (B-62443E-1)	Interface between PMC and CNC	PMC-SB3 PMC-SB4 PMC-SC3 PMC-SC4
FANUC Series 16/18/160/180-MODEL C CONNECTION MANUAL (FUNCTION) (B-62753EN-1)	Interface between PMC and CNC	PMC-SB5 PMC-SB6 PMC-SC3 PMC-SC4
FANUC Series 16i/18i/21i/160i/180i/210i–MODEL A CONNECTION MANUAL (FUNCTION) (B-63003EN-1)	Interface between PMC and CNC	PMC-SA1 PMC-SA5 PMC-SB5 PMC-SB6
FANUC Series 16i/18i/21i/160i/180i/210i–MODEL B CONNECTION MANUAL (FUNCTION) (B-63523EN-1)	Interface between PMC and CNC	PMC-SB7
FANUC Series 20–FA/TA CONNECTION MANUAL (B–62173E)	Interface between PMC and CNC	PMC-SA1 PMC-SA3
FANUC Series 21/210–MODEL B CONNECTION MANUAL (FUNCTION) (B–62703EN–1)	Interface between PMC and CNC	PMC-SA1 PMC-SA3
FANUC Series 15-MODEL B BMI INTERFACE CONNECTION MANUAL (B-62073E-1)	Interface between PMC and CNC	PMC-NB PMC-NB2
FANUC Series 15i/150i-MODEL A CONNECTION MANUAL (FUNCTION) (B–63323EN–1)	Interface between PMC and CNC	PMC-NB6
FANUC PMC C LANGUAGE PROGRAMMING MANUAL (B-61863E-1)	C language programming	PMC-SC PMC-SC3 PMC-SC4 PMC-NB PMC-NB2

Other application model names

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

Product name	Abbreviation	CNC for
FANUC PMC-MODEL P	PMC-P	FANUC Power Mate-MODEL C
FANUC PMC-MODEL NA	PMC-NA	FANUC Series 15-MODEL B

Volume 1

DEFINITION OF WARNING, CAUTION, AND NOTE	s–1
PREFACE	p–I

I. PMC SEQUENCE PROGRAM

1. S	EQUEN	ICE F	PROGRAM CREATING PROCEDURE	. 3
	1.1	SPE	CIFICATION OF PMCs	. 5
	1.2	SUM	IMARY OF SPECIFICATION OF LADDER PROGRAM	28
	1.3	WHA	AT IS A SEQUENCE PROGRAM?	35
	1.4	CRE	ATION OF INTERFACE SPECIFICATIONS (STEPS 1 TO 3)	37
	1.5	CRE	ATION OF LADDER DIAGRAM (STEP 4)	37
	1.6	COD	DING (STEP 5)	38
	1.7	SEQ	UENCE PROGRAM ENTRY (STEPS 6, 7)	39
	1.8	SEQ	UENCE PROGRAM CHECK AND WRITE INTO ROM (STEPS 8 TO 11)	39
	1.9	STO	RAGE AND CONTROL OF SEQUENCE PROGRAM (STEPS 12 TO 14)	40
2. S	EQUEN	ICE F	PROGRAM	41
	2.1	EXE	CUTION PROCEDURE OF SEQUENCE PROGRAM	42
	2.2	REP	ETITIVE OPERATION	43
	2.3	PRIC	ORITY OF EXECUTION (1ST LEVEL, 2ND LEVEL AND 3RD LEVEL)	44
	2.4	SEO	UENCE PROGRAM STRUCTURING	48
	2.5	PRO	CESSING I/O SIGNALS	50
	2.	5.1	Input Signal Processing	51
	2	5.2	Output Signal Processing	51
	2	5.3	I/O Signals to CNC	52
	2.	5.4	Difference of Status of Signals between 1st Level and 2nd Level	53
	2.6	INTI	ERLOCKING	54
	2.7	SEQ	UENCE PROGRAM PROCESSING TIME	55
	2.8	SEQ	UENCE PROGRAM MEMORY CAPACITY	57
3Δ	DDRES	ss		59
0.71				
	3.1	ADL	DRESSES BETWEEN PMC AND UNC (PMC↔NC)	74
	3.2	ADL	PRESSES BETWEEN PMC AND MACHINE TOOL (PMC↔MT)	75
	3.	2.1	Addresses Between PMC and Machine Tool	75
	3.	2.2	Assignment of I/O Module Addresses	77
	3.	2.3	I/O Link Connection Unit Assignment	84
	3.	2.4	I/O Unit MODEL B Assignment	86
	3.	2.5	Power Mate Assignment	87
	3.	2.6	FS16 <i>i</i> –LA Assignment	88
	3.	2.7	Distribution I/O Connection I/O Module and Distribution I/O Operator's Panel I/O Module Assignment Methods	89

	3.3 IN	VTERNAL RELAY ADDRESSES (R)	95
	3.3.1	Area Managed by the System Program	97
	3.4 A	DDRESSES FOR MESSAGE SELECTION DISPLAYED ON CRT (A)	. 102
	3.5 A	DDRESS OF COUNTER (C)	. 104
	3.6 A	DDRESS OF KEEP RELAY AND NONVOLATILE MEMORY CONTROL (K)	. 106
	3.7 A	DDRESS OF DATA TABLE (D)	. 108
	3.8 T	IMER ADDRESSES (T)	. 110
	3.9 L	ABEL ADDRESSES (JMPB, JMPC, LBL) (L)	. 112
	3.10 SI	UBPROGRAM NUMBERS (CALL, CALLU, SP) (P)	. 113
4 P		INSTRUCTIONS	114
	41 D	ETAILS OF BASIC INSTRUCTIONS	116
	4.1.1	RD	. 118
	4.1.2	RD NOT	. 119
	4.1.3	WBT	. 120
	4.1.4	WBT NOT	. 121
	4.1.5	AND	. 122
	4.1.6	AND. NOT	. 122
	4.1.7	OR	. 122
	4.1.8	OR NOT	. 122
	4.1.9	RD STK	. 123
	4.1.10	RD. NOT. STK	. 124
	4.1.11	AND. STK	. 125
	4.1.12	OR. STK	. 125
	4.1.13	SET	. 126
	4.1.14	RST	. 128
5 5			120
J. I			157
	5.1 El	NDI (ISI LEVEL SEQUENCE PROGRAM END)	. 15/
	5.2 EI	$ND2 (2ND LEVEL SEQUENCE PROGRAM END) \dots D2 (2ND LEVEL SEQUENCE) (DMC SC/SC2/SC4/ND ND2/ND4/ND7 ONLY)$. 158
	5.5 E	ND3 (END OF 3RD LEVEL SEQUENCE) (PMC-SC/SC3/SC4/NB/NB2/NB0/NB7 ONLY) .	. 159
	5.4 II	MR (IIMER)	. 100
	5.5 II	MRB (FIAED TIMER)	. 102
	5.0 II	MRC (TIMER)	. 104
	5.7 D	$ECP (PECODE) \dots ECODING$. 10/
	5.0 C	TP (COUNTED)	. 109
	5.10 C	TRB (Fixed counter)	. 172
	5.10 C	TRC (COUNTER)	180
	5.12 R	OT (ROTATION CONTROL)	183
	5.12 R	OTB (BINARY ROTATION CONTROL)	186
	5.14 C	OD (CODE CONVERSION)	. 189
	5.15 C	ODB (BINARY CODE CONVERSION)	. 192
	5.16 M	IOVE (LOGICAL PRODUCT TRANSFER)	. 194
	5.17 M	OVOR (DATA TRANSFER AFTER LOGICAL SUM)	. 196
	5.18 C	OM (COMMON LINE CONTROL)	. 197
		· · · · · · · · · · · · · · · · · · ·	
	5.19 C	OME (COMMON LINE CONTROL END)	. 203

5.20	JMP (JUMP)	204
5.21	JMPE (JUMP END)	209
5.22	PARI (PARITY CHECK)	210
5.23	DCNV (DATA CONVERSION)	212
5.24	DCNVB (EXTENDED DATA CONVERSION)	214
5.25	COMP (COMPARISON)	216
5.26	COMPB (COMPARISON BETWEEN BINARY DATA)	218
5.27	COIN (COINCIDENCE CHECK)	220
5.28	SFT (SHIFT REGISTER)	222
5.29	DSCH (DATA SEARCH)	224
5.30	DSCHB (BINARY DATA SEARCH)	227
5.31	XMOV (INDEXED DATA TRANSFER)	229
5.32	XMOVB (BINARY INDEX MODIFIER DATA TRANSFER)	232
5.33	ADD (ADDITION)	239
5.34	ADDB (BINARY ADDITION)	241
5.35	SUB (SUBTRACTION)	243
5.36	SUBB (BINARY SUBTRACTION)	245
5.37	MUL (MULTIPLICATION)	247
5.38	MULB (BINARY MULTIPLICATION)	249
5.39	DIV (DIVISION)	251
5.40	DIVB (BINARY DIVISION)	253
5.41	NUME (DEFINITION OF CONSTANT)	255
5.42	NUMEB (DEFINITION OF BINARY CONSTANTS)	256
5.43	DISP (MESSAGE DISPLAY) (PMC–SB/SB2/SB3/SB4/SB5/SB6/SC/SC3/SC4 ONLY)	258
5.44	DISPB	272
5.45	EXIN (EXTERNAL DATA INPUT)	283
5.46	WINDR (READING CNC WINDOW DATA)	288
5.47	WINDW (WRITING CNC WINDOW DATA)	291
5.48	ARBITRARY FUNCTIONAL INSTRUCTIONS	294
5.49	MMCWR (READING MMC WINDOW DATA) (OTHER THAN PMC–PA1/PA3)	297
5.50	MMCWW (WRITING MMC WINDOW DATA) (OTHER THAN PMC–PA1/PA3)	299
5.51	MOVB (TRANSFER OF 1 BYTE)	301
5.52	MOVW (TRANSFER OF 2 BYTES)	302
5.53	MOVN (TRANSFER OF AN ARBITRARY NUMBER OF BYTES)	303
5.54	MOVD (TRANSFER OF 4 BYTES)	304
5.55 5.50	DIFU (RISING EDGE DETECTION)	305
5.50	DIFD (FALLING EDGE DETECTION)	306
5.51		200
5.50		309
5.59		212
5.60	MMC3 P (MMC II WINDOW DATA PEAD)	315
5.67	MMC3W (MMC_III WINDOW DATA WRITE)	313
5.62	SPCNT (SPINDI E CONTROL)	321
5.05 5.64	FND (FND OF A LADDER PROGRAM)	321
5.65	CALL (CONDITIONAL SUBPROGRAM CALL)	329
5.05		550
5 66	CALLU (UNCONDITIONAL SUBPROGRAM CALL)	331

	5.67	SP (SUBPROGRAM)	332
	5.68	SPE (END OF A SUBPROGRAM)	333
	5.69	JMPB (LABEL JUMP)	334
	5.70	JMPC (LABEL JUMP)	335
	5.71	LBL (LABEL)	336
	5.72	AXCTL (AXIS CONTROL BY PMC)	337
	5.73	PSGNL (POSITION SIGNAL OUTPUT)	343
	5.74	PSGN2 (POSITION SIGNAL OUTPUT 2)	347
	5.75	NOP (NO OPERATION)	351
6. N	ONVOL	ATILE MEMORY	353
	6.1	TIMER, COUNTER, KEEP RELAY, NONVOLATILE MEMORY CONTROL, DATA TABLE	. 354
	6.2	READING AND WRITING OF NONVOLATILE MEMORY DATA	356
	6.3	PMC DATA TABLE	357
7. L	ADDER	DIAGRAM FORMAT	362
	7.1	ADDRESSES, SIGNAL NAMES, COMMENTS, AND LINE NUMBERS	363
	7.1	.1 Addresses	363
	7.1	.2 Signal Names	364
	7.1	.3 Comments	364
	7.1	.4 Line Numbers	364
	7.2	SYMBOLS USED IN THE LADDER DIAGRAM	365
	7.3	LADDER DIAGRAM FORMAT	366
	7.4	INFINITE NUMBER OF RELAY CONTACTS	367
8. N	IISCELL	ANEOUS ITEM	368
9. S		CE PROGRAM STRUCTURING	369
	9.1	EXAMPLES OF STRUCTURED PROGRAMMING	370
	9.1	.1 Implementation Techniques	370
	9.1	.2 Applications	371
	9.1	.3 Specifications	373
	9.2	SUBPROGRAMMING AND NESTING	374
	9.2	.1 Function	374
	9.2	.2 Execution Method	375
	9.2	.3 Creating a Program	376
	9.3	CAUTIONS FOR SUBROUTINES	378
10.	JMP INS	TRUCTIONS WITH LABEL SPECIFICATION	380
	10.1	SPECIFICATIONS	381
	10.2	RESTRICTIONS	385
11.	INTERR	UPT-TYPE PMC FUNCTION	391
	11 1	OVERVIEW	302
	11.1	SETTING	393
	·		

11.3 IN	TERRUPT PROCESSING	394
11.3.1	Interrupt Program	394
11.3.2	Input/Output Signal Processing	395
11.3.3	Response Time	395
11.3.4	Execution Time	396
11.3.5	Interrupt Enable/Disable/Clear	396
11.3.6	Cautions	398
11.4 SE	QUENCE PROGRAM EXAMPLES	400

II. PMC OPERATION (CRT/MDI)

1. G	ENERAL		405
	1.1 FO	OR MDI UNITS OTHER THAN STANDARD MDI UNITS	
	(F	FOR FS20 PMC–SA1 AND SA3)	412
	1.2 A	UTOMATIC OPERATION WHEN THE POWER IS TURNED ON	412
	1.3 C	LEARING THE SEQUENCE PROGRAM	412
	1.4 LO	OADING THE STANDARD LADDER (FOR Power Mate-D/F PMC-PA1 AND P	A3) 413
	1.5 FS	S15 <i>i</i> PMC–NB6 OPERATING PROCEDURE	413
	1.6 FS	S16 <i>i</i> /18 <i>i</i> /21 <i>i</i> –B PMC–SA1/SB7 OPERATING PROCEDURE	413
	1.7 L	ADDER PASSWORD FUNCTION	414
	1.8 PI	MC OPERATION FOR LOADER CONTROL FUNCTION	417
2. P	MC MENU	SELECTION PROCEDURE BY SOFTKEY	418
3. P	MC I/O SIG	GNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) 424
	3.1 D	ISPLAYING TITLE DATA	425
	3.2 D	ISPLAY OF SIGNAL STATUS (STATUS)	
	3.3 A	LARM SCREEN (ALARM)	428
	3.4 TI	RACE FUNCTION (TRACE)	429
	3.4.1	Operation	
	3.4.2	Parameter Setting Screen	
	3.4.3	Starting or Stopping the Trace Function	430
	3.4.4	Trace Screen	
	3.4.5	Automatic Tracing Function at Power on	
	3.5 D	ISPLAYING THE CONTENTS OF MEMORY	432
	3.5.1	Operation	432
	3.5.2	Function for Storing Data in Memory	433
	3.6 FU	UNCTION FOR DISPLAYING SIGNAL WAVEFORMS (ANALYS)	434
	3.6.1	Specifications	434
	3.6.2	Operation	
	3.6.3	Parameter Screen	
	3.6.4	Signal Diagnosis Screen	438
	3.6.5	Reading Signals Automatically at Power on	439
	3.7 D	ISPLAYING THE RUNNING STATE OF A USER TASK (USRDGN)	
	3.8 D	ISPLAYING AND SETTING THE CONFIGURATION STATUS OF I/O	
	D	EVICES (IOCHK)	442

Table of contents

	3.8	8.1	I/O Link Connecting Check Screen	444
	3.8	8.2	I/O Link–II Parameter Setting Screen	446
	3.9	FOR	CED INPUT/OUTPUT FUNCTION	449
	3.9	9.1	Overview	449
	3.9	9.2	Setting/Operation for Enabling Forced Input/Output	453
	3.9	9.3	Screen Display	454
	3.9	9.4	Modifying the Values of Signals by Forced Input/Output	457
	3.9	9.5	Setting/Clearing OVERRIDE	458
4. PM	C PAI	RAM	ETERS SETTING AND DISPLAY (PMCPRM)	460
	4.1	OUT	LINE	461
	4.2	INPU	UT PMC PARAMETERS FROM MDI PANEL	462
	4.2	2.1	Multiple Data Input	462
	4.3	SET	TING AND DISPLAY SCREEN	463
	4.3	3.1	Timer Screen (TIMER)	463
	4.3	3.2	Counter Screen (COUNTR)	464
	4.3	3.3	Keep Relay (KEEPRL)	464
	4.3	3.4	Data Table (DATA)	470
	4.4	SET	TING SCREEN	472
	4.4	4.1	Other Setting Screens	475
	4.5	NOT	Έ	478
5. PM	C LAI	DDEF	R DIAGRAM DISPLAY (PMCLAD)	479
	5.1	LAD	DER DIAGRAM DISPLAY	480
	5.2	DUN	/IP DISPLAY ON LADDER DIAGRAM	481
	5.3	PAR	AMETER DISPLAY ON LADDER DIAGRAM	482
	5.3	3.1	The Value of Functional Instruction Parameter	483
	5.4	SYM	IBOL AND COMMENT DISPLAY	485
	5.5	SEA	RCH OF SPECIFIED RELAY COIL POINTS IN LADDER DIAGRAM	486
	5.6	STO	P OF LADDER DIAGRAM DISPLAY BY TRIGGER OF SIGNAL	487
	5.7	DIV	IDING DISPLAY OF LADDER DIAGRAM	489
	5.8	ON-	LINE EDIT	490
6. USE	er pi	NC S	CREEN (PMCMDI)	492
	6.1	FOR	THE FS16 (PMC–SC OR PMC–SC3)	493
	6.2	FOR	. THE FS15 (PMC–NB)	493

III. PMC PROGRAMMER (CRT/MDI)

1. GE	NERA	L	497
	1.1	ABOUT THE FS15 <i>i</i> PMC–NB6 PROGRAMMER	498
	1.2	ABOUT THE FS16i/18i/21i-B PMC-SA1/SB7 PROGRAMMER	498
2. CC	MPON	NENT UNITS AND CONNECTIONS	499
	2.1	COMPONENT UNITS	500

2.2 CC	DNNECTING COMPONENT UNITS	505
3. SELECTION	OF PROGRAMMER MENUS BY SOFTKEYS	506
4. SPECIFYING	G AND DISPLAYING SYSTEM PARAMETERS (SYSPRM)	508
5. EDITING OF	SEQUENCE PROGRAM (EDIT)	515
5.1 SP	ECIFYING AND DISPLAYING TITLE DATA (TITLE)	516
5.1.1	Entering Title Data	516
5.1.2	Deleting Title Data	516
5.1.3	Editing Character Strings of Title Data	517
5.2 SE	QUENCE PROGRAM GENERATION (LADDER)	519
5.2.1	Sequence Program Input	520
5.2.2	Alteration of Sequence Programs	524
5.2.3	Insert of Sequence Program	524
5.2.4	Delete of Sequence Program	527
5.2.5	Search of Sequence Program	528
5.2.6	Copying the Sequence Program	530
5.2.7	Moving the Sequence Program	531
5.2.8	Editing Symbol Data and Comment at Once	532
5.2.9	Address Change of Sequence Program	532
5.3 I/C	UNIT ADDRESS SETTING (MODULE)	534
5.4 SY	MBOL DATA SETTING (SYMBOL)	537
5.4.1	Symbol Data and Comment Input	538
5.4.2	Symbol Data Search (SRCH)	538
5.4.3	Delete of Symbol Data and Comment	538
5.4.4	Editing Character Strings of Symbol Data and Comment Data	539
5.4.5	Function for Editing Symbol Data and Comment Data at One Time	539
5.4.6	Function of Copying Symbol and Comment Data	540
5.5 MI	ESSAGE DATA SETTING (MESSAGE)	541
5.5.1	Message Data Input	542
5.5.2	Searching for an Address (SRCH)	542
5.5.3	Editing a Character String in Message Data	
5.5.4	Input with a Katakana Identification Code	543
5.5.5	Copying Message Data (COPY)	543
5.5.6	Inputting a Multi-byte Character (D.CHAR)	543
5.5.7		543
5.6 CL TH	EARING THE SEQUENCE PROGRAM AND CONDENSATION OF	544
5.6.1	Clearing the Sequence Program	544
5.6.2	Compress the Sequence Program	545
5.6.3	Clearing the PMC Parameter	545
5.7 CR	OSS REFERENCE DISPLAY	546
5.7.1	Operation	546
5.7.2	Parameter Setting Screen	547
6. EXECUTION	OF A SEQUENCE PROGRAM	551
6.1 ST	ART AND STOP OF A SEQUENCE PROGRAM	552

6.2	STA	ARTING THE SEQUENCE PROGRAM	553
6.3	FOF	RCIBLY STOPPING THE SEQUENCE PROGRAM	553
WRITING	G, RE	EADING, AND VERIFYING THE SEQUENCE PROGRAM	
			554
7.1		ERVIEW	555
7.	1.1		
7.2	SEI	TIEMS	
1.3	0Pf 2 1	Transfer to and from a EAPT LADDED	
7.	3.1 2.2	Transfer to and from a FANIC ED Cospette	
7.	3.2 2.2	Storage to Eleck DOM	
7.	3.3 2.4	Storage to a Mamoria Cond	
7.	3.4 2.5	Deta Input to and Output from other Daviage	
7.	3.3 2.6	Setting the Transfer Speed (ISPEED) Soft Key)	
7.	3.0 2 7	Transfer to and from a DOM WDITED	
7.	5.1 20	Notes on Using on MDI Keyboard without Current Vaya	
7.	3.0	(when using the FS20 PMC–MODEL SA1/SA3)	572
7.4	I/O	ERROR MESSAGES	573
7.5	SEC	DUENCE PROGRAM COPY FUNCTION	
7.	5.1	Copy Title Data [CPYTTL]	
7.	5.2	Copy a Ladder Program [CPYLAD]	575
7.	5.3	Copy Symbol Data and Comment Data [CPYSYM]	
7.	5.4	Copy Message Data [CPYMSG]	
7.	5.5	Copy the Sequence Programs [CPYALL]	
7.	5.6	Copy I/O Module Data [CPYMDL]	576
7.6	RES	STRICTIONS	
	ONS	FOR DISPLATING MEMORY AREAS AND DEBUGGING	
THE PR	ROGF	RAM (MONIT)	577
8.1	DIS	PLAYING THE GDT (GLOBAL DESCRIPTOR TABLE)	578
8.	1.1	Operation	578
8.	1.2	Descriptions of Displayed Items	
8.2	DIS	PLAYING THE MEMORY ALLOCATION INFORMATION OF A USER	
	PRO	OGRAM CODED IN C	581
8.	2.1	Operation	581
8.	2.2	Displayed Items	583
8.3	DEI	BUGGING	584
8.	3.1	Specifications	584
8.	3.2	Operation	584
8.	3.3	Parameter Screen	585
8.	3.4	Screen for Displaying Traced Data	588
8.	3.5	Enabling Automatic Debugging at Power-on	588
8.	3.6	Notes	588
8.4	LAI	DDER DEBUGGING FUNCTION	589
8.	4.1	Screen of Ladder Debugging Function	589
8.	4.2	Soft Key Menu of Ladder Debugging Function	590

Soft Key Menu of Ladder Debugging Function	590

	8.4.3	Step Operation [STEP]	591
	8.4.4	Stop Function of Break with Condition [BRKCTL]	592
8.5	5 (ONLINE FUNCTION	594
	8.5.1	Online Setting Screen	595
	8.5.2	Setting of online connection	595
	8.5.3	Online Function by Ethernet	599
	8.5.4	Communication Status	602
	8.5.5	About Connection Log of Ethernet	603
9. ERRO 10. ERR	r Mi Or N	ESSAGES (FOR EDIT)	605 607
11. PMC	PRC	OGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate)	608
11	.1 \$	SELECTING THE PMC PROGRAMMER MENU	610
11	.2 5	SETTING AND DISPLAYING SYSTEM PARAMETERS (SYSTEM PARAM)	611
11	.3 I	EDITING THE SEQUENCE PROGRAM (EDIT)	612
11	.4 I	EDITING LADDER MNEMONICS	613
	11.4.	1 Starting Ladder Mnemonics Editing	613
	11.4.	2 Confirming the Ladder Mnemonics	614
	11.4.	3 Modifying the Ladder Mnemonics	616
	11.4.	4 Ending Ladder Mnemonics Editing	618
11	.5 \$	STARTING AND STOPPING THE SEQUENCE PROGRAM (RUN/STOP)	619
11	.6 I	ERROR MESSAGES (FOR LADDER MNEMONICS EDITING)	620
11	.7 S	STORING THE SEQUENCE PROGRAM INTO FLASH EEPROM (I/O) ONLY FOR THE Power Mate–H AND Power Mate <i>i</i> –D/H)	621
11	.8 E	ERROR DETAILS	622
11	.9 I	NPUT/OUTPUT LADDER/PMC–PARAMETER BY MDI/DPL	623
	11.9.	1 Input/Output Method to Office Programmer (P-g Mate/Mark II) (Fixed 9600bit/Sec.)	623
	11.9.	2 Input/Output Method to FANUC FLOPPY CASSETTE (Fixed 4800bit/Sec.)	623
11	.10 (DN-LINE DEBUGGING FUNCTION (ONLY FOR Power Mate–D/H and Power Mate i –D/H).	624
	11.10).1 Starting and Stopping the On-line Debugging Function	624
11	.11 F	ERROR LIST	626

IV. PMC-NB6 MANIPULATION

1. OVERV	IEW	629
2. SOFT K	XEY-BASED PMC MENU SELECTION PROCEDURE	630
2.1	PMC BASIC MENU	
2.2	PMC SCREEN TRANSITION AND RELATED SOFT KEYS	633
3. DISPLA INTER	YING PMC INPUT/OUTPUT SIGNALS AND NAL RELAY (PMCDGN)	635
3.1	TITLE DATA DISPLAY (TITLE)	636

	3.2	TI	TLE DATA EDITOR SCREEN	637
	3.3	SI	GNAL STATUS DISPLAY (STATUS)	638
	3.4	AI	ARM SCREEN (ALARM)	639
	3.5	TF	ACE SCREEN	640
		3.5.1	Overview	640
		3.5.2	Setting of Trace Parameter	641
		3.5.3	Execution of Trace	647
		3.5.4	Operation after Execution of Trace	648
		3.5.5	Comparison With the Function of FS15–B (PMC–NB/NB2)	650
	3.6	I/C) LINK MONITOR AND ALLOCATION OF I/O UNIT	654
		3.6.1	I/O Link Monitor screen	655
		3.6.2	I/O Unit Allocation Viewer screen	657
		3.6.3	I/O Unit Allocation Editor screen	658
4 PM	СГ	ΔΤΔ	SETTING AND DISPLAY (PMCPRM)	662
	4 1	01		
	4.1			663
	4.2	M.	ETHOD FOR ENTERING PMC PARAMETERS	664
	4.3		DNTINUOUS DATA ENTRY	665
	4.4		MER SCREEN (TIMER)	666
	4.5	C	DUNTER SCREEN (COUNTR)	667
	4.6	KI	EEP RELAY SCREEN (KEEPRL)	668
	4.7	DA	ATA TABLE (DATA)	670
5. SE	TTI	NG M	ENU (SETING)	. 672
	5.1	SC	REEN FOR DISPLAYING GENERAL SETTINGS (GENERAL)	673
	5.2	SC	REEN FOR DISPLAYING THE SETTING DATA RELATED TO	
		EI	DITING AND DEBUGGING	674
	5.3	SC	TREEN FOR SETTING/DISPLAYING ONLINE MONITOR PARAMETERS (ONLINE)	675
	5.4	SC	CREEN FOR DISPLAYING SETTING DATA RELATED TO C LANGUAGE FUNCTION .	677
6. LAI	DDI	ER DI	AGRAM MONITOR AND EDITOR	. 678
	6.1	LA	ADDER DIAGRAM MONITOR SCREEN	681
		6.1.1	Screen Structures	682
		6.1.2	Operations	684
		6.1.3	Functional Instruction Data Table Viewer Screen	688
		6.1.4	Setting Screen	691
		6.1.5	Display Format for Parameters	698
		6.1.6	Functional Instructions of Special Monitor for Mat	701
	6.2	LA	ADDER DIAGRAM EDITOR SCREEN	702
		6.2.1	Screen Structures	704
		6.2.2	Operations	705
		6.2.3	Setting Screen	710
		6.2.4	Error Messages and Their Solutions	715
		6.2.5	How to Correct LADDER Program that Never Stops	720
	6.3	NI	ET EDITOR SCREEN	721
		6.3.1	Screen Structures	722
		632	Operations	723

	6.3.3	Functional Instruction List Screen	727
	6.3.4	Functional Instruction Data Table Editor Screen	728
	6.3.5	Error Messages and Their Solutions	732
	6.3.6	Structure of Valid Net	734
	6.4 PRO	GRAM LIST VIEWER SCREEN	735
	6.4.1	Screen Structures	736
	6.4.2	Operations	739
	6.4.3	Setting Screen	740
	6.5 PRO	GRAM LIST EDITOR SCREEN	742
	6.5.1	Screen Structures	743
	6.5.2	Operations	745
	6.5.3	Setting Screen	746
	6.6 COL	LECTIVE MONITOR FUNCTION	747
	6.6.1	Structures of Collective Monitor Screen	748
	6.6.2	Operations of Collective Monitor Screen	749
	6.6.3	How to Pick Up a Ladder Diagram Net	750
7.	PASSWORD	PROTECTION	752
	7.1 SPE	CIFICATIONS OF PASSWORD	753
	7.2 PAS	SWORD AND SWITCHING SCREEN	754
	7.2.1	Using One of R-Password and RW-Password	754
	7.2.2	Using Both of R-Password and RW-Password	755
8.	WRITING, RE	ADING, AND COMPARING SEQUENCE PROGRAMS	
	AND PMC P	ARAMETERS	756
	8.1 I/O S	SCREEN	757
	8.2 OUT	PUTTING TO AND INPUTTING FROM MEMORY CARDS	760
	8.3 MEN	MORY CARD LIST SCREEN	764
	8.4 OUT	PUTTING TO AND INPUTTING FROM FLASH ROM	766
	8.5 OUT	PUTTING TO AND INPUTTING FROM FLOPPY	769
	8.6 FLO	PPY LIST SCREEN	773
	8.7 OUT	TPUTTING TO AND INPUTTING FROM OTHER INPUT/OUTPUT DEVICES \dots	774
	8.8 POR	T SETTING SCREEN	777
	8.9 I/O S	SCREEN ERROR MESSAGES (PMC–NB6)	779
9.	STARTING A	ND STOPPING SEQUENCE PROGRAMS	783

V. FS16i/18i/21i-B PMC-SA1/SB7 MANIPULATION

1. SCREEN OP	ERATION FOR PMC–SA1/SB7 7	787
1.1 SCI	REEN OPERATION FOR PMC-SA1	788
1.1.1	Structure of Software Key in PMC	788
1.1.2	Addition of System Keep Relays	788
1.2 SCI	REEN OPERATION FOR PMC–SB7	789
1.2.1	Structure of Software Key in PMC	789

Table of contents

	1.2	2.2 Expansion of Symbol Size	790
	1.2	2.3 Addition of System Keep Relays	790
2. SE	TTING	FUNCTION	791
	2.1	SWITCHING SCREENS FOR SETTING PARAMETERS	792
	2.2	SETTING SCREEN OF GENERAL FUNCTION	793
	2.3	CAUTION SCREEN OF THE SETTING PARAMETER FOR THE SELECTABLE I/O LINK ASSIGNMENT FUNCTION	795
	2.4	SETTING PARAMETER SCREEN FOR THE SELECTABLE I/O LINK ASSIGNMENT FUNCTION	796
	2.5	PROGRAMMER PROTECTION FUNCTION	797
	2.5	5.1 PROGRAMMER ENABLE (PMC–SB7: K900.1, PMC–SA1: K17.1)	798
	2.5	5.2 HIDE PMC PROGRAM (PMC–SB7: K900.0, PMC–SA1: K17.0)	798
	2.5	5.3 EDIT ENABLE (PMC–SB7: K901.6, PMC–SA1: K18.6)	799
	2.5	5.4 ALLOW PMC STOP (PMC–SB7: K902.2, PMC–SA1: K19.2)	799
	2.5	5.5 RAM WRITE ENABLE (PMC–SB7: K900.4, PMC–SA1: K17.4)	800
	2.5	DATA TBL CNTL SCREEN (PMC–SB7: K900.7, PMC–SA1: K17.7)	800
	2.5	5.7 IO GROUP SELECTION (K906.1)	800
	2.6	EXAMPLE FOR SETTING PARAMETERS	801
3 1 4			803
J. LA			
	3.1	LADDER DIAGRAM MONITOR SCREEN	805
	3.1	1.1 Screen Structures	806
	3.1	1.2 Operations	808
	3.1	1.3 Setting the Screen	812
	3.1	1.4 Display Format for Parameters	818
	3.2	FUNCTIONAL INSTRUCTION DATA TABLE VIEWER SCREEN	
	3.2	2.1 Screen Structures	
	3.2	2.2 Operations	
	3.3	PROGRAM LIST VIEWER SCREEN	824
	3.3 2.2	2.2 Drief Sereen	824 926
	3.3 2.2	3.2 Bhei Scieeli	820
	3.3 2.2	3.5 Operations	
	5.5	5.4 Setting Screen	020
4. CC	DLLEC	TIVE MONITOR FUNCTION (PMC–SB7)	831
	4.1	CALLING SCREEN	832
	4.2	SCREEN CONFIGURATION	833
	4.3	SCREEN OPERATIONS	834
	4.4	ALARM MESSAGE AND COUNTERMEASURE	837
5. LA	DDER	DIAGRAM EDITOR FUNCTION (PMC–SB7)	838
	5 1	LADDER DIAGRAM EDITOR SCREEN	830
	5.1	SCREEN CONFIGURATION	059 8/0
	53	SCREEN OPERATIONS	841
	5.4	SETTING SCREEN	

B-61863E/14

	5.5	NET EDITOR SCREEN	846
	5.5	5.1 Screen Configuration	846
	5.5	5.2 Screen Operations	847
	5.6	STRUCTURE OF VALID NET	852
	5.7	FUNCTIONAL INSTRUCTION LIST SCREEN	853
	5.8	FUNCTIONAL INSTRUCTION DATA TABLE EDITOR SCREEN	854
	5.9	PROGRAM LIST EDITOR SCREEN	855
	5.9	0.1 Screen Operations	856
	5.9	0.2 Setting Screen	857
	5.10	ALARM MESSAGE AND COUNTERMEASURE	858
6. SIG		TRACE FUNCTION (PMC–SB7)	864
	6.1	SIGNAL TRACE SCREEN (INITIAL SCREEN)	865
	6.2	SETTING OF TRACE PARAMETER	
	6.3	SETTING OF SAMPLING ADDRESS	868
	6.4	EXECUTION OF TRACE	870
	6.5	OPERATION AFTER EXECUTION OF TRACE	871
	6.6	AUTOMATIC START OF TRACE SETTING	874
7. SE	LECTA	ABLE I/O LINK ASSIGNMENT FUNCTION (PMC–SB7)	875
	7.1	OUTLINE	876
	7.2	EXAMPLE	879
	7.3	NOTES	883
8. SY	STEM	PARAMETER (PMC–SB7)	884
9. I/O	LINK	CONNECTING CHECK SCREEN	886
10. O	NLINE	FUNCTION	887
	10.1	ONLINE SETTING SCREEN	888
	10.2	SETTING OF ONLINE CONNECTION	889
	10.	.2.1 How to Set at PMC Screen	889
	10.	.2.2 Setting of Online Connection by NC Parameter	890
	10.3	ONLINE FUNCTION BY ETHERNET	891
	10.	.3.1 Setting of Ethernet Parameters	891
	10.	.3.2 Starting online communication by offline programmer (Ethernet connection)	892
	10.4	COMMUNICATION STATUS	895

Volume 2

VI. STEP SEQUENCE FUNCTION

1. GEN	IERA	L	901
	1.1	STEP SEQUENCE METHOD	902
	1.2	GRAPHICAL SYMBOLS	. 905
	1.3	PROGRAMMING	906
2. STE	P SE	QUENCE BASICS	908
	2.1	TERMINOLOGY	909
	2.2	EXECUTION OF STEP SEQUENCE	918
3. COM	NFIGU	JRATION AND OPERATION OF STEP-SEQUENCE PROGRAMS	922
	3.1	STEP	. 923
	3.2	INITIAL STEP	925
	3.3	TRANSITION	926
	3.4	DIVERGENCE OF SELECTIVE SEQUENCE	927
	3.5	CONVERGENCE OF SELECTIVE SEQUENCE	927
	3.6	DIVERGENCE OF SIMULTANEOUS SEQUENCE	928
	3.7	CONVERGENCE OF SIMULTANEOUS SEQUENCE	929
	3.8	JUMP	930
	3.9		. 930
	3.10 2.11	BLUCK SIEP	022
	3.11 3.12	END OF BLOCK STEP	932 932
4. EXT	ENDI	ED LADDER INSTRUCTIONS	933
	41	FUNCTIONAL INSTRUCTION TRSET	934
	4.2	PMC ADDRESS (S ADDRESS)	. 934
5. SPE		CATION OF STEP SEQUENCE	935
	5.1	SPECIFICATION	936
	5.2	GENERAL RULES	937
	5.3	EXCLUSIVE CONTROL FOR FUNCTIONAL INSTRUCTIONS	943
6. CRT	/MDI	OPERATION	946
	6.1	DISPLAYING OF SEQUENCE PROGRAM	947
	6.1	.1 Program Configuration List (Main Screen)	947
	6.1	.2 Step Sequence Screen	948
	6.1	.3 Ladder Screen	950
	6.2	TIMER SCREEN	953
	6.2	.1 11me Screen	. 953

6.2.2	Monitoring Elapsed Time	954
6.3 MC	DNITOR TIME SCREEN	955
6.4 ED	ITING FUNCTION OF LADDER DIAGRAM	957
6.4.1	Program Configuration List (Main Screen)	957
6.4.2	Step Sequence Screen	958
6.4.3	Ladder Screen	960
6.5 CO	RRESPONDING FUNCTION	962
6.6 CO	MPATIBILITY OF LADDER DIAGRAM	963

VII. PMC PROGRAMMER (SYSTEM P series)

1. GENERAL	
2. FUNCTION	S OF PROCESSING
3. COMPONE	NT UNITS AND CONNECTIONS 970
3.1 CO	OMPONENT UNITS 971
3.2 CC	JNNECTIONS OF UNITS
2.3 NI	LOAD Kay (System Program Loading Kay) 076
3.3.1	EVAD Key (System Frogram Loading Key)
3.3.2	P Keys (P1 to P0)
334	Data Keys and Screen Scroll Key 978
3.4 SE	ETTING OF I/O DEVICE
4. OPERATION	N 981
4.1 GI	ENERAL 982
4.2 PF	REPARATION BEFORE OPERATION
4.2.1	System Floppy
4.2.2	Limitations with the SYSTEM P Mate
4.2.3	Loading of Floppy
4.2.4	FAPT LADDER System Floppy Loading
4.2.5	Programmer Menu Screen
4.2.6	Parameter Setting and Display
4.3 PF	ROGRAM EDITING 990
4.3.1	Data Display and Setting (Title, Symbol, Ladder Program, Comment, Message, I/O Module)
4.3.2	Programming from Keyboard
4.3.3	Alter
4.3.4	Insert
4.3.5	Delete
4.3.6	Location Search
4.3.7	Display of Ladder Diagram 1003
4.3.8	Help Screen
4.3.9	Editing end

4.4	INPLIT OF PROGRAM	1005
 <i>A A</i>	1 Source Program	1005
4.4 1 A	2 ROM Format Program	1005
 		1010
ч.5 Д 5	1 Source Program	1010
4.5	2 Paper Command	1013
4.5	3 POM Format Program	1013
4.5		1014
4.0	Collation of Source Programs	1017
4.0	2 POM Format Program	1017
4.0	DELETION OF PROCEAMS	1017
4.7	SPECIAL LISES OF THE P3 KEV	1010
4.8		1019
4.7	DIRECT EDITING BT LADDER DIAGRAM	1020
4.9	0.2 Limitations in SVSTEM P Mate	1020
4.9	3 Salection of Program Manu by Soft Keys	1020
4.9	A Sequence Program Input	1020
4.9	5 Substitution of Sequence Programs	1022
4.9	Additions to Sequence Programs	1025
4.9	7 Deleting a Sequence Program	1025
4.9 / 9	8 Searching a Sequence Program	1020
4.9	9 Conving a Sequence Program	1027
4.9	10 Moving a Sequence Program	1031
4.9	11 Symbol Data Display	1032
4.9	12 Compressed Input by [COMAND] Key	1033
4.9	12 Ending Edit of a Sequence Program	1034
4 10	INPUT/OUTPUT OF LADDER PROGRAM WITH P_G AND ELOPPY	1054
4.10	CASSETTE/FA CARD	1035
4.1	0.1 General	1035
4.1	0.2 Setting I/O Commands	1035
4.1	0.3 Program Input	1035
4.1	0.4 Program Output	1036
4.1	0.5 Program Collation	1037
		~~~
5. FILE EDI	TING FUNCTION	038
5.1	GENERAL	1039
5.2	CONFIGURATION OF COMMAND	1042
5.3	FDLIST COMMAND — FILE ATTRIBUTE DISPLAY	1043
5.4	RENAME COMMAND — FILE ATTRIBUTE CHANGE	1044
5.5	SCRATCH COMMAND — DELETION OF FILES	1045
5.6	CONDENSE COMMAND — RELEASE OF DELETED AREA	1045
5.7	REMOVE COMMAND — FILE COPY	1046
		051
A. ERRUR	$GODES LIST(FOR FAFT LADDER F^{-}G) \ldots \ldots \ldots \ldots \ldots \ldots T$	031

B. WINDOW FUNCTION DESCRIPTION (PMC-PA1/PA3/					
SA1/SA2	2/SA3/SA5/SB/SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)	1054			
B.1	FUNCTION	. 1055			

B-61863E/14

B.2	LOW	–SPEED RESPONSE AND HIGH–SPEED RESPONSE OF WINDOW FUNCTION	1055	
B.3	LIST	COF WINDOW FUNCTIONS 105		
B.4	FOR	MATS AND DETAILS OF CONTROL DATA	1059	
B.	.4.1	Reading CNC System Information	1060	
В.	.4.2	Reading a Tool Offset	1062	
В.	.4.3	Writing a Tool Offset (*Low-speed Response)	1064	
B.	.4.4	Reading a Workpiece Origin Offset Value (not Supported by the Power Mate–D or –F)	1066	
В.	.4.5	Writing a Workpiece Origin Offset Value (*Low-speed Response) (not Supported by the Power Mate-D or -F)	1068	
B.	.4.6	Reading a Parameter (*Low-speed Response)	1070	
B.	.4.7	Writing a Parameter (*Low-speed Response)	1072	
B.	.4.8	Reading Setting Data (*Low-speed Response)	1074	
B.	.4.9	Writing Setting Data (*Low-speed Response)	1076	
B.	.4.10	Reading a Custom Macro Variable (*Low-speed Response)	1078	
B.	.4.11	Writing a Custom Macro Variable (*Low-speed Response)	1081	
B.	.4.12	Reading the CNC Alarm Status	1083	
B.	.4.13	Reading the Current Program Number	1088	
B.	.4.14	Reading the Current Sequence Number	1090	
B.	.4.15	Reading the Actual Velocity of Controlled Axes	1092	
B.	.4.16	Reading the Absolute Position (Absolute Coordinates) of Controlled Axes	1094	
B.	.4.17	Reading the Machine Position (Machine Coordinates) of Controlled Axes	1096	
B.	.4.18	Reading a Skip Position (Stop Position of Skip Operation (G31)) of Controlled Axes	1098	
B.	.4.19	Reading the Servo Delay for Controlled Axes	1100	
B.	.4.20	Reading the Acceleration/Deceleration Delay on Controlled Axes	1102	
B.	.4.21	Reading Modal Data	1104	
B.	.4.22	Reading Diagnosis Data (*Low-speed Response)	1109	
B	.4.23	Reading A/D Conversion Data	1111	
B	.4.24	Reading Tool Life Management Data (Tool Group No.)	1115	
B	.4.25	Reading Tool Life Management Data (Number of Tool Groups)	1117	
B	.4.26	Reading Tool Life Management Data (Number of Tools)	1119	
B	.4.27	Reading Tool Life Management Data (Tool Life)	,	
		(not Supported by the Power Mate D, -F, or Series 21–TA)	1121	
B.	.4.28	Reading Tool Life Management Data (Tool Life Counter)	1123	
B.	.4.29	Reading Tool Life Management Data (Tool Length Compensation No. (1): Tool No.)	1125	
Β.	.4.30	Reading Tool Life Management Data (Tool Length Compensation No. (2): Tool Order No.)	1127	
B.	.4.31	Reading Tool Life Management Data (Cutter Compensation No. (1): Tool No.)	1129	
B.	.4.32	Reading Tool Life Management Data (Cutter Compensation No. (2): Tool Order No.)	1131	
B.	.4.33	Reading Tool Life Management Data (Tool Information (1) : Tool No.)	1133	
B.	.4.34	Reading Tool Life Management Data (Tool Information (2): Tool Order No.)	1135	
B.	.4.35	Reading Tool Life Management Data (Tool No.)	1137	
B.	.4.36	Reading the Actual Spindle Speed	1139	
Β.	.4.37	Entering Data on the Program Check Screen (*Low-speed Response) (not available for Power Mate–D/F, Series 21–TA)	1141	
B.	.4.38	Reading Clock Data (Date and Time) (not available for Power Mate–F)	1143	
B.	.4.39	Entering Torque Limit Data for the Digital Servo Motor (*Low-speed Response)	1145	
B.	.4.40	Reading Load Information of the Spindle Motor (Serial Interface)	1147	
B.	.4.41	Reading a Parameter (not available for Power Mate–D/F, Series 21–TA)	1150	

B.4.42	Reading Set Data (not available for Power Mate–D/F, Series 21–TA)	1151
B.4.43	Reading Diagnosis Data (not available for Power Mate-D/F, Series 21-TA)	1152
B.4.44	Reading a Character String of the CNC Program Being Executed in the Buffer	1153
B.4.45	Reading the Relative Position on a Controlled Axis	1155
B.4.46	Reading the Remaining Travel	1157
B.4.47	Reading CNC Status Information	1159
B.4.48	Reading Value of the P-code Macro Variable (*Low-speed Response)	1161
B.4.49	Writing Value of the P-code Macro Variable (*Low-speed Response)	1163
B.4.50	Reading the Tool Life Management Data (Tool Life Counter Type)	1165
B.4.51	Registering the Tool Life Management Data (Tool Group) (*Low-speed Response)	1167
B.4.52	Writing the Tool Life Management Data (Tool Life) (*Low-speed Response)	1169
B.4.53	Writing the Tool Life Management Data (Tool Life Counter) (*Low-speed Response)	1171
B.4.54	Writing the Tool Life Management Data (Tool Life Counter Type) (*Low-speed Response)	1173
B.4.55	Writing the Tool Life Management Data (Tool Length Offset Number (1) : Tool Number) (*Low-speed Response)	1175
B.4.56	Writing the Tool Life Management Data (Tool Length Offset Number (2) :	
	Tool Operation Sequence Number) (*Low-speed Response)	1177
B.4.57	Writing the Tool Life Management Data (Cutter Compensation Number (1) :	
	Tool Number) (*Low-speed Response)	1179
B.4.58	Writing the Tool Life Management Data (Cutter Compensation Number (2) :	1101
D 4 50	Tool Operation Sequence Number) (*Low-speed Response)	1181
B.4.59	(Tool Condition (1) : Tool Number) (*Low-speed Response)	1183
B.4.60	Writing the Tool Management Data (Tool Condition (2) : Tool Operation Sequence Number) (*Low-speed Response)	1185
B.4.61	Writing the Tool Life Management Data (Tool Number) (*Low-speed Response)	1187
B.4.62	Reading the Estimate Disturbance Torque Data	1189
B.4.63	Reading the Current Program Number (8–digit Program Numbers) (not available for Power Mate–D/F, Series 21–TA)	1193
B.4.64	Reading Tool Life Management Data (Tool Group Number)	1195
B.4.65	Reading Tool Life Management Data (Tool Length Offset Number 1)	1197
B.4.66	Reading Tool Life Management Data (Tool Diameter Offset Number 1)	1199
B.4.67	Reading Tool Life Management Data (Tool Information 1)	1201
B.4.68	Writing (Registering) Tool Life Management Data (Tool Group Number) (*Low-speed Response)	1203
B.4.69	Writing Tool Life Management Data (Tool Length Offset Number 1) (*Low-speed Response)	1206
B.4.70	Writing Tool Life Management Data (Tool Radius Offset Number 1) (*Low-speed Response)	1208
B.4.71	Writing Tool Life Management Data (Tool Information 1) (*Low-speed Response)	1210
B.4.72	Reading Actual Spindle Speeds	1213
B.4.73	Reading Fine Torque Sensing Data (Statistical Calculation Results)	1217
B.4.74	Reading Fine Torque Sensing Data (Store Data)	1219
B.4.75	Specifying the Number of the Program for I/O Link	1227
B.4.76	Preset of relative coordinate (*Low-speed response)	
	(not available for Power Mate and Series 21–TA)	1229
B.4.77	Deleting the Tool life Management Data (Tool group) (*Low-speed response)	1232
B.4.78	Deleting the Tool life Management Data (Tool data) (*Low-speed response)	1234
B.4.79	Deleting the Tool life Management Data (Tool life counter and Tool condition) (*Low-speed response)	1236

	B.4.80	Writing the Tool life Management Data (Arbitrary group number) (*Low-speed response)	1238
	B.4.81	Writing the Tool life Management Data (Remaining tool life) (*Low-speed response).	. 1240
	B.4.82	Reading the Current Screen Number	1242
	B.4.83	Reading Detailed Alarm Information (*Low-speed type)	1245
C. WIN	NDOW FU	INCTION DESCRIPTION (PMC–NB/NB2/NB6)	1249
	C.1 FUN	NCTION	1250
	C.2 LOV	W-SPEED RESPONSE AND HIGH-SPEED RESPONSE OF WINDOW FUNCTION	1251
	C.2.1	Functional Instruction WINDR	1252
	C.2.2	Functional Instruction WINDW	1254
	C.3 FOR THE	RMAT AND DETAILS OF THE CONTROL DATA OF E WINDR FUNCTIONAL INSTRUCTION	1256
	C.3.1	Reading a Tool Offset (Low-speed Response)	1256
	C.3.2	Reading a Workpiece Origin Offset Value	1258
	C.3.3	Reading a Parameter (Setting Data) (Low-speed Response)	1259
	C.3.4	Reading a Custom Macro Variable (Low-speed Response)	1260
	C.3.5	Reading the CNC Alarm Status (Low-speed Response)	1261
	C.3.6	Reading the Current Program Number (Low-speed Response)	1262
	C.3.7	Reading the Current Sequence Number (Low-speed Response)	1263
	C.3.8	Reading the Actual Velocity of Controlled Axes (Low-speed Response)	1264
	C.3.9	Reading the Absolute Position on a Controlled Axis	1264
	C.3.10	Reading the Machine Position (Machine Coordinates) of Controlled Axes	1266
	C.3.11	Reading a Skip Position (Stop Position of Skip Operation (G31)) of Controlled Axes (Low–speed Response)	1267
	C.3.12	Reading the Servo Delay for Controlled Axes	1268
	C.3.13	Reading the Acceleration/Deceleration Delay on Controlled Axes	1268
	C.3.14	Reading Modal Data (Low-speed Response)	1269
	C.3.15	Reading Diagnosis Data (Low-speed Response)	1275
	C.3.16	Reading A/D Conversion Data for the Feed Motor	1276
	C.3.17	Reading the Tool Life Management Data (Tool Group Number) (Low-speed Response)	1278
	C.3.18	Reading the Tool Life Management Data (Number of Tool Groups) (Low–speed Response)	1279
	C.3.19	Reading Tool Life Management Data (Number of Tools) (Low-speed Response)	1279
	C.3.20	Reading Tool Life Management Data (Tool Life) (Low-speed response)	1280
	C.3.21	Reading Tool Life Management Data (Tool Life Counter) (Low-speed Response)	1280
	C.3.22	Reading Tool Life Management Data (Tool Life Counter Type) (Low-speed Response)	1281
	C.3.23	Reading Tool Life Management Data (Tool Length Compensation No.1) (Low–speed Response)	1282
	C.3.24	Reading Tool Life Management Data (Tool Length Compensation No.2) (Low–speed response)	1283
	C.3.25	Reading Tool Life Management Data (Cutter Compensation No.1) (Low–speed Response)	1284
	C.3.26	Reading Tool Life Management Data (Cutter Compensation No.2) (Low–speed Response)	1285
	C.3.27	Reading Tool Life Management Data (Tool Information 1) (Low-speed Response)	1286
	C.3.28	Reading Tool Life Management Data (Tool Information 2) (Low-speed Response)	1287
	C.3.29	Reading Tool Life Management Data (Tool Number) (Low-speed Response)	1287
	C.3.30	Reading Clock Data (Low-speed Response)	1288

Table of contents

C.3.31	Reading the Relative Position on a Controlled Axis	1289
C.3.32	Reading the Remaining Travel	1290
C.3.33	Reading an Estimate Disturbance Torque Data	1291
C.3.34	Reading the Machining Time (Low–speed Response)	1292
C.3.35	Reading the Load Current (A/D Conversion Data) for the Spindle Motor	1293
C.3.36	Reading the Tool Offset Data According to the Specified Tool Number	1294
C.3.37	Reading Tool Life Management Data (Tool Group Numbers) (Low-speed Type)	1295
C.3.38	Reading Tool Life Management Data (Tool Length Compensation Number 1)	
	(Low-speed Type)	1296
C.3.39	Reading Tool Life Management Data (Cutter Compensation Number 1)	
	(Low-speed Type)	1297
C.3.40	Reading Tool Life Management Data (Tool Information 1) (Low-speed Type)	1298
C.3.41	Reading Real Parameters (Low-speed Type)	1299
C.3.42	Reading the actual Machine Position (Machine Coordinates) of Controlled Axes	1300
C.3.43	Reading Fine Torque Sensing Data (Statistical Calculation Results).	1301
C.3.44	Reading Fine Torque Sensing Data (Store Data)	1302
C.3.45	Reading detailed information of CNC alarm.	1306
C.4 FOR	MAT AND DETAILS OF THE CONTROL DATA OF THE	
WIN	IDW FUNCTIONAL INSTRUCTION	1309
C.4.1	Writing a Tool Offset Data	1310
C.4.2	Writing a Parameter (Setting Data)	1311
C.4.3	Writing a Custom Macro Variable	1312
C.4.4	Writing a Data on the Program Check Screen	1313
C.4.5	Writing the Torque Limit Override	1314
C.4.6	Writing the Tool Life Management Data (Tool Group Number)	1315
C.4.7	Writing the Tool Life Management Data (Tool Life)	1315
C.4.8	Writing the Tool Life Management Data (Tool Life Counter)	1316
C.4.9	Writing the Tool Life Management Data (Tool Life Counter Type)	1316
C.4.10	Writing the Tool Life Management Data (Tool Length Compensation Number 1)	1317
C.4.11	Writing the Tool Life Management Data (Tool Length Compensation Number 2)	1317
C.4.12	Writing the Tool Life Management Data (Cutter Compensation Number 1)	1318
C.4.13	Writing the Tool Life Management Data (Cutter Compensation Number 2)	1318
C.4.14	Writing the Tool Life Management Data (Tool Information 1)	1319
C.4.15	Writing the Tool Life Management Data (Tool Information 2)	1319
C.4.16	Writing the Tool Life Management Data (Tool Number)	1320
C.4.17	Writing the Tool Offset Data According to the Specified Tool Number	1321
C.4.18	Writing the Superposition Move Command	1322
C.4.19	Writing the Feedrate	1324
C.4.20	Writing Tool Life Management Data (Tool Group Numbers)	1325
C.4.21	Writing Tool Life Management Data (Tool Length Compensation Number 1)	1326
C.4.22	Writing Tool Life Management Data (Cutter Compensation Number 1)	1327
C.4.23	Writing Tool Life Management Data (Tool Information 1)	1328
C.4.24	Writing Real Parameters (Low-speed Type)	1329
D. WINDOW FU	NCTION DESCRIPTION (FS16/16 <i>i</i> –LA)	1330
D.1 OUT	LINE	1331
D.2 FUN	ICTION	1332
D.2.1	Transfer Between Data Area and Non-volatile Memory	1332

D.2	2.2 Reading of the Comment	1336
D.2	2.3 Reading and Writing the Laser Command Data and Laser Setting Data	1337
E. WINDOW	/ FUNCTION DESCRIPTION (FS16–W) 1	1340
E.1	READING THE WIRE DIAMETER OFFSET	1341
E.2	WRITING THE WIRE DIAMETER OFFSET (*LOW-SPEED RESPONSE)	1343
E.3	READING THE PARAMETER (*LOW-SPEED RESPONSE)	1345
E.4	WRITING THE PARAMETER (*LOW-SPEED RESPONSE)	1347
E.5	READING SETTING DATA	1352
E.6	WRITING SETTING DATA (LOW–SPEED TYPE)	1353
E.7	READING THE CNC ALARM STATUS	1356
E.8	READING MODEL DATA	1357
E.9	READING MACHINING DISTANCES	1361
E.10	READING THE MEASURED POINT	1362
E.11	WRITING THE MEASURED POINT (*LOW-SPEED RESPONSE)	1364
F. WINDOW	FUNCTION DESCRIPTION (FS16–PA) 1	1365
F.1	READING OF TOOL SETTING DATA	1366
F.1.	.1 Data Number, Data Attribute, Data Length, Data Area	1368
F.2	WRITING OF TOOL SETTING DATA (LOW–SPEED RESPONSE)	1370
F.3	READING TOOL SETTING DATA BY SPECIFYING TOOL NUMBER	1372
F.4	OTHER WINDOW FUNCTIONS	1374
G. SIGNAL	ADDRESS CONVERSION (FROM THE PMC-MODEL	1075
	THE PMC-MODEL SB/SC) 1	13/5
G.1	GENERAL	1376
G.2	FUNCTION	1376
G.3	CONVERSION	1376
G.4	MODIFYING THE CONVERTED SEQUENCE PROGRAM	1377
G.4	4.1 Modification Procedure	1377
H. CONNEC	TING THE OPERATOR'S PANEL FOR FS 0 WITH FS16,	
FS18, FS	S21, OR Power Mate 1	1378
H.1	GENERAL	1379
H.2	CONNECTION	1382
H.2	2.1 Connecting the I/O Unit	1382
Н.2	2.2 Connecting the I/O Card	1382
Н.3	SIGNALS FOR CONNECTING THE OPERATOR'S PANEL	1383
Н.3	B.1       Emergency Stop Signal (*ESP)	1383
Н.3	3.2 Override Signals (*OV1 to *OV8) and Program Protect Key Signal (KEY)	1383
Н.3	3.3 Key Switch Signals (Xn, Xn+2)	1383
Н.3	3.4 LED Signals (Ym)	1384
H.4	SPECIFYING ADDRESSES	1386
H.4	4.1   Parameter Menu	1386
H.4	#.2       Procedure	1386
I. EDITING F	FOR Power Mate–MODEL D (PMC–PA1/PA3) 1	388
I.1	OUTLINE	1389

I.	2	СОМ	PATIBILITY WITH CNC BASIC SOFTWARE	1389
I.	3	PMC	PROGRAMMER (CRT/MDI OR PDP/MDI) [LADDER EDITING FUNCTION]	1390
	I.3.1	1	Component Units and Connections	1390
	I.3.1	1.1	Component units	1391
	I.3.1	1.2	Connection of components	1392
	I.3.1	1.3	Parameter	1392
	I.3.2	2	Specification and Display of System Parameters (SYSPRM)	1393
	I.3.3	3	Condense	1393
I.	4	SYST	EM DIAGRAM OF SOFT KEY	1395
J. APPL		BLE	FAPT LADDER EDITIONS	. 1396
J.	.1	FAPT	LADDER, FAPT LADDER–II, FAPT LADDER–III,	
		LADI	DER EDITING PACKAGES FAPT LADDER-III (IBM PC/AT)	
		A08B	-9210-J505, LADDER EDITING PACKAGE (WINDOUS) (IBM PC/AI)	1307
I	2	FAPT	LADDER (SYSTEM P SERIES)	1399
K. LEVE		POF		. 1401
K	.1	OUTI	LINE OF LEVELED UP CONTENTS	1402
K	5.2	OPEF	ATION	1403
	K.2.	.1	$CNC \rightarrow Offline Programmer$	1403
	K.2.	.2	Offline Programmer $\rightarrow$ CNC	1405
TZ	K.2.	.5		1406
K	3	NB/N	B2 DATA COMPATIBILITY	1408
	K.3.	.1	Data Transfer Between NB (404/ Series) and FAPT LADDER	1408
	K.J.	.2	Data Transfer Between NB/NB2 (4048 Series) and FAPT LADDER	1408
	К.Э.	.5	Data Transfer Between NB (4047 Series) and NB (4048 Series)	1409
L. MIGF	RATIC	ON C	F LADDER PROGRAMS BETWEEN DIFFERENT MODELS	. 1410
L	1	MIGF	RATION OF LADDER PROGRAMS FROM Power Mate–D/H TO Power Mate $i$ –D/H .	1411
L	2	MIGF	RATION FROM THE PMC–NB/NB2 TO THE PMC–NB6	1413
L	3	MIGF	RATION FROM THE PMC–SA5/SB5/SB6 TO THE PMC–SB7	1414
	L.3.	.1	Addition of System Parameter Ladder Execution Time	1415
L	.4	MIGF	RATION FROM THE SERIES 0 AND SERIES 21–B TO SERIES 0 <i>i</i>	1416
	L.4.	.1	Compatible With Series 0	1416
	L.4.	.2	Compatibility With Series 21–MODEL B	1420
	L.4.	.3	Limited Function	1420
M. ALA	RMN	NES	SAGE LIST	. 1421
Ν	1.1	ALAI SB6/9	RM MESSAGE (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/SB2/SB3/SB4/SB5/ SC/SC3/SC4/NB/NB2/NB6)	1422
ν	1.2	ALAI	RM MESSAGE (PMC–SB7)	1429
N	1.3	SYST	EM ALARM MESSAGES (PMC–SC)	. 1434
N	1.4	SYST	EM ALARM MESSAGES (C LANGUAGE FUNCTION FOR PMC–NB/NB2/NB6)	1436
N	1.5	SYST	EM ALARM MESSAGES (PMC–SB5/SB6/NB6)	1437
N	1.6	PMC	SYSTEM ALARM MESSAGES (PMC–SB7)	1439
N. EXAI	MPLE	E OF	STEP SEQUENCE PROGRAMS	. 1441
ה פדבי	) e=4			1 / / =
0.316	- 321			1443
C	0.1	WHII	LE STATEMENT	1446

P. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST					
	0.5	SWITCH STATEMENT	1454		
	O.4	IF ELSE STATEMENT	1452		
	0.3	FOR STATEMENT	1450		
	O.2	DO-WHILE STATEMENT	1448		

# VI. STEP SEQUENCE FUNCTION

## GENERAL

#### 1.1 STEP SEQUENCE METHOD

The ladder method is most often used for programming the sequence control governed by a programmable controller. This method, shown in Fig.1.1(a), was derived from relay-panel control circuits. Since it has been in use for years, many sequence control engineers are already familiar with it. This method is also used in PMC sequence programming.



Fig. 1.1 (a) Ladder method

The greater the number of functions implemented by the PMC for a CNC system, the larger and the more complicated the sequence program becomes. A large-scale system requires a larger program and a greater number of processes, making it hard for the ladder method to control the overall process. This is because the ladder method does not describe the order of control. While the ladder method is suitable for describing partial control, it is hard to apply it to the description of the flow of control overall.

To overcome this problem, structured programming has been introduced into sequence control. A PMC that supports the subprogram function enables the use of modular programs. As shown in Fig.1.1(b), a large-scale program is divided into subprograms for each function, simplifying the unit of processing. Since the programmer determines how to divide the main program into subprograms and the control flow used to call the subprograms, however, the programs are not necessarily easy-to-understand by other programmers.



Fig. 1.1 (b) Module method

Given these conditions, a step sequence method has been created to describe programs structurally. It is well-suited to the control of entire processes and provides an easy-to-understand visualized flow of the process. The step sequence programming features the direct representation of the control flow on a flow chart, as shown in Fig.1.1(c). Each block of processing is described as a subprogram, using the ladder method. The entire program is then created by combining these subprograms.



#### Fig. 1.1(c) Step sequence method

The step sequence method has the following features:

- (1) Increased programming efficiency
  - Since the flow of processes can be programmed directly, simple, correct programming is enabled, reducing the time required for programming.
  - Even for complicated control, programming proceeds from the main flow to detailed flow in each process, creating a structured, top-down program, which is easy-to-understand by persons other than the original creator.
  - Structured modules can be used again easily.
- (2) Easy debugging and maintenance
  - Graphical display enables the operator to easily understand the execution state of a program visually.
  - Erroneous steps in a program can be found easily.
  - A part of a program can be easily modified.
- (3) High-speed program
  - Since only the subprograms required for a certain process are executed, the cycle time is reduced.
- (4) Transition from ladder programs
  - Since steps and transitions consist of conventional ladder programs, conventional ladder programs can be converted to new step sequence programs, without discarding ladder-program resources.

In step sequence programming, a sequence control program is divided into two types of subprograms, steps and transitions. Steps describe processes. Transitions connect steps and determine whether the transition conditions from one step to another evaluate true. As shown in Fig.1.1(d), a step sequence program is described using graphical symbols.



Fig. 1.1 (d) Example of machining the workpiece

As shown in this example, the program flow from process 1 through process 5 is expressed visually. Detailed programs related to the movements performed aspart of each process, and the signals used for determining whether transition conditions for proceeding to the next step are satisfied, are not described here. To program complicated control flows, many other functions are supported, such as divergence, jump, and nesting functions. The details of these functions are described later.

Step sequence programming is suitable for creating programs which control processes sequentially. Programs used for controlling a unit which operates according to a certain sequence, such as a loader, ATC, and other peripheral units, are best suited to step sequence programming. For programs which control units with no particular sequence, such as that of the operator's panel which is always monitoring the emergency stop signal or mode signals, however, are not well-suited to step sequence programming. The PMC supports the advantages of both methods, ladder and step sequence programming, by calling subprograms written according to a step sequence and those written as a ladder, from the main program.

#### 1.2 GRAPHICAL SYMBOLS

This manual uses the graphical symbols listed in Table 1.2 to describe step sequence flowcharts. Depending on the character font being used, the actually displayed symbols may differ slightly from those listed here. These graphical symbols are described in the subsequent chapters.

Table 1.2	List of	graphical	symbols
		grupinoui	0,1110010

	Display of programming manual	Display		
Contents		CNC Device	FAPT LADDER of Personal Computer	
Step	│ □ Sn │	│ □ Sn │	 □ Sn 	
Initial Step	 [□]] Sn 	 [□]] Sn 	 [□]] Sn 	
Transition	┿ Pn	- Pn	Pn	
Divergence of Selective Sequence				
Convergence of Selective Sequence	+ + +	+ + +	+ + +	
Divergence of Simultaneous Sequence		+	+	
Convergence of Simultaneous Sequence			+	
Jump	∟ → Ln	 _ > Ln	_ > Ln	
Label	− Ln →	 < _ Ln 	 < _ Ln 	
Block Step	│ □ ] Sn │	│ □ ] Sn │	│ □ ] Sn │	
Initial Block Step	│ [□] Sn │	 [□] Sn 	 [□] ] Sn 	
End of Block Step				

#### 1.3 PROGRAMMING

Follow the procedure below to create a step sequence program. Use a personal computer on which the FAPT LADDER software package is installed to code (edit) a program. Use a CNC to execute, debug and correct the ladder subprogram.

- (1) Create step sequence program (editing)
- (2) Create a subprogram of ladder diagram (editing)
- (3) Compile
- (4) Transfer to the CNC device (with the memory card or RS232C)
- (5) Write to the FlashROM
- (6) Execute
- (7) Diagnosis and debugging
- (8) Correct a subprogram of ladder diagram (editing)



#### Fig. 1.3 Programming to create a program

Table1.3 lists the step sequence functions supported by a personal computer (on which the FAPT LADDER software package is installed) and CNC.
#### Table 1.3 Step sequence functions

 $\bigcirc$  : usable

	PMC-SB4/ SB6 PM		PMC-NB2	Personal computer		
Functions		PMC-SC4		FAPT LADDER	FAPT LADDER II	FAPT LADDER III
Display and edit of a program						
<ul> <li>Display of subprogram list</li> </ul>	0	0	0	0	0	0
Create a new subprogram				0	0	0
Delete a subprogram				0	0	0
Edit a subprogram of Step Sequence form				0	0	0
Edit a subprogram of ladder diagram	0	0	0	0	0	0
Compile				0	0	0
Decompile				0	0	0
Input and output						
<ul> <li>Input and output with a memory card</li> </ul>	0	0	0	0	0	0
<ul> <li>Input and output with RS232C</li> </ul>	0	0	0	0	0	0
Write to a FlashROM	0	0	0		0	0
Execution of program						
<ul> <li>execution of a ladder diagram</li> </ul>	0	0	0		0	0
<ul> <li>execution of Step Sequence program</li> </ul>	0	0	0		0	0
Diagnosis and debugging (note1)						
Diagnosis of Step Sequence program	0	0	0			
Diagnosis of a ladder diagram	0	0	0		0	0
<ul> <li>Set and display a monitoring timer</li> </ul>	0	0	0			

## NOTE

While step sequence functions are being used, some of the diagnosis and debug functions supported by the ladder method cannot be used. For details, see 6.4 (Support Functions).



# 2.1 TERMINOLOGY

A step sequence program is created using a variety of graphical symbols, as shown in Fig. 2.1 (a). The main terms used in the step sequence are described below.



Fig. 2.1 (a) Step sequence elements





A step indicates a process, which is the basic processing unit in a step sequence program. In a step, specify the S address (Sn), which is a step number, and P address (Pm), which indicates a subprogram (action program) specifying the details of processing in each step.

(2) Step state transition

When a step sequence program is executed, the process proceeds as program processing advances, the state of each step changs accordingly. Each step can assume any of the logical states listed in Table 2.1, its state changes as shown in Fig. 2.1 (b). Activation refers to the changing of a step from the inactive state to the active state. Inactivation refers to the changing of a step from the active state to the inactive state.

State		Processing	Display	
Active	Execution	Activated step. The action program (subprogram) is being executed.	 ■ Sn 	
Inactive	active Transition Transition from execution to halt. to halt The action program (subprogram) is executed once only, then the step automatically transits to halt.		   \$2	
	Halt	Not activated state. The action program (subprogram) has not yet been executed.	Sn   _	

Table 2.1 Step state



Fig. 2.1 (b) Step state transition

#### (3) Transition



A transition denotes the transition conditions. When these evaluate true, the step of the corresponding state changes from the inactive to active state or vice the reverse. Specify the P address (Pn), which indicates a subprogram describing the transition conditions in detail.

As shown in Fig. 2.1 (c), step S2 changes its state from inactive to active when the conditions described in transition P10 evaluate true, while step S2 changes its state from active to inactive when the conditions described in transition P20 evaluate true.





Note that the step immediately before a transition must be active in order to switch the next step from inactive to active when the conditions specified in the transition evaluate true. As shown in Fig. 2.1 (d), step S3 does not change to the active state, even when transition P20 evaluates true, if step S1 is active and step S2 is inactive. An active state passes from a certain step to the next step when the corresponding transition conditions evaluate true, the execution of the step sequence program advancing one step.



Fig. 2.1 (d) Transition of step state by transition

— 911 —

#### (4) Initial Step



While a normal step can be activated by a transition, the initial step is activated automatically when execution of the program starts, as shown in Fig. 2.1 (e).



Fig. 2.1 (e) Activate of initial step

Although the initial step, which is usually executed first, is often placed at the top of a program, it can also be specified at some point within a program. It is always activated first. After being deactivated once, it can be subsequently be activated again. In this case, it acts in the same way as a normal step. (5) Divergence and Convergence of Selective Sequence

To describe a complicated sequence, selective sequences can be used. A selective sequence offers multiple choices, from among which the condition becomes true first activates the corresponding step, as shown in Fig. 2.1 (f). The divergent paths join to generate the mai sequence.



Fig. 2.1 (f) Selective sequence

(6) Divergence and Convergence of Simultaneous Sequence

A Simultaneous sequence can be used to execute multiple processes simultaneously. In a Simultaneous sequence, as shown in Fig. 2.1 (g), one transition activates multiple steps. The activated multiple steps are executed independently. Once all steps along the multiple paths have been completed, the divergent paths join to generate the main sequence.



Fig. 2.1(g) Simultaneous sequence

#### (7) Jump and Label

The jump function is used to describe a non-serial sequence, such as a repeated loop. As shown in Fig. 2.1 (h), when a jump designation is activated, the sequence jumps to the step having the corresponding jump destination label, after which that step is activated. To specify a label number, the L address is used in the same way as a jump instruction in ladder programming. A jump can be made to a previous or subsequent step.



Fig. 2.1 (h) Jump and Label

#### (8) Block

A block refers to a group of consecutive steps and transitions. A block can be a step sequence program. The more complicated the sequence becomes, the larger and more complex the block is. A program can be divided into multiple blocks in the same way as for subprograms in ladder programming, based on the concept of modular programming. Each block is identified by a P address, which corresponds to the subprogram number in ladder programming.

A block is executed as the main program in a step sequence, or called from another step sequence program as a subprogram.



Fig. 2.1 (i) Block

#### 2. STEP SEQUENCE BASICS

(9) Calling block

To execute a block as the main program in a step sequence, call the block with the CALLU (SUB 66) or CALL (SUB65) instruction in the same way as for ladder subprogram calling from the second level ladder program.



## Fig. 2.1 (j) Calling block

(10) Block step (calling step sequence program)

```
│
□ ] Sn
│ (Pm)
```

To call a block from the step sequence program as a subprogram, specify a block step in the step sequence program which calls the block, as shown in Fig. 2.1 (k). This is called bloc nesting.



Fig. 2.1 (k) Block nesting

The program shown in Fig. 2.1 (k) is equivalent to in Fig. 2.1 (l). which does not use a block step.



Fig. 2.1 (I) Program without block step

(11)End of block step



Use an end block step to terminate nested-block-step calling and to return to the calling sequence.

## 2.2 EXECUTION OF STEP SEQUENCE



Fig. 2.2 (a) Structure of program

In the step sequence method, a program is created (edited) in units of subprograms. The edited source program is compiled and converted to an executable ROM–format program, thenlinked, as shown in Fig. 2.2 (a). A ROM–format program is a kind of a modular program, created using conventional subprograms. A step sequence block is also a type of a subprogram. Step sequence blocks are linked to the end of the first level to third level ladder programs, together with other ladder subprograms.

In the same way as in the ladder method, a program is activated at certain intervals, namely every 8 ms, as shown in Fig. 2.2 (b). The first level and second level ladders are executed for a certain period (T ms), then the third level ladder is executed for the remaining time. The period in which the first level and second level ladders are executed varies with the PMC model and the setting of the system parameter (LADDER EXEC). Whether the third level ladder can be used depends on the PMC model.



#### Fig. 2.2 (b) Execution of program cyclically

After the first level ladder has been executed, the second level ladder i executed for the remaining time. If the second level ladder cannot be fully executed within one execution period, it is suspended part–way, with the remainder being executed in the nextperiod. This type of execution is called divided execution. Where the second level ladder is divided varies with the execution time of the first level ladder and that of the executed instructions of the second level ladder. Divided execution is divided into two types, divided system and undivided system. In the divided system, the position where the second level ladder is divided is determined in advance, a divided instruction code being inserted at that position. In the undivided system, in contrast, where the second level ladder is divided upon the determined period elapsing. A PMC which allows step sequence programming executes the second level ladder in undivided system.

In divided execution, the second level ladder is executed at an interval that is a multiple of 8 ms (e. g., 8, 16, 24 ms). Input signals referenced in the second level ladder, such as addresses X and F, are refreshed in synchronization with the execution period for the second level ladder, so that they do not change during the execution.

All subprograms, created using either the ladder or step sequence method, are called from the second level ladder. Hence, the execution time of the second level ladder includes those of ladder subprograms, step sequence programs (blocks), steps, and transitions. Since only the activated step and the transition which checks the transition condition from the step to the next step are executed in a step sequence program, the second level ladder is executed much more frequently than may be expected from the total number of steps.



Fig. 2.2 (c) Execution of step sequence

In the step sequence program shown in Fig. 2.2 (c), when step S1 is activated, subprograms are executed according to the timing illustrated in Fig. 2.2 (d).



Fig. 2.2 (d) Timing of execution of step sequenceprogram

In this case, step sequence program P2, step P3, transition P4, and ladder subprogram P1 are executed. Step P5 and transition P6 are not executed.





# CONFIGURATION AND OPERATION OF STEP-SEQUENCE PROGRAMS

# 3.1 STEP

A step is a unit of processing in a program.

#### [Display]



#### [Contents]

- Define a step number (Sn), necessary for controlling execution, and subprogram number (Pm) specifying actua processing, for a step.
- Assign a step number to a step.
- The same step number cannot be used twice in a program.
- A step has three logical states: the execution, transition to halt, and halt states. The execution state is also called the active state. The transition to halt and halt states are collectively called the inactive state.

S	tate	Contents of operation	Display	Sn.0 NOTE)
Activate	Execution	Activated step. The action program (subprogram) is being executed.	 ■ Sn 	1
Inactivate	Transition to halt	Transition from execution to halt. The action program (subprogram) is executed once only, then the step automatically transits to halt.	 □ Sn 	0
	Stop	Not activated state. The action program (subprogram) has not yet been executed.	 □ Sn 	0

## NOTE Refer to VI–4.2 PMC ADDRESS (S ADDRESS)

#### Example) State transition of Step B



## [Example]

After the M7 code is decoded, control is transferred to the next step using a DEC functional instruction.



# 3.2 INITIAL STEP

An initial step is automatically activated when execution of the program starts. Once it has been activated, it operates in the same way as a normal step. The program can be returned to this step through other steps.

## [Display]



#### [Contents]

- Define a step number (Sn), necessary for controlling execution, and subprogram number (Pm) specifying the actual processing, for an initial step.
- All initial steps are activated when the other steps are not activated.
- Each block must contain at least one initial step. No limit is applied to the number of initial steps contained in a block.
- A block having no initial step cannot be executed if called.
- Assign a step number to an initial step.
- The same step number cannot be used more than once in a program.
- In parallel branch, one initial step is required for each path. (See example 2.)

[Example1]



## 3. CONFIGURATION AND OPERATION OF STEP–SEQUENCE PROGRAMS STEP SEQUENCE FUNCTION

B-61863E/14





# 3.3 TRANSITION

A transition specifies the conditions governing the transition from the step to the next step.

#### [Display]



#### [Contents]

- Only one transition is required between steps.
- Transition between steps is performed as described below.



• When a signal is set to 1 in a transition, it remains the state even if the control is transferred to the subsequentstep. To set the signal to 0, use another subprogram to do so.

## [Example]

Refer an example described on the Step function (3.1).

# 3.4 DIVERGENCE OF SELECTIVE SEQUENCE

A selective sequence branches to two or more sequences. When the transition evaluates true, the corresponding step is activated.

#### [Display]

L			
+	-	$\rightarrow$	

#### [Contents]

- Transitions are placed after a divergence of selective sequence.
- The step connected to the transition for which the conditions are true is first activated.
- When the conditions for any transition are true simultaneously, the leftmost step is activated.
- A selective sequence can create up to 16 paths.

#### [Example]



# 3.5 CONVERGENCE OF SELECTIVE SEQUENCE

It combines two or more divergent paths to the main sequence.

## [Display]



[Contents]

The number of divergent paths must match that of the convergent paths.

## [Example]



# 3.6 DIVERGENCE OF SIMULTANEOUS SEQUENCE

A simultaneous sequence branches to two or more sequences, and all steps are activated simultaneously.

[Display]



[Contents]

- A transition must be placed before a divergence of simultaneous sequence.
- All branched steps are activated simultaneously, then executed.
- A simultaneous sequence can create up to 16 paths.

## [Example]



# 3.7 CONVERGENCE OF SIMULTANEOUS SEQUENCE

It combines two or more divergent paths to the main sequence.

## [Display]



## [Contents]

• A convergence of simultaneous sequence is processed as follows.



• Wait processing is processed as follows. case1)







A jump controls the execution of steps non–sequentially, together with a transition.

[Display]

## [Contents]

- Specify a jump destination label (Ln).
- The step to which control is transferred (jumped) is activated.
- The jump destination must be within the same program.
- A jump cannot be performed from outside a simultaneous sequence to within the simultaneous sequence, or from within a simultaneous sequence to outside.
- A jump cannot be performed between parallel–branched paths.

[Example]



3.9 LABEL

## A label specifies the jump destination.

## [Display]

**⊢**≁ Ln

[Contents]

Specify the jump destination label (Ln).

## [Example]

Refer to an example described on the jump function (3.8).

# 3.10 BLOCK STEP

A block step specifies the step sequence subprogram to be executed.

#### [Display]

│ □ ]Sn │ (Pm)

[Contents]

Define a step number (Sn), which controls the execution of a bloc step, and a sub-program (Pm) specifying the actual process, for a block step.

## CAUTION

- Assign a step number to a block step.
- The same step number cannot be used twice in a program.
- A transition must be placed after a block step.



- Transition P102 cannot be omitted due to the syntax of the step sequence method. Specify a dummy transition, which becomes always true, for transition P102.
- Transition P121 must specify the transition condition for the termination of the step S21.
- When the conditions of transitions P102 and P121 are switched, step S21 will not be correctly executed.

# 3.11 INITIAL BLOCK STEP

This is an initial step on the block step.

#### [Display]

#### [Contents]

- Define a step number (Sn), necessary for controlling execution, and subprogram number (Pm)specifying the actual processing, for an initial step.
- This step has the same function and graphical symbol asan initial step.

# 3.12 END OF BLOCK STEP

This terminates a block step.

## [Display]

## [Contents]

- Use this step to terminate a block step.
- Each block requires at least one end block step. No limit is applied to the number of end block steps.

## [Example]



# 4

# **EXTENDED LADDER INSTRUCTIONS**

To enable the specification of steps and transitions, the components of a step sequence program, by means of the ladder method, the following signals and functional instructions are provided. These signals and instructions can only be used in subprograms in which step sequence step and transitions are specified.

# 4.1 FUNCTIONAL INSTRUCTION TRSET

[Function]

- This instruction describes that the conditions for a transition have been true.
- This instruction is used in a subprogram which is called from a transition.

[Format]



# 4.2 PMC ADDRESS (S ADDRESS)

#### [Contens]

- This address is used to read the logical state of a specified step.
  - 0 : Transition to halt state, or halt state
  - 1: Execution state
- This address is used for creating a program in which detailed transitions of the execution states between steps are considered. Specify the number of the step to be read.

Example) To reference the state of the step S100

S100. 0

- This address allows any subprogram to reference the state of any step.
- Data cannot be written into state signal Sn. 0.
- A ladder can be configured for the TRSET transition instruction using state signal Sn. 0. Referencing state signal Sn. 0, however, adversely affects the portability and comprehensibility. Use this feature sparingly.

[Example]

This address is used to reference the activation states of steps in a step in which this address has been specified, and performs complicated wait processing in a program including a simultaneous sequence.





# 5.1 SPECIFICATION

Contents/Kind of PMC	PMC-RB4/RB6/RC4/NB2
Number of subprogram	Up to 2000 (P1 to P2000)
Number of step	Up to 1000 (S1 to S1000)
Number of label	Up to 9999 (L1 to L9999)
Number of jump in block	Up to 256
Nesting depth of block step	Up to 8 levels
Size of block	64 lines × 32 columns
Number of paths	Up to 16 paths



# 5.2 GENERAL RULES

• One transition must exist between step and step.



• The transition shall never be repeated even at the point of the divergence and the convergence.



#### 5. SPECIFICATION OF STEP SEQUENCE



• When a simultaneous sequence is specified in another simultaneous sequence, one convergence must not be used for each sequence.

• When a selective sequence is specified in a simultaneous sequence, dummy steps must be required both after the divergence and before convergence.





• In case of branching again immediately after the convergence, a step/transition is required between the divergence and convergence.

• Immediately after the block step, a dummy transition which is always true is needed.



#### 5. SPECIFICATION OF STEP SEQUENCE



## • The divergence must be terminated with the same type of convergence.

## • The number of convergences must match that of divergences.



• The number of convergences must match that of divergences, even at the end of a block step.





• It is not possible to jump to the other subprogram.

• It is not possible to jump from a simultaneous sequence to another simultaneous sequence.



• It is not allowed to jump from inside of the simultaneous sequence to outside.


#### 5.3 EXCLUSIVE CONTROL FOR FUNCTIONAL INSTRUCTIONS

The use of the following functional instructions is restricted in steps and transitions.

Group		Description	Functional instructions
A	Theins	structions operate when a signal changes	CTR (SUB5) CTRC (SUB60)
	Con– dition	Multiple functional instructions having the same number are used.	TMR (SUB3) TMRB (SUB24) TMRC (SUB54)
	Prob –lem	Not activated. Correct operation cannot be guaranteed.	DIFU (SUB57) DIFD (SUB58)
В	Restric	tion due to the interface.	WINDR (SUB51)
	Con– dition	Data is input or output by using two subprograms.	DISP (SUB49) DISPB (SUB41)
	Prob –lem	Invalid return value. Notterminated.	EXIN (SUB40) AXCTL (SUB53)

(1) Functional instructions of group A

Since these functional instructions operate when the corresponding signals change, they may not operate correctly when called from multiplesteps.

Example)

While multiple CTR functional instructions are used, when control passes from S1 to S2 with ACT of CTR not set to off, CTR is not counted when called from step S2.



#### 5. SPECIFICATION OF STEP SEQUENCE

#### Correct program



Divide the subprogram so that ACT of CTR is called after it is set to off.

(2) Functional instructions of group B

While an instruction is being executed through the interface with the NC, other same instructions cannot be executed. PMC control software does not receive the process when the instruction is not at a same position (net).

If ACT is set to on and off in different instructions (or subprograms), these processes are not terminated.

#### NOTE

In the window instructions (WINDR and WINDW), low-speed-type is included the functional instructions of group B.





Correct program

Correct the program so that ACT is set to on and off within one subprogram.



6

#### **CRT/MDI OPERATION**

The following operations are supported to enable the diagnosis and debugging of a step sequence program.

- (1) Displaying the sequence diagram
- (2) Displaying the run time of the step sequence program
- (3) Monitoring the run time of the step sequence program

#### 6.1 DISPLAYING OF SEQUENCE PROGRAM

The diagnosis and debugging of a step sequence program have four screens.

- (1) Program configuration list (main screen)
- (2) Step sequence screen
- (3) List screen
- (4) Ladder screen

Press the [STPSEQ] key and display the program configuration list.

#### 6.1.1 Program Configuration List (Main Screen)

STPSEQ < <ma< th=""><th>IN&gt;&gt; PRC PC</th><th>GRAM:(STE)001 (</th><th>P SEQUENCE ) SUB 1</th><th>E DEMO PROG PROGRAM NO.</th><th>RAM) MONIT</th><th>RUN</th></ma<>	IN>> PRC PC	GRAM:(STE)001 (	P SEQUENCE ) SUB 1	E DEMO PROG PROGRAM NO.	RAM) MONIT	RUN
LEVEL1	LEVEL2	LEVEL3				
□ P0001	□ P0002	□ P0004	P0005	□ <b>P0006</b>	□ <b>P0007</b>	
<b>P0008</b>	P0009	P0014	P0015	P0016	P0017	
P0021	P0022	P0024	P0025	P0026	□] P0027	
P0101	P0202	□]P0304	□]P0405	□]P0406	□]P0407	
	1 [ 00	TNT 1 [		פמתג ת	1 700	1
1 0P	J [ DOW		IIME ]	[ P-ADRS	J [ ZOOM	' /
: P0101 [        UP	□ ₽0202 ] [ DOW		□]P0405 TIME ]	D]P0406	□]p0407 ][ zoom	]

Items displayed on the screen

Display	Contents	Display by [ZOOM] key
LEVEL1	Ladder first level	Ladderdiagram
LEVEL2	Ladder second level	Ladderdiagram
LEVEL3	Ladder third level note1)	Ladderdiagram
D Pxxx	Subprogram	Ladderdiagram
□] Pxxx	Subprogram	Step sequence diagram

Pxxx indicates a subprogram number.

#### NOTE

The third level ladder can be omitted.

#### [ZOOM] key

To display the contents of a program, position the cursor to the program number and press the [ZOOM] key. The step sequence diagram (Fig. 6.1.2 (b)) or ladder diagram (Fig. 6.1.3 (c) is automatically displayed according to the type of the program.

#### [TIME] key

Press the [TIME] key to display the time display screen (Fig. 6.2.1) and time monitor screen (Fig. 6.3).

#### [P-ADRS/P-SYMB] key

Displays the addresses specified to subprograms, using addresses or symbols, if symbols have been assigned. When the [P–ADRS] key ispressed, the addresses are displayed. When the [P–SYMB] key is pressed, the symbols are displayed.

#### 6.1.2 Step Sequence Screen

(1) Position the cursor to a program indicated by □], then press the [ZOOM] key.

STPSEQ < <ma< td=""><td>IN&gt;&gt; PRO P(</td><td>GRAM:(STE) )407 (MAIN</td><td>P SEQUENCE ) STEP</td><td>E DEMO PROG SEQUENCE N</td><td>RAM) MONIT</td><td>RUN</td></ma<>	IN>> PRO P(	GRAM:(STE) )407 (MAIN	P SEQUENCE ) STEP	E DEMO PROG SEQUENCE N	RAM) MONIT	RUN
LEVEL1	LEVEL2	LEVEL3				
P0001	D P0002	□ P0004	P0005	P0006	P0007	
P0008	P0009	P0014	P0015	P0016	P0017	
P0021	P0022	P0024	P0025	P0026	□] P0027	
· ·			•			
· ·	•	•	•	•		
P0101	P0202	□]P0304	□]P0405	□]P0406	□]P0407	
[ UP	][ DOW	NN ] [	TIME ]	[ P-ADRS	] [ ZOOM	J
<b>`</b>						



Example)

When the cursor is positioned to  $\Box$ ] P0407 and press the [ZOOM] key, the subprogram P407 is displayed.

(2) Displayed Step Sequence

Activated steps are indicated by red  $\Box$  (highlighted  $\Box$  on a monochrome display). (In this manual, activated steps are indicated by  $\blacksquare$ .)



Fig. 6.1.2 (b) Step sequence screen

Display	Contents	Display by [ZOOM] key
[D] Sxxx	Initial step	Ladderdiagram
□ Sxxx	Step	Ladderdiagram
□] Sxxx	Block step	Step sequence diagram
+ Pxxx	Transition	Ladderdiagram
	Selective sequence	Cannot zoom.
	Simultaneoussequence	Cannot zoom.
└ <b>→</b> L2	Jump	Cannot zoom.
<b>←</b> L2	Label	Cannot zoom.

Meaning of display

Pxxx means the subprogram number.

[ZOOM] key

To display the contents of a program, position the cursor to the program number and press the [ZOOM] key. The step sequence diagram (Fig.6.1.2 (b)) or ladder diagram (Fig.6.1.3 (c)) is automatically displayed according to the type of the program.

#### [MAIN] key

Press the [MAIN] key to return to the program configuration list.

#### [CHANGE] key

Press the [CHANGE] key to list the subprograms referenced in the step sequence program.

#### [TIME] key

Press the [TIME] key to display the time display screen (Fig. 6.2.1).

#### [P-ADRS/P-SYMB/S-ADRS/S-SYMB] key

Displays the addresses specified with steps and transitions, using addresses or symbols, if symbols have been assigned. And the display of steps is changed to display the S addresses or P addresses.

Press the [P–ADRS] key to display the addresses of P addresses. Press the [P–SYMB] key to display the symbols of P addresses. Press the [S–ADRS] key to display the addresses of S addresses. Press the [S–SYMB] key to display the symbols of S addresses.

(3) Displaying the list screen

While the step sequence screen is displayed and press the [CHANGE] key, a list screen of the subprograms referenced in this step sequence program is displayed.

MONTT RUN PCLAD <<LIST>> PROGRAM: (STEP SEQUENCE DEMO PROGRAM) S0001 P0001 (ROTATE) ROTATE THE WORK TIP

] P0001 ] P0002 ] P0004 ] P0005 ] P0006 ] P0007 D P0008 □ P0015 D P0016 □ P0009 D P0014 □ P0017 □ P0021 P0022 □ P0024 P0025 □ P0026 □1P0027 P0101 □]P0202 □]P0304 □]P0405 □]P0406 □1P0407 ] [ DOWN ſ UP 1 [ 1 [ ] [ ZOOM 1

] [

] [

]

MAIN ] [ CHANGE ] [

E

— 949 —

#### [ZOOM] soft ke

To display a program, position the cursor to the program number and press the [ZOOM] key. The step sequence screen (Fig. 6.1.2 (b)) or ladder screen (Fig. 6.1.3 (c)) is automatically displayed according to the type of the program.

#### [MAIN] key

Press the [MAIN] key to return to the program configuration list.

#### [TIME] key

Press the [TIME] key to display the time display screen (Fig. 6.2.1).

#### [P-ADRS/P-SYMB] key

Displays the addresses specified to subprograms, using addresses or symbols, if symbols have been assigned. When the [P–ADRS] key is pressed, the addresses are displayed. When the [P–SYMB] key is pressed, the symbols are displayed.

#### [CHANGE] key

Press the [CHANGE] key to return to the step sequence diagram.

(1) Position the cursor to a program indicated by □, then press the [ZOOM] key.

STPSEQ < <ma< th=""><th>AIN&gt;&gt; PRO</th><th>GRAM: (STE</th><th>P SEQUENCE</th><th>DEMO PROG</th><th>RAM) MONIT</th><th>RUN</th></ma<>	AIN>> PRO	GRAM: (STE	P SEQUENCE	DEMO PROG	RAM) MONIT	RUN
LEVEL1 P0001 P0008 P0021	LEVEL2 P0002 P0009 P0022	LEVEL3 D P0004 P0014 P0024	<ul> <li>P0005</li> <li>P0015</li> <li>P0025</li> <li>.</li> </ul>	<ul> <li>□ P0006</li> <li>□ P0016</li> <li>□ P0026</li> </ul>	<ul> <li>P0007</li> <li>P0017</li> <li>P0027</li> </ul>	
- P0101 [ UP	] P0202 ] [ D07	□]₽0304 ฑง][	] P0405 TIME ]	] P0406 [ P-ADRS	□]₽0407 ][ ZOOM	1

## **Fig. 6.1.3 (a) Program configuration list (main screen)** Example)

When the cursor is positioned to LEVEL1, press the[ZOOM] key, the first level ladder is displayed.



#### Fig. 6.1.3 (b) Step Sequence screen

#### 6.1.3 Ladder Screen

— 950 —

Example)

When the cursor is positioned to "P2", press the [ZOOM] key, subprogram P2 is displayed.

(2) Ladder Screen

The signals currently set to on are displayed in white (highlighted on a monochrome display).



#### Fig. 6.1.3 (c) Ladder screen

[SEARCH] key

Used for search within a subprogram.



#### [TOP] key

Displays the top of a subprogram.

#### [BOTTOM] key

Displays the bottom of a subprogram.

#### [SRCH] key

Searches for the specified address.

#### [W-SRCH] key

Displays the ladder in which the specified address is used as a coil address.

#### [N-SRCH] key

Displays the ladder having the specified net number, at the top of the screen.

#### [F-SRCH] key

Displays the specified functional instruction, at the to of the screen.

#### [ADRESS/SYMBOL] key

Displays the addresses specified with relays and coils, using addresses or symbols, if symbols have been assigned. When the [ADRESS] key is pressed, the addresses are displayed. When the [SYMBOL] key is pressed, the symbols are displayed.

#### [TRIGER] key

With a manual operation or a signal trigger function, a renewal screen of a ladder monitoring function is stopped. By this function, the signal status when one signal is changed is certainly checked.

#### [WINDOW] key

Splits the screen into two sections, allowing the display of two ladder positions in a subprogram.

#### [DUMP] key

Displays the contents of addresses at the bottom of the screen.

#### [DPARA] key

Displays the data specified with functional instructions.

#### [ONLEDT] key

While a sequence program is executing, a part of the ladder diagram can be changed.

#### 6.2 TIMER SCREEN

#### 6.2.1 Time Screen

The elapsed time of a step sequence program is displayed.

/							)
	STPSEQ	< <stati< td=""><td>JS&gt;&gt; PRO</td><td>GRAM: (SI</td><td>TEP SEQUENCE</td><td>DEMO PROGRAM)</td><td>MONIT RUN</td></stati<>	JS>> PRO	GRAM: (SI	TEP SEQUENCE	DEMO PROGRAM)	MONIT RUN
	STEP S0001( S0002( S0003( S0004(	NO. ) ) )	STATUS EXEC EXEC EXEC	ELAPSE 1000000 100 10000 1000000	MONITOR T(1) OVER T(3) T(4)	STEP NO. S0010(TILE) S0011() S0012() S0013()	STATUS. EXEC
	[ T	JP ]	 [ DOI	NN ] [	SEARCH ]	[ RESET ] [	MONIT ]

Fig. 6.2.1 Time screen

Meaning of display

Display	Contents
STEP NO.	Step number S0001 : Step number (123456) : symbol display
STATUS	Step state EXEC : Active space : Inactive
ELAPSE	Actual elapsed time (per msec) The time is increasing during active state.
MONITOR	Monitortime T (1) : monitoring time number OVER : An elapsed time is over monitoring time

#### [UP] [DOWN] key

Scrolls the screen up or down, in units of pages, to display the operation time of other steps. Acts in the same way as the page up or down key.

#### [MONIT] key

Displays the screen used for setting the timer to monitor the operation time. (See 6.3)

#### [SEARCH] key

Search and display the specified step number.

example) Display the S100 address.

Key in "100" and press the [SEARCH] key.

#### [RESET] key

For all of monitoring steps, the error status which occurred by the monitoring function is canceled.

To cancel the status per steps, press the [DELETE] key on the monitor time screen. (Please refer to 6.3 Monitor Time Screen below)

# 6.2.2 When an activated state remains set for longer than the specified time, the state may be determined as being erroneous. The elapsed time can be specified for up to eight steps. When an activated state remains set for longer than the specified time, (1) OVER is displayed at the corresponding step number on the STPSEQ/TIME screen. (2) Execution of the ladder continues. (3) The bit of address R9118 which corresponds with the step number is set to

1. The processes for the error status can be program by the ladder diagram. And the following message is displayed on the PMC/ALARM screen.

"ER48 STEP SEQUENCE TIME OVER (xxH)"

"xx" displays the content of address R9118 in hexadecimal code.

Time Number	Corresponding Address	Time Number	Corresponding Address
1	R9118.0	5	R9118.4
2	R9188.1	6	R9118.5
3	R9188.2	7	R9188.6
4	R9188.3	8	R9188.7

— 954 —

#### 6.3 MONITOR TIME SCREEN

Operation time limits can be specified for a step sequence program. Up to eight steps can be monitored.

									)
STPSEQ< <mon< td=""><td>ITOR&gt;&gt; PROGI</td><td>RAM:</td><td>(STEP</td><td>SEQUEN</td><td>ICE DEMO</td><td>PROGRAM)</td><td>MONIT</td><td>RUN</td><td></td></mon<>	ITOR>> PROGI	RAM:	(STEP	SEQUEN	ICE DEMO	PROGRAM)	MONIT	RUN	
NO.	STEP NO.		EL.	APSE	MONTTOR				
T(1)	S0001(	)	100	0000	2000				
T(2)	S0010 (MOVE	ś		100	1000				
T(3)	S0002(	ś		100	2000				
т(4)	50003(	ś	1	0000	20000				
T(5)	20000	'	-						
T(6)									
T(7)									
T(8)									
- ( - )									
[ DELETE	] [	]	[		] [	] [		]	

#### Fig. 6.3 monitor time screen

Meaning of display

Display	Meaning
NO.	Monitortime number T (1) : means monitor time 1.
STEP NO.	Step number S0001 : Step number (123456) : symbol display
ELAPSE	Actual elapsed time (per msec) The time is increasing during active state.
MONITOR	Monitor time (per msec)

[DELETE] key

Delete the definition of monitor time.

#### Operation

Definition of monitor

(1) Position the cursor at the input position and input a step (or symbol).



Key in "MOVE" and push [INPUT] key.

(2) Position the cursor at the input position and define a monitor time.



Keyin "100" and push [INPUT] key.

#### Deletion of monitor

Position the cursor at the deletion and press [DELETE] key.

NO. STEP NO	Σ.	ELAPSE	MONITOR	
T(1) S0001(	)	1000000	2000	
T(2) S0010(N	IOVE )	100	1000	

Alteration of monitor

Position the cursor at the alteration position and input again.

NO.	STEP NO.		ELAPSE	MONITOR	
T(1)	S0001(	)	1000000	2000	
T(2)	S0100(	)	2000	1000	

Key in "S100" and push <INPUT> key.

#### 6.4 EDITING FUNCTION OF LADDER DIAGRAM

#### 6.4.1 Program Configuration List (Main Screen)

The display and editing of a step sequence program per subprogram are supported.

A step sequence program is allowed to be displayed and a ladder diagram is allowed to be displayed and edited.

Press the [EDIT] and [LADDER] key and display the program configuration list.

S	TPSEQ <<	MAIN>> PR P(	OGRAM:(STE )001 ( ) ST	P SEQUENCE JB PROGRAM	DEMO PROG NO.1	RAM) MONIT	STOP
	LEVEL1	LEVEL2	LEVEL3				
	D P0001	□ P0002	P0004	P0005	P0006	P0007	
	□ P0008	B 🗆 P0009	P0014	P0015	P0016	P0017	
	P0021	□ P0022	P0024	P0025	P0026	□] P0027	
		· ·	•		•		
		· ·	•		•		
	P0101	□ P0202	□]P0304	□]P0405	□]P0406	□]P0407	
	[ UP	] [ DO	WIN ] [	1	[ P-ADRS	] [ ZOOM	1

#### Items displayed on the screen

Display	Contents	Display by [ZOOM] key	
LEVEL1	Ladder first level	Ladderdiagram	
LEVEL2	Ladder second level	Ladderdiagram	
LEVEL3	Ladder third level (Note)	Ladderdiagram	
D Pxxx	Subprogram	Ladderdiagram	
□] Pxxx	Subprogram	Step sequence diagram	

Pxxx indicates a subprogram number.

#### NOTE

The third level ladder can be omitted.

#### [ZOOM] key

To display the contents of a program, position the cursor to the program number and press the [ZOOM] key. The step sequence diagram (Fig. 6.4.2 (b)) or ladder diagram (Fig. 6.4.3 (c)) is automatically displayed according to the type of the program.

#### [P-ADRS/P-SYMB] key

Displays the addresses specified to subprograms, using addresses or symbols, if symbols have been assigned. When the [P–ADRS] key is pressed, the addresses are displayed. When the [P–SYMB] key is pressed, the symbols are displayed.

#### 6.4.2 Step Sequence Screen

(1) Position the cursor to a program indicated by  $\Box$ ], then press the [ZOOM] key.

/							
LADDEI	R < <main></main>	> PROG P04	RAM:(STEP 07 ( ) ST	SEQUENCE EP SEQUENC	DEMO PROGI CE NO.1	RAM) MONIT	STOP
LEVI	EL1 LI	EVEL2	LEVEL3				
P	0001 🗆	P0002	P0004	P0005	P0006	P0007	
🗆 P	0008 🗆	P0009	P0014	P0015	P0016	P0017	
🗆 P	0021 🗆	P0022	P0024	P0025	P0026	□] P0027	
•	•	·	•	•	•		
□ ₽	0101 🗆	P0202	[] P0304	[] P0405	[] P0406	□]P0407	
[	UP ]	[ DOWN	1 [	TIME ]	[ P-ADRS	][ ZOOM	ı



Example)

When the cursor is positioned to  $\Box$ ] P0407 and press the [ZOOM] key, the subprogram P407 is displayed.

(2) Displayed Step Sequence

Activated steps are indicated by red  $\Box$  (highlighted  $\Box$  on a monochrome display). (In this manual, activated steps are indicated by  $\blacksquare$ .)



Fig. 6.4.2 (b) Step sequence screen

B-61863E/14

Display	Contents	Display by [ZOOM] key		
[□] Sxxx	Initial step	Ladderdiagram		
Sxxx	Step	Ladderdiagram		
□] Sxxx	Block step	Step sequence diagram		
+ Pxxx	Transition	Ladderdiagram		
	Selective sequence	Cannot zoom.		
	Simultaneoussequence	Cannot zoom.		
└ <b>→</b> L2	Jump	Cannot zoom.		
<b>←</b> L2	Label	Cannot zoom.		

Meaning of display

Pxxx means the subprogram number.

[ZOOM] key

To display the contents of a program, position the cursor to the program number and press the [ZOOM] key. The step sequence diagram (Fig. 6.4.2 (b)) or ladder diagram (Fig. 6.4.3 (c)) is automatically displayed according to the type of the program.

#### [MAIN] key

Press the [MAIN] key to return to the program configuration list.

#### [CHANGE] key

Press the [CHANGE] key to list the subprograms referenced in the step sequence program.

#### [P-ADRS/P-SYMB/S-ADRS/S-SYMB] key

Displays the addresses specified with steps and transitions, using addresses or symbols, if symbols have been assigned.

And the display of steps is changed to display the S addresses or P addresses.

Press the [P–ADRS] key to display the addresses of P addresses. Press the [P–SYMB] key to display the symbols of P addresses. Press the [S–ADRS] key to display the addresses of S addresses. Press the [S–SYMB] key to display the symbols of S addresses.

(3) Displaying the list screen

While the step sequence screen is displayed and press the [CHANGE] key, a list screen of the subprograms referenced in this step sequence program is displayed.

(													)
	LADDER	< <li< th=""><th>ST&gt;&gt;</th><th>• P</th><th>ROGR</th><th>AM: (S</th><th>TEP</th><th>SEQUEN</th><th>CE</th><th>DEMO PRO</th><th>GRAM)</th><th>MONIT</th><th>STOP</th></li<>	ST>>	• P	ROGR	AM: (S	TEP	SEQUEN	CE	DEMO PRO	GRAM)	MONIT	STOP
		S00	001 H	P0001	(ROT	'ATE)	ROTA	TE THE	WO	RK TIP		_	
		20001		P0002		P0004		P0005 L	11	P0006 []	P0007		
	P(	008		P0009		P0014		P0015		P0016	D P00	- 17	
	🗆 P(	0021		P0022		P0024		P0025		P0026	□] P00	27	
	•	•		•		•		•		•			
							<b>—</b> 1	D0405		D0406		07	
		101		P0202		P0304		P0405		P0406		07	
	[	UP	] [	СНА	NGE	] [		]	[	P-ADRS	] [	ZOOM	1
													)

#### [ZOOM] soft key

To display a program, position the cursor to the program number and press the [ZOOM] key. The step sequence screen (Fig. 6.4.2 (b)) or ladder screen (Fig. 6.4.3 (c)) is automatically displayed according to the type of the program.

#### [MAIN] key

Press the [MAIN] key to return to the program configuration list.

#### [CHANGE] key

Press the [CHANGE] key to return to the step sequence diagram.

#### [P-ADRS/P-SYMB] key

Displays the addresses specified to subprograms, using addresses or symbols, if symbols have been assigned. When the [P–ADRS] key is pressed, the addresses are displayed. When the [P–SYMB] key is pressed, the symbols are displayed.

#### 6.4.3 Ladder Screen

(1) Position the cursor to a program indicated by  $\Box$ , then press the [ZOOM] key.

LADDER < <ma< th=""><th>IN&gt;&gt; PRO</th><th>GRAM: (STE</th><th>P SEQUENCE</th><th>DEMO PROG</th><th>RAM) MONIT</th><th>STOP</th></ma<>	IN>> PRO	GRAM: (STE	P SEQUENCE	DEMO PROG	RAM) MONIT	STOP
LEVEL1 D P0001 P0008 P0021	LEVEL2 P0002 P0009 P0022	LEVEL3 D P0004 P0014 P0024	<ul> <li>P0005</li> <li>P0015</li> <li>P0025</li> </ul>	<ul> <li>□ P0006</li> <li>□ P0016</li> <li>□ P0026</li> </ul>	<ul> <li>P0007</li> <li>P0017</li> <li>P0027</li> </ul>	
P0101 [ UP	P0202 ] [ DOW	□]₽0304 /N][	[] p0405 ]	] P0406 [ P-ADRS	□]₽0407 ][ ZOOM	1

#### Fig. 6.4.3 (a) Program configuration list (main screen)

#### Example)

When the cursor is positioned to LEVEL1, press the [ZOOM] key, the first level ladder is displayed.



Fig. 6.4.3 (b) Step Sequence screen

Example)

When the cursor is positioned to "+P2", press the [ZOOM] key, the subprogram P2 is displayed.

(2) Ladder Screen



#### Fig. 6.4.3 (c) Ladder screen

Please refer to the following section about the operations of editing a ladder diagram.

III PMC PROGRAMMER(CRT/MDI) 5.2 Sequence Program Generation(LADDER)

#### 6.5 CORRESPONDING FUNCTION

The following ladder diagnosis and debugging functions can be used together with the step sequence functions.

 $\bigcirc$  : can be used  $\Delta$  : can be used on condition  $\times$  : cannot be used

Functions	PMC-SB4/ SB6	PMC-SC4	PMC-NB2
PMC Ladder diagram display (PMCLAD)	×	×	×
PMC I/O signal display (PMCDGN) Title screen (TITLE) Signal status screen (STATUS) Alarm screen (ALARM) Trace screen (TRACE) Contents of Memory (MEMORY) Signal Waveforms screen (ANALYS) Running State of a User Task (USRDGN)	Ο Ο Ο Δ NOTE1 Χ	000000	0000000
PMC Parameters screen (PMCPRM) Timer screen (TIMER) Counter screen (COUNTR) Keep relay screen (KEEPRL) Data table screen (DATA) Simple setting screen (SETING)	000000	00000	00000
Step Sequence screen (STPSEQ) Displaying Step Sequence screen Displaying Ladder screen SEARCH Display address and symbol Trigger function (TRIGER) Divided screen function (WINDOW) Contents of memory (DUMP) Contents of parameter (DPARA) online editting (ONLEDT) Time screen (TIME) Monitor time screen (MONIT)	00000000 ▲000	00000000000	000000000000000000000000000000000000000
Execute or stop the sequence program (RUN)	0	0	0
Edit function (EDIT) Title screen (TITLE) Ladder diagram (LADDER) Symbol screen (SYMBOL) Message screen (MESAGE) Definition of I/O (MODULE) Cross reference (CROSS) Memory clear (CLEAR)	NOTE1 Δ Δ Δ Δ Δ Δ Δ	000000	000000
Input and output FAPT LADDER (HOST) Floppy cassette (FDCAS) FlashROM (F–ROM) Memory card (M–CARD) Other I/O device (OTHERS)	О О А NOTE2	00000	00000
System Parameter (SYSPRM)	$\Delta$ NOTE1	0	0
Debug function (MONIT) Ladder debug function (DBGLAD) Descriptor table screen (GDT) User memory screen (USRMEM) User program debug function (DEBUG)	NOTE1 × × × ×	× 000	× 0 000

#### NOTE

- 1 An Editor card is needed.
- 2 It is possible to use while an Editor card is not mounted. (Can be used with the 16*i*/18*i*/21*i*.)

#### 6.6 COMPATIBILITY OF LADDER DIAGRAM

The PMC–SB4, SB6, SC4 and NB2 can be used with either the ladder method or step sequence method. When a step sequence program is transferred to the old version of the PMC, ER08 OBJECT UNMATCH is displayed on the PMC/ALARM screen.

The model setting of FAPT LADDER determines whether the ladder or step sequence method is used. The STEP SEQUENCE item has been added to the system parameter screen for future expansion. Specify the parameter according to the model setting of FAPT LADDER.

To create a program with the built–in edit function, after the parameter has been set execute CLEAR ALL. Alternatively, while holding down "X" and "O" key, turn the power off and on.

When the step sequence method is used: STEP SEQUENCE = YES.

When the ladder method is used: STEP SEQUENCE = NO.

PMC SYSTEM PARAMETER	(1/	2)	MONIT STOP
COUNTER DATA TYPE	=	BINARY /	BCD
STEP SEQUENCE	=	YES /	NO
>			
[BINARY] [ BCD ] [		] [	1[]

Fig. 6.6 (a) PMC–SB4/SB6 system parameter screen (first page)

PMC SYSTEM PARAMETER	(2/2)	MONIT S	тор
FS0 OPERATOR PANEL	=	YES / NO	
KEY DI ADDRESS	=		
LED DO ADDRESS	=		
KEY BIT IMAGE ADDRES	S =		
LED BIT IMAGE ADDRES	S =		
>			
VES INO I		1 1 1 1	1



```
PMC SYSTEM PARAMETER (1/2) MONIT STOP
 COUNTER DATA TYPE = BINARY / BCD
 LADDER EXEC = \% (1-150)
 LANGUAGE EXEC RATIO = \% (0-99)
 LANGUAGE ORIGIN =
(LANGUAGE AREA = H,
                        н
                  H, SIZE = KB)
 STEP SEQUENCE = YES / NO
>
 [BINARY] [ BCD ] [ ] [ ] [ ]
```



PMC SYSTEM PARAMETER (2/2	2) MONIT STOP
FS0 OPERATOR PANEL =	YES / NO
KEY DI ADDRESS =	
LED DO ADDRESS =	
KEY BIT IMAGE ADDRESS =	
LED BIT IMAGE ADDRESS =	
>	
[ YES ] [ NO ] [	] [ ] [ ]

Fig. 6.6 (d) PMC–SC4/NB2 system parameter screen (second page)

# VII. PMC PROGRAMMER (SYSTEM P series)

#### GENERAL

The FAPT LADDER system can easily prepare sequence programs, symbol data, titles, and message of PMC–SB and PMC–SC, and also easily define addresses of the modules to be installed in an I/O unit by using SYSTEM P series.

Major functions of this FAPT LADDER are as described below.

- (1) Input, display and editing of sequence programs
- (2) Transfer of sequence programs (including write into EPROM for PMC or ROM module.)
- (3) Collation of sequence programs
- (4) Program error display

The SYSTEM P series is used in the stage of preparing a sequence program only and separated from PMC after the sequence program has been completed. The SYSTEM P series can be connected to PMC only when the PMC is operated with the RAM card and cannot be connected when PMC is operated with a EPROM for PMC or ROM module.





# **2** FUNCTIONS OF PROCESSING

(1) Input of sequence programs

Input sequence programs using the following units when sequence programs are loaded into the memory of the SYSTEM P series.

- a) SYSTEM P series keyboard
- b) PPR tape reader (paper tape)
- c) Floppy
- d) PMC memory
- e) EPROM for PMC or ROM module
- (2) Sequence program display

Sequence programs can be displayed on the 12" graphic display of SYSTEM P series as follows.

- a) Sequence programs can be displayed using mnemonic symbols.
- b) Sequence programs can also be displayed in the ladder diagram format.
- (3) Editing of sequence programs

A sequence program can be edited by using the SYSTEM P series keyboard in the following three ways.

- a) Alteration
- b) Insertion
- c) Deletion
- (4) Transfer of sequence programs

Sequence programs can be transferred as follows.

- a) From SYSTEM P series memory to PMC memory
- b) From PMC memory to SYSTEM P series memory
- c) From SYSTEM P series memory to floppy
- d) From floppy to SYSTEM P series memory
- e) From SYSTEM P series memory to EPROM or ROM module for PMC (Write into EPROM for PMC or ROM module)
- f) From EPROM for PMC or ROM module to SYSTEM P series memory
- (5) Collation of sequence programs

Sequence programs can be checked by collating them between the following memories.

- a) SYSTEM P series memory PMC memory
- b) SYSTEM P series memory floppy
- c) SYSTEM P series memory EPROM for PMC or ROM module
- d) SYSTEM P series memory paper tape

#### (6) Hard copy

- a) Since FANUC PPR is connectable to SYSTEM P series, the paper tape output and list output (mnemonic symbol) are obtainable.
- b) A ladder diagram can be printed out.
- (7) Program error display

Sequence program errors are displayed on the screen of the SYSTEM P series.

Error codes are displayed at the lower right of the screen as ALARM=XXX.

Refer to list of error codes in Appendix.



#### 3.1 COMPONENT UNITS

#### (1) SYSTEM P series

This system serves as a programmer to generate and edit sequence programs.

(2) Series 16

This system transfers a generated sequence program to CNC.

(3) FANUC PPR

This PPR inputs/outputs a sequence program by using a paper tape, and also output a source list to the printer.

(4) FANUC printer

This printer prints out the sequence program.

(5) FANUC PMC writer

This unit is used for writing a sequence program to the EPROM for PMC or ROM module when the sequence program has been completed.

(6) FANUC FA Writer

This unit is used for writing data to the EPROM or ROM module for the PMC after a sequence program has been created.

#### 3.2 CONNECTIONS OF UNITS

For details of the connections of SYSTEM P series unit power supply, PPR, and other units as well as their operation, refer to the following operator's manuals.

SYSTEM P-G Mark II: B-66014E

SYSTEM P–G Mate: B–66003E

This chapter mainly describes the connections between SYSTEM P series and I/O devices.



Fig. 3.2 (a) External view of SYSTEM P Mark II



Fig. 3.2 (b) External view of SYSTEM P Mate

Since a volatile RAM is employed as the SYSTEM P series memory, all programs (FAPT LADDER system programs and sequence program) being loaded into memory are operation should be started with the input of FAPT LADDER system programs (called system loading).

If the SYSTEM P series power supply is turned off halfway in the curse of inputting a sequence program from the keyboard, the sequence program must be stored in advance, and this FAPT LADDER provides an output function to a floppy for this purpose.

### 3. COMPONENT UNITS AND CONNECTIONS



#### Fig. 3.2 (c) Connection of SYSTEM P series with each unit

- (1) Connect FANUC PPR to connector CN1.
- (2) Connect FANUC PMC writer or FANUC FA writer to connector CN2.
- (3) Connect FANUC printer to connector CN3.
- (4) Connect connector CN4 to PMC-SA1/SA2/SB/SB2/SC. It is connected to a channel preset by a PMC I/O. For details, refer to "Setting and display of I/O in PMC programmer (CRT/MDI) in III".
  Connector JD5A on MAIN PCB → 1 CHANNEL Connector JD5B on MAIN PCB → 2 CHANNEL

#### 3.3 KEYBOARD OF SYSTEM P SERIES

Figs. 3.3(a) - (b) show the panel of the SYSTEM P series keyboard.

It is not necessary to memorize the meanings of keys on the keyboard.

Descriptions of these keys and menus are displayed on the SYSTEM P series screen by operation, and you can easily operate the SYSTEM P series board while monitoring the SYSTEM P series screen.

In this chapter, you should understand an outline of functions of these keys.







Fig. 3.3 (b) Panel of the SYSTEM P Mate key board

3.3.1 LOAD Key (System Program Loading Key)	This key is used to load the FAPT LADDER system program into the SYSTEM P series memory through a floppy disk at the first time after turning on power.					
3.3.2 F Keys (F1 to F0)	F key is used to select an I/O device among I/O devices connected at that time.					
	These F keys are provided with an LED. When depressing a key, the LE lights, and when depressing the key once more, the LED goes out. The lighting condition of of this LED indicates that an I/O has been designated. No I/O device is operable when its corresponding LED is no lighting.					
	The correspondence between F keys and I/O devices is as sh (I) shows an input, while (O) shows an output.					
	(1)	<f1> key: FANUC PPR paper tape reader (I)</f1>				
	(2)	<f2> key: Floppy disk input (I)</f2>				
	(3)	<f3> key: Not used</f3>				
	(4)	<f4> key: Display of ladder diagram on SYSTEM P series screen (O)</f4>				
	(5)	<f5> key: FANUC PPR printer (O)</f5>				
	(6)	<f6> key: FANUC PPR paper tape puncher (O)</f6>				
	(7)	<f7> key: Floppy disk output (O)</f7>				
	(8)	<f8> key: PMC–PA1/PA2/SA1/SA2/SB/SB2/SC (I/O)</f8>				
	(9)	<f9> key: FANUC PMC writer, FANUC FA writer (I/O)</f9>				
	(10)	<f10> key: FANUC printer (O)</f10>				
		(The ladder diagram is printed on the printer.)				
	(11)	<f13> key: FANUC Floppy Cassette/FANUC FA Card adapter (I)</f13>				
	(12)	<f14> key: FANUC Floppy Cassette/FANUC FA Card adapter (O)</f14>				
	Comb which	ination of F key and menu number of FAPT LADDER decided function is to be executed.				

R Keys (R0 to R3)	Four R keys $<$ R0> to $<$ R3> are provided. The meaning of these keys differ according to the screen conditions at their operating time, even in case of the same key.	
	(1) R key menu screen	
	This screen is obtained just after loading a FAPT LADDER system program (1/2) or when pressing <nl> key only in a menu screen Refer to Fig. 4.2.2.</nl>	
	<r0> FAPT ladder start.</r0>	
	. A menu screen appears.	
	<r1> Editing a ladder diagram starts.</r1>	
	<r2> Not used in FAPT ladder. (Not accepted when pressing these keys)</r2>	
	<r3> Request key (see 4.8)</r3>	
	Press NL keys, if a wrong key was pressed by mistake. The screen i reset to the condition before pressing the wrong R key.	
	(2) Other than R key menu screen	
	<r0> This key operation is accepted when EDIT is displayed at the lower left part of the screen (called EDIT screen hereafter during sequence program editing. The screen is switched to the sequence program, symbol, message, I/O module, and title, each time this R key is pressed.</r0>	
	<r1> 1 When this key is pressed during printing of a ladde diagram on an external printer, the printer stops every pag to be ready for key entry.</r1>	
	2 When this key is pressed during data transfer between SYSTEM P series and PMC–SB/SC, data transfer i stopped.	
	<b>3</b> The signal display in a sequence program is alternately selected to symbols and addresses, each time this R1 i pressed during the display of the sequence program on the screen.	
	<r2> Data on the last page are displayed, each time this key i pressed on the EDIT screen.</r2>	
	<r3> 1 Data on the next page are displayed, each time this key i pressed on the EDIT screen.</r3>	
	2 Transfer is aborted when this key is pressed during ROM data transfer between SYSTEM P series and PMC WPITEP or floppy	

#### 3.3.4 Data Keys and Screen Scroll Key

Data keys are used to enter data. To switch the output of such keys between the upper character and lower-character, use the [SHIFT] key or [LOCK] key. Pressing the [SHIFT] key together with an arbitrary key changes the output of the arbitrary key to the upper character, and pressing the [LOCK] key changes the output of all keys to upper character. To release the upper character mode, press the [LOCK] key again. Special keys are described below.

(1)  $\langle NL \rangle$  key

Data entry from the SYSTEM P series keyboard are input into SYSTEM P series by depressing <NL> key.

Two <NL> keys are located on the keyboard for easily operation.

(2)  $\langle CAN \rangle$  key

Data being entered from the keyboard are cancelled.

(3) BS key

Data being entered from the keyboard are sequentially deleted leftward, each time this key is depressed.

(4) Arrow keys  $<\uparrow><\downarrow><\leftrightarrow><\rightarrow>$ 

These keys are accepted only when a ladder diagram is being displayed on the screen, and used for scrolling the ladder diagram.

#### NOTE

None of [INS] [DEL] [CHG] [AUX] keys and K key is employable in the FAPT LADDER.

DEVICE

3.4

# **SETTING OF I/O**

#### (1) SYSTEM P series Mate

An initial I/O device setting of 'FAPT LADDER' for SYSTEM P series Mate is as follows.

#### Table 3.4(a) FAPT LADDER (Mate) of table

Channel	I/O device	F key
CN1	PMC-RAM	F8
CN2	PMC WRITER FA WRITER	F9
CN3	External printer	F10

Alter the setting of the I/O device by under-mentioned 'IO command' when using FANUC PPR.

i) Press the R3 key in the menu screen of R keys.

'REQUEST =' is displayed in the left bottom of screen and becomes the state which can be typed in.

ii) Type in IO PPR, CN1 <NL>.

PPR is allocated to channel 1.

iii) Type in as follows when channel 1 allocation is returned to PMC-RAM.

IO, NC, CN1, F8, BR10 <NL>

(2) SYSTEM P Mark II

The initial setting of I/O devices of FAPT LADDER for the SYSTEM P Mark II is as follows.

Table 3.4(b) FAPT LADDER (Mark II)

Channel	I/O device	F key
CN1	FANUC PPR	F1, F5, F6
CN2	PMC WRITER FA WRITER	F9
CN3	External printer	F10
CN4	PMC-RAM	F8

- (3) When a FANUC Floppy Cassette or FANUC FA Card adapter is used, change the setting of the I/O device by executing the following I/O command:
  - Press the [R3] key on the menu screen for the [R] keys. i)

Then REQUEST = appears at the lower left of the screen allowing data to be entered.

- ii) Type IO BCA, CN2, F13, F14, then press the <NL> key. The FANUC Floppy Cassette or FA Card adapter is allocated to channel 2.
- iii) To initialize the setting of channel 2 again, type IO AUX, CN2, F9, then press the  $\langle NL \rangle$  key.
- (4) Setting of the ROM writer

The PMC–SA1, –SA2 or –SB uses one of 1MB EPROM (27C1024).

The PMC-SC uses one of ROM module (128KB, 256KB, or 512KB).

#### (Setting method of IO command)
For this reason, when the PMC–SA1, –SA2 or –SB is used, both the FA Writer and PMC Writer can be used. When the PMC–SC is used, only the FA Writer is available.

When the PMC–SA1, –SA2 or –SB is used, the ROM writer used can be selected on the REQUEST screen as follows.

- 1 On the REQUEST screen, enter WRITER then press the <NL> key.
- **2** The following message appears. To select the FA Writer, enter 0 or press the <NL> key. To select the PMC Writer, enter 1.

SET KIND OF ROM WRITER (0:FA WRITER, 1:PMC WRITER) WRITER=

The current setting of the ROM writer can be checked on the system parameter screen.

(1) PMC Writer

The PMC Writer is required when the PMC–SA1, –SA2, –SB or SB2 is available. To use a 1MB EPROM (27C1024), the 1M EPROM adapter (A13B0147–B001) is required. Set the EPROM select switch to the 271024 position before using the 1M EPROM adapter.

(2) FA Writer

When the FA Writer is used with the PMC–SA1, –SA2 or –SB, the EPROM adapter (1MB) for the FA Writer is required. When the FA writer is used with the PMC–RC, the ROM module adapter is required.



# **4.1**Various operations of FAPT ladder are done on the specified screen.**GENERAL**Fig. 4.1 shows the relation between various operations and corresponding screens.



Fig. 4.1 (a) Relation between various operations and screens

#### B-61863E/14

## 4. OPERATION



Fig. 4.1 (b) Outline of operation

# 4.2 PREPARATION BEFORE OPERATION

4.2.1 System Floppy	The system floppy disk contains the system of FAPT LADDER for PMC–SA1/SA2/SB/SB2/SC.			
4.2.2 Limitations with the	To apply the FAPT LADDER system for PMC–SA1/SA2/SB/SB2/SC to the SYSTEM P Mate will overlay each of the following functions.			
SYSTEM P Mate	• The function to display the ladder diagram on the screen and output on an external printer, which is operated using the menu numbers 03 F4 or 04 to F10.			
	• The function to input/output the ROM formatted program and make its comparison, which is operated combining the menu numbers 03, 05 or 07 and F2, F7, F8, F9, F13 or F14.			
	• Ladder diagram direct editing, which is operated by pressing <r1> key on the R key menu screen and executing ladder diagram direct editing.</r1>			
	The SYSTEM P Mate has less memory than the SYSTEM P Mark II and cannot load the system program on the system floppy disk at a time. The remainder left unloaded will be loaded automatically when each of the functions above is used. However, only in the case the system floppy disk has not been installed into the drive, the message "MOUNT SYSTEM FLOPPY DISK" is displayed as follows:			
	SET SYSTEM FD & KEY I 'OK' OR 'NO' FDD =OK ODRIVEJ (VOL =01) FDD =			
	Install the system floppy disk into the drive #0 or #1 and key in 'OK 0' or 'OK 1'. If the system floppy disk is installed into the drive #0, it is possible to key in only 'OK' without specifying the drive number.			
4.2.3 Loading of Floppy	FAPT LADDER system programs are loaded into the floppy. Also, sequence programs can be written from SYSTEM P series into the floppy or input from the floppy.			
	The loading method of the floppy is described in detail in the operator's manual for SYSTEM P series.			
	The following describes the loading direction of the floppy.			
	Loading direction of floppy			



Fig. 4.2.3 Loading direction of floppy

# 4.2.4 FAPT LADDER System Floppy Loading

- (1) Turn on the SYSTEM P series power supply.
- (2) Set the system floppy or prepared exclusive system floppy into the floppy disk.
- (3) Continue depressing <LOAD> key for 2 to 3 seconds on the keyboard.
- (4) The system loading is started. After this system loading, "FAPT LADDER" is displayed on the CRT screen and R key menu also appears.

This R key menu screen is shown in Fig. 4.2.6.

After this screen is displayed, take out system floppy or exclusive system floppy.

Fí Lái	APT DDER	,
*** P110	-MODEL RB/RC ····	
Edition 01.1 -91.02.25-	Copyright (C) 1991, FANNE LTD.	
PRESS R KEY		
F0 ; FROGRAMMEF R2 ; (UNUSED)	R1; (UNUSED) R3; REQ KEY	

Fig. 4.2.4 R key menu screen

## 4.2.5 Programmer Menu Screen

A programmer menu screen (hereinafter called menu screen) is displayed by pressing <R0> key from the R key menu screen. Key in a menu number to be executed Fig. 4.2.5 shows the menu screen.

The parameter setting screen is displayed by pressing <R0> key just after loading the 1/2 system floppy.

Set parameters as required, referring to 4.2.6.

Proceed to the menu screen by pressing <NL> key.

Parameters are displayable and settable from the menu screen, too.

Programmer menu screen (The programmer menu and function keys are displayed.)

The following figure shows the screen to be displayed when the programmer key (R0) is pressed.

The programmer menu, function keys with I/O indication, and statuses are displayed on this screen.



Fig. 4.2.5 Programmer menu screen

## 4.2.6 Parameter Setting and Display

Set parameters before inputting a sequence program without fail. Set necessary parameters by changing from the menu screen to the parameter setting screen (Fig.4.2.6), provided that the parameter setting screen is automatically displayed just after loading the system floppy.

(Operation in step 1 is not required in the procedure below.)

1 Key in menu number "10 <NL>" from the menu screen.

Turn off all F keys. The screen is switched, and the parameter setting screen shown in Fig. 4.2.6 is displayed. The initial value of each parameter is as shown in Fig. 4.2.6.

KEY IN ONE OF THE FOLLOWING NO.S WHICH YOU WANT TO SET PARA, S. NO. ITEMS CURRENT PARAMETERS 01 (UNUSED) ; BINARY 02 COUNTER DATA TYPE 03 OPERATOR PANEL ; NO KEY/LED ADDRESS ; KEY/LED BIT IMAGE ADRS. 1 ; ; PMC-RC 04 PMC TYPE 05 LANGUAGE ORIGIN ; 000000H 06 (UNUSED) 07 LADDER EXEC. ; 100% (1-150%) 08 (UNUSED) ; NO 09 IGNORE DIVIDE CODE 10 (UNUSED) ; 00 NOTHING TO SET ; ROM WRITER=FA WRITER NO.=

#### Fig. 4.2.6 Parameter setting screen (PMC–RC)

- 2 Key in "00 <NL>" to proceed to the menu screen, if displayed parameters are employed as they are.
- **3** Set parameters according to the following procedure when it is necessary to change the displayed parameters.

No operation is required for an item in which no change is required.

- a) Set a counter data type. The initial value is set to the binary format.
  - 1 Key in "02 <NL>"
  - 2 Select a binary or BCD notation, and key in the corresponding number "@@<NL>".
- b) OPERATOR PANEL

Specifies whether the operator's panel is used.

The initial value is already set to NO (unused).

- 1 Enter 03 and press the <NL> key.
- 2 The following message appears at the lower left of the screen.

EXAMPLE 0:NO, 1:YES OP PANEL=

**3** To disable the operator's panel, enter 0 and press the <NL> key. To enable the operator's panel, enter 1 and press the <NL> key.

- 4 Selecting YES in step 3 displays the following message: SET KEY/LED ADDRESS (KEY ADRS, LED ADRS.) ADDR=
- 5 Enter a Y-address to specify the KEY address and a Y-address to specify the LED address. For example, enter X0,Y0 and press the <NL> key.
- 6 Entering data as shown above displays the following message:

SET KEY/LED IMAGE ADDRESS (KEY ADRS, LED ADRS.) ADDR=

- 7 Enter addresses other than X- and F-addresses. For example, enter R0,R10 and press the  $\langle NL \rangle$  key.
- c) Select the type of PMC.

The initial value has been set to the PMC–SB or –SA1.

- **1** Type 04 and press the <NL> key.
- 2 The following message appears at the lower left of the screen:

EXAMPLE 0:PMC–SB, 1:PMC–SC PMC TYPE= or EXAMPLE 0:PMC–SA1, 1:PMC–SA2 PMC TYPE=

- **3** To select the PMC–SB or –SA1, enter 0 and press the <NL> key. To select the PMC–SC or –SA2, enter 1 and press the <NL> key.
- 4 When the type of PMC is changed, all data items including ladder data are cleared. The following message is displayed for confirmation:

CLEAR ALL DATA TO CHANGE PMC TYPE (0:NO, 1:YES) CLEAR/KEEP=

- 5 To cancel changing the type of the PMC, enter 0 and press the <NL> key. To change the type of the PMC, enter 1 and press the <NL> key.
- d) LANGUAGE ORIGN (for PMC–SC only)

The initial value is already set to 0.

- 1 Enter 05 and press the <NL> key.
- 2 Enter @@@@@@ (hexadecimal) and press the <NL> key to specify the first address of the TCB in a C program.
- e) LADDER EXEC (only for PMC–SC)

The parameter value for LADDER EXEC is fixed to 100% for the PMC–RB. For the PMC–RC, the parameter value can be set as follows.

- **1** key in "07 <NL>".
- 2 Key in "@@@ <NL>" by numeric characters within a range of 1% to 150%.

After setting, key in "@@ <NL>" or "<NL>" to set the menu screen.

This parameter is not supported for PMC–SA1, PMC–SA2, PMC–SB or PMC–SB2.

f) IGNORE DIVIDE CODE (only for PMC–SB/SC)

It is possible to select whether to execute a ladder program by dividing it into smaller units or without dividing it.

This parameter can be specified as follows for PMC–SB and PMC–SC:

- 1 Enter 09 and press the <NL> key.
- 2 To execute the ladder program by dividing it into smaller units, enter 0 and press the <NL> key. To execute it without dividing it, enter 1 and press the <NL> key.

This parameter is not supported for PMC-SA1 or PMC-SA2.

The ladder program is always executed without being divided.

# 4.3 PROGRAM EDITING

4.3.1 Data Display and Setting (Title, Symbol, Ladder Program, Comment, Message, I/O Module) Display the EDIT screen by keying in "1 <NL>" from the menu screen. Press <R0> by necessary times until a desired screen appears from the title to I/O module. The screen is switched in the sequence shown in Fig. 4.3.1, each time <R0> key is pressed.

Individual screens are reset to the menu screen by "E <NL>". In this paragraph, only the input and editing operation of each data from the keyboard is described.

For the I/O operations using a paper tape or a floppy, see 4.4 and 4.5.

(1) Title data (title data list screen).

Set the following data on sequence program as a comment.





/		
[	*** TITLE DATA LIST ***	
	01 MACHINE TOOL BUILDER NAME	
	02 MACHINE TOOL NAME	
	03 PMC & NC NAME FANUC PMC-MODEL RB & F16MA	
	04 PMC PROGRAM NO.	
	05 EDITION NO.	
	06 PROGRAM DRAWING NO.	
	07 DATE OF PROGRAMMING	
	08 PROGRAM DESIGNED BY	
	09 ROM WRITTEN BY	
	10 REMARKS	
	PMC CONTROL PROGRAM SERIES : 4061 EDITION : 01	
	MEMORY USED : 00.0 KBYTE SCAN TIME : 008 MSEC	
	0003 ALTERED EDIT	
١		

Fig. 4.3.1 (b) Title data list screen

a) MACHINE TOOL BUILDER NAME Set the machine tool builder name (max. 32 characters). Key in "A1<u>@@@</u>......@@@@ NL". - Machine tool builder name to be set Example) "A1 ***MACHINE(LTD) NL" b) MACHINE TOOL NAME Set the machine tool name (max. 32 characters). Key in "A2<u>@@@</u>......@@@@ NL". - Machine tool name to be set Example) "A2 ***MACHINE NL" c) CNC & PMC NAME Set the CNC and PMC name (max. 32 characters). Key in "A3 @@@.....@@@@ NL". - NC and PMC name to be set Example) "A3 F16MA.&.PMC-N NL" d) PMC PROGRAM NO. Set the sequence program number (max. 4 characters). Key in "A4 @@@@ NL". - Number to be set Example) "A4 0001 NL" e) EDITION NO. Set the edition number (max. 2 characters). Key in "A5<u>@@</u> NL". - Edition number to be set Example) "A5 G NL" f) PROGRAM DRAWING NO. Set the sequence program drawing number (max. 32 characters). Key in "A6<u>@@@</u>.....@@@@ NL". - Drawing number to be set Example) "A6 0001-0002-000A NL" g) DATE OF PROGRAMMING Set the sequence programming date (max. 16 characters). Key in "A7<u>@@</u>.....@@ NL". - Date to be set Example) "A7 1990.10.23 NL"

#### h) PROGRAM DESIGNED BY

Set the sequence program designer name (max. 32 characters).

Key in "A8 @@@.....@@@@ NL".

- Name to be set

Example) "A8 MR.***&MISS *** NL"

i) ROM WRITTEN BY

Set the name of the programmer who wrote a program into ROM cassette (max. 32 characters).

Key in "A9 @@@.....@@@ NL".

- Name to be set

Example) "A9 MR.***&MISS *** NL"

j) REMARKS

Set remarks (memo) (max. 32 characters).

Key in "A10 @@.....@@ NL".

— Remarks to be set

Example) "A10 MEMO-COMMENT NL"

Set title data about all items in the above format for both entry and alteration.

All characters are settable so long as they can be keyed in from the SYSTEM P series keyboard. Set easy-to-understand data about individual items.

(2) Symbol and comment data (SYMBOL & COMMENT LIST screen).

A symbol means a signal name to be attached to each PMC I/O signal. The comment data is a comment statement of the signal name.

The symbol is optionally settable within maximum 6 characters, while the comment data are optionally settable within maximum 30 characters.

a) Input from keyboard (Insert)

Key in "<u>G0.1 SYMNAM COMMENT</u> NL".

Mode selection(IS..., AS...) and line selection (I..., A...) need not be specified when symbols or comment data are input or edited. Also addresses (G, F, X, Y,...) can be entered in any sequence.

b) Alter

The operation is completely the same as that described in 2) a).

c) Insert

The operation is completely the same as that described in 2) a).



Fig. 4.3.1 (c) Message data list screen

Maximum 255 characters are entered to one address as message data. Input message data every maximum 32 characters/line by dividing them into 8 lines.

a) Input and alter from keyboard

Set message data in the alter format for both entry and alter. All characters are settable so long as they can be keyed in from the SYSTEM P series keyboard.

Key in message data every line in the following format.

"A @@@ MESSAGE-DATA1 NL".

└── Message data (maximum 32 characters) Line number (maximum 3 digits)

A means alter.

b) Delete

Delete message data every line in the following format. Key in "D@@@ NL".

- Line number to be deleted

c) Search

Search message data by address.

"A@@.@ NL"

- Address of message data to be searched

(4) I/O module data (I/O MODULE DATA LIST screen)

I/O module data are used for determining addresses in a sequence program of each I/O module.



Fig. 4.3.1 (d) I/O module data list screen

a) Input and alter from keyboard Set I/O module data in the following format when inputting or altering them from the keyboard. Key in I/O module data in the format of: "@@@@@@@@@@@@NL" I/O module name (maximum 5 digits) Slot number (maximum 2 digits within a range of 1 to 10) Base number (1 digit within a range of 0 to 3) Group number (1 digit within a range of 0 to 15) Address (input X0, Y0, ...)

— <u>994</u> —

b) Delete

Delete I/O module data every address by specifying it as follows: Key in "@@@@ NL".

Address of I/O module data to be deleted (input X0, Y0, ...)

## CAUTION

- 1 If the same slot number is specified when the group and base numbers are equal to each other, alarm No. 88 occurs.
- 2 If an output module is specified at an input address or an input module is specified at an output address, alarm No. 87 occurs.
- 3 If a module is set doubly to a preset address, alarm No. 81 occurs.

Example) When two IO modules, b are set as shown in the following figure;

address group base slot name



## 4.3.2 Programming from Keyboard

Input a sequence program from the keyboard.

Set the EDIT screen (LADDER PROGRAM LIST screen).

Press menu number "1 <NL>" on the menu screen, or press <R0> key on the symbol or I/O module screen. Turn off all F keys at this time.

Key in "IS0 <NL>" (Insert Succession) to set the sequential insert mode, and then, input a sequence program.

"*IS MODE*" is displayed at the lower right part of the screen. key in desired instructions sequentially in the following format.

(Key in sequence)

1 IS0 <NL> (Sequential input start command)

 $\rightarrow$  *IS MODE* is displayed at the lower right part of the screen.

- 2 R X0.1 <NL>
- 3 W R1.1 <NL>
- 4 IE <NL> (Sequential input end command)
  - → *IS MODE* display disappears from the lower right part of the screen.

#### NOTE

1 Instructions to be keyed in are entered by abbreviated symbols as shown above for the purpose of preventing a key-in failure and improving the operability by reducing the number of key-in times. It is also allowable to input these instructions by using their full names, like "RD X0.1 <NL>". Table 4.3.2 shows the correspondence between abbreviated symbols and full names.

2 No severe format checking is performed for mnemonic program. For example, the following program may be correct with mnemonic programming.

However this program cannot be displayed as ladder diagram nor printed out on the printer.

Usually do not program as shown below:



Input format from keyboard (Simple symbol)	Display format on screen (Full name)		
R X0.1	RD X0.1		
RN X0.2	RD. NOT X0.2		
RNS X0.3	RD.NOT.STK X0. 3		
W R0.4	WRT R0.4		
WN R0.5	WRT.NOT R0.5		
O Y1.0	OR Y1.0		
ON Y1.1	OR.NOT Y1.1		
OS	OR. STK		
AG2. 0	AND G2.0		
AN G2.1	AND.NOT G2.1		
AS	AND. STK		
Т 5	TMR 5		
D F0	DEC F0		
S 5	SUB 5		
P 1234	(Parameter)		

Table 4.3.2 Keyboard input format and screen display format

Correct a generated sequence program by alter operation.

Set the EDIT screen (LADDER PROGRAM LIST screen) first and display the generated source program. Turn off all F keys at this time.

a) Alter every instruction

Key in "<u>A@@@@@ R X0.1</u> NL"

Line number of the instruction to be altered (maximum 5 digits)

A means alter.

- b) Sequential alter
  - i) Key in "AS@@@@@ NL" (Alter Succession) to set the sequential alter mode.

@@@@@@: Line number to be sequentially altered (maximum 5 digits) "AS MODE" is displayed at the lower right part of the screen.

- ii) Instructions are sequentially altered starting with the Line specified by @@@@@@, each time the key-in operation is done in the "R X0.1 <NL>" format.
- iii) After sequential alter, key in "AE <NL>" (Alter End).

Example) Example of sequential alter of sequence program For altering all step numbers 20 to 23;

(Key in sequence)

1 AS20 <NL> (Sequential alter start command)

- → *AS MODE* is displayed at the lower right part of screen.
- 00020 RD Y0.1 2 R Y0.1 <NL>

4.3.3 Alter

— <u>997</u> —

```
00021 WRT R0.1 3 W R0.1 <NL>

00022 RD F1.1 4 R Y1.2 <NL>

00023 WRT R1.1 5 W R1.2 <NL>

6 AE <NL> (Sequential alter end

command)

\rightarrow *AS MODE* display disappears

from the lower right part of the

screen.
```

c) Wiring change function

<u>All of address</u> used in Ladder Program is changed to a <u>new</u> <u>address</u> independently of a command. Only bit address can be changed.

Type in '<u>CA Address 1 Address 2</u> <NL>

(Symbol can not be changed) A new address

└── A previous address to be changed

Abbreviation of CHANGE ALL

Example) 'CA R0.1 R1.2 <NL>' — All "R0.1" used in Ladder Program is changed to "R1.2".

## CAUTION

If an address is specified which can not be changed to a new address, an alarm 09 occurs when the specified line will be changed. In that case, previous lines correctly changed to that line can be acceptable.



As shown above, an alarm No. 09 occurs when a ladder program i is changed by an operation of **1** and a ladder program ii will be produced.

(Special use of wiring

change function)

Then, it may be impossible to return a ladder program ii to a ladder program i by an operation 2.

<u>All address</u> used in ladder program of specified line number of subsequent, is changed <u>a new address</u> independently of a command.

Only bit address can be changed.



lower right part of screen.

00020 RD Y200.0 2 R.S R200.1 <NL> 00021 WRT R300.7 3 R.S R200.2 <NL> 4 R 5 <NL> 5 P 9 <NL> 6 IE <NL> (Sequential insert end command)  $\rightarrow$  *IS MODE* display disappears from the lower right part of the screen. 4.3.5 i) Delete every instruction Delete Key in "D@@@@@ NL". Line number to be deleted (maximum 5 digits) D means delete. ii) Sequential delete Key in "D@@@@@@@@ NL". Line number to complete delete (maximum 5 digits) Line number to start delete 4.3.6 Search a sequence number by a line number or instruction **Location Search** i) Search by line number Key in "L@@@@@ NL". Line number to be searched (maximum 5 digits) L means location search. ii) Search by instruction (Search by address) Key in "L@@@@@ R X0.1 NL". - Instruction to be searched Line number with which the search is to be started CAUTION Input data after changing the symbol display into address display by passing R1 key, if the address of the instruction to be searched is defined by a symbol and displayed by the symbol.

iii) Search by instruction (Search by symbol)

Key in "L@@@@@ R ACT NL".

 Instruction to be searched (ACT: Symbol name)

Line number with which the search is to be started

## CAUTION

This search applies to such a case as the address of the instruction to be searched is defined by a symbol and the symbol is displayed.

iv) Search by the bit address or its symbol name

The specified address (only bit address) or its symbol name is searched from the specified line number independently of a command.

Type in '<u>L@@@@</u> ????? NL'.

— Bit address or its symbol name

Search start line number

Example) 'L1 R1.0 NL'

- Start searching bit address "R1.0" from 1st line.
   'L7 SMB NL'
- Start searching symbol name "SM BL" defined at bit address from 7th line.
- v) Continuous search

A specified command, address (only bit address) or its symbol name is searched from 2nd line displayed on the screen.

Type in 'L <u>?????</u> NL'.

Command, bit address or its symbol name to be searched

## Type in '<u>F</u> NL'.

- FIND: Search the same command, bit address or its symbol name as that searched just before, from 2nd line displayed on the screen.

Example) 'L R R0.1 <NL>'

- Search the command "RD R0.1" from 2nd line displayed on the screen.
- 'L R0.1 <NL>'
- Search the bit address "R0.1" from 2nd line displayed on the screen.
- 'L SYMBOL <NL>'
- Search the symbol name "SYMBOL" defined at bit address from 2nd line displayed on the screen.

## 4.3.7 Display of Ladder Diagram

The ladder diagram can be displayed on the programmer function EDIT screen.

Set the screen to EDIT screen (LADDER PROGRAM LIST)

- a) Turn on F4 key.
- b) Depress <NL> key

The ladder diagram is displayed on the screen.

For displaying the sequence program in the mnemonic format from the ladder diagram, turn off F4 key, and depress <NL> key.

The ladder diagram at an optional point can be displayed by the step number search or instruction search method.

If a ladder diagram cannot be displayed on one screen, it can be displayed by scrolling it leftward, rightward, upward, and downward as shown in the following table.

c) Edition during LADDER diagram display

Sequence programs can be edited even on the LADDER diagram screen display, (This function is convenience when sequence programs are edited with seeing LADDER diagram print out list.) From 'EDIT=' in the LADDER diagram screen display, sequence programs can be edited by the same operation as in editing programs in the 'LADDER PROGRAM LIST' screen.

	Scroll direction	Кеу
Left	(Left ladder on screen is displayed.)	← NI
Right	(Right ladder on screen is displayed.)	6 NI → NI
Upper	(Upper ladder on the screen is displayed.)	1 NI
Lower	(Lower ladder on the screen is displayed.)	2 NI
Upper h	alfpage	R2 NI
Lower h	alfpage	R2 NI

Fig. 4.3.7 Ladder diagram display screen

## 4. OPERATION



# 4.3.8 Help Screen

Editing command explanation screen can be displayed from Ladder program edition screen. (LADDER PROGRAM LIST or LADDER DIAGRAM)

key in 'H <NL>' to display the following screen.

```
*** HELP LIST ***
                                            (@@@@@=SEQUENCE NO.)
<INSERT>
 I@@@@@ OPERATION CODE
                                  : INSERT
 IS@@@@@
                                  : INSERT SUCCESSION START
 ΙE
                                  : INSERT SUCCESSION END
<ALTER>
 A@@@@@ OPERATION CODE
                                  : ALTER
 AS@@@@@
                                  : ALTER SUCCESSION START
 AE
                                  : ALTER SUCCESSION END
<DELETE>
 D@@@@(,@@@@@)
                                  : DELETE (SUCCESSION END)
<LOCATION SEACH>
 L@@@@@
                                  : SEQUENCE NO. SEARCH
  L@@@@@ OPERATION CODE
                                  : OPERATION CODE SEARCH
 L@@@@@ <ADDRESS OR SYMBOL>
                                  : ADDRESS OR SYMBOL SEARCH
 L OPERATION CODE
                                  : SEARCH FROM DISPLAY 2ND LINE
 L <ADDRESS OR SYMBOL>
                                  : SEARCH FROM DISPLAY 2ND LINE
                                  : FIND FROM DISPLAY 2ND LINE
 F
<CHANGE ALL ADDRESS>
  CA ADDRESS1 ADDRESS2
                                  :CHANGE ALL ADDRESS1 TO ADDRESS2
```

Key in "<NL>" to return it to ladder program editing screen.

Key is "E < NL >" (End) after editing a sequence program, and the EDIT screen is reset to the menu screen.

# 4.4 INPUT OF PROGRAM

4.4.1ReadSource ProgramsymbdesigP series	Read source programs (parameters, titles, symbols, ladders, messages, and I/O modules) from an input unit designated by an F key on the menu screen, and load them into SYSTEM P series memory.				
(1) P	aper tape format of source programs				
P p	aper tape format of source programs is of ISO code. No EIA code aper tape can be used.				
a	Parameter date				
Feed % @0 CR	% CR Feed				
b	) Title date				
Feed % @1 CR 1 MA	CHINE TOOL CR % CR Feed				
с	) Symbol date				
Feed % @2 CR F0.1 S	SYMBOL COMMENT CR % CR Feed				
d	) Ladder program				
Feed % @3 CR RD X0.	CR WRT WORK01 CR % CR Feed				
e) Message date					
Feed % @4 CR A0.0 MES WRT WORK01 CR	SSAGE = 1 $(CR)$ A24.7 MESSAGE = 24 $(CR)$ %				
CR ~ Feed					
f) I/O module date					
Feed % @5 CR X 0 0	0 5 ID32C CR % CR Feed				

- (2) Input method from PPR reader
  - 1 Turn on F1 key.
  - 2 Key in menu number "2 <NL>".
  - **3** The screen is switched, and the entry of a source program is started.
  - 4 After the source program has been normally entered, the screen is automatically reset to the programmer menu screen. If an error was detected during entry, "PART–" is displayed on the lower left part of the screen. Check error contents, and key in "E NL". The screen is reset to the programmer menu screen.
- (3) Entry method from floppy
  - 1 Turn on F2 key.
  - 2 Key in menu number "2 <NL>".
  - 3 The following message is displayed at the lower part of the screen.

SET FD & KEY IN "OK" "KILL" OR "NO" FD0=OK <DRIVE> <@NAME OR : NUMBER> FD0=

Fig. 4.4.1 shows the menu screen in the floppy entry mode.

[Screen when source programs are input form floppy]

```
SET I/O KEY & KEY IN ONE OF THE FOLLOWING NO.S WHICH YOU WANT.
             NO.ITEMS
             01 EDIT LADDER PROGRAM.
             02 INPUT LADDER PROGRAM FROM PTR OR FD.
             03 INPUT ROM DATA FROM FD. PMC-RAM OR ROM.
             04 OUTPUT LADDER PROGRAM TO PTP. FD OR PRINTER.
             05 OUTPUT ROM DATA TO FD. PMC-RAM OR ROM.
06 COMPARE LADDER PROGRAM WITH PTR OR FD.
             07 COMPARE ROM DATA WITH FD. PMC-RAM OR ROM.
             08 (UNUSED)
             09 CLEAR OF TITLE. SYMBOL. LADDER OR MESSAGE DATA.
             10 PARAMETER SET.
             00 END EDIT & DISPLAY.
             F1 : PTR (I) . F4 : GRP (O) SYMBOL =00.0KB
F2 : FD (I) . F10 : EPRT (O) LADDER =00.0KB
                                                   SYMBOL =00.0KB SCAN TIME-008MS
             F5
                  :
                     PRT (O)
                                                    MESSAGE=00.0KB
             F6
                     PTP (O)
                 :
             F7
                     FD
                          (0)
                                                    END SEO.NO=00000
                  :
             F8
                 : PMC (I/O)
                                                    ERR SEQ.NO=00000
             F9
                    ROM (I/O)
                                                    ERR BLOCK =00000
             SET FD & KEY IN 'OK'
                                       'KILL' OR 'NO'
             FD0 = OK '@FILE NAME'
                        <DRIVE><@NAME OR NUMBER>
             NO.=
Example
```

Key in file names to be input from floppy as shown in the example.

#### Fig. 4.4.1 Floppy input menu screen

4 Insert the floppy into the disk, and enter the following data. Characters in <> need not be keyed in.

#### OK @LADDER1 NL

— File name (provisional file name)

5 The screen is switched, and the entry of source programs is started from the floppy.

- 6 The following procedure is the same as in 4.4.1 2) 4.
- 7 A file name is inputtable up to maximum 17 characters. All characters on the SYSTEM P series keyboard are employable for this entry. The kinds of capitals are not limited.

"@" (at mark) shows a file name input identifier. Key in it just before the file name as shown in example \$\$ without fail.

#### CAUTION

If sequence program instructions are sequentially entered while a sequence program is loaded in the SYSTEM P series memory, the instructions are entered into the SYSTEM P series memory following the previously loaded program. Clear SYSTEM P series memory, if a new program is entered from the floppy. (see 4.7) The SYSTEM P series memory is cleared by turning off the

SYSTEM P series power supply.

## 4.4.2 ROM Format Program

(1) Transfer of sequence program from the PMC-SA1/SA2/SB/SC

The created sequence program is transferred from the PMC-SA1/SA2/SB/SC.

First, connect the SYSTEM P Series and the CNC with a Reader/Puncher interface cable. (Refer to Appendix 1 for details of the cable.) For the method and location of connection, refer to the section "3.2 Configuration devices and their connection". In the following procedure, operations 1 to 6 are NC side operations.

The keys enclosed in [ ] are soft keys.

- 1 Pressing soft keys [SYSTEM] and [PMC] displays the PMC screen. Steps 2 to 4 below must be performed when [I/O] is not displayed on the PMC screen. For a 9–inch CRT, press soft key [NEXT] to check that [I/O] is not on the screen.
- **2** Pressing soft keys [PMCPRM] and [KEEPRL] on the PMC screen displays the keep relay setting screen.
- 3 Set  $\underline{K17.1}$  to 1 on the keep relay setting screen.
- 4 Pressing soft key [RETURN] displays the PMC screen.
- **5** On the PMC screen, pressing soft key [I/O] displays the I/O screen. For a 9–inch CRT, press soft key [NEXT] before pressing soft key [I/O].
- 6 Pressing soft key [EXEC] on the I/O screen puts the system in the EXECUTING state.
- 7 Turn on the F8 key on the SYSTEM P series menu screen. (Turn on the F12 key at the same time when the C–language program is included.)
- 8 If the menu number '3 [NL]' is keyed in, the message shown below will be displayed. PMC–SA1/SA2/SB/SC is not displayed. Key in the type of ROM module to be used from now on. (Refer to Note 1 when selecting ROM module B, C or D.)

SELECT THE TYPE OF ROM MODULE ACCORDING TO THE FOLLOWING NO. ROM MODULE 0:A 1:B, 2:C, 3:D

NO.=

By means of the above-described operations, the program transfer is started. The transfer screen is displayed on the SYSTEM P Series screen and the transfer counter counts. The screen returns to the menu screen after the end of transfer.

- (2) Input from a floppy disk
  - 1 Turn on the F2 key. (Turn on the F12 key at the same time when the C language program is included.)
  - 2 If the menu number '3 [NL]' is keyed in, the message shown below will be displayed. PMC–SA1/SA2/SB/SC is not displayed. Key in the type of ROM module to be used from now on. (Refer to Note 1 when selecting ROM module B, C or D.)

SELECT THE TYPE OF ROM MODULE ACCORDING TO THE FOLLOWING NO. ROM MODULE 0:A 1:B, 2:C, 3:D NO.=

**3** The following message is displayed at the lower left part of the screen.

SET FD & KEY IN 'OK', 'KILL' OR 'NO' FD=OK <@FILE NAME> FD0=OK <DRIVE><@NAME OR : NUMBER> FD0=

- 4 Insert the floppy into the disk and enter the following data
- 5 'OK<u>@LADDER2</u> [NL]

- File name

- **6** The screen is switched and the ROM format program is started from the floppy disk.
- 7 After reading is ended, the screen is automatically changed to the program menu screen if no problem occurs. When an error is detected during reading, 'PART' = is displayed on the left lower part of the screen. Check the error and key in 'E [NL]' to return the screen to the program menu screen.
- (3) Method of inputting from the FA writer and PMC writer
  - 1 Check the setting of the ROM writer. (See Section 3.4, "Setting of I/O Device.")
  - **2** Put the FA Writer in the REMOTE mode by the [REMOTE/LOCAL] key before using it.
  - **3** Turn on the F9 key. (Turn on the F12 key at the same time when the C language program is included.)
  - 4 If the menu number '3 <NL>' is keyed in, the message shown below will be displayed. PMC–SA1/SA2/SB/SC is not displayed. Key in the type of ROM module to be used from now on. (Refer to Note 1 when selecting ROM module B, C or D.)

SELECT THE TYPE OF ROM MODULE ACCORDING TO THE FOLLOWING NO. ROM MODULE 0:A 1:B, 2:C, 3:D

```
NO.=
```

**5** The screen is switched and the message shown below is displayed.

SET EPROM OR ROM MODULE & KEY IN 'OK' OR 'NO' KEY IN=

- 6 Check the above message. For the PMC–SA1/SA2/SB, insert the EPROM for the PMC into the FA Writer or PMC Writer. For the PMC–RC, insert the ROM module for the PMC into the FA Writer or PMC Writer. Note, however, that ROM modules are not available with the PMC Writer.
- 7 Key in 'OK <NL>' or 'NO <NL>'.

When 'OK <NL>' is keyed in, the sequence program written into the EPROM and ROM module for PMC is entered into P–G memory.

The screen returns to the menu screen if it ends with no problems occurring.

When 'NO <NL>' is keyed in, the screen returns to the menu screen.

## CAUTION

When using the SYSTEM P Mate, if ROM module B, C or D is selected, overlay occurs. When cassette B or C is selected, set the work floppy disk for external memory in drive 1.

# 4.5 OUTPUT OF PROGRAM

## 4.5.1 Source Program

By selecting '04 <NL>' (OUTPUT LADDER PROGRAM) from menu no.4, the following detail menu is displayed.

```
SET I/O KEY & KEY IN ONE OF THE FOLLOWING NO.S WHICH YOU WANT.
NO.
     ITEMS
01
    OUTPUT ALL DATA.
    OUTPUT SYSTEM PARAMETER.
02
    OUTPUT TITLE DATA
03
04
    OUTPUT SYMBOL DATA
05
    OUTPUT MESSAGE DATA.
06
    OUTPUT I/O MODUL DATA.
07
    OUTPUT LADDER PROGRAM (MNEMONIC).
    OUTPUT LADDER DIAGRAM (ONLY FANUC PRINTER).
OUTPUT CROSS REFERENCE (SEQUENCE NO.)
80
09
    END
00
               , F10 : FANUC PRINTER (O)
F5 : PRT (0)
F6 : PTP (O)
                  F13 : CROSS REFERENCE (NO.8)
               ,
F7 : FD (0)
NO. =
```

Select a desired data and device from the above details menu screen by combining the menu numbers and F keys.

#### (1) OUTPUT ALL DATA

All data of system parameters, titles, symbols, messages, I/O modules and ladder programs (source format) are output to a device specified by an F key.

Turn on an F key corresponding to the device to be output, and key in detail menu number '01 <NL>'.

If F10 key is turned on, all data are output to the FANUC printer (external printer) and the ladder diagram is output last. If F13 key is turned on furthermore, the ladder diagram is output with a cross reference.

#### (2) OUTPUT SYSTEM PARAMETER

System parameter data are output to a device specified by an F key. Turn on an F key corresponding to the device to be output, and key in detail menu number '02 <NL>'.

#### (3) OUTPUT TITLE, DATA

Title data are output to device specified by an F key. Turn on an F key corresponding to a device to be output, and key in detail menu No. '03 <NL>'.

#### (4) OUTPUT SYMBOL DATA

Symbol data are output to device specified by an F key. Turn on an F key corresponding to a device to be output and key in detail menu number '04 <NL>'.

The screen is switched and the following display appears.

OUTPUT = 'L@@@@ (,@@@@)' OUTPUT =

Specify the output range by line numbers as follows.

Example)

Key in 'L1, 100 NL'

— Output end liner number (If this parameter is omitted, data are output to the last one.)

Output start line number

#### (5) OUTPUT MESSAGE DATA

Message data are output to a device specified by an F key. Turn on an F key corresponding to a device to be output, and key in detail menu number '05 <NL>'.

The screen is switched and the following display appears.

OUTPUT = 'A@@.@ (,@@.@)'

OUTPUT =_

Specify the output range by addresses as follows.

Example)

Key in 'A1.0,10.1'

— Output end address (If this parameter is omitted, data are output to the last one.)

Output start address

#### (6) OUTPUT I/O MODULE DATA

I/O module data are output to a device specified by an F key. Turn on an F key corresponding to a device to be output, and key in detail menu number '06 <NL>'.

(7) OUTPUT LADDER PROGRAM (MNEMONIC)

Ladder program (source format) data are output to a device specified by an F key.

Turn on an F key corresponding to a device to be output, and key in detail menu number '07 <NL>'.

The screen is switched and the following display appears.

OUTPUT = 'L@@@@ (,@@@@)' OUTPUT =

Specify the output range by line numbers as follows.

Example)

Key in 'L1,100 NL'

— Output end line number (If this parameter is omitted, data are output to the last one.)

Output start line number

#### (8) OUTPUT LADDER DIAGRAM (ONLY FANUC PRINTER)

A ladder diagram is output to the FANUC printer (external printer). Key in detail menu number '08 <NL>', and then, turn on F10 key. Turn on F13 key furthermore, if it is desired to output the ladder diagram with a cross reference.

The screen is switched and the following display appears.

OUTPUT = 'L@@@@ (,@@@@)' OUTPUT =

Specify the output range by line numbers as follows. (Partial output is also possible.)

#### Example)

Key in 'L1,100 <NL>'

— Output end line number (If this parameter is omitted, data are output to the last one.)

Output start line number



## CAUTION

- It takes time more or less from the end of operation on end to the start of printer operation when outputting the LADDER diagram with cross reference. (EXECUTING is displayed on the screen.) This time depends upon the size and complexity of sequence programs. The cross reference is displayed by the page number and the line number of the LADDER diagram every contact. See Appendix printout example.
- If R1 key is pressed when each data is being output to the FANUC printer (External printer), the output is cancelled.

#### (9) OUTPUT CROSS REFERENCE (SEQUENCE NO)

Addresses (symbols, comments) are printed with cross reference Nos. by FANUC external PRINTER.

These Nos. correspond to the Mnemonic format list (screen) or Ladder diagram (RD command line number).

Key in above detailed memo No. '09 <NL>' and turn F10 key on. The screen changes to display the key in example and 'ADDR=' as below.

Key in addresses to be output according to examples.

	Key in example	Address to be output
ALL	ALL <nl></nl>	All addresses (G,F,Y,X,A,R,T,K,C,D in order)
Address initial	R <nl></nl>	All address with the specified initial
Bit address	X1.0 <nl></nl>	Only bit address specified address
Byte address	R58 <nl></nl>	Bit 0 – 7 of specified
Addressrangespecification	F8.0, X7.2 <nl></nl>	Specified addresses in order of G,F,Y,X,A,R,T,K, C,D
	X0.2–END <nl></nl>	All address after specified address

,		***	CROSS	REFEREN	CE LIST	***		DACE-1
ADDRESS	SYMBOL		COMME	NT DAT	A			FAGE-1
G0000.0 653	*IT							
G0000.1 653	*CST							
G0000.4	*ESP							
22	568 *CD	901	912	1177	1189	1288	2800	
45	2802							
G0000.7	ERS							
3435	3512							
G0001.0	*AIT							
656								
				~				

#### NOTE

- 1 When the same address performs double writing,"* MULTIPLE COIL USED *" is displayed.
- 2 If the F10 key is set to OFF and output performed, the cross reference table is displayed on the screen.

# 4.5.2 Paper Command

A 12-inch chart is also applicable to the FANUC printer (external printer).

(The standard chart size is 11 inches.) Enter the command for changing the chart by the following operation.

- (1) Press [R3] key from the R key menu screen.
- (2) 'REQUEST=' is displayed at the lower left part of the screen.
- (3) Key in 'PAPER <NL>'.
- (4) The following message is displayed at the lower left part of the screen.
   KEY IN NUMBER OF PAPER LENGTH EXAMPLE 11–INCH;0,12–INCH;1.
   LINE NUM.=
- (5) Key in '0  $\langle NL \rangle$ ' for 11-inch chart, or '1  $\langle NL \rangle$ ' for 12-inch chart.

4.5.3	(1) Transfer of sequence program into PMC-SA1/SA2/SB/SC					
ROM Format Program	A generated sequence program is transferred into PMC–SA1/SA2/SB/SC. Connect SYSTEM P series to CNC by using a Reader/Puncher interface cable. (For this cable, see Appendix 1.) for the connection method and places, see Section 3.2. Steps 1 to 6 show the operation on the CNC side					
	<ol> <li>Pressing soft keys <system> and [PMC] displays the PMC screen. Steps 2 to 4 below must be performed when [I/O] is not displayed on the PMC screen. For a 9–inch CRT, press soft key [NEXT] to check that [I/O] is not on the screen.</system></li> </ol>					
	2 Pressing soft keys [PMCPRM] and [KEEPRL] on the PMC screen displays the keep relay setting screen.					
	3 Set K17.1 to 1 on the keep relay setting screen.					
	4 Pressing soft key [RETURN] displays the PMC screen.					
	<b>5</b> On the PMC screen, pressing soft key [I/O] displays the I/O screen. For a 9–inch CRT, press soft key [NEXT] before pressing soft key [I/O].					
	6 Pressing soft key [EXEC] on the I/O screen puts the system in the EXECUTING state.					
	7 Turn on F8 key from the SYSTEM P series menu screen. (Also turn on F12 key when the C language program is included.)					
	<b>8</b> Key in menu number "5 <nl>".</nl>					
	Now, the program transfer is started. In SYSTEM P series, the transfer screen is displayed and the transfer counter is counted up. After transfer, the screen is reset to menu screen. In CNC screen, the COUNTER display is counted up.					
*Procedure when a	i) When an alarm 31 occurs on SYSTEM P series screen;					
program cannot be transferred from SYSTEM P series to RAM of PMC	<ul> <li>Cause 1 : Reader/Puncher interface cable is defective.</li> <li>Remedy : Use the specified cable.</li> <li>Cause 2 : Reader/Puncher interface connector is not connected to correct channel SYSTEM P series.</li> <li>Remedy : Connect the connector correctly.</li> </ul>					
	<li>ii) When the transfer counter of SYSTEM P series screen is counted up and normally terminated, but data are not transferred to the PMC RAM correctly;</li>					
	<ul> <li>Cause 1 : Reader/Puncher interface connector is not connected to CNC.</li> <li>Remedy : Connect it correctly.</li> <li>Cause 2 : CNC screen is not set to "I/O of PMC" screen.</li> <li>Remedy : Set the I/O screen by the soft key.</li> <li>Cause 3 : An error occurs in ACI channel due to a certain cause.</li> <li>Remedy : Turn off the power supply once, and turn it on again.</li> </ul>					

- (2) Output method to floppy
  - **1** Turn on F7 key. (Also turn on F12 key when the C language program is included.)
  - 2 Set the floppy to the disk.
  - **3** Key in menu number "5 <NL>".
  - **4** The screen is switched and the following message is displayed:

SET FD & KEY IN "OK", "KILL" OR "NO". FD0= OK <INT OR ADD><P OR NP,></DATE,> <DRIVE>@NAME FD0 =

When loading data starting with the start of the floppy, specify INT. When loading data after the loaded files, specify ADD. After outputting all data, the screen is reset to the program menu screen. The menu screen is also reset by keying in "NO <NL>".

- (3) Method of outputting data to FA writer or PMC writer (EPROM for PMC/ROM module write)
  - 1 Check the setting of the ROM writer. (See Section 3.4, "Setting of I/O Device.")
  - 2 Put the FA writer in the REMOTE mode by the [REMOTE/LOCAL] key before using it.
  - **3** Turn on F9 key. (Turn on F12 key when the C language program is included.)
  - 4 Key in menu number "5 NL".
  - 5 The screen is switched to the title screen, and the following message is displayed.

SET EPROM OR ROM MODULE & KEY IN "OK" OR "NO". KEY IN =

Check the above message. For the PMC–SA1/SA2/SB, insert the EPROM for the PMC into the FA Writer or PMC Writer. For the PMC–RC, insert the ROM module for the PMC into the FA Writer or PMC Writer. Note, however, that ROM modules are not available with the PMC Writer.

6 Key in "OK <NL>" or "NO <NL>".

When "OK <NL>" is keyed in, data are output from the SYSTEM P series memory to the EPROM for PMC or ROM module. After normal end, the screen is reset to the menu screen.
```
*** TRANSFER ROM DATA TO PMC WRITER ***
01 MACHINE TOOL BUILDER NAME
02 MACHINE TOOL NAME
03 PMC & NC NAME
04 PMC PROGRAM NO
05 EDITION NO
06 PROGRAM DRAWING NO
07 DATE OF PROGRAMING
08 PROGRAM DESIGNED BY
09 ROM WRITTEN BY
10 REMARKS
PMC CONTROL PROGRAM SERIES : 4061 EDITION :01
MEMORY USED : 00.0 KBYT SCAN TIME : 008 MSEC
SET EPROM OR ROM MODULE & KEY IN 'OK' OR 'NO'
KEY IN =
```

(	*** TRANSFER ROM DATA TO PMC WRITER ***	
01	MACHINE TOOL BUILDER NAME	
02	MACHINE TOOL NAME	
03	PMC & NC NAME	
04	PMC PROGRAM NO	
05	EDITION NO	
06	PROGRAM DRAWING NO	
07	DATE OF PROGRAMING	
08	PROGRAM DESIGNED BY	
09	ROM WRITTEN BY	
10	REMARKS	
	PMC CONTROL PROGRAM SERIES : 4061 EDITION :01 MEMORY USED : 00.0 KBYT SCAN TIME : 008 MSEC	Displaymode
		BLANK : Blank check
	MODE=BLANK	PROGRAM : Write
	ROML=10 ROMH=EF MEM FF AD=000000	VERIFY : Compare
	ALARM=083	
	rolitr-	Frornumber
ا ت		Ensinamber
$\overline{)}$		
_		

Enter 'E NL', and restart from menu.

## 4.6 COLLATION OF PROGRAM

4.6.1 Collation of Source Programs	Enter source programs from the designated input unit, and compare them. The operation method is the same as source program entry, except that "6" shall be designated as the menu number.
riogramo	(1) Comparison with PTR
	1 Turn on F1 key.
	2 Key in menu number "6 NL".
	(2) Comparison with FD
	1 Turn on F2 key.
	2 Key in menu number "6 <nl>".</nl>
	<b>3</b> The screen is switched, and the following message is displayed.
	SET FD & KEY IN "OK", "KILL" OR "NO", FD0=OK <drive> &lt;@NAME OR : NUMBER&gt; FD0=</drive>
	Specify the file name to be compared.
	After normal end, the screen is automatically reset to the menu screen.
	Also, this menu screen is reset by keying in "KILL
4.6.2 ROM Format Program	Compare ROM format program by reading it from the specified input device. The operation method is the same as in ROM format program input, except that menu number "7" is specified.
	(1) Comparison with FD
	1 Turn on F2 key.
	2 Key in menu number "7 <nl>".</nl>
	<b>3</b> The following operation is the same as in $4.6.1 \ 2$ )– <b>3</b> and later.
	(2) Comparison with PMC–RAM
	Display the I/O of PMC screen on the CRT/MDI before executing the following operation.
	1 Turn on F8 key.
	2 Key in menu number "7 <nl>".</nl>
	Note when comparing P–G and PMC–RAM : The comparison between P–G and PMC–RAM should be performed immediately after the data transfer. (When the comparison is made after the output of ROM format data, the parity portion of data may become error.)
	(3) Comparison with EPROM for PMC and ROM module
	1 Turn on F9 key.
	2 Key in menu number "7 <nl>".</nl>
	The screen is switched, and the comparison of ROM program is started. After normal end, the screen is automatically reset to the menu screen.

## 4.7 DELETION OF PROGRAMS

Delete ladder programs, symbols, message, titles, and I/O module data being loaded into SYSTEM P series memory according to the following procedure.

- **1** Put the screen to menu screen.
- 2 Key in menu No. "9 < NL >".
- 3 The screen is switched, and the following message is displayed at the lower left part of the screen. See Fig. 4.7.

KEY IN "1,2,3,4 OR 5" OR "NO" CLEAR/KEEP =

4 Key in data number of the data to be deleted or key in "NO <NL>", if it is not desired to delete any data. After processing, the screen is automatically reset to the programmer menu screen.

```
KEY IN ONE OF THE FOLLOWING NO.S WHICH YOU WANT TO CLEAR DATA
NO. ITEMS
01 TITLE DATA
02 SYMBOL DATA
03 LADDER DATA
04 MESSAGE DATA
05 I/O MODULE DATA
06 ALL DATA CLEAR
KEY IN '1. 2. 3. 4. 5 OR 6 OR 'NO'
CLEAR/KEEP =
```

#### Fig. 4.7 Delection of sepuence programs

Example)

- i) When all title data are to be deleted; Key in "1 <NL>".
- ii) When all symbol data are to be deleted; Key in "2 <NL>".
- iii) When all ladder programs are to be deleted; Key in "3 <NL>".
- iv) When all message data are to be deleted; Key in "4 <NL>".
- v) When I/O module data are to be deleted; Key in "5 <NL>".
- vi) When all titles, symbols, ladders, messages and I/O module data are to be deleted;

Key in "6 <NL>".

vii) When no data are to be deleted;

Key in "NO <NL>".

# 4.8 SPECIAL USES OF THE R3 KEY

Key in  $\langle NL \rangle$  alone at the menu screen to display the R key menu screen. Key in R3 at the R screen, and the display 'REQUEST=' will appear at bottom left of the screen, making key inputs possible. Key in  $\langle NL \rangle$  on this screen to return to the R key menu screen.

R3 executes a large number of processings. For the FAPT LADDER system, however, note the following two points:

- (1) Floppy file name output
  - 1 Press R3 key at the R key menu screen.
  - 2 This will change the screen contents, displaying 'REQUEST=' at its left bottom.
  - 3 Key in FDLIST <NL>.
  - 4 The file name will appear on the CRT display. To print out the file name, turn on the F5 (printer) key in advance.
- (2) Change of I/O devices (for output to a printer other than that of PPR)
  - 1 Key in IO PRT, CN3, F5 <NL> while the screen displays 'REQUEST='. When the F5 key has been turned on in advance, the data is printed on the printer connected to connector CN3 on the SYSTEM P series rear side.

## 4.9 DIRECT EDITING BY LADDER DIAGRAM

4.9.1 Outline	Using the P–G Mate/Mark II software keys (in the case of P–G Mate, the F keys), sequence program creation and editing can be performed directly by the ladder diagram.
	In the following explanation, [P–G Mate] is called [Mate] and [P–G Mark II] is called [Mark II]. When it is possible to use this function, in the R key menu screen
	R1: EDIT
	is displayed. (In systems where [UNUSED] is displayed, it cannot be used.)
	The following items are present in the edit function.
	• Ladder diagram direct editing by software key and cursor (input, addition, deletion and substitution)
	• Copying, moving and deletion of multiple lines of the ladder
	• Optional relay and coil reference
	• Comment display on ladder diagram
4.9.2 Limitations in SYSTEM	<ul> <li>(1) This function operates only when the P–G Mate main unit is version 04 and later. (When the power supply is turned on, it is displayed in the lower right part of the initial screen.)</li> </ul>
P Mate	(2) The function keys <f keys=""> are used instead of the soft keys (P–G Mark II). In the description that follows, an explanation for the soft keys (P–G Mark II) is given. When P–G Mate is used, operate with the function keys. At this time, in order to make the F key respond and display the screen bottom line, the F key lamp illuminates to correspond to those items displayed with shaded characters on the screen.</f>
4.9.3	The program menu appears in order to operate this function.
Selection of Program Menu by Soft Keys	The program menu is displayed when the $\langle R1 \rangle$ key is pressed from the R menu screen. The program menu is displayed above the soft keys (in the case of P–G Mate, the function keys) as shown in the screen below, and gives significance to the keys.
	(1) Keyboard
	Refer to Section "3.3 SYSTEM P keyboard".
	(2) Relationship betweeen program menus and soft keys
	The relationship between the program menus and the soft keys is shown in the following for each function. These menus are changed by pressing the related keys. For menu contents, refer to the explanations described later. Utilize this figure when operating.



## 4.9.4 Sequence Program Input

In order to input the sequence program, press the  $\langle R1 \rangle$  key from the R key menu. The soft key menu program is displayed, and in the case that the sequence program has not yet been input, only the left and right vertical lines of the ladder diagram are displayed on the screen.

Start inputting a program with the screen in this state.

Input a ladder diagram program by moving the cursor to the desired input position using the cursor key.

The following description shows an example of the input of a program of basic instruction and a program of functional instruction.



(1) Basic instruction program input

1 Press the soft key [──] after moving the cursor to the start position.

Symbol [--+] is input at the cursor position and HORIZONTAL LINE ILLEGAL is displayed at the lower right part of the screen. This is a cautionary message which shows that the ladder diagram horizontal line is not yet completely created. Input the continuation address and bit data.

- 2 Press the <NL> key after inputting R0.1 using the keyboard. The address is set on the contact and the cursor shifts rightward.
- 3 Input A contact with address R10.2 by the above methods 1 and 2.
- 4 Input B contact R1.7.

Press the soft key [_____], input address R1.7, and then press the <NL> key. The address is set on the B contact and the cursor shifts rightward.

- 5 Press software key [- -] with the cursor position unchanged.
   A right horizontal line is automatically drawn, and a relay coil symbol is entered near the right vertical line.
- 6 Press the <NL> key after inputting address R20.2.

The cursor automatically shifts to the input start position of the next line.

7 Next, input the OR condition.

Press the soft key [____], input address X2.4, and then press the <NL> key. The address is set on the B contact and the cursor shifts rightward.

8 Press the soft key [-----] to input a horizontal line

When inputting the horizontal bar key [____], by keying in a numerical value and pressing this bar key, a horizontal line for the frequency will be drawn. However, this horizontal line will not be drawn over the LINE.

9 Because the upper right line OR is necessary, press the soft key[▲] and input the upper right vertical line to end.

#### NOTE

- 1 When the ladder program displayed on the screen is incomplete (when, for example, addresses have not been entered) or erroneous, the screen cannot be scrolled even when a page key is pressed. Before attempting to scroll the screen, therefore, ensure that the ladder program is complete and error-free.
- 2 Since 8 contacts + coil are specified to be inputtable per line from the screen, any more contacts in excess of this amount cannot be input. However, this restriction does not apply to a sequence program created with mnemonic format.

When a sequence program, transferred from the offline programmer to the PMC, exceeds the length which can be displayed on a single line, the program is displayed using two or more lines, linked with a continuation symbol.

This continuation symbol is not erasable by software key [----].

- 3 Below is shown an example with an error net, or part of it, erased with no error display.
  - 1 Case of multiple nets on 1 LINE



Downward from the net is erased

2 Case of multiple WRT results in 1 NET difference as shown in the diagram below.





#### (2) Case of functional instruction program input

To input a functional instruction, input the soft key [FUNCTN], and then input the functional instruction name or SUB number.

Further, when inputting a functional instruction, after keying in the functional instruction number, it does not matter if the [FUNCTN] key is pressed.

When you can not remember the instruction name or SUB number, the functional instruction table corresponding to the instruction symbol and SUB number can be displayed on the screen.

The functional instruction table is automatically displayed after inputting an incorrect instruction name or SUB number and then pressing the [FUNCTN] key, or by pressing the [FUNCTN] key only without inputting any other key.

In order to return from the functional instruction table to the original ladder diagram, press the [FUNCTN] key.



1	Input	a	control	condition.
---	-------	---	---------	------------

Press soft key [---], input the address and bit data, and then press the  $\langle NL \rangle$  key. The cursor shifts rightward.

2 Input an instruction

Press the soft key [FUNCTN], input SUB number 8, and then press the <NL> key. A functional instruction diagram appears as shown in the above figure.

3 Input an instruction parameter

Input the high rank 4 bit logic data of the first parameter, and then press the *<*NL> key. The cursor automatically lowers downwards. Input the three residual parameters in order.

4.9.5 Substitution of	The method of substituting a created sequence program is the same as that described earlier in Section 4.9.4.
Sequence Programs	Move the cursor to the program part you want to alter and input the change data.

4.9.6 Additions to Sequence	From the soft key program menu, press the soft key [COMAND] and operate with the soft keys shown below.
Programs	When you want to end the program menu shown below, press the soft key at the extreme left.

INSNET	DELNET	INSERT	ADRESS	SEARCH	COPY	MOVE		
		$\checkmark$						
INSNET	INSLIN	INSELM						

A sequence program is added in four ways on the ladder diagram as described below.

(1) Case of adding a relay contact in the horizontal direction



Move the cursor to the position where you want to add, and input te program by the method described in Section 4.9.4.



- **1** Move the cursor to the above position.
- 2 Press the soft key [▲] in order to erase the upper left vertical line. The upper left line, vertical to the cursor disappears.
- 3 Press the soft key [▲] in order to produce an upper right line vertical to the cursor. Then, press the soft key [—]. Both vertical and horizontal lines are created.
- 4 Shift the cursor to a line of contact addition position.
- **5** Press the soft key [--] to add contacts.
- (2) Adding a vertical line

For adding a vertical line as shown in the above diagram, the area to be added is required. In order to produce this area, shift the entire part after the part to be added by one line by moving the cursor to the ladder diagram within the dotted line range (an optional part is allowable) and then pressing the soft key [INSNET].

The lower ladder diagram shifts downward by one line, each time the [INSNET] key is pressed thereby producing the area to which a line is to be added.

If a surplus addition area remains unused after the addition processing ends (for example, if an area corresponding to 3 lines has been reserved when two lines have been added), there is no problem if the area is left remaining.



- **1** Move the cursor to the ladder diagram bounded by a dotted line.
- 2 Press the soft key [INSNET].
- **3** Pressing the [INSNET] key without keying in numeric values will cause one line to be inserted.

- 4 Pressing the [INSNET] key with keying in numeric values will cause the line to be inserted the number of times specified by the numeric value input.
- 5 After setting the cursor to a position to which you want to add, press the soft key [----]. After setting address data, press the <NL> key. The cursor shifts rightward.
- 6 Press the shift key  $[\land]$  to create an OR circuit.
- (3) Inserting the 1 NET sequence program LINE.

Space lines are inserted in units of 1 LINE.

1 Key in the number of lines you want to insert and press the [INSLIN] key. The inputted number of lines will be inserted. (If the number of lines to be inserted is not keyed in, but the [INSLIN] key is pressed, one line will be inserted.)



If the [INSLIN] key is pressed with the cursor in the above position, the state shown in the diagram on the right will occur.

(4) Inserting the 1 NET sequence program elements

Elements are inserted in 1 element units.

1 Key in the number of elements you want to insert and press the [INSELM] key. The inputted number of elements will be inserted. If a number of elements prefixed by the character "A" are keyed in and the [INSELM] key is pressed, the elements are inserted after the cursor.

(If the number of elements to be inserted is not keyed in, but the [INSELM] key is pressed, one element is inserted.)





If the [INSELM] key is pressed with the cursor in the position on the left, the state shown in the diagram on the right will occur.

If the character "A" is keyed in and the [INSELM] key is pressed with the cursor in the position on the left, the state shown in the diagram on the right will occur.

4.9.7 Deleting a Sequence Program	<ul> <li>(1) For deleting part of a program, use the following three kinds of soft keys and delete after setting the cursor to the unnecessary part.</li> <li>[]: Deletion of horizontal lines, relay contacts coils, etc.</li> <li>[▲]: Deletion of upper left vertical line to the cursor</li> <li>[▲]: Deletion of upper right vertical line to the cursor</li> </ul>
	<ul><li>(2) For the deletion of a program net (part corresponding to the section from RD instruction to WRT instruction), use the [DELNET] key.</li></ul>
	(3) Deleting multiple NETs in NET units

INSNET	DELNET	INSERT	ADRESS	SEARCH	COPY	MOVE		
	$\checkmark$							
EXEC	CANCEL	SEARCH	C-DOWN	C–UP				

1 Deletion

Move the cursor to the NET you want to delete and press the [DELETE] key. The net you want to delete will be displayed in red. (In the case of Mate, in reversal display.)

2 Deleting multiple nets

Move the cursor with the cursor DOWN key, [C–DOWN] key, or [SEARCH] key to display in red the NET you want to delete. (In the case of Mate, in reversal display.) Further, key in a numerical value and press the [C–DOWN] key to move the cursor the number of times specified by this value.

**3** Execution ..... Press the [EXEC] key

Cancellation .... Press the [CANCEL] key

4 If you already know the NET you want to delete, move the cursor to the first NET, key in the number of NETs, and press the [DELNET] key to omit steps 1 and 2.

# 4.9.8 Searching a Sequence (1 Program

Search a sequence program by using the following soft keys.

(1) Soft key [TOP]

When this key is pressed, the start of the sequence program is desplayed on the screen and the cursor also sifts to the program start position.

(2) Soft key [BOTTOM]

When this key is pressed, the last of the sequence program is displayed on the screen and the cursor also shifts to this program end position.

(3) Soft key [SRCH]

In this search, you specify an address you want to search and it searches the specified address from the program of the cursor part on this screen to the last part of the program and displays the address on the screen. There are two methods to specify the address you want to search.

(a) Method of specifying the address by the cursor

Set the cursor to the relay contact part of the address you want to search and press the soft key [SRCH].

The system searches the same address as the address specified by the cursor from the cursor part of the program currently displayed on the screen to the end of the program.

When the same address is found, the program part is displayed on the screen, and the cursor shifts to that address part. If the same address is not found as a result of this search, the cursor remains in the same position.

When finishing, press the soft key on the extreme left.

(b) Method of specifying the address by input

Input the address you want to searcch by using address and numeric keys, then press the soft key [SRCH].

The same address as specified is searched from the program of the cursor part currently displayed on the screen to the last part of the program.

When the same address is found, the program part is displayed on the screen, and the cursor shifts to that address part.

If the same address is not found as a result of this search, an error is displayed.

(4) Soft key [W–SRCH]

This key specifies an address of the relay coil to be searched, and then searches the relay coil of the specified address from the program at the cursor part to the end of the program on this screen. Then, it displays the relay coil on the screen.

Two methods are available to specify the address of the relay coil to be searched.

(a) Method of specifying the address by cursor

Set the cursor to the relay contact of the relay coil to be searched, and press the soft key [W–SRCH].

The corresponding relay coil is searched from the program of the cursor part to the end of the program.

When the relay coil is found, the program part is displayed on the screen, and the cursor shifts to the relay coil.



If no corresponding relay coil is found as a result of the search, an error occurs.

(b) Method of specifying the address by input

Input the address of the relay coil to be searched by both address and numeric keys, and then press the soft key [W–SRCH].

The specified address relay coil is searched from the program of the cursor part currently displayed on the screen to the end of the program.

When the specified address relay coil is found, the program part is displayed on the screen, and the cursor shifts to the relay coil.

If no relay coil is found as a result of the search, an error occurs.

(5) Soft key [N–SRCH]

This displays the ladder with the specified NET number from the top of the screen. If the number is not keyed in, but the [N–SRCH] key is pressed, the display is scrolled down by one NET.

(6) Soft key [S–SRCH]

Key in the functional instruction name or number and press the [S–SRCH] key to start searching the functional instruction. When the [S–SRCH] key is pressed during execution of a functional instruction, the functional instruction with the same number as this instruction is searched.

- (7) Searching with cursor keys ( $< \rightarrow >$  )
  - Key in the address or symbol and press the cursor to start searching the NET No.
  - Key in the NET NO. and press the cursor key to start searching the NET NO.

- Key in the functional instruction name or functional instruction number starting with "S" and press the cursor key to start searching the functional instruction. Key in "END1" or "S1" and press the cursor to search Example) functional instruction END1. 4.9.9 The sequence program with multiple NETs is copied in units of NETs. Specify the NET to be copied and specify the copy position with the **Copying a Sequence** cursor. When copying, the number of copies can also be specified. Program 1 Copying Move the cursor to the NET you want to copy and press the [COPY] key. The NET you want to copy will be displayed in yellow (in the case of Mate, in reversal display). Copying multiple NETs 2 Move the cursor with the cursor UP/DOWN key, [C-UP] key, [C-DOWN] key, or [SEARCH] key to display in yellow the NET to be copied. (In the case of Mate, in reversal display.) Further, if you in a numerical value and press the [C–UP] or [C–DOWN] key, you can scroll up or down the screen by the number of times specified by this value. Setting the NET to be copied 3 Press the [UNTIL] key. 4 Specifying the copying address Copying is performed by the [TO] key. At this time, the NET is copied in the direction above the cursor. If the number of copies is keyed in before the [TO] key is pressed, the NET is copied that
  - 5 Further, if the NET you want to copy is already known, if the cursor is moved to the first NET and the number of NETs is keyed in, then by pressing the [COPY] key, steps 1 to 3 can be omitted.

	INSNET	DELNET	INSERT	ADRESS	SEARCH	COPY	MOVE						
	UNTIL	CANCEL	SEARCH	C-DOWN	C–UP								
	$\checkmark$												
	то	CANCEL	SEARCH	C-DOWN	C–UP								

specified number of times.

#### CAUTION

An error NET cannot be copied.

4.9.10 Moving a Sequence Program	A s Spe cur	sequence program with multiple NETS is moved in units of NETs. ecify the NET to be moved and specify the move position with the rsor. When moving, the number of moves can also be specified.
riogram	1	Moving
		Move the cursor to the NET you want to move and press the [MOVE] key. The NET you want to move will be displayed in yellow. (In the case of Mate, in reversal display.)
	2	Moving multiple NETs
		Move the cursor with the cursor UP/DOWN key, [C–UP] key, [C–DOWN] key, or [SEARCH] key to display in yellow the NET to be moved. (In the case of Mate, in reversal display.) Further, if you key in a numerical value and press the [C–UP] or [C–DOWN] key, you can scroll up or down the screen by the number of times specified by this value.
	3	Setting the NET to be moved
		Press the [UNTIL] key.
	4	Specifying the moving address
		Moving is performed by the [TO] key. At this time, the NET is moved in the direction above the cursor.
	5	Further, if the NET you want to move is already known, if the cursor is moved to the first NET and the number of NETs is keyed in, then by pressing the [MOVE] key, steps $1-3$ can be omitted.

	INSNET	DELNET	INSERT	ADRESS	SEARCH	COPY	MOVE						
$\checkmark$													
	UNTIL	CANCEL	SEARCH	C-DOWN	C–UP								
	$\checkmark$												
	то	CANCEL	SEARCH	C-DOWN	C–UP								

## CAUTION

An error NET cannot be moved.

## 4.9.11 Symbol Data Display

(1) Symbol and comment data display

Symbol data and comment are displayed together with a ladder diagram on the screen as follows.

When symbol data and comment are defined in signal addresses in the program, the signal name and comment are displayed as shown in the above diagram.

When converting the symbol and address display, press the shift key [ADRESS or SYMBOL].

(2) Symbol input and search in the sequence program

When symbol data is defined in signal addresses in the sequence program, input and reference can be performed by the symbols. (Address and symbol are only different in operation.)

If neither symbol data nor comment is defined at an address, the address is displayed as it is.



## 4.9.12 Compressed Input by [COMAND] Key

The main function of each soft key can be directly selected from the [COMAND] key.

After keying in the characters shown below, press the [COMAND] key. [] shows parts that can be omitted. Further, the "n" appearing after the characters signifies that it is also posssible to input a numerical value. For example, after keying in "D2", pressing the [COMAND] key results in the same operation as keying in 2 and pressing the [DELNET] key.

- I [NSERT]
- D [ELNET] [n ]
- A [DRESS]
- SY [MBOL]
- S [EARCH]
- C [OPY] [n ]
- M [OVE] [n ] n :numerical value

The creation and search of programs is performed by pressing the software keys of the above menu.

#### NOTE

The software keys [ $\bigwedge$  or  $\bigwedge$ ] and [ $\bigwedge$  or  $\bigwedge$ ] are used to create or delete the upper left vertical line or the upper right vertical line on the ladder diagram. The solid line display of the vertical line indicates creation; the dotted line display of the vertical line indicates deletion. As to which menu will appear above the software keys, is decided by the ladder diagram form and the cursor position.

## 4.9.13 Ending Edit of a Sequence Program

In the program menu shown below, press the extreme left software key.

#### 

#### CAUTION

When an error NET exists, ERROR NET NO. is displayed and you cannot end the edit. End after correcting the erroneous NET.

4.10 INPUT/OUTPUT OF LADDER PROGRAM WITH P–G AND FLOPPY CASSETTE/FA CARD	
4.10.1 General	The ladder program can be stored in or fetched out of a floppy cassette/FA card by connecting P–G and floppy cassette adapter/FA card adapter by using this function enables reading the program stored in a floppy cassette/FA card by using PMC RAM into P–G or reading the program stored in a floppy cassette/FA card by using P–G into PMC RAM. The usable adapters are as follows:
	• FANUC cassette adapter 3 (A13B–0131–B001)/cassette F1 (A87L–0001–0038)
	• FANUC floppy cassette adapter (A13B–0150–B001)/floppy cassette
	(A8/L-0001-0039) • EA card adapter (A13B-0148-B001)/EA card (A87B-0001-0108)
	• 1A card adapter (A15D=0140=D001)/1A card (A07D=0001=0100)
4.10.2 Setting I/O Commands	When using the FANUC floppy cassette adapter/FA card adapter, change the settings of the input/output devices by the following 'IO commands'.
J	<b>1</b> Press the R3 key on the R key menu screen. 'REQUEST=' is displayed lower left on the screen, and keying in is permitted.
	2 Key in 'IO BCA, CN2, F13, F14 [NL]'. The floppy cassette adapter/FA card adapter is assigned to channel 2.
	<b>3</b> To return the assignment to channel 2 to PMC WRITER, key in 'IO AUX, CN2, F9 [NL]'.
4.10.3	1 Turn on F13 key.
Program Input	(Turn on F12 too, when C language program is included.)
	2 Key in the menu No. '3 [NL]'.
	<b>3</b> (For PMC–SC only) Enter the type of a ROM module to be used. (See the following note for selecting ROM module B or C.)
	SELECT THE TYPE OF ROM MODULE ACCORDING TO THE FOLLOWING NO. ROM MODULE 0:A, 1:B, 2:C No. =
	<ul> <li>The message is displayed lower left on the screen.</li> <li>SET BC &amp; KEY IN 'OK' OR 'NO'</li> <li>BC = OK <file next="" no.="" or=""></file></li> <li>BC =</li> </ul>
	5 Set the floppy cassette/FA card in the adapter, and enter the following data.

- 6 'OK 1 [NL]' (specify file No.) or 'OK NEXT [NL]' (read the next file).
- 7 The screen changes, and reading the program from the floppy cassette/FA card starts.
- 8 When the program reading ends normally, the screen will automatically return to the programmer menu. If any error is detected during the program reading, 'PART=' is displayed lower left on the screen. Check the error contents, and key in 'E [NL]'. The screen will return to the programmer menu.

#### CAUTION

When ROM module B or C is selected during use of SYSTEM P Mate, the program is overlaid. In this case, insert the work floppy disk for the external memory into drive 1.

4.10.4	
Program	Output

1 Turn on F14 key.

(Turn on F12 too, when C language program is included.)

- 2 Key in the menu No. '5 [NL]'.
- The message is displayed lower left on the screen.
   SET BC & KEY IN 'OK' OR 'NO'
   BC = OK <INT OR ADD OR FILE NO.>
   BC =
- 4 Set the floppy cassette/FA card in the adapter, and enter the following data.
- 5 'OK INT [NL]' (write at the floppy head),'OK ADD [NL]' (write in the next file) or 'OK1 [NL]' (specify file No.).

## CAUTION

When specifying file number, put the numbers in the ascending order. If the file No. located at the middle of a floppy disk is specified, the files after that will be deleted.

- 6 The screen changes, and writing the program into the floppy cassette/FA card starts.
- 7 When the program writing ends normally, the screen will automatically return to the programmer menu. If any error is detected during the program reading, 'PART=' is displayed lower left on the screen. Check the error contents, and key in 'E [NL]'. The screen will return to the programmer menu.

# 4.10.5 Program Collation

**1** Turn on F13 key.

(Turn on F12 too, when C language program is included.)

- 2 Key in the menu No. '7 [NL]'.
- **3** The following operations are the same as those after **3** in 'Program input'.

## CAUTION

For the program which is output from PMC–SA1/SA2/SB/SC RAM board to the floppy cassette/FA card by specifying LADDER of ALL, there is no problem in the input/collation. It is impossible to make input/collation for the program which is output by specifying PARAM.



# 5.1 GENERAL

This function edits floppy disk data in the unit of file. When key in only <NL> the menu screen of R key appears key in R3 key on the R key menu. 'REQUEST=' will be displayed on the left below part of the screen to show a key–in enable condition.

#### NOTE

The format for file designation is as follows: [drive No.] @ file name : file No.

The file attributes are as shown below.

- (1) File number
- (2) File name
- (3) File creation date
- (4) Identification of protection file (protect)
- (5) File size
- (6) Multi-volume number

These file attributes are attached when writing data into floppy disk.

When writing, the next floppy disk set request message is displayed, so specify date and protection file.

· · · · · · · · · · · · · · · · · · ·			
Contents of jobs	Name of command (Instruction)	Contents inputted from keyboard (NL key is inputted at the end of a command)	
Display of file name, or file size	FDLIST	FDLI { [D, [P,] [S,] [F,]	
Change of file name, date, etc.	RENAME	RENA file designation [, ${P \ NP}$ [,/date] [,@ new file name]	
Deletion of file	SCRATCH	SCRA file designation	
File area condensation	CONDENSE	COND [drive No.]	
Copy of file (This command is effective for SYSTEM P series with 2–floppy disk unit.)	REMOVE	$\begin{array}{l} REMO\left[\begin{array}{c} \left\{ \begin{matrix} M \\ A \end{matrix} \right] [file \ designation \right] \\ [, \left\{ \begin{matrix} INT \\ ADD \end{matrix} \right] [, \left\{ \begin{matrix} P \\ NP \end{matrix} \right] [,/date ] \\ [, @ \ new. \ file \ name ] \end{array} \right. \end{array}$	

#### File editing command table

Set FD, and	key in 'OK', KILL, or 'NO'.
FD=OK	<int add,="" or=""> <p np,="" or="">  <drive>&lt;@name&gt;</drive></p></int>
FD=_	

Set the floppy disk and key in as follows.



When reading, the following floppy set request message is displayed.



Set floppy disk and key in as follows.



In file editing function, the above floppy disk set request message key input parameter can also be used. Now, parameter used in common here here has the following meaning. Specify 'OK', 'NO', 'KILL' and instruct the answer to the set request.

- OK . . . . After instructing execution of read and write, specify parameter.
- NO . . . . Cancel read/write to floppy only.
- KILL . . Cancel the specified process.

Parameters instructing details of read and write is as follows.

Parameter	Function	Notes
INT	When writing, write from the head of the floppy	When omitted, it is regarded as ADD. If INT is specified to protection file on error generators
ADD	When writing, add after exising fie	protectionnie, an error generaters.
Р	Prepare as protection file	When omitted, it is regarded as NP. Ready files can be changed by
NP	Prepare as ordinary file	RENAME command.
Date	Specify file preparation date with 6 numbers	Blank when omitted.
Drive number	Specify drive number 0 or 1 set with read/write floppy disk. 0 ; Upper unit 1; Lower unit	When omitted, it is regarded as 0. See Note).
@ File name	Specify file name (Max. 17 characters). When reading, the first name correspond-ing to the specified names is vallid.	Always specify when writing. When reading, if omitted, the file is valid.
; File number	When reading, specify the necessary file number after the :.	With the FDLIST command, file number and file name list can be displayed.

## CAUTION

When specifying drive number and file name or file number, specify without separating, as follows. Example) 0 @ ABC or 1 : 5

When displaying set request message, drive number is decided by the system, and 'FD0=' or 'FD1=' is displayed, instead of the 'FD=' message, to check the drive (unit) to be used. If a drive number is specified then, it will be ignored. (FD0 shows drive 0, and FD1, drive 1).

# 5.2 CONFIGURATION OF COMMAND

#### (1) General form of command

Operation	Space	Operand
Command name or its abbreviated form (4 leading characters)	-	List of one or more parameters delimited by delimiter symbol ', ' (comma).

A command name consists of plural alphabetic characters, and it can be abbreviated by four leading characters.

An operand consists of parameters peculiar to commands and parameters specified in floppy disk mounting request message.

(2) Execution of operands and commands

If operands are fully designated, a command is executed without displaying any floppy disk mounting request message.

However, a certain command may require many parameters. If these parameters cannot be recalled, specify the command name only. Necessary parameters are indicated in the floppy disk mounting request message. Accordingly, parameters can be input from the keyboard according to this display. The message may be displayed twice separately according to commands. (Old and new names are requested separately in RENAME command, for example.)

## 5.3 FDLIST COMMAND — FILE ATTRIBUTE DISPLAY

This command displays the attributes of files in the floppy disk, such as file name, file size, etc.

a) Input format

FDLIST { [D,] [P,] [S,] [F,] [L,] [Drive No.] { @ file name : file No.

- b) Operand
  - D: Display of file creation date consisting of 6 characters
  - P: Identification display of protection files
  - S: Display of file size
  - F: Display of size of unoccupied area
  - L: Executes all display by parameters D, P, S, F.

@	file name	Displays a file having the designated file name or designated
:	file No.	file number only. If this designation

- is omitted, all files are treated as processing objects.
- c) This command displays the information (attributes) on the floppy disk files.

If no attribute to be displayed is designated, the file number, file name, and multi–volume number only are displayed. The following example shows the display of all information (L designation)

NO.	FILE NA	ME	V.	DATI	E SIZE	P.
001	DATA1			83092	28 72	Р
002	DATA2			83102	28 60	
003	DATA3			83102	28 8	Р
****	DELETE	D FILE	****	• 10		
005	DATA4			90102	22 10	Р
006	DATA5			90102	22 5	
	FILE	USED	ARE	A	= 155	
	DELET	fed fil	E A	REA	= 10	
	FREE	AREA			= 1019	

#### NOTE

The numeric characters shown in SIZE, FILE USED AREA, DELETED FILE AREA, and FREE AREA are displayed assuming that 256 characters are 1.

## 5.4 **RENAME COMMAND** — FILE ATTRIBUTE CHANGE

This command designates a change of the file name, file creation date, and the designation of protection file.

a) Input format



Designation of file to be changed

#### b) Function

File attributes are renewed when they are designated by operand parameters. Attributes which are not designated are stored as they are. Protection files can be cancelled, but neither dates nor file names are changeable. The designation of protection files must be cancelled once before changing their attributes.

If all operands are omitted, the system displays an input message to request the designation of a file to be changed. When the file to be changed is designated by keying operation, a message is displayed to input attributes of the file to be changed by keying operation. Designate new data.

If the file to be changed only is designated together with the command, the system asks the file attributes to be changed.

(RENAME: 5 <NL>, for example)

Old attributes (B) and new attributes (A) are displayed by executing this command as shown below, for example.

Example)

RENAME :3, @ NEWNAME <F11>NL> **RENA: 3,@NEWNAME** NO. FILE NAME V. DATE P. B: 003 DATA3 901020 A: 003 NEWNAME 901020

## 5.5 SCRATCH COMMAND - DELETION OF **FILES**

This command deletes files of floppy disk.

a) Input format

```
SCRATCH [drive No.] { @ file name 
: file No.
```

Function b)

> This command deletes the designated file. Even if the file name is displayed by FDLIST, the file name is not displayed any longer. The area occupied by the deleted file must be released by CONDENSE command before writing new data into the area.

5.6 CONDENSE COMMAND -**RELEASE OF DELETED AREA**  This command releases the deleted file area to be employable.

a) Input format

CONDENSE [drive No.]

b) Function

The area occupied by the file deleted by SCRATCH command cannot be employed for writing new data under that condition. By executing this command, all unemploy-able areas can be released. Since it takes time to execute this command, it is recommended to arrange these areas when there are many files to be deleted and the residual capacity of the floppy disk is small.



Output designation : File name is "NEW", drive No.1 floppy disk with protection

- ii) The system asks every file to check if the file is to be copied or not. (M designation)
  - (Ex. 2) REMOVE M,1 @ A <NL>

In this example, the system asks to copy or not every file with file name starting with "A" of drive No.1 floppy disk.

A change of attributes such as file name, date, and file protection can be designated.

iii) All designated files are copied. (A designation)

(Ex. 3) REMOVE A,: 3, INT,/830920 <NL>

In this example, files with file name starting with "A" and with file No.3 and later of drive No.0 floppy disk are copied to drive No.1 floppy disk from the head of it with designated date "830930". The copied file names cannot be changed in this method.

When this command is executed, input file attributed (I) and output file attributes (O) are displayed. The next display example shows the execution of 'REMOVE A, 1 @TO, INT, P <NL>'.

NO.	FILE NAM	МE		V.DATE P.
I:001	T01 ZX	1.	100/40	830920
O:001	T01 ZX	1.	100/40	830920 P
I:002	T02 ZX	1.	150/50	830920
O:002	T02 ZX	1.	150/50	830920 P
I:003	T04 ZX	1.	100/50	830920
O:003	T04 ZX	1.	100/50	830920 P
I:004	T05 ZX	1.	20/50	830920
O:004	T05 ZX	1.	20/50	830920 P

If REMOVE command is only designated, key in operands according to the request message. The following are general designation format for file copy.

i) Without M, A designation (One file is copied.)

$$\operatorname{REMOVE} \left[ \begin{array}{c} \left\{ \begin{matrix} 0 \\ -1 \end{matrix} \right] & \left\{ \begin{array}{c} @ \text{ file name} \\ & \\ & \\ \end{array} \right. \text{ file No. } & \left[ , \left\{ \begin{matrix} \mathrm{INT} \\ \\ \underline{\mathrm{ADD}} \end{matrix} \right] & \left[ , \left\{ \begin{matrix} \mathrm{P} \\ \\ \mathrm{NP} \end{matrix} \right] \end{array} \right] \end{array} \right]$$

[,/ date] [, @ new file name] <NL>

ii) With M designation (request message is displayed for each objected file.)

REMOVE 
$$M, \begin{bmatrix} 0 \\ -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
 (  $\oplus$  file name : file No

iii) With A designation (All subjected files are copied.)

# APPENDIX



# ERROR CODES LIST (FOR FAPT LADDER P-G)

Error codes	Details of errors
01	Sequence program area over
02	No. of divisions has exceeded 99.
03	High level program time over
05	An error block was detected.
07	No designated step number is found.
08	An undefined instruction was specified.
09	An undefined address was specified.
10	Parameter data error
11	An address was employed in OR.STK and AND.STK.
12	An unemployable subroutine number was specified.
13	An unemployable timer number was specified.
14	A comparison error occurred.
15	A jump instruction was specified, exceeding END1 and END2.
16	A common instruction was specified, exceeding END1 and END2.
17	An instruction format error
18	An attempt was made to delete a parameter.
19	An attempt was made to add a parameter.
20	An erroneous system parameter data
21	A parameter was specified in a mode other than subroutine mode.
24	END2 is not specified.
25	WRT instruction is not specified in WRT instruction subroutine.
27	END1 is not specified.
29	A data sent from PMC–SB/SC is in error.
30	R1 key is pressed during data transmission between SYSTEM P series and PMC–SB/SC.
31	Input/output unit error
32	Read error
33	Hardware error of floppy disk
34	No designated file name is found.
41	An error occurred when inputting ROM data from ROM writer.
43	An error occurred when writing ROM data into ROM writer.
44	An error is deleted during comparison between SYSTEM P series-memory data and floppy data.
45	An error occurred when comparing ROM data with ROM writer data.
46	Key input data over
47	No designated symbol name is found.
48	A numeric value was directly specified to address parameters.
49	Counter number error
50	Decode functional instruction error
51	Symbol name (max. 6 characters) over

Error codes	Details of errors
52	Input data error
53	Comment data are in error.
54	Symbol table over
55	Comment data area over
56	Designated symbol name is already employed.
57	Symbol table sequence is in error.
58	Designated symbol name is not found.
59	END1 was detected in COM mode.
60	END1 was detected in JMP mode.
61	END2 was detected in COM mode.
62	END2 was detected in JMP mode.
63	END 3 was detected in COM mode.
64	END 3 was detected in JMP mode.
65	END 3 is not specified.
66	COM functional instruction was specified in COM mode.
67	JMP functional instruction was specified in JMP mode.
68	Message address error
69	Message data area over
70	Message data error
71	No symbol table is prepared.
72	NC model error in title
73	Title number error
74	Title data error
75	I/O port address error
76	Group number error
77	Base number error
78	Slot number error
79	I/O module name error
80	I/O port data are not prepared yet.
81	I/O port data were doubly specified.
82	Specified symbol or address is missing.
83	An invalid unit is loaded in the ROM WRITER or the specification of ROM WRITER does not meet the unit.
84	ROM module type is different from the specified one.
87	Output (or input) module was specified as an input (or output) address.
88	The same slot number was specified in the same group and the same base number.
89	The model of PMC is different.
93	The number of coils is specified by the COM or JMP command. (This causes an error for PMC–SA1 and PMC–SA2.)
150	Parity error of transfer data (check the cable.)
151	Excessive or insufficient data to be transferred (Check the cable.)
152	An EPROM or ROM module is not inserted in the ROM writer, or specification of the ROM writer is invalid.
153	Blank check error (Ultraviolet ray is not sufficiently irradiated or the EPROM, ROM module is defective.)

B-61863E/14

Error oodoo	Dataile of arrang
Error coues	Details of errors
154	Write error (EPROM or ROM module is defective.)
155	Verifyerror (EPROM or ROM module is defective.)
156	Data output level error (EPROM or ROM module is defective.)
157	Timer test error is ROM writer (ROM writer is defective.)
158	I/O test error in ROM write (ROM writer is defective.)
159	A/D converter test error in ROM writer (ROM writer is defective.)
160	Power test error in ROM writer (ROM writer is defective.)
161	Power (VPP) is defective (EPROM, ROM module or ROM writer is defective.)
162	Power supply (VCC) is defective (EPROM, ROM module or ROM writer is defective.)
163	ROM test error in ROM writer (ROM writer is defective.)
164	RAM test error in ROM writer (ROM writer is defective.)
170	An initialization error in the external memory floppy disk.
171	The inputted ROM format data is greater than the specified cassette type. The PASCAL origin of the inputted PASCAL load module is unsuitable.
172	The specified ROM format data cannot be edited with the P–G Mate. Outputting data in the ROM format is possible, however.
B-61863E/14



# WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/ SA1/SA2/SA3/SA5/SB/SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.1 FUNCTION

This window function is a functional instruction by which the data on the CNC is read or is written.

APPENDIX

# B.2 LOW-SPEED RESPONSE AND HIGH-SPEED RESPONSE OF WINDOW FUNCTION

In the way to process, there are window function high speed and one processed at low speed.

In case of a low–speed response, The data is read or written by the control between CNC and PMC

Therefore, it is necessary to ACT=1 of the window instruction must be held until the transfer completion information (W1) becomes 1 (interlock).

In a high–speed response, it is not necessity for take the interlock because the data is directly read.

# CAUTION

The window instruction of a low-speed response is controlled exclusively with the other window instructions of low-speed response.

Therefore, when the data is read or written continuously, it is necessary to clear ACT of the functional instruction once when the completion information (W1) become 1.

It does not work about ACT=1 of the other window instructions of low-speed response such as W1=1 and ACT=1 of the window instruction of a low-speed response. The window instruction of a high-speed response is not exclusively controlled like a low-speed response. Therefore, when the data is read or written continuously, yow need not make ACT=0.

The scan number of times to complete the processing is summarized on the following table.

TYPE	SCAN TIMES UNTIL PROCESSING ENDS
LOW	TWO SCAN TIMES OR MORE(This depends on the state of CNC)
HIGH	1SCAN TIME

# CAUTION

Enter the desired function code (to which 1000 is added when data of the second tool post (HEAD2) is read or written in the TT series, or when data of the second path is read or written in two-path control of the Power Mate-D.

To perform path 3 read/write operation in 3–path control, enter a function code + 2000.

# B.3 LIST OF WINDOW FUNCTIONS

Number	Description	Function code	R/W
1	Read CNC system information	0	R
2	Read the tool offset	13	R
3	Write a tool offset         *Low-speed response	14	W
4	Read the work origin offset *PM	15	R
5	Write work origin offset         *PM         *Low-speed response	16	W
6	Read parameters         *SB56         *Low-speed response	17	R
7	Write parameters *Low-speed response	18	W
8	Read setting data         *SB56         *Low-speed response	19	R
9	Write setting data *Low-speed response	20	W
10	Read custom macro variables         *SB56         *Low-speed response	21	R
11	Write custom macro variables         *Low-speed response	22	W
12	Read the CNC alarm state	23, 186 (*8)	R
13	Read the current program number	24	R
14	Read the current sequence number	25	R
15	Read an actual velocity for controlled axes	26	R
16	Read an absolute position (absolute coordinate value) on controlled axes	27	R
17	Read a machine position (machine coordinate value) on controlled axes	28	R
18	Read a skip operation (G31) stop position (coordinate value) on controlled axes	29	R
19	Read a servo delay amount	30	R
20	Read acceleration/deceleration delay amount on controlled axes	31	R
21	Read modal data	32	R
22	Read diagnosis data         *SB56         *Low-speed response	33	R
23	Read a feed motor load current value (A/D conversion data)	34	R
24	Reading tool life management data (tool group No.) *PM *21T *TM	38	R
25	Reading tool life management data (number of tool group s) *PM *21T *TM	39	R
26	Reading tool life management data (number of tools) *PM *21T *TM	40	R
27	Reading tool life management data (usable life of tool) *PM *21T *TM	41	R
28	Reading tool life management data (tool usage counter) *PM *21T *TM	42	R
29	Reading tool life management data (tool length compensation No. (1): Tool No.) *PM *21T *TM	43	R
30	Reading tool life management data (tool length compensation No. (2): Tool order No.) *PM *21T *TM	44	R
31	Reading tool life management data (cutter compensation No. (1): Tool No.) *PM *21T *TM	45	R
32	Reading tool life management data (cutter compensation No. (2): Tool order No.) *PM *21T *TM	46	R
33	Reading tool life management data (tool information (1): Tool No.) *PM *21T *TM	47	R
34	Reading tool life management data (tool information (2): Tool order No.) *PM *21T *TM	48	R
35	Reading tool life management data (tool No.) *PM *21T *TM	49	R
36	Reading the actual spindle speed	50	R

APPENDIX

Number	Description	Function code	R/W
37	Entering data on the program check screen *PM *21T	150	W
38	Reading clock data (date and time)	151	R
39	Writing torque limit data for the digital servo motor $st$ low–speed response	152	W
40	Reading load information of the spindle motor (serial interface)	153	R
41	Reading a parameter *PM *21T	154	R
42	Reading setting data *PM *21T	155	R
43	Reading diagnosis data *PM *21T	156	R
44	Reading a character string of the CNC program being executed in the buffer *C	157	R
45	Reading the relative position of a controlled axis	74	R
46	Reading the remaining travel	75	R
47	Reading CNC status information	76	R
48	Reading value of the P– code macro variable *SB56 *low-speed response *TM	59	R
49	Writing value of the P- code macro variable *low-speed response *TM	60	W
50	Reading the Tool life management data (Tool life counter type)	160	R
51	Registering the Tool life management data (Tool group) *low-speed response *TM	163	W
52	Writing the Tool life management data (Tool life) *low-speed response *TM	164	W
53	Writing the Tool life management data (Tool life counter) *low-speed response *TM	165	W
54	Writing the Tool life management data (Tool life counter type) *low-speed response *TM	166	W
55	Writing the Tool life management data (Tool length offset number (1): Tool number) *low-speed response *TM	167	W
56	Writing the Tool life management data (Tool length offset num-ber (2): Tool operation sequence number)         *low-speed response *TM	168	W
57	Writing the Tool life management data (Cutter compensation number (1):         Tool number)       *low-speed response *TM	169	W
58	Writing the Tool life management data (Cutter compensation nu–mber (2):         Tool operation sequence number)       *low-speed response *TM	170	W
59	Writing the Tool life management data (Tool condition (1): Tool number) *low-speed response *TM	171	W
60	Writing the Tool management data (Tool condition (2):         Tooloperation sequence number)         *low-speed response *TM	172	W
61	Writing the Tool life management data (Tool number) $*low-speed$ response *TM	173	W
62	Reading the Estimate disturbance torque data	211	R
63	Reading the current program number (8–digit program numbers) *PM *21T	90	R
64	Writing (registering) tool life management data (tool group number) *PM *21T *TM	200	R
65	Reading tool life management data (tool length offset number 1) *PM *21T *TM	227	R
66	Reading tool life management data (tool diameter offset number 1) *PM *21T *TM	228	R
67	Reading tool life management data (tool information 1) *PM *21T *TM	201	R
68	Writing tool life management data (tool group number) *low-speed response *TM	202	R
69	Writing tool life management data (tool length offset number 1) *low-speed response *TM	229	W

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

B-61863E/14

Number	Description		Function code	R/W
70	Writing tool life management data (tool radius offset numb $st$ lo $st$	oer 1) ow–speed response *TM	230	W
71	Writing tool life management data (tool information 1) $*$ lo	ow-speed response *TM	231	W
72	Reading actual spindle speeds		138	R
73	Reading fine torque sensing data (statistical calculation re	esults)	226	R
74	Reading fine torque sensing data (store data)		232	R
75	Specification of the number of the program for I/O Link		194	W
76	Presetting the relative coordinate *lo	ow-speed response	249	W
77	Deleting the Tool life Management Data (Tool group) $+$ k	ow–speed response *TM	324	W
78	Deleting the Tool life Management Data (Tool data) $*$ lo	ow–speed response *TM	325	W
79	Clearing the Tool life Management Data (Tool life counter $\%$	and Tool condition) ow–speed response *TM	326	W
80	Writing the Tool life Management Data (Arbitrary group nu $st$ lo	ımber) ow–speed response *TM	327	W
81	Writing the Tool life Management Data (Remaining tool life $st$ local tool life	e) ow–speed response *TM	328	W
82	Reading the current screen number		120(*8)	R
83	Reading detailed alarm information *Io	ow-speed response	186(*8)	R

*1 Function codes that have R in the R/W column are window read functions specifiable with the WINDR function command. Function codes that have W in the R/W column are window write functions specifiable with the WINDW function command.

- *2 For window functions mark with "Low–speed response," reading and writing parameters, setting data, diagnostic data and so on starts after the PMC receives the response for request of reading and writing from the CNC. On the contrary, the other window functions can read or write data at once in response to the request from PMC.
- *3 Functions marked with *PM are not provided for the Power Mate–D or F.
- *4 Functions marked with *21T are not provided for the Series 21T.
- *5 Functions marked with *SB5/6 support high-speed window response for the SB5/SB6.
- *6 Functions marked with *C are not provided for the SB5/SB6/SB7.
- *7 Functions marked with *TM are restricted on CNC for the compound machining function.
  - 1. Function can not be used.
  - 2. Function can not be used in the T mode.

Refer to each window function for details.

*8 This function code is supported only by the Power Mate *i*–D/H.

# B.4 FORMATS AND DETAILS OF CONTROL DATA

- (1) In the explanation of the window functions, minuses (-) in the data structure fields indicate that input data need not be set in these fields or that output data in these fields is not significant.
- (2) All data is in binary unless otherwise specified.
- (3) All data block lengths and data lengths are indicated in bytes.
- (4) Output data is valid only when window processing terminates normally.
- (5) Output data always includes one of the following completion codes. Note, however, that all of the completion codes listed are not always provided for each function.

Completion code	Meaning
0	Normal termination
1	Error (invalid function code)
2	Error (invalid data block length)
3	Error (invalid data number)
4	Error (invalid data attribute)
5	Error (invalid data)
6	Error (necessary option missing)
7	Error (write-protected)

Input and output control data has the following structure.



# B.4.1 Reading CNC System Information

# [Description]

System information peculiar to the CNC can be read. Such system information includes the series name of the CNC (16 as series name, for example), the machine type applied to the CNC, such s a machining center (M) and a lathe (T), the series code and version of the ROM containing the CNC system software, and the number of controlled axes.



#### [Input data structure]

# [Completion codes]

0: CNC system information has been read normally.

#### B-61863E/14

	[		
top address + 0	(Function code) 0		
2	(Completion code) 0 (Always terminates normally.)		
4	(Data length) 14		
6	(Data number) —		
8	(Data attribute)		
		Value	
10	CNC series name (2 bytes)	ASCII characters (16)	
12	Machine type M/T/TT (2 bytes)	ASCII characters ( M, T. TT, )	
14	ROM series of CNC system software(4 bytes)	ASCII characters (B 0 0 0 1, )	
18	ROM version of CNC system software(4 bytes)	ASCII characters (0 0 0 1, 0 0 0 2, , )	
22	Number of controlled axes (2 bytes)	ASCII characters ( 2, 3, 4, )	

# [Output data structure]

## NOTE

- 1 Data is stored from the upper digit in each lower byte.
- 2 In the Power Mate–D and –F, the data corresponding to the CNC series name and machine type are left as spaces.
- 3 In two-path control of the Power Mate-D, the data for the first path is the same as that for the second path.

# B.4.2 Reading a Tool Offset

[Description]

A tool offset value recorded in the CNC can be read.

Wear offset data, geometry offset data, cutter compensation data, and tool length offset data can be read as a tool offset.



#### [Input data structure]

#### (a) Offset types (for machining centers, Power Mate–D, F)

	Cutter	Tool length
Wear	0	2
Figure	1	3

If the type of tool offset need not be specified, enter 0.

#### NOTE

In the Power Mate–D and –F, read tool offsets without specifying the classification (i.e. cutter compensation, tool length, tool wear, and tool geometry).

#### (b) Offset types (for lathes)

	X axis	Z axis	Tool tip R	Virtual tool tip	Y axis
Wear	0	2	4	6	8
Figure	1	3	5	7	9

- 0: The tool offset has been read normally.
- 3: The offset number specified for reading is invalid. (This completion code is returned when the specified offset number data is not from 1 to the maximum number of offsets.)
- 4 : There are mistakes in the data attribute that specifies the type of the offset to be read.
- 6: For the offset number specified for reading, an additional tool offset number option is required, but it is missing. The offset number is not available for Power Mate–D/F.

## [Output data structure]



## Output data unit

		Input system	Increment system IS–B	Increment system IS–C
Machining center system		mm, deg system	0.001	0.0001
Powe	er Mate–D, F	inch system	0.0001	0.00001
Lathe system	Radius specification	mm, deg	0.001	0.0001
	Diameter specification	system	0.002	0.0002
	Radius specification	inch system	0.0001	0.00001
	Diameter specification	inon system	0.0001	0.00001

# B.4.3 Writing a Tool Offset (*Low-speed Response)

[Description]

The tool offset value can be directly written into the CNC.

Wear offset data, geometry offset data, cutter compensation data, and tool length offset data can be written as a tool offset.





(a) Offset types (for machining centers, Power Mate–D, F)

	Cutter	Tool length
Wear	0	2
Figure	1	3

If the type of tool offset need not be specified, enter 0.

## NOTE

In the Power Mate–D and –F, write tool offsets without specifying the classification (i.e. cutter compensation, tool length, tool wear, and tool geometry).

(b) Offset types (for lathes)

	X axis	Z axis	Tool tip R	Virtual tool tip	Y axis
Wear	0	2	4	6	8
Figure	1	3	5	7	9

B-61863E/14

# APPENDIX

input dutu unit	Input	data	unit
-----------------	-------	------	------

		Input system	Increment system IS–B	Increment system IS–C	
Machining center system		mm, deg system	0.001	0.0001	
Power Mate–D, F		inch system	0.0001	0.00001	
	Radius specification mm, deg		0.001	0.0001	
Lathe	Diameter specification	system	0.001	0.0001	
system	Radius specification	inch system	0.0001	0.00001	
	Diameter specification	inen system	0.0001	0.00001	

[Completion codes]

- 0: The tool offset has been written normally.
- 2: The data byte length for the tool offset specified for writing is invalid. (It is not set to 4.)
- 3: The offset number specified for writing is invalid. (This completion code is returned when the specified offset number data is not from 1 to the maximum number of offsets.)
- 4 : There are mistakes in the data attribute that specifies the type of the offset to be written.
- 6: For the offset number specified for writing, the additional tool offset number option is required, but it is missing. The specified offset number is out of range. (Power Mate–D, F)



# B.4.4 Reading a Workpiece Origin Offset Value (not Supported by the Power Mate–D or –F)

# [Description]

The workpiece origin offset recorded in the CNC can be read.

A workpiece origin offset is provided for each controlled axis (the first axis to the eighth axis) in the CNC. Either the workpiece origin offset for a specific axis can be read, or the workpiece origin offsets for all axes can be read at one time. If the additional axis option is not provided, however, the workpiece origin offset for the additional axis cannot be read.



## [Input data structure]

- 0: The workpiece origin offset has been read normally.
- 3: The specified data number is invalid because the number is not from 0 to 6.
- 4 : The specified data attribute is invalid because the attribute data is neither -1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.
- 6: There is no workpiece coordinate shift option added.



	Ο	utpı	ut c	lata	un	it
--	---	------	------	------	----	----

		Input system	Increment system IS–B	Increment system IS–C
Machining center system Power Mate–D, F		mm, deg system	0.001	0.0001
		inch system	0.0001	0.00001
	Radius specification	mm, deg	0.001	0.0001
Lathe	Diameter specification	system	0.001	0.0001
system	Radius specification	inch system	0.0001	0.00001
	Diameter specification	inon system	0.0001	0.00001

# B.4.5 Writing a Workpiece Origin Offset Value (⊁Low–speed Response) (not Supported by the Power Mate–D or –F)

# [Description]

Data can be written directly as a workpiece origin offset value in the CNC.

A workpiece origin offset is provided for each controlled axis (the first axis to the eighth axis) in the CNC. Either the workpiece origin offset value for a specific axis can be written, or the workpiece origin offset values for all axes can be written at one time. If the additional axis option is not provided, however, the workpiece origin offset value for the additional axis cannot be written.





Input	data	unit
-------	------	------

		Input system	Increment system IS–B	Increment system IS–C
Machining center system Power Mate–D, F		mm, deg system	0.001	0.0001
		inch system	0.0001	0.00001
	Radius specification	mm, deg	0.001	0.0001
Lathe	Diameter specification	system	0.001	0.0001
system	Radius specification	inch system	0.0001	0.00001
	Diameter specification		0.0001	0.00001

- 0: The workpiece origin offset has been written normally.
- 2: The specified data length is invalid.
- 3 : The data number is invalid because the specified number is not from 0 to 6.
- 4 : The specified data attribute is invalid because the attribute data is neither -1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.
- 6: There is no workpiece coordinate shift option added.

(Function code) 16		
(Completion code)	_	
(See the explanation of the completion codes.)	_	
(Data length)		
(L: Input data)	_	
(Data number) N (N = Input data)		
(Data attribute) M	_	
(M = Input data)	Value	
Workpiece origin offset value	Signed binary number (A negative value is represented in 2's complement.)	
	(Function code) 16 (Completion code) ? (See the explanation of the completion codes.) (Data length) L (L: Input data) (Data number) N (N = Input data) (Data attribute) M (M = Input data) Workpiece origin offset value	(Function code)       (Completion code)         ?       (See the explanation of the completion codes.)         (Data length)       (Data length)         (L: Input data)       (Data number)         (N = Input data)       Value         (M = Input data)       Value         Workpiece origin offset value       Signed binary number (A negative value is represented in 2's complement.)

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

B.4.6	[Description]
Reading a Parameter	Parameter data in the CNC can be read.
(*Low–speed Response)	There are four types of parameters in the CNC: Bit parameters having a definite meaning for each bit, byte parameters holding 1–byte data, word parameters holding 2–byte data, and double word parameters holding 4–byte data. Therefore, the length of the read data varies according to the parameter number specified.
	Note that bit parameters cannot be read in bit units. The eight bits (one byte) for a parameter number must be read at a time.
	For axis parameters, data for a specific axis can be read, or data for all axes can be read at a time.
	Specify pitch error compensation data in data Nos. 10000 to 11023 (1024 points in total).
	For details of parameter data, refer to the Operator's manual of the CNC.
	[Input data structure]



- 0: Parameter data has been read normally.
- 3: The parameter number specified for reading is invalid.
- 4 : The specified data attribute is invalid because it is neither 0, -1, nor a value 1 to n (n is the number of axes).
- 6 : Although a certain option, such as the pitch error compensation option, is required for the data of the parameter number specified for reading, it is not provided.

#### [Output data structure]



## CAUTION

For the SB5/SB6/SB7, macro executor parameters 9000 to 9011 cannot be read.

B.4.7	[Description]
Writing a Parameter	Data can be written in a parameter in the CNC.
( <b>*Low–speed</b> Response)	There are four types of parameters in the CNC: Bit parameters having a definite meaning for each bit, byte parameters holding 1–byte data, word parameters holding 2–byte data, and double word parameters holding 4–byte data. Therefore, the length of the written data varies according to the parameter specified.
	Note that bit parameters cannot be written in bit units. The eight bits (one byte) for the parameter number must be written at a time. This means that when a bit needs to be written, the whole data for the corresponding parameter number shall be read first, modify the target bit in the read data, then the data shall be rewritten.
	For axis parameters, data for a specific axis can be read, or data for all axes can be read at a time.
	For details of parameter data, refer to the Operator's manual of the CNC.
	Some parameters cause a P/S alarm 000 when data is written. (The power must be turned off before continuing operation.)
	[Input data structure]
Top address + 0	



- 0: Parameter data has been written normally.
- 2: The data byte length of the parameter specified for writing is invalid.
- 3: The parameter number specified for writing is invalid.
- 4 : The specified data attribute is invalid because it is neither 0, -1, nor a value from 1 to n (n is the number of axes).
- 6 : Although a certain option, such as the pitch error compensation option, is required for the data of the parameter number specified for writing, it is not provided.

#### [Output data structure]



# CAUTION

Parameters may not become effective immediately depending on the parameter numbers.

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

# B.4.8 Reading Setting Data (*Low-speed Response)

## [Description]

The CNC setting data can be read.

There are four types of setting data in the CNC: Bit setting data having a definite meaning for each bit, byte setting data stored in bytes, word setting data stored in 2–byte units, and double–word setting data stored in 4–byte units. Therefore, the length of the read data varies according to the setting data specified.

Note that bit setting data cannot be read in bit units. The eight bits (one byte) for the setting data number must be read at a time.

For axis parameters, data for a specific axis can be read, or data for all axes can be read at a time.

For details of setting data, refer to the Operator's manual of the CNC.

[Input data structure]



- 0: Setting data has been read normally.
- 3: The setting number specified for reading is invalid.
- 4 : The specified data attribute is invalid because it is neither 0, -1, nor a value from 1 to n (n is the number of axes).



# B.4.9 Writing Setting Data (*Low-speed Response)

# [Description]

Data can be written as setting data in the CNC.

For details of setting data, refer to the Operator's manual of the CNC.

[Input data structure]



- 0: Setting data has been written normally.
- 2: The byte length of the setting data specified for writing is invalid.
- 3: The setting data number specified for writing is invalid.
- 4 : The specified data attribute is invalid because it is neither 0, -1, nor a value from 1 to n (n is the number of axes).
- 5 : Data exceeding the allowable range was specified as setting data to be written. For example, when data outside the range from 0 to 3 is specified as the setting data to be written for I/O data, this completion code is returned.

Top address + 0	(Function code) 20		
2	(Completion code) ? (See the explanation of		
4	(Data length) L (N = Input data)		
6	(Data number) N (N = Input data)		
8	(Data attribute) M (M = Input data)	Value	
10	Setting data: Input data	Setting data-dependent form	

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

B.4.10 Reading a Custom Macro Variable (*Low-speed Response)

#### [Description]

A custom macro variable in the CNC can be read.

Custom macro variables may or may not be read depending on the variable type.

(1) Local variables

Local variables (#1 to #33) cannot be read.

(2) Common variables

Common variables (#100 to #149 and #500 to #531) can be read in floating–point representation. When the option to add common variables is provided, however, common variables range from #100 to #199 and #500 to #999.

# NOTE

1 Power Mate–D (two–path control), Power Mate–F: #100 to 199, #500 to 699.

Memory module A of one-path control: #100 to #149, #500 to #531.

Memory module B/C of one-path control: #100 to #199, #500 to #699.

- 2 On the Power Mate *i*–MODEL D/H, common variables #100 to #199 and #500 to #699 can be read and written. Set the variable number within these ranges as the data number of the input data, and read and write custom macro variables.
- (3) System variables

System variables (#1000 and up) can be read in floating-point representation.

For details of the custom macro variables, refer to the Operator's Manual for the CNC.

# CAUTION

For the SB5/SB6/SB7, system variables cannot be read.



#### [Input data structure]

APPENDIX

#### [Completion codes]

- 0: The custom macro variable has been read normally.
- 3 : The number of a custom macro variable that cannot be read was specified as the data number. Only common variables can be read as custom macro variables by this library command.
- 5: The custom macro variable is not within the range from 0.0000001 to 99999999.
- 6 : The custom macro option is not provided. The specified variable number is out of range. ( Power Mate–D, F)

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

B-61863E/14



#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.11 [Description] Writing a Custom Data can be written in a custom macro variable in the CNC. Macro Variable For details of common variables, refer to the Operator's manual of the CNC. (*Low-speed Response) CNC.

# [Input data structure]



- 0: The custom macro variable has been written normally.
- 2: The specified data length is invalid because it is not 6.
- 3: A custom macro variable number that cannot be written as the data number was specified.
- 6: The custom macro option has not been provided.

The specified variable number is out of range. (Power Mate–D, F)



#### APPENDIX

# B.4.12 Reading the CNC Alarm Status

B.4.12.1 Except Power Mate–D/F/H

#### [Description]

When the CNC is in the alarm status, the alarm status data can be read. The following alarm status data can be read:

(1) First byte of alarm status data



- PS1 : P/S alarm 100 (PWE (parameter write enable) is set to 1.)
- PS2 : P/S alarm 000 (Turn off the power before continuing operation. Some parameters activate this alarm status when they are written.)
- PS3 : P/S alarm 101 (The part program recording area is disordered. This alarm is activated when the power to the CNC is turned off during part program editing or reading of a machining program. To release the alarm, then press the RESET key while holding down the PROG key.)
- PS : A P/S alarm other than the above alarm is generated
- OTS : Stroke limit alarm
- OH : Overheat alarm
- SV : Servo alarm
- MALM: Memory alarm

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

(2) Second byte of alarm status data





- SPA : Spindle alarm
- [Input data structure]



B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.12.2 For Power Mate–D/F/H (Low–speed type)

#### (1) Overview

PMC application programs can read CNC alarm information.

- (2) Alarm information
  - 1) Alarm status

APPENDIX

Information concerning the alarm type

2) Detailed alarm

Information concerning the alarm number and axis information

(3) Input data configuration

Top addres	s+0	+2	+4	+6	+8	+10
	Function code	Completion code	Data length	Data number	Data attribute	Data area

Function code	: 23 (fixed)	)			
Completion code	: No specif	ication required.			
Data length	: No specif	No specification required.			
Data number	: Number of If 31 or m be 30.	of alarms which can be stored. (Up to 30). Hore are specified, the value is assumed to			
Data area	: No specif	ication required.			
Data attribute	: 0:	Alarm status information			
	Other that	n 0 : Detailed alarm information, indicated in two-byte bit-type data described below (multiple bits can be specified.)			

15         14         13         12         11         10         9         8         7         6         5         4	3 2 1	0 bit

Bit 0 : P/S alarm 100 (PS1)

(PWE, parameter write enable, is set to 1.)

Bit 1 : P/S alarm 000 (PS2)

(Turn off the power. Writing data into certain parameters may cause this alarm.)

Bit 2 : P/S alarm 101 (PS3)

(Part program storage has been disrupted. This alarm is issued when the CNC is turned off during tape editing or machining program reading. To release this alarm, press the RESET key while holding down the PROG key.)

- Bit 3 : A P/S alarm (PS) other than those described above has been issued. (Up to 255)
- Bit 4 : Stroke limit alarm (OTS)
- Bit 5 : Overheat alarm (OH)

- Bit 6 : Servo alarm (SV)
- Bit 7 : Not used
- Bit 8 : APC alarm (APAL)
- Bit 9 : Spindle alarm (SPA)
- Bit 10 : P/S alarm 5000 or greater (PS_2)
- Bit 11 to Bit 15 : Not used
- (4) Output data configuration

Top address+0		+2	+4	+6	+8	+10	
	Function	Completion	Data	Data	Data	Data	
	code	code	length	number	attribute	area	
Function code		: 23 (fixed)					
Completion code		: Always 0.					
Data length		: 2 when the input data attribute is set to 0 and no alarm is issued.					
		2 + 4*n when the input data attribute is set to other than 0 (n stands for the number of alarms issued).					
Data number		Same as that for the input data.					
Data attribute		Same as that for the input data.					
Data area attribute informatio	on	: Two–by i	te bit–tyj s set to as that	pe data 0 (each l t for the ir	when the oit indica put data).	e input c tes the sa	lata ime
		(2 + 4*n)	)–bvte da	ta. descrit	bed below.	for all ala	arm

 $(2 + 4^*n)$ -byte data, described below, for all alarm states specified in the input data attribute when the input data attribute is other than 0.



- (5) Completion code
  - 0: CNC alarm status has been read normally.

0: This alarm status in the CNC has been read normally.

# Top address + 0 (Function code) 23 2 (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) 2 6 (Data number) 8 (Data attribute) Value 2 byte bit data. For the meanings of the bits, see [Description] in this section. 10 CNC alarm status data

# B.4.13 Reading the Current Program Number

[Description]

The program number of a machining program being executed or selected on the CNC can be read.

When a subprogram is executed on the CNC, the program number of the main program can also be read. Note that the program number that can be read is the first program number (first loop main program).

This function accepts only 4–digit program numbers. When the specification supports 8–digit program numbers, specify function code 90 to read 8–digit program numbers.



[Input data structure]

## [Completion codes]

- 0: The program number of the currently executing program was read successfully.
- 6: The program number is an 8–digit program number. (Use function code 90.)



# [Output data structure]

(a) Current program number (ON)

The program number of the program being executed is set.

(b) Program number of main program (OMN)

When the currently executing program is a subprogram, the program number of its main program (first loop main program) is set, When the currently executing program is not a subprogram, 0 is set.
### **B.4.14** Reading the Current Sequence Number

[Description]

The sequence number of a machining program being executed on the CNC can be read. If sequence numbers are not assigned to all blocks of the machining program, the sequence number of the most recently executed block is read.

[Input	data	structure]
--------	------	------------



0: The current sequence number has been read normally.

### [Output data structure]



### B.4.15 Reading the Actual Velocity of Controlled Axes

### [Description]

The actual velocity of a movement on CNC–controlled axes can be read. Note that the read speed is the composite velocity for the controlled axes. When movement involves only the basic three axes, the X, Y, and Z axes, the composite velocity equals the actual velocity. When movement, however, involves the fourth axis, such as a rotation axis or a parallel axis, as well as some of the basic three axes, the composite velocity for all the relevant axes does not equal the actual velocity.



### [Input data structure]

### APPENDIX

### [Completion codes]

0: The actual velocity for the controlled axes has been read normally.

### [Output data structure]



## B.4.16 [Description] Reading the Absolute The absolute coordinates of the CNC-controlled axes for movement can be read. The absolute coordinates indicate those after cutter compensation or tool length compensation. Coordinates) of Controlled Axes [Input data structure]



### [Completion codes]

- 0: The absolute coordinates of the controlled axes have been read normally.
- 4: Data specified as the data attribute is invalid because it is neither -1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.

### B-61863E/14

### [Output data structure]

	(Function code)		
	27		
2	(Completion code)		
	(See the explanation of the completion codes.)		
4	(Data length)		
	(L = 4*n, n is the number of axes specified.)		
6	(Data number) —		
8			
	(Data attribute) M		
	(M: Input data)	Value	
10 At ax	bsolute coordinate of the controlled xis specified (4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
When the	number of controlled axes is 4		
		Value	
10 At (4	bsolute coordinate of the first axis 4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
14 At (4	bsolute coordinate of the second axis 4 bytes)		
18 _{At}	bsolute coordinate of the third axis 4 bytes)		
22 At (4	bsolute coordinate of the fourth axis 4 bytes)		

### Output data unit

		Input system	Increment system IS–B	Increment system IS–C
Mach	nining center system	mm, deg system	0.001 0.0001	
Powe	r Mate–D, F	inch system	0.0001	0.00001
Lathe system	Radius specification	mm, deg	0.001	0.0001
	Diameter specification	system	0.001	0.0001
	Radius specification	inch system	0.0001	0.00001
	Diameter specification	inon system	0.0001	0.00001

# B.4.17 [Description] Reading the Machine The machine coordinates of CNC-controlled axes for movement can be read. The read value equals the machine coordinate indicated on the current position display screen displayed in the CNC. (This screen can be displayed by pressing the function button POS.) Top address + 0 Top address + 0



### CAUTION

When an inch machine is used in metric input, or when a millimeter machine is used in inch input, the machine position that is read with bit 0 of parameter No. 3104 set to 1 differs from the value indicated by the CNC. In this case, therefore, the value read through the ladder must be calculated (converted).

[Completion codes]

- 0: The machine coordinates of the controlled axes have been read normally.
- 4: Data specified as the data attribute is invalid because it is neither -1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of the controlled axes.

### [Output data structure]

Top address + 0			
	(Function code) 28		
2	(Completion code)		
	(See the explanation of the completion codes.)		
4	(Data length)		
	(L = 4*n, n is the number of axes specified.)		
6	(Dete surplus)		
	(Data number) —		
8	(Data attribute)		
	(M: Input data)	Value	l
10	Machine coordinate of the controlled axis specified (4 bytes)	Signed binary (A negative value is represented in 2's complement.)	1
When t	the number of controlled axes is 4		
		Value	l
10	Machine coordinate of the first axis (4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
14	Machine coordinate of the second axis (4 bytes)	<u>_</u> c copionon()	
18	Machine coordinate of the third axis (4 bytes)		
22	Machine coordinate of the fourth axis (4 bytes)		

### Output data unit

		Input system	Increment system IS–B	Increment system IS–C
Mach	ining center system	mm, deg system	0.001 0.0001	
Powe	er Mate–D, F	inch system	0.0001	0.00001
	Radius specification	mm, deg	0.001	0.0001
Lathe system	Diameter specification	system	0.001	0.0001
	Radius specification	inch system	0.0001	0.00001
	Diameter specification		0.0001	0.00001

### B.4.18 Reading a Skip Position (Stop Position of Skip Operation (G31)) of Controlled Axes

### [Description]

When a block of the skip operation (G31) is executed by the CNC and the skip signal goes on to stop the machine, the absolute coordinates of the stop position on the axes of movement can be read.

[Input data structure]



### [Completion codes]

- 0: The coordinates of the skip stop position for the controlled axes have been read normally.
- 4: Data specified for the data attribute is invalid because it is neither -1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.

### B-61863E/14

### [Output data structure]

Top address + 0		]	
	(Function code) 29		
2	(Completion code)	-	
	(See the explanation of the completion codes.)		
4	(Data length)		
	(L = 4*n, n is the number of axes specified.)		
6	(Data number) —		
8		-	
-	(Data attribute) M		
	(M: Input data)	Value	
10	Skip coordinate of the controlled axis specified(4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
When t	he number of controlled axes is 4		
		Value	
10	Skip coordinate of the second axis (4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
14	Skip coordinate of the third axis (4 bytes)		
18	Skip coordinate of the fourth axis (4 bytes)		
22	Skip coordinate of the first axis (4 bytes)		

### Output data unit

Inp		Input system	Increment system IS–B	Increment system IS–C
Mach	ining center system	mm, deg system	0.001	0.0001
Powe	er Mate–D, F	inch system	0.0001	0.00001
	Radius specification	mm, deg	0.001	0.0001
Lathe	Diameter specification	system	0.001	0.0001
system	Radius specification	inch system	0.0001	0.00001
	Diameter specification		0.0001	0.00001

### B.4.19 Reading the Servo Delay for Controlled Axes

### [Description]

The servo delay, which is the difference between the specified coordinates of CNC–controlled axes and the actual servo position, can be read.

[Input data structure]



APPENDIX

- 0: The servo delay for the controlled axes have been read normally.
- 4 : The data specified as the data attribute is invalid because it is neither -1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.

Top address + 0			
	(Function code) 30		
2	(Completion code)		
	(See the explanation of the completion codes.)		
4	(Data length)		
	L (L = 4*n, n is the number of axes specified.)		
6	(Data number) —		
8			
Ū	(Data attribute) M		
	(M: Input data)	Value	
10	Servo delay for the controlled axis spe- cified (4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
When t	he number of controlled axes is 4		
		Value	
10	Servo delay for the first axis (4 bytes)	Signed binary (A negative value is represented in 2's	
14	Servo delay for the second axis (4 bytes)	complement.)	
18	Servo delay for the third axis (4 bytes)		
22	Servo delay for the fourth axis (4 bytes)		

## B.4.20 [Description] Reading the The acceleration/deceleration delay, which is the difference between the coordinates of controlled axes programmed in the CNC and the position after acceleration/deceleration is performed, can be read. Deceleration Delay on Controlled Axes Image: Controlled Axes



### [Completion codes]

- 0: The acceleration/deceleration delay for the control axis has been read normally.
- 4 : The data specified as the data attribute is invalid because it is neither -1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.

### [Output data structure]

Top address + 0			
	(Function code) 31		
2	(Completion code)		
	(See the explanation of the completion codes.)		
4	(Data length)		
	L (L = 4*n, n is the number of axes specified.)		
6	(Data number) —		
8	(Data attribute) M		
	(M: Input data)	Value	
10	Acceleration/deceleration delay for the controlled axis specified (4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
When t	he number of controlled axes is 4		
10	Acceleration/deceleration delay for the first axis (4 bytes)	Signed binary (A negative value is represented in 2's complement )	
14	Acceleration/deceleration delay for the second axis (4 bytes)		
18	Acceleration/deceleration delay for the third axis (4 bytes)		
22	Acceleration/deceleration delay for the fourth axis (4 bytes)		

### Output data unit

Inp		Input system	Increment system IS–B	Increment system IS–C	
Mach	ining center system	mm, deg system	0.001	0.0001	
Powe	r Mate–D, F	inch system	0.0001	0.00001	
	Radius specification	mm, deg	0.001	0.0001	
Lathe	Diameter specification	system	0.001	0.0001	
system	Radius specification	inch system	0.0001	0.00001	
	Diameter specification		0.0001	0.00001	

### B.4.21 Reading Modal Data

### [Description]

Modal information can be read from the CNC.

(1) Format and types of modal data for the G function

Data corresponding to the specified identification code is read and stored in the data area. Whether the data is specified in the block specified in the attribute of the data is determined by the value at the most significant bit.



### NOTE

G codes for machining centers are also used for the Power Mate–D and –F except those marked with *. G codes marked with ** are not provided for the Power Mate–F.

						(1/2)
	Data type	Data		Data type		Data
Identificati on code	G code for machining center (M)	Code in a group	G A series	code for lathe (T, B series	G) C series	Code in a group
0	G00 G01 **G02 **G03 *G33	0 1 2 3 4	G00 G01 G02 G03 G32 G33 G34 G90 G92 G94 G71 G72 G73 G74 G1y G74	G00 G01 G02 G03 G33 G34 G77 G78 G79 G71 G72 G73 G73 G74 G series only	G00 G01 G02 G03 G33 G34 G20 G21 G24 G72 G73 G73 G74 G75 Only	0 1 2 3 4 8 9 5 6 7 10 11 12 13
1	G17 G18 G19	0 8 4	G96 G97	G96 G97	G96 G97	1 0
2	G90 G91	0 1		G90 G91	G90 G91	0 1
3			G68 G69	G68 G69	G68 G69	1 0
4	G94 G95	0 1	G98 G99	G94 G95	G94 G95	0 1
5	G20 G21	0 1	G20 G21	G20 G21	G70 G71	0 1

APPENDIX

						(2/2)
	Data type	Data	Data type			Data
Identificati	G code for machining	Code in a	G code for lathe (T, G)		Code in a	
on code	center (M)	group	A series	B series	C series	group
6	*G40 *G41 *G42	0 1 2	G40 G41 G42	G40 G41 G42	G40 G41 G42	0 1 2
7	G43 G44 G49	1 2 0	G25 G26	G25 G26	G25 G26	0 1
8	G73 G74 G76 G80 G81 G82 G83 G84 G85 G86 G87 G88 G89	10 11 12 0 1 2 3 4 5 6 7 8 9	G22 G23	G22 G23	G22 G23	1 0
9	<b>∦G98</b> ∦G99	0 1	G80 G83 G84 G85 G87 G88 G89	G80 G83 G84 G85 G87 G88 G89	G80 G83 G84 G85 G87 G88 G89	0 1 2 3 5 6 7
10	%G50 ∦G51	0 1		G98 G99	G98 G99	0 1
11	G66 G67	1 0	G66 G67	G66 G67	G66 G67	1 0
13	*G54 *G55 *G56 *G57 *G58 *G59	0 1 2 3 4 5	G54 G55 G56 G57 G58 G59	G54 G55 G56 G57 G58 G59	G54 G55 G56 G57 G58 G59	0 1 2 3 4 5
14	*G61 *G62 *G63 *G64	1 2 3 0				
15	*G68 *G69	1 0				
16	+G15 +∀G16	0 1				
17	G40.1 G41.1 G42.1	1 2 0				
18	G25 G26	0 1				
19			G50.2 G51.2	G50.2 G51.2	G50.2 G51.2	0 1
20	G13.1 G12.1	0 1	G13.1 G12.1	G13.1 G12.1	G13.1 G12.1	0 1



### (2) Format and types of modal data for other than the G function

The specification of whether a decimal point is specified or not, in FLAG1, and the specification of the number of decimal places, in FLAG2, are valid only for F code. Even if a decimal point is not specified, the number of deci-mal places may not be 0. Note) As the numbers of input digits, M, S, T, and B, in a command address, the allow-able numbers of digits that are specified for the appropriate parameters are returned.

- M: Allowable number of digits of M code No. 3030
- S: Allowable number of digits of S code No. 3031
- T: Allowable number of digits of D code No. 3032 B: Allowable number of digits of B code No. 3033

Da		
Identification code	Specified address	
-2	Enter identification codes 100 to 126 at one time.	
100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125	100 to 126 at one time. B D E F H L M S T R P Q A C I J K N O U V W X Y Z	(second auxiliary function) (reserved)

### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

### CAUTION

The Power Mate–D/F is not provided with the second auxiliary function.

[Input data structure]



When all data items are specified to be read, the data items are all output simultaneously in the order specified in the above data table.

- 0: Modal information has been read normally.
- 3: Invalid data is specified as the data number.
- 4: Invalid data is specified as the data attribute.

### [Output data structure]

Top address + 0	(Function code) 32 (See the explanation above		
2	(Completion code)	-	
	(See the explanation of the completion codes.)	- L - 2 · Grunction	
4	(Data length) L (L = 2, 6, 2*n, 6*m)	$L = 2^{*}n : \qquad \text{All data for G function}$	ion
6	(Data number) N (N: Input data)	L = 6*m : All data for other th (n: Number of groups for the G (m: Number of types other than	an G function function) i for the G function)
8	(Data attribute) M	- 	
	(M: Input data)	Value	
10	Modal data for G function (2 bytes)	See the data format for the G function. The upper byte must always be set to 0.	
Or	1		
8	(Data attribute)		
	(M: Input data)	Value	
10	Data part of modal data for other than G function(4 bytes)	See the data format for other than the G function.	
14	Flag part of modal data for other than G function(2 bytes)	See the flag format of the data for other than the G function. The upper byte must always be set to 0.	

When all data items are specified to be read, the data items are all output simultaneously in the order specified in the above data table.

B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

### B.4.22 Reading Diagnosis Data (*Low-speed Response)

### [Description]

The information displayed on the diagnosis data screen in the CNC can be read.

[Input data structure]

APPENDIX



- 0: Diagnosis data has been read from the CNC normally.
- 3: The specified diagnosis data number is invalid.
- 4: The data specified as the data attribute is invalid because it is neither 0, -1, nor a value from 1 to n (n is the number of axes).
- 6: An option required for reading the specified diagnosis data, such as the remote buffer option, is not provided.

[Output data structure]



B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

### B.4.23[Description]Reading A/DThe digital value converted from the load current of the CNC-controlled<br/>axis can be read. The input data from the general-purpose analog input<br/>can also be read.

APPENDIX

[Input data structure]



(a) Type of analog voltage (data number)

N	Type of analog voltage		
0	General–purpose analog voltage information (for four channels)		
2	Load information for the CNC–controlled axes		

### CAUTION

Only one-path control of the Power Mate MODEL-D is provided with one channel of general-purpose analog voltage information. (b) Specifying a CNC-controlled axis (data attribute)

Specify a CNC-controlled axis for which the voltage conversion data for the load current is to be read. Data must be specified according to the following table (Valid with the FS16A/18A and FS16B (interface type A)):

Specification of CNC controlled-axis			
Specified data Connector in the CNC			
1	JV1 (MAIN BOARD)		
2	JV2 (MAIN BOARD)		
3	JV3 (MAIN BOARD)		
4	JV4 (MAIN BOARD)		
5	JV5 (OPTION BOARD)		
6	JV6 (OPTION BOARD)		
7	JV7 (OPTION BOARD)		
8	JV8 (OPTION BOARD)		

### CAUTION

For general-purpose analog input Other than the *i* series

The OPTION2 board is required. Analog input is performed through the analog signal interface. For details, refer to the description of the analog signal interface in the connection manual (hardware).

### i series

General-purpose analog input is not supported. Use the I/O Link analog input module.

### [Completion codes]

- 0: A/D conversion data has been read normally.
- 3: The data specified for the data number is invalid.
- 4: The data specified for the data attribute is invalid, or the specified axis number is greater than the number of controlled axes.
- 6: No analog input module is connected.



### [Output data structure]

(a) A/D conversion data (AD) of CNC controlled axis load information

The load current for the specified CNC controlled axis is converted into analog voltage, the input to the A/D converter to output a digital data.

The value actually set in the AD field is obtained from the following formula:

$$\begin{array}{l} (\text{AD})\times \frac{\text{N}}{6554} = \text{Load current } [\text{A}_{\text{peak}}] \\ \text{AD} = \text{A/D conversion data } [\text{Value read by the window function } (\pm)] \\ \text{N} = \text{Nominal current limit for the amplifier corresponding to the motor} \\ \text{For the nominal current limits, see the table below or the} \\ \text{descriptions of the control motor.} \end{array}$$

(b) A/D conversion data (A/D) for general-purpose analog voltage information

In A/D conversion data (A/D), 0 corresponds to -10V, 128 corresponds to 0V, 255 corresponds to +10V, and other values correspond in a direct proportion to these values.

 $\begin{array}{l} (\text{AD}-128)\times\frac{N}{128}=\text{Load current }[\text{A}_{\text{peak}}]\\ \text{AD}=\text{A/D conversion data }[\text{Value read by the window function }(\pm)]\\ \text{N}=\text{Nominal current limit for the amplifier corresponding to the motor}\\ \text{For the nominal current limits, see the table below or the descriptions of the control motor.} \end{array}$ 

### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

Servo amplifier module		Applicable motor model	Nominal current limit
Model	Connected axis		
SVM1-12 SVM2-12/12 SVM2-12/20 SVM2-12/40 SVM3-12/12/12 SVM3-12/12/20 SVM3-12/20/20 SVM3-12/12/40 SVM3-12/20/40	L and M axes L axis L axis L, M, and N axes L and M axes L axis L and M axes L axis L axis	α 0.5/3000 α 1/3000 α 2/2000 α 2/3000	12Ар
SVM1-20 SVM2-12/20 SVM2-20/20 SVM2-20/40 SVM3-12/12/20 SVM3-12/20/20 SVM3-20/20/20 SVM3-12/20/40 SVM3-20/20/40	M axis L and M axes L axis N axis M and N axes L, M, and N axes M axis L and M axes	α C3/2000 α C6/2000 α C12/2000	20Ар
SVM1-40S		α 3/3000 α 6/2000 α M3/3000 α L3/3000	40Ap
SVM2-12/40 SVM2-20/40 SVM2-40/40 SVM3-12/12/40 SVM3-12/20/40 SVM3-20/20/40	M axis M axis M axis N axis N axis N axis N axis	α 3/3000 α 6/2000 α 12/2000 α M3/3000 α L3/3000 α C22/1500	40Ap
SVM1-40L SVM2-40/80	L axis	α 3/3000 α 6/2000 α 12/2000 α 22/1500 α M3/3000 α L3/3000 α C22/1500	40Ap
SVM1-80 SVM2-40/80 SVM2-80/80	L axis L and M axes	α 6/3000 α 12/3000 α 22/2000 α 30/1200 α M6/3000 α M9/3000 α L6/3000 α L9/3000	80Ap
SVM1-130		$\begin{array}{c} \alpha \ 30/2000 \\ \alpha \ 40/2000 \\ \alpha \ 22/3000 \\ \alpha \ 30/3000 \\ \alpha \ 40/2000 \ (with a fan) \\ \alpha \ L25/3000 \\ \alpha \ L50/2000 \end{array}$	130Ap

### NOTE

- 1 The current limits (peak values) are standard values. The operation value variation due to a circuit constant is about +10%.
- 2 SVM1–130 requires forced air cooling when the  $\alpha$  22/3000,  $\alpha$  30/3000,  $\alpha$  40/2000 (with a fan),  $\alpha$  L25/3000, or  $\alpha$  L50/2000 is driven. At this time, the rated output current is 51.0 Arms.

B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

## B.4.24[Description]Reading Tool Life<br/>Management Data<br/>(Tool Group No.)By specifying a tool No., the No. of the tool group to which the specified<br/>tool belongs can be read from tool life management data.

APPENDIX

### [Input data structure]



### CAUTION

If 0 is specified for the tool No., the No. of the tool group currently used is read. In this case, if a tool group No. has not been specified since the power to the CNC was turned on, 0 is output.

If the same tool belongs to two or more tool groups, the Nos. of all tool groups to which the tool belongs are displayed.

- 0: The tool group No. has been read normally.
- 4: The value specified for the data attribute is invalid.
- 5: The specified tool No. was not found.
- 6: The tool life management option has not been added.

### [Output data structure]



### NOTE

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

### B-61863E/14

### B.4.25 Reading Tool Life Management Data (Number of Tool Groups)

### [Description]

The number of tool groups in tool life management data can be read.

The number of tool groups that can be registered varies depending on the setting of parameter 6800 of the CNC, as indicated in the following table.

### Parameter 6800

GS2	GS1	Number of tools The numbers in parentheses apply when the additional option is used		
		M series	T series	
0	0	1 to 16 (1 to 64)	1 to 16 (1 to 16)	
0	1	1 to 32 (1 to 128)	1 to 32 (1 to 32)	
1	0	1 to 64 (1 to 256)	1 to 64 (1 to 64)	
1	1	1 to 128 (1 to 512)	1 to 16 (1 to 128)	

M series: For Machining Centers T series: For Lathes





- 0: The number of tool group Nos. has been read normally.
- 6: The tool life management option has not been added.





### NOTE

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

### B.4.26 Reading Tool Life Management Data (Number of Tools)

### [Description]

By specifying a tool group No., the number of tools that belong to the tool group can be read from tool life management data.

The number of tools that can be registered varies depending on the setting of parameter 6800 of the CNC, as indicated in the following table.

### Parameter 6800

Second stateNumber of toolsGS2GS1The numbers in parentheses apply when the option is used			s apply when the additional
		M series	T series
0	0	1 to 16 (1 to 64)	1 to 16 (1 to 16)
0	1	1 to 32 (1 to 128)	1 to 32 (1 to 32)
1	0	1 to 64 (1 to 256)	1 to 64 (1 to 64)
1	1	1 to 128 (1 to 512)	1 to 16 (1 to 128)

M series: For Machining Centers T series: For Lathes

### [Input data structure]



### CAUTION

If 0 is specified for the tool group No., the number of tools that belong to the tool group currently used is read. In this case, if a tool group No. has not been specified since the power to the CNC was turned on, 0 is output.

[Completion codes]

- 0: The number of tools has been read normally.
- 3: The specified tool group No. is invalid.
- 6: The tool life management option has not been added.

### [Output data structure]



### NOTE

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

### B-61863E/14

B.4.27 Reading Tool Life Management Data (Tool Life) (not Supported by the Power Mate D, –F, or Series 21–TA)

### [Description]

By specifying a tool group No., the life of tools belonging to the tool group can be read from tool life management data.

Whether to display the tool life in minutes or the number of cycles is selected by bit 2 of parameter 6800 (LTM) for the CNC.

[Input data structure]



### CAUTION

If 0 is specified for the tool group No., the tool life of the tool group currently used is read. In this case, if a tool group No. has not been specified since the power to the CNC was turned on, 0 is output.

- 0: The tool life has been read normally.
- 3: The specified tool group No. is invalid.
- 6: The tool life management option has not been added.

### [Output data structure]



### NOTE

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

## B.4.28[Description]Reading Tool LifeBy specifying a tool group No., the tool life counter for the specified tool<br/>group can be read from tool life management data.Management Data<br/>(Tool Life Counter)Image: Counter for the specified tool<br/>group can be read from tool life management data.

APPENDIX

### [Input data structure]



### CAUTION

If 0 is specified for the tool group No., the tool life counter for the tool group currently used is read. In this case, if a tool group No. has not been specified since the power to the CNC was turned on, 0 is output.

— 1123 —

- 0: The tool life has been read normally.
- 3: The specified tool group No. is invalid.
- 6: The tool life management option has not been added.

### [Output data structure]



### NOTE

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

### B-61863E/14

B.4.29 Reading Tool Life Management Data (Tool Length Compensation No. (1): Tool No.)

### [Description]

By specifying a tool group No. and a tool No., the tool length compensation No. for the specified tool can be read from tool life management data. This function is available only with the M series CNCs.





### CAUTION

If 0 is specified for both the tool group No. and tool No., the Nos. of the tool group and tool currently used are read. In this case, if a tool group No. has not been specified since the power to the CNC was turned on, 0 is output. For the T series CNCs, 0 is always output.
- 0: The tool length compensation No. has been read normally.
- 3: The specified tool group No. is invalid.
- 4 : The specified tool No. is invalid.
- 5: The specified tool No. was not found in the specified tool group.
- 6: The tool life management option has not been added.

#### Top address + 0 (Function code) 43 2 (Completion code) (See the explanation of the completion codes.) 4 (Data length) 4 6 (Data number) Ν (N: Input data) 8 (Data attribute) Μ (M: Input data) Value Tool length compensation No. (4 bytes) Unsigned binary 10

#### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

# B.4.30 Reading Tool Life Management Data (Tool Length Compensation No. (2): Tool Order No.)

# [Description]

APPENDIX

By specifying a tool group No. and tool order No., the tool length compensation No. for the specified tool can be read from tool life management data. This function is available only with the M series CNCs.

[Input data structure]



# CAUTION

If 0 is specified for the tool group No., the No. of the tool group currently used is read. In this case, if a tool group No. has not been specified since the power to the CNC was turned on, 0 is output.

When 0 is specified for the tool order No., if the specified tool group has been used, the tool currently used is read. In this case, if the specified tool group has not been used, the first tool in the group is read.

For the T series CNCs, 0 is always output.

- 0: The tool length compensation No. has been read normally.
- 3: The specified tool group No. is invalid.
- 4 : The specified tool order is invalid.
- 5: The tool having the specified tool order is not registered in the specified tool group.
- 6: The tool life management option has not been added.



#### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

B–61863E/14	APPENDIX	(PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

**B. WINDOW FUNCTION DESCRIPTION** 

# B.4.31[Description]Reading Tool LifeBy specifying a tool group No. and a tool No., the cutter compensation<br/>No. for the specified tool can be read from tool life management data.<br/>This function is available only with the M series CNCs.No. (1): Tool No.)[Description]



#### [Input data structure]

### CAUTION

If 0 is specified for both tool group No. and tool No., the Nos. of the tool group and tool currently used are read. If a tool group No. has not been specified since the power to the CNC was turned on, 0 is output. For the T series CNCs, 0 is always read.

- 0: The cutter compensation No. has been read normally.
- 3: The specified tool group No. is invalid.
- 4 : The specified tool No. is invalid.
- 5: The specified tool No. was not found in the specified tool group.
- 6: The tool life management option has not been added.



#### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

B–61863E/14	APPENDIX	(PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.32 Reading Tool Life Management Data (Cutter Compensation No. (2): Tool Order No.)

### [Description]

By specifying a tool group No. and a tool order No., the cutter compensation No. for the specified tool can be read from tool life management data. This function is available only with the M series CNCs.

**B. WINDOW FUNCTION DESCRIPTION** 

#### [Input data structure]



### CAUTION

If 0 is specified for the tool group No., the No. of the tool group currently used is referenced. In this case, if a tool group No. has not been specified since the power to the CNC was turned on, 0 is output.

When 0 is specified for the tool order No., if the specified tool group has been used, the tool currently used is read. In this case, if the specified tool group has not been used, the first tool in the group is referred to.

For the T series CNCs, 0 is always output.

- 0: The cutter compensation No. has been read normally.
- 3: The specified tool group No. is invalid.
- 4: The specified tool order No. is invalid.
- 5: The tool having the specified tool order is not registered in the specified tool group.
- 6: The tool life management option has not been added.



#### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

	B. WINDOW FUNCTION DESCRIPTION
	(PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/
APPENDIX	SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

#### B-61863E/14

# B.4.33 Reading Tool Life Management Data (Tool Information (1) : Tool No.)

# [Description]

By specifying a tool group No. and a tool No., the information for the specified tool can be read from tool life management data.





# CAUTION

If 0 is specified for both tool group No. and tool No., the Nos. of the tool group and tool currently used are referenced. If neither a tool group No. nor a tool No. has been specified since the power to the CNC was turned on, 0 is output.

- 0: The tool group No. has been read normally.
- 3: The specified tool group No. is invalid.
- 4: The specified tool No. is invalid.
- 5: The specified tool No. was not found in the specified tool group.
- 6: The tool life management option has not been added.

#### Top address + 0 (Function code) 47 2 (Completion code) (See the explanation of the completion codes.) 4 (Data length) 4 6 (Data number) N (N: Input data) 8 (Data attribute) Μ (M: Input data) Value 10 Number of tools (4 bytes) 0: See Note) on the previous page. 1: The tool is registered. 2: The tool has reached the end of its life. 3: The tool was skipped. The three high-order bytes are fixed to 0.

#### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

APPENDIX

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.34[Description]Reading Tool Life<br/>Management Data<br/>(Tool Information (2):<br/>Tool Order No.)By specifying a tool group No. and a tool order No., the information for<br/>the specified tool can be read from tool life management data.

[Input data structure]



#### CAUTION

If 0 is specified for the tool group No., the No. of the tool group currently used is read. If a tool group No. has not been specified since the power to the CNC was turned on, 0 is output.

When 0 is specified for the tool order No., if the specified tool group has ever been used, the tool currently used is read. In this case, if the specified tool group has not been used, the first tool in the group is referred to.

- 0: The tool group No. has been read normally.
- 3: The specified tool group No. is invalid.
- 4: The specified tool order No. is invalid.
- 5: The tool having the specified tool order is not registered in the specified tool group.
- 6: The tool life management option has not been added.



#### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.35[Description]Reading Tool Life<br/>Management Data<br/>(Tool No.)By specifying a tool group No. and a tool order No., the No. of the<br/>corresponding tool can be read from tool life management data.

APPENDIX



#### [Input data structure]

#### CAUTION

When 0 is specified for the tool group No., the tool group currently used is referenced. If neither a tool group No. nor a tool No. has been specified since the power to the CNC was turned on, however, 0 is output for the tool group No. When 0 is specified for the tool order No., if the specified tool group has been used, the tool currently used is referred to. If the specified tool group has not been used, the first tool in the group is referenced.

- 0: The tool No. has been read normally.
- 3: The specified tool group No. is invalid.
- 4: The specified tool order No. is invalid.
- 6: The tool life management option has not been added.



#### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

# B.4.36 Reading the Actual Spindle Speed

#### [Description]

The actual speed of the spindle can be read from the CNC.

#### [Input data structure]

APPENDIX



0: The actual speed of the spindle has been read normally.

[Output data structure]



APPENDIX

B.4.37	[Description]
Entering Data on the	On the program check screen of the CNC, data can be entered for the
Program Check Screen	spindle tool No. (HD.T) and the next tool No. (NX.T). This function is
(*Low-speed	available only with the M series CNCs.
Response)	This function is effective only when bit 2 of parameter 3108 is 1.
(not available for	
Power Mate–D/F,	
Sorios 21_TA)	[Input data structure]



- 0: Data has been entered on the program check screen normally.
- 2: The data length in bytes is invalid.
- 3: The data No. is invalid.

#### [Output data structure]



4

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.38 [Description] Reading Clock Data (Date and Time) (not available for Power Mate--F) The current date (year, month, day) and time (hours, minutes, seconds) can be read from the clock built into the CNC. Image: Top address + 0 [Input data structure] 2 (Function code) 151 2 (Completion code) (Need not be set)

APPENDIX

#### (Need not be set) N = -1: Reads current date and time. 6 (Data number) Reads current date. Ν N = 0: (N = 0, 1)N = 1: Reads current time. 8 (Data attribute) 0 10 (Data area) (Need not be set)

(Data length)

- 0: Data of the clock built into the CNC has been read normally.
- 3 : A value other than 0, 1, and -1 was specified for the data No.

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

#### B-61863E/14



#### [Output data structure]

When both the current date and current time are specified to be read by entering [-1] for the data No.

	(Input data)	Value
10	Current date (year)	Unsigned binary
12	Current date (month)	
14	Current date (day)	
16	Current time (hours)	
18	Current time (minutes)	
20	Current time (seconds)	

#### [Example] September 10th, 1990

#### [Example] 23:59:59 (hours:minutes:seconds)

Data area	1990	Data area	23
+2	9	+2	59
+4	10	+4	59

# B.4.39 [Description] Entering Torque Limit Torque limit values for the digital servo motor can be entered. Data for the Digital Servo Motor Servo Motor [kLow-speed Response) [Input data structure]

APPENDIX



[Example] To specify a torque limit of 50%, enter 128.

- 0: Torque limit data has been entered normally.
- 4: The specified data attribute is invalid. That is, a value other than 1 to n (number of axes) was specified, or the specified axis No. was greater than the number of controlled axes.

#### Top address + 0 (Function code) 152 2 (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) 2 (Input data) 6 (Data number) (Input data) 8 (Data attribute) Μ (M: Input data) Value 10 Torque limit data (1 byte): Input data The high–order byte is always set to 0. Unsigned binary <Unit: %> Values from 0 to 255 correspond to 0% to 100%.

#### [Output data structure]

# B.4.40

Reading Load Information of the Spindle Motor (Serial Interface)

#### [Description]

Load information of the serial spindle can be read.

The equation to normalize the load information is shown below

Load (%) = 
$$\frac{L}{32767} \times \lambda$$

- L: Data read from the window
- $\lambda$ : The percentage of the maximum output of the motor to the continuous rated output of the motor (When the maximum output is 180% and the continuous rated output is 100%, the percentage is 180.)

#### CAUTION

 $\lambda$  is equal to the value of parameter No. 4127.

[Input data structure]



0: Load information of the serial spindle has been read normally.



#### [Output data structure]

#### B-61863E/14

8		
	(Data attribute)	
	_	Value
10	First axis in the load information of the serial spindle	Unsigned binary, 2 bytes long
12	Second axis in the load information of the serial spindle	
14	Third axis in the load information of the serial spindle	
/hen four axes a	re specified	
Vhen four axes a 8	re specified (Data attribute) —	Value
Vhen four axes a 8 10	re specified (Data attribute) — First axis in the load information of the serial spindle	Value Unsigned binary, 2 bytes long
Vhen four axes a 8 10 12	re specified (Data attribute) — First axis in the load information of the serial spindle Second axis in the load information of the serial spindle	Value Unsigned binary, 2 bytes long
/hen four axes a 8 10 12 14	(Data attribute) (Data attribute) First axis in the load information of the serial spindle Second axis in the load information of the serial spindle Third axis in the load information of the serial spindle	Value Unsigned binary, 2 bytes long

APPENDIX

# B.4.41[Description]Reading a Parameter<br/>(not available for<br/>Power Mate-D/F,<br/>Series 21-TA)[Description]Parameter data in the CNC can be read directly from the CNC via the<br/>FANUC bus.This function is basically the same as the function described in Section<br/>B.4.6 "Reading a Parameter," except that the function code is 154 and

some of the completion codes are different.



[Input data structure]

- 0: Parameter data has been read normally.
- 3: The parameter No. specified to be read is invalid.
- 4: A value other than 0, -1, and 1 to n (number of axes) was specified for the data attribute.
- 6: An option required for setting the parameter to be read, such as the error compensation option, is not provided.

B.4.42[Description]Reading Set Data<br/>(not available for<br/>Power Mate-D/F,<br/>Series 21-TA)Set data stored in the CNC can be read directly from the CNC via the<br/>FANUC bus.B.4.8 "Reading Set Data," except that the function code is 155 and some<br/>of the completion codes are different.



#### [Input data structure]

- 0: Set data has been read normally.
- 3: The set data No. specified to be read is invalid.
- 4: A value other than 0, -1, and 1 to n (number of axes) was specified for the data attribute.

# B.4.43[Description]Reading DiagnosisData displayed on the diagnosis data screen of the CNC can be read<br/>directly from the CNC via the FANUC bus.Dower Mate-D/F,<br/>Series 21-TA)This function is basically the same as the function described in Section<br/>B.4.22 "Reading Diagnosis Data," except that the function code is 156<br/>and some of the completion codes are different.



[Input data structure]

- 0: Diagnosis data has been read normally from the CNC.
- 3: The diagnosis No. specified to be read is invalid.
- 4: A value other than 0, -1, and 1 to n (number of axes) was specified for the data attribute.
- 6: An option required for using the diagnosis data to be read, such as the remote buffer option, is not provided.

# B.4.44 Reading a Character String of the CNC Program Being Executed in the Buffer

[Description]

In a machining program being executed on the CNC, the block currently executed, the next block, and the next block but one can be read in the CNC program format. That is, these blocks can be read in the form of a character string of ASCII codes. This function is available only with the M series CNCs.

Comments in a block can also be read.

APPENDIX

The maximum number of characters in a character string is fixed to 64.



[Input data structure]

- 1 When data specified by the NC is a macro statement, the character string cannot be read correctly.
- 2 When data attribute M is set to 2, the next block but one can be read only when the next block is an instruction for tool diameter compensation C.

- 0: The character string of the CNC program being executed in the buffer has been read normally.
- 4: The value specified for the data attribute is invalid.

#### [Output data structure]



#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.45[Description]Reading the Relative<br/>Position on a<br/>Controlled Axis[Description]<br/>The relative coordinates of the machine moving along an axis controlled<br/>by the CNC can be read.



- 0: The relative coordinates on the controlled axis have been read normally.
- 4 : The specified data attribute is invalid. That is, a value other than -1 and 1 to n (number of axes) was specified, or the specified axis No. was greater than the number of controlled axes.

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

Top address + 0			
	(Function code)		
	74		
2	(Completion code)		
	(See the explanation of the completion codes.)		
4	(Data length)		
	(L = 4*n. n is the number of specified axes.)		
6			
	(Data number)		
8			
-	(Data attribute)		
	(M: Input data)	Value	
10	Relative coordinates on the specified controlled axis (4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
When the number	of controlled axes is 4		
		l	
		Value	
10	Relative coordinates on the first axis (4 bytes)	Signed binary (A negative value is represented in 2's	
14	Relative coordinates on the second axis (4 bytes)	complement.)	
18	Relative coordinates on the third axis (4 bytes)		
22	Relative coordinates on the fourth axis (4 bytes)		

[Output data structure]

#### Output data unit

Input system	Increment system IS-B	Increment system IS-C
mm, deg system	0.001	0.0001
inch system	0.0001	0.00001
mm dag ovetem	0.001	0.0001
nin, deg system	0.001	0.0001
inch system	0.0001	0.00001
men system	0.0001	0.00001

Double values can be read for a machining center system or when radius specification is used for the relevant axis of a lathe system.

**B.4.46** 

Travel

**Reading the Remaining** 

[Description]

The remaining travel of the machine along an axis controlled by the CNC can be read. The read value equals the remaining travel indicated on the current position display screen on the CNC. (This screen can be called by pressing the function button POS.)



[Input data structure]

APPENDIX

- 0: The remaining travel along the controlled axis has beenread normally.
- 4 : The specified data attribute is invalid. That is, a value other than -1 and 1 to n (number of axes) was specified, or the specified axis No. was greater than the number of controlled axes.

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

#### B-61863E/14

Top address + 0	(Function code) 75		
2	(Completion code)		
	(See the explanation of the completion codes.)		
4	(Data length) L		
6	(L = 4"n. n is the number of specified axes.)		
	(Data number) —		
8	(Data attribute)		
	M (M: Input data)	Value	
10	Remaining travel along the specified controlled axis (4 bytes)	Signed binary (A negative value is represented in 2's complement.)	
When the number	r of controlled axes is 4		
		Value	
10	Remaining travel along the first axis (4 bytes)	Signed binary (A negative value is represented in 2's	
14	Remaining travel along the second axis (4 bytes)	complement.)	
18	Remaining travel along the third axis (4 bytes)		
22	Pompining travel along the fourth axis	1	

# [Output data structure]

# Output data unit

		Input system	Increment system IS–B	Increment system IS–C
Machining center system		mm, deg system	0.001	0.0001
Powe	er Mate–D, F	inch system	0.0001	0.00001
	Radius specification	mm, deg system	0.001	0.0001
Lathe system	Diameter specification		0.0005	0.00005
	Radius specification		0.0001	0.00001
	Diameter specification		0.00005	0.000005

# B.4.47 Reading CNC Status Information

B-61863E/14

#### [Description]

Status information (status indication on the screen) can be read from the CNC.

The types of status information that can be read are as follows.

- (1) Indication of which mode is selected, automatic or manual
- (2) Status of automatic operation
- (3) Status of movement along the axis and dwelling
- (4) Status of M, S, T, and B functions
- (5) Statuses of emergency stop and the reset signal
- (6) Alarm status
- (7) Status of program edit

#### (Indication)



[Input data structure]



0: CNC status information has been read normally.

# [Output data structure]

Top address + 0		]	
	(Function code) 76		
2	(Completion code)		
	(See the explanation of the completion codes.)		
4	(Data length) 14		
6	(Data number)		
	(Input data)		
8	(Data attribute)		
	(Input data)	Value	
10	Indication of which mode is currently selected, automatic or manual (2 bytes)	0 : MDI 1 : MEMory 2 :**** (Other states) 3 :EDIT 4 : HaNDle 5 : JOG 6 :Teach in JOG 7 :Teach in HND 8 : INC. feed 9 : REFerence 10: ReMoTe	
12	Status of automatic operation (2 bytes)	0 :**** (Reset states) 1 :STOP 2 :HOLD 3 :STaRT	
14	Status of movement along the axis or dwelling (2 bytes)	0 :*** (Other states) 1 :MoTioN 2 :DWell	
16	Status of M, S, T, and B functions (2 bytes)	0 :*** (Other states) 1 :FIN	
18	Status of emergency stop (2 bytes)	<ul> <li>0 : (Releases the emergency stop state)</li> <li>1 :——EMerGency——</li> <li>2 :— RESET — (The reset signal is on.)</li> </ul>	
20	Alarm status (2 bytes)	0 :*** (Other states) 1 :ALarM 2 :BATtery low	
22	Status of program edit (2 bytes)	0 :******* (Non editing) 1 :EDIT 2 :SeaRCH 3 :OUTPUT 4 :INPUT 5 :COMPARE 6 :LabelSKip 7 :OFST 8 :WSFT 9 :ReSTaRt	

B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.48 Reading Value of the P-code Macro Variable (*Low-speed Response)

#### [Description]

This function gets the value of variable for Macro–compiler (P–code macro variable) of specified number.

The extended P-code macro variable is not able to be read.

[Input data structure]

APPENDIX



#### CAUTION

The 'Data number' occupies 4 bytes instead of 2 bytes of usual data structure.
- 0: Success to read the value of P-code macro variable.
- 3 : The P-code macro variable specified by 'Data number' is not able to be read.
- 5: The value of the P-code macro variable is out of range  $(\pm 0.0000001 \pm 99999999)$ .
- 6: No option, or no Macro ROM module.

### [Output data structure]



B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

## B.4.49 Writing Value of the P-code Macro Variable (*Low-speed Response)

### [Description]

This function stores the value into the variable for Macro–compiler (P–code macro variable) of specified number.

The extended P-code macro variable is not able to be written into.

[Input data structure]

APPENDIX



### CAUTION

The 'data number' occupies 4 bytes instead of 2 bytes of usual data structure.

- 0: Success to store the value into P-code macro variable.
- 2: The data length has illegal data (is not 6).
- 3: The P-code macro variable specified by 'Data number' is not able to be written.
- 6: No option, or no Macro ROM module.

### [Output data structure]



APPENDIX

# B.4.50[Description]Reading the Tool LifeThis function gets the Tool life counter type of specified tool group in the<br/>Tool life management data. (M series only)Management Data<br/>(Tool Life Counter<br/>Type)Tool life management data. (M series only)

### [Input data structure]



### CAUTION

About Tool group number (in 'Data number') "0" as Tool group number indicates the Tool group currently used. When Tool group has never specified since power-on, "0" of Tool group number results "0" as counter type. "0" of counter type will be returned on T series.

- 0: Success to read the Tool life counter type.
- 3 : The Tool group number is out of range from 0 to 512, or exceeds the maximum number of registered Tool group.
- 6: No option for Tool life management.

### [Output data structure]



- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

B–61863E/14	APPENDIX	SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)
		(PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/

## B.4.51 Registering the Tool Life Management Data (Tool Group) (*Low-speed Response)

### [Description]

This function registers the Tool group in Tool life management data, with Tool number, length of life and Tool life counter type. On T series, the Tool life counter type will be specified by the NC parameter "LTM" (No.6800#2), and this function cannot set/change the counter type.

**B. WINDOW FUNCTION DESCRIPTION** 





### NOTE

The tool life count is changed by parameter 6801#5. 6801#5 = 0: the life is counted every 4seconds. 6801#5 = 1: the life is counted every 1second.

[Completion codes]

- 0: Success to register the Tool group.
- 3: The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool number in 'Data attribute' has wrong value.
- 5: The length of Tool life in 'Data area' is out of range. The Tool life counter type does not match on T series.
- 6: No option for Tool life management.

### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX



### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

### 





### NOTE

The tool life count is changed by parameter 6801#5. 6801#5 = 0: the life is counted every 4seconds. 6801#5 = 1: the life is counted every 1second.

- 0: Success to set the length of Tool life.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 5: The length of Tool life is out of range.
- 6: No option for Tool life management.
- 13 : The data of the currently selected tool group or the next tool group cannot be rewritten. An attempt was made to rewrite the data of the currently selected tool group or the next group.



### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.53 [Description] Writing the Tool Life This function sets the Tool life counter in the specified Tool group in the Tool life management data. Management Data Tool life management data. (Tool Life Counter) + Low-speed Response) - Speed





### NOTE

The tool life count is changed by parameter 6801#5.

- 6801#5 = 0: the life is counted every 4seconds.
- 6801#5 = 1: the life is counted every 1second.

- 0: Success to set the Tool life counter.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 5: The value for Tool life counter is out of range.
- 6: No option for Tool life management.

### [Output data structure]



- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

B. WINDOW FUNCTION DESCRIPTION
(PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/
SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

## B.4.54 Writing the Tool Life Management Data (Tool Life Counter Type) (*Low-speed Response)

### [Description]

APPENDIX

This function sets the Tool life counter type of specified Tool group in the Tool life management data. (M series only)

### [Input data structure]



- 0: Success to set the Tool life counter type.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 5: The value for Tool life counter type is wrong.
- 6: No option for Tool life management.

### [Output data structure]



- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

B-61863E/14	B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ APPENDIX SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)
B.4.55	[Description]
Writing the Tool Life	This function sets the Tool length offset number of the specified Tool
Management Data	group in the Tool life management data. (M series only)
(Tool Length Offset	
Number (1) : Tool	
Number) (米Low–speed	

### [Input data structure]

Response)



- 0: Success to set the Tool length offset number.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool number in 'Data attribute' has wrong value.
- 5: The Tool number is not found in the Tool group.
- 6: No option for Tool life management.



### [Output data structure]

### CAUTION

The effective value for Tool length offset number depends on Tool compensation number available on NC.

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

3–61863E/14	APPENDIX	SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)
		B. WINDOW FUNCTION DESCRIPTION (PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/

B.4.56 Writing the Tool Life Management Data (Tool Length Offset Number (2) : Tool Operation Sequence Number) (*Low-speed Response)

### [Description]

This function sets the Tool length offset number of the Tool of the specified Tool operation sequence number in the Tool life management data. (M series only)

### [Input data structure]



- 0: Success to set the Tool length offset number.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool operation sequence number is wrong.
- 6: No option for Tool life management.

### [Output data structure]



### CAUTION

The effective value for Tool length offset number depends on Tool compensation number available on NC.

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

B-61863E/14	APPENDIX	(PMC-PA1/PA3/SA1/SA2/SA3/SA3/SA) SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)
B.4.57	[Description]	
Writing the Tool Life Management Data (Cutter Compensation	This function sets the Cu group in the Tool life m	atter compensation number of the specified Tool anagement data. (M series only)

**B. WINDOW FUNCTION DESCRIPTION** 

### [Input data structure]

Number (1) : Tool Number) (*Low-speed

Response)



- 0: Success to set the Cutter compensation number.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool number in 'Data attribute' has wrong value.
- 5: The Tool number is not found in the Tool group.
- 6: No option for Tool life management.



### [Output data structure]

### CAUTION

The effective value for Cutter compensation number depends on Tool compensation number available on NC.

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

		B. WINDOW FUNCTION DESCRIPTION (PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/
B-61863E/14	APPENDIX	SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

B.4.58 Writing the Tool Life Management Data (Cutter Compensation Number (2) : Tool Operation Sequence Number) (*Low-speed	[Description] This function sets the Cutter compensation number of the Tool of the specified Tool operation sequence number in the Tool life management data. (M series only)
Number) ( <b>米Low–speed</b> Response)	



### [Input data structure]

- 0: Success to set the Cutter compensation number.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool operation sequence number is wrong.
- 6: No option for Tool life management.

### [Output data structure]



### CAUTION

The effective value for Cutter compensation number depends on Tool compensation number available on NC.

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

B-61863E/14	APPENDIX SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)
B.4.59	[Description]
Writing the Tool Life	This function sets the Tool condition of the specified Tool group in the
Management Data	Tool life management data.(M series only)
(Tool Condition (1) :	
Tool Number)	
(*Low–speed	
Response)	

B. WINDOW FUNCTION DESCRIPTION (PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/

### [Input data structure]



— 1183 —

- 0: Success to set the Tool condition.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool number in 'Data attribute' has wrong value.
- 5: The Tool number is not found in the Tool group.
- 6: No option for Tool life management.



[Output data structure]

This function changes Tool condition as below.

command	before call		after o	all
clear	skip	(#)	usable	( )
	skip	( <b>#</b> )	in use	(@)
	consumed	(*)	usable	( )
skip	unused	( )	skip	(#)
	in use	(@)	skip	(#)
	consumed	(*)	skip	(*)

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

B61863E/14	APPENDIX	B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)
B.4.60	[Description]	
Writing the Tool Management Data (Tool Condition (2) : Tool Operation	This function sets the Tool condition of the Tool of the specified Tool operation sequence number in the Tool life management data.	



Sequence Number)

(*Low–speed Response)



- 0: Success to set the Tool condition.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool operation sequence number is wrong.
- 6: No option for Tool life management.

### [Output data structure]



This function changes Tool condition as shown in B.4.59.

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

### **B.4.61** [Description] Writing the Tool Life **Management Data** (Tool Number) (*Low-speed **Response**)

This function registers a tool to the specified Tool group in the Tool life management data.

### [Input data structure]



- 0: Success to register the Tool number.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool operation sequence number is wrong.
- 6: No option for Tool life management.

### [Output data structure]



- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function.

### B.4.62 Reading the Estimate Disturbance Torque Data

Power Mate	FS20	FS18	FS16
Δ	×	Δ	Δ

 $\Delta$ : The support is decided by CNC series

×: No support

### CAUTION

The abnormal load detection function option is required. For detailed settings of parameters and so forth, refer to the description of abnormal load detection in the connection manual (functions).

### (1) servo axis

[Description ]

The load torques except a necessary torque for acceleration/deceleration of the digital servo axis are read.





- 0: The estimate disturbance torque data have been read normally.
- 4 : The data specified as the data attribute is invalid because it is neither -1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.



### [Output data structure]

APPENDIX

(2) spindle axis

### [Description]

The load torques except a necessary torque for acceleration/deceleration of the serial spindle axis are read.

[Input data structure]



- 0: The estimate disturbance torque data have been read normally.
- 4: The data specified as the data attribute is invalid because it is neither 1 nor a value from 1 to n (n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.

[Output	data	structure]
---------	------	------------



### [supporting soft]

CNC	:	<b>B005 SERIES</b>	Edition K or later
		<b>B105 SERIES</b>	Edition H or later
		<b>B205 SERIES</b>	Edition H or later
		<b>BD03 SERIES</b>	Edition L or later
		<b>BE03 SERIES</b>	Edition I or later
		<b>BF03 SERIES</b>	Edition I or later
SERVO	:	9060 SERIES	Edition J or later
SPINDLE	:	9A50 SERIES	Edition Q or later
. ~ .			

* Some of the series not listed above are supported by the first edition.

APPENDIX

B-61863E/14

B.4.63	[Description]
Reading the Current	This function reads CNC program numbers extended to 8 digits from the
Program Number	usual 4 digits.
(8–digit Program	Basically, this function is the same as function number 24 excluding the
Numbers)	different data length of function code 90.
(not available for	
Power Mate–D/F,	
Series 21–TA)	[Input data structure]



### [Completion codes]

- -1: The read command of the currently executing program could not be executed. That is, the same command could not be executed as the data of the program number was being updated on the CNC.
- 0: The program number of the currently executing program has been read normally.

### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

### B-61863E/14



### [Output data structure]

APPENDIX

### B.4.64 Reading Tool Life Management Data (Tool Group Number)

[Description]

This function reads the tool group number to which the tool number is currently registered.



### [Input data structure]

### NOTE

When the tool number is set to "0", the tool group number of the currently used tool is read.

If a tool group number is not specified after the power is turned ON, tool group number "0" is read. Also, if a tool number is registered to two or more tool group numbers, the tool group numbers of all tool groups to which the tool number is registered are read.

- 0: The tool group number was read successfully.
- 4: The tool number in 'Data Attribute' has a wrong value.
- 5: The tool number is not registered.
- 6: The tool life management option has not been added on.

### [Output data structure]



- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

## B.4.65[Description]Reading Tool LifeThis function reads the tool length offset number according to the<br/>specified tool group number and tool number. (M series only)Management Data<br/>(Tool Length Offset<br/>Number 1)This function reads the tool length offset number according to the<br/>specified tool group number and tool number. (M series only)





### CAUTION

When the tool group number and tool number are set to "0", the currently used tool group and tool number are referenced.

If a tool group number is not specified after the power is turned ON, tool group number "0" is read.

"0" is always read on the T series.
- 0: The tool length offset number was read successfully.
- 3: The specified tool group number is incorrect.
- 4: The specified tool number is incorrect.
- 5 : The specified tool number is not registered to the specified tool group.
- 6: The tool life management option has not been added on.



### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

APPENDIX

## B.4.66[Description]Reading Tool LifeThis function reads the tool radius offset number according to the<br/>specified tool group number and tool number. (M series only)Management Data<br/>(Tool Diameter Offset<br/>Number 1)[Description]

### [Input data structure]



### CAUTION

When the tool group number and tool number are set to "0", the currently used tool group and tool number are referenced.

If a tool group number is not specified after the power is turned ON, tool group number "0" is read.

"0" is always read on the T series.

- 0: The tool radius offset number was read successfully.
- 3: The specified tool group number is incorrect.
- 4: The specified tool number is incorrect.
- 5: The specified tool number is not registered to the specified tool group.
- 6: The tool life management option has not been added on.



### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

APPENDIX

### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

### B.4.67[Description]Reading Tool Life<br/>Management Data<br/>(Tool Information 1)This function reads the tool information (status) according to the specified<br/>tool group number and tool number.

[Input data structure]

### Top address 0 (Function code) 201 2 (Completion code) (Need not be set) 4 (Data length) (Need not be set) 6 (Data number) Ν (N=Tool group number) 8 (Data attribute) Μ (M=Tool number) 12 ((Data area)) (Need not be set) 16

### CAUTION

When the tool group number and tool number are set to "0", the currently used tool group and tool number are referenced.

If a tool group number is not specified after the power is turned ON, tool group number "0" is read.

- 0: The tool information was read successfully.
- 3: The specified tool group number is incorrect.
- 4: The specified tool number is incorrect.
- 5: The specified tool number is not registered to the specified tool group.
- 6: The tool life management option has not been added on.



### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

		B. WINDOW FUNCTION DESCRIPTION
		(PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/
B-61863E/14	APPENDIX	SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

### B.4.68 Writing (Registering) Tool Life Management Data (Tool Group Number) (*Low-speed Response)

### [Description]

This function registers the tool group number to tool life management data. Set the tool number, life value and life counter type to the specified tool group. On the T series, since the life counter type is specified by CNC parameter LTM (No. 6800#2), it cannot be set nor changed here.

[Input	data	structure]
--------	------	------------



### NOTE

The tool life count is changed by parameter 6801#5. 6801#5 = 0: the life is counted every 4seconds. 6801#5 = 1: the life is counted every 1second.

6801#5 = 1: the life is counted every 1second.

- 0: The tool length was registered successfully.
- 3 : The tool group number exceeded the range 1 to 512 or maximum number of registered groups.
- 4: The tool number in 'Data Attribute' has a wrong value.
- 5: The tool life value is out–of–range. On the T series, the tool life counter type is different.
- 6: The tool life management option has not been added on.



### [Output data structure]

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

# B.4.69[Description]Writing Tool LifeThis function sets the tool length offset number of a specified tool group<br/>in the tool life management data. (M series only)Management Data<br/>(Tool Length Offset<br/>Number 1)<br/>(*Low-speed<br/>Response)[Description]

### [Input data structure]



- 0: The tool length offset number was written successfully.
- 3: The tool group number exceeded the range 1 to 512 or maximum number of registered groups.
- 4: The specified tool number is incorrect.
- 5: The specified tool number is not registered to the specified tool group.
- 6: The tool life management option has not been added on.

[Output data structure]



### CAUTION

The tool length offset number that can be actually specified is reliant on the tool offsets available on the NC.

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

B.4.70 Writing Tool Life Management Data (Tool Radius Offset Number 1) (*Low-speed Response)

### [Description]

This function sets the tool radius offset number of a tool belonging to a specified tool group in the tool life management data. (M series only)

### [Input data structure]



- 0: The tool radius offset number was written successfully.
- 3: The tool group number exceeded the range 1 to 512 or maximum number of registered groups.
- 4: The specified tool number is incorrect.
- 5: The specified tool number is not registered to the specified tool group.
- 6: The tool life management option has not been added on.
- [Output data structure]



### CAUTION

The tool length offset number that can be actually specified is reliant on the tool offsets available on the NC.

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

### B-61863E/14

### B.4.71 Writing Tool Life Management Data (Tool Information 1) (*Low-speed Response)

### [Description]

This function sets the tool information of a tool belonging to a specified tool group in the tool life management data. (M series only)

### [Input data structure]



- 0: The tool information was written successfully.
- 3 : The tool group number exceeded the range 1 to 512 or maximum number of registered groups.
- 4: The specified tool number is incorrect.
- 5 : The specified tool number is not registered to the specified tool group.
- 6: The tool life management option has not been added on.

[Output data structure]



The following table shows how the tool status changes before and after this function is specified.

command	Pre-command Status		Post-command Status	
clear	skip	(#)	unused	( )
	skip	(#)	in use	(@)
	used	(*)	unused	( )
skip	unused	( )	skip	(#)
	in use	(@)	skip	(#)
	used	(*)	skip	(*)

- 1 This cannot be used with those models that do not have the tool life management function.
- 2 Tool life management data is not applied to the T mode of the compound machining function. And compound machining function is not applied to the function of tool life management data B.

### B.4.72 Reading Actual Spindle Speeds

(1) Actual spindle speed

APPENDIX

[Description]

This function reads the actual speed of the No.1 to No.4 serial spindles.

[Input data structure]



### [Completion codes]

- 0: The actual spindle speed was read successfully.
- 4: The spindle speed in 'Data Attribute' has wrong values, that is , a value outside of the range -1 to -(n-1) or 1 to n (n: number of spindles).



### [Output data structure]

(2) Position coder-less actual spindle speed

### [Description]

This function reads the actual spindle speed (position coder–less actual spindle speed) obtained by calculating the spindle motor speed of the No.1 to No.4 serial spindles.

[Input data structure]



### [Completion codes]

- 0: The actual spindle speed was read successfully.
- 4 : The spindle speed in 'Data Attribute' has wrong values, that is , a value outside of the range -11 to -(9+1) or 11 to (10+n) (n: number of spindles).



[Output data structure]

### B.4.73 Reading Fine Torque Sensing Data (Statistical Calculation Results)

### [Description]

This function reads the statistical calculation results (average value, maximum value, distribution) in the fine torque sensing function.

[Input data structure]

APPENDIX



### [Completion codes]

- 0: The statistical calculation results were read successfully.
- 3 : The fine torque sensing data in 'Data Attribute' has a wrong value, that is, a value outside of the range -1 or 1 to n (n: number of spindles).
- 6: The fine torque sensing option has not been added on.



### [Output data structure]

		Value
-10	Average value of target axis 1	Signed binary
12	Maximum value of target axis 1	specified to parameter Nos. 6390
14	Distribution of target axis 1	- 10 6363)
16	Average value of target axis 2	
18	Maximum value of target axis 2	
20	Distribution of target axis 2	
	:	
	:	
	:	
32	Average value of target axis 4	
34		

## B.4.74(1) Store counterReading Fine Torque<br/>Sensing Data<br/>(Store Data)[Description]This function reads the number of stored torque data items.



APPENDIX

### [Completion codes]

- 0: The store counter was read successfully.
- 3: Incorrect data number, that is, a value other than 0 is specified.
- 4: The fine torque sensing data in 'Data Attribute' has wrong values, that is, a value other than 01.
- 6: The fine torque sensing option has not been added on.

### B-61863E/14



### [Output data structure]

(2) Stored torque data (latest data)

APPENDIX

### [Description]

This function reads the latest stored data among stored torque data.





### [Completion codes]

- 0: The stored torque data (latest data) was read successfully.
- 3: Incorrect data number, that is, a value other than 11 to (10+n) (n: number of spindles) is specified.
- 4: The fine torque sensing data in 'Data Attribute' has a wrong value, that is, a value other than 0 or 1.
- 6: The fine torque sensing option has not been added on.

B-61863E/14



[Output data structure]

### NOTE

- 1 When data has not been stored, data is not output, and processing ends successfully with L set to 0.
- 2 When sample data is selected by data attribute, the sample data corresponding to the latest stored data is output.
  - Example) When 10000 sample data items (data numbers 0 to 9999) and latest data items 5000 (data numbers 0 to 4999) are stored, data number 4999 in the latest data is output when data attribute M is set to "0", and data number 4999 in the sample data is output when data attribute M is set to "1".
- When sample data is selected by data attribute, and there is no sample data corresponding to the latest stored data, data is not output, and processing ends successfully with L set to 0.
   Example) When 5000 sample data items (data numbers 0 to 4999) and 10000 latest data items (data numbers 0 to 9999) are stored, data is not output, and processing ends

successfully with L set to 0 when data attribute M is set to "1".

(3) Stored torque data (any data)

[Description]

This function reads the arbitrary data among stored torque data.

[Input data structure]



### NOTE

The valid range of data number n is calculated as follows:  $0 \le n \le (524288 \times \frac{1}{a} \times \frac{1}{b}) - 1$ where,  $a = \begin{cases}
1: \text{ Number of target axes 1} \\
2: \text{ Number of target axes 2} \\
4: \text{ Number of target axes 3 and 4} \\
b = \begin{cases}
1: \text{ Sample data store function OFF} \\
2: \text{ Sample data store function ON} \\
\text{The valid range of number of data items I is calculated as follows:} \\
1 \le l \le 20 \end{cases}$ 

[Completion codes]

- 0: The stored torque data (any data) was read successfully.
- 2: Incorrect data length, that is, a value other than 6 is specified.
- 3: Incorrect data number, that is, a value other than 11 to (10+n) (n: number of spindles) is specified.
- 4: The fine torque sensing data in 'Data Attribute' has a wrong value, that is, a value other than 0 or 1.
- 5: Incorrect data area is specified. See Note for details of value ranges.
- 6: The fine torque sensing option has not been added on.

APPENDIX

### [Output data structure]

Top address +0		1	
·	(Function code) 232		
+2	(Completion code)		
	؛ (See the above explanation.)		
74	(Data length)		
+6	(L=6 + number of data items I $\times$ 2)	_	
	(Data number) N		
+8	(Entered data)		
10	(Data attribute)		
	M (Entered data)	Value	
+10	Data number n (Entered data)	Signedbinary	
+14		-	
	Number of data items I (Entered data)		
+16	Distribution of torgot avec	-	
	Distribution of target axes		
+18 .	Number n data	-	
+20	Number n+1 data		
+22	Number n+2 data		
+24	+24 :		
	:	-	
	:		
	Number n+l-1 data		

### NOTE

- 1 When the number of actually stored data items is exceeded even though data number n is in the valid range, data is not output and processing ends successfully by number of data items I set to 0.
  - Example) When the number of target axes is 2, and the sample data store function is enabled (parameter No.6350#2=1), data numbers 0 to 13107 are valid. However, if an attempt is made to read (example (1) in figure below) data from data number n = 131020 when the number of actually stored data items is 131000 (data numbers 0 to 130999), data is not output, and the number of data items I becomes 0.
- 2 When data number n is within the number of actually stored data items, and (n+I 1) exceeds the number of actually stored data items, data of the stored data items is output, and processing ends successfully. In this case, number of data items I is updated to the number of data items that was output.

Example) If an attempt is made to read (example (2) in figure below) number of data items I (120) from data number 130900 under the same conditions as in the example above, the data of data numbers 130900 to 130999 is output, and number of data items I becomes 100.

Also, if an attempt is made to read (example (3) in figure below) number of data items I (120) from data number 130999 under the same conditions as in the example above, the data of data numbers 130990 to 130999 is output, and number of data items becomes 10.



### B.4.75 Specifying the Number of the Program for I/O Link

[Explanation of data]

APPENDIX

Specify the number of the program to be input/output using the data input/output function with I/O Link.

[Input data structure]



[Completion codes]

- 0: The specification of the program number terminated normally.
- 5 : Invalid data was specified for the program number, i.e., the data falls outside the range of 1 to 9999 or is not –9999.

### Top address +0 (Function code) 194 +2 (Completion code) (See the explanation of completion codes, above.) +4 (Data length) 2 (Data at input time) +6 (Data number) 0 (Data at input time) +8 (Data attribute) 0 (Data at input time) Value +10 Signed binary format Programnumber (Data at input time) +12

### [Output data structure]

### CAUTION

For details of this function, see the section on data input/output functions using I/O Link in the "CNC Connection Manual (Functions)."

B.4.76 Preset of relative coordinate (⊁Low–speed response) (not available for Power Mate and Series 21–TA)

### [Description]

The preset data is set to the relative coordinate controlled by CNC. If 0 is set as preset data it becomes to origin.

But it is impossible to write the value of preset data to the transferring axis. In the case of the preset of relative coordinate of all axes is executed by using this function, if only one axis is transferring, the preset of relative coordinate cannot be executed, neither.

### [Input data structure]

Case of writing data on each axis.

APPENDIX



### Input data unit

		Input system	Increment system IS–B	Increment system IS–C
Machining center system		mm, deg system	0.001 0.0001	
		inch system	0.0001	0.00001
	Radius specification	mm, deg	0.001	0.0001
Lathe system	Diameter specification	system	0.001	0.0001
	Radius specification	inch system	0.0001	0.00001
	Diameter specification	inon system	0.0001	0.00001

### B-61863E/14



Case of writing data on all axes (controlled axes are 4).

### [Completion codes]

- 0: Success to set the value of relative coordinate.
- 4 : Data specified for the data attribute is invalid because it is neither -1 nor a value from 1 to n(n is the number of axes). Alternatively, the specified axis number is greater than the number of controlled axes.
- 5: Relative coordinate is out of range.
- 13 : Axis is moving now.

### B-61863E/14



### [Output data structure]

APPENDIX

### B.4.77 Deleting the Tool life Management Data (Tool group) (*Low-speed response)

(Not available for Power Mate, Series 21-TA)

### [Description]

The function deletes the specified Tool group in the Tool life management data. In short, it makes the condition that Tool group is not registered.



- 0: Success to delete the Tool group number.
- 3 : The Tool group number is out of range from 1 to 512,or exceeds the maximum number of registered Tool group.
- 6: No option for Tool life management.

### [Output data structure]



- 1 Those models that do not provide the tool life management function cannot be used.
- 2 The turning mode of the complex machining function does not support tool life management data.
#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

# B.4.78 Deleting the Tool life Management Data (Tool data) (*Low-speed response)

## [Description]

The function deletes the Tool data of the Tool of the specified Tool operation sequence number in the Tool life management data. (M series only)

### [Input data structure]



- 0: Success to delete the Tool group number.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 4: The Tool operation sequence number is wrong.
- 6: No option for Tool life management.

#### [Output data structure]



### NOTE

- 1 Those models that do not provide the tool life management function cannot be used.
- 2 The turning mode of the complex machining function does not support tool life management data.

#### B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4) APPENDIX

## B.4.79 Deleting the Tool life Management Data (Tool life counter and Tool condition) (*Low-speed response)

### [Description]

The function clears the Tool life counter and all Tool condition of the specified Tool group in the Tool life management data.

#### [Input data structure]



- 0: Success to clear the Tool life counter and the Tool condition.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 6: No option for Tool life management.

#### Top address +0 (Function code) 326 +2 (Completion code) ? (See the explanation above) +4 (Data length) 0 (Same as input data) +6 (Data number) Ν (Same as input data) +8 (Data attribute) 0 (Same as input data) +10

#### [Output data structure]

### NOTE

- 1 Those models that do not provide the tool life management function cannot be used.
- 2 The turning mode of the complex machining function does not support tool life management data.

# B.4.80 Writing the Tool life Management Data (Arbitrary group number) (*Low-speed response)

[Description]

This function sets Arbitrary group number of the specified Tool group in the Tool life management data.

[Input	data	structure]
--------	------	------------



### NOTE

Writing the Tool life Management Data (Arbitrary group number) is available for Tool life management data B

- 0: Success to set the Arbitrary group number.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 5: Arbitrary group number is out of range.
- 6: No option for Tool life management.

#### [Output data structure]



#### NOTE

- 1 Those models that do not provide the tool life management function cannot be used.
- 2 The turning mode of the complex machining function does not support tool life management data.
  The complex machining function does not support the tool life management B function.

# B.4.81 Writing the Tool life Management Data (Remaining tool life) (*Low-speed response)

(Not available for Power Mate, Series 21-TA)

[Description]

This function sets the length of Remaining Tool life of the specified Tool group in the Tool life management data.

[Input data structure]



### NOTE

Writing the Tool life Management Data (Remaining tool life) is available for Tool life management data B

- 0: Success to set the length of Remaining tool life.
- 3 : The Tool group number is out of range from 1 to 512, or exceeds the maximum number of registered Tool group.
- 5: Remaining tool life is out of range.
- 6: No option for Tool life management.

#### [Output data structure]



#### NOTE

- 1 Those models that do not provide the tool life management function cannot be used.
- 2 The turning mode of the complex machining function does not support tool life management data.
  The complex machining function does not support the tool life management B function.

# B.4.82 Reading the Current Screen Number

[Description]

The current screen number to display can be read.

[Input data structure]



APPENDIX

- 0: Reading the current screen number has been read normally.
- 1: This function can not be used because series and edition of CNC or PMC system software is not corresponded.

### [Output data structure]



### NOTE

This function is used for Power Mate *i*–D/H only.

[The screen	number	of Power	Mate	<i>i</i> –D/H ]
-------------	--------	----------	------	-----------------

#### Table B.4.82

Large	Large classification		Small classification			
00	POSITION	00	ABSOLUTE			
		01	RELATIVE			
		02	ALL			
		03	HANDLE INTERRUPT			
		04	OPERATING MONITOR			
		05	C Executor			
01	PROGRAM	00	MDI PROGRAM			
		01	PROGRAM			
		02	LIBRARY			
		03	CURRENT BLOCK			
		04	NEXT BLOCK			
		05	PROGRAM CHECK			
		07	FLOPPY DIRECTORY			
		0d	C Executor			
02	OFFSET	00	OFFSET			
		01	SETTING PARAMETER			
		03	MACRO VARIABLE			
		04	MENU			
		05	OPERATOR'S PANEL			
		0a	C Executor			
03	SYSTEM	00	PARAMETER			
		01	DIAGNOSE			
		02	PMC			
		03	SYSTEM			
		04	MEMORY			
		06	SERVO SETTING			
		07	SPINDLE SETTING			
		0b	OPERATION HISTORY			
		0c	ALL I/O			
		13	PERIODICAL MAINTENANCE DISP.			
		14	MAINTENANCE INFORMATION DISP.			
		18	FSSB SETTING DISPLAY			
		19	C Executor			
04	MESSAGE	00	ALARM			
		01	EXTERNAL MESSAGE			
		02	ALARM HISTORY			
		0a	C Executor			
05	USER	01	MACRO1			
		02	MACRO2			
		03	MACRO3			

B. WINDOW FUNCTION DESCRIPTION (PMC–PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SB7/SC/SC3/SC4)

# B.4.83 Reading Detailed Alarm Information (*Low-speed type)

### [Description]

If the CNC is in the alarm status, the details can be read. Detailed information includes the following:

(a) Alarm status information

APPENDIX

- Information of the type of alarm which occurs
- (b) Detailed alarm information
  - Axis information of the alarm which occurs and alarm number information
- (1) Reading alarm status information

#### [Input data structure]



0: Reading the CNC alarm status has terminated normally.

### [Output data structure]



Alarm status information

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	bit
Bit 0:	P/S alarm 100 (PS1) (The alarm write enable setting PWE is set to 1.)																
Bit 1:		P/S alarm 000 (PS2) (Turn the power off once. A written parameter may cause this alarm.)															
Bit 2:	P/S alarm 101 (PS3) (The part program storage area is disordered. If the power to the CNC is turned off during part program editing or machining program reading, this alarm occurs. To release this alarm, press the RESET key while holding the PROG key down.)																
Bit 3:			А	P/S	ala	rm	oth	er t	han	the	ab	ove	oco	curs	s. (I	PS)	(Up to 255)
Bit 4:			St	rok	e li	mit	ala	rm	(0	ΓS)							
Bit 5:			0	verl	heat	t ala	ırm	(0	H)								
Bit 6:			Se	ervo	o ala	arm	(S	V)									
Bit 7:			N	ot u	ised												
Bit 8:			A	PC	ala	rm	(AF	PAL	.)								
Bit 9:			SI	pinc	lle a	alar	m (	SP	A)								
Bit 10:			P/	'S a	larr	n ha	avii	ng a	ı nu	mb	er 5	500	0 01	: aft	er (	(PS_	_2)
Bits 11	to	15:	N	ot u	ised												

(2) Reading detailed alarm information

APPENDIX

### [Input data structure]



0: Reading the CNC alarm status has terminated normally.

[Output data structure]



• Axis information

For an axis-type alarm, the bit indicating the corresponding axis number is on.

(When an alarm occurs on axis 1, bit 0 is 1.)

For a non-axis-type alarm, the bit is 0.



C.1 FUNCTION	This window function is a functional instruction by which the data on the CNC is read or is written.				
	Option (FS15B : A02B–0162–J917, FS15 $i$ : A02B–0261–J950) of NC window is necessary.				

# C.2 LOW–SPEED RESPONSE AND HIGH–SPEED RESPONSE OF WINDOW FUNCTION

In the way to process, there are window function high speed and one processed at low speed.

In case of a low–speed response, The data is read or written by the control between CNC and PMC.

Therefore, it is necessary to ACT=1 of the window instrucion must be held until the transfer completion information (W1) becomes 1 (interlock).

In a high–speed response, it is not necessity for take the interlock because the data is directly read.

To read tool offset data, tool life management data, and the processing time, the FS15B requires the installation of the NC window B option (A02B–0162–J984), in addition to the NC window option.

## CAUTION

1 The window instruction of a low–speed response is controlled exclusively with the other window instructions of low–speed response.

Therefore, when the data is read or written continuously, it is necessary to clear ACT of the functional instruction once when the completion information (W1) become 1.

It does not work about ACT=1 of the other window instructions of low-speed response such as W1=1 and ACT=1 of the window instruction of a low-speed response.

The window instruction of a high–speed response is not exclusively controlled like a low–speed response. Therefore, when the data is read or written continuously, yow need not make ACT=0. The scan number of times to complete the processing is summarized on the following table.

TYPE	SCAN TIMES UNTIL PROCESSING ENDS
LOW	TWO SCAN TIMES OR MORE (This depends on the state of CNC)
HIGH	1SCAN TIME

(Only FS15B)

2 There is a version which does not support in the reading or writing of the window data by a new format.

<b>ROM VERSION</b>		CONTENT					
4047	A – E	It does not support a new format. Please use #4 of NC parameter 7401 as 0.					
	F –	It supports a new form.					
4078	A –	when the window function of a new format is used, please set #4 of NC parameter 7401 as 1.					

Function that is effected by #4 of NC parameter 7401.

EUNCTION	CONTENT						
	7401#4						
Tool life management	0	The data of tool life management for 128 sets of tools can be read and written.					
data	1	The data of tool life management for 512 sets of tools can be read and written.					
Tool offset data	0	This function can not be used.					
specified tool number	1	The tool offset data can be read and written.					

3 Functions except the above–mentioned are not related to #4 of NC parameter 7401. If there is no option of the corresponding function, window instructions can not be used. APPENDIX

# C.2.1 Functional Instruction WINDR



Data tura	Type of		Data		
Data type	processing	CTL0	CTL1	CTL2	length
Tool offset data	(low)	13	Offset number	Offset format	4 byte
Work origin offset	(high)	15	0	Axis number	4 byte
Parameter data Setting data	(low)	17	Parameter number	Axis number	4 byte
Custom macro variables	(low)	21	Custom macro number	0	6 byte
CNC alarm state	(low)	23	0	0	2 byte
Current program number	(low)	24	0	0	6 byte
Current sequence number	(low)	25	0	0	6 byte
Actual velocity for controlled axes	(low)	26	0	0	4 byte
Absolute position on controlled axes	(high)	27	0	Axis number	4 byte
Machine position on controlled axes	(high)	28	0	Axis number	4 byte
Skip operation stop position on controlled axes	(low)	29	0	Axis number	4 byte
Servo delay amount on controlled axes	(high)	30	0	Axis number	4 byte
Acceleration/deceleration delay amount on controlled axes	(high)	31	0	Axis number	4 byte
Modal data (G function) (other than G function)	(low)	32	Data type	Specified block	2 byte 6 byte
Diagnosis data	(low)	33	Diagnosis number	0	2 byte
Feed motor load current value General–purpose analog input	(high) (high)	34 34	200 0	Axis number Axis number	2 byte 2 byte

part1

part2

	Type of		Data		
Data type	processing	CTL0	CTL1	CTL2	length
Tool life management data Tool group No. Number of tool groups Number of tools	(low) (low) (low)	38 39 40	0 Tool group No. Tool group No.	Tool No. 0 0	4 byte 4 byte 4 byte
Tool life Tool life counter Tool life counter type Tool length compensation No.1 Tool length compensation No.2 Cutter compensation No.1 Cutter compensation No.2 Tool information 1	(low) (low) (low) (low) (low) (low) (low) (low)	41 42 160 43 44 45 46 46 47	Tool group No. Tool group No.	0 0 Tool No. Tool order number Tool No. Tool order number Tool No.	4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte
Tool information 2 Tool No.	(low) (low)	48 49	Tool group No. Tool group No.	Tool order number Tool order number	4 byte 4 byte
Clock data	(IOW)	151	Data format	0	6 byte
Relative position of controlled axes	(nign)	74	0		4 byte
Remaining travel of controlled axes	(nign)	75	0	Axis number	4 byte
Estimate disturbance torque data of a digital Estimate disturbance torque data of a serial spindle	(high) (high)	211 211	1	Axis number Axis number	2 byte 2 byte
Machining time	(low)	178	Program number	1	6 byte
Load information of the spindle motor	(high)	153	0	Axis number	2 byte
Tool offset data according to the specified tool number	(low)	213	Data format	Tool number	4 byte
Tool life management data (Note) Tool group number (supporting 8–digit tool numbers)	(low)	200	0	Tool number (4 bytes)	4 bytes
Tool length compensation number 1	(low)	227	Tool group number	Tool number (4 bytes)	4 bytes
Cutter compensation number 1 (supporting 8–digit tool numbers)	(low)	228	Tool group number	Tool number (4 bytes)	4 bytes
Tool information 1 (supporting 8–digit tool numbers)	(low)	201	Tool group number	Tool number (4 bytes)	4 bytes
Real parameter data (Note)	(low)	321	Parameter number (4 bytes)	Axis number	4 bytes
Actual machine position on controlled axes (Note)	(high)	329	0	Axis number	4 bytes
Reading Fine Torque Sensing Data (Statistical Calculation Results). (Note)	(low)	226	Data number	Axis number	4 bytes
Reading Fine Torque Sensing Data (Store Data). (Note)	(low)	232	Data attribute	Axis number	4 bytes
Reading detailed information of CNC alarm. (Note)	(low)	330	Alarm type	Alarm count	8 bytes

### NOTE

This function is supported only by the FS15*i*–A (PMC–NB6).

APPENDIX

part1

# C.2.2 Functional Instruction WINDW



Data tura	Type of		Data		
Data type	processing	CTL0	CTL1	CTL2	length
Tool offset data	(low)	14	Offset number	Offset format	4 byte
Parameter data Setting data	(low)	18	Parameter number	Axis number	4 byte
Custom macro variables	(low)	22	Custom macro number	0	6 byte
Data on the program check screen Spindle tool number Number of the tool to be used	(low)	150	Data type	0	2 byte
next	(low)	150	201	0	2 byte
Torque limit override	(low)	152	0	Axis number	2 byte
Tool life management data Number of tool groups Tool life Tool life counter Tool life counter type Tool length compensation No.1 Tool length compensation No.2 Cutter compensation No.1 Cutter compensation No.2 Tool information 1 Tool information 2 Tool No.	(low) (low) (low) (low) (low) (low) (low) (low) (low) (low)	163 164 165 166 167 168 169 170 171 172 173	0 Tool group No. Tool group No.	Tool No. 0 0 Tool No. Tool order number Tool order number Tool order number Tool order number Tool order number	4 byte 4 byte
Tool offset data according to the specified tool number	(low)	214	Data format	Tool number	4 byte
Superposition move command (for three axes) Superposition move command (for four axes)	(high) (high)	215 215	0 Axis specification mode	0 Axis number	6 byte 8 byte
Feedrate	(high)	216	0	0	6 byte

					part2		
Data typo	Type of		Type of control data				
Data type	processing	CTL0	CTL1	CTL2	length		
Tool life management data (Note) Tool group number (supporting 8-digit tool numbers)	(low)	202	0	Tool number (4 bytes)	6 bytes		
Tool length compensation number 1 (supporting 8–digit tool numbers)	(low)	229	Tool group number	Tool number (4 bytes)	4 bytes		
Cutter compensation number 1 (supporting 8–digit tool numbers)	(low)	230	Tool group number	Tool number (4 bytes)	4 bytes		
Tool information 1 (supporting 8–digit tool numbers)	(low)	231	Tool group number	Tool number (4 bytes)	4 bytes		
Real parameter data (Note)	(low)	323	Parameter number (4 bytes)	Axis number	4 bytes		

APPENDIX

**NOTE** This function is supported only by the FS15*i*–A (PMC–NB6).

# C.3 FORMAT AND DETAILS OF THE CONTROL DATA OF THE WINDR FUNCTIONAL INSTRUCTION

- (1) See the description of the window function. The data item marked with a dash (–) in the description of the data structure need not be entered.
- (2) The length of all data blocks and data items is represented in bytes.
- (3) The read data becomes valid only when the instruction terminates normally.

Completion code	Meaning
-10	The window instruction is being processed. Hold ACT until W1 is set to 1.
0	The instruction terminated normally.
1	An error occurred. The corresponding function number is not found.
2	An error occurred. Possible causes include the following: Wrong data is found in the CTL area. The NC does not have the corresponding function.
3	An error occurred. The specified axis is not provided.
5	An error occurred. It is a mistake of data form. Though the function supports only a new format, a old format is specified.

## C.3.1 Reading a Tool Offset (Low–speed Response)

[Description]

The tool offset value (tool compensation) is read from the CNC.



### C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

(Note 1) Offset format
M series (machining center system)

Data type	Format	Offset number (CTL+2, 3)
Tool compensation A Compensation	1	Offset number
Tool compensation B Geometry compensation Wear compensation	1	Offset number Offset number +1000
Tool compensation C Tool length Geometry compensation Wear compensation Cutter Geometry compensation Wear compensation	1 1 2 2	Offset number Offset number +1000 Offset number Offset number +1000

T series (lathe system)						
Data type	Format	Offset number (CTL+2, 3)				
Tool compensation A Compensationalong the X–axis	1	Offset number				
Compensationalong the Z-axis	2	Offset number				
Tool-tip radius com-	3	Offset number				
Compensationalong	4	Offset number				
Compensation re- lated to the position of the virtual tool	5	Offset number Offset number				
Tool compensation B						
Geometry compensation Compensation along the X-axis	1	Offset number				
Compensation	2	Offset number				
Tool-tip radius	3	Offset number				
Compensation along the Y–axis Wear	4	Offset number				
compensation Compensation along the X–axis	1	Offset number +1000				
Compensation along the Z-axis	2	Offset number +1000				
Tool-tip radius compensation	3	Offset number +1000				
Compensation along the Y-axis	4	Offset number +1000				
Compensation related to the position of the virtual tool	5	Offset number				

#### 

## C.3.2 Reading a Workpiece Origin Offset Value

### [Description]

The offset from the workpiece reference point of the current coordinate system (including a shared offset) of the CNC is read.

The offset from the workpiece reference point for each axis can be read individually. The offset from the workpiece reference point for an additional axis can be read only when the additional axis is provided.



# C.3.3 Reading a Parameter (Setting Data) (Low-speed Response)

[Description]

A parameter of the CNC is read.

APPENDIX



### NOTE

A real parameter cannot be read using function code 17. Read a parameter using:

- Real parameter: Function code 321
- Integer or bit parameter: Function code 17

APPENDIX

## C.3.4 Reading a Custom Macro Variable (Low–speed Response)

[Description]

A custom macro variable is read from the CNC.

### NOTE

The position of the decimal point must be specified beforehand.



#### NOTE

In the case of reading a Custom Macro Variable of upper 100000.

Please input "10" to "Data attribute", and input last four digits of variable number to "Custom macro variable number".

## **Examples**

The relationship between the read value and the stored variable is:

(Read value) =

(Custom macro variable in the NC)  $\times 10^{(Position of decimal point)}$ 

Read value	Custom macro variable in the NC	Position of decimal point
1		0
12	1.234	1
123		2
1234		3
12340		4

# C.3.5 Reading the CNC Alarm Status (Low–speed Response)

[Description]

If the CNC is in the alarm state, the details of the alarm are read.



The following alarm states can be read:



#### C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

# C.3.6 Reading the Current Program Number (Low–speed Response)

[Description]

The number of a running machining program is read from the CNC.



# C.3.7 Reading the Current Sequence Number (Low–speed Response)

## [Description]

The sequence number of the running machining program is read from the CNC. If the blocks of the running machining program have no sequence numbers, the sequence number of the block most recently executed is read.



# C.3.8

## Reading the Actual Velocity of Controlled Axes (Low-speed Response)

[Description]

The actual speed of the feed axes controlled by the CNC is read.

The composite speed of the controlled axes is read. If the X–, Y–, and Z–axes, the basic three axes, are controlled as feed axes, the composite speed of the three axes is read.



## C.3.9

## Reading the Absolute Position on a Controlled Axis

#### [Description]

The absolute position (absolute coordinates) on a feed axis controlled by the CNC is read.



#### Data specification



## C.3.10 Reading the Machine Position (Machine [I Coordinates) of T Controlled Axes th

[Description]

The machine position (machine coordinates) on a feed axis controlled by the CNC is read.



- 1) Read value is determined as follows:
  - (1) For the machining center system or when the radius is specified for the axis of the lathe systm. The data shows double of the present position with the least input increment as a unit.
  - (2) When the diameter is specified for the lathe system

The data shows the present position with the least input increment.

(3) When the input unit is multiplied by 10

The data shows twenty-times the present position (radius programming) or ten-times the present position (diameter programming) with the last command increment as a unit.

2) The present position of a moving axis can be read whenever the function instruction is executed.

## C.3.11 Reading a Skip Position (Stop Position of Skip Operation (G31)) of Controlled Axes (Low-speed Response)

[Description]

The absolute coordinates of the skip position specified in the CNC are read.



APPENDIX

- 1) Read value is determined as follows:
  - For the machining center system or when the radius is specified for the axis of the lathe systm.
    The data shows double of the present position with the least input increment as a unit.
  - (2) When the diameter is specified for the lathe system

The data shows the present position with the least input increment.

(3) When the input unit is multiplied by 10

The data shows twenty-times the present position (radius programming) or ten-times the present position (diameter programming) with the last command increment as a unit.

2) Once the skip signal has been input to the NC, movement along the relevant axis is stopped then, after the elapse of the servo delay, the absolute position can be read.

## C.3.12 Reading the Servo Delay for Controlled Axes

### [Description]

A servo delay, which is the difference between the specified position on a controlled axis and the actual servo position, is read from the CNC.



# C.3.13 Reading the Acceleration/ Deceleration Delay on Controlled Axes

#### [Description]

An acceleration/deceleration delay, which is the difference between the programmed position on a controlled axis and the actual position after the acceleration or deceleration, is read from the CNC.



# C.3.14 Reading Modal Data (Low–speed Response)

[Description]

The continuous-state data is read from the CNC.

The continuous–state data can be broadly classified into two types: Data of the preparatory function and data of other functions. When CTL2 (specified block) is set to 0, the continuous–state data of the previous block is read. When CTL2 is set to 2, the continuous–state data of the next block is read.

(1) Continuous-state data of the preparatory function


(2) Continuous-state data of a function other than the preparatory function

The following eleven data items of an NC part program can be read: addresses D, E, H, L, M, N, O, S, T, and F, and second miscellaneous function.



#### Kind of data **Data specification** Modal data 1) Modal data of G function The relationship between the numbers specified in the CTL1 (kinds of data), modal data codes is shown below. Into CTL2 (the specified block), specify 0 (previous data), 1 (present date), or 2 (next data) in accordance with the necessarv modal data. The G code for the lathe system is expressed with the G code system B. Refer to the table indicating the G function system. For example, the G32 of the G code system A corresponds to the G33 of the G code system B. As a result, the code fetched in the DATA + 0 is 4. G code for lathe Specified number in G code for machining Code fetched in system CTL1 (kinds of data) center system DATA + 0 (G code system B) 00 G00 G00 0 G01 G01 1 2 3 G02 G02 G03 G03 4 G33 G33 8 G77 G78 9 G79 10 0 01 G17 G97 G18 G96 1 2 G19 ____ 02 G90 G90 1 G91 G91 0 03 G22 G22 0 G23 G23 1 G93 04 2 G94 G94 0 G95 G95 1 G20 05 G20 1 G21 0 G21 G40 06 G40 0 G41 G41 1 G42 G42 2 G43 07 1 G44 2 G49 0 ____ 08 G80 G80 0 G81 G81 1 2 3 G82 G82 G83 G83 G84 4 G84 5 G85 G85 6 G86 G86 G87 G87 7 G88 G88 8 G89 G89 9 10 G73 G83.1 G74 G84.1 11 G76 G86.1 12 09 G98 G98 0 G99 G99 1 10 G50 0 G51 _ 1

## (3) Data specification

# C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

APPENDIX

Kind of data	Data specification					
Modal data	Specified number in CTL1 (kinds of data)	G code for machining center system	G code for lathe system (G code system B)	Code fetched in DATA + 0		
	11	G66 G67 G66.1	G66 G66 G66.1	1 0 2		
	12	G96 G97	G68 G69	1 0		
	13	G54 G55 G56 G57 G58 G59	G54 G55 G56 G57 G58 G59	0 1 2 3 4 5		
	14	G61 G62 G63 G64	G61 — — G64	1 2 3 0		
	15	G69 G68 —	G17 G18 G19	0 1 2		
	16	G15 G16		0 1		
	17	G50.1 G51.1	G50.1 G51.1	0 1		

## B-61863E/14

Kind of data	Data specification					
Modal data						
	Table — 1 of G code system for a lathe system					
	Go	ode system	*1)			
	A	B	C	Function		
	G00	600	GOO	Positioning		
	G01	G01	G01	Linear interpolation		
	G02	G02	G02	Circular interpolation CW		
	G03	G03	G03	Circular interpolation CCW		
	G04 G07	G04 G07	G04 G07	Hypotherical axis interpolation		
	G09	G09	G09	Exact stop		
	G10	G10	G10	Data setting		
	G10.1	G10.1	G10.1	PC data setting		
	G17	G17	G17	XpYp plane selection Xp: X axis or its parallel axis		
	G18	G18	G18	ZpXp plane selection Yp: Y axix or its parallel axis		
	G19	G19	G19	YpZp plane selection Zp: Z axis or its parallel axis		
	G20 G21	G20 G21	G70 G71	Inch input		
	G22	G22	G22	Stored stroke check on		
	G23	G23	G23	Stored stroke check off		
	G27	G27	G27	Reference point return check		
	G28 G29	G28 G29	G28 G29	Reference point return Return from reference point		
	G30	G30	G30	Return to 2nd, 3rd, 4th reference point		
	G31	G31	G31	Skip function		
	G32	G32	G32	Thread cutting		
	G35	G35	G35	Circular thread cutting CW		
	G36	G36	G36	Circular thread cutting CCW or automatic tool		
	0.07	0.07	0.07	compensation (X axis)		
	G37	G37	G37	Automatic tool compensation #1 or automatic tool		
	G37.1	G37.1	G37.1	Automatic tool compensation #1		
	G37.2	G37.2	G37.2	Automatic tool compensation #2		
	G37.3	G37.3	G37.3	Automatic tool compensation #3		
	G40 G41	G40 G41	G40 G41	Tool nose radius compensation cancel		
	G42	G42	G42	Tool nose radius compensation right		
	G50	G92	G92	Work coordinates change/maximum spindle spped		
	050.1	050.1	050.1	setting		
	G50.1	G50.1 G51.1	G50.1 G51.1	Programmable mirror image		
	G52	G52	G52	Local coordinate system setting		
	G53	G53	G53	Machine coordinate system selection		
	G54	G54	G54	Work coordinate system 1 selection		
	G55 G56	G56	G56	Work coordinate system 2 selection		
	G57	G57	G57	Work coordinate system 4 selection		
	G58	G58	G58	Work coordinate system 5 selection		
	G59 G61	G59 G61	G59 G61	Work coordinate system 6 selection		
	G62	G62	G62	Automatic corner override		
	G64	G64	G64	Cutting mode		
	G65	G65	G65	Macro call		
	G66 1	G66 1	G66 1	Macro modal call A		
	G67	G67	G67	Macro modal call A/B cancel		
	G68	G68	G68	Mirror image for double currets on		
	G69	G69	G69	Mirror image for double currets cancel		
	G70 G71	G70 G71	G72 G73	Stock removal in turning		
		0/1	0,0			

# C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

APPENDIX

Kind of data	Data specification						
Modal data		Tal	ble — 2	2 of G c	ode syster	n for a lathe syst	em
	G	code system	*1)			Euna	tion
	A	В	C	2		Func	
	G72 G73 G74	G72 G73 G74	G G G	74 75 76	Stock rem Pattern re Peck drillir	oval in facing peating ng Z axis	
	G75 G76 G80	G75 G76 G80	G G G	77 78 80	Grooving i Threading Canned cy	n X axis cycle /cle for drilling can	cel
	G81 G82 G83.1 G84 G84.1 G85 G86 G86.1 G87 G88	G81 G82 G83.1 G84 G84.1 G85 G86 G86.1 G87 G88	000000000000000000000000000000000000000	i81 i82 i83 i83.1 i84 i84.1 i85 i86 i86.1 i87 i88	Drilling cyc Drilling cyc Peck drillin Peck drillin Tapping cyc Counter ta Boring cyc Boring cyc Fine borin Back borin Boring cyc	cle, spot boring cle, counter boring ng cycle /cle upping cycle cle cle g cycle ng cycle cle	
	G89 G90 G92 G94 G96 G97 G98 G99 — — — — — —	G89 G77 G78 G96 G97 G94 G95 G90 G91 G98 G99	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	89 20 21 24 96 97 94 95 99 99 99 99 99	Boring cyc Outting cy Thread cu Outting cy Constant s Feed per r Absolute c Increment Canned cy Canned cy	te cle A tting cycle cle B surface speed con surface speed con ninute evolution command al command al command /cle initial level ret /cle R point level r	trol trol urn eturn
	<ul> <li>*1) G code system A/B can be selected by parameter setting (basic function). Gcode sytem C is optinal function. However, when this option is selected, G code system A is selectable.</li> <li>2) Modal data other than the G function.</li> </ul>						
	Modal data other than CTL1 (kinds of data)						
	the ( (address in	G function the part prog	ram)	For m	nachining ystem	For turning system	Field from which to fetch data
		D			24	_	
		E			25	24	
		H			26	25	
		 			27	26	
		N			20	28	DATA+0 to DATA+5
					30	29	DAINTO LO DAINTO
	s				31	30	
		Т			32	31	
		F			33	32	
	Second a	uxiliary function	on		34	33	

# C.3.15 **Reading Diagnosis** Data (Low-speed Response)

[Description]

The data on the diagnostic data screen of the CNC is read.



## NOTE

- 1 The valid range of diagnosis numbers is 0 to 103 and 200 to 303. (FS15B)
  - For the FS15*i*, the valid range of numbers is 1000 and above. (FS15iA)
- 2 Only integer values can be read as diagnosis data. (FS15*i*A)
- 3 For the FS15*i*, axis data can be read by specifying an axis number for the data attribute.

# C.3.16 [Description] Reading A/D 1. The load current for an axis controlled by the CNC is converted to adigital value and the digital value is read. Conversion Data for the Feed Motor 2. The analog data input to the CNC is converted to a digital value by the A/D converter and the digital value is read.

Set data		Readdata		_
0	Function code 34	0	Function code 34	
2	Completion code —	2	Completion code	
4	Data length	4	Data length 2	
6	Datanumber	6	Datanumber	
8	Axis number	8	Axis number	•
10	Data area (2 byte)	10	A/D conversion data (2 byte)	
12		12		]

An analog voltage ranging from -10V to +10V is input to the A/D converter of the NC. The voltage is converted to a digital value ranging form 0 to +255 and transferred by the window function to the PMC. This value is called the A/D conversion data.

The digital value is proportional to the analog voltage: 0 corresponds to -10V, 128 corresponds to 0V, and 255 corresponds to +10V.

Type of analog voltage input	Data number	Axis number
General–purpose analog input	0	1 2 3 4 5 6
Analog input of a voltage caluculated from the load current for the axis con- trolled by the NC (AC servo motor only)	200	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Method of calculation of the load current of controlled axis from the read A/D conversion data is as follows.

**Examples** 

a) In the case of peak current [Ao–p] of load current is calculated.

LOAD CURRENT[Ao-p] =  $\frac{(\text{READ DATA}) - 128}{(\text{COEFFICIENT})}$  [Ao-p]

b) In the case of ratings currents [Arms] of load current are calculated.

LOAD CURRENT[Arms] =  $\frac{(\text{READ DATA}) - 128}{(\text{COEFFICIENT}) \times \sqrt{2}}$  [Arms]

c) In the case of percent (rate) is calculated.

APPENDIX

rate of load[%] =  $\frac{\text{LOAD CURRENT[Ao-p]}}{\text{PEAK CURRENT OF SERVO MOTOR}} \times 100[\%]$ 

COEFFICIENT: It decides depending on the capacity of the amplifier to be used.

PEAK CURRENT OF SERVO MOTOR : It dicides with the servo motor.

When the AC motor model "30s" is used and the read A/D conversion data is 150, method of calculating each load current.

The following is understood from manual of the servo.

AC motor model	Ratings currents(Arms)
30S	16

Moreover, the amplifier of 80A is used for the motor of 30S.

The coefficient is calculated.

The coefficient is a value by which the peak current of amplifier is converted by 128.

$$COEFFICIENT = \frac{128}{PEAK CURRENT VALUE OF AMPLIFIER} = \frac{128}{80} = 1.6$$

The peak current of the servo motor is calculated.

PEAK CURRENT[Ao-p] = (ratings currents)  $\times \sqrt{2}$ = 16  $\times \sqrt{2}$  = 22.62742

 $\doteq 23$  [Ao-p]

Since the rade A/D conversion data is 150, the peak current, the ratings currents and the rate of the load can be calculated.

a) Peak current[Ao-p] of load current

LOAD CURRENT[Ao-p] = 
$$\frac{(\text{READ DATA}) - 128}{(\text{COEFFICIENT})} = \frac{150 - 128}{1.6}$$

b) Ratings currents[Arms] of load current

RATINGS CURRENTS[Arms] =  $\frac{(\text{READ DATA}) - 128}{(\text{COEFFICIENT}) \times \sqrt{2}} = \frac{150 - 128}{1.6 \times \sqrt{2}}$ = 9.72 [Arms]

c) PERCENT(RATE)

RATE OF LOAD[%] = 
$$\frac{\text{LOAD CURRENT[Ao-p]}}{\text{PEAK CURRENT OF SERVO MOTOR}} \times 100$$
$$= \frac{13.75}{23} \times 100 = 59.8 \text{ [\%]}$$

## [Hardware]

When general analog input data is read, one of the following hardware items is required:

- 1 Sub-CPU board
- 2 Analog I/O module on the additional axis board

For details of the relationship between input numbers and connectors, refer to the connection manual. When an NC controlled axis load current is read, the hardware described above is not required.

### [NC parameters]

When this function is used, the NC parameters listed below need to be set. For details, refer to the parameter descriptions.

- 1 Bit 6 of parameter No. 1810 = 0 (A/D conversion is performed.)
- 2 Bit 0 of parameter No. 1811 = 1 (A/D conversion data is output in high–speed mode.)

# C.3.17 Reading the Tool Life Management Data (Tool Group Number) (Low–speed Response)

## [Description]

The number of the tool group in which the tool number is cataloged is read.



### NOTE

The data can be read only when the tool life management data function is provided.

# C.3.18 Reading the Tool Life Management Data (Number of Tool Groups) (Low–speed Response)

# [Description]

The number of tool groups contained in the tool life management data is read.

Set data		Read data	
0	Function code 39	0	Function code 39
2	Completion code —	2	Completion code
4	Data length —	4	Data length 4
6	Data number 0	6	Datanumber
8	Data attribute 0	8	Data attribute
10	Data area (4 byte)	10	Number of tool groups (4 byte)
14		14	

# C.3.19

Reading Tool Life Management Data (Number of Tools) (Low–speed Response)

[Description]

The number of tools cataloged in the specified tool group is read.



# C.3.20 Reading Tool Life Management Data (Tool Life) (Low–speed response)

[Description]

The tool life of the specified tool group is read.



# C.3.21

Reading Tool Life Management Data (Tool Life Counter) (Low-speed Response)

[Description]

The tool life counter of the specified tool group is read.



# C.3.22 Reading Tool Life Management Data (Tool Life Counter Type) (Low–speed Response)

[Description]

The tool life counter type of the specified tool group is read.



# C.3.23 Reading Tool Life Management Data (Tool Length Compensation No.1) (Low-speed Response)

## [Description]

A tool length compensation number is read according to the specified tool group number and tool number.



If nothing is specified after the H code, the NC transfers 255 (FFH).

# C.3.24 Reading Tool Life Management Data (Tool Length Compensation No.2) (Low–speed response)

# [Description]

A tool length compensation number is read according to the specified tool group number and tool order number.



APPENDIX

If nothing is specified after the H code, the NC transfers 255 (FFH).

# C.3.25 Reading Tool Life Management Data (Cutter Compensation No.1) (Low-speed Response)

# [Description]

A cutter compensation number is read according to the specified tool group number and tool number.



If nothing is specified after the D code, the NC transfers 255 (FFH).

# C.3.26 Reading Tool Life Management Data (Cutter Compensation No.2) (Low–speed Response)

# [Description]

A cutter compensation number is read according to the specified tool group number and tool order number.



APPENDIX

If nothing is specified after the D code, the NC transfers 255 (FFH).

## C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

# C.3.27 Reading Tool Life Management Data (Tool Information 1) (Low-speed Response)

## [Description]

The tool information (status) is read according to the specified tool group number and tool number.



# C.3.28 Reading Tool Life Management Data (Tool Information 2) (Low-speed Response)

## [Description]

The tool information (status) is read according to the specified tool group number and tool order number.



# C.3.29 Reading Tool Life Management Data

(Tool Number) (Low-speed Response)

## [Description]

A tool number is read according to the specified tool group number and tool order number.



APPENDIX

#### C.3.30 [Description] **Reading Clock Data** The current data (year, month, day) and current time (hours, minutes, (Low-speed Response) seconds) can be read from the clock built into the CNC. Set data Read data 0 0 Function code Function code 151 151 2 2 Completion code Completion code 4 4 Data length Data length 6 0 : Current data 6 6 Data format Data format - 1 : Current time 8 8 Data attribute Data attribute 0 10 10 Data area Clock data (6 byte) (6 byte) 16 16 (Note) Format of clock data The data is binary. Current date DATA+0 Years (Example: 1992) +2 Months (Example: 12) +4 Days (Example: 16) +6 Current time DATA+0 Hours (Example: 23) +2 Minutes (Example: 59) +4 Seconds (Example: 59) +6

....

— 1288 —

# C.3.31 Reading the Relative Position on a Controlled Axis

## [Description]

The relative position (relative coordinates) on a feed axis controlled by the CNC is read.



1) The unit of the read value is determined as follows:

- (1) For the machining center system or when the radius is specified for the axis of the lathe systm. The data shows double of the present position with the least input increment as a unit.
- (2) When the diameter is specified for the lathe system

The data shows the present position with the least input increment.

(3) When the input unit is multiplied by 10

The data shows twenty-times the present position (radius programming) or ten-times the present position (diameter programming) with the last command increment as a unit.

2) The present position of a moving axis can be read whenever the function instruction is executed.

# C.3.32 Reading the Remaining Travel

[Description]

The remaining traveling distance on a feed axis controlled by the CNC is read.



APPENDIX

# C.3.33 Reading an Estimate Disturbance Torque Data

## [Description]

- 1) The load torques except a necessary torque for acceleration/ deceleration of the torques of the servo axis are read.
- 2) The load torques except a necessary torque for acceleration/ deceleration of the torques of the serial spindle axis are read.



Kind of data	Data specification
Estimate disturbance torque data of a digital servo	Please refer to "FANUC AC SERVO AMPLIFIER AMINTENANCE MANUAL (B–65005E)" for correspondence of the load torque with the value of the read data.
Estimate disteurbance torque data of a serial spindle	Please refer to "FANUC AC SPINDLE SERVO UNIT (SERIAL INTERFACE) MAINTE- NANCE MANUAL (B–65045E)" for correspondence of the load torque with the value of the read data. The load torque of the spindle is understood from the undermentioned calculation type.
	Load torque = $\frac{\text{The read data}}{16384} \times \text{Maximum output torque of spindle}$

* For an explanation of the CNC parameters, refer to the "CNC Parameter Description."

# C.3.34 **Reading the Machining** Time (Low-speed Response)

[Description]

The machining time currently specified for a program is read.



APPENDIX

# C.3.35 Reading the Load [Description] Current (A/D The load current Conversion Data) for the Spindle Motor Current (A/D Corrent (A/D Co

The load current for the spindle (spindle motor) is converted to a digital value and the digital value is read. (See Section 3.16, "Reading the Load Current (A/D Conversion Data) for the Feed Motor.")



## [Hardware]

When general analog input data is read, one of the following hardware items is required:

- 1 Sub-CPU board
- 2 Analog I/O module on the additional axis board

With a serial spindle, however, the hardware described above is not required if CNC software of mass production version IV or later is used. For the relationship between input numbers and connectors, refer to the connection manual.

## [NC parameters]

When this function is used, the NC parameters listed below must be set. For details, refer to the parameter descriptions.

- 1 Bit 6 of parameter No. 1810 = 0 (A/D conversion is performed.)
- 2 Bit 0 of parameter No. 1811 = 1 (A/D conversion data is output in high–speed mode.)

## C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

# C.3.36 Reading the Tool Offset Data According to the Specified Tool Number

[Description]

The tool number is spedified and the tool offset data is read.



Kind of the data to be read	The data form CTL1	Tool number CTL2
Tool number	01	Tool display number
Pot number	10	Tool No.
Pot number	11	Tool display number
Tool length compensation value	20	Tool No.
Tool length compensation value	21	Tool display number
Cutter compensation value	30	Tool No.
Cutter compensation value	31	Tool display number

## NOTE

Please use the bit 4 of NC parameter as 1. When the completion code "5" is returned, change the format of the window in the SETTING Screen. (REFERENCE:chapter II 4.4 SETTING Screen) APPENDIX

# C.3.37 Reading Tool Life Management Data (Tool Group Numbers) (Low–speed Type)

# [Explanation of data]

The tool group number in which a tool number is registered is read. Note that the tool number area is four bytes in length so that a tool number of up to eight digits can be specified.

## [Input data structure]

Settings		Afterreading	
0	Function code 200	0	Function code 200
+2	Completion code —	+2	Completion code
+4	Data length —	+4	Data length 4
+6	Data format 0	+6	Data format
+8		+8	
+10	Toolnumber	+10	Toolnumber
+12	Data area	+12	Tool group number
+14	(4 byte)	+14	(4 byte)
+16		+16	

C.3.38	[Explanation of data]
Reading Tool Life	The tool length compensation number corresponding to the specified tool
Management Data	group number and tool number is read.
(Tool Length	Note that the tool number area is four bytes in length so that a tool number
Compensation Number	of up to eight digits can be specified.
1) (Low–speed Type)	

## [Input data structure]



C.3.39	[Explanation of data]
Reading Tool Life Management Data	The cutter compensation number corresponding to the specified tool group number and tool number is read.
(Cutter Compensation Number 1)	Note that the tool number area is four bytes in length so that a tool number of up to eight digits can be specified.
(Low-speed Type)	

APPENDIX

# [Input data structure]

Settings		Afterreading		
0	Function code 228	0	Function code 228	
+2	Completion code	+2	Completion code	
+4	Data length —	+4	Data length 4	
+6	Tool group number	+6	Tool group number	
+8		+8		
+10	Toolnumber	+10	Toolnumber	
+12	Data area	+12	Cuttercompensation	
+14	(4 byte)	+14	number (4 byte)	
+16		+16		

C.3.40	[Explanation of data]
Reading Tool Life	The tool information (state) corresponding to the specified tool group
Management Data	number and tool number is read.
(Tool Information 1)	Note that the tool number area is four bytes in length so that a tool number
(Low-speed Type)	of up to eight digits can be specified.

[Input data structure]



Explanation of tool information

- 1 : The tool is registered.
- 2 : The tool has expired.
- 3 : The tool has been skipped.

# C.3.41 Reading Real [Explanation of data] Parameters Real parameters are read from the CNC. (Low-speed Type) [Input data structure]



## NOTE

An integer or bit parameter cannot be read using function code 321.

Read a parameter using:

- Real parameter: Function code 321
- Integer or bit parameter: Function code 17

Example) The value of a read-out parameter is as follows:

(Value of a read–out parameter) =

(value of the parameter on the NC)  $\times$ 10^(specified decimal point position)

Parameter value	Value on the NC	Decimal point position
1 12 123 1234 1234	1.123	0 1 2 3 4

# C.3.42 Reading the actual Machine Position (Machine Coordinates) of Controlled Axes.

## [Description]

The machine position (machine coordinates) on a feed axis controlled by the CNC can be read.

The machine position value includes the servo delay and the acceleration / deceleration delay.

So, this value might be not corresponding to the machine position value on the position screen in the CNC.(This screen can be displayed by pressing the function button "POS".)

## [Structure]



The unit of the output value is determined as follows:

- The case of radius programming axis. The output data shows twice value of the present position with the least input increment.
- (2) The case of diameter programming axis. The output data shows a value of the present position with the least input increment.
- (3) The case that the least input increment is set to ten-times value as large as the least command increment. The output data shows twenty-times value of the present position (radius programming) or ten-times value of the present position (diameter programming) with the least command increment.

## NOTE

- 1 This function is provided by the FS15*i*A (PMC–NB6) only.
- 2 To use this function, please set NC parameter 1013#7 to 1. This parameter should be set only for necessary axes because this parameter increases the load of CNC CPU.

# C.3.43 Reading Fine Torque Sensing Data (Statistical Calculation Results).

# [Description]

This function reads the statistical calculation results (average value, maximum value and distribution) in the fine torque sensing function.

[Structure]



**NOTE** This function is provided by the FS15*i*A (PMC–NB6) only.

#### C.3.44 (1) Store Counter **Reading Fine Torque** [Description] Sensing Data (Store This function reads the number of stored torque data items. (Store Data) Counter)

[Structure]



NOTE This function is provided by the FS15*i*A (PMC–NB6) only. (2) Stored torque data (newest data)

[Description]

This function reads the newest stored data among stored torque data [Structure]



## NOTE

- 1 When no data is stored, "0" is read as data length and no data is read. The function completes normally.
- 2 When data attribute is set to "11", the sample data whose number is same as the last number of newest data is read.
  - Ex.) In case that 10000 sample data items (the data number from 0 to 9999) and newest data items 5000 (the data number from 0 to 4999) are stored, newest data which number is 4999 is read when data attribute is set to "10". And sample data which number is 4999 is read when data attribute is set to "11".
- 3 When data attribute is set to "11" and no sample data whose number is same as the last number of newest data is stored, "0" is read as data length and no data is read. The function completes normally.
  - Ex.) In case that 5000 sample data items (the data number from 0 to 4999) and newest data items 10000 (the data number from 0 to 9999) are stored, no data is read, and the function completes normally when data attribute is set to "11".

(3) Stored torque data (any data)

## [Description]

This function reads the newest stored data among stored torque data [Structure]



NOTE					
1. The valid area of data number m is as follows: $0 < m < (524288 \times 1/6 \times 1/6) = 1$					
$0 \le 111 \le (524200 \times 1/8 \times 1/b) = 1$					
for $a = 1.2$ : the number of the axes for Fine Torque Sensing is 2					
a = 2. The number of the axes for Fine Torque Sensing is 2 or 4.					
4. The humber of the axes for Fine forque Sensing is 5 of 4					
$b = \frac{1}{2}$					
2 : sample data saving function is enabled					
The valid area of the number of data n is as follows:					
1 < n < 120					
2 When data number m which is in the valid range exceeds the number of actually stored data					
items "0" is read as data length and no data is read. The function completes normally					
Ex.) In case that the number of target axes is 2 and the sample data store function is enabled					
the data numbers are valid in the range between 0 and 13107. However, if an attempt is					
made to read data from data number m "131020" when the number of actually stored data					
items is 131000 (the data number from 0 to 130999) "0" is read as data length and no data					
is read. The function completes normally.					
(Example 1 in figure below)					
3. When data number m is within the number of actually stored data items, and the number					
"m+n-1" exceeds the number of actually stored data items, data of the stored data items is read.					
The function completes normally.					
In this case, double of the number of read data items is read as data length.					
Ex.) If an attempt is made to read number of data items n "120" from data number m "130900"					
under the same conditions as in the example above, the data number from 130900 to					
130999 is read and "200" is read as data length.					
(Example 2 in figure below)					
Also if an attempt is made to read number of data items n "120" from data number m					
130990 under the same conditions as in the example above, the data number from 130990					
to 130999 is read, and "20"is read as data length.					
(Example 3 in figure below)					
Data number vallid range Range of actually stored data					




## NOTE

This function is provided by the FS15*i*A (PMC–NB6) only.

#### [Alarm type]

Following is a correspondence table of the alarm type and the alarm type number.

Alarm type number	Alarm type	Alarm type number	Alarm type	Alarm type number	Alarm type
0	BG alarm	10	SR alarm	20	SP alarm
1	PS alarm	11			
2	OH alarm	12	SV alarm		
3		13	IO alarm		
4	SN alarm	14	PW alarm		
5	SW alarm	15	SY alarm		
6	OT alarm	16			
7	PC alarm	17			
8	EX alarm	18	—		
9		19	MC alarm		

#### B-61863E/14





## [Example]

NC alarm : OT alarm NO.6 (An alarm is issued for the 1st axis.) : SV alarm NO.10,11 (Alarms are issued for the 3rd axis.) : Ex alarm NO.5 (Not axis type)

When these alarms are issued, the read data is as follows.

1. In case that the SV alarm is read.

Set data			Read da	ata	
0	Function code	330	0	Function code	330
2	Completion code	_	2	Completion code	0
4	Data length	_	4	Datalength	18
6	Number of alarm	10	6	Number of alarm	10
8	Alarm type	12	8	Alarm type	12
10			10	Number of read alarms	2
			12	Axis information 1	4
			16	Alarm type 1	12
			18	Alarm number 1	10
			20	Axis information 2	4
			24	Alarm type 2	12
			26	Alarm number 2	11
			28		

## C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

#### APPENDIX

Set data			Read da	ita	
0	Function code	330	0	Function code	330
2	Completion code	_	2	Completion code	0
4	Datalength	_	4	Datalength	34
6	Number of alarm	10	6	Number of alarm	10
8	Alarm type	-1	8	Alarm type	-1
10			10	Number of read alarms	4
			12	Axis information 1	1
			16	Alarm type 1	6
			18	Alarm number 1	6
			20	Axis information 2	4
			24	Alarm type 2	12
			26	Alarm number 2	10
			28	Axis information 3	4
			32	Alarm type 3	12
			34	Alarm number 3	11
			36	Axis information 4	0
			40	Alarm type 4	8
			42	Alarm number 4	5
			44		

## 2. In case that all alarms are read.

# C.4 FORMAT AND DETAILS OF THE CONTROL DATA OF THE WINDW FUNCTIONAL INSTRUCTION

(1) See the description of the window function. The data item marked with a dash (–) in the description of the data structure need not be entered.

When output, the data item has no meaning.

APPENDIX

- (2) The length of all data blocks and data items is represented in bytes.
- (3) The output data becomes valid only when the instruction terminates normally.

Completion code	Description
-10	The window instruction is being processed. Hold ACT until W1 is set to 1.
0	The instruction terminated normally.
1	An error occurred. The corresponding function number is not found.
2	An error occurred. Possible causes include the following: Wrong data is found in the CTL area. The NC does not have the corresponding function.
3	An error occurred. The specified axis is not provided.
5	An error occurred. it is a mistake of data form. Though the function supports only a new format, a old format is specified.

#### C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

APPENDIX

#### C.4.1 [Description] Writing a Tool Offset The data is directly written into the tool offset value (tool compensation) Data area of the CNC. Set data 0 Function code 14 2 Completion code 4 Data length 4 6 Offset number 8 Offset format 10 Tool offset value (4 byte) 14 (Note 1) Offset format M system (machining center system) T system (lathe system) Format Offset number Data type Format Offset number Data type (CTL+2, 3) (CTL+2, 3) Tool compensation A Tool compensation A 1 Offset number Compensation 1 Offset number Compensation along the X-axis Tool compensation B Compensation along the 2 Offset number Geometry Offset number Z–axis 1 Tool-tip radius compensation compensation 3 Offset number 1 Compensation along the Offset number Wear Offset number 4 compensation +1000 Y–axis Compensation related to the posi-5 Offset number tion of the virtual tool Offset number Tool compensation C Tool length Geometry Offset number 1 Tool compensation B compensation Geometry compensation Offset number Wear 1 Compensation along the X-axis 1 2 Offset number compensation +1000 Compensation along the Z-axis Offsetnumber Tool-tip radius compensation 3 Offset number Cutter Compensation along the Y-axis 4 Offset number Geometry Offset number 2 Wearcompensation compensation Compensation along the X-axis 1 Offset number 2 Wear Offset number +1000 compensation +1000 2 Offset number Compensation along the Z-axis +1000 Tool-tip radius compensation 3 Offset number +1000 Offset number Compensation along the Y-axis 4 +1000Compensation related to the 5 Offset number position of the virtual tool

## 

# C.4.2 Writing a Parameter (Setting Data)

[Description]

The data is written into the parameter area of the CNC.

The parameters of the CNC are classified into four types according to the smallest unit that has a meaning. Bit parameter: Each bit has a meaning. Byte parameter: Each byte has a meaning. Word parameter: Each set of two bytes has a meaning. Double word parameter: Each set of four bytes has a meaning.

Each bit of a bit parameter cannot be written individually. The eight bits (one byte) of the parameter must be written at a time. To change a bit of a bit parameter, read the entire parameter, change the desired bit, then write the entire parameter.



## NOTE

For the PMC–NB6, real–type parameters cannot be written, causing a CTL ERROR to occur.

#### C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

# C.4.3 Writing a Custom Macro Variable

## [Description]

The data is written into the custom macro variable area of the CNC.



## Examples

(Value written in the NC) =  $\frac{\text{(value of custom macro variable)}}{10^{\text{(Positon of decimal point)}}}$ 

Value in the NC	Custom macro variable value	Position of decimal point
1234. 000 123. 400 12. 340 1. 124 0. 1234	1234	0 1 2 3 4

# C.4.4 Writing a Data on the Program Check Screen

[Description]

The data to be displayed on the program check screen of the CNC is rewritten.



Data type	Data I	ength	Data	Attrib	
	15B	15 <i>i</i>	type	ute	
M code which is being executed (1 to 5)	2	2	1 to 5	0	
Spindle speed range	2	2	100	0	
Spindle tool number	2	2, 4	200	0	
Number of the tool to be used next	2	2, 4	201	0	

## NOTE

- 1 As much program check screen data as the amount specified with the data length can be written.
- 2 When the 15*i* is used, the spindle tool number and the next machining tool number can each be written into a 2– or 4–byte area.

#### C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

APPENDIX

# C.4.5 Writing the Torque Limit Override

[Description]

The torque limit override of the specified feed axis is rewritten.

Set data			
0	Function code 152		
2	Completion code		
4	Data length 2		
6	Data type 0		
8			
	Axis number	Value	
10	Torque limit override	Un–signed binary <unit: %=""> The values from 0 to 255 correspond to 0% to 100%.</unit:>	
12			

## [Example]

If the torque limit override is 50%, please set to 128.

## NOTE

This window is valid for parameters set on the CNC. Parameter 1802#4 { 0: Fixed to override 100% 1: This window enabled.

# C.4.6 Writing the Tool Life Management Data (Tool Group Number)

## [Description]

The tool number and the tool life value are written into the specified tool group.

Ŭ	Function code 163		
2	Completion code -	/ Data type / Number of	Register the tool group.
4	Data length 4		transmission data. 7 6 5 4 3 2 1 0
6	Data number 0		DATA+0 DATA+1 DATA+1
8	Tool No.		DATA+2 Tool life value
10	Tool group No.		Group number 1 to 512 Tool life value 1 to 9999 (Specified number of time) 1 to 4300 (Specified time)
12	— (4 byte) — Tool life value		1001 lite counter type ( DATA+1 BIT 7) 0: Number of time 1: Time (minute)

# C.4.7Writing the Tool LifeManagement Data(Tool Life)The data is written into the tool life value area of the specified tool group.



## C.4.8 Writing the Tool Life Management Data (Tool Life Counter) [Description] The data is written into the tool life counter area of the specified tool group. Set data



# C.4.9

Writing the Tool Life Management Data (Tool Life Counter Type)

## [Description]

The data is written into the tool life counter type area of the specified tool group.



# C.4.10 Writing the Tool Life Management Data (Tool Length Compensation Number 1)

## [Description]

The data is written into the tool length compensation number area specified by the tool group number and tool number.



## C.4.11 Writing the Tool Life Management Data (Tool Length Compensation Number 2)

## [Description]

The data is written into the tool length compensation number area specified by the tool group number and tool order number.



## C.4.12 Writing the Tool Life Management Data (Cutter Compensation Number 1)

#### [Description]

The data is written into the cutter compensation number area specified by the tool group number and tool number.



## C.4.13 Writing the Tool Life Management Data (Cutter Compensation Number 2)

## [Description]

The data is written into the cutter compensation number area specified by the tool group number and tool order number.



# C.4.14Writing the Tool Life<br/>Management Data<br/>(Tool Information 1)[Description]The data is written into the tool information (status) area specified by the<br/>tool group number and tool number.



## C.4.15 Writing the Tool Life [Description] Management Data (Tool Information 2) The data is written into the tool information (status) area specified by the tool group number and tool order number.



#### C.4.16 Writing the Tool Life Management Data (Tool Number) (Description] A tool number is written into (added to) the area specified by the tool group number and tool order number. Set data 0 Function code



#### 

## C.4.17 Writing the Tool Offset Data According to the Specified Tool Number

[Description]

The tool number is specified and the tool offset data is written.



Kind of the data to be written	The data form CTL1	Tool number CTL2
Change of Tool number	00	Tool No.
Change of Tool number	01	Tool display number
Pot number	10	Tool No.
Pot number	11	Tool display number
Tool length compensation value	20	Tool No.
Tool length compensation value	21	Tool display number
Cutter compensation value	30	Tool No.
Cutter compensation value	31	Tool display number
Addition of Tool number	40	Tool No.
Addition of Tool number	41	Tool display number

## NOTE

(Only Series 15B)

Please use the bit 4 of NC parameter as 1.

When the completion code "5" is returned, change the format of the window in the SETTING Screen.

(REFERENCE : chapter II 4.4 SETTING Screen)

#### C. WINDOW FUNCTION DESCRIPTION (PMC–NB/NB2/NB6)

APPENDIX

## C.4.18 Writing the Superposition Move Command

### [Description]

After the axes for manual handle feed are selected in the manual handle feed mode, the traveling distances (number of pulses) corresponding to three manual pulse generators are written. The set value ranges from -256 to +256.

The specified number of pulses is assumed to be the number of pulses entered from the manual pulse generator. The speed is calculated as follows: (specified number of pulses)  $\times$  (magnification)  $\times$  62.5 (pulses/second) The data in parameters 1413 and 1414 of the CNC is valid for this function.

(1) For three axes



The data for using the fourth manual pulse generator is as shown on the next page.





# C.4.19 Writing the Feedrate

## [Description]

Feedrate writing can be specified only in the feed-per-minute mode. The velocity command is specified with  $F \times 10^{-d}$ . A flag is provided to validate either the command of the PMC or the feedrate of the CNC. After the flag is set, the velocity command specified in the NC is invalidated. To validate the feedrate of the NC, set the flag to 0.



#### NOTE

With the FS15*i*–A (PMC–NB6), this function is of the low–speed type. Therefore, this function need not be interlocked with other windows.

# C.4.20[Explanation of data]Writing Tool Life<br/>Management Data<br/>(Tool Group Numbers)A tool number and a tool life value are written to a specified tool number.<br/>Note that the tool number area is four bytes in length so that a tool number<br/>of up to eight digits can be specified.





C.4.21 Writing Tool Life Management Data (Tool Length Compensation	[Explanation of data] Data is written to the tool length compensation number specified by a tool group number and a tool number. Note that the tool number area is four bytes in length so that a tool number of up to eight digits can be specified
Number 1)	of up to eight digits can be specified.

## [Input data structure]



## NOTE

This function is provided by the FS15*i*A (PMC–NB6) only.

C.4.22	[Explanation of data]
Writing Tool Life Management Data	Data is written to the cutter compensation number specified by a tool group number and a tool number.
(Cutter Compensation Number 1)	Note that the tool number area is four bytes in length so that a tool number of up to eight digits can be specified.

## [Input data structure]



## NOTE

This function is provided by the FS15iA (PMC-NB6) only.

C.4.23	[Explanation of data]
Writing Tool Life Management Data	Data is written to the tool information (state) specified by a tool group number and a tool number.
(Tool Information 1)	Note that the tool number area is four bytes in length so that a tool number of up to eight digits can be specified.

#### [Input data structure]



Explanation of tool information

- 1 : The tool is registered.
- 2 : The tool has expired.
- 3 : The tool has been skipped.

## NOTE

This function is provided by the FS15iA (PMC–NB6) only.

C.4.24 Writing Real Parameters (Low–speed Type)	[Explanation of data] Real parameters are written to the CNC. [Input data structure]					
	Settings 0 +2 +4 +6 +8 +10 +12 +14 +16 +18	Function code 323 Completion code — Data length 6 Parameternumber Data attribute Decimal point position Data area (4 byte)	0: no axis 1 to n: axis specified			

## NOTE

Integer parameters cannot be written.

## Example)

(Value to be set on the NC) =  $\frac{\text{(Parameter variable value)}}{10^{\text{(specified decimal point position)}}}$ 

Value to be set on the NC	Custom macro variable value	Decimal point position
1234.000 123.400 12.640 1.234 0.1234	1234	0 1 2 3 4



# D.1 OUTLINE

The following function is added to PMC–CNC window function for FS16–LA.

- (1) Transferring a processing condition file in non-volatile memory to data area in CNC memory, and vice versa.
- (2) Reading a comment command in a part program.
- (3) Reading data commanded to laser oscillator

The following functions are added to the PMC–CNC window function on the FS16*i*–LA:

- Reading of comments Comments specified within programs can be read.
- (2) Laser command value data and laser setting data can be read and written, and command value data to the laser oscillator can be read.

## CAUTION

Data transfer of machining condition files between data areas is not possible by ladder diagrams on the FS16*i*–LA.

# D.2 FUNCTION

# D.2.1

## Transfer Between Data Area and Non–volatile Memory

(1) Transfer from data area to non-volatile memory. (% low-speed type)

#### [Contents of data]

The data can be transferred from the data area in CNC to PMC non-volatile memory by PMC-RC application.

Setting the original data set or group in data attribute M.

Setting the written address of non–volatile memory for the offset address from the top address in address N.

Setting the total byte No. of written data set in address L.

Setting the forward structure of data set in data.

And the data can be transferred set or group from data area to non–volatile memory.

[Structure of input data]



(2) Transfer from non-volatile memory to data area (% low-speed type)

#### [Contents of data]

The data can be transferred from the processing condition file registered in non–volatile memory to the data area in CNC by PMC–RC application.

Setting the forward data set or group in data attribute M.

Setting the original read address for the offset address from non–volatile memory top address in address N.

Setting the total byte No. of original data set in address L.

Setting the original structure of set in data.

And the data of set or group can be transferred from non–volatile memory to data area.



[Structure of input data]

## (3) Data structure of data area

	D	•	1 .	
(9)	Process	nn	data	COT
(a)	1100033	SHI2	uata	SUL
(/				

Address	Data item	Byte No.
0	Feed-rate	4
4	Peak power	2
6	Pulse frequency	2
8	Pulse duty	2
10	Assist gas pres.	2
12	Assist gas select	2
14	Assist gas settling time	2
16	Reference displacement	2
18	Offset amount	4
22	Edge process select	2
24	Start-up process select	2

## (b) Piercing data set

Address	Data item	Byte No.
0	Peak power	2
2	Initial frequency	2
4	Initial duty	2
6	Frequency increment	2
8	Duty increment	2
10	Step time	2
12	Step No.	2
14	Piercing time	4
18	Assist gas pres.	2
20	Assist gas select	2
22	Assist gas settling time	2
24	Reference displacement	2

## (c) Edge processing data set

Address	Data item	Byte No.
0	Judge angle	2
2	Peak power	2
4	Pulse frequency	2
6	Pulse duty	2
8	Piercing time	4
12	Assist gas pres.	2
14	Assist gas select	2
16	Return distance	4
20	Return feed rate	2
22	Return frequency	2
24	Return duty	2

## NOTE

Example of data set

The address in Data is set as follows, for example, in case of the following data structure of processing condition file in non-volatile memory.

Data struc	cture of data area		The examp the process non–volatil	le of data structure sing condition file in ememory	for
Address	5 Data		Address	Data	
0	Feed-rate		0	Feed-rate	
4	Peak power 🔫		4	Peak power	
6	Pulse frequency		6	Pulse frequency	
8	Pulse duty		8	Pulse duty	
10	Assist gas pres.		10	Focus distance	
12	Assist gas select		12	Assist gas pres.	
14	Assist gas time		14	Assist gas select	
16	Ref. displacement		16	Assist gas time	
18	Offset amount		18	Ref. displacement	t
22	Edge select		20	Offset amount	
24	Start-up select	_	24	Pulse type	
I			26	Edge select	
			28	Start-up select	
	•				
	Data item in data area	Address	Da	ta setting value	
F P P A A	eed–rate leak power lulse frequency lulse duty lssist gas pres. lssist gas select	Top address +10 +12 +14 +16 +18 +20		0 4 6 8 12 14	
A R O E S	ssist gas time tef. displacement Offset amount dge select start-up select	+22 +24 +26 +28 +30		16 18 20 26 28	

# D.2.2 The data in the parentheses is written in the comment area, if the following M-code is commanded in a part program. This comment can be read from PMC.

#### [Contents of data]

Contents of the data can be read for ASCII code.

Less than 24 characters,

including alphabet, numeral, decimal-point and +/-

#### [Structure of input data]



#### NOTE

- 1 The comment is over-written if the next comment is input.
- 2 M-code number for reading of the comment is set to parameter number 15350. Setting value is 0 to 999.

#### D.2.3 (1) Reading the laser command data and laser setting data (*high-speed type) **Reading and Writing** the Laser Command [Contents of the data] **Data and Laser Setting** Data

The laser command data and laser setting data for CNC can be read by PMC-RC application. The data are separated to groups and can be read by the group.



## [Structure of input data]

(2) Writing the laser command data and laser setting data (*low-speed type)

#### [Contents of the data]

The data can be written to the laser command data for CNC by PMC–RC application. The data are separated to groups and can be written by the group.

#### [Structure of input data]



Group No.	Address Top add. +	Byte No. for everv item	Data length Byte No.	Item
0	10 12	2 2	4	Power control duty const Power control minimum duty
1	10 12 14 16	2 2 2 4	10	Power monitor(Read only)Power offset(Read only)Actual power(Read only)Actual feedrate(Read only)
2	10	2	2	Power input offset coe.
3	10 12	2 2	4	Assist gas select Assist gas flow select
4	10 12 14 16 18 20 22 24 26 28 30 32 34 36 38	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	30	Assist gas flow–1 pre–time pre–pres wrk–pres aft–time aft–pres Assist gas flow–2 pre–time pre–pres wrk–pres aft–time aft–pres Assist gas flow–3 pre–time pre–pres wrk–pres aft–time aft–pres
5	10 12 14	2 2 2	6	Processing peak power Processing pulse frequency Processing pulse duty
6	0 12 14 16	2 2 2 4	10	Piercing peak power Piercing pulse frequency Piercing pulse duty Piercing time
7	10 14 16 18 20 22 24 26 28	4 2 2 2 2 2 2 2 2 4	22	Feed-rate command Peak power command Pulse frequency command Pulse duty command Assist gas select command Assist gas settling time Assist gas pressure Reference displacement Offset amount
10	10	2	2	Ref. displacement command

(3) The data structure of the laser command data and laser setting data



#### — 1340 —

# E.1 READING THE WIRE DIAMETER OFFSET

[Description]

The wire diameter offset value recorded in the CNC can be read.

[Input data structure]

Top address	(Function) 13				
+ 2	(Completion)				
+ 4	(Data length)				
	L	Offset	Corner-R	Clearance	Condition
+ 6	(Number) N	0–15	16	17	_
+ 8	(Attribute) M	0	0	0	1
+ 10	(Data area) —				

## [Output data structure]

Top address	(Function) 13				
+ 2	(Completion) ?	Offset	Corner-R	Clearance	Condition
+ 4	(Data length) L	4	4	4	8
+ 6	(Number) N	0–15	16	17	_
+ 8	(Attribute) M	0	0	0	1
+ 10	(Data area) D	Offset value	Corner-R value	Clearancevalue	Actual offset value
+ 14			I		Direction 0, 1, 2
+ 16					Offset mode 0, 1
+ 18		]			
#### [Data number]

- 0-15: Reads the Offset value.
  - 16 : Reads the Corner–R value.
  - 17 : Reads the Clearance value.

#### [Data attribute]

- 0 : Reads the Offset value, Corner–R value or Clearance value.
- 1 : Reads the condition.

#### [Contents of data]

a) Unit of Offset, Corner–R, Clearance and actual offset value Metric system input : 10⁻³ [mm]

(In case the incremental system is 1/10, output data unit is  $10^{-4}$  [mm].)

Inch system input :  $10^{-5}$  [inch]

- b) Direction in condition data
  - 0 : Cancel offset (G40)
  - 1 : Wire diameter compensation left (G41)
  - 2: Wire diameter compensation right (G42)
- c) Offset mode in condition data
  - 0 : Offset mode is 0.
  - 1: Offset mode is 1.

- 0 : The data has been read normally.
- 3 : Invalid data is specified as the data number.
- 4 : Invalid data is specified as the data attribute.

#### E.2 WRITING THE WIRE DIAMETER OFFSET (*LOW-SPEED RESPONSE)

[Description]

The wire diameter offset value can be written into the CNC.

[Input data structure]

APPENDIX

Top address	(Function) 14				
+2	(Completion)	Offset	Corner–R	Clearance	Condition
+ 4	(Data length) L	4	4	4	8
+ 6	(Number) N	0–15	16	17	_
+ 8	(Attribute) M	0	0	0	1
+ 10	(Data area) D	Offset value	Corner-R value	Clearancevalue	0, 1
+ 12					_
+ 14			1		

#### [Output data structure]

The address [	
lop address	(Function) 14
+2	(Completion)
+ 4	(Data length) L
+6	(Number) N
+ 8	(Attribute) M
+ 10	(Data area) D

#### [Data number]

- 0–15 : Writes the Offset value.
  - 16 : Writes the Corner–R value.
  - 17 : Writes the Clearance value.

#### [Data attribute]

- 0 : Writes the Offset value, Corner-R value or Clearance value.
- 1 : Writes the condition.

#### [Contents of data]

a) Unit of Offset, Corner–R, Clearance and actual offset value

Offset, Corner–R, Clearance or Actual offset value is signed binary in 4 bytes. A negative value is represented in 2's complement.

- b) Offset mode in condition data
  - 0: Offset mode is 0.
  - 1: Offset mode is 1.

- 0 : The data has been written normally.
- 2 : Invalid data is specified as the data length.
- 3 : Invalid data is specified as the data number.
- 4 : Invalid data is specified as the data attribute.
- 5 : Invalid data is specified as the data value.

#### E.3 READING THE PARAMETER (*LOW-SPEED RESPONSE)

[Description]

Parameter data in the CNC can be read.

APPENDIX

There are four types of parameters in the CNC: Bit parameters having a definite meaning for each bit, byte parameters holding 1–byte data, word parameters holding 2–byte data, and double word parameters holding 4–byte data. Therefore, the length of the read data varies according to the parameter number specified.

Note that bit parameters cannot be read in bit units. The eighth bits (one byte) for a parameter number must be read at a time.

For axis parameters (servo parameters), data for a specific axis can be read, or data for all axes can be read at a time.



#### [Data length]

- L = 1: Reads bit or byte type parameter or pitch error compensation value.
  - 2 : Reads word type parameter.
  - 4 : Reads 2 words type parameter.

When the servo parameter all axis is specified

- $L = 1 \times 8$ : Reads bit or byte type parameter. 2×8: Reads word type parameter.
  - $4 \times 8$ : Reads 2 words type parameter.

#### [Data number]

When the pitch error compensation value

- N = 11000 to 11511: Pitch error compensation value of the  $1^{st}(X)$  axis
- N = 12000 to 12511: Pitch error compensation value of the  $2^{nd}(Y)$  axis
- N = 13000 to 13511: Pitch error compensation value of the  $3^{rd}(U)$  axis
- N = 14000 to 14511: Pitch error compensation value of the  $4^{th}(V)$  axis
- N = 15000 to 15511: Pitch error compensation value of the 5th (Z) axis

- N = 16000 to 16511: Pitch error compensation value of the  $6^{th}(W)$  axis
- $N\,{=}\,17000$  to 17511 : Pitch error compensation value of the  $7^{th}\left(A\right)$  axis
- N = 18000 to 18511 : Pitch error compensation value of the  $8^{th}\left(E\right)$  axis

#### [Data attribute]

When the servo parameter

M = 0: Reads the no axis parameter.

1 to n: Reads the specific axis parameter

-1: Reads the all axes parameter.

(Note: n is the axis number.)

If a parameter other than the servo parameters is read, M is set to 0.

- 0: Parameter data has been read normally.
- 3: Invalid data is specified as the data number.
- 4: Invalid data is specified as the data attribute.
- 6: When the data with the parameter number specified for reading cannot be used if options such as pitch error compensation are not available, those options are not provided.

#### E.4 WRITING THE PARAMETER (*LOW-SPEED RESPONSE)

[Description]

Parameter data in the CNC can be written.

There are four types of parameters in the CNC: Bit parameters having a definite meaning for each bit, byte parameters holding 1–byte data, word parameters holding 2–byte data, and double word parameters holding 4–byte data. Therefore, the length of the written data varies according to the parameter specified.

Note that bit parameters cannot be written in bit unit. The eighth bits (one byte) for the parameter number must be written at a time. This means that when a bit needs to be written, the whole data for the corresponding parameter number shall be read first, then the target bit in the read data shall be written.

For axis parameters (servo parameters), data for a specific axis can be written, or data for all axes can be written at a time.

Some parameters cause a P/S alarm 000 when data is written. (The power must be turned off before continuing operation.)



[Data length]

- L = 1: Reads bit or byte type parameter or pitch error compensation value.
  - 2 : Reads word type parameter.
  - 4 : Reads 2 words type parameter.

When the servo parameter all axis is specified

- $L = 1 \times 8$ : Writes bit or byte type parameter.
  - $2 \times 8$ : Writes word type parameter.
  - $4 \times 8$ : Writes 2 words type parameter.

#### [Data number]

When the pitch error compensation value

N = 11000 to 11511: Pitch error compensation value of the 1st (X) axis

- N = 12000 to 12511: Pitch error compensation value of the  $2^{nd}(Y)$  axis
- N = 13000 to 13511: Pitch error compensation value of the  $3^{rd}(U)$  axis
- N = 14000 to 14511: Pitch error compensation value of the  $4^{th}$  (V) axis
- N = 15000 to 15511: Pitch error compensation value of the 5th (Z) axis
- N = 16000 to 16511: Pitch error compensation value of the 6th (W) axis
- N = 17000 to 17511: Pitch error compensation value of the 7th (A) axis
- N = 18000 to 18511: Pitch error compensation value of the  $8^{th}(E)$  axis

#### [Data attribute]

M = 0: Writes the no axis parameter.

1 to n: Writes the specific axis parameter

-1: Writes the all axes parameter.

(Note: n is the axis number.)

If a parameter other than the servo parameters is written, M is set to 0.

- 0: Parameter data has been written normally.
- 2: Invalid data is specified as the data length.
- 3: Invalid data is specified as the data number.
- 4 : Invalid data is specified as the data attribute.
- 6: When the data with the parameter number specified for writing cannot be used if options such as pitch error compensation are not available, those options are not provided; or, the system is not in the parameter writing state.

#### [Types of parameters]

#### In the B908 system, data type of ( ) are used.

No.	Length	No.	Data type	No.	Data type	No.	Data type
0000	Bit	0070	Byte	0140	Byte	0210	2W(—)
0001	Bit	0071	Byte	0141	Byte	0211	
0002	Bit	0072	Byte	0142	Byte	0212	—
0003	Bit	0073	Byte	0143	Byte	0213	—
0004	Bit	0074	Byte	0144	Byte	0214	—
0005	Bit	0075	Byte	0145	Byte	0215	—
0006	Bit	0076	Byte	0146	Word	0216	—
0007	Bit	0077	Byte	0147	Word	0217	—
0008	Bit	0078	Byte	0148	Word	0218	—
0009	Bit	0079	Byte	0149	Word	0219	—
0010	Bit	0080	Byte	0150	Word	0220	—
0011	Bit	0081	Byte	0151	Word	0221	—
0012	Bit	0082	Byte	0152	Word	0222	—
0013	Bit	0083	Byte	0153	Word	0223	—
0014	Bit	0084	Byte	0154	VVord	0224	—
0015	Bit	0085	Byte	0155	Word	0225	—
0016	Bit	0086	Byte	0156	Byte	0226	_
0017	Bit	0087	Byte	0157	vvord	0227	_
0018	BIt	0088	Byte	0158	VVord	0228	_
0019	BIL	0089	Byte	0159	VVOrd	0229	_
0020	DIL Dit	0090	Buto	0160	Word	0230	_
0021	DIL Dit	0091	Dyte Word	0161	Word	0231	_
0022	Bit	0092	Word	0162	Word	0232	_
0023	Bit	0093	Word	0103	2words	0233	_
0024	Bit	0094	Word	0165	Word	0234	_
0025	Bit	0000	Word	0105	2words	0235	
0020	Bit	0090	Word	0167	Word	0230	_
0027	Bit	0098	Word	0168	Word	0238	
0020	Bit	0099	Word	0169	Word	0239	_
0030	Bit	0100	Word	0170	Word	0240	_
0031	Bit	0101	Word	0171	Word	0241	_
0032	Bit	0102	Word	0172	Word	0242	_
0033	Bit	0103	Word	0173	Word	0243	_
0034	Bit	0104	Word	0174	Word	0244	_
0035	Bit	0105	Word	0175	Word	0245	_
0036	Bit	0106	Word	0176	Word	0246	_
0037	Bit	0107	Word	0177	Byte	0247	—
0038	Bit	0108	Word	0178	2W(Byte)	0248	—
0039	Bit	0109	Word	0179	2words	0249	—
0040	Byte	0110	Word	0180	Byte(2W)	0250	—
0041	Byte	0111	Word	0181	Byte	0251	—
0042	Word	0112	2words	0182	Byte	0252	—
0043	Word	0113	Word	0183	Byte	0253	—
0044	Word	0114	Word	0184	Byte	0254	—
0045	Word	0115	Word	0185	Byte	0255	—
0046	Byte	0116	Word	0186	Byte	0256	—
0047	VVord	0117	VVord	0187	Byte	0257	—
0048	VVOrd	0118	VVOrd	0188	vvora Durte	0258	_
0049	VVord	0119	VVord	0189	Byte	0259	_
0050	Word	0120	VVOID	0190	Dyte	0260	_
0051	Word	0121	Word	0191	2words	0201	_
0052	Word	0122	2words	0192	2words	0202	_
0053	Word	0123	Byte	0193	2words	0203	_
0055	Word	0124	Byte	0194	2words	0265	
0056	Word	0125	Byte	0196	Word	0265	
0057	Word	0120	Byte	0197	2words	0267	
0058	Byte()	0128	Byte	0198	Byte	0268	_
0059	Byte(—)	0129	Byte	0199	2words	0269	_
0060	Byte	0130	Byte	0200	Byte	0270	_
0061	Byte	0131	Byte	0201	word	0271	_
0062	Byte	0132	Byte	0202		0272	_
0063	Byte	0133	Byte	0203	_	0273	_
0064	Byte	0134	Byte	0204	_	0274	—
0065	Byte	0135	Byte	0205	_	0275	—
0066	Byte	0136	Byte	0206	_	0276	—
0067	Byte	0137	Byte	0207	—	0277	—
0068	Word	0138	Byte	0208	—	0278	—
0069	2words	0139	Byte	0209	—	0279	—

#### E. WINDOW FUNCTION DESCRIPTION (FS16–W)

APPENDIX

B-61863E/14

No.	Length	No.	Data type	No.	Data type	No.	Data type
0280	_	0350	2words	0620	Bvte	0690	Word
0281	_	0351	Bvte	0621	Bvte	0691	Word
0282	_	0352	Byte	0622	Byte	0692	Word
0283	_	0353	Byte	0623	Byte	0693	Word
0284	_	0354	Byte	0624	Word	0694	Word
0285	_	0355	Byte	0625	Word	0695	Word
0286	_	0356	Byte	0626	Word	0696	Word
0287	_	0357	Word	0627	Word	0697	Word
0288	_	0358	_	0628	Word	0698	Word
0289	_	0359	_	0629	Word	0699	Word
0290	_	0360	Word	0630	Word	0700	Word
0291	_	0361	2words	0631	Word	0701	Word
0292	_	0362	2words	0632	2words	0702	Word
0293	_	0363	2words	0633	2words	0703	Word
0294	_	0364	2words	0634	2words	0704	Word
0295	_	0365	2words	0635	2words	0705	Word
0296	_	0366	2words	0636	2words	0706	Word
0297	_	0367	2words	0637	2words	0707	Word
0298	_	0368	2words	0638	2words	0708	Word
0299	_	0369	2words	0639	2words	0709	Word
0300	Bit	0370	2words	0640	2words	0710	Word
0301	Bit	0371	2words	0641	2words	0711	Word
0302	Bit	0372	2words	0642	2words	0712	Word
0303	Bit	0373	2words	0643	2words	0713	Word
0304	Bit	0374	2words	0644	2words	0714	Word
0305	Bit	0375	2words	0645	2words	0715	Word
0306	Bit	0376	2words	0646	2words	0716	Word
0307	Bit	0377	2words	0647	2words	0717	Word
0308	Bit	0378	2words	0648	Word	0718	Word
0309	Bit	0379	2words	0649	Word	0719	Word
0310	Bit	0380	2words	0650	Word	0720	Word
0311	Bit	0381	2words	0651	Word	0721	Word
0312	Bit	0382	2words	0652	Word	0722	Word
0313	Bit	0383	2words	0653	Word	0723	Word
0314	Bit	0384	2words	0654	Word	0724	Word
0315	Bit	0385	— .	0655	Word	0725	Word
0316	Word	0386	2words	0656	Word	0726	Word
0317	—	0387	2words	0657	Word	0727	Word
0318	—	0388	—	0658	Word	0728	Word
0319	_	0389	-	0659	Word	0729	Word
0320	Byte	0390	-	0660	Word	0730	Word
0321	Byte	0391	-	0661	Word	0731	Word
0322	Byte	0392	_	0662	Word	0732	Word
0323	Byte	0393	_	0663	Word	0733	Word
0324	Byte	0394		0664	VVord	0734	VVord
0325	Byte	0395		0665	VVord	0735	VVord
0326	Byte	0396		0666	vvord	0736	vvord
0327	Byte	0397			VVOrd	0/3/	VVOrd
0320	Dyte	0390	_	0000	VVOID	0730	VVOI U
0329	Byte	0399	 B:+	0670	Word	0739	Word
0330	Word			0070	Word	0740	VVOIU Mord
0331	2wordo	0001		0071	2wordo	0741	Word
0332	2words	0002		0672	2words	0742	Word
0333	2word	0003		0073	2words	0743	Word
0334	Word	0604	Rit	0675	2words	0744	Word
0335	2worde	0000	Bit	0676	2worde	0745	Word
0330	2worde	0000	Rit	0677	2words	0740	2worde
0337	2worde	8090	Rit	0678	2words	0742	2worde
0339	2words	0609	Bit	0679	2words	0740	2words
0340	2worde	0610	Rit	0880	Word	0750	2words
0341	2words	0611	Bit	0681	Word	0751	2words
0342	2words	0612	Bit	0682	Word	0752	2words
0343	Bvte	0613	Bit	0683	Word	0753	2words
0344	Word	0614	Bit	0684	Word	0754	2words
0345	Byte	0615	Bit	0685	Word	0755	Word
0346	Word	0616	Byte	0686	Word	0756	Word
0347	Bvte	0617	Byte	0687	Word	0757	Word
0348	2words	0618	Byte	0688	Word	0758	2words
0349	2words	0619	Byte	0689	Word	0759	2words

B-61863E/14

No.	Length	No.	Data type	No.	Data type	No.	Data type
0760	2words	0810	2words	0860	Word	0910	_
0761	2words	0811	Byte	0861	Bit	0911	—
0762	2words	0812	Byte	0862	Bit	0912	—
0763	2words	0813	Byte	0863	Byte	0913	_
0764	2words	0814	Byte	0864	Byte	0914	_
0765	2words	0815	Byte	0865	Byte	0915	_
0766	2words	0816	Bvte	0866	Bvte	0916	_
0767	2words	0817	Byte	0867	Byte	0917	_
0768	2words	0818	Byte	0868	Byte	0918	_
0769	2words	0819	Word	0869	Byte	0919	_
0770	2words	0820	Word	0870	Byte	0920	_
0771	2words	0821	Word	0871	Byte	0921	_
0772	2words	0822	Word	0872	Word	0922	_
0773	2words	0823	Word	0873	Word	0923	
0774	2words	0824	Word	0874	Word	0924	_
0775	2words	0825	Word	0875	Word	0925	
0776	2words	0826	Word	0876	Word	0926	_
0777	2words	0827	Word	0877	Word	0927	
0778	2words	0828	Word	0878	Word	0928	
0779	2words	0829	Word	0879	Word	0929	
0780	2words	0830	Word	0880	Word	0930	
0781	2words	0831	Word	0881	Word	0000	
0782	2words	0832	Word	0882	Word	0037	
0783	2words	0833	Word	0883	Word	0002	
0703	2words	0834	Word	0884	Word	0000	
0785	2words	0835	vvoiu	0885	Word	0934	
0786	2words	0836	Word	0886	Word	0000	
0787	2words	0837	Word	0887	Word	0930	
0788	2words	0838	Word	0888	Word	0007	
0780	2words	0830	Word	0880	Word	0000	
0700	2words	0840	Word	0800	Word	0000	
0700	2words	0841	Word	0801	Word	0040	
0791	2words	0842	Word	0802	Word	0941	
0732	2words	0042	Word	0802	Word	0042	
0793	2words	0843	Word	0893	Word	0943	_
0705	2words	0845	Word	0805	Word	0045	
0795	2words	0845	Word	0895	Word	0945	_
0730	2words	0847	Word	0897	Word	0940	
0709	2words	0949	Word	0000	Word	0049	
0730	2words	0840	Word	0000	(Mord)	0040	
0733	2words	0850	Word	0000	(VOIU) (Bit)	0343	
0800	2words	0050	Word	0001	(Dit) (Bit)		
0802	2words	0852	Word(2)//)	0002	(Dit) (Bit)		
0002	2worde	0002	\/\ord(2\/\)	0302	(Bit)		
0804	2words	0854	Mord(2\W)	0903	(Byte)		
0004	2words	0004	\/\ord(2\/\)	0904	(Byte)		
0806	2words	0000	\/\ord(2\/\)	0905			
0000	2words	0000	\/\ord(2\/\)	0900			
0007	2worde	0007	\/\ord(2\/\)	0907			
0000	2words	0050	Mord(2\\/)	0000			
0009	Zworus	0009	vvoiu(Zvv)	0909	_		

APPENDIX

#### E.5 READING SETTING DATA

[Data contents] Setting data on the CNC can be read.



[Data number]

See the setting data list.

- 0: Read operation was terminated normally.
- 3: An incorrect data number was specified.

#### E.6 WRITING SETTING DATA (LOW-SPEED TYPE)

[Data contents]

Setting data on the CNC can be written.

[Input data structur	e]	[Output data structu	ure]	
Top address	(Function code) 20	Top address	(Function code) 20	
+2	(Completion code) -	+ 2	(Completion code) ?	
+ 4	(Data length) L	+ 4	(Data length) Input data	
+ 6	(Data number) N	+ 6	(Data number) Input data	
+ 8	(Data attribute) -	+ 8	(Data attribute) -	
+ 10	(Data area) Setting data	+ 10	(Data area) Input data	

[Data length]

See the setting data list.

[Data number]

See the setting data list.

- 0: Read operation was terminated normally.
- 2: An incorrect data length was specified.
- 3: An incorrect data number was specified.

## E. WINDOW FUNCTION DESCRIPTION (FS16–W)

APPENDIX

#### Setting Data List

Setting data	Screen	Data number	Data length	Bit name
X mirror image	Handy	1	1	Bit 0
Y mirror image	Handy	1	1	Bit 1
Axis switching	Handy	1	1	Bit 2
TV check	Handy	1	1	Bit 3
Output code	Handy	1	1	Bit 4
Inputunit	Handy	1	1	Bit 5
Parameterwritable	Handy	1	1	Bit 6
Input unit multiplication by 10 times	Handy	1		Bit 7
Automatic recovery from power failure	Handy	2		Bit 3
Automatic recovery non power failure	Handy	2		Bit 0
Automaticpower-off M00/M10	Handy	3	1	Bit 1
Automatic power backward movement	Handy	3	1	Bit 2
alarm	Tiandy	3		Dit 2
Automatic nowor disconnection	Handy	2	1	Rit 2
	Handy	3		Dit 3
Figure magnification ratio	Landy	4		_
Figure magnification ratio	Handy	C C	4	_
Figure rotation angle	Handy	0	4	I
Taper machining mode	Taper	15	1	-
Guide type	Taper	16	1	Bit 0
Program surface position	Taper	17	4	_
Workpiece thickness	Taper	18	4	-
Drawing surface position	Taper	19	4	_
Upper guide position	Taper	20	4	-
(Lower guide position)	Taper	21	4	-
(Vertical position U)	Taper	22	4	_
(Vertical position V)	Taper	23	4	-
Enable/disable	PWB	2	1	_
Wire diameter		7	1	
Machining groove width		/ 0	4	_
Warkpiece thickness		0	4	-
Wire deflection		9	4	-
		10	4	—
Effectiveness of concave	PVVB	10	4	_
Automotio avorrido	PWB	12	4	_
Automaticovernde		2	4	
Enable/disable	PVVB	2 10		BIt 5
Dillerentiarvoltage	FVVB	13	2	-
Enable/disable	AWF	2	1	Bit 2
Disconnection repair	AWF	2	1	Bit 1
Prepared hole of 0.5	AWF	2	1	Bit 6
Sumpmachining	AWF	2	1	Bit 7
Portion to be left uncut	AWF	14	1	_
Power reduction ratio (setting)	AWF	24	1	_
Connection position U1	AWF	26	4	-
Connection position V1	AWF	27	4	_
Connection position Z1	AWF	28	4	-
Number of retries (setting)	AWF	40	1	_
Number of allowable disconnections	AWF	41	1	-
(setting)				
Number of retries (current)	AWF	42	1	-
Number of allowable disconnections	AWF	43	1	-
(current)				
WIRE REST	LIFE	30	2	_
WIRE CUTTER REST	LIFE	31	2	_
() REST	LIFE	32	2	_
( ) REST	LIFE	33	2	_
( ) REST	LIFE	34	2	_
( ) REST	LIFE	35	2	_
( ) REST	LIFE	36	2	_
( ) REST	LIFE	37	2	_
· · · · · · · · · · · · · · · · · · ·			-	

# E. WINDOW FUNCTION DESCRIPTION (FS16–W)

#### Setting Data List

Setting data	Screen	Data number	Data length	Bit name
OP. BLOCK SKIP /0	Others	38	1	Bit 0
OP. BLOCK SKIP /1	Others	38	1	Bit 1
OP. BLOCK SKIP /2	Others	38	1	Bit 2
OP. BLOCK SKIP /3	Others	38	1	Bit 3
OP. BLOCK SKIP /4	Others	38	1	Bit 4
OP. BLOCK SKIP /5	Others	38	1	Bit 5
OP. BLOCK SKIP /6	Others	38	1	Bit 6
OP. BLOCK SKIP /7	Others	38	1	Bit 7
OP. BLOCK SKIP /8	Others	39	1	Bit 0
OP. BLOCK SKIP /9	Others	39	1	Bit 1
Number or tries	AWF	40	1	-
Number of retries	AWF	41	1	-
Number or tries	AWF	42	1	-
Number of retries	AWF	43	1	-
Program number	Graphic	44	2	-
(for machining distance calculation)				

APPENDIX

#### E.7 READING THE CNC ALARM STATUS

[Description]

When the CNC is placed in the alarm status, the alarm status data can be read.



#### [Contents of data]

(1) Alarm status data in first byte.

	#7	#6	#5	#4	#3	#2	#1	#0	
	EOR	OTM	OTS	ОН		SV	OTH	PS	
	PS	: P/S al	: P/S alarm						
	OTH	: Over t	ravel al	arm					
	SV	: Servo alarm							
	OH	: Overheat alarm							
	OTS	: First s	: First stroke limit alarm						
	OTM	: Secon	: Second stroke limit alarm						
	EOR	: Edit a	larm						
(2)	Alarm statu	ıs data in	second	byte.					

#7	#6	#5	#4	#3	#2	#1	#0
							APCER

APCER : Absolute pulco alarm

#### [Completion codes]

0 : This alarm status in the CNC has been read normally.

#### E.8 READING MODEL DATA

[Description]

Modal information in the CNC can be read.

(1) Format and types of modal data for the G function Data with a specified identification code is read and stored in the data area. Whether the data is located in the block specified with the data attribute can be determined with the most significant bit.



Identification code	Data type	Data
5	G94	0
	G95	1
6	G20	0
	G21	1
7	G40	0
	G41	1
	G42	2
8	G50	0
	G51	1
	G52	2
9	G60	0
	G61	1
	G62	2
	G63	3
10	G48	1
	G49	0
11	G65	26
	G66	0
	G67	1



(2) Format and types of modal data for other than the G function

Identification code	Specified address	Meaning of value
100	В	
101	D	
102	E	Offset number
103	F	
104	Н Н	Feedrate
105	L	
106	M	
107	S	
108	Т	Tapper data
109	R	
110	P	
111	Q	



— 1358 —

#### E. WINDOW FUNCTION DESCRIPTION (FS16–W)

[Output	data	structure]
---------	------	------------



#### [Data length]

- L = 2 : G function
  - 2*16 : All data of G functions
  - 6 : Other than G functions
  - 6*12 : All data of other than G functions

#### [Data number]

- N = 0 and up: See each data list.
  - -1 : All data of G functions
  - -2 : All data of other than G functions

#### [Data attribute]

- M = 0 : Current block
  - 1 : Next block
  - 2 : Block after the next block

- -1 : The modal data read command could not be executed. This means that, because the modal data was being updated on the CNC, the command could not be executed.
- 0 : Modal data read operation terminated normally.
- 3 : An incorrect data number was specified.
- 4 : An incorrect data attribute was specified.

#### E.9 READING MACHINING DISTANCES

#### [Data contents]

The distance (machining distance) from the machining start point to the current point, and the distance (whole distance) from the machining start point to the point where M02 or M30 is specified can be read.



#### [Data number]

N = 0	: Machining distance
N = 1	: Whole length obtained with the dry run function
N = 2	: Whole length obtained by drawing
N - 3	· Whole length obtained with the machining dis

# N = 3 : Whole length obtained with the machining distance calculation function

#### [Data unit]

Metric input : 10⁻³ [mm] (When the increment system is 1/10: 10⁻⁴ [mm])

Inch input :  $10^{-5}$  [inch]

- -1: Data could not be read. This means that, because the data was being updated on the CNC, the command could not be executed.
- 0 : Data read operation terminated normally.
- 3 : An incorrect data number was specified.

#### E.10 READING THE MEASURED POINT

#### [Description]

The measured point that are get by positioning can be read. Also, the slit width by slitting and the hole diameter by centering can be read.

[Input	data	structure]
--------	------	------------

Top address	(Function) 185			
+2	(Completion)			
+ 4	(Length)	Reads measuredpoint	Reads slit width of hole diameter	
+ 6	(Number) N	Pointnumber	0	
+ 8	(Attribute) M	0	1	
+ 10	(Data area) —			

#### [Output data structure]

Top address	(Function) 185			
+2	(Completion) ?	Reads measuredpoint	Reads slit width of hole diameter	
+ 4	(Length) L	10	4	
+ 6	(Number) N	Pointnumber	0	
+ 8	(Attribute) M	0	1	
+ 10	(Data area)	Туре	Slit width of hole diameter	
+ 12		Machine		
+ 14		Coordinate of X axis		
+ 16		Machine		
+ 18		coordinate of Y axis		
+ 20		1	J	

[Data unit]

Metric system input : 10⁻³ [mm]

APPENDIX

(In case the incremental system is 1/10, output data unit is  $10^{-4}$  [mm].)

Inch system input :  $10^{-5}$  [inch]

[Completion codes]

0 : The measured point has been read normally.

3 : Invalid data is specified as data number.

4 : Invalid data is specified as data attribute.

#### E.11 WRITING THE MEASURED POINT (*LOW-SPEED RESPONSE)

#### [Description]

The measured point that are get by positioning can be written. Also, the slit width by slitting and the hole diameter by centering can be written.

[Input data structur	e]	[Output data structur	e]	
Top address	(Function) 186	Top address	(Function) 186	
+2	(Completion)	+ 2	(Completion) ?	
+ 4	(Length) 10	+ 4	(Length) 10	
+ 6	(Number) Pointnumber	+ 6	(Number) N	
+ 8	(Attribute) 0	+ 8	(Attribute) 0	
+ 10	(Data area) Type	+ 10	(Data area)	
+ 12	Machine coordinate of X axis	+ 12		
+ 14		+ 14		
+ 16	Machine coordinate of Y axis	+ 16		
+ 18		+ 18		
+ 20		+ 20		

[Data unit]

Metric system input :  $10^{-3}$  [mm]

(In case the incremental system is 1/10, output data unit is  $10^{-4}$  [mm].)

Inch system input :  $10^{-5}$  [inch]

- 0 : The measured point has been written normally.
- 3 : Invalid data is specified as data number.
- 4 : Invalid data is specified as data attribute.



APPENDIX

#### F.1 READING OF TOOL SETTING DATA

[Description]

Various Tool setting data recorded in the CNC can be read.

[Input data structure]



- 0: The tool setting data has been read normally.
- 3: The data number specified for reading is invalid.
- 4 : There are mistakes in the data attribute that specifies the type of the tool setting data to be read.
- 6: For the tool setting data specified for reading, an additional option (graphic or multi-tool control) is required, but it is missing.

#### F. WINDOW FUNCTION DESCRIPTION (FS16–PA)

### Top address + 0 (Function code) 188 2 (Completion code) ? (See the explanation of codes) 4 (Data length) ? (See 1.1) 6 (Data Number) N (N : Input data) 8 (Data attribute) (M : Input data) 10 (Data area) ? (See 1.1) 48

#### [Output data structure]

#### F.1.1

#### Data Number, Data Attribute, Data Length Data A

Data number, data attribute, data length and data area of various tool setting data are as follows.

Data Length, Data Area				
Various tool setting data	Data number (N)	Data attribute (M)	Data length	Data area
Used tool number		0	2 bytes	Binary 1 to 136
Number of turret indexing		1	2 bytes	Binary 1 to 136
Tool number of reference point	0	2	2 bytes	Binary 1 to 136
Feed amount per revolution of turret		3	4 bytes	Binary 1 to 99999999
Total punch count		4	8 bytes	Binary 1 to 99999999
Tool number		0	2 bytes	Binary 1 to 9999
Punch count		1	4 bytes	Binary 1 to 99999999
Tool position compensation of X		2	4 bytes	Binary ±99999999
Tool position compensation of Y		3	4 bytes	Binary ±99999999
Machine position of tool	Number of tool setting data	4	4 bytes	Binary ±99999999
Tool number for tool change		5	2 bytes	Binary 1 to 136
(Not used)	1 10 100	6	4 bytes	
Tool shape (C) for graphic		7	2 bytes High byte=0	Binary 0 to 4
Tool shape (I) for graphic		8	4 bytes	Binary 0 to 999999
Tool shape (J) for graphic		9	4 bytes	Binary 0 to 999999
Tool shape (K) for graphic		10	4 bytes	Binary 0 to 360000
Tool number for multi-tool		0	2 bytes High byte=0	Binary 0 to 99
Tool angle for multi-tool		1	4 bytes	Binary ±360000
Tool position compensation of Y	Number of	2	4 bytes	Binary ±99999999
Tool shape (C) for multi-tool	Multi–tool setting data +200 201 to 264	3	2 bytes High byte=0	Binary 0 to 4
Tool shape (I) for multi-tool		4	4 bytes	Binary 0 to 999999
Tool shape (J) for multi-tool		5	4 bytes	Binary 0 to 999999
Tool shape (K) for multi-tool		6	4 bytes	Binary 0 to 360000

#### data unit

	Machine	Input of IS-A	Input of IS-B
Tool position	mm	0. 01	0. 001
compensation	inch	0. 001	0. 0001

	Input unit	Input of IS-A	Input of IS-B
Tool shape and	mm	0. 01	0. 001
angle for graphic	inch	0. 001	0. 0001
Tool angle for multi–tool	deg	0. 01	0. 001

APPENDIX

# F.2[Description]WRITING OF TOOLThe various tool setting data can be directly written into the CNC.SETTING DATAIterational setting data can be directly written into the CNC.(LOW-SPEED<br/>RESPONSE)Iterational setting data can be directly written into the CNC.



- 0: The tool setting data has been written normally.
- 2: The data length specified for writing is invalid.
- 3: The data number specified for writing is invalid.
- 4: The data attribute specified for writing is invalid.
- 5: The data specified for writing is invalid.
- 6: The additional option (multi-tool control or graphic) is required but it is missing.

#### F. WINDOW FUNCTION DESCRIPTION (FS16–PA)



#### [Output data structure]



#### F.3 READING TOOL SETTING DATA BY SPECIFYING TOOL NUMBER

#### [Description]

Setting data for a tool (such as registration order, tool punch count, and tool shape) can be read by specifying the tool number.

#### [Input data structure]



#### NOTE

- 1 The area for specifying the data number consists of four bytes.
- 2 As the data attribute, specify the type of the tool setting data to be read, in the same way as for function code 188. If 0 is specified as the data attribute, the registration order of the tool is read.

[Completion code]

APPENDIX

- 0: The tool setting data has been read normally.
- 3: The specified data number is invalid.
- 4: The specified data attribute is invalid.
- 6: For the tool setting data specified for reading, an additional option (graphic or multi-tool control) is required, but it is missing.

[Output data structure]



#### F.4 OTHER WINDOW FUNCTIONS

The FS16–PA supports the following window functions, described in Appendix B of this manual.

Number		Function code
1	Reading CNC system information	0
2	Reading a tool offset	13
3	Writing a tool offset         *low-speed response	14
4	Reading a workpiece origin offset	15
5	Writing a workpiece origin offset         *low-speed response	16
6	Reading a parameter         *low-speed response	17
7	Writing a parameter         *low-speed response	18
8	Reading setting data *low-speed response	19
9	Writing setting data *low-speed response	20
10	Reading a custom macro variable         *low-speed response	21
11	Writing a custom macro variable         *low-speed response	22
12	Reading the CNC alarm state	23
13	Reading the current program number	24
14	Reading the current sequence number	25
15	Reading an actual velocity for a controlled axis	
16	Reading an absolute position on a controlled axis	
17	Reading a machine position on a controlled axis	
18	Reading a skip position on a controlled axis	
19	Reading a servo delay amount on a controlled axis	
20	Reading an acceleration/deceleration delay amount on a controlled axis	
21	Reading modal data	32
22	Reading diagnostic data *low-speed response	33
38	Reading clock data (date and time)	151
41	Reading a parameter	154
42	Reading setting data	
43	Reading diagnostic data	
44	Reading a character string of the CNC program being executed in the buffer	
45	Reading the relative position on a controlled axis	
46	Reading the remaining travel on a controlled axis	
47	Reading CNC status information	
48	Reading an operator message	83

# SIGNAL ADDRESS CONVERSION (FROM THE PMC-MODEL L/M TO THE PMC-MODEL SB/SC)

G.1 GENERAL	DI/ sigr for	DI/DO signals used in the PMC–MODEL L/M can be converted to signals for the PMC–MODEL SB/SC using the FAPT LADDER program for the PMC–MODEL SB/SC.		
G.2 FUNCTION	The to w bit	DI/DO signals used between the word addresses consisting of addr type are converted. The program	e NC unit and the PMC correspond esses and values. Word addresses of n is not logically converted.	
	The	conversion is performed under	the following conditions.	
	(1)	A word address of bit type u converted.	sed in a basic instruction is to be	
	(2)	A word address of byte type us converted.	sed in a functional instruction is not	
	(3)	Word addresses used in the s converted to those used in the s a value in a word address is converted.	tandard FANUC Series 0–T/M are standard FANUC Series 16–T/M. If 1000.0 or more, the address is not	
	(4)	When the same signal name is and the addresses correspondin have one-to-one relationship, details, see the signal conversion	used in the FANUC Series 0 and 16, g to the signal in the Series 0 and 16 the word address is converted. For on table.	
G.3	(1)	Load the FAPT LADDER prog	gram for the PMC–SB/SC.	
CONVERSION	(2)	Press the R0 key to display the	programmer menu screen.	
	(3)	Press the F2 key. Enter 2 and pro- message appears on the scree PMC–SB/SC. Select the nam- conversion from Table G.3 and	ess the <nl> key. then the following en. Insert a data floppy for the ne of the file corresponding to the l enter it.</nl>	
		SET FD & KEYIN 'OK', 'K FD0 = OK <drive> &lt;@NAM FD0 =</drive>	ILL' OR 'NO' IE OR :NUMBER>	
		Table G.3 File Name in the Da	ta Floppy for the PMC–RB/RC	
			File name	
	FS	$0-T \rightarrow FS16-T$	COMV.FS0-T	

$FS0-M \rightarrow FS16-M$		COMV.FS0-M
(4)	Read a source ladder program of PMC–L/M from the floppy in	created with FAPT LADDER for the the same way as in Item 3.

If an address not listed in the signal conversion table is used in the ladder program file, an error occurs. In this case, enter E, then press the <NL> key to return to the programmer menu screen.

PART= E <NL>

(5) Entering 9 and pressing the <NL> key on the programmer menu screen changes the screen. The following message appears at the lower left corner of the screen. Enter 2, then press the <NL> key to delete the symbol data.

KEYIN '1, 2, 3, 4, 5 OR 6 OR 'NO' CLEAR/KEEP=

B–61863E/14	(FROM THE PMC–MODEL L/M TO APPENDIX THE PMC–MODEL SB/SC)
G.4 MODIFYING THE CONVERTED SEQUENCE PROGRAM	The above operation terminates the conversion. Check the converted program. If an error occurs in the conversion, modify the program. Enter 1 on the programmer menu to change the screen to the screen for editing a sequence program. Editing operation is the same as usual.           NOTE           Some addresses not converted have no error indication. After modifying the program, check that all addresses are correct according to the signal conversion table and the connecting manual.
G.4.1 Modification Procedure	<ul> <li>When the Series 0 and 16 differ in the number of parameters used in a functional instruction</li> </ul>
	Because the Series 0 and 16 differ in the numbers of parameters used for TMR (timer), TMRB (timer), and CTR (counter), errors are indicated at the parameters. Check the program, then delete the parameter. Set the timer and counter again.
	(2) When an address not used in functional instructions is specified
	When an address used in the ladder program for the Series 0 is not defined in the Series 16, the messages (NO PARAMETER) and #PARAM.ERROR# appear as follows. Set the parameter again and delete the latter message.
	Example 00001 RD XXX.X 00002 SUB 8 00003 XXXX 00004 XXXX 00005 XXXX 00006 (NO PARAMETER) Set the parameter again. 00007 #PARAM.ERR# Delete the message. This message may not appear.
	(XXX.X and XXXX are addresses and values.)
	(3) Deleting SUB48 (END3) (In the PMC–SB)
	If SUB48 (END3) is specified in the PMC–SB, an error occurs because the PMC–SB is not provided with SUB48 (END3). When this error occurs, delete third–level programs, or change the third–level programs to second–level programs and delete SUB48.
	(4) Address conversion for signals not listed on the signal conversion table Modify the address for a signal by referring to the connection manual.

G. SIGNAL ADDRESS CONVERSION
# CONNECTING THE OPERATOR'S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate

# H.1 GENERAL

The Series 0 operator's panel consists of key switches, LEDs, a rotary switch, and so on. Because the states of key switches and lamps are coded, the number of the signal lines required for connecting the operator's panel with the CNC may not be the same as the number of actual switches. PMC management software automatically codes the states of the key switches and lamps and transmits data.

Therefore, simple bit images of switches and LEDs must only be manipulated with the PMC ladder program.



APPENDIX

Fig. H.1 (a) Connection between the CNC and the operator's panel

The operator's panel is made up of the following keys, LEDs, etc.

- Key switch (Seat key)
- 42 keys (0–TC)
- 46 keys (0–MC)
- LEDs (red) .....Prepared for all key switches
- Override rotary switch ..... 4 bits
- Emergency stop button .... 1 bit
- Program protect key ...... 1 bit



Fig. H.1 (b) Front view of operator's panel for 0-TC



Fig. H.1 (c) Front view of operator's panel for 0-MC

APPENDIX



Fig. H.1 (d) External view of operator's panel for 9" CRT/MDI with full-keyboard (0-TC)



Fig. H.1 (e) External view of operator's panel for 9" CRT/MDI with full-keyboard (0-MC)

APPENDIX

# H.2 CONNECTION

### H.2.1 Connecting the I/O Unit



### H.2.2 Connecting the I/O Card



# H.3 SIGNALS FOR CONNECTING THE OPERATOR'S PANEL

H.3.1 Emergency Stop Signal (*ESP)	This signal is used for the fixed address directly monitored by the CNC. For connecting the signal, refer to the description of the interface between the CNC and the PMC in the "Series 16 or 18 Connection Manual."				
H.3.2 Override Signals (*OV1	Their key switch contact signals are directly input to the PMC. Handle them with the PMC ladder program.				
to *OV8) and Program Protect Key Signal (KEY)	For connecting these signals, refer to the description of the interface between the CNC and the PMC in the "Series 16 or 18 Connection Manual."				
H.3.3 Key Switch Signals	The key switch signals are coded by the PMC management software, and input to the area indicated by address R in the form of to the bit image				
(Xn, Xn+2)	Whether necessary keys are already pressed can be checked by the bit image of the key switches using the user PMC ladder program. (See Tables H.3.4 (a), H.3.4 (b), and H.3.4 (c))				
	While a key is pressed, the bit corresponding to the key is 1.				
	Two keys can be pressed at the same time. Create a user PMC program so that it does not require pressing more than two keys at a time. If more than two keys are pressed simultaneously, the relevant data is not entered correctly.				
	A maximum of 60 ms is required before the corresponding bit is set to 1 or 0 after a key is pressed (released).				
	Key switch signal addresses (Xn to Xn+2: Table H.3.4 (a)) and their bit image addresses (Rk to Rk+7: Tables H.3.4 (b) and H.3.4 (c)) can be defined using fixed addresses or unused addresses as desired. (In Series 0, the key switch signal addresses are fixed to X20 and after. The bit image addresses are fixed to F292 and after.)				

APPENDIX

H.3.4 LED Signals (Ym)	Specify the LED signals at PMC address R using the user PMC ladder program in the form of a bit image. PMC management software changes the bit image LED signals to the coded output signals. (See Tables H.3.4 (a), H.3.4 (b), and H.3.4 (c))
	While 1 is written in a LED bit image, the relevant LED automatically goes on. When 0 is written in the LED bit image, the relevant LED goes off. All LEDs are off before the power is turned on.
	A maximum of 200 ms is required before the LED goes on or off after 1 or 0 is written in a bit image in the PMC.
	LED signal address (Ym: Table H.3.4 (a)) and the bit image addresses (Rl to Rl+7: Tables H.3.4 (b), and H.3.4 (c)) can be defined using fixed

to Rl+7: Tables H.3.4 (b), and H.3.4 (c)) can be defined using fixed addresses or unused addresses as desired. (In Series 0, the LED signal address is fixed to Y51. The bit image addresses are fixed to G242 and after.)

 Table H.3.4 (a) Key switch and LED signal addresses

	#7	#6	#5	#4	#3	#2	#1	#0
Xn	KD7	KD6	KD5	KD4	KD3	KD2	KD1	KD0
Xn+1								
Xn+2	KST				KA3	KA2	KA1	KA0
Ym	LD7	LD6	LD5	LD4	LD3	LD2	LD1	LD0

# Table H.3.4 (b) Bit Image addresses of key switch and LED signals (for the small operator's panel)

(··· ··· ··· · · · · · · · · · · · · ·									
	#7	#6	#5	#4	#3	#2	#1	#0	
KEY/LED	F3	F2	F1 D1 C1 B1		B1	A1			
	r								
Rk/RI	F4				D2	C2	B2	A2	
Rk+1/Rl+1	D4	D3	C4	C3	B4	B3	A4	A3	
Rk+2/Rl+2		F6	F5		D5	C5	B5	A5	
Rk+3/Rl+3	F8				D6	C6	B6	A6	
Rk+4/Rl+4	D8		C8		B8		A8	A7	
Rk+5/Rl+5			F9		D9	C9	B9	A9	
Rk+6/RI+6			F10		D10	C10	B10	A10	

	(for the operator's panel with the full keyboard)												
	#7	#6	#5	#4	#3	#2	#1	#0					
KEY/LED	E1	C1	A1	E6	D6	C6	B6	A6					
Rk/RI	E2	C2 A2		E7 D7		C7	B7	A7					
Rk+1/Rl+1	E3	C3	A3	E8	D8	C8	B8	A8					
Rk+2/RI+2	E5	C4	A4	E9	D9	C9	B9	A9					
Rk+3/Rl+3	D2	C5	A5	E10	D10	C10	B10	A10					
Rk+4/RI+4	D4	D4 D5 B2		E11 D11		1 C11 B1'		A11					
Rk+5/Rl+5	D1	B1	B4	E12	D12	C12	B12	A12					
Rk+6/Rl+6	D3	B3	B5	E13	D13	C13	B13	A13					

# Table H.3.4 (c) Bit image addresses of key switch and LED signals (for the operator's papel with the full keyboard)

APPENDIX

### H.4 SPECIFYING ADDRESSES

### H.4.1 Parameter Menu

The following section describes how to specify key switch and LED signal addresses and the bit image addresses.

1				
	KEY	IN ONE OF THE FOLLOWING N	10	S WHICH YOU WANT TO SET PARA.S
	NO.	ITEMS		CURRENT PARAMETERS
	01	(UNUSED)	;	
	02	COUNTER DATA TYPE	;	BINARY
	03	OPERATOR PANEL	;	YES
		KEY/LED ADDRESS	;	X0000/Y0000
		KEY/LED BIT IMAGE ADRS.	;	R0900/R0910
	04	PMC TYPE	;	PMC-RB
	05	(UNUSED)	;	
	06	(UNUSED)	;	
	07	(UNUSED)	;	
	08	(UNUSED)	;	
	09	IGNORE DIVIDED CODE	;	NO
	10	(UNUSED)	;	
	00	NOTHING TO SET	;	
			;	ROM WRITER = FA WRITER
	NO.	=		
-				

H.4.2 Procedure

1) Select 3 from the parameter menu. Then, the following message is displayed:

EXAMPLE 0:NO, 1:YES OP.PANEL=_

2) Select 1(:YES). Then, the following message is displayed:

SET KEY/LED ADDRESS(KEY ADRS., LED ADRS.) ADDR=_

3) Specify a key or LED address (X or Y). For example, to specify a key switch address as X0 and LED address as Y0, enter X0,Y0 and press the [NL] key ([NL]: New line key). The following message is then displayed:

SET KEY/LED BIT IMAGE ADDRESS(KEY ADRS., LED ADRS.) ADDR=_

4) Specify bit image addresses. For example, to specify R900 and R910, enter R900,R910 and press the [NL] key.

Then, the current display returns to the original parameter menu, and the following message appears:

(						
ĺ ,	03 [:]	: OPERATOR	: PANEL		;	YES
		KEY/LED	ADDRESS		;	x0000/Y0000
l	:	KEY/LED :	BIT IMAGE :	ADRS.	;	R0900/R0910

### CAUTION

APPENDIX

1 After the above procedure, the addresses in Tables 3.1, 3.2-A, and 3.2-B are defined as the following PMC addresses: Xn  $\rightarrow$  X0000 Rk / RI →R0900/R0910 Xn+1  $\rightarrow$  X0001 Rk+1 / RI+1→R0901/R0911 Rk+2 / RI+2→R0902/R0912 Xn+2  $\rightarrow$  X0002 Rk+3 / RI+3→R0903/R0913  $Ym \rightarrow Y0000$ Rk+4 / RI+4→R0904/R0914 Rk+5 / Rl+5→R0905/R0915 Rk+6 / RI+6→R0906/R0916 Rk+7 / RI+7→R0907/R0917 2 Since the PMC addresses for the I/O card are already fixed, specify the signals to be used at the fixed addresses.

### **Examples**

To use X1000, X1001, X1002, and Y1000 for key switches and LEDs, enter the following:

SET KEY/LED ADDRESS(KEY ADRS., LED ADRS.) ADDR= X1000,Y1000 [NL]

# EDITING FOR Power Mate-MODEL D (PMC-PA1/PA3)

# I.1 OUTLINE

Ladder diagram editing function for FANUC PMC–MODEL PA1/PA3 has high compatibility in a basic specification between ladder diagram editing function for FANUC PMC–MODEL SA1/SA2.

Following abbreviations are used in this chapter.

CNC Model	Product/Card Name	Abbr.
FANUC Power Mate-MODEL D	FANUC PMC-MODEL PA1	PMC-PA1
	FANUC PMC-MODEL PA3	PMC-PA3
	Ladder diagram editing memory card	Editing card
FANUC Series 18	FANUC PMC-MODEL SA1	PMC-SA1
	FANUC PMC-MODEL SA2	PMC-SA2

I.2 COMPATIBILITY WITH CNC BASIC SOFTWARE

Editing card described herein apply to the following software or later.

### CNC

 Version 08(H) or later of Power Mate–MODEL D basic software 8830 Series.

### PMC

.

Version 04(D) or later of PMC–PA1/PA3 control software 4075 Series.

APPENDIX

# I.3 PMC PROGRAMMER (CRT/MDI OR PDP/MDI) [LADDER EDITING FUNCTION]

This function is used to set PMC system parameters and also generate and execute sequence programs by using soft keys a on the CRT/MDI unit or PDP/MDI unit. You can not use following function because FANUC Power Mate–MODEL D does not use ROM for sequence program.

- Sequence Program Copy Function
- Writing, Reading, and Verification of the Sequence Program and PMC Parameter Data to/from/with ROM.

### I.3.1 Component Units and Connections

The units required for generating a sequence program and connection methods are described below.

I.3.1.1	(1) Editing card
Component units	This is used for editing sequence program.
	If this card is inserted in CNC at the time of its power-on, PMC displays the programmer menu.
	When you want to put on and take off, you must turn off the CNC power.
	<b>CAUTION</b> Please do not release the write protect switch of editing card for preventing a mistake deleting.



Fig. I.3.1.1

(2) CRT/MDI unit, PDP/MDI unit

CRT/MDI unit or PDP/MDI unit are necessary when you generate or edit sequence program using editing card.

CRT/MDI unit (A02B-0166-C001)

PDP/MDI unit (A02B-0166-C010, A02B-0166-C011)

APPENDIX



Fig. I.3.1.2



(The programmer menu is displayed.)

### I.3.2 Specification and Display of System Parameters (SYSPRM)

FANUC Power Mate-MODEL D can set only COUNTER DATA TYPE.

The meaning of this parameter is same as PMC-SA1/SA2.

1									
	PMC SYSTEM COUNTER	M PARAM DATA T	ETER YPE	=	BINA	RY/BC	D		
	[DTNADY]			-	,	r	,	,	1
$\left( \right)$	[DINARI]	L BCD	1 [	L	1	L	1	L	· ·

Fig. I.3.2 PMC–PA1 or PA3 System Parameter Screen

 I.3.3
 When the following condition is satisfied, the CONDNS key will be used in FANUC Power Mate–MODEL D.

 • Some unused area remain by repeating the addition or the deletion of the symbol/comment and the message in the memory.

• Ladder might be able to be made more by compressing the unused area by pushing [CONDNS] key when the memory is insufficient while ladder is added.

[Example: When you want to expand ladder area by deleting symbol/comment data at the memory status Fig.I.3.3 (a)]

- (1) Delete symbol data(0.2KB).
- (2) Push [CONDNS] key.
- (3) The memory status becomes as Fig.I.3.3 (c) and LADDER can be edited more.

### NOTE

- 1 Sequence program area in Fig. I.3.3 (a) (c) is 64KB.
- 2 The underlined memory in Fig. I.3.3 (a) (c) is the same as the memory display of the TITLE screen.
- 3 The symbol/comment area in Fig. I.3.3 (a) is 20KB (Unused area 0.9KB is contained.)
- 4 In case of deleting message and expending another area, it is as same as this example.

I. EDITING FOR Power Mate–MODEL D (PMC–PA1/PA3)

APPENDIX



# I.4 SYSTEM DIAGRAM OF SOFT KEY



APPENDIX

Fig. I.4 (a)



Fig. I.4 (b)

# **APPLICABLE FAPT LADDER EDITIONS**

The following tables list the editions of offline programs required to program each PMC model.

# J.1 FAPT LADDER, FAPT LADDER-II, FAPT LADDER-III, LADDER EDITING PACKAGES FAPT LADDER-III (IBM PC/AT) A08B-9210-J505, LADDER EDITING PACKAGE (WINDOWS) (IBM PC/AT) A08B-9210-J511)

Body of FAPT LADDER	(PC-9801)	A08B-9200-J502#JP
Body of FAPT LADDER	(IBM PC/AT)	A08B-9201-J502#EN
PMC–SA module	(PC-9801)	A08B-9200-J603#JP
PMC–SA module	(IBM PC/AT)	A08B-9201-J603#EN
PMC–SB/SC module	(PC-9801)	A08B-9200-J604#JP
PMC–SB/SC module	(IBM PC/AT)	A08B-9201-J604#EN
PMC–NB module	(PC-9801)	A08B-9200-J606#JP
PMC–NB module	(IBM PC/AT)	A08B-9201-J606#EN
FAPT LADDER–II	(IBM PC/AT)	A08B-9201-J503
Ladder editing package	(IBM PC/AT)	A08B-9201-J510
FAPT LADDER–III	(IBM PC/AT)	A08B-9210-J505
Ladder editing package		
(Windows)	(IBM PC/AT)	A08B-9210-J511

Model Drawing number	PMC- PA1	PMC- PA3	PMC- SA1	PMC- SA2	PMC- SA3	PMC- SA5	PMC- SB	PMC- SB2	PMC- SB3
A08B–9200–J502#JP A08B–9201–J502#EN	5.0 and later	5.0 and later	2.0 and later	2.0 and later	5.0 and later	6.2 and later (Note)	1.0 and later	2.0 and later	5.0 and later
A08B–9200–J603#JP A08B–9201–J603#EN	4.0 and later	4.0 and later	1.0 and later	1.0 and later	4.0 and later	4.2 and later (Note)	_	-	_
A08B–9200–J604#JP A08B–9201–J604#EN	-	-	-	-	-	-	1.0 and later	3.0 and later	4.5 and later
A08B–9201–J503	_	1.0 and later	2.0 and later	-	1.0 and later	2.0 and later	-	-	1.0 and later
A08B–9201–J510	-	-	2.1 and later	-	1.3 and later	2.2 and later	-	-	1.0 and later
A08B–9210–J505	-	1.0 and later	1.0 and later	-	1.0 and later	1.0 and later	-	-	1.0 and later
A08B–9210J511	-	-	1.0 and later	-	1.0 and later	1.0 and later	-	-	1.0 and later

### J. APPLICABLE FAPT LADDER EDITIONS APPENDIX

B-61863E/14

Model Drawing number	PMC- SB4	PMC- SB5	PMC- SB6	PMC- SB7	PMC- SC	PMC- SC3	PMC- SC4	PMC- NB	PMC- NB2	PMC- NB6
A08B–9200–J502#JP A08B–9201–J502#EN	7.1 and later	8.5 and later	8.5 and later	-	1.0 and later	5.0 and later	7.1 and later	6.1 and later	8.5 and later	-
A08B–9200–J603#JP A08B–9201–J603#EN	-	-	-	-	1.0 and later	4.5 and later	5.0 and later	-	-	-
A08B–9200–J604#JP A08B–9201–J604#EN	5.0 and later	7.0 and later	7.0 and later	-	-	-	_	1.0 and later	3.0 and later	-
A08B–9201–J503	1.0 and later	1.1 and later	1.1 and later	-	-	1.0 and later	1.0 and later	1.0 and later	1.0 and later	3.2 and later
A08B–9201–J510	1.0 and later	1.0 and later	1.0 and later	-	-	1.0 and later	1.0 and later	2.1 and later	2.1 and later	3.1 and later
A08B–9210–J505	1.0 and later	1.0 and later	1.0 and later	2.0 and later	-	1.0 and later				
A08B–9210–J511	1.0 and later	1.0 and later	1.0 and later	2.0 and later	_	1.0 and later				

### NOTE

When a PMC–SA5 ladder is to be created in the FAPT LADDER, set the model to PMC–SA3 (RA3).

# J.2 FAPT LADDER (SYSTEM P SERIES)

A08B–0035–J595#E (P–G Mark II): FAPT LADDER PMC–SA1/SA2

### A08B-0036-J595#E (P-G Mate):

Model Edition	PMC- PA1	PMC- PA3	PMC- SA1	PMC- SA2	PMC- SA3
1.1 and later	×	×	0	×	×
2.1 and later	×	×	0	0	Δ
3.1 and later	×	0	×	×	×
4.1 and later	0	0	0	0	Δ

 $\times$ : Not supported,  $\bigcirc$ : Supported,  $\triangle$ : Restrictedly supported (Note)

### NOTE

A sequence program cannot be transferred from the PMC–SA1 of the FANUC Series 20 to the offline programmer (edition 6.0 or an earlier edition). If this is attempted, alarm 89 occurs in the offline programmer.

### A08B–0036–J964 (P–G Mark II and P–G Mate): PMC–SA1/SA2/SB/SC/PA1/PA3 data

Model Edition	PMC- PA1	PMC- PA3	PMC– SA1	PMC- SA2	PMC- SA3
1.1 and later	×	×	0	×	×
2.1 and later	0	0	0	0	×

 $\times$ : Not supported,  $\bigcirc$ : Supported,  $\triangle$ : Restrictedly supported (Note)

### A08B–0035–J595#E (P–G Mark II): FAPT LADDER PMC–SB/SB2/SC

### A08B-0036-J595#E (P-G Mate):

Model Edition	PMC- SB	PMC- SB2	PMC- SB3	PMC- SC	PMC- SC3	PMC- NB
1.1 and later	0	×	×	0	×	х
4.1 and later	0	0	Δ	0	×	×

 $\times$ : Not supported,  $\bigcirc$ : Supported,  $\triangle$ : Restrictedly supported (Note)

### NOTE

The edition of FAPT LADDER adopted for the PMC–SA2 or PMC–SB2 can be used to program the PMC–SA3 or PMC–SB3 as long as some functional instructions including structured programming are not used (as long as FAPT LADDER is used within the range of the specifications of the PMC–SA2 or PMC–SB2).

When this edition is used:

- (1)The following functional instructions cannot be used. (For details, see Section 5 of Part I.)
  - MOVB, MOVW, MOVN
  - DIFU, DIFD
  - AND, OR, NOT, EOR
  - END, CALL, CALLU, SP, SPE
  - JMPB, JMPC, LBL
- (2) A sequence program created by the editing function (ladder editing module) contained in the PMC–SA3/SB3 cannot be edited after it is read into the offline programmer.
- (3) A sequence program created by the offline programmer and transferred to the PMC (sequence program transferred and edited by the built–in editing function) can be edited again after it is read into the offline programmer.



# K.1 OUTLINE OF LEVELED UP CONTENTS

The function is leveled up, that is Input/Output function with Memory Card by CNC or Offline Programmer. The leveled up contents are as follows.

- (1) The time is reduced in Inputing/Outputing between CNC and Memory Card by PMC I/O function. This is the same between Offline Programmer and Memory Card.
- (2) Sequence programs can be inputted from Memory Card by BOOT SYSTEM, by which CNC management software or so can be inputted.

Memory Card function can be used in the following editions of CNC basic software and PMC management software and FAPT LADDER for Personal Computer.

•	CNC	basic	software
---	-----	-------	----------

	non leveled up	leveled up
FANUC Series 20–FA basic software (D001)	05–06	More than 07
FANUC Series 20–TA basic software (D101)	02	More than 03

· PMC management software

	non leveled up	leveled up
PMC–SA1/SA3 management software (4080)	04–05	More than 06

### • FAPT LADDER for Personal Computer

	non lev	eled up	leveled up
FAPT LADDER PMC–SA1/SA2/SB/SB2/SC SYSTEM (A08B–9200–J502#JP (PC–9801)) (A08–9201–J502#EN (IBM PC/AT))	6.1	6.2	More than 6.3
PMC–SA1/SA2 MODULE (PMC–SA1/SA2/SA3/PA1/PA3) (A08B–9200–J603#JP (PC–9801)) (A08–9201–J603#EN (IBM PC/AT))	4.1		More than 4.2

# K.2 OPERATION

### K.2.1 CNC $\rightarrow$ Offline Programmer

- (1) Operation of CNC
  - 1) On PMC I/O screen, specify M–CARD as "DEVICE", WRITE as "FUNCTION", LADDER as "DATA KIND", any file name, which is omissible, as "FILE NO." (See Fig. K.2.1 (a)) and press the soft key [EXEC].

```
PMC I/O PROGRAM MONIT STOP
CHANNEL = 1
DEVICE = M-CARD
FUNCTION = WRITE
DATA KIND = LADDER
FILE NO. =
( #NAME )
[ EXEC ] [CANCEL] [ WRITE ] [ READ ] [COMPAR]
[DELETE] [ LIST ] [FORMAT ] [ ] [SETUP ]
```

Fig. K.2.1 (a) PMC I/O screen

- (2) Operation of Offline Programmer (FAPT LADDER for Personal Computer)
  - 2) Mount a Memory Card interface on the personal computer.
  - 3) Select [INOUT] (I/O) from the main menu.
  - Select [M–CARD] (Memory Card) from the I/O menu. (See Fig. K.2.1 (b))



Fig. K.2.1 (b) I/O Menu screen

Select [READ] (PROGRAMMER ← Memory Card). (See Fig. K.2.1 (c))

/											
	I/	о (м_	CAR	D)	PMC-RA	A1	<0>	[A:¥FLA	DDER¥		]
	F1	KEY	:	WRITE	(PROGRAM	MER ->	Memory	Card)			
	F2	KEY	:	READ	(PROGRAM	IMER <-	Memory	Card)			
	F1	0 КЕҮ	:	END							
	WRITE	READ								END	

### Fig. K.2.1 (c) I/O (I/O M_CARD) screen

- 6) Specify the followings:
  - Name of the Memory Card file
     Specify the name of the file in the Memory Card which is to be converted and the Memory Card drive on which the Memory Card is mounted.
  - Name of the ROM format file to be created Specify a file name to be given to the converted ROM format data.

- //	(75.014)		<b>DVG D</b>		1. T. 1707	100001		,
1/0	) (FROM I	MC)	PMC-RA	T <0	)> [A:¥FI	ADDER¥		1
REA	D (PROG	RAMMER <-	Memory	Card)				
Mer (S <u>r</u>	ory Car ecify t	d FILE NAM he MEMORY	ME : CARD dr:	ive)				
ROI	I FORMAT	FILE NAME	3 :					
EXEC							END	
	I/C REA Men (Sp ROM	I/O (FROM X READ (PROG Memory Car (Specify t ROM FORMAT	I/O (FROM MC) READ (PROGRAMMER < Memory Card FILE NAM (Specify the MEMORY ROM FORMAT FILE NAM)	I/O (FROM MC) PMC-RA READ (PROGRAMMER <- Memory O Memory Card FILE NAME : (Specify the MEMORY CARD dr. ROM FORMAT FILE NAME :	I/O (FROM MC) PMC-RA1 <0 READ (PROGRAMMER <- Memory Card) Memory Card FILE NAME : (Specify the MEMORY CARD drive) ROM FORMAT FILE NAME : EXEC	I/O (FROM MC)     PMC-RA1     <0> [A: ¥FI       READ (PROGRAMMER <- Memory Card)       Memory Card FILE NAME :       (Specify the MEMORY CARD drive)       ROM FORMAT FILE NAME :	I/O (FROM MC)     PMC-RA1     <0> [A:¥FLADDER¥       READ (PROGRAMMER <- Memory Card)     Memory Card FILE NAME : (Specify the MEMORY CARD drive)       ROM FORMAT FILE NAME :	I/O (FROM MC)     PMC-RA1 <o> [A:YFLADDERY       READ (PROGRAMMER &lt;- Memory Card)       Memory Card FILE NAME :       (Specify the MEMORY CARD drive)       ROM FORMAT FILE NAME :</o>

Fig. K.2.1 (d) I/O (FROM MC) screen

7) After it is decompiled, the converted ROM format file can be edited by the personal computer.

# K.2.2 (1) Operation of Offline Programmer (FAPT LADDER for Personal computer) 1) Mount a Memory Card interface on the personal computer. 2) Compile a source program and create a ROM format file. 3) Return to the main menu and select [INOUT] (I/O). 4) From the I/O menu, select {M-CARD] (memory Card). 5) Select [WRITE] (PROGRAMMER → Memory Card). (See Fig. K.2.1(c)).

- 6) Specify the following:
  - Name of the ROM format file
     Specify the name of the ROM format file to be converted.
  - Name of Memory Card file name
     Specify the name to be given to the converted Memory Card file and the Memory Card drive to which the data is output.
     (The file can be accessed by the Memory Card interface incorporated into the CNC.)

I,	(TO MC)	PMC	- RA1	<0>	[A:¥FLADDER}	ſ	]
WI	RITE (PROGR	AMMER -> Me	mory Card	1)			
RC	OM FORMAT	FILE NAME	:				
Me (S	emory Card Specify th	I FILE NAME Le MEMORY CA	: RD drive)	)			
EXEC						END	

Fig. K.2.2 I/O (TO MC) screen

(2) Operation of CNC

There are 2 methods by which the sequence program can be inputted from Memory Card.

- The method of using I/O function of PMC
   On PMC I/O screen, specify M–CARD as "DEVICE", READ as "FUNCTION", the file name or file No. you want to input as "FILE NO." and press the soft key [EXEC].
- The method of using BOOT SYSTEM (When CNC starting up) Refer to K.2.3.





The case of (1), (2), (3) and (4) are explained as follows.

 In case of (1), (2)
 Output operation : There is no special operation
 Input operation : Input sequence programs buy BOOT SYSTEM. (Refer to K.2.3)

- · In case of (3)
  - Output operation : Output sequence programs by setting the output format to 1 (:S–FORMAT) on the following SETUP screen PMC I/O. The default output format is 0 (:BINARY).

```
PMC I/O PROGRAM
                                        MONIT STOP
      CHANNEL
                       1
      DEVICE
                     M-CARD
      FUNCTION
                     WRITE
                  =
      DATA KIND = LADDER
      FILE NO.
        ( #NAME )
  [ EXEC ] [CANCEL] [ WRITE ] [ READ ] [COMPAR]
  [DELETE] [ LIST ] [FORMAT ] [
                                    ][SETUP]
  PMC SETUP M-CARD
                                   MONIT STOP
  OUTPUT FORMAT (PROGRAM) =
                                1
   (0:BINARY, 1:S-FORMAT)
  [ INPUT ] [
                  ][
                           ][
                                   ][ INIT ]
                          Press [INIT] key to reset default value 0.
Input operation
               : Input sequence programs by selecting F6:"I/O"
                  on main menu screen of FAPT LADDER, then
                  F3:"Handy File & Memory Card".
In case of (4)
Output operation :
                  Output sequence programs by selecting F6:"I/O"
                  on main menu screen of FAPT LADDER, then
                  F4:"Handy File".
Input operation : Input sequence programs by selecting F6:"I/O" on
                  main menu screen of FAPT LADDER, then
                  F3:"Handy File & Memory Card".
```

.

# K.3Ladder data can be transferred by using a memory card.NB/NB2 DATA<br/>COMPATIBILITYTwo data formats are used:<br/>- Handy file format<br/>- Memory card formatCOMPATIBILITY- Memory card formatThe handy file format defines the S format data used with RS-232C.<br/>The memory card format defines the binary format data used for a boot.<br/>Data output to a memory card from the I/O screen of an NB of the 4047<br/>series is handy file format data.

Data output to a memory card from the I/O screen of an NB/NB2 of the 4048 series is memory card format data.

### K.3.1 Data Transfer Between NB (4047 Series) and FAPT LADDER



### K.3.2 Data Transfer Between NB/NB2 (4048 Series) and FAPT LADDER



### K.3.3 Data Transfer Between NB (4047 Series) and NB (4048 Series)





# MIGRATION OF LADDER PROGRAMS BETWEEN DIFFERENT MODELS

B–61863E/14 APPENDIX	BETWEEN DIFFERENT MODELS

### L.1 MIGRATION OF LADDER PROGRAMS FROM Power Mate–D/H TO Power Mate *i*–D/H

Differences between the PMC–PA1/PA3 for the Power Mate–D/H and the PMC–SB5/SB6 for the Power Mate *i*–D/H The PMC–SB5/SB6 for the Power Mate *i*–D/H are upward–compatible with the PMC–PA1/PA3 for the Power Mate–D/H, according to their basic specifications. Because, however, sequence programs are not object–compatible with the PMC–PA1/PA3 for the Mate–D/H, Ladder programs sent from the Power Mate–D/H to a memory card cannot be directly loaded into the Power Mate *i*–D/H. For this reason, to migrate from the Power Mate–D/H to the Power Mate *i*–D/H, you must convert the PMC sequence programs.

This section provides information about the PMC programming compatibility that should be noted when you convert sequence programs.

### (1) PMC model change

Sequence programs for one PMC model must be converted to those for the other. These models are Ladder–source compatible with each other, and conversion is possible using the procedure described in the following manual:

Appendix 3.3, "Converting Sequence Programs for One PMC Model to Those for the Other" in the "FAPT LADDER–II Operator's Manual"

(2) Changes in the interface signals between CNC and PMC and between PMC and machine

The F, G, X, and Y address signals have been partially changed. Change the sequence programs, referring to the following manual: "FANUC Power Mate *i*-MODEL D/H Connection Manual (Functions)"

(3) Change in the basic command processing time Because of the increase in command execution time, the following changes may occur:

- Changes in the execution cycle at the second Ladder level
- Changes in the timing of second-level division and first-level execution
- Changes in the timing of Ladder program execution and I/O transfer
- (4) PMC addresses expansion

In the PMC–SB5, the inner relay (R) and the data table (D) have been expanded.

In the PMC–SB6, the inner relay (R), message request signal (A), variable timer (T), counter (C), keep relay (K), data table (D), and subprogram number (P) have been expanded.

### L. MIGRATION OF LADDER PROGRAMS BETWEEN DIFFERENT MODELS

APPENDIX

	Model			
PMC address type	Power Mate-D	Power Mate–D/H Power		Mate <i>i</i> –D/H
	PMC-PA1	PMC-PA3	PMC-SB5	PMC-SB6
Inner relay (R)	R0 to R999 R9000 to R9099	R0 to R999 R9000 to R9117	R0 to R1499 R9000 to R9117	R0 to R2999 R9000 to R9199
Message request signal (A)	A0 to A24			A0 to A124
Variable timer (T)	T0 to T79			T0 to T299
Counter (C)	C0 to C79			C0 to C199
Keep relay (K)	K0 to K19			K0 to K39, K900 to K909
Data table (D)	D0 to D1859		D0 to D2999	D0 to D7999
Subprogram number (P)	_	P1 to P512		P1 to P2000

(5) Function command specification expansion

Expanded specifications have been added to the function commands DECB, NUMEB, XMOVB, and PSGN2. The conventional, basic specifications are still valid and, therefore, sequence programs need not be changed for migration. For an explanation of the expanded specifications, see Section I.5, "PMC Function Commands."

(6) Changes in a window function

The CNC alarm state read function code has been changed from 23 to 186. In addition, the window function has been changed to the low-speed type. For details, see Section B.4.83.

# L.2 MIGRATION FROM THE PMC–NB/NB2 TO THE PMC–NB6

If you previously used the Series 15–B PMC–NB/NB2, you must convert from the PMC–NB/NB2 to the PMC–NB6. The conversion procedure is as described below.

(1)Model change from the PMC–NB/NB2 to the PMC–NB6 (source programs)

When changing the model from the PMC–NB/NB2 to the NB6, you must convert the source programs for one PMC model to those for the other using FAPT LADDER–II. Conversion is possible using the procedure described in the following manual:

Appendix 3.3, "Converting Sequence Programs for One PMC Model to Those for the Other" in the "FAPT LADDER–II Operator's Manual" (B–66184EN)

The mnemonic file format for the PMC–NB6 system parameters used for model conversion is as follows:

%@0			
2 BCD	2. Counter data type	(BINARY or BCD)	
3 NO	3. Whether the operator panel exists (exits: YES, does not exist: NO)		
4 PMC–NB6	4. PMC type	(PMC–NB6)	
7 100	7. Ladder execution time	(100%)	

The specification changes in the NB6, described in Section I.1.2, "Overview of the Ladder Specifications," must be dealt with separately.

(2) When using FAPT LADDER

When creating Ladder programs for the PMC–NB6 using FAPT LADDER, specify use of the PMC–NB2. When PMC–NB2 is specified, however, the following restrictions are imposed:

- a) Do not use the functions supported by the PMC–NB6. (See Section I.1.2, "Overview of the Ladder Specifications" for details.)
- b) Do not use sequence programs with C programs linked.
- (3) If using sequence programs located on the FANUC Series 15-B
  - a) If using sequence programs for the PMC–NB2 that are located on the Series 15–B in the Series 15*i*, the same restrictions as those described in (2) are imposed. Provided that these restrictions are observed, programs for the PMC–NB2 can be output to a memory card (by specifying "LADDER" for DATA KIND on the I/O screen) and directly loaded into the Series 15*i*.
  - b) Sequence programs for the PMC–NB that are located on the Series 15–B cannot be used in the Series 15*i*, without first being converted. They must be converted into programs for the PMC–NB6 with the model change procedure using FAPT LADDER–II, described in (1).
L.3 MIGRATION FROM THE PMC–SA5/ SB5/SB6 TO THE PMC–SB7

When you want to change ladder from PMC– SA5/SB5/SB6 to PMC–SB7, you can convert the source program using FAPT LADDER–III. Please refer to following manual for details.

FAPT LADDER-III OPERATOR'S MANUAL B-66234EN

"10.3 CONVERTING SEQUENCE PROGRAM BETWEEN PMC MODELS"

The sequence of the conversion is as follows.

- i) Convert a source program into the mnemonic file by FAPT LADDER-III.
- ii) Change the system parameters in the mnemonic file for PMC–SB7 by text editor.
- iii) Create a new source program for PMC-SB7 by FAPT LADDER-III.
  - If you want to change the model PMC–SA5/SB5/SB6 to model PMC–SB7, change the item "PMC TYPE" to "PMC–SB7".
  - If the mnemonic file has insufficient parameters for PMC–SB7, the initial values are set with conversion for the source program.
  - The mnemonic file format of the system parameter for PMC–SB7 is as follows.

%@0 2 BINARY 3 NO 4 PMC–SB7 7 150 %

Counter type
 Operator panel
 PMC type
 Ladder execution ratio

(BINARY or BCD) (YES or NO)

n ratio (150%)*

#### NOTE

The default value is 150. The meaning is described in "L.3.1 Addition of LADDER EXEC in the system parameter"

iv) Convert the mnemonic file to the source program and overwrite it on the new source program for PMC–SB7.

## L.3.1 Addition of System Parameter Ladder Execution Time

For the PMC–SB7, with addition of the third ladder level, the ladder execution time is added as a system parameter.

Set the time by which the processing time for the first and second levels is to be increased or decreased.

Initial value 150 Range of valid settings 1 to 15		al value	150	Range of valid settings	1 to 150
---------------------------------------------------	--	----------	-----	-------------------------	----------

- If the ladder uses only the first and second levels, set the upper limit (150).
- If the upper limit (150) is set for a ladder which uses the third level, the third level may not operate satisfactorily. In this case, set this parameter so that the processing time for the first and second levels is decreased.

The processing time for the first and second levels of a ladder is obtained using the following expression:

Processing time for the first and second levels of a ladder =5msec × Ladder execution time 100

The processing time for the third level of a ladder is obtained using the following expression:

Processing time for the third level of a ladder =

APPENDIX

7.5 msec – (processing time for the first and second levels of the ladder)

For example, when this parameter is set to 100, the processing time for the first and second levels of the ladder is 5 ms and that for the third level is 2.5 ms in a cycle of 8 ms.

# L.4 MIGRATION FROM THE SERIES 0 AND SERIES 21–B TO SERIES 0*i*

L.4.1 Compatible With Series 0 The object and the source of the ladder program are not compatible between PMC–SA1/SA3 for Series 0*i* and PMC–L/M for Series 0. Therefore, to apply the ladder program of PMC–L/M for Series 0 to the ladder program of PMC–SA1/SA3 for Series 0*i*, you need to convert from the ladder program of PMC–L/M to the mnemonic form, correct the mnemonic form data, and convert from the mnemonic form data to the ladder program of PMC–SA1/SA3.

- The difference between PMC–L/M for Series 0 and PMC–SA1/SA3 for Series 0*i* and the countermeasure
  - TMR, TMRB, and CTR instruction. PMC–SA1/SA3 are different from PMC–L/M in the number of the parameter in TMR, TMRB and CTR instructions. Therefore, please check the mnemonic form data, and delete the parameter, and set up the timer or the counter. (Example)

Instruction	Before the modification (PMC–L/M)	After the modification (PMC–SA1/SA3)
TMR	RD R400.0 TMR 1 D300 WRT R401.1	RD R400.0 TMR 1 WRT R401.1
TMRB	RD R400.1 SUB 24 2 1000 D310 WRT R401.2	RD R400.1 SUB 24 2 1000 WRT R401.2
CTR	RD.STK R400.2 SUB 5 1 D320 WRT R401.3	RD.STK R400.2 SUB 5 1 WRT R401.3

#### 2. END3(SUB48), PACTL(SUB25) and DISP(SUB49) instruction

Instruction		Countermeasure
END3	(SUB48)	This instruction has not been supported in PMC–SA1/SA3. Please delete the ladder program of third level. Or, please move to second level from third level, and delete END3(SUB48).
PACTL	(SUB25)	This instruction has not been supported in PMC–SA1/SA3. Please delete PACTL(SUB25).
DISP	(SUB49)	This instruction has not been supported in PMC–SA1/SA3. Please change the ladder, and use the DISPB instruction. Please delete DISP(SUB49).

3. The difference of the addresses

APPENDIX

- Converting of the addresses except F/G/X addresses
   Please correct the range of the address when the address which is out of range in PMC–SA1/SA3 is used.
  - (Example) The below address cannot be used in PMC–SA1/ SA3.

D3584–D3839, R3840–R4095 etc.

Converting of F/G/X addresses
 Please refer to the connection manual of Series 0 and Series 0*i*, and convert F/G/X addresses.

Also, FAPT LADDER and FAPT LADDER–II can convert the F/G signals between NC and PMC from Series 0 to Series 0*i*. This conversion has the following limitations.

- (1) The ladder program converted into a mnemonic form is necessary.
- (2) The bit address that is used in basic instruction is converted.
- (3) The byte address that is used in functional instruction is not converted.
- (4) This conversion can avail only the case that the addresses are converted from standard Series 0–T/M to standard Series 0*i*–TA/MA. The F/G addresses of 1000.0 or more are not converted.
- (5) When the same signal name is used in the Series 0 and Series 0i, and the addresses corresponding to the signal in the Series 0 and Series 0*i* have one-to-one relationship, the bit address is converted.

#### NOTE

FAPT LADDER-III cannot convert the address.

[Operation of conversion]

In case of FAPT LADDER (Example: Converting the ladder to PMC–SA1 for Series 0*i*–MA from PMC–M for Series 0M)

- (1) Prepare the mnemonic form file of PMC–M that you want to convert. (*A)
- (2) Input a new source program name in "EDIT" of FAPT LADDER, and select the PMC–RA1(SA1) in "SETUP", and quit the edit screen without edit operation.
- (3) Convert to the mnemonic form file from source program of (2). (*B) The empty mnemonic form file is made.
- (4) Exit FAPT LADDER, and start up the text editor. Specify the mnemonic form file of (3) as the edit file name.
- (5) Replace the symbol data of the mnemonic form file(PMC–SA1) of (3) with the converter file of "FS0T_CNV.SYM" or "FS0M_CNV.SYM". (*C)

The converter file is stored in "¥Appendix" of the installed directory of FAPT LADDER. FS0M–CNV.SYM: For converting to Series 0*i*–MA from Series 0M FS0T–CNV.SYM: For converting to Series 0*i* –TA from Series 0T

- (6) Replace the ladder data of the mnemonic form file(PMC–SA1) of (3) with the ladder data of the mnemonic form file(PMC–M) of (1). (*D)
- (7) Exit the text editor, and start up FAPT LADDER.
- (8) Specify the PMC–RA1(SA1) as PMC model, and convert to the source program from the mnemonic form file of (5).
- (9) Delete all symbols and comment on the "EDIT" screen.



In case of FAPT LADDER–II (Example: Converting the ladder to PMC–SA3 for Series 0*i*–TA from PMC–M for Series 0T)

- (1) Prepare the mnemonic form file of PMC–M that you want to convert. (**A)
- (2) Input the new source program name on the "Program Selection" in FAPT LADDER–II, and select the "PMC–SA3/RA3", and select the "Off–Line function".
- (3) Convert to the mnemonic form file from source program of (2). (**B) Select the "MNEMONIC EDIT", and specify the "SOURCE PROGRAM → MNEMONIC FILE". The empty mnemonic form file is made.
- (4) Exit FAPT LADDER–II, and start up the text editor. Specify the mnemonic form file of (3) as the edit file name.

(5) Replace the symbol data of the mnemonic form file(PMC–SA3) of (3) with the converter file of "FS0T_CNV.SYM" or "FS0M_CNV.SYM". (**C)

The converter file is stored in "\Appendix" of the installed directory or system floppy of FAPT LADDER-II. FS0M-CNV.SYM: For converting to Series 0*i*-MA from Series 0M FS0T-CNV.SYM: For converting to Series 0*i*-TA from Series 0T

- (6) Replace the ladder data of the mnemonic form file(PMC–SA3) of (3) with the ladder data of the mnemonic form file(PMC–M) of (1). (**D)
- (7) Exit the text editor, and start up FAPT LADDER-II.
- (8) Select the PMC–RA1(SA1) as PMC model, and convert to the source program from the mnemonic form file of (5). Select the "MNEMONIC EDIT", and specify the "MNEMONIC FILE  $\rightarrow$  SOURCE PROGRAM".
- (9) Delete all symbols and comment data on the "SYMBOL & COMMENT" of the "EDIT".



L.4.2 Compatibility With Series 21–MODEL B	<ul> <li>The ladder program of PMC–SA1/SA3 for Series 0<i>i</i> has the objective compatibility with the ladder program of PMC–SA1/SA3 for Series 21–B or PMC–SA1/SA5 for Series 21<i>i</i>/210<i>i</i>.</li> <li>Therefore, you can load the ladder program outputted from Series 21–B by using memory card on BOOT screen of Series 0<i>i</i>.</li> <li>However, correction of the ladder program might be needed in the following cases.</li> <li>(1) Ladder program depending on execution time The execution timing might change as follows because the execution time of the instruction is changed.</li> <li>Change in the execution cycle at the second ladder level.</li> <li>Change in the timing of second–level division and first–level execution.</li> <li>Change in the timing of ladder program execution and I/O transfer.</li> </ul>
L.4.3 Limited Function	<ul> <li>The PMC specification of Series 0<i>i</i> is the same as PMC–SA1/SA3 for Series 21–B. However, the following function is not supported.</li> <li>The memory card cannot be used on the I/O Screen of PMC.</li> <li>In other words, the ladder program and the PMC parameter cannot be input or output on the PMC screen with a memory card.</li> <li>Please use the BOOT screen when you input or output the ladder program with a memory card.</li> <li>Please use the FANUC floppy disk cassette, the Handy File or PC(personal computer) when you input or output the PMC parameter.</li> <li>The on–line function of FAPT LADDER–II or FAPT LADDER–III is not supported.</li> </ul>



# M.1 ALARM MESSAGE (PMC-PA1/PA3/SA1/ SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/ SB6/SC/SC3/SC4/NB/ NB2/NB6)

### Alarm messages 1 (alarm screen)

Message	Contents and solution	
ALARM NOTHING	Normal status	
NO ALARM (In case of PMC–NB6)		
ER00 PROGRAM DATA ERROR (ROM)	The sequence program in the ROM is not written correctly. (solution) Please exchange ROM for the sequence program.	
ER01 PROGRAM DATA ERROR (RAM)	The sequence program in the debugging RAM is defective. (solution) Please clear the debugging RAM and input LADDER again. The debugging RAM is not installed though the RAM is selected. (solution) Please install the debugging RAM or install ROM for sequence program and select ROM with K17#3=0.	
ER02 PROGRAM SIZE OVER	The size of a sequence program exceeded the maximum allowable ladder size. (solution) The ordered RAM size is smaller than the option. Contact FA- NUC. Change the value of MAX LADDER AREA SIZE on the SYSPRM screen, then turn the power off then back on (only with PMC–SC).	
ER03 PROGRAM SIZE ERROR (OPTION)	The size of sequence program exceeds the option specification size. (solution) Please increase the option specification size. Or, reduce the size of sequence program.	
ER04 PMC TYPE UNMATCH	The PMC model setting of the sequence program is not corresponding to an actual model. (solution) Please change the PMC model setting by the offline program- mer.	
ER05 PMC MODULE TYPE ERROR	The module type of the PMC engine is not correct. (solution) Please exchange the module of PMC engine for a correct one.	
ER06 PROGRAM MODULE NOTHING	Both ROM for sequence program and the debugging RAM do not exist (PMC–SC only). For a 3–path system, the PMC model must be SB6. (solution) Contact FANUC.	
ER07 NO OPTION (LADDER STEP)	There is no step number option of LADDER.	
ER08 OBJECT UNMATCH	Sequence Program has 2 channels of I/O Link, but PMC control software or Ladder Editor Card or PMC C language control software does not support the I/O Link expansion. (solution) To use only the I/O Link channel 1, use sequence program for only channel 1. To use the I/O Link channel 2, update PMC control software and Ladder Editor Card and PMC C language control software supported the I/O Link expansion.	
ER10 OPTION AREA NOTHING (SERIES–NAME)	The PMC–SB management software is not transferred. (solution) There is a mismatch between the order and delivered the software. Contact FANUC.	
ER11 OPTION AREA NOTHING (SERIES–NAME)	The PMC C language board management software is not transferred. (solution) There is a mismatch between the order and delivered the software. Contact FANUC.	
ER12 OPTION AREA ERROR (SERIES-NAME)	There is a series mismatch between the basic and option of the PMC–SB management software. (solution) Contact FANUC.	

Message	Contents and solution
ER13 OPTION AREA ERROR (SERIES-NAME)	There is a series mismatch between the basic and option of the PMC C language board management software. (solution) Contact FANUC.
ER14 OPTION AREA VERSION ERROR (SERIES–NAME)	There is an edition mismatch between the basic and option of the PMC–SB management software. (solution) Contact FANUC.
ER15 OPTION AREA VERSION ERROR (SERIES–NAME)	There is an edition mismatch between the basic and option of the PMC C language board management software. (solution) Contact FANUC.
ER16 RAM CHECK ERROR (PROGRAM RAM)	The debugging RAM cannot be read/written normally. (solution) Please exchange the debugging RAM.
ER17 PROGRAM PARITY	The parity error occurred on ROM for sequence program or the debugging RAM. (solution) ROM:The deterioration of ROM may be deteriorated Please exchange ROM for the sequence program Please edit the sequence program once on PMC Still the error occurs, exchange the debugging RAM.F-ROM:(PMC-NB/FS-20) 

#### NOTE

1 The PMC–SB3/SC3 for the Series 16 MODEL–B does not support ER00 and ER06.

2 For the PMC–SB3/SC3 for the Series 16 MODEL–B, the "debugging RAM" and "ROM for sequence program," described in the table, are not supported but the relevant descriptions apply to ordinary RAM.

Alarm messages 2 (alarm screen)
---------------------------------

Message	Contents and solution
ER18 PROGRAM DATA ERROR BY I/O	Transferring the sequence program from offline programmer was interrupted by the power off etc. (solution) Please clear the sequence program and transfer the se- quence program again.
ER19 LADDER DATA ERROR	Editing the LADDER was interrupted by the power off or by the switch to the CNC screen by the function key etc. (solution) Please edit LADDER once on PMC. Or, please input LADDER again.
ER20 SYMBOL/ COMMENT DATA ERROR	Editing the symbol and comment was interrupted by the power off or by the switch to the CNC screen by the function key etc. (solution) Please edit symbol and comment once on PMC. Or, please input symbol and comment again.
ER21 MESSAGE DATA ERROR	Editing the message data was interrupted by the power off or the switch to the CNC screen by the function key etc. (solution) Please edit message data once on PMC. Or, please input message data again.
ER22 PROGRAM NOTHING or ER 22 NO PROGRAM(In case of PMC–NB6)	There is no sequence program. (solution) Load a proper program at I/O screen or create program at LAD- DER Diagram Editor screen. After a program is loaded or created, you have to write it to flash ROM not to lose it at power off.
ER23 PLEASE TURN OFF POWER	There is a change in setting LADDER MAX AREA SIZE etc. (solution) Please restart the system to make the change effective.
ER24 LADDER, LANGUAGE AREA OVERLAP	The ladder area overlaps the C language area. (solution) Adjust the C program address range.
ER25 SOFTWARE VERSION ERROR (xx)	<ul> <li>xx=PMCAOPT : The version in the PMC–SB management software does not match.</li> <li>xx=PMCBAS–2 : The version in the PMC–NB6 management software does not match.</li> </ul>
ER26 PMC CONTROL MODULE ERROT (xx)	<ul> <li>xx=PMCAOPT : The PMC-SB management software has not been initialized.</li> <li>xx=PMCBAS-2 : The PMC-NB6 management software has not been initialized.</li> </ul>
ER28 NO OPTION(I/O LINK EXPANSION)	Sequence Program has 2 channels of I/O Link, but the option of I/O Link expansion is not provided. (solution) To use only I/O Link channel 1, use sequence program for only channel 1. To use the I/O Link channel 2, the option of the I/O Link expansion is necessary.
ER29 NO SUPPORT HARDWARE(I/O LINK 2CH)	The I/O Link 2–channel is not supported by the edition of printed circuit board. (solution) To use only the I/O Link channel 1, use sequence program for only channel 1. To use the I/O Link channel 2, exchange main CPU board with one which supports the I/O Link 2– channel.
ER32 NO I/O DEVICE	Any I/O Unit or the connection unit etc. is not connected on each channel. When built–in I/O card is connected, this message is not displayed. (solution) When built–in I/O card is used: Please confirm whether the built–in I/O card is certainly con- nected with. When I/O Link is used: Please confirm whether the DI/DO units turning on. Or please confirm the connection of the cable.
ER33 SLC ERROR or ER33 SLC ERROR(CH2)	The LSI for I/O Link is defective. Display of "CH2" means the defectiveness of I/O Link channel 2. (solution) Please exchange the module of PMC engine.

Message	Contents and solution
ER34 SLC ERROR(xx) or ER34 SLC ERROR(CH2 xx)	The communication with the DI/DO units of the xx group of I/O Link failed. Display of "CH2" means the xx group of I/O Link channel 2. (solution) Please confirm the connection of the cable connected to the DI/DO units of the xx group. Please confirm that the DI/DO units turned on earlier than CNC and PMC. Or, please exchange the main CPU board.
ER35 TOO MUCH OUTPUT DATA IN GROUP (xx) or ER35 TOO MUCH OUTPUT DATA (CH2 xx)	The amount of the output data in the xx group of I/O Link exceeded the max. The data, which exceed 32 bytes, become ineffective. Display of "CH2" means the xx group of I/O Link channel 2. (solution) Please refer to the following for the number of the data for each group. "FANUC I/O Unit–MODEL A connecting and maintenance manual" (B–61813E) "FANUC I/O Unit–MODEL B connecting manual"(B–62163E)
ER36 TOO MUCH INPUT DATA IN GROUP (xx) or ER36 TOO MUCH INPUT DATA (CH2 xx)	The number of the input data in the xx group of I/O Link exceeded the max. The data, which exceed 32 bytes, become ineffective. Display of "CH2" means the xx group of I/O Link channel 2. (solution) Please refer to the following for the number of the data for each group. "FANUC I/O Unit–MODEL A connecting and maintenance manual" (B–61813E) "FANUC I/O Unit–MODEL B connecting manual"(B–62163E)
ER38 MAX SETTING OUTPUT DATA OVER (xx) or ER38 MAX SETTING OUTPUT DATA (CH2 xx)	The assignment data for a group exceeds 128 bytes. (The assignment data of output side of xx group or later become ineffective.) Display of "CH2" means the xx group of I/O Link channel 2. (solution) Please reduce the assignment data to 128 bytes or less for the number of the output data of each group.
ER39 MAX SETTING INPUT DATA OVER (xx) or ER39 MAX SETTING INPUT DATA (CH2 xx)	The assignment data for a group exceeds 128 bytes. (The assignment data of input side of xx group or later become infective.) Display of "CH2" means the xx group of I/O Link channel 2. (solution) Please reduce the assignment data to 128 bytes or less for the number of the input data of each group.
ER48 STEP SEQUENCE TIME OVER (xxH)	Time–out occurred during step sequence processing. For details, see IV–6.2.2.
ER98 ILLEGAL LASER CONNECTION	The I/O unit group for the laser does not match the assignment data. (solution) Make sure that the actual I/O unit configuration matches the assignment data in the ladder.
ER99 X, Y96–127 ARE ALLOCATED	Laser I/O assignments are assigned to X96 to X127 and Y96 to Y127 when the laser I/O link is supported. (solution) Delete I/O assignment data of X96 to X127 and Y96 to Y127.

Alarm messages 3 (alari	m screen)
-------------------------	-----------

Message	Contents and solution
WN01 LADDER MAX SIZE ERROR	The MAX LADDER AREA SIZE in the system parameter is illegal. (solution) Set the correct value to MAX LADDER AREA SIZE and restart the system.
WN02 OPERATE PANEL ADDRESS ERROR	The address setting data of the operator's panel for FS–0 is illegal. (solution) Please correct the address setting data.
WN03 ABORT NC-WINDOW/EXIN	LADDER was stopped while CNC and PMC were communicating. The functional instruction WINDR, WINDW, EXIN, DISPB, and etc. may not work normally. (solution) When restarting the system, this alarm will be released. Execute the sequence program(Press RUN key) after confirm- ing whether there is a problem in LADDER or not.
WN04 UNAVAIL EDIT MODULE	The LADDER editing module cannot be recognized. (PMC–SA1/SA2/SA3/SB/SB2/SB3, except SA1/SA3 for FS–20) (solution) Please confirm the slot position installed. Please confirm the installed module.
WN05 PMC TYPE NO CONVERSION	A PMC–SA3/SA5 ladder was transferred to PMC–SB5. (solution) Correct the ladder type.
WN06 TASK STOPPED BY DEBUG FUNC	Some user tasks are stopped by break point of the debugging function.
WN07 LADDER SP ERROR (STACK)	When functional instruction CALL(SUB65) or CALLU(SUB66) was executed, the stack of the LADDER overflowed. (solution) Please reduce the nesting of the subprogram to 8 or less.
WN09 SEQUENCE PROGRAM IS NOT WRITTEN TO F-ROM	The sequence program has been modified at LADDER Diagram Editor screen, I/O Unit Allocation Editor screenor or I/O screen, and the modified program has not written to flash ROM yet. If CNC's power is turned off before writing the modified program to flash ROM, it will be lost at power off. (solution) If the modified program is expected to run also at next time when CNC's power is turned on, write the modified program to Flash ROM at I/O screen. If the sequence program has been modified by accident, read correct program from flash ROM at I/O screen.
WN17 NO OPTION (LANGUAGE)	There is no C language option.
WN18 ORIGIN ADDRESS ERROR	The LANGUAGE ORIGIN address of the system parameter is wrong (solution) Please set the address of symbol RC_CTLB_INIT in the map file to the LANGUAGE ORIGIN of the system parameter.
WN19 GDT ERROR (BASE, LIMIT)	The value of BASE, LIMIT or ENTRY of user defined GDT is illegal. (solution) Please correct the address in link control statement and build file.
WN20 COMMON MEM. COUNT OVER	The number of common memories exceeds 8. (solution) Please reduce the number of common memories to 8 or less. It is necessary to correct a link control statement,build file and the source file for the common memory.
WN21 COMMON MEM. ENTRY ERROR	GDT ENTRY of the common memory is out of range. (solution) Please correct the address of GDT ENTRY of the common memory in the link control statement.
WN22 LADDER 3 PRIORITY ERROR	The priority of LADDER LEVEL 3 is out of range. (solution) Please correct the value of LADDER LEVEL 3 in the link con- trol statement within the range of 0 or 10–99 or –1.
WN23 TASK COUNT OVER	The number of user tasks exceeds 16. (solution) Please confirm TASK COUNT in the link control statement. When the number of tasks is changed, it is necessary to cor- rect the link control statement, build file and the composition of the files to be linked.
WN24 TASK ENTRY ADDR ERROR	The selector of the entry address to the user task is out of range. (solution) Please correct the table of GDT in build file to the value within 32(20H)–95(5FH).

Message	Contents and solution
WN25 DATA SEG ENTRY ERROR	The entry address of the data segment is out of range. (solution) Please correct DATA SEGMENT GDT ENTRY in the link con- trol statement and the table of GDT in build file within 32(20H)–95(5FH).
WN26 USER TASK PRIORITY ERROR	The priority of the user task is out of range. (solution) Please correct the TASK LEVEL in link control statement within the range of 10–99 or –1. Note: Only one task can have TASK LEVEL –1 (including LAD- DER LEVEL 3).
WN27 CODE SEG TYPE ERROR	The code segment type is illegal. The code segment of RENAMESEG in the binding control file is wrong. (solution) Please correct the entry of the code segment in the link control statement to correspond to the entry in the build file.
WN28 DATA SEG TYPE ERROR	The data segment type is illegal. The data segment of RENAMESEG in the binding control file is wrong. (solution) Please correct the entry of the code segment in the link control statement to correspond to the entry in the build file.
WN29 COMMON MEM SEG TYPE ERROR	The segment type of common memory is illegal. The segment of RENAMESEG in the building control file of the common memory is wrong. (solution) Please correct the entry of common memory in the link control statement to correspond to the entry in the build file.
WN30 IMOPSSIBLE ALLOCATE MEM.	The memories for the data and stack etc. cannot be allocated. (solution) Please confirm whether the value of code segment in build file and USER GDT ADDRESS in link control statement is correct or not. Or please reduce the value of MAX LADDER AREA SIZE of the system parameter and the size of the stack in link control statement at the least.
WN31 IMPOSSIBLE EXECUTE LIBRARY	The library function cannot be executed. (solution) Please confirm the object model of the library. Or, system ROM of PMC must be replaced with one of later ver- sion.
WN32 LNK CONTROL DATA ERROR	Link control statement data is illegal. (solution) Please confirm whether the address of symbol RC_CTLB_INIT in map file is set to LANGUAGE ORIGIN of the system parameter. Or, please make the link control statement again.
WN33 LNK CONTROL VER. ERROR	A link control statement data version error occurred. (solution) Correct the link control statement in the C program.
WN34 LOAD MODULE COUNT OVER	There are more than eight independent load modules. (solution) Reduce the number of independent load modules to eight or less.
WN35 CODE AREA OUT OF RANGE	The code segment area is outside the RAM area. (solution) Check the link map, and place segments within the RAM area.
WN36 LANGUAGE SIZE ERROR (OPTION)	The size of the language area exceeds the option. (solution) Check the free space, and increase the option.
WN37 PROGRAM DATA ERROR (LANG.)	The language program area is invalid. (solution) Install the correct language program.
WN38 RAM CHECK ERROR (LANG.)	A RAM check error occurred in the language program area. (solution) Replace the RAM.
WN39 PROGRAM PARITY (LANG.)	A parity error occurred in the language program area. (solution) Reenter each language program. If an error still occurs, re- place the RAM.
WN40 PROGRAM DATA ERROR BY I/O (LANG.)	Language program read operation was interrupted. (solution) Reenter the language program.

#### M. ALARM MESSAGE LIST

## Alarm messages 4 (alarm screen)

Message		Contents and solution
WN41 LANGUAGE TYPE UNMATCH		There is a C program type mismatch. (solution) Correct the C program.
WN42 UNDEFINE LANGUAGE OF ADDRESS	RIGIN	No language origin address is set. (solution) Set a language origin address.

# M.2 ALARM MESSAGE (PMC–SB7)

Alarm number	Faulty location/corrective action	Contents
ER01 PROGRAM DATA ERROR	<ol> <li>Re-input the sequence program.</li> <li>Replace the master printed circuit board.</li> </ol>	The sequence program is invalid.
ER02 PROGRAM SIZE OVER	<ol> <li>Reduce the sequence program.</li> <li>Contact FANUC to have a larger number-of-Ladder-steps option specified.</li> </ol>	The sequence program is too large. The sequence program is invalid.
ER03 PROGRAM SIZE ERROR (OPTION)	<ol> <li>Reduce the sequence program.</li> <li>Contact FANUC to have a larger number-of-Ladder-steps option specified.</li> </ol>	The sequence program exceeds the size specified by the number–of–Ladder–steps option.
ER04 PMC TYPE UNMATCH	Using an offline programmer, change the sequence program to that for the correct PMC type.	The setting of the type in the sequence program differs from the actual type.
ER06 PMC CONTROL SOFTWARE TYPE UNMATCH	Contact FANUC to specify certain PMC type	The combination of CNC system configuration and PMC type is invalid. (Example: PMC–SB5 is used for a 3–path CNC system.)
ER07 NO OPTION (LADDER STEP)	<ol> <li>Restore the backed up CNC parameter data.</li> <li>Check the data sheet and re-input the CNC parameters.</li> <li>Contact FANUC to specify a number-of-Ladder-steps option of the necessary size.</li> </ol>	No number-of-Ladder-steps option is found.
ER08 OBJECT UNMATCH	1) Contact FANUC.	An unsupported function is used in the sequence program.
ER09 PMC LABEL CHECK ERRORPLEASE TURN ON POWER AGAIN WITH PUSH 'O'&'Z'. (CLEAR PMC SRAM)	<ol> <li>Press and hold down the 'O' and 'Z' key combination, and turn the CNC back on.</li> <li>Replace the backup battery.</li> <li>Replace the master printed circuit board.</li> </ol>	With a change in the PMC type, for example, the retention–type memory of the PMC must be initialized.
ER10 OPTION AREA NOTHING (xxxx)	Contact FANUC to reconfigure the PMC management software.	The PMC management software is not loaded correctly.
ER11 OPTION AREA NOTHING (xxxx)	Contact FANUC to reconfigure the PMC management software.	The PMC C board management software is not loaded correctly.
ER12 OPTION AREA ERROR (xxxx)	Contact FANUC to reconfigure the PMC management software.	The PMC management software is invalid. (The series of BASIC and OPTION do not match.)
ER13 OPTION AREA ERROR (xxxx)	Contact FANUC to reconfigure the PMC management software.	The PMC C board management software is invalid. (The series of BASIC and OPTION do not match.)
ER14 OPTION AREA VERSION ERROR (xxxx)	Contact FANUC to reconfigure the PMC management software.	The PMC management software is invalid. (The editions of BASIC and OPTION do not match.)
ER15 OPTION AREA VERSION ERROR (xxxx)	Contact FANUC to reconfigure the PMC management software.	The PMC C board management software is invalid. (The editions of BASIC and OPTION do not match.)

Alarm number	Faulty location/corrective action	Contents
ER16 RAM CHECK ERROR (PROGRAM RAM)	Replace the master printed circuit board	The initialization of the memory used to store the sequence program failed.
ER17 PROGRAM PARITY	<ol> <li>Re-input the sequence program.</li> <li>Replace the master printed circuit board.</li> </ol>	The parity of the sequence program is invalid.
ER18 PROGRAM DATA ERROR BY I/O	Re–input the sequence program.	While the sequence program was being read, an interrupt command was generated.
ER19 LADDER DATA ERROR	Display the Ladder edit screen again and exit from editing by using the [<<] key.	During Ladder editing, the system was forcibly switched to the CNC screen with a function key.
ER20 SYMBOL/COMMENT DATA ERROR	Display the symbol/comment edit screen again and exit from editing by using the [<<] key.	During symbol/comment editing, the system was forcibly switched to the CNC screen with a function key.
ER21 MESSAGE DATA ERROR	Display the message data edit screen again and exit from editing by using the [<<] key.	During message data editing, the system was forcibly switched to the CNC screen with a function key.
ER22 PROGRAM NOTHING	<ol> <li>Re–input the sequence program.</li> <li>Replace the master printed circuit board.</li> </ol>	The sequence program is empty.
ER23 PLEASE TURN OFF POWER	Turn the CNC off and then back on.	With a change in the PMC type, for example, the power must be turned off and then back on.
ER25 SOFTWARE VERSION ERROR (PMCAOPT)	Contact FANUC to reconfigure the PMC management software.	The PMC management software is invalid. (The edition of PMCAOPT does not match.)
ER26 PMC CONTROL MODULE ERROR (PMCAOPT)	<ol> <li>Contact FANUC to reconfigure the PMC management software.</li> <li>2) 2) Replace the master printed circuit board.</li> </ol>	The initialization of the PMC management software failed.
ER27 LADDER FUNC. PRM IS OUT OF RANGE	Modify the sequence program. Change the parameter number of the function instruction to a value within the valid range.	An out–of–range parameter number is specified with function instruction TMR, TMRB, CTR, DIFU, or DIFD.
ER32 NO I/O DEVICE	<ol> <li>Check that the I/O device is on.</li> <li>Check that the I/O device was turned on before the CNC was turned on.</li> <li>Check the connection of the cable.</li> </ol>	An I/O device such as the I/O Link, connection unit, and Power Mate is not connected.
ER33 I/O LINK ERROR	Replace the master printed circuit board.	The LSI of the I/O Link is defective.
ER34 I/O LINK ERROR (xx)	<ol> <li>Check the connection of the cable leading to a device in group xx.</li> <li>Check that the I/O device was turned on before the CNC.</li> <li>Replace that device in group xx in which the PMC control module is installed.</li> </ol>	In a slave in group xx, an error occurred in communication with an I/O device.
ER35 TOO MUCH OUTPUT DATA IN GROUP (xx)	Reduce the amount of output data in group xx.	The amount of output data in I/O Link group xx exceeds the limit (33 bytes). The excess data is nullified.
ER36 TOO MUCH INPUT DATA IN GROUP (xx)	Reduce the amount of input data in group xx.	The amount of input data in I/O Link group xx exceeds the limit (33 bytes). The excess data is nullified.
ER38 MAX SETTING OUTPUT DATA OVER (xx)	Modify the total amount of output data in each group to 128 bytes or less.	The I/O Link I/O area is insufficient. (The allocation of any group after group xx on the output side is nullified.)

Alarm number	Faulty location/corrective action	Contents
ER39 MAX SETTING INPUT DATA OVER (xx)	Modify the total amount of input data in each group to 128 bytes or less.	The I/O Link I/O area is insufficient. (The allocation of any group after group xx on the input side is nullified.)
ER40 I/O LINK–II SETTING ERROR (CHx)	Reconfigure the I/O Link–II.	The I/O Link–II setting is invalid. (CH1: Primary board, CH2: Secondary board)
ER41 I/O LINK–II MODE ERROR(CHx)	Reconfigure the I/O Link–II.	The I/O Link–II mode setting is invalid. (CH1: Primary board, CH2: Secondary board)
ER42 I/O LINK–II STATION NO.ERROR (CHx)	Reconfigure the I/O Link–II.	The I/O Link–II station number setting is invalid. (CH1: Primary board, CH2: Secondary board)
ER97 IO LINK (CHxyyGROUP)	<ol> <li>Check whether the cables of I/O devices in group yy are connected properly.</li> <li>Check the power to each I/O device.</li> <li>Check the parameter setting of the I/O link assignment data selection function.</li> </ol>	The number of assigned I/O modules in group yy differs from that of I/O devices actually connected.
ER98 ILLEGAL LASER CONNECTION	Modify the allocation of the I/O module.	When an I/O device for a laser is used, the allocation of the I/O module does not match the actual I/O device configuration.
ER99 X,Y96–127 ARE ALLOCATED	Modify the allocation of the I/O module.	When an I/O device for a laser is used, another I/O device is allocated to X96–127/Y96–127. X96–127/Y96–127 are used for I/O devices for a laser, and cannot be used for other devices.
WN02 OPERATE ADDRESS ERROR	Modify the setting of the PMC system parameter, address of the operator's panel for Series 0.	The setting of the PMC system parameter, address of the operator's panel for Series 0, is invalid.
WN03 ABORT NC-WINDOW/EXIN	<ol> <li>Check that the Ladder program is free from problems and then restart the Ladder program (by pressing the RUN key).</li> <li>Turn the CNC off and then back on.</li> </ol>	The Ladder program was stopped during communication between the CNC and PMC.Function instructions such as WINDR, WINDW, EXIN, and DISPB may not be executed normally.
WN05 PMC TYPE NO CONVERSION	Using an offline programmer, change the sequence program to that for the correct PMC type.	The setting of the type in the sequence program differs from the actual type. (Example: For the PMC–SB5, the Ladder program of the PMC–SA3/SA5 was transferred.)
WN06 TASK STOPPED BY DEBUG FUNC	To restart a user task that has been stopped, stop the sequence program and then execute it again.	When a PMC C board is used, a user task has been stopped due to a break by a debug function.
WN07 LADDER SP ERROR (STACK)	Modify the sequence program so that the subprogram nesting level is eight or less.	For a subprogram call with the function instruction CALL or CALLU, the nesting level is too deep (exceeds 8).
WN17 NO OPTION (LANGUAGE)	<ol> <li>Restore the backed up parameter data.</li> <li>Check the data sheet and re-input the parameters.</li> <li>Contact FANUC to specify a PMC C program option of the necessary size.</li> </ol>	When a PMC C board is used, no PMC C program option is found.

Alarm number	Faulty location/corrective action	Contents
WN18 ORIGIN ADDRESS ERROR	<ol> <li>On the PMC system parameter screen, press [ORIGIN].</li> <li>Set the PMC system parameter, LANGUGE ORIGIN, to the address indicated by the RC_CTLB_INIT in the map file.</li> </ol>	When a PMC C board is used, the PMC system parameter, LANGUAGE ORIGIN, is invalid.
WN19 GDT ERROR (BASE,LIMIT)	Modify the setting in the user-defined GDT in the link control statement or build file.	When a PMC C board is used, the BASELIMIT or ENTRY in the user-defined GDT is invalid.
WN20 COMMON MEM. COUNT OVER	Change the number of shared memories to eight or less.Modify the link control statement, build file, or other source files for shared memories.	When a PMC C board is used, the number of shared memories exceeds eight.
WN21 COMMON MEM. ENTRY ERROR	Modify the ENTRY in the shared memory GDT in the link control statement.	When a PMC C board is used, the ENTRY in the shared memory GDT is out of range.
WN22 LADDER 3 PRIORITY ERROR	Change the value of the TASK LEVEL (LADDER LEVEL 3) in the link control statement to 0, 10 to 99, or -1.	When a PMC C board is used, the priority of LADDER LEVEL 3 is out of range.
WN23 TASK COUNT OVER	Change the TASK COUNT in the link control statement to 16 or less. (To change the task count, modify the link control statement, build file, and the configuration of the files to be linked.)	When a PMC C board is used, the number of user tasks exceeds 16.
WN24 TASK ENTRY ADDR ERROR	Change the GDT table in the build file to 32 (20H) to 95 (5FH)	When a PMC C board is used, the user task entry address selector is out of range.
WN25 DATA SEG ENTRY ERROR	Change the DATA SEGMENT GDT ENTRY value in the link control statement and the GDT table in the build file to 32 (20H) to 95 (5FH).	When a PMC C board is used, the data segment entry address is out of range.
WN26 USER TASK PRIORITY ERROR	Change the TASK LEVEL of each task in the link control statement to a value from 10 to 99 or -1. (Note that -1 can be specified for the TASK LEVEL of only one task, including the third Ladder level)	When a PMC C board is used, the priority of the user task is out of range.
WN27 CODE SEG TYPE ERROR	Change the value of the code segment according to the segment setting in the link control statement and build file.	When a PMC C board is used, the code segment type is invalid. The setting of the RENA–MESEG code segment in the bind control file is wrong.
WN28 DATA SEG TYPE ERROR	Change the value of the data segment according to the segment setting in the link control statement and build file.	When a PMC C board is used, the data segment type is invalid. The setting of the RENA–MESEG data segment in the bind control file is wrong.
WN29 COMMON MEM SEG TYPE ERROR	Change the value according to the segment setting in the link control statement and build file.	When a PMC C board is used, the shared memory segment type is invalid. The setting of the RENAMESEG segment in the shared memory bind control file is wrong.

Alarm number	Faulty location/corrective action	Contents
WN30 IMPOSSIBLE ALLOCATE MEM.	<ol> <li>Check that the USER GDT ADDRESS in the link control statement and the start address of the code segment in the build file are correct.</li> </ol>	When a PMC C board is used, the memory area for data, stacks, and others cannot be reserved.
	<ol> <li>Change the PMC system parameter, MAX LADDER AREA SIZE, to a minimum.</li> <li>Change the stack size in the link control state to a minimum.</li> </ol>	
WN31 IMPOSSIBLE EXECUTE LIBRARY	<ol> <li>Check the types supported by the library.</li> <li>Reconfigure the PMC management software and contact FANUC.</li> </ol>	When a PMC C board is used, library functions cannot be executed.
WN32 LNK CONTROL DATA ERROR	<ol> <li>Check that the address of RC_CTLNB_INIT is set for the PMC system parameter, LANGUAGE ORIGIN.</li> <li>Create the link control statement again.</li> </ol>	When a PMC C board is used, link control statement (program control) data is invalid.
WN33 LNK CONTROL VER.ERROR	Modify the link control statement in the PMC C program.	When a PMC C board is used, a link control statement data edition error has occurred.
WN34 LOAD MODULE COUNT OVER	Change the number of independent load modules to eight or less.	When a PMC C board is used, the number of independent load modules exceeds eight.
WN35 CODE AREA OUT OF RANGE	Check the link map and allocate segments within the range of RAM.	When a PMC C board is used, the code segment area is out of the range of the RAM.
WN36 LANGUAGE SIZE ERROR (OPTION)	<ol> <li>Reduce the PMC C program.</li> <li>Contact FANUC to specify a PMC C program option of a larger size.</li> </ol>	When a PMC C board is used, the PMC C program exceeds the size specified for the PMC C program option.
WN37 PROGRAM DATA ERROR (LANG.)	Initialize the PMC C program memory. ([EDIT] $\rightarrow$ [CLEAR] $\rightarrow$ [CLRLNG] $\rightarrow$ [EXEC])	The PMC C program memory must be initialized.
WN38 RAM CHECK ERROR (LANG.)	Replace the master printed circuit board.	The initialization of the PMC C program memory failed.
WN39 PROGRAM PARITY (LANG.)	<ol> <li>Re–input the PMC C program.</li> <li>Replace the master printed circuit board.</li> </ol>	The parity of the PMC C program parity is invalid.
WN40 PROGRAM DATA ERROR BY I/O (LANG.)	Re-input the language program.	While the PMC C program was being read, an interrupt command was generated.
WN41 LANGUAGE TYPE UNMATCH	<ol> <li>Re–input the PMC C program.</li> <li>Replace the master printed circuit board.</li> </ol>	When a PMC C board is used, an unusable C program is input.
WN42 UNDEFINE LANGUAGE ORIGIN ADDRESS	<ol> <li>On the PMC system parameter screen, click [ORIGIN].</li> <li>Set the PMC system parameter, LANGUGE ORIGIN, to the address indicated by the RC_CTLB_INIT in the map file.</li> </ol>	When a PMC C board is used, the PMC parameter, LANGUAGE ORIGIN, is not set.
WN48 UNAVAIL LANGUAGE BY CNC UNMATCH	Remove the PMC C board.	A PMC C board is installed in a CNC in which a PMC C board cannot be used.

# M.3 SYSTEM ALARM MESSAGES (PMC-SC)

Message STATUS LED	Contents and solution
PC1nn CPU INTERPT xxxx yyyyyy	<ul> <li>A CPU error (abnormal interrupt) occurred.</li> <li>nn : CPU exception handling code         <ul> <li>It is an exception code of i80386. For details, please refer to             the manual of the CPU.</li> <li>00 Division error such as a divisor is 0 in division instruction.</li> <li>12 Stack exception such as violations of limit of stack segment.</li> <li>13 General protection exception such as segment limit over.</li> </ul> </li> <li>xxxx : Segment selector where system error occurred.         <ul> <li>The selector of 0103–02FB is used by C language.</li> <li>vvvvvv: Offset address where system error occurred.</li> </ul> </li> </ul>
PC130 RAM PARITY aa xxxx yyyyyy	The parity error occurred on the debugging RAM of PMC.
STATUS LED 🛛 🛨	aa : RAM PARTY ERROR Information. xxxx : Segment selector where system error occurred. yyyyyy: Offset address where system error occurred.
PC140 NMI BOC bb xxxx yyyyyy	The RAM parity error or NMI(Non Maskable Interrupt) generated in module of PMC engine.bb: RAM PARITY ERROR information.1, 2, 4, 8Parity error occurred on basic DRAM.14, 18Parity error occurred on option DRAM.20, 60, A0, E0Parity error occurred on SRAM.xxxx: Segment selector where system error occurred.
STATUS LED *	yyyyyy: Offset address where system error occurred.
PC150 NMI SLC aa cc	<ul> <li>The communication error occurred in the I/O Link.</li> <li>aa, cc : I/O Link error information.</li> <li>This error may occur by the following causes.</li> <li>1. When I/O Unit–MODEL A is used, base1, 2 or 3 is not connected though allocated.</li> <li>2. The connection of cable is insufficient.</li> <li>3. Defects of cable.</li> <li>4. Defects of DI/DO units (I/O unit, Power Mate etc.)</li> <li>5. Defects of PMC board (printed circuit board on host side where I/O Link cable is connected.)</li> <li>(solution) Investigate the cause of error.</li> <li>1. Please confirm the allocation data (by "EDIT"→"MODULE" screen) and compare with the actual connection.</li> <li>2. Please confirm whether the cable is correctly connected. If you cannot find the cause with the ways above, it may be the defect of hardware.</li> <li>Please investigate a defective place by the following methods.</li> <li>3. Please confirm the specification of the cable referring to "FANUC I/O Unit–MODEL B connecting manuals(B–62163E)".</li> <li>4. Exchange the interface module of I/O Unit, the cable and the PMC board, etc. one by one and, confirm whether this error occurs again. The communication may fail by the noise etc. when this error still occurs after replacing all DI/DO units.</li> </ul>
STATUS LED ■★	Please investigate the cause of noise.
PC160 F–BUS ERROR XXXX: yyyyyyyy PC161 F–BUS ERROR XXXX: yyyyyyyy PC162 F–BUS ERROR XXXX: yyyyyyyy STATUS LED ★□	A bus error occurred on the PMC. xxxx : Segment selector for which a bus error occurred. yyyyyyyy : Offset address where a bus error occurred.
PC170 L–BUS ERROR xxxx:yyyyyyyy PC171 L–BUS ERROR xxxx:yyyyyyyy PC172 L–BUS ERROR xxxx:yyyyyyyy STATUS LED ★□	A bus error occurred on the PMC. xxxx : Segment selector for which a bus error occurred. yyyyyyyy : Offset address where a bus error occurred.
PC199 ROM PARITY eeeeeeee STATUS LED ★☆	The parity error occur in PMC system ROM. eeeeeeee : ROM parity error information.

STATUS LED (green) are LED1, LED2 on PMC–RC. CAP–II is LED3 and LED4.

 $\Box$ : Off  $\blacksquare$ : On  $\Rightarrow \bigstar$ : Blinking

#### NOTE

The system error on PMC–SA1,SA2,SA3,SB,SB2 and SB3 is displayed as a system error on the CNC side. (Refer to the "FANUC Series 16–MA Operator's Manual (B–61874E)" and "FANUC Series 16–TA Operator's Manual (B–61804E)".)

# M.4 SYSTEM ALARM MESSAGES (C LANGUAGE FUNCTION FOR PMC-NB/NB2/NB6)

Message STATUS LED	Contents and solution
RAM ERROR <a> bbcc xxxx: yyyyyyyy: PC010 STATUS LED ★■ or □★</a>	The parity error occurs on the debugging RAM of PMC. a : RAM which generates RAM parity. B BASIC RAM O OPTION RAM S STATIC RAM bb, cc : RAM PARITY information. xxxx : Segment selector where system error occurred. yyyyyyyy : Offset address where system error occurred.
ROM ERROR aaaaaaaa: PC020 STATUS LED     ☆★	The parity error occurs in PMC system ROM. aaaaaaaa : ROM parity information
DIVIDE ERROR xxxx: yyyyyyyy: PC040 STATUS LED      ★★	Division error occurs such as a divisor is 0 in the division instruction. xxxx : Segment selector where system error occurred. yyyyyyyy : Offset address where system error occurred.
BUS ERROR xxxx: уууууууу: PC040 STATUS LED      ☆★	The BUS error (access on illegal address). xxxx : Segment selector where system error occurred. yyyyyyyy : Offset address where system error occurred.
STACK FAULT xxxx: уууууууу: PC040 STATUS LED      ★★	The stack exception such as the violation of the limit of the stack. xxxx : Segment selector where system error occurred. yyyyyyyy : Offset address where system error occurred.
GENERAL PROTECTION XXXX: yyyyyyyy: PC040	The general protection exception such as segment limit over was generated. xxxx : Segment selector where system error occurred.
SLC ERROR aa (cc) : PC050	<ul> <li>The communication error occurred in the I/O Link.</li> <li>aa, cc : I/O Link error information.</li> <li>This error may occur by the following causes.</li> <li>1. When I/O Unit–MODEL A is used, base1, 2 or 3 is not connected though allocated.</li> <li>2. The connection of cable is insufficient.</li> <li>3. Defects of cable.</li> <li>4. Defects of DI/DO units (I/O unit, Power Mate etc.)</li> <li>5. Defects of PMC board (printed circuit board on host side where I/O Link cable is connected.)</li> <li>(solution) Investigate the cause of error.</li> <li>1. Please confirm the allocation data (by "EDIT"→"MODULE" screen) and compare with the actual connection.</li> <li>2. Please confirm the the cable is correctly connected. If you cannot find the cause with the ways above, it may be the defect of hardware.</li> <li>Please investigate a defective place by the following methods.</li> <li>3. Please confirm the specification of the cable referring to "FANUC I/O Unit–MODEL B connecting manuals(B–62163E)".</li> <li>4. Exchange the interface module of I/O Unit, the cable and the PMC board, etc. one by one and, confirm whether this error occurs again. The communication may fail by the noise etc. when this error still occurs after replacing all DI/DO units.</li> </ul>

STATUS LED (green) are LED1, LED2 on PMC-NB.

 $\Box$ : Off  $\blacksquare$ : On  $\Rightarrow \bigstar$ : Blinking

# M.5 SYSTEM ALARM MESSAGES (PMC-SB5/SB6/NB6)

Message		Contents and solution	
PC0nn CPU INTERRUPT	xxxxxxx	CPU error	
		nn :Exception code xxxxxxxx :Address at which an error occurred	
PC004 CPU ERR PC006 CPU ERR PC009 CPU ERR	xxxxxxxx:yyyyyyyy xxxxxxxx:yyyyyyyy xxxxxxxx	A CPU error occurred on the PMC. xxxxxxx and yyyyyyy represent an internal error code.	
PC010 CPU ERR	xxxxxxxx:yyyyyyyy	This error may be caused by a main board failure.	
		(solution) Replace the main board, then check if this error occurs again. If this error still occurs, contact FANUC with the error status information (system con- figuration, operation, error occurrence timing, error occurrence frequen- cy, and so forth).	
PC030 RAM PARITY	aa:bb	A RAM parity error occurred on the PMC. aa and bb represent an internal error code.	
		This error may be caused by a main board failure.	
		(solution) Replace the main board, then check if this error occurs again. If this error still occurs, contact FANUC with the error status information (system con- figuration, operation, error occurrence timing, error occurrence frequen- cy, and so forth) and the internal error code above.	
PC040 NMI BOC	xxxxxxx	A non-maskable interrupt (NMI) occurred in the PMC control module.	
		xxxxxxxx :Address at which an error occurred	
PC050 NMI SLC (CHn) a0 or PC050 I/O Link (CHn) a0:c	:d0 a1:d1 d0 a1:d1	<ul> <li>A communication error occurred on the I/O Link channel n. a0 and d0 represent an internal error code for the I/O Link channel 1. a1 and d1 represent an internal error code for the I/O Link channel 2.</li> <li>This error may occur as a result of one of the followings:</li> <li>(1) When I/O Unit A is used, an I/O assignment is made for base expansion, but no base is connected.</li> <li>(2) Cables are not connected correctly.</li> <li>(3) Cables are faulty.</li> <li>(4) I/O devices (I/O Unit, Power Mate, and so forth) are faulty.</li> <li>(5) Power failure of Master or Slave device on I/O Link</li> <li>(6) Short circuit of DO terminal on I/O device</li> <li>(7) The main board is faulty.</li> <li>(solution)</li> <li>(1) Check if the I/O assignment data matches the actual connections of the I/O devices.</li> <li>(2) Check if the cables are connected correctly.</li> <li>(3) Check the cables are connected correctly.</li> <li>(3) Check the cables are connected correctly.</li> <li>(4) Replace the interface module, cables, and main board of the I/O Unit-MODEL B Connection Manual (B–61813E)" or "FANUC I/O Unit-MODEL B Connection Manual (B–61813E)" or "FANUC I/O Unit-MODEL B Connection Manual (B–6163E).</li> <li>(4) Replace the interface module, cables, and main board of the I/O Unit, then check whether this error occurs again.</li> <li>(5) Check the connection of DO terminal line.</li> <li>(6) Check the connection of DO terminal line.</li> <li>(7) If this error still occurs after replacement of all the devices related to the I/O Link according to Action (4), the communication error may have been caused by noise. Attempt to identify the source of the noise.</li> </ul>	

#### M. ALARM MESSAGE LIST

Message		Contents and solution	
PC060 FBUS PC061 FL-R PC062 FL-W	xxxxxxxx:yyyyyyyyy xxxxxxxx:yyyyyyyyy aa:xxxxxxxx:yyyyyyyyy	A bus error occurred on the PMC. aa, xxxxxxx, and yyyyyyy represent an internal error code. This error may be caused by a hardware failure.	
		(solution) Contact FANUC with the error status information (system configuration, operation, error occurrence timing, error occurrence frequency, and so forth), the internal error code above, and the LED statuses on each board.	
PC070 SUB65 CALL (	STACK)	A stack error occurred with the ladder functional instruction CALL/CAL-LU.	
		(solution) Check the correspondence between the CALL/CALLU instruction and the SPE instruction. If the cause of the fault cannot be found, contact FA- NUC with the error status information and the ladder program.	
PC080 SYS EMG PC081 FL EMG	xxxxxxxx:yyyyyyyy xxxxxxx:yyyyyyyyy	A system alarm was issued due to other software.	
		(solution) Contact FANUC with the error status information (system configuration, operation, error occurrence timing, error occurrence frequency, and so forth), the internal error code above, and the LED statuses on each board.	
PC097 PARITY ERR ( PC098 PARITY ERR (	LADDER) DRAM)	A parity error occurred on the PMC system.	
PC099 PARITY ERR (	SRAM)	This error may be caused by a main board failure.	
		(solution) Replace the main board, then check whether this error occurs again. If this error still occurs, contact FANUC with the error status information (system configuration, operation, error occurrence timing, error occur- rence frequency, and so forth).	

# M.6 PMC SYSTEM ALARM MESSAGES (PMC-SB7)

Alarm number	Faulty location/corrective action	Contents
PC004 CPU ERR xxxxxxx:yyyyyyy PC006 CPU ERR xxxxxxx:yyyyyyy PC009 CPU ERR xxxxxxx:yyyyyyyy PC010 CPU ERR xxxxxxx:yyyyyyyy PC012 CPU ERR xxxxxxx:yyyyyyyyy	<ol> <li>Replace the master printed circuit board.</li> <li>If the error recurs even after the replacement, contact FANUC to report the status (displayed message, system configuration, operation, when the error occurs, occurrence frequency, etc.)</li> </ol>	A CPU error occurred in the PMC.xxxxxxx and yyyyyyyyy are internal error codes.
PC030 RAM PARITY aa:bb	<ol> <li>Replace the master printed circuit board.</li> <li>If the error recurs even after the replacement, contact FANUC to report the status (displayed message, system configuration, operation, when the error occurs, occurrence frequency, etc.) and the above internal error codes.</li> </ol>	A RAM parity error occurred in the PMC.aa and bb are internal error codes.
PC050 NMI SLC aa:bb PC050 I/O LINK(CH_) aa:bb aa:bb PC050 IOLINK CH_ aaaa- bbbb:cccc	<ol> <li>Check that the I/O allocation data matches the actual I/O device connection.</li> <li>Check that the cable is connected properly.</li> <li>Check the cable specifications.</li> <li>Replace the I/O device interface module, cable, master printed circuit board, etc.</li> </ol>	<ul> <li>A communication error occurred in the I/O LINK.</li> <li>aa, bb and cc are internal error codes. If this alarm is generated, probable causes include the following:</li> <li>1) Although the base expansion is assigned when the I/O Unit A is used, the base is not connected.</li> <li>2) A cable is not connected securely.</li> <li>3) Cabling is faulty.</li> <li>4) I/O equipment (I/O unit, Power Mate, etc.) is faulty.</li> <li>5) Power failure of Master or Slave device on I/O Link.</li> <li>6) Short circuit of DO terminal on I/O device.</li> <li>7) The motherboard is faulty.</li> </ul>
PC060 FBUS xxxxxxx:yyyyyyyy PC061 FL–R xxxxxxx:yyyyyyyy	<ol> <li>Replace the master printed circuit board.</li> <li>If the error recurs even after the replacement, contact FANUC to report the status (displayed message, system configuration, operation, when the error occurs, occurrence frequency, etc.)</li> </ol>	A bus error occurred in the PMC.
PC070 SUB65 CALL (STACK)	Check the correspondence between the CALL/CALLU and SPE instructions	A stack error occurred in Ladder function instruction CALL/CALLU.
PC090 NMI() xxxxxxx:yyyyyyyy	<ol> <li>Replace the master printed circuit board.</li> <li>If the error recurs even after the replacement, contact FANUC to report the status (displayed message, system configuration, operation, when the error occurs, occurrence frequency, etc.)</li> </ol>	An NMI with an unknown cause occurred in the PMC management software.

Alarm number	Faulty location/corrective action	Contents
PC092 USER TRAP aa:xxxxxxx	<ol> <li>Replace the master printed circuit board.</li> <li>If the error recurs even after the replacement, contact FANUC to report the status (displayed message, system configuration, operation, when the error occurs, occurrence frequency, etc.)</li> </ol>	The TRAP instruction, which is not used in the PMC management software, was executed.
PC093 INT(SYS) xxxxxxx:yyyyyyy PC094 INT(TRAP) xxxxxxx:yyyyyyyy PC095 INT(EX) xxxxxxx:yyyyyyyy PC096 INT(IN) xxxxxxx:yyyyyyyyy	<ol> <li>Replace the master printed circuit board.</li> <li>If the error recurs even after the replacement, contact FANUC to report the status (displayed message, system configuration, operation, when the error occurs, occurrence frequency, etc.)</li> </ol>	An interrupt with an unknown cause occurred in the PMC management software.
PC087 PARITY ERR (LADDER–2) PC097 PARITY ERR (LADDER) PC098 PARITY ERR (DRAM)	<ol> <li>Replace the master printed circuit board.</li> <li>If the error recurs even after the replacement, contact FANUC to report the status (displayed message, system configuration, operation, when the error occurs, occurrence frequency, etc.).</li> </ol>	An error occurred in a RAM check.
PC501 NC/PMC INTERFACE ERR PATH_	<ol> <li>Replace the master printed circuit board.</li> <li>If the error recurs even after the replacement, contact FANUC to report the status (displayed message, system configuration, operation, when the error occurs, occurrence frequency, etc.)</li> </ol>	The reading/writing of signals between the CNC and the PMC failed.
PC502 ILLEGAL FUNCTION (SUB xx)	Modify the sequence program so that instruction function xx is not used.	Unsupported function instruction xx is used.



# **EXAMPLE OF STEP SEQUENCE PROGRAMS**

ſ

CNC	<u>]</u>	Connection Unit.		
		I/O Unit		
		X0.0:Safety switch Y1.0:Ready end#1 Y1.1:Drive start#1 Y2.0:Ready end#2 Y2.1:Drive start#2 Y3.0:Ready end#3 Y3.1:Drive start#3		
		Power Mate #1 X100.0:Readyend#1 Y100.0:Emergency#1 X100.1:Driving#1 Y100.1:Drive start#1		
		Power Mate #2 X110.0:Ready end#2 Y110.0:Emergency#2 X110.1:Driving#2 Y110.1:Drive start#2		
		Power Mate #3 X120.0:Readyend#3 Y120.0:Emergency#3 X120.1:Driving#3 Y120.1:Drive start#3		

#### The CNC is connected two or three Power Mate units.

## NOTE

The addresses indicate the single addresses, as viewed from the CNC.

The CNC controls the Power Mate units at the following signal timing.

Ready end #1 (Power Mate to CNC)		
Drive start #1 (CNC to Power Mate)	1sec	1sec
Driving #1 (Power Mate to CNC)	1sec	



The following flowchart illustrates the interface with the Power Mate units.

The interface with the Power Mate units is changed to the Step Sequence program.



### Example 1

The Step sequence program for three sequentially driven Power Mate units:







— 1444 —

B-61863E/14



# O.1 WHILE STATEMENT

The operation is continued while the condition is true.

Format



## Examples



# O.2<br/>DO-WHILE<br/>STATEMENTThe operation is continued while the condition is true after executing the<br/>operation.<br/>The difference between do-while and while is that the operation is<br/>executed at least one time.Format✓ L1



## Examples


### O.3 FOR STATEMENT

After the initial data is set, the operation is continued while the condition is true.

### Format



### Examples



APPENDIX

### O.4 IF ELSE STATEMENT

If the condition is true, the operation 1 is executed and if the condition is false, the operation 2 is executed.

### Format

condition	
The condition is true.	The condition is false.

### Examples



APPENDIX

### O.5 SWITCH STATEMENT

The operation connected to the condition is executed.

Format



### Examples



### O. STEP SEQUENCE CORRESPONDED C LANGUAGE

APPENDIX





# CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST

Pronun	119	Shift							Seg	gment	and po	oint						
ciation	515	JIS	0	1	2	3	4	5	6	7	8	9	А	В	С	D	Е	F
alpha	2330	824F	0	1	2	3	4	5	6	7	8	9						
nume	2340	825F		Α	В	С	D	$\mathbf{E}$	$\mathbf{F}$	G	Η	Ι	J	Κ	$\mathbf{L}$	Μ	Ν	0
ric	2350	826F	Р	$\mathbf{Q}$	$\mathbf{R}$	$\mathbf{S}$	Т	U	V	W	Х	Y	$\mathbf{Z}$					
×	2360	8280		a	b	с	d	е	f	g	h	i	j	k	1	m	n	0
	2370	8290	р	q	r	s	t	u	v	w	x	У	Z					
hi	2420	829E		あ	あ	い	い	う	う	え	え	お	お	か	が	き	ぎ	<
ra	2430	82AE	ぐ	け	げ	Ľ	Ľ	さ	ざ	L	じ	す	ず	せ	ぜ	そ	ぞ	た
ga	2440	82BE	だ	ち	ぢ	っ	っ	づ	τ	で	と	ど	な	に	ぬ	ね	の	は
na	2450	82CE	ば	ぱ	ひ	び	ぴ	ふ	ぶ	ぷ	$\sim$	べ	ペ	ほ	ぼ	ぽ	ま	み
	2460	82DE	む	め	も	や	や	Ŵ	ゆ	よ	よ	5	り	る	れ	ろ	わ	わ
	2470	82EE			を	h												
ka	2520	833F		7	ア	1	イ	ウ	ウ	I	I	オ	才	カ	ガ	キ	ギ	ク
ta	2530	834F	グ	ケ	ゲ	Ц	ゴ	サ	ザ	シ	ジ	ス	ズ	セ	ゼ	ソ	ゾ	タ
ka na	2540	835F	ダ	チ	ヂ	ッ	ツ	ヅ	テ	デ	ト	ド	ナ	Ξ	ヌ	ネ	ノ	ハ
	2550	836F	バ	パ	ヒ	ビ	ピ	フ	ブ	プ	$\sim$	ベ	ペ	ホ	ボ	ポ	マ	Ξ
	2560	8380	ム	メ	モ	ヤ	ヤ	ユ	ユ	Е	Ξ	ラ	リ	ル	${\cal V}$	П	ワ	ワ
	2570	8390			ヲ	ン	ヴ	カ	ケ									
G	2620	839E					Δ											
r	2630	83AE			Σ						Ω							
e	2640	83BE		α	β	γ		ε			$\theta$				$\mu$			
e k	2650	83CE	π					$\phi$			ω							
n n																		
*																		

hThe characters with mark h cannot be displayed on FANUC Series 16/18-MODEL A.

### P. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST APPENDIX

B-61863E/14

Pronun		Shift							Seg	gment	and p	oint						
ciation	313	JIS	0	1	2	3	4	5	6	7	8	9	A	В	С	D	Е	F
T	3020	889E		亜	唖	娃	阿	哀	愛	挨	姶	逢	葵	茜	穐	悪	握	渥
	3030	88AE	旭	葦	芦	鯵	梓	圧	斡	扱	宛	妞	虻	飴	絢	綾	鮎	或
	3040	88BE	粟	袷	安	庵	按	暗	案	罶	鞍	杏						
イ	3040	88BE											以	伊	位	依	偉	荓
	3050	88CE	夷	委	威	尉	惟	意	慰	易	椅	為	畏	異	移	維	緯	冐
	3060	88DE	萎	衣	謂	違	遺	医	井	亥	域	育	郁	磯		壱	溢	逸
	3070	88EE	稲	茨	芋	鰯	允	印	昞	員	因	姻	引	飲	淫	胤	蔭	
	3120	893F		院	陰	隠	韻	吋										
ウ	3120	893F							右	宇	烏	羽	迂	雨	卯	鵜	窺	丑
	3130	894F	碓	臼	渦	嘘	唄	科科	蔚	鰻	姥	厩	浦	瓜	閠	噂	궃	運
	3140	895F	雲															
I	3140	895F		荏	餌	叡	営	嬰	影	眏	曳	栄	永	泳	洩	瑛	盈	穎
	3150	896F	頴	英	衛	詠	鋭	液	疫	益	駅	悦	謁	越	閲	榎	厭	Щ
	3160	8980	袁	堰	奄	宴	延	怨	掩	援	沿	演	炎	焔	煙	燕	猿	縁
	3170	8990	艶	苑	薗	遠	鉛	鴛	塩									
オ	3170	8990								於	汚	甥	凹	央	奥	往	応	
	3220	899E		押	旺	横	欧	殴	王	翁	襖	鴬	鳩	黄	畄	沖	荻	億
	3230	89AE	屋	憶	臆	桶	牡	乙	俺	釗	恩	温	穏	音				
カ	3230	89AE													下	化	仮	何
	3240	89BE	伽	価	佳	加	न	嘉	夏	嫁	家	寡	科	暇	果	架	歌	河
	3250	89CE	火	珂	禍	禾	稼	箇	花	苛	茄	荷	華	菓	蝦	課	嘩	貨
	3260	89DE	迦	過	霞	蚊	俄	峨	我	牙	画	臥	芽	蛾	賀	雅	餓	駕
	3270	89EE	介	会	解	日	塊	壊	迥	快	怪	悔	恢	懐	戒	拐	改	
	3320	8A3F		魁	晦	械	海	灰	界	皆	絵	芥	蟹	開	階	貝	凱	劾
	3330	8A4F	外	咳	害	崖	慨	概	涯	碍	蓋	街	該	鎧	骸	浬	馨	蛙
	3340	8A5F	垣	杮	蛎	鈎	劃	嚇	各	廓	拡	撹	格	核	殻	獲	確	穫
	3350	8A6F	覚	角	赫	較	郭	閣	隔	革	学	岳	楽	額	顎	掛	笠	樫
	3360	8A80	橿	梶	鰍	潟	割	喝	恰	括	活	渇	滑	葛	褐	轄	且.	鰹
	3370	8A90	叶	椛	樺	鞄	株	兜	竃	蒲	釜	鎌	噛	鴨	栢	茅	萓	
	3420	8A9E		粥	刈	苅	瓦	乾	侃	冠	寒	刊	勘	勧	巻	喚	堪	姦
	3430	8AAE	完	官	寛	Ŧ	幹	患	感	慣	憾	擙	敢	柑	桓	棺	款	歓
	3440	8ABE	汗	漢	澗	潅	環	甘	監	看	竿	管	簡	緩	缶	翰	肝	艦
	3450	8ACE	莞	観	諌	貫	還	鑑	間	閑	関	陥	韓	館	舘	丸	含	岸
	3460	8ADE	巌	玩	癌	眼	岩	翫	贋	雁	頑	顏	顅					
+	3460	8ADE												企	伎	危	喜	器
	3470	8AEE	基	奇	嬉	寄	岐	希	幾	忌	揮	机	旗	既	期	棋	棄	
	3520	8B3F		機	帰	毅	気	汽	畿	祈	季	稀	紀	徽	規	記	貴	起
	3530	8B4F	軌	輝	飢	騎	鬼	亀	偽	儀	妏	宜	戯	技	擬	欺	犠	疑
	3540	8B5F	祇	義	蟻	誼	議	掬	菊	鞠	吉	吃	喫	桔	橘	詰	砧	杵
	3550	8B6F	黍	却	客	脚	虐	逆	丘	久	化	休	及	吸	宮	弓	急	救
	3560	8B80	朽	求	汲	泣	灸	球	究	窮	笈	級	糾	給	[日	4÷	去	居
	3570	8B90	巨	拒	拠	挙	渠	虚	許	距	鋸	漁	禦	魚	亨	享	京	
	3620	8B9E		供	侠	僑	兇	競	共	X	協	王	卿	Щ	喬	境	峡	強
	3630	8BAE	彊	怯	恐	恭	挟	教	橋	沉	狂	狭	矯	胸	脅	興	蕎	郷
	3640	8BBE	鏡	響	饗	驚	仰	凝	尭	暁	業	局	曲	極	玉	桐	粁	僅
	3650	8BCE	勤	均	τh	錦	斤	欣	欽	琴	禁	禽	筋	緊	芹	菌	衿	襟
	3660	8BDE	謹	近	金	吟	銀											
ク	3660	8BDE						九	俱	句	区	狗	玖	矩	苦	躯	駆	駈
	3670	8BEE	駒	具	愚	虞	喰	空	偶	寓	遇	阳	串	櫛	釧	屑	屈	
	3720	8C3F		掘	窟	沓	靴	轡	窪	熊	隈	粂	栗	縔	桑	鍬	勲	君

B-61863E/14

### APPENDIX P. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST

Pronun		Shift							Seg	ment	and po	oint						
ciation	JIS	JIS	0	1	2	3	4	5	6	7	8	9	А	В	С	D	Е	F
ク	3730	8C4F	薫	訓	群	軍	郡											
ケ	3730	8C4F						卦	袈	祁	係	傾	刑	兄	啓	圭	珪	型
	3740	8C5F	契	形	径	恵	慶	慧	憩	揭	携	敬	景	桂	渓	畦	稽	系
	3750	8C6F	経	継	繋	罫	茎	荊	蛍	計	詣	警	軽	頚	鶏	芸	迎	鯨
	3760	8C80	劇	戟	撃	激	隙	桁	傑	欠	決	潔	穴	結	血.	訣	月	件
	3770	8C90	倹	倦	健	兼	券	剣	喧	巻	堅	嫌	建	憲	懸	拳	捲	
	3820	8C9E		検	権	牽	犬	献	研	硯	絹	県	肩	見	謙	賢	軒	遣
	3830	8CAE	鍵	険	顕	験	鹸	元	原	厳	幻	弦	減	源	玄	現	絃	舷
	3840	8CBE	言	諺	限													
Е	3840	8CBE				乎	個	古	呼	固	姑	孤	5	庫	弧	戸	故	枯
	3850	8CCE	湖	狐	糊	袴	股	胡	菰	虎	誇	跨	鈷	雇	顧	鼓	Ŧī.	互.
	3860	8CDE	伍	午	呉	吾	娯	後	御	悟	梧	檎	瑚	碁	語	誤	護	醐
	3870	8CEE	乞	鯉	交	佼	侯	候	倖	光	公	功	効	勾	厚	Π	向	
	3920	8D3F		后	喉	坑	垢	好	孔	孝	宏	Т.	巧	巷	幸	広	庚	康
	3930	8D4F	弘	恒	慌	抗	拘	控	攻	昂	晃	更	杭	校	梗	構	江	洪
	3940	8D5F	浩	港	溝	甲	皇	硬	稿	糠	紅	紘	絞	綱	耕	考	背	肱
	3950	8D6F	腔	膏	航	荒	行	衡	講	貢	購	郊	酵	鉱	砿	鋼	閤	降
	3960	8D80	項	香	高	鴻	剛	劫	一号	合	壕	拷	濠	豪	轟	麹	克	刻
	3970	8D90	告	玉	穀	酷	鵠	黒	獄	漉	腰	甑	忽	愡	骨	狛	込	
	3A20	8D9E		此	頃	今	木	坤	墾	婚	恨	懇	各	昆	根	梱	混	痕
	3A30	8DAE	紺	艮	魂					1.56	,		t .				-1	ANU.
サ	3A30	8DAE	Just			些	佐	叉	唆	嵯	左	差	査	沙	瑳	砂	詐	鎖
	3A40	8DBE	裟	坐	座	挫	債	催	丹	最	哉	塞	妻	室	彩	才	採	栽
	3A50	8DCE	蒇	済	災	米	犀	砕	砦	祭	斎	洲	采	裁	載	際	剤	在
	3A60	8DDE	材	罪	財	没	坂	阪	堺	榊	有	咲	倘	埼	偷	篇	作	削
	3A70	8DEE	咋	搾	昨	朔	枻	窄	策	索	錯	桜	鮭	笹	匙	₩ <u></u>	刷	
	3B20	8E3F		祭	拶	撮	擦	札	殺	隆	雜	皐	鯖	捌	錆	鮫	Ш. ш	地
	3B30	8E4F	<u> </u>	秋	参	Щ æc	修	撤	餀	桟	爃	圳	産	舁	暴	蚕	顮	筫
21	3B40	8E5F	酸	役		暫	残	11.	17	/=1	11:	±1			머키		Γ.	+/\
<i>У</i>	3B40 9DF0	8E9F 9ECE	4- <del>1:</del>	VE	7	E	+	任	一行	们可	伊	────	リフケ	史	闸	四日	工	始
	3000	8E0F	炯	安	丁	一 夕七	口 11	い	态	忠	佰	文	灯	斯坦	──────────────────────────────────────	百 ≓t	忟	IĽ. ≠t
	3D00 2D70		9Ľ ≣%r	氏次	104 HEI	11E. 11#	仏	术	紙	杀	戊	旧	王	们 一	司	討 +±	武	茚否
	3070	OL9U OFOF	邰	頁	見勿 いみ	に正	即	洲	争	队	1寸 - 二	冗	ੱ ਸਿ	री ट	怒	行	寸 シカ	庙
	3020	OE9E	4-	沃益	山白	(百 <i>标</i> 个	気料 山山	壁.	行電	11 <u>5</u> 55	小	却	부 	日	時安	 -₩	/) 泪	庇沐
	2040	OLAL OFDF	式库	祇	呵	二	翔	六個	下此	_T. _T	൛	സ	大姑	妖全	主. 定	心	他	像劫
	3040	OLDE SECE	伏剑	貝去	天灶	台) 东小	條	1忑 ∋計	市	一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一一	废蛇	瓜瓜	不同 /土	古	与日	豹	宿	<b>火</b> 函
	3030	SECE	示十 而力	和	イム。 全見	ホンニン	1日   信	記れ ここ	中 若	心	町	에 고	1日 千	可生	八	イリ <i>本</i> 立	乃	时
	2070	0EDE 0FFF	的	柳	奶	白 安	水	য়য ত	心	土	収	可掛	丁	不量	2不 □□	<b>小</b> 寸 11寸	坏国	作里
	3070	8E3E	胆	应应	伯		価	又秋	虹	对	1又 禾	毛小	- 市又 - 幺久	前線	24 সূর	山	向点	古
	3D20	of of 8F4F	竝	示龍	小儿	ク11 5計	移	调	〕口 〕口	高切	一乃住	イス 新館	ポマ   	一個和	白云	天山	<b>万</b> 従	龙
	3D30	0141 8555	承矛	表	言	SSNL 当小	半日 叙注	過舌	留弦	却	乐	陀定	11 河	1主 <b>1</b> 豆	近線	ま	1)C 载	丸動
	3D40 3D50	of of SFGF	未山	行		台八 /允	而无	里奉	<b>兆</b>	小人	戸い	1日 軍允	似	征	相	床	至	沉
	3D60	8780	山淮	741) 消弭	心居	反結	一 一 夜 一 二 次 一 一 次 一 一 次 一 一 次 一 一 次 一 一 次 一 一 次 一 一 次 一 一 次 一 一 一 次 一 一 一 次 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一	百	呼	收	ク平 加	雨久 之口	1出 正	阳显	可服	1/日 注述	7円 由	子丝
	3D70	8F00	- 平 - 聖	山ま	山	が出 書転	) (世)	近曲	印	順 <i>十</i>	应应	似	7/1	有細	「石」	伯	心障	邢日
	3E90	SEOE	有	盲	省	· 百 二	「相」	可	承	メ	<b>パ</b> 登	1示 将	恣	如功	际	100	[貝   人	715
	3E20	SEVE	坚	)))) 	正	ノ「	白	可可	山	'日 切	百世	天」	女見	川	日	1寸 日	小 私	ン お
	3E40	SFRF	同	准	が辺	· 加入 沙出	¥) 法	/手	1ブ   佐	旧	手	泥	一升	日本	旧	旧	环	百日
	3E50	SECE	1平	北京	1日 皐刀	旧	び	蓝	花	而	加	2011. 言A	司	司日	司局	1十 负	小小	平
	3E60	8FDF	不	循	俗	I] []	目出	14	于	岡	云垂	пд ГГ	ELL. 垂lí	다 다라	11 倶	<b>豕</b> 憧	貝	四倍
	0000	OLDE	迎上	哩	ツ里	垾	ΨH		乂	小	术	76	不り	ウズ	吻	垠	外表	茚

### P. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST APPENDIX

B-61863E/14

Pronun		Shift							Seg	gment	and po	oint						
ciation	JIS	JIS	0	1	2	3	4	5	6	7	8	9	А	В	С	D	Е	F
シ	3E70	8FEE	情	擾	条	杖	浄	状	畳	穣	蒸	譲	醸	錠	嘱	埴	飾	
	3F20	903F		拭	植	殖	燭	織	職	色	触	食	蝕	辱	尻	伸	信	侵
	3F30	904F	唇	娠	寝	審	心	慎	振	新	晋	森	榛	浸	深	申	疹	真
	3F40	905F	神	秦	紳	臣	芯	薪	親	診	身	辛	進	針	震	人	仁	刃
	3F50	906F	塵	Ŧ	尋	甚	尽	腎	訊	迅	陣	靭						
ス	3F50	906F											笥	諏	須	酢	$\mathbb{X}$	厨
	3F60	9080	逗	吹	垂	帥	推	水	炊	睡	粋	翠	衰	遂	酔	錐	錘	随
	3F70	9090	瑞	髄	崇	嵩	数	枢	趨	雛	据	杉	椙	菅	頗	雀	裾	
	4020	909E		澄	摺	寸												
セ	4020	909E					世	瀬	畝	是	凄	制	勢	姓	征	性	成	政
	4030	90AE	整	星	晴	棲	栖	正	清	牲	生	盛	精	聖	声	製	西	誠
	4040	90BE	誓	請	逝	醒	青	静	斉	税	脆	隻	席	惜	戚	斥	昔	析
	4050	90CE	石	積	籍	績	脊	責	赤	跡	蹟	碩	切	拙	接	摂	折	設
	4060	90DE	窃	節	説	雪	絶	舌	蝉	仙	先	Ŧ	占	宣	専	尖	JII	戦
	4070	90EE	扇	撰	栓	栴	泉	浅	洗	染	潜	煎	煽	旋	穿	節	線	
	4120	913F		繊	羨	腺	舛	船	薦	詮	賎	践	選	遷	銭	銑	閃	鮮
	4130	914F	前	善	漸	然	全	禅	繕	膳	糎							
レソ	4130	914F								_		噌	塑	岨	措	曾	曽	楚
	4140	915F	狙	疏	疎	礎	祖	租	粗	素	組	蘇	訴	阻	遡	鼠	僧	創
	4150	916F	双	叢	倉	喪	壮	奏	爽	宋	層	匝	惣	想	捜	掃	挿	掻
	4160	9180	操	早	曹	巣	槍	槽	漕	燥	争	痩	相	窓	糟	総	綜	聡
	4170	9190	草	荘	葬	蒼	藻	装	走	送	遭	鎗	霜	騒	像	増	憎	
	4220	919E		臓	蔵	贈	造	促	側	則	即	息	捉	束	測	足	速	俗
	4230	91AE	属	賊	族	続	卒	袖	其	揃	存	孫	尊	損	村	遜		
<i>y</i>	4230	91AE															他	多
	4240	91BE	太	汰	詑	唾	堕	妥	惰	打	柁	舵	楕	陀	駄	騨	体	堆
	4250	91CE	対	耐	岱	帯	待	怠	態	戴	替	泰	滞	胎	腿	苔	袋	貸
	4260	91DE	退	逮	隊	黛	鯛	代	台	大	第	醍	題	鷹	滝	瀧	卓	啄
	4270	91EE	宅	托	択	拓	沢	濯	琢	託	鐸	濁	諾	茸	凧	蛸	只	
	4320	923F		叩	但	達	辰	奪	脱	巽	竪	辿	棚	谷	狸	鱈	樽	誰
	4330	924F	丹	単	嘆	坦	担	探	<u> </u>	歎	淡	湛	炭	短	端	箪	綻	耽
	4340	925F	胆	蛋	誕	鍛	寸	壇	弹	断	暖	檀	段	男	談	1.1.		
ーチ	4340	915F	7.0		61-1						1.1			L.L.		值	知	地
	4350	926F	弛	恥	智	池	痴	椎	置	纹	蜘	送	馳	染	畜	竹	筑	畜
	4360	9280	逐	秩	窒	奈	嫡	者	甲	仲	笛	思	抽	昼	在	注	虫	农
	4370	9290	誑	町	鋳	駐	樗	術	猪	守	者	貯		兆	<i>満</i>	喋	龍	et II.
	4420	929E	77-12	- 阳G	帳	厅	书	張	彫	徴	窓	珧	畅	朝	潮	除	町	眺
	4430	92AE	肥	服	肠	檗	調	髞	超	跳	銚	長	頂	局	籾	莎	直	肤
	4440	92BE	沉	岁	頁	鉙	陳	Sch.	1736	+#	白体	ነሳ	ውሳ	1÷	<u>بح</u> ر	147	+151	슈코
9	4440	92BE	448	/==1	油	<del>т</del> 4	NL	洋	堅	惟	恒	垣	逅	捕	世	LAK V-L	「「「」	旭
	4450	92CE	槻	佃	湏	怕	ヹ	鳥	潊	釫	傄	〔〕〔〕	冸	重	帰	紺	Л	币
	4460	92DE	到	低時		Irr.	一	/占	-41	Ъ		14	<del>با</del> ر	*	Þ	占	71	
	4460	92DE	1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1	+rr	(学	1氏 +14	1宁	[貝 ·汀	利	貝	±. ≠₽	埞	正	审 ≣⊤	此	庭	廷	퐈
	4470	92EE	吊	山	挺	捉	協	7] 201	<b>灰</b>	但	住	<b>が</b> 第 ンプ	版	計	前	町	一 地	±⊷
	4520	933F	1th	氏り	彩	逝] \止	鼎	泥	摘		創	滴	的	田	週	· 폐	湖	召
	4530	934F	徹	撤	戦	迭	跃	一	項	大	脹	店	冻	耀	ΤΤ	肟	<b>甲</b> 五	則
	4540	935F	点	厶	殿	澱	田	電	H	11	1-14	· <b>〉</b>	47		4-1-	N	14	Serie:
	4540	935F	_∞	- <u>+</u>	П-14	<u>م</u>	1717	於古	児	<u> </u> ビ	省	堡	州	者	使	가	忙	波
	4550	936F	1 ①	免	「賄」	团	都	遐	低	砺	労	皮	土	火	怒	倒	兄	冬
	4560	9380	凍	刀	唐	塔	塘	套	宕	局	喝	悼	投	谸	曱	秘	梼	棟

B-61863E/14

Pronun		Shift							Seg	gment	and po	oint						
ciation	JI2	JIS	0	1	2	3	4	5	6	7	8	9	А	В	С	D	Е	F
ト	4570	9390	盗	淘	湯	涛	灯	燈	当	痘	祷	等	答	筒	糖	統	到	
	4620	939E		董	蕩	藤	討	謄	豆	踏	逃	透	鐙	陶	頭	騰	闘	働
	4630	93AE	動	н,	堂	導	憧	撞	洞	瞳	童	胴	萄	道	銅	峠	鴇	匿
	4640	93BE	得	徳	涜	特	督	禿	篤	毒	独	読	栃	橡	凸	穾	椴	届
	4650	93CE	鳶	苫	寅	酉	瀞	噸	屯	惇	敦	沌	豚	遁	頓	呑	曇	鈍
ナ	$\begin{array}{c} 4660 \\ 4670 \end{array}$	93DE 93EE	奈	那難	内	乍	凪	薙	謎	灘	捺	鍋	楢	馴	繩	畷	南	楠
_	4670	93EE	<del>+</del> /\	大田	12	_	尼	1=	浙	匂	賑	肉	φŢ	÷	H	핃	7	
	4720	943F		如	尿	韮	任	ス 妊	忍	認	ЯД	Ρų	ملح <b>تر</b>	н	н	20	Χ	
ヌ	4720	943F									濡							
ネ	4720	943F										禰	袮	寧	葱	猫	熱	年
	4730	944F	念	捻	撚	燃	粘			,			bar	_#L				vtalla
)	4730	944F						乃	廼	え	埜	嚢	悩	濃	納	能	脳	膿
	4740	945F	晨	覗		rrt .	1 ml	100	TTT	لتتل	24	1	<b>1</b> 111	тњ	रेश हि	m	-++-	œ
л 	4740	945F	711.	1. der	477	巴	把	播	朝	祀	波	派	包	- 仮	妥	馬	巴	馬
	4750	946F	19F	廃	件 7日	排	敗	竹	血	牌	自首	肺	軍	四亡	倍	出	保	附
	4760	9480	保	保	狽	貝	元	胎	尚	這	蝿	₩ 1	別	秋	1山	羽	(円) (一)	抇
	4770	9490	们	旧	日	泊	柏	加	海	坦	塨	没	爆	将	旲	馭	友	<b>.</b> 7%
	4820	949E		区	相	俗	者	筆	舌	加速	喻	肌	加	自住		₩4 1	洗	<b>光</b>
	4830	94AE	凹光	发	1又 	割	拔	伐	伐	帰	「「「	词	蹈	毕	1日 ⁻	刊	手	反
	4840	94BE	刑	岘	版	斑	<b>仆</b> 乂	7巴 17な	孔	加入	3E	班	呼	系	加又	潘	舣	車巳
14	4850	94CE	术	煩	<i>顶</i>	耿	拀	呪	偣	隘	省	畨	蛮	मा	白	不	47	品
Ľ	4000	940E	加快	⊐E	=	+LL	+.++	司由	LLa	হম	vite	ц <del>г</del> ,	拍	更上  毛辺	平	百靈	火亡	此
	4000	94DE 04FF	1火 ⇒北	心弗	肺	11L ∃1:	加	安 155	厶	化供	放	区侧	11年 十日	化化	形牛	能思	<u>北</u> 主	彻又
	4070	94EE 052F	矿	貝自	地	チャ	戒	他	敗	1/用 	比	100	印化	起	比	用風	天堂	」可
	4920	955F 054F	14	异 加	个 <del></del> 極		프	्र ≓बब्र	距住	厚	脉	変业	川	切り	必	華主	■■	迎
	4930	954F	伝	炉	版	ポゴ モ小	日世	診	依	尼士	际	小	保日	瓢	宗	衣	計	初分
	4940 $4950$	955F 956F	南	拥	· 府 敏	砂瓶	Ш	亚田	珙	亦亦	蚷	周	白白	111岁	XIL	供	伊只	貝
フ	4950	956F		///		/#4	不	付	埠	夫	婦	富	富	布	府	怖	扶	敷
	4960	9580	斧	普	浮	父	符	腐	膚	芙	譜	負	賦	赴	阜	附	侮	撫
	4970	9590	武	舞	葡	蕪	部	封	楓	風	葺	蕗	伏	副	復	幅	服	•••••
	4A20	959E		福	腹	複	覆	淵	弗	払	沸	仏	物	鮒	分	吻	噴	墳
	4A30	95AE	憤	扮	焚	奮	粉	糞	紛	雰	文	聞						
~	4A30	95AE											丙	併	兵	塀	幣	平.
	4A40	95BE	弊	柄	並	蔽	閉	陛	米	頁	僻	壁	癖	碧	別	瞥	蔑	箆
	4A50	95CE	偏	変	片	篇	編	辺	返	遍	便	勉	娩	弁	鞭			
ホ	4A50	95CE														保	舗	鋪
	4A60	95DE	圃	捕	歩	甫	補	輔	穂	募	墓	慕	戊	暮	母	簿	菩	倣
	4A70	95 EE	俸	包	呆	報	奉	宝	峰	峯	崩	庖	抱	捧	放	方	朋	
	4B20	963F		法	泡	烹	砲	縫	胞	芳	萌	蓬	蜂	褒	訪	豊	邦	鋒
	4B30	964F	飽	鳳	鵬	乏	ť	傍	剖	坊	妨	帽	忘	忙	房	暴	望	某
	4B40	965F	棒	冒	紡	肪	膨	謀	貌	貿	鉾	防	吠	頬	北	僕	1	墨
	4B50	966F	撲	朴	牧	睦	穆	釦	勃	没	殆	堀	幌	奔	本	翻	凡	盆
マ	4B60	9680	摩	磨	魔	麻	埋	妹	昧	枚	毎	哩	槙	幕	膜	枕	鮪	柾
	4B70	9690	鱒	桝	亦	俣	又	抺	末	沫	迄	侭	繭	麿	万	慢	満	
	4C20	969E		漫	蔓													
""	4C20	969E	, lest			味	未	魅	E	箕	岬	密	蜜	湊	蓑	稔	脈	妙
,	4C30	96AE	粍	民	- 眠	767	**	<u>ر</u> ست	4	~	æ	北中		4 <b>ग</b> '	<u>+4</u>			
4	4C30	96AE				務	岁	燕	半	オ	務	武局	尔	焇	姄			

### P. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST APPENDIX

B-61863E/14

Pronun		Shift							Seg	gment	and p	oint						
ciation	112	JIS	0	1	2	3	4	5	6	7	8	9	А	В	С	D	Е	F
×	4C30	96AE														冥	名	命
	4C40	96BE	明	盟	迷	銘	鳴	姪	牝	滅	免	棉	綿	緬	面	麺		
モ	4C40	96BE															摸	模
	4C50	96CE	茂	妄	孟	毛	猛	盲	網	耗	蒙	儲	木	黙	冃	杢	勿	餅
	4C60	96DE	尤	戻	籾	貰	問	悶	紋	門	匁							
ヤ	4C60	96DE										也	冶	夜	爺	耶	野	弥
	4C70	96EE	矢	厄	役	約	薬	訳	躍	靖	柳	薮	鑓					
고	4C70	96EE												偷	愈	油	癒	
	4D20	973F		諭	輸	唯	佑	優	勇	友	宥	R	悠	憂	揖	有	柚	湧
	4D30	974F	涌	猶	酋犬	山	祐	裕	誘	遊	邑	郵	雄	融	タ			
Е	4D30	974F														予	余	与
	4D40	975F	誉	輿	預	傭	幼	妖	容	庸	揚	揺	擁	曜	楊	様	洋	溶
	4D50	976F	熔	用	窯	羊	耀	葉	蓉	要	謡	踊	遥	陽	養	慾	抑	欲
	4D60	977F	沃	浴	꼬	翼	淀											
ラ	4D60	977F						羅	螺	裸	来	莱	頼	雷	洛	絡	落	酪
	4D70	978F	乱	卵	嵐	檷	濫	藍	蘭	覧								
リ	4D70	978F									利	吏	履	李	梨	理	璃	
	4E20	979E		痢	裹	裡	里	離	陸	律	率	$\overline{\underline{M}}$	葎	掠	略	劉	流	溜
	4E30	97AE	琉	留	硫	粒	隆	竜	龍	侶	慮	旅	虜	了	亮	僚	両	凌
	4E40	97BE	寮	料	梁	涼	猟	療	瞭	稜	糧	良	諒	遼	量	陵	領	力
	4E50	97CE	緑	倫	厘	林	淋	燐	琳	臨	輪	隣	鱗	鹿粦				
ル	4E50	97CE													瑠	塁	涙	累
	4E60	97DE	類															
ν	4E60	97DE		令	伶	例	冷	励	嶺	怜	玲	礼	苓	鈴	隷	零	霊	麗
	4E70	97EE	齢	暦	歴	列	劣	烈	裂	廉	恋	憐	漣	煉	簾	練	聯	
	4F20	983F		蓮	連	錬												
П	4F20	983F					呂	魯	櫓	炉	賂	路	露	労	婁	廊	弄	朗
	4F30	984F	楼	榔	浪	漏	牢	狼	篭	老	聾	蝋	郎	六	麓	禄	肋	録
	4F40	985F	論		-													-
ワ	4F40	985F		倭	和	話	歪	賄	脇	惑	枠	鷲	亙	亘	鰐	詫	藁	蕨
	4F50	986F	椀	湾	碗	腕												

## Index

### Note

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

### ≪A≫

About connection log of Ethernet, 603, 897 About the FS15i PMC-NB6 programmer, 498 About the FS16i/18i/21i-B PMC-SA1/SB7 programmer, 498 ADD (addition), 239 ADDB (binary addition), 241 Addition of system keep relays, 788, 790 Addition of system parameter ladder execution time, 1415 Additions to sequence programs, 1025 Address, 59 Address change of sequence program, 532 Address of counter (C), 104 Address of data table (D), 108 Address of keep relay and nonvolatile memory control (K), 106 Addresses, 363 Addresses between PMC and CNC (PMC↔NC), 74 Addresses between PMC and machine tool, 75 Addresses between PMC and machine tool (PMC↔MT), 75 Addresses for message selection displayed on CRT (A), 102 Addresses, signal names, comments, and line numbers, 363 Alarm message (PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/ SB2/SB3/SB4/SB5/SB6/SC/SC3/SC4/NB/NB2/NB6),1422 Alarm message (PMC-SB7), 1429 Alarm message and countermeasure, 837, 858 Alarm message list, 1421 Alarm screen (ALARM), 428, 639 ALLOW PMC STOP (PMC-SB7: K902.2, PMC-SA1: K19.2), 799 Alter, 997 Alteration of sequence programs, 524 AND, 122 AND, NOT, 122 AND. STK. 125 Applicable FAPT ladder editions, 1396 Applications, 371 Arbitrary functional instructions, 294 Area managed by the system program, 97 Assignment of I/O module addresses, 77 Automatic operation when the power is turned on, 412 Automatic start of trace setting, 874 Automatic tracing function at power on, 431 AXCTL (axis control by PMC), 337

### ≪**B**≫

Block step, 931 Brief screen, 826

### ≪C≫

C input/output, 556 CALL (conditional subprogram call), 330 Calling screen, 832 CALLU (unconditional subprogram call), 331 Caution screen of the setting parameter for the selectable I/O link assignment function, 795 Cautions for subroutines 378 Chinese character code, Hiragana code, and special code list, 1457 Clearing the PMC parameter, 545 Clearing the sequence program, 412, 544 Clearing the sequence program and condensation of the sequence program, 544  $CNC \rightarrow Offline Programmer, 1403$ COD (code conversion), 189 CODB (binary code conversion), 192 Coding (step 5), 38 COIN (coincidence check), 220 Collation of program, 1017 Collation of source programs, 1017 Collective monitor function, 747 Collective monitor function (PMC-SB7), 831 COM (common line control), 197 COME (common line Control end), 203 Comments, 364 Communication Status, 602 Communication status, 895 COMP (comparison), 216 Comparison with the function of FS15-B (PMC-NB/NB2), 650 Compatibility of ladder diagram, 963 Compatibility with CNC basic software, 1389 Compatibility with Series 21-MODEL B, 1420 Compatible with Series 0, 1416 COMPB (comparison between binary data), 218 Component units, 500, 971, 1391 Component units and connections, 499, 970, 1390 Compress the sequence program, 545 Compressed input by [COMAND] key, 1034 Condense, 1393 Condense command - release of deleted area, 1045 Configuration and operation of step-sequence programs, 922 Configuration of command, 1042 Confirming the ladder mnemonics, 614 Connecting component units, 505 Connecting the I/O card, 1382 Connecting the I/O unit, 1382 Connecting the operator's panel for FS 0 with FS16, FS18, FS21, or Power Mate, 1378

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

Connection, 1382 Connection of components, 1392 Connections of units, 972 Continuous data entry, 665 Convergence of selective sequence, 927 Convergence of simultaneous sequence, 929 Conversion, 1376 Copy a ladder program [CPYLAD], 575 Copy I/O module data [CPYMDL], 576 Copy message data [CPYMSG], 575 Copy symbol data and comment data [CPYSYM], 575 Copy the sequence programs [CPYALL], 576 Copy title data [CPYTTL], 575 Copying a sequence program, 1031 Copying message data (COPY), 543 Copying the sequence program, 530 Corresponding function, 962 Counter screen (COUNTR), 464, 667 Creating a program, 376 Creation of interface specifications (steps 1 to 3), 37 Creation of ladder diagram (step 4), 37 Cross reference display, 546 CRT/MDI operation, 946 CTR (counter), 172 CTRB (fixed counter), 178 CTRC (counter), 180

### ≪D≫

Data display and setting (title, symbol, ladder program, comment, message, I/O module), 990 Data input to and output from other devices, 569 Data keys and screen scroll key, 978 Data number, data attribute, data length, data area, 1368 Data table (DATA), 470, 670 DATA TBL CNTL SCREEN (PMC-SB7: K900.7, PMC-SA1: K17.7), 800 Data transfer between NB (4047 series) and FAPT LADDER, 1408 Data transfer between NB (4047 series) and NB (4048 series), 1409 Data transfer between NB/NB2 (4048 series) and FAPT LADDER, 1408 DCNV (data conversion), 212 DCNVB (extended data conversion), 214 Debugging, 584 DEC (decode), 167 DECB (binary decoding), 169 Delete, 1001 Delete of sequence program, 527

Delete of symbol data and comment, 538 Deleting a sequence program, 1028 Deleting title data, 516 Deletion of programs, 1018 Descriptions of displayed items, 580 Detail screen, 824 Details of basic instructions, 116 DIFD (falling edge detection), 306 Difference of status of signals between 1st level and 2nd level, 53 DIFU (rising edge detection), 305 Direct editing by ladder diagram, 1020 DISP (message display) (PMC–SB/SB2/SB3/SB4/SB5/SB6/SC/ SC3/SC4 Only), 258 DISPB, 272 Display format for parameters, 818 Display of ladder diagram, 1003 Display of signal status (STATUS), 427 Display PMC input/output signals and internal relay (PMCDGN), 635 Displayed items, 583 Displaying and setting the configuration status of I/O devices (IOCHK), 442 Displaying input code (DSPMOD), 543 Displaying of sequence program, 947 Displaying the contents of memory, 432 Displaying the GDT (global descriptor table), 578 Displaying the memory allocation information of a user program Coded in C., 581 Displaying the running state of a user task (USRDGN), 440 Displaying title data, 425 Distribution I/O connection I/O module and distribution I/O operator's panel I/O module assignment methods, 89 DIV (division), 251 DIVB (binary division), 253 Divergence of selective sequence, 927 Divergence of simultaneous sequence, 928 Dividing display of ladder diagram, 489 Do-while statement, 1448 DSCH (data search), 224 DSCHB (binary data search), 227 DUMP display on ladder diagram, 481

### ≪E≫

EDIT ENABLE (PMC–SB7: K901.6, PMC–SA1: K18.6), 799 Editing a character string in message data, 542 Editing character strings of symbol data and comment data, 539 Editing character strings of title data, 517 Editing end, 1004

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

Editing for Power Mate-MODEL D (PMC-PA1/PA3), 1388 FAPT LADDER system floppy loading, 985 Editing function of ladder diagram, 957 FAPT LADDER, FAPT LADDER-II, FAPT LADER-III, ladder editing packages, FAPT LADDER-III (IBM PC/AT) A08B–9210–J505, ladder editing package (windows) (IBM PC/ Editing ladder mnemonics, 613 AT) A08B-9210-J511, 1397 Editing of sequence program (EDIT), 515 FDLIST command - file attribute display, 1043 Editing symbol data and comment at once, 532 File editing function, 1038 Editing the sequence program (EDIT), 612 Floppy list screen, 773 Emergency stop signal (*ESP), 1383 For MDI Units other than Standard MDI Units (for FS20 PMC-SA1 Enabling automatic debugging at power-on, 588 and SA3), 412 END (end of a ladder program), 329 For statement, 1450 End of block step, 932 For the FS15 (PMC-NB), 493 END1 (1st level sequence program end), 157 For the FS16 (PMC-SC or PMC-SC3), 493 END2 (2nd level sequence program end), 158 Forced input/output function, 449 END3 (end of 3rd level sequence) (PMC–SC/SC3/SC4/NB/ NB2/NB6/NB7 only), 159 Forcibly stopping the sequence program, 553 Format and details of the control data of the WINDR functional Ending edit of a sequence program, 1034 instruction, 1256 Ending ladder mnemonics editing, 618 Format and details of the control data of the WINDW functional Entering data on the program check screen (*low-speed response) (not available for Power Mate-D/F, Series 21-TA), 1141 instruction, 1309 Formats and details of control data, 1059 Entering title data, 516 FS15i PMC-NB6 operating procedure, 413 Entering torque limit data for the digital servo motor (*low-speed FS16i-LA assignment, 88 response), 1145 FS16i/18i/21i-B PMC-SA1/SB7 operating procedure, 413 EOR (exclusive OR), 307 Function, 374, 1055, 1250, 1332, 1376 Error codes list (for FAPT Ladder P-G), 1051 Function for displaying signal waveforms (ANALYS), 434 Error details, 622 Function for storing data in memory, 433 Error list, 626 Function of copying symbol and comment data, 540 Error messages (for EDIT), 605 Functional instruction data table editor screen, 854 Error messages (for I/O), 607 Functional instruction data table viewer screen, 821 Error messages (for ladder mnemonics editing), 620 Functional instruction list screen, 853 Example, 879 Functional instruction TRSET, 934 Example for setting parameters, 801 Functional instruction WINDR, 1252 Example of step sequence programs, 1441 Functional instruction WINDW, 1254 Examples of structured programming, 370 Functional instructions, 130 Exclusive control for functional instructions, 943 Functions for displaying memory areas and debugging the program Execution method, 375 (MONIT), 57 Execution of a sequence program, 551 Functions of processing, 968 Execution of step sequence, 918 Execution of Trace, 647 Execution of trace, 870 ≪G≫ Execution procedure of sequence program, 42 General rules, 937 Execution time, 396 Graphical symbols, 905 EXIN (external data input), 283 Expansion of symbol size, 790 Extended ladder instructions, 933 *≪***H**≫

### ≪**F**≫

F keys (F1 to F0), 976 FAPT LADDER (system P series), 1399

# Help screen, 1004 HIDE PMC PROGRAM (PMC–SB7: K900.0, PMC–SA1: K17.0), 798 How to pick up a ladder diagram net, 750 How to set at NC parameter, 597

### B-61863E/14

### Note

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

How to set at NC parameter (Power Mate–D/H), 598 How to set at PMC screen, 596, 889

### ≪**I**≫

I/O error messages, 573 I/O Link connecting check screen, 444 I/O link connecting check screen, 886 I/O Link connection unit assignment, 84 I/O Link-II parameter setting screen, 446 I/O screen, 757 I/O screen error messages (PMC-NB6), 779 I/O signals to CNC, 52 I/O unit address setting (MODULE), 534 I/O Unit MODEL B assignment, 86 If else statement, 1452 Implementation techniques, 370 Infinite number of relay contacts, 367 Initial block step, 932 Initial step, 925 Input of program, 1005 Input PMC parameters from MDI panel, 462 Input signal processing, 51 Input with a katakana identification code, 543 Input/outpupt signal processing, 395 Input/output ladder/PMC-parameter by MDI/DPL, 623 Input/output method to FANUC FLOPPY CASSETTE (fixed 4800bit/sec.), 623 Input/output method to office programmer (P-g Mate/Mark II) (fixed 9600bit/sec.), 623 Input/output of ladder program with P-G and floppy cassette/FA card, 1035 Inputting a multi-byte character (D.CHAR), 543 Insert, 1000 Insert of sequence program, 524 Interlocking, 54 Internal relay addresses (R), 95 Interrupt enable/disable/clear, 396 Interrupt processing, 394 Interrupt program, 394 Interrupt-type PMC function, 391 IO GROUP SELECTION (K906.1), 800

### ≪J≫

JMP (jump), 204 JMP Instructions with Label Specification, 380 JMPB (label jump), 334 JMPC (label jump), 335 JMPE (jump end), 209 Jump, 930

### ≪**K**≫

Keep relay (KEEPRL), 464 Keep relay screen (KEEPRL), 668 Key switch signals (Xn, Xn+2), 1383 Keyboard of system P series, 975

### ≪L≫

Label, 930 Label addresses (JMPB, JMPC, LBL) (L), 112 Ladder debugging function, 589 Ladder diagram display, 480 Ladder diagram editor function (PMC-SB7), 838 LADDER diagram editor screen, 702 Ladder diagram format, 362, 366 LADDER diagram monitor (PMC-SB7), 803 Ladder diagram monitor and editor, 678 LADDER diagram monitor screen, 681, 805 Ladder password function, 414 Ladder screen, 950, 960 Lader diagram editor screen, 839 LBL (label), 336 Level up of input/output function with memory card, 1401 Limitations in SYSTEM P Mate, 1020 Limitations with the SYSTEM P Mate, 984 Limited function, 1420 Line numbers, 364 List of window functions, 1056 LOAD key (system program loading key), 976 Loading of floppy, 984 Loading the standard ladder (for Power Mate-D/F PMC-PA1 and PA3), 413 Location search, 1001 Logical AND, 309 Logical OR, 311 Low-speed response and high-speed response of window function, 1055, 1251

### ≪M≫

Memory card list screen, 764

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

Message data input, 542

Message data setting (MESSAGE), 541 Method for entering PMC parameters, 664 Migration from the PMC-NB/NB2 to the PMC-NB6, 1413 Migration from the PMC-SA5/SB5/SB6 to the PMC-SB7, 1414 Migration from the Series 0 and Series 21-B to Series 0i, 1416 Migration of ladder programs between different models, 1410 Migration of ladder programs from Power Mate-D/H to Power Mate *i*–D/H, 1411 Miscellaneous item, 368 MMC3 R (MMC-III window data read), 315 MMC3W (MMC-III window data write), 318 MMCWR (reading MMC window data) (other than PMC-PA1/PA3), 297 MMCWW (writing MMC window data) (other than PMC-PA1/PA3), 299 Modification procedure, 1377 Modifying the converted sequence program, 1377 Modifying the ladder mnemonics, 616 Modifying the values of signals by forced input/output, 457 Monitor time screen, 955 Monitoring elapsed time, 954 MOVB (transfer of 1 byte), 301 MOVD (transfer of 4 bytes), 304 MOVE (logical product transfer), 194 Moving a sequence program, 1032 Moving the sequence program, 531 MOVN (transfer of an arbitrary number of bytes), 303 MOVOR (data transfer after logical sum), 196 MOVW (transfer of 2 bytes), 302 MUL (multiplication), 247 MULB (binary multiplication), 249 Multiple data input, 462

### ≪**N**≫

NB/NB2 data compatibility, 1408 Net editor screen, 846 Nonvolatile memory, 353 NOP (no operation), 351 NOT (logical NOT), 313 Notes, 883 Notes on using an MDI keyboard without cursor keys (when using the FS20 PMC–MODEL SA1/SA3), 572 NUME (definition of constant), 255 NUMEB (definition of binary constants), 256

### ≪**0**≫

Offline Programmer  $\rightarrow$  CNC, 1405 On-line debugging function (only for Power Mate-D/H), 624 On-line edit, 490 Online function, 594, 887 Online function by Ethernet, 891 Online function by ethernet, 599 Online setting screen, 595, 888 Operation, 429, 432, 434, 546, 578, 581, 584, 981, 1403 Operation after execution of trace, 648, 871 Operations, 559, 808, 822, 827 OR, 122 OR. NOT, 122 OR. STK, 125 Other setting screens, 475 Other window functions, 1374 Outline, 876 Outline of leveled up contents, 1402 Output of program, 1010 Output signal processing, 51 Outputting to and inputting from flash ROM, 766 Outputting to and inputting from floppy, 769 Outputting to and inputting from memory cards, 760 Outputting to and inputting from other input/output devices, 774 Override signals (*OV1 to *OV8) and program protect key signal (KEY), 1383

### ≪**P**≫

Paper command, 1013 Parameter, 1392 Parameter display on ladder diagram, 482 Parameter menu, 1386 Parameter screen, 435, 585 Parameter setting and display, 987 Parameter setting screen, 429, 547 PARI (parity check), 210 Password and switching screen, 754 Password protection, 752 PMC address (S ADDRESS), 934 PMC basic instructions, 114 PMC basic manu, 631 PMC data setting and display (PMCPRM), 662 PMC data table, 357 PMC I/O signal display and internal relay display (PMCDGN), 424 PMC ladder diagram display (PMCLAD), 479 PMC menu selection procedure by softkey, 418

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

PMC operation for loader control function, 417 PMC parameters setting and display (PMCPRM), 460 PMC programmer (CRT/MDI or PDP/MDI) [ladder editing function], 1390 PMC programmer (DPL/MDI) (only for the Power Mate-D/F/H), 608 PMC screen transition and related soft keys, 633 PMC system alarm messages (PMC-SB7), 1439 Port setting screen, 777 Power Mate assignment, 87 Preparation before operation, 984 Priority of execution (1st level, 2nd level and 3rd level), 44 Procedure, 1386 Processing I/O signals, 50 Program collation, 1037 Program configuration list (main screen), 947, 957 Program editing, 990 Program input, 1035 Program list editor screen, 742, 855 Program list viewer screen, 735, 824 Program output, 1036 PROGRAMMER ENABLE (PMC-SB7: K900.1, PMC-SA1: K17.1), 798 Programmer menu screen, 986 Programmer protection function, 797 Programming, 906 Programming from keyboard, 996 PSGN2 (position signal output 2), 347 PSGNL (position signal output), 343

### ≪**R**≫

R keys (R0 to R3), 977 RAM WRITE ENABLE (PMC-SB7: K900.4, PMC-SA1: K17.4), 800 RD, 118 RD. NOT, 119 RD. NOT. STK, 124 RD. STK, 123 Reading a character string of the CNC program being executed in the buffer, 1153 Reading a custom macro variable (*low-speed response), 1078 Reading a custom macro variable (low-speed response), 1260 Reading a parameter (*low-speed response), 1070 Reading a parameter (not available for Power Mate-D/F, Series 21-TA), 1150 Reading a parameter (setting data) (low-speed response), 1259 Reading a skip position (stop position of skip operation (G31)) of controlled axes, 1098

Reading a skip position (stop position of skip operation (G31)) of controlled axes (low-speed response), 1267 Reading a tool offset, 1062 Reading a tool offset (low-speed response), 1256 Reading a workpiece origin offset value, 1258 Reading a workpiece origin offset value (not supported by the Power Mate-D or -F), 1066 Reading A/D conversion data, 1111 Reading A/D conversion data for the feed motor, 1276 Reading actual spindle speeds, 1213 Reading an estimate disturbance torque data, 1291 Reading and writing of nonvolatile memory data, 356 Reading and writing the laser command data and laser setting data, Reading clock data (date and time) (not available for Power Mate-F), 1143 Reading clock data (low-speed response), 1288 Reading CNC status information, 1159 Reading CNC system information, 1060 Reading detailed alarm information (*low-speed type), 1245 Reading diagnosis data (*low-speed response), 1109 Reading diagnosis data (low-speed response), 1275 Reading diagnosis data (not available for Power Mate–D/F, Series 21–TA), 1152 Reading fine torque sensing data (statistical calculation results), 1217 Reading fine torque sensing data (store data), 1219 Reading load information of the spindle motor (serial interface), 1147 Reading machining distances, 1361 Reading modal data, 1104 Reading modal data (low-speed response), 1269 Reading model data, 1357 Reading of the comment, 1336 Reading of tool setting data, 1366 Reading real parameters (low-speed type), 1299 Reading set data (not available for Power Mate-D/F, Series 21-TA), 1151 Reading setting data, 1352 Reading setting data (*low-speed response), 1074 Reading signals automatically at power on, 439 Reading the absolute position (absolute coordinates) of controlled axes, 1094 Reading the absolute position on a controlled axis, 1264 Reading the acceleration/deceleration delay on controlled axes, 1102, 1268 Reading the actual spindle speed, 1139 Reading the actual velocity of controlled axes, 1092 Reading the actual velocity of controlled axes (low-speed response), 1264 Reading the CNC alarm status, 1083, 1356 Reading the CNC alarm status (low-speed response), 1261 Reading the current program number, 1088

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

- Reading the current program number (8-digit program numbers) (not available for Power Mate–D/F, Series 21–TA), 1193
- Reading the current program number (low-speed response), 1262
- Reading the current screen number, 1242
- Reading the current sequence number, 1090
- Reading the current sequence number (low-speed response, 1263
- Reading the estimate disturbance torque data, 1189
- Reading the load current (A/D conversion data) for the spindle motor, 1293
- Reading the machine position (machine coordinates) of controlled axes, 1096, 1266
- Reading the machining time (low-speed response), 1292
- Reading the measured point, 1362
- Reading the parameter (*low-speed response), 1345
- Reading the relative position on a controlled axis, 1155, 1289
- Reading the remaining travel, 1157, 1290
- Reading the servo delay for controlled axes, 1100, 1268
- Reading the tool life management data (number of tool groups) (low-speed response), 1279
- Reading the tool life management data (tool group number) (lowspeed response), 1278
- Reading the tool life management data (tool life counter type) (not available for Power Mate–D/F, Series 21–TA), 1165
- Reading the tool offset data according to the specified tool number, 1294
- Reading the wire diameter offset, 1341
- Reading tool lefe management data (cutter compensation number 1) (low-speed type), 1297
- Reading tool life management data (cutter compensation No. (1): tool No.) (not available for Power Mate–D/F, Series 21–TA), 1129
- Reading tool life management data (cutter compensation No. (2): tool order No.) (not available for Power Mate–D/F, Series 21–TA), 1131
- Reading tool life management data (cutter compensation No.1) (lowspeed response), 1284
- Reading tool life management data (cutter compensation No.2) (lowspeed response), 1285
- Reading tool life management data (number of tool groups) (not available for Power Mate–D/F, Series 21–TA), 1117
- Reading tool life management data (number of tools) (low-speed response), 1279
- Reading tool life management data (number of tools) (not available for Power Mate–D/F, Series 21–TA), 1119
- Reading tool life management data (tool diameter offset number 1) (not available for Power Mate–D/F, Series 21–TA), 1199
- Reading tool life management data (tool group No.) (not available for Power Mate–D/F, Series 21–TA), 1115
- Reading tool life management data (tool group number) (not available for Power Mate–D/F, Series 21–TA), 1195
- Reading tool life management data (tool group numbers) (low-speed type), 1295
- Reading tool life management data (tool information (1) : tool No.) (not available for Power Mate–D/F, Series 21–TA), 1133
- Reading tool life management data (tool information (2): tool order No.) (not available for Power Mate–D/F, Series 21–TA), 1135

- Reading tool life management data (tool information 1) (low-speed response), 1286
- Reading tool life management data (tool information 1) (low-speed type), 1298
- Reading tool life management data (tool information 1) (not available for Power Mate–D/F, Series 21–TA), 1201
- Reading tool life management data (tool information 2) (low-speed response), 1287
- Reading tool life management data (tool length compensation No. (1): tool No.) (not available for Power Mate–D/F, Series 21–TA), 1125
- Reading tool life management data (tool length compensation No. (2): tool order No.) (not available for Power Mate–D/F, Series 21–TA), 1127
- Reading tool life management data (tool length compensation No.1) (low-speed response), 1282
- Reading tool life management data (tool length compensation number 1) (low-speed type), 1296
- Reading tool life management data (tool length offset number 1) (not available for Power Mate–D/F, Series 21–TA), 1197
- Reading tool life management data (tool life counter type) (low-speed response), 1281
- Reading tool life management data (tool life counter) (low-speed response), 1280
- Reading tool life management data (tool life counter) (not available for Power Mate–D/F, Series 21–TA), 1123
- Reading tool life management data (tool life) (low-speed response), 1280
- Reading tool life management data (tool life) (not available for Power Mate–D/F, Series 21–TA), 1121
- Reading tool life management data (tool No.) (not available for Power Mate–D/F, Series 21–TA), 1137
- Reading tool life management data (tool number) (low-speed response), 1287

Reading tool setting data by specifying tool number, 1372

- Reading value of the P-code macro variable (*low-speed response), 1161
- Registering the tool life management data (tool group) (*low-speed response) (not available for Power Mate–D/F, Series 21–TA), 1167
- Remove command file copy, 1046
- Rename command file attribute change, 1044

Repetitive operation, 43

- Response time, 395
- Restrictions, 385, 576
- ROM format program, 1007, 1014, 1017
- ROT (rotation control), 183
- ROTB (binary rotation control), 186

RST, 128

### ≪S≫

Scratch command – deletion of files, 1045 Screen configuration, 833, 840, 846 Screen display, 454

Screen for displaying general settings (GENERAL), 673

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

Screen for displaying setting data related to C language function, 677 Screen for displaying the setting data related to editing and debugging, 674 Screen for displaying traced data, 588 Screen for setting/displaying online monitor parameters (ONLINE), Screen of ladder debugging function, 589 Screen operation for PMC-SA1, 788 Screen operation for PMC-SA1/SB7, 787 Screen operation for PMC-SB7, 789 Screen operations, 834, 841, 847, 856 Screen Structure, 806 Screen Structures, 822 Search of sequence program, 528 Search of specified relay coil points in ladder diagram, 486 Searching a sequence program, 1029 Searching for an address (SRCH), 542 Selectable I/O link assignment function (PMC-SB7), 875 Selecting the PMC programmer menu, 610 Selection of program menu by soft keys, 1020 Selection of programmer menus by softkeys, 506 Sequence program, 41 Sequence program check and write into ROM (steps 8 to 11), 39 Sequence program copy function, 575 Sequence program creating procedure, 3 Sequence program entry (steps 6, 7), 39 Sequence program examples, 400 Sequence program generation (LADDER), 519 Sequence program input, 520, 1022 Sequence program memory capacity, 57 Sequence program processing time, 55 Sequence program structuring, 48, 369 SET, 126 Set items, 557 Setting, 393 Setting and display screen, 463 Setting and displaying system parameters (SYSTEM PARAM), 611 Setting function, 791 Setting I/O commands, 1035 Setting menu (SETING), 672 Setting of Ethernet parameters, 599, 891 Setting of I/O device, 979 Setting of online connection, 595, 889 Setting of online connection by NC parameter, 890 Setting of sampling address, 868 Setting of Trace Parameter, 641 Setting of trace parameter, 866

Setting parameter screen for the selectable I/O link assignment function. 796 SETTING screen, 472 Setting Screen, 828 Setting screen, 844, 857 Setting screen of general function, 793 Setting the screen, 812 Setting the transfer speed ([SPEED] soft key), 570 Setting/clearing OVERRIDE, 458 Setting/operation for enabling forced input/output, 453 SFT (shift register), 222 Signal address conversion (from the PMC-MODEL L/M to the PMC-MODEL SB/SC), 1375 Signal diagnosis screen, 438 Signal names, 364 Signal status display (STATUS), 638 Signal trace function (PMC-SB7), 864 Signal trace screen (initial screen), 865 Signals for connecting the operator's panel, 1383 Soft key menu of ladder debugging function, 590 Soft key-based PMC menu selection procedure, 630 Source program, 1005, 1010 SP (subprogram), 332 SPCNT (spindle control), 321 SPE (end of a subprogram), 333 Special uses of the R3 key, 1019 Specification, 936 Specification and display of system parameters (SYSPRM), 1393 Specification of PMCs, 5 Specification of step sequence, 935 Specifications, 373, 381, 434, 584 Specifications of password, 753 Specifying addresses, 1386 Specifying and displaying system parameters (SYSPRM), 508 Specifying and displaying title data (TITLE), 516 Specifying the number of the program for I/O Link, 1227 Start and stop of a sequence program, 552 Starting and stopping sequence programs, 783 Starting and stopping the on-line debugging function, 624 Starting and stopping the sequence program (RUN/STOP), 619 Starting ladder mnemonics editing, 613 Starting online communication by offline programmer (Ethernet connection), 600 Starting or stopping the trace function, 430 Starting the sequence program, 553 Step, 923 Step operation [STEP], 591 Step sequence basics, 908

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

Step sequence corresponded C language, 1445 Step sequence method, 902 Step sequence screen, 948, 958 Stop function of break with condition [BRKCTL], 592 Stop of ladder diagram display by trigger of signal, 487 Storage and control of sequence program (steps 12 to 14), 40 Storage to a memory card, 563 Storage to Flash ROM, 561 Storing the sequence program into flash EEPROM (I/O) (only for the Power Mate-H), 621 Structure of software key in PMC, 788, 789 Structure of valid net, 852 SUB (subtraction), 243 SUBB (binary subtraction), 245 Subprogram numbers (CALL, CALLU, SP) (P), 113 Subprogramming and nesting, 374 Substitution of sequence programs, 1025 Summary of specification of ladder program, 28 Switch statement, 1454 Switching screens for setting parameters, 792 Symbol and comment display, 485 Symbol data and comment input, 538 Symbol data display, 1033 Symbol data search (SRCH), 538 Symbol data setting (SYMBOL), 537 Symbols used in the ladder diagram, 365 System alarm messages (C language function for PMC-NB/ NB2/NB6), 1436 System alarm messages (PMC-SB5/SB6/NB6), 1437 System alarm messages (PMC-SC), 1434 System diagram of soft key, 1395 System floppy, 984 System parameter (PMC-SB7), 884

### ≪**T**≫

Terminology, 909 The value of functional instruction parameter, 483 Time screen, 953 Timer addresses (T), 110 Timer screen, 953 Timer screen (TIMER), 463, 666 Timer set time address, 165 Timer, counter, keep relay, nonvolatile memory control, data table, 354 Title data display (TITLE), 636 Title data editor screen, 637 TMR (timer), 160 TMRB (fixed timer), 162 TMRC (timer), 164 Trace function (TRACE), 429 Trace screen, 431, 640 Transfer between data area and non–volatile memory, 1332 Transfer to and from a FANUC FD Cassette, 559 Transfer to and from a FAPT LADDER, 559 Transfer to and from a ROM WRITER, 571 Transition, 926

### *≪U*≫

User PMC Screen (PMCMDI), 492

### ≪W≫

What is a sequence program?, 35 While statement, 1446 Window function description (FS16-PA), 1365 Window function description (FS16-W), 1340 Window function description (FS16/16i-LA), 1330 Window function description (PMC-NB/NB2/NB6), 1249 Window function description (PMC-PA1/PA3/SA1/SA2/SA3/SA5/SB/SB2/SB3/SB4/SB5/SB6/SC/ SC3/SC4), 1054 WINDR (reading CNC window data), 288 WINDW (writing CNC window data), 291 Writing (registering) tool life management data (tool group number) (*low-speed response) (not available for Power Mate-D/F, Series 21–TA), 1203 Writing a custom macro variable, 1312 Writing a custom macro variable (*low-speed response), 1081 Writing a data on the program check screen, 1313 Writing a parameter (*low-speed response), 1072 Writing a parameter (setting data), 1311 Writing a tool offset (*Low-speed response), 1064 Writing a tool offset data, 1310 Writing a workpiece origin offset value (*low-speed response) (not supported by the Power Mate–D or –F), 1068 Writing of tool setting data (low-speed response), 1370 Writing real parameters (low-speed type), 1329 Writing setting data (*low-speed response), 1076 Writing setting data (low-speed type), 1353 Writing the feedrate, 1324 Writing the measured point (*low-speed response), 1364 Writing the parameter (*low-speed response), 1347 Writing the superposition move command, 1322 Writing the tool life management data (cutter compensation number (1) : tool number) (*low-speed response) (not available for Power Mate-D/F, Series 21-TA), 1179

### Volume 1 : Up to Page 898 / Volume 2 : Page 899 and later

- Writing the tool life management data (cutter compensation number (2) : tool operation sequence number) (*low–speed response) (not available for Power Mate–D/F, Series 21–TA), 1181
- Writing the tool life management data (cutter compensation number 1), 1318
- Writing the tool life management data (cutter compensation number 2), 1318
- Writing the tool life management data (tool condition (1) : tool number) (*low-speed response) (not available for Power Mate–D/F, Series 21–TA), 1183
- Writing the tool life management data (tool group number), 1315
- Writing the tool life management data (tool information 1), 1319
- Writing the tool life management data (tool information 2), 1319
- Writing the tool life management data (tool length compensation number 1), 1317
- Writing the tool life management data (tool length compensation number 2), 1317
- Writing the tool life management data (tool length offset number (1) : tool number) (*low-speed response) (not available for Power Mate-D/F, Series 21-TA), 1175
- Writing the tool life management data (tool length offset number (2) : tool operation sequence number) (*low-speed response) (not available for Power Mate–D/F, Series 21–TA), 1177
- Writing the tool life management data (tool life counter type), 1316
- Writing the tool life management data (tool life counter type) (*lowspeed response) (not available for Power Mate–D/F, Series 21–TA), 1173
- Writing the tool life management data (tool life counter), 1316
- Writing the tool life management data (tool life counter) (*low– speed response) (not available for Power Mate–D/F, Series 21–TA), 1171
- Writing the tool life management data (tool life), 1315
- Writing the tool life management data (tool life) (*low-speed response) (not available for Power Mate-D/F, Series 21-TA), 1169
- Writing the tool life management data (tool number), 1320
- Writing the tool life management data (tool number) (*low-speed response) (not available for Power Mate–D/F, Series 21–TA), 1187

- Writing the tool management data (tool condition (2) : tool operation sequence number) (*low-speed response) (not available for Power Mate–D/F, Series 21–TA), 1185
- Writing the tool offset data according to the specified tool number, 1321
- Writing the torque limit override, 1314
- Writing the wire diameter offset (*low-speed response), 1343
- Writing tool life management data (cutter compensation number 1), 1327
- Writing tool life management data (tool group numbers), 1325
- Writing tool life management data (tool information 1), 1328
- Writing tool life management data (tool information 1) (*low-speed response) (not available for Power Mate-D/F, Series 210TA), 1210
- Writing tool life management data (tool length compensation number 1), 1326
- Writing tool life management data (tool length offset number 1) (*low-speed response) (not available for Power Mate–D/F, Series 21–TA), 1206
- Writing tool life management data (tool radius offset number 1) (*low-speed response) (not available for Power Mate–D/F, Series 21–TA), 1208
- Writing value of the P–code macro variable (*low–speed response),  $1163\,$
- Writing, reading, and comparing sequence programs and PMC parameters, 756
- Writing, reading, and verifying the sequence program and PMC parameter data, 554

WRT, 120

WRT. NOT, 121

### ≪X≫

XMOV (indexed data transfer), 229

XMOVB (binary index modifier data transfer), 232

**Revision Record** 

# FANUC PMC-MODEL PA1/PA3/SA1/SA2/SA3/SA5/SB/SB2/SB3/SB4/SB5/SB6/SC/SC3/SC4/NB/NB2/NB6

# LADDER LANGUAGE PROGRAMMING MANUAL (B-61863E)

07	Apr.,'95	Total revision	14	Jul., 2001	Addition of PMC-SB7
90	Nov.,'94	Corresponds to 18–B	5	May, 2000	<ul> <li>Addition of ladder display/editing function for PMC-NB6</li> <li>Addition of expansion of I/O Link points for FS 16i/18i/21i</li> <li>Addition of window functions (tool life management B function and so forth)</li> <li>Correction of errors</li> </ul>
05	May,'94	<ul> <li>Addition of PMC-MODEL RB4/RC4</li> <li>Addition of the following Appendix.</li> <li>Window function description</li> <li>Window function description</li> <li>Window function description</li> <li>FS16-W)</li> <li>Window function description</li> <li>FS16PA)</li> <li>PMC MODEL RA1/RA3 Supplementary Explanation of Programming</li> </ul>	12	Mar., '99	Addition of PMC–NB6
04	Aug.,'93	Addition of PMC-MODEL PA1/PA3/RB3/RC3/NB.	1	Nov., '98	Addition of Power Mate <i>i</i>
03	Mar., '92	Addition of PMC-MODEL RA1/RA2/RB2	10	Dec., '97	Corresponds to 16i/18i/21i–MODEL A
02	Aug., '91	All pages are revised. PMC-MODEL RC is added.	60	Mar.,'96	Addition of PMC–NB2
01	Oct., '90		08	Oct.,'95	Addition of PMC-RB5/RB6
Edition	Date	Contents	Edition	Date	Contents

- No part of this manual may be reproduced in any form.
- All specifications and designs are subject to change without notice.